

Інститут (факультет) автоматизації і комп'ютерних систем
Кафедра інформаційних систем

« » _____ 20 р.

« ____ » _____ 20__ р.

Київ - 2020р.

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ХАРЧОВИХ ТЕХНОЛОГІЙ

Інститут (факультет) автоматизації і комп'ютерних систем

Кафедра інформаційних систем

Освітній ступінь бакалавр

Спеціальність 122 Комп'ютерні науки та інформаційні технології

(код і назва)

Освітньо-професійна програма Комп'ютерні науки

(назва)

ЗАТВЕРДЖУЮ

Завідувач

кафедри _____

“ _____ ” _____ 20 ____ року

З А В Д А Н Н Я

НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧА

Кудрицькому Денису Романовичу

(прізвище, ім'я, по батькові)

1. Тема роботи Створення підсистеми обробки заявок на ремонт та обслуговування програмно-технічного комплексу для відділу інформаційних технологій ПАТ «Христинівський молокозавод»

керівник роботи Савченко Юлій Григорович, професор, доктор технічних наук

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

затверджені наказом закладу вищої освіти від “27” квітня 2020 року № 269-кв

2. Строк подання здобувачем роботи 23 травня 2020

3. Вихідні дані до роботи Організаційна структура підприємства, структура відділу ІТ, інформація про виконання процесу обробки заявок, вимоги керівництва до створеної системи, інформація про ПЗ, яке використовується у відділі ІТ, інформація про охорону праці на підприємстві

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити) _____

Системний аналіз підприємства та постановка задачі на проектування

Розробка логічної та фізичної моделей бази даних в середовищі ERwin

Генерація бази даних в середовищі MS SQL Server

Проектування інтерфейсу користувача в середовищі Visual Studio 2019

5. Перелік графічного матеріалу _____

Організаційна структура підприємства, структура відділу ІТ, ілюстрації по створенню інтерфейсу користувача, інструкція користувача, функціональна моделі, концептуальна модель, логічна модель даних, фізична модель даних, схема бази даних в СУБД MS SQL Server

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1	Савченко Ю.Г. професор, доктор технічних наук		
2	Савченко Ю.Г. професор, доктор технічних наук		
3	Савченко Ю.Г. професор, доктор технічних наук		

7. Дата видачі завдання 28 квітня 2020

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів виконання кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Системний аналіз підприємства	28.04. – 01.05.	виконано
2	Постановка задачі на проектування	01.05. – 02.05.	виконано
3	Розробка логічної та фізичної моделей бази даних	02.05. – 04.05.	виконано
4	Генерація бази даних в середовище MS SQL Server	05.05. – 06.05.	виконано
5	Розробка інтерфейсу користувача	06.05. – 13.05.	виконано
6	Оформлення пояснювальної записки	13.05. – 21.05.	виконано
7	Оформлення презентації	21.05. – 22.05.	виконано

Здобувач

(підпис)

Кудрицький Д. Р.

(прізвище та ініціали)

Керівник роботи

(підпис)

Савченко Ю. Г.

(прізвище та ініціали)

АНОТАЦІЯ

Кваліфікаційна робота викладена на 71 сторінці, вона містить 43 рисунки, 1 таблицю, 5 формул та 27 використаних джерел.

Завданням кваліфікаційної роботи є закріплення навичок використання вивчених технологій на прикладі створення інформаційної підсистеми для харчового підприємства, згідно прийнятих міжнародних стандартів, відповідно до індивідуального завдання.

Об'єктом розробки є підсистема обробки заявок на ремонт та обслуговування програмно-технічного комплексу для відділу інформаційних технологій ПАТ «Христинівський молокозавод».

В першому розділі роботи проведений системний аналіз об'єкту автоматизації та створено постановку задачі на проектування. У другому розділі описано процес розробки ІС: проектування БД, створення інтерфейсу користувача, створено інструкцію використання ІС для користувачів та проведено техніко-економічне обґрунтування розробки ІС. В останньому розділі були розглянуті небезпечні для життя та здоров'я працівників чинники, можливі наслідки, причини виникнення та заходи їх передбачення.

Ключові слова: ІНФОРМАЦІЙНА СИСТЕМА, КОРИСТУВАЧ, БАЗА ДАНИХ, ІНТЕРФЕЙС, РОЗРОБКА, ОБРОБКА ЗАЯВОК, ПРОБЛЕМА, ВІДДІЛ ІТ.

ABSTRACT

The qualifying work is presented on 71 pages, it contains 43 figures, 1 table, 5 formulas and 27 sources used.

The task of the qualification work is to consolidate the skills of using the studied technologies on the example of creating an information subsystem for a food company, according to accepted international standards, in accordance with the individual task.

The object of development is a subsystem for processing applications for repair and maintenance of software and hardware for the information technology department of PJSC "Khrystynivsky Dairy".

In the first section of the work the system analysis of the object of automation is carried out and the statement of the problem on designing is created. The second section describes the process of IP development: database design, user interface creation, created instructions for the use of IP for users and feasibility study of IP development. The last section considered the factors dangerous to the life and health of workers, the possible consequences, causes and measures to prevision them.

Keywords: INFORMATION SYSTEM, USER, DATABASE, INTERFACE, DEVELOPMENT, APPLICATION PROCESSING, PROBLEM, IT DEPARTMENT.

ЗМІСТ

ВСТУП	8
РОЗДІЛ 1. СИСТЕМНИЙ АНАЛІЗ ПАТ «ХРИСТИНІВСЬКИЙ МОЛОКОЗАВОД» ТА ПОСТАНОВКА ЗАДАЧІ НА ПРОЕКТУВАННЯ...	9
1.1. Загальна характеристика ПАТ «Христинівський молокозавод»	9
1.2. Організаційна структура ПАТ «Христинівський молокозавод»	9
1.2.1. Загальна схема організаційної структури	10
1.2.2. Схема структури відділу	11
1.2.3. Взаємодія з іншими відділами.....	12
1.3. Аналіз нинішнього стану автоматизації.....	12
1.4. Розроблення функціональної моделі існуючих бізнес-процесів «AS-IS»	13
1.4.1. Опис діаграм процесів.....	13
1.4.2. Виявлені проблеми	15
1.4.3. Задачі автоматизації	15
1.5. Аналіз існуючих систем - аналогів	16
1.5.1. Відкрита система обробки заявок OTRS.....	16
1.5.2. Підсистема «Технічна підтримка» на базі «1С: Підприємство»	16
1.5.3. Порівняння систем-аналогів.....	17
1.6. Обґрунтування доцільності проектування й розроблення «Системи обробки заявок для ІТ – відділу ПАТ «Христинівський молокозавод».....	19
1.7. Концептуальна модель системи	19
1.8. Постановка задачі	20
1.8.1. Призначення та цілі створення системи.....	20
1.8.2. Вимоги до створюваної системи.....	20
1.8.3. Функції, які має виконувати система	21
1.8.4. Вхідні та вихідні дані системи	21
РОЗДІЛ 2. ОПИС КОМПЛЕКСУ ЗАДАЧ АВТОМАТИЗАЦІЇ.....	22
2.1. Перелік використаних технологій	22
2.2. Проектування БД	22
2.3. Створення інтерфейсу користувача.....	24
2.4. Інструкція користувача	35
2.4.1. Інструкція для працівників не ІТ-відділу.....	35
2.4.2. Інструкція для працівників ІТ-відділу.....	37
2.4.3. Загальна інструкція для ситуації, коли користувач забув пароль ...	40

2.5. Техніко-економічне обґрунтування	42
РОЗДІЛ 3. ОХОРОНА ПРАЦІ	45
ВИСНОВКИ.....	49
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	50
ДОДАТКИ.....	53
Додаток А. Функціональні моделі.....	53
Додаток Б. Розробка БД.....	58
Додаток В. Скріншоти розробки та інструкції користувача	61
Додаток Г. Фрагменти коду програми	67

ВСТУП

Обробка та облік заявок відома проблема, з якою на певних етапах розвитку зіштовхуються підприємства [1]. Це відбувається тому, що кількість проблем, які потребують вирішення зростає, а отже і зростає кількість заявок, а вони вже потребують обробки та обліку. На багатьох підприємствах, які працюють вже доволі довго облік заявок або не відбувається взагалі, або відбувається за допомогою, наприклад, телефону та журналу обліку. При цьому заявки обробляються і фіксуються вручну однією людиною – диспетчером.

З розвитком технологій почали з'являтися нові способи обробки заявок, і якщо підприємству це потрібно, можна обрати спосіб або систему обробки заявок, яка буде підходити конкретному підприємству.

В даній роботі буде розроблено систему обробки заявок на ремонт та обслуговування програмно-технічного комплексу для відділу інформаційних технологій ПАТ «Христинівський молокозавод». Система буде використовуватись виключно на даному підприємстві, оскільки вона розробляється конкретно для нього. Вона допоможе обробляти та облікувати всі створені заявки: всі дані будуть збережені, і в разі потреби їх можна використовувати.

РОЗДІЛ 1. СИСТЕМНИЙ АНАЛІЗ ПАТ «ХРИСТИНІВСЬКИЙ МОЛОКОЗАВОД» ТА ПОСТАНОВКА ЗАДАЧІ НА ПРОЕКТУВАННЯ

1.1. Загальна характеристика ПАТ «Христинівський молокозавод»

Христинівський молокозавод був створений в 1999 році [2]. Зараз вид економічної діяльності підприємства – виробництво сиру та масла з переробленого молока, на початковому етапі розвитку до економічної діяльності входила гуртова торгівля: харчовими жирами та оліями, молочними продуктами.

Підприємство входить до групи компаній «Агрополіс», яка має багаторічний досвід роботи в області молочної продукції [3]. Велика увага приділяється вибору якісної сировини та технологіям її обробки. Через це молокозавод завжди виготовляв продукцію високої якості, яка на ринку України мала високі показники, а в певний період ця продукція експортувалась в країни ЄС.

Проте, незважаючи на успішність Христинівського молокозаводу в минулому, зараз його показники спали: підприємство функціонує не на повну силу через ряд різних проблем і причин [4].

1.2. Організаційна структура ПАТ «Христинівський молокозавод»

1.2.1. Загальна схема організаційної структури

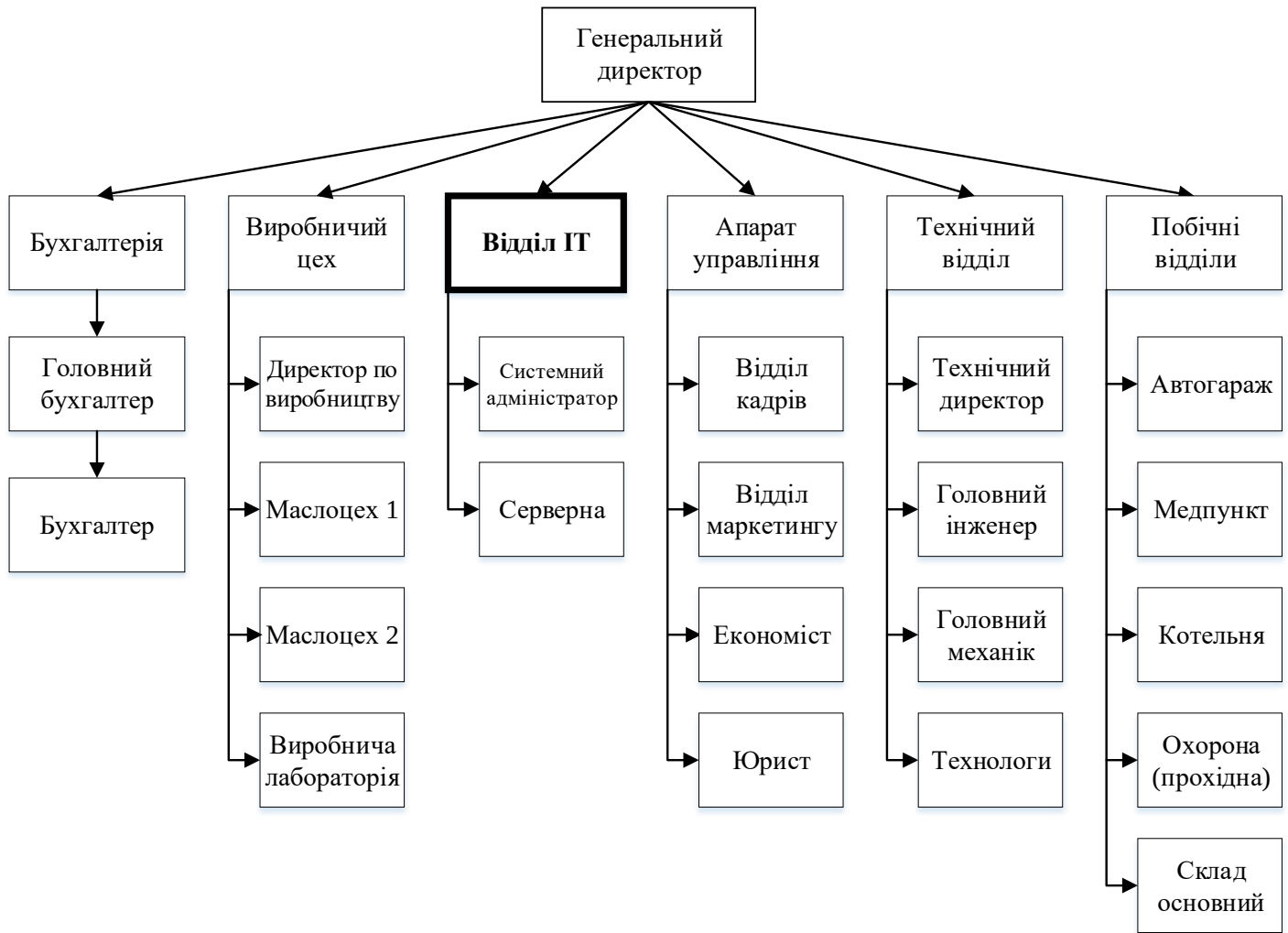


Рис.1.1. Загальна схема організаційної структури ПАТ "Христинівський молокозавод"

Усі підрозділи підприємства поділяються на: основний виробничий підрозділ, підрозділи, що забезпечують функціонування виробничого та допоміжні підрозділи для роботи всього підприємства. До підрозділів, що забезпечують функціонування виробничого підрозділу відносяться: ІТ відділ та технічний відділ. До допоміжних, а тобто таких, що забезпечують функціонування всього підприємства відносяться: бухгалтерія, відділ кадрів, відділ маркетингу, охорона, склади, котельня тощо. Кількість працівників молокозаводу – близько 60. Керує цим всім генеральний директор, у нього є наближенні помічники. Вони відповідають за питання, пов’язані в основному з заводом в цілому. Після них йде апарат управління, який відповідає за питання, пов’язані з підрозділами. Потім, у кожного підрозділу є свій начальник, який відповідає за контрений відділ або відділи

і приймає рішення з питань, які відносяться до роботи його підпорядкованих підрозділів.

1.2.2. Схема структури відділу

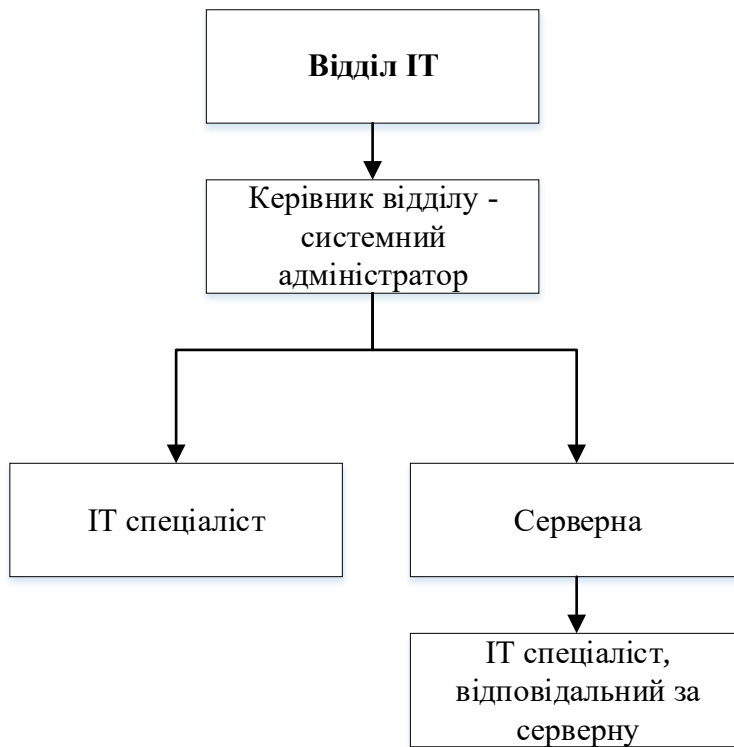


Рис.1.2. Схема структури ІТ відділу ПАТ "Христинівський молокозавод"

ІТ відділ молокозаводу виконує такі функції:

- Впроваджує інформаційні системи, програмне забезпечення та комп'ютерну техніку на підприємстві;
- забезпечує роботу і підтримку інформаційних мереж та мережевого обладнання передачі даних;
- виконує ремонт та обслуговування програмно-технічного комплексу всього підприємства;
- допомагає структурним підрозділам вирішувати питання пов'язані з технікою чи програмним забезпеченням;
- забезпечує стабільну роботу локальної мережі тощо.

Керівник ІТ відділу виконує всю основну і важливу роботу, так як рівень його знань і досвіду більший, ніж в інших працівників відділу. ІТ спеціалісти виконують роботу, яка на них покладена: робітник серверної повинен забезпечувати безперебійну роботу серверів. Також він разом з іншим ІТ спеціалістом допомагає іншим підрозділам розв'язувати питання, виконує ремонт та обслуговування програмно-технічного комплексу.

1.2.3. Взаємодія з іншими відділами

Відділ інформаційних технологій є найважливішим відділом, тому що він забезпечує роботу як і основного виробничого відділу, так і допомагає в роботі деяким іншим відділам підприємства.

В першу чергу відділ ІТ забезпечує роботу автоматизованих систем управління технологічними процесами на виробництві. Тобто за допомогою технічних і програмних засобів здійснюється передача і обробка даних між ІТ відділом і устаткуванням у виробничому відділі.

Так як ІТ відділ відповідає за програмно – технічний комплекс підприємства, то він завжди отримує інформацію від інших відділів з приводу проблем, які пов'язані з функціонуванням програмного забезпечення чи технічних засобів, оскільки ремонт та обслуговування цього – обов'язок робітників відділу інформаційних технологій. Тобто, до відділу надходить інформація про проблему, а відділ ІТ повинен вирішити проблему або надати потрібну інформацію.

1.3. Аналіз нинішнього стану автоматизації

Зараз до ІТ відділу надходять заявки, але це відбувається за допомогою телефонних дзвінків і ці заявки ніяк не фіксуються. Тільки при заміні якоїсь техніки це вноситься вручну в журнал, та коли заявка була виконана вноситься запис в журнал виконаних заявок без додаткової інформації, лише дата. Тому повну інформацію про заявки не можна ніяк

отримати і тим більше обробити. Ці дані були б дуже корисні, тому що в разі виникнення питань чи проблем до них можна було б звернутись і перевірити.

Тому можна сказати, що нинішній стан автоматизації процесу обробки заявок у відділі інформаційних технологій підприємства є незадовільним. На сьогоднішній день цей процес не автоматизований взагалі. Ця автоматизація дозволить підвищити ефективність роботи ІТ відділу у вирішенні питань, пов'язаних з технікою чи програмним забезпеченням та їх обслуговуванням, та дозволить зберігати та обробляти всю інформацію про заявки.

1.4. Розроблення функціональної моделі існуючих бізнес-процесів «AS-IS»

1.4.1. Опис діаграм процесів

Функціональну модель наведено на рис. 1-3 у додатку А.

Розроблена функціональна модель описує процес обробки заявок у відділі інформаційних технологій ПАТ "Христинівський молокозавод" в такому вигляді, як цей процес відбувається зараз [5, с 6].

Кількість рівнів декомпозиції функціональної моделі – два.

До першого рівня декомпозиції (рис. 2 у додатку А) входить три блоки:

- Прийом дзвінка з проблемою. Цей блок призначений для відображення процесу прийому заявок в ІТ відділу шляхом телефонного дзвінка. На вхід цього блоку йде стрілка «Заявка на ремонт або обслуговування», тобто це сама заявка, з якої починається процес обробки у ІТ відділі. Стрілка управління у цьому блоці – «Посадові інструкції», яка містить інформації про те, хто і як повинен приймати заявку у ІТ відділі. Механізмом є «Персонал відділу», на який покладено обов'язок приймати заявки. На виході маємо стрілку «Визначена проблема», яка означає, що

після прийому заявки працівник відділу зрозумів в чому полягає поставлена проблема.

- Виконання заявки. Вхідні стрілки: вихідна із першого блоку «Визначена проблема», «Заявка на ремонт або обслуговування», яку безпосередньо треба виконати та «Програмно - технічний комплекс», тобто дані про техніку та програмне забезпечення, які перебувають на балансі підприємства. Механізм – стрілка «Персонал відділу». До управління входять такі стрілки: «Посадові інструкції», «Документація на ПЗ та техніку» та «Накази та розпорядження». Вихідні стрілки: «Виконані заявки» та «Журнал виконаних заявок».

- Занесення даних до журналів. Вхід – «Виконані заявки». Стрілки управління цього блоку - «Посадові інструкції», «Документація на ПЗ та техніку» та «Накази та розпорядження». Механізм - стрілка «Персонал відділу». На виході стрілки: «Журнал виконаних заявок», «Інвентарний журнал».

Декомпозиція блоку «Виконання заявки» - другий рівень.

До другого рівня декомпозиції (рис. 3 у додатку А) також входить три блоки:

- Визначення конкретного завдання. Він відображає процес, коли працівник відділу сформував конкретне завдання, вирішення якого і є вирішенням проблеми, тобто виконанням заявки. Вхідні стрілки: «Визначена проблема», «Заявки на ремонт або обслуговування». Вихідна стрілка: «Визначене завдання». Стрілка механізму – так, як і усіх попередніх блоках – «Персонал відділу». Стрілки управління – такі ж самі, як і у всіх попереднім блоках.

- Створення алгоритму розв'язання проблеми. З попереднього блоку в цей надходить стрілка «Визначене завдання», тобто цей блок призначений для створення плану дій, які розв'яжуть потрібне завдання. З нього виходить стрілка «Алгоритм вирішення».

- Вирішення поставленої проблеми. З попереднього блоку входить стрілка «Алгоритм вирішення», тому в цьому блоці відбувається

процес розв'язання завдання, тобто виконання заявки, яка надійшла до відділу інформаційних технологій. З цього блоку виходять стрілки: «Виконані заявки» та «Журнал виконаних заявок». Тобто після виконання заявки вона вручну вноситься в журнал.

1.4.2. Виявлені проблеми

Описавши існуючі процеси, які пов'язані з обробкою заявок у відділі ІТ та проаналізувавши їх, можна виокремити такі проблеми:

1. Заявки надходять у вигляді телефонних дзвінків;
2. детальна інформація про заявки ніяк не зберігається;
3. у разі відсутності працівника ІТ відділу на робочому місці, заявка не буде прийнята;
4. паперовий журнал – єдине місце, де зберігається хоч якась інформація про заявки;
5. єдиний механізм у процесі обробки заявок зараз – працівники відділу інформаційних технологій.

1.4.3. Задачі автоматизації

1. Створення і обробка заявок за допомогою ІС;
2. збереження інформації про заявки в базі даних;
3. користувачі, які створюють заявки повинні мати можливість перегляду всіх своїх заявок;
4. працівники ІТ-відділу повинні мати можливість переглядати і змінювати всю інформація, яка пов'язана з заявками;
5. працівникам ІТ-відділу треба забезпечити можливість переглядати звіти по заявках.

1.5. Аналіз існуючих систем - аналогів

1.5.1. Відкрита система обробки заявок OTRS

Open source Ticket Request System (OTRS) – одна з найвідоміших систем обробки заявок [6]. З її допомогою можна вирішувати такі завдання: управління різними типами заявок та відповідями на них, організація роботи технічної підтримки підприємства, впровадження на підприємстві корпоративних IT – сервісів за стандартом ITIL [7]. Ця система функціонує вже майже 20 років, за які добре себе зарекомендувала. На даний момент актуальна версія – OTRS 8.

Впроваджена система поєднує основні компоненти управління підприємством на одній платформі: обробка заявок, автоматизовані IT - процеси тощо. Це скорочує час необхідний для розв'язання різних питань та проблем. OTRS також має можливість надання експертних консультацій, які забезпечують підтримку компанії, з метою максимізації використання функцій їх системи для конкретного підприємства [8].

1.5.2. Підсистема «Технічна підтримка» на базі «1С: Підприємство»

Дана підсистема дозволяє проводити обробку і облік заявок, які пов'язані з роботою IT відділу підприємства або службою технічної підтримки. Так як і OTRS, ця система працює за стандартами ITIL [9]. Так, як «Технічна підтримка» є підсистемою, то вона розширює функціональні можливості системи «Корпоративний документообіг», яка входить до пакету «1С:Підприємство». Це розширення допомагає виконувати наступні функції :

- обробка заявки і відслідковування її виконання з можливістю сповіщення користувачів про це;
- облік техніки, обладнання та проблем, які з ними виникали;
- облік ліцензій на ПЗ та гарантійних термінів експлуатації техніки;
- взаємодія з користувачами використовуючи електронну пошту;

- можливість редагування бізнес-процесу обробки заявки

За бажанням цю підсистему можна доповнювати і розширювати відповідно до потреб конкретного підприємства [10].

1.5.3. Порівняння систем-аналогів

Перелік параметрів для порівняння систем-аналогів:

- простота. Це означає, що система є простою у використанні і не перенавантаженою непотрібними для даного підприємства функціями;
- вартість. Цей параметр показує вартість системи для даного підприємства;
- можливість зберігати інформацію в базі даних SQL Server;
- доступна на Windows;
- потребує доступу до мережі Internet;
- підтримка української або російської мов;
- невисокі системні вимоги;
- можливість змінювати дані про заявки;
- можливість перегляду звітів по заявках та виконаних заявках.

Результати порівняння параметрів цих двох систем-аналогів наведені нижче у табл. 1.1.

Наявність або відсутність якогось із параметрів в системі буде позначатись «+» та «-». Параметр «Вартість» буде виражатись в грошовій одиниці – гривні.

Таблиця 1.1

Результати порівняння систем-аналогів

Параметр для порівняння	Система – аналог	
	OTRS	Технічна підтримка
Простота	-	-
Вартість	0 або 133700 грн. на рік	62700 грн.
Інтеграція з SQL Server	+	+

Продовження табл. 1.1.

Доступність на Windows	-	+
Потребує доступу до Internet	-	-
Підтримка української або російської мов	+	+
Невисокі системні вимоги	-	+
Можливість змінювати дані про заявки	+	-
Перегляд звітів по заявках	+	+

Вартість системи OTRS залежить від версії: безкоштовна та бізнес. Можна використовувати безкоштовну версію, але в ній не буде можливості налаштування окремих функцій і системи в цілому під конкретне підприємство, однак ця можливість є в бізнес версії, яка обійдеться компанії більше ніж в 133000 гривень на рік [11].

Отже, із таблиці порівняння видно, що система «Технічна підтримка» є занадто дорогою для підприємства, вона працює лише на основі системи «Корпоративний документообіг», яка входить до пакету «1С:Підприємство», через що ціна така висока [12]. Для підприємства система OTRS цікава лише в бізнес версії, оскільки тільки її можна буде налаштувати конкретно під наше підприємство, а ціна цієї бізнес версії майже в два рази перевищує ціну системи «Технічна підтримка», що зовсім нам не підходить. Якщо брати до уваги безкоштовні системи, які не були розглянуті при порівнянні, то вони не зможуть розв'язати всі задачі та виконувати всі функції, які є нам необхідними.

Таким чином, впровадження та використання цих систем у відділі інформаційних технологій «Христинівського молокозаводу» є недоцільним.

1.6. Обґрунтування доцільності проектування й розроблення «Системи обробки заявок для ІТ – відділу ПАТ «Христинівський молокозавод»

Відповідно до п. 1.3, у відділі ІТ не встановлено інформаційної системи для автоматизації процесу обробки заявок від інших відділів. У п. 1.5 було порівняно існуючі на ринку системи-аналоги, з результатів цього порівняння можна побачити, що розглянуті системи занадто дорогі для цього підприємства. Ціна була одним із найпріоритетніших параметрів вибору системи, оскільки підприємство не має можливості віддавати великі гроші за рішення такої проблеми.

Тому було вирішено розробити свою підсистему обробки заявок для ІТ-відділу молокозаводу, яка повинна реалізувати задачі та вирішити проблеми, описані в п. 1.4.

1.7. Концептуальна модель системи

Концептуальну модель наведено на рис. 4-5 у додатку А.

У моделі ТО-ВЕ з'являються нові блоки процесів [13]: блок «Прийом дзвінка з проблемою (заявка)» функціональної моделі змінюється на блок «Розгляд заявки», блок «Занесення даних до журналів» замінюється блоком «Внесення або редагування даних до ІС», тим самим зникають паперові журнали, також додається блок «Формування звітності». З'являються нові механізми: інформаційна система та ПК. Додається новий вид документів – звіти та нова інформація: інвентарні відомості та акти виконаних заявок.

Додавання нового механізму, тобто впровадження інформаційної системи допоможе усунути виявлені проблеми та реалізувати поставлені задачі автоматизації: ніяк не автоматизований процес обробки заявок буде відбуватись за допомогою інформаційної системи, яка буде виконувати усі необхідні функції.

1.8. Постановка задачі

1.8.1. Призначення та цілі створення системи

Користувачами системи будуть як працівники ІТ-відділу, так і працівники інших відділів молокозаводу. Оскільки це система обробки заявок в ІТ відділі, а заявки до нього надходять від всього підприємства, тому працівники інших відділів за допомогою цієї ІС будуть створювати заявки, а працівники ІТ-відділу будуть бачити ці заявки та виконувати їх. Тому буде передбачено розмежування доступу до системи і користувачі матимуть різні права.

Система повинна підвищити ефективність роботи ІТ-відділу у питанні обробки заявок від співробітників. Вона дасть можливість створювати заявки, обробляти, зберігати та використовувати дані про ці заявки, що з теперішніми умовами робити не можливо.

1.8.2. Вимоги до створюваної системи

Вимоги до технічного забезпечення: RAM не менше 2 ГБ, процесор з частотою не нижче 1.6 GHz.

Вимоги щодо операційної системи: система повинна працювати на ОС Windows версії 7 і вище.

Вимоги до СУБД: система повинна бути сумісною з СУБД Microsoft SQL Server версії 8 і вище.

Вимоги до інтерфейсу користувача: інтерфейс повинен бути не навантаженим та інтуїтивно зрозумілим. Усі елементи взаємодії користувача з системою повинні бути підписані або виглядати відповідно до загальноприйнятих норм. Кольорова гама не повинна бути перенавантаженою, а читабельність тексту повинна бути високою [14].

Вимоги до захисту даних: повинно бути дві групи користувачів з різними правами доступу. Перша група – співробітники інших відділів, які повинні мати змогу створювати заявки та переглядати всі свої заявки. Друга група – працівники ІТ-відділу, які повинні мати розширені можливості роботи з даними.

1.8.3. Функції, які має виконувати система

1. автентифікація та авторизація користувачів в системі;
2. створення заявок до ІТ-відділу;
3. збереження інформації в базі даних;
4. перегляд даних по своїх заявках для працівників не ІТ-відділу;
5. перегляд, змінювання і додавання інформації до всіх таблиць бази даних для працівників ІТ-відділу;
6. можливість переглядати звіти по заявках для працівників ІТ-відділу.

1.8.4. Вхідні та вихідні дані системи

Вхідні дані:

- заявки на ремонт або обслуговування;
- перелік програмно-технічного комплексу підприємства;
- дані про співробітників та відділи.

Вихідні дані:

- акти виконаних заявок;
- інвентарні відомості;
- звіти;
- виконані заявки.

РОЗДІЛ 2. ОПИС КОМПЛЕКСУ ЗАДАЧ АВТОМАТИЗАЦІЇ

Оскільки проектується підсистема, необхідним є вказати те, що на підприємстві, але не в ІТ-відділі функціонує система 1С:Бухгалтерія, до якої підключена функціонуюча база даних. При впровадженні системи, яка розробляється, буде здійснено злиття баз даних: існуючої на підприємстві та розробленої з метою створення однієї спільної, яка дасть змогу обміну даними між цими системами та «об'єднання» їх в одну на підставі використання спільних даних. Пізніше можливе створення та підключення інших підсистем, які зможуть стати однією, використовуючи спільні дані.

2.1. Перелік використаних технологій

Логічна та фізична моделі даних розроблялись за допомогою програмного засобу AllFusion ERWin Data Modeler.

В якості СУБД використовувалась MS SQL Server 2008, оскільки саме вона використовується на підприємстві для зберігання даних.

Для створення інтерфейсу користувача використовувалось інтегроване середовище розробки Microsoft Visual Studio 2019.

2.2. Проектування БД

Для початку створюємо логічну модель даних на рівні визначень. Логічна модель наведена на рис. 6 в додатку Б. Потім на основі створеної логічної моделі створюємо фізичну модель на рівні атрибутів. Фізична модель наведена на рис. 7 додатку Б. Переходимо до Середовища SQL Server Management Studio, підключаємось до SQL Server 2008, створюємо порожню базу даних, в яку з ERWin Data Modeler буде згенерована готова база даних. Після цього на основі фізичної моделі генеруємо базу даних в MS SQL Server 2008, встановивши відповідні налаштування [5]. Схему згенерованої бази даних представлено на рис. 8 в додатку Б.

База даних складається з 8 таблиць. Далі буде наведено опис та призначення усіх таблиць.

Таблиця «Співробітник» зберігає дані про всіх співробітників підприємства та має такі поля:

- Код співробітника – первинний ключ, тип даних – integer;
- ПІБ – varchar(50);
- E-mail – varchar(50);
- Номер телефону – varchar(20);
- Посада – varchar(50);
- Код відділу - зовнішній ключ, тип даних – integer.

Таблиця «Відділ» - перелік відділів підприємства, має такі поля:

- Код відділу – первинний ключ, тип даних – integer;
- Назва відділу – varchar(50);

Таблиця «Користувач» - список даних користувачів для автентифікації та авторизації в ІС. Має наступні поля:

- Код користувача – первинний ключ, тип даних – integer;
- login – varchar(50);
- password – varchar(50);
- Код співробітника – зовнішній ключ, тип даних – integer;
- access – integer. Поле для авторизації, по ньому програма буде розуміти, з якими даними користувач має право взаємодіяти.

Таблиця «Тип заявки» створена для можливості вибору і підстановки типу заявки в ІС. Має два поля:

- Код типу заявки – первинний ключ, тип даних - integer;
- Тип заявки – varchar(50);

Таблиця «Заявка» використовується для створення заявки з описом проблеми, яку необхідно вирішити до ІТ-відділу підприємства. Має такі поля:

- Код заявки – первинний ключ, тип даних – integer;
- Код типу заявки – зовнішній ключ, тип даних – integer;
- Заявка – varchar(50);
- Статус виконання – varchar(50);

- Код співробітника – зовнішній ключ, тип даних – integer;
- Код відомості – зовнішній ключ, тип даних – integer;
- Дата створення заявки – datetime;

Таблиця «**Акти виконаних заявок**» використовується для збереження даних про заявки, які були виконані: хто її виконав і коли.

Містить такі поля:

- Код акту – первинний ключ, тип даних – integer;
- Код заявки – зовнішній ключ, тип даних – integer;
- Дата виконання – datetime;
- Код співробітника – зовнішній ключ, тип даних – integer;
- Примітка – varchar(20);

Таблиця «**Інвентарна відомість**» - перелік програмно-технічного комплексу підприємства. Має наступні поля:

- Код відомості – первинний ключ, тип даних – integer;
- Код типу відомості – зовнішній ключ, тип даних – integer;
- Кількість – integer;
- Назва – varchar(50);
- Код відділу – зовнішній ключ, тип даних – integer;
- Дата внесення в базу – datetime;

Таблиця «**Тип відомості**» створена для можливості вибору і підстановки типу відомості в ІС. Має два наступних поля:

- Код типу відомості – первинний ключ, тип даних – integer;
- Тип відомості – varchar(50);

2.3. Створення інтерфейсу користувача

Для роботи з базою даних необхідно в проєкті Visual Studio створити джерело даних, яке забезпечить зв'язок інтерфейсу користувача з сервером та базою, яка на ньому зберігається [15].

Для створення джерела даних переходимо по пункту меню Проєкт – Додати нове джерело даних – обираємо сервер та базу даних, яку необхідно підключити:

Рис. 2.1. Підключення до джерела даних

Одразу можна перевірити підключення, натиснувши кнопку «Проверить подключение» в лівому нижньому куті на рис. 2.1. Якщо з підключенням все гаразд, то джерело даних створено і згенеровано рядок підключення, який виглядає наступним чином: *Data Source=DESKTOP-DGAB6C4\SQLDEN;Initial Catalog=itdep;Integrated Security=True*.

Далі потрібно обрати об'єкти бази даних, які будуть передані до середовища розробки: таблиці та представлення, якщо вони є. Після цього джерело даних повністю сформоване та готове до використання.

Проект створено на основі шаблону Додаток Windows Forms. Тобто основною робочою областю, на якій будуть розміщені всі елементи управління системою буде форма, яка має такий вигляд:

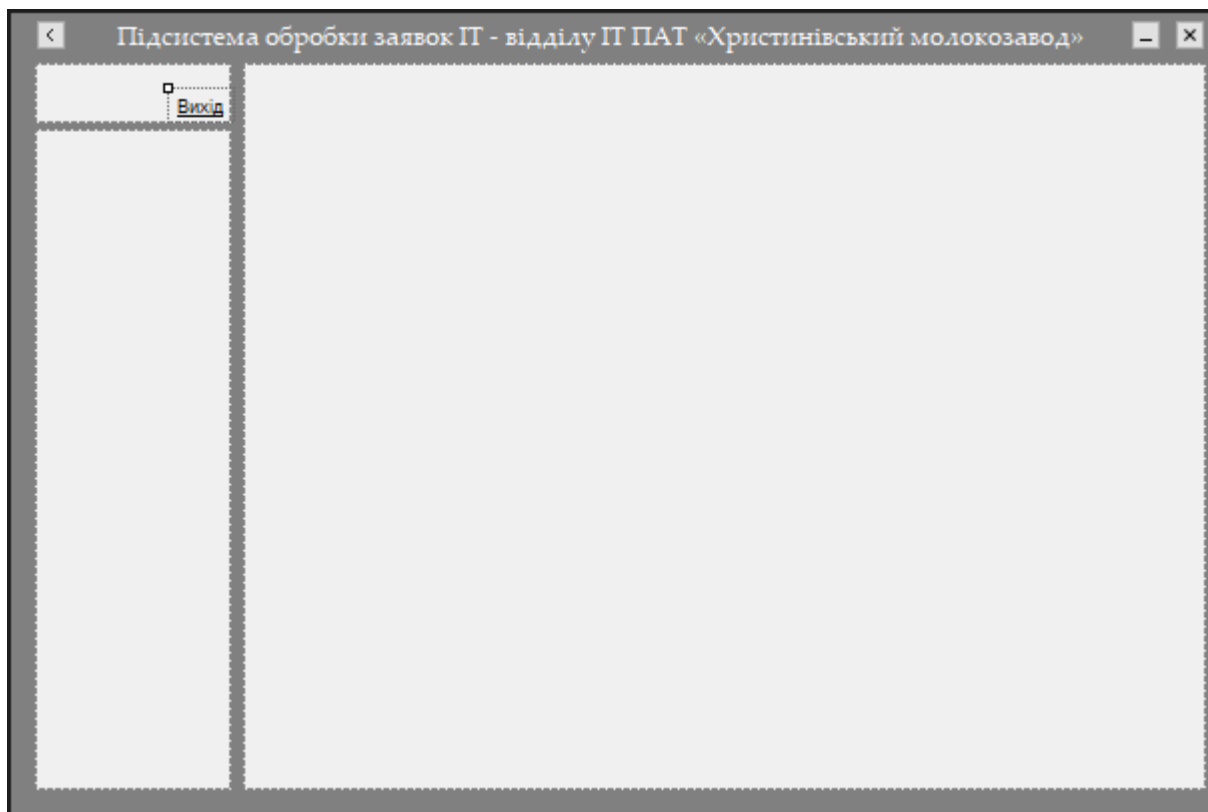


Рис. 2.2. Головна сторінка проекту

Для цього елементу було змінено такі властивості: Size змінено на 600; 400, FormBorderStyle змінено на None, що відключає можливість зміни розміру елемента користувачем, відключає верхню панель управління (кнопка масштабування, закриття, згортування) та відключає можливість переміщення вікна по поверхні робочого столу). Це було зроблено для відсутності можливості масштабування вікна та для більш естетичного вигляду. Натомість були створені кнопки для закриття і згортування системи, також було створено процедуру для можливості переміщення вікна по робочому столу, код якої наведено в додатку Г.

Для доступу до даних використовується елемент User Control, для його створення переходимо по пункту меню Проект – Додати новий User Control [16]. Для відображення User Control використовується контейнер Panel із стандартних елементів управління Windows Forms.

На рис. 2.2. можемо побачити два елементи Panel [17], всередині яких будуть розміщуватись User Control-и. Правий елемент займає найбільше місця, тому що всередині нього будуть розміщуватись User Control-и з даними, які користувач зможе переглядати або змінювати, лівий елемент Panel буде використовуватись для розміщення в ньому користувацького меню, користувачі з різними рівнями доступу матимуть різне меню.

User Control для перегляду заявок, які створював кожен користувач має наступний вигляд:

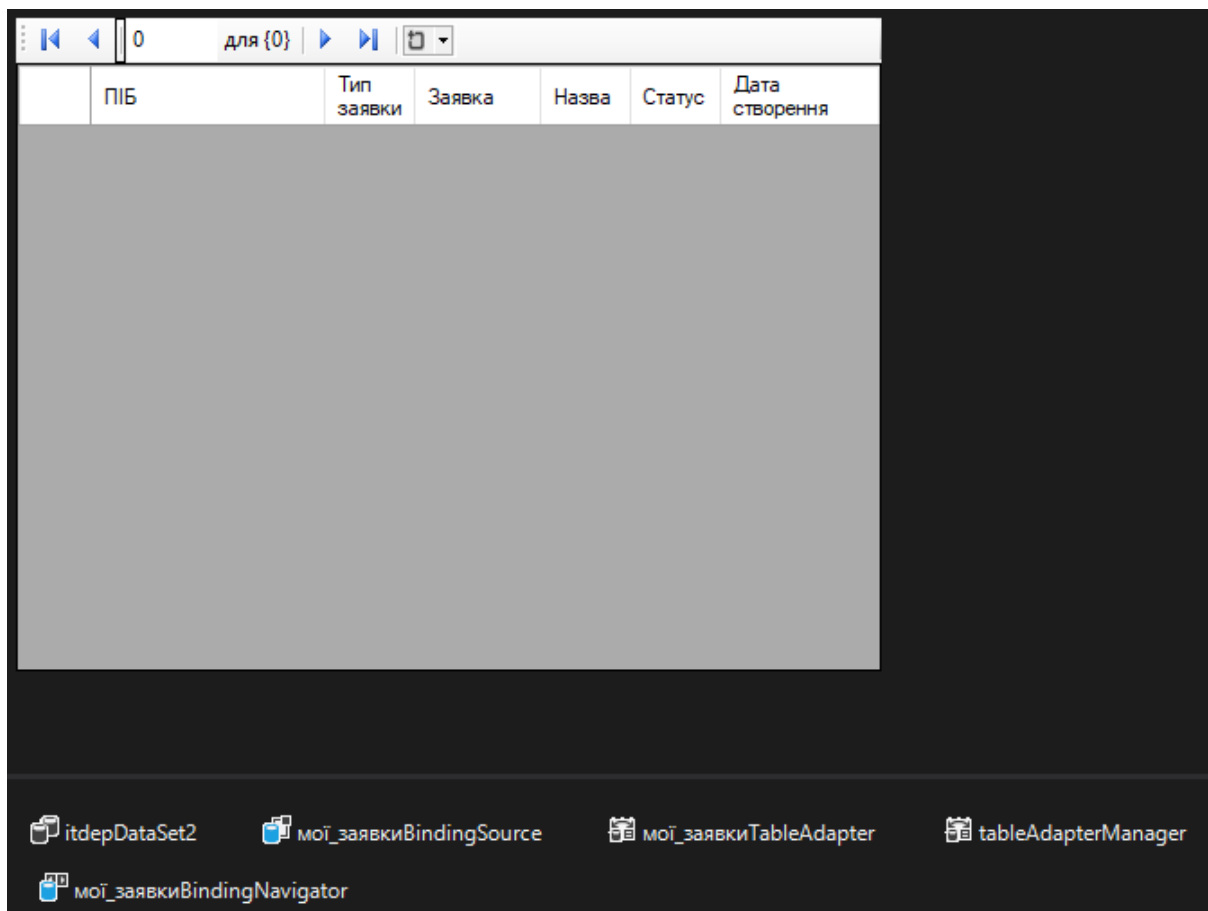
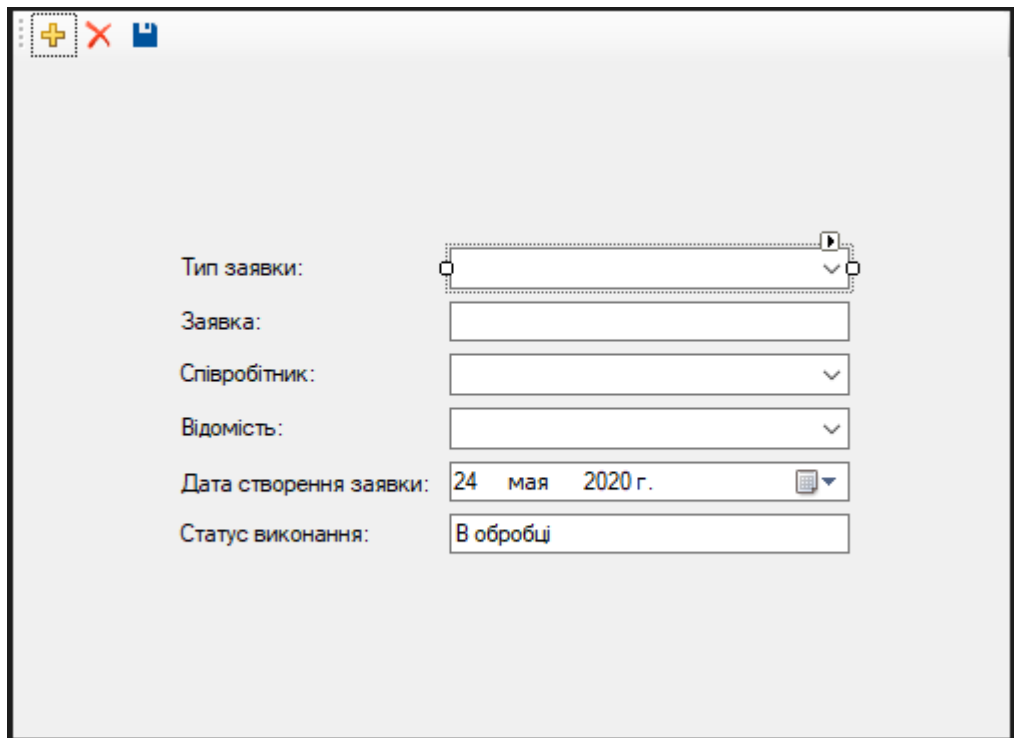


Рис. 2.3. User Control для перегляду створених заявок для працівників не ІТ-відділу

Щоб забезпечити доступ до даних всередині User Control, потрібно з вікна Джерела даних перетягнути потрібну таблицю, або сформований перед цим запит, про який мова буде йти пізніше. Перед переміщенням можемо обрати вигляд, в якому дані з таблиці будуть відображені: DataGridView або вигляд Відомості. На рис. 2.3. дані розміщено у вигляді DataGridView – таблиця зі сформованими у звіті полями. Для цього елементу було відключено можливість додавання та редагування даних,

оскільки ця сторінка призначена лише для перегляду створених раніше заявок. Під час переміщення таблиці з Джерел даних, на User Control автоматично з'явився навігатор BindingNavigator, який призначений для переміщення по записах, але з нього було видалено кнопки для додавання та зберігання даних. Крім того, в області компонентів з'являться itdepDataSet, tableTableAdapter, tableBindingSource і tableBindingNavigator, які відповідають за заповнення DataGridView даними.

User Control для створення заявок від співробітників має такий вигляд:



The screenshot shows a Windows-style User Control window with a title bar containing a maximize button, a close button (red X), and a save button (blue disk). The form is titled "Заявка" (Request) and contains the following fields:

- Тип заявки: (Request type) - A dropdown menu with a downward arrow.
- Заявка: (Request) - A text input field.
- Співробітник: (Employee) - A dropdown menu with a downward arrow.
- Відомість: (Notice) - A dropdown menu with a downward arrow.
- Дата створення заявки: (Request creation date) - A date picker showing "24 мая 2020 г." with a calendar icon.
- Статус виконання: (Execution status) - A dropdown menu showing "В обробці" (In processing).

Рис. 2.4. User Control для створення заявок від співробітників

Тут дані відображені у вигляді Відомість. Оскільки таблиця «Заявка» в базі даних має зовнішні ключі, доцільно буде забезпечити підстановку даних до цих полів. Підстановка даних відбувається за допомогою елемента ComboBox. У вікні Джерела даних для окремих полів, в які потрібно буде підставити дані, змінимо вигляд на ComboBox. Для підстановки перемістимо відповідні дочірні таблиці: «Тип заявки», «Співробітник» та «Інвентарна відомість» на ComboBox-и, які відповідають цим таблицям. Наприклад, підстановка даних до поля «Співробітник» має такий вигляд:

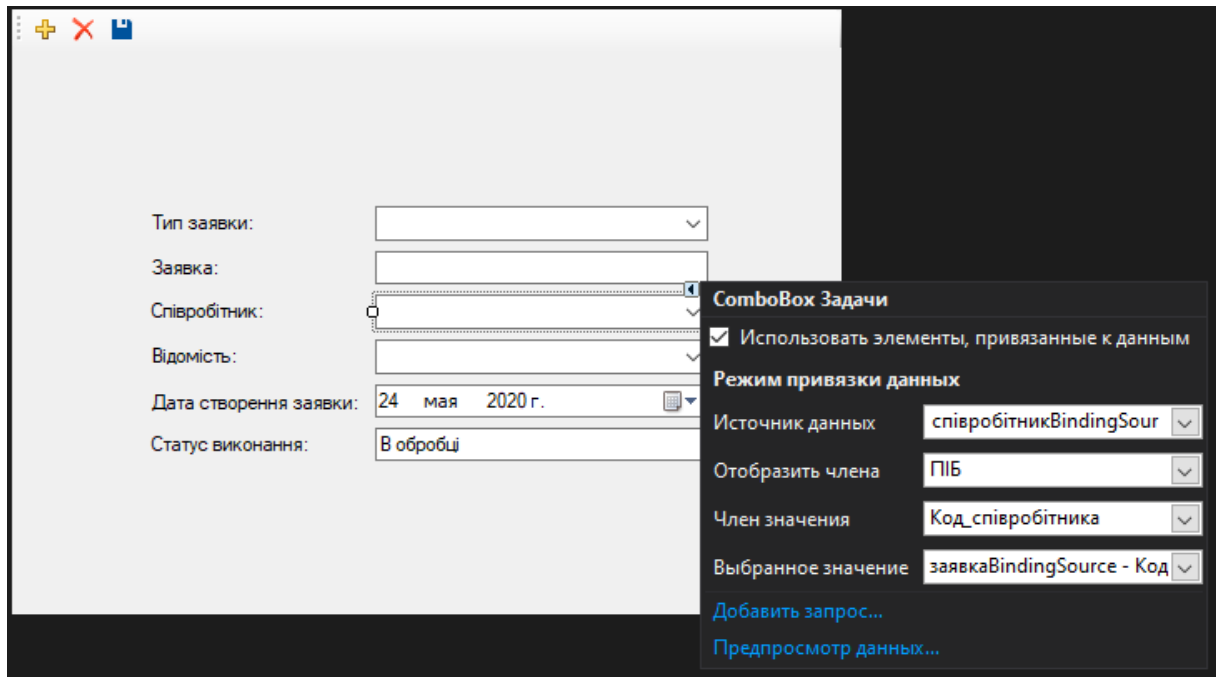


Рис. 2.5. Налаштування підстановки даних з дочірніх таблиць

Для того, щоб дані в цих полях відображалися, в процедурі завантаження цього User Control-а треба додати функцію заповнення для кожної з дочірніх таблиць, наприклад:

```
this.співробітникTableAdapter.Fill(this.itdepDataSet2.Співробітник);
```

Користувачьке меню має такий вигляд:

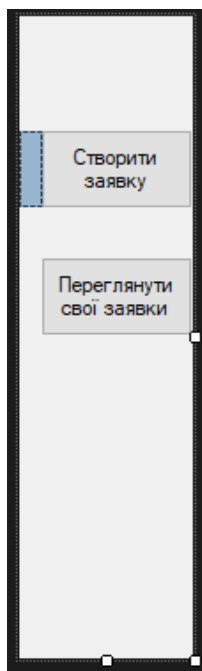


Рис. 2.6. Користувачьке меню для співробітників не IT-відділу

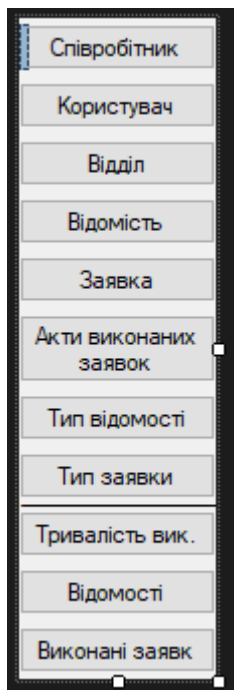


Рис. 2.7. Користувацьке меню для співробітників ІТ-відділу

Як можна побачити з рис. 2.6 та 2.7, користувацьке меню складається з кнопок. При натиску на будь-який пункт меню, буде відкрита відповідна сторінка.

Приклади реалізації процедури натиску кнопок в користувацькому меню наведені в додатку Г.

В системі реалізована автентифікація та авторизація. Для входу в систему використовується окремий User Control, на якому розміщено два елементи вводу даних TextBox, куди користувачі будуть вводити логін та пароль та кнопка «Увійти», яка виконує основну роль на цій сторінці. Сторінка входу має наступний вигляд:

Запросити пароль'."/>

Щоб продовжити роботу, потрібно увійти

Введіть логін:

Введіть пароль:

Користувача з такими даними не знайдено.
Можливо Ви ввели дані некоректно.
Ви можете запросити пароль, натиснувши:
[Запросити пароль](#)

Рис. 2.8. User Control для входу в систему

Повний код обробки натискання кнопки «Увійти» додано в додатку Г. Проте, одним із найважливіших в коді обробки є код SQL – запиту, який повертає дані по введених користувачем логіну і паролю, якщо такі дані присутні в базі, потім ці дані використовуються для авторизації та автентифікації. Код SQL – запиту:

```
SELECT [login],[password],[access] FROM [Користувач] WHERE  
[login] COLLATE Cyrillic_General_CS_AI = "" + login + ""and [password]  
COLLATE Cyrillic_General_CS_AI="" + pass + "", де login та pass - змінні, які  
користувач ввів в поля логіну та пароля. COLLATE Cyrillic_General_CS_AI  
– цей фрагмент коду забезпечує вразливість запиту до регістру: якщо логін  
користувача в базі даних буде наприклад user, а користувач в полі логіну  
його введе як User, то запит не буде виконано [18].
```

Для поля введення пароля властивість UseSystemPasswordChar було змінено на True, що дозволяє відображати текст в цьому полі символами пароля по замовчуванню.

З рис. 2.5. видно, що під кнопкою «Увійти» є текст, який по замовчуванню для користувача невидимий і з’являється лише тоді, коли користувача за введеними даними не було знайдено та елемент linklabel, за

допомогою якого користувач може запросити свій забутий пароль. При натисненні на цей елемент з'являється новий User Control:

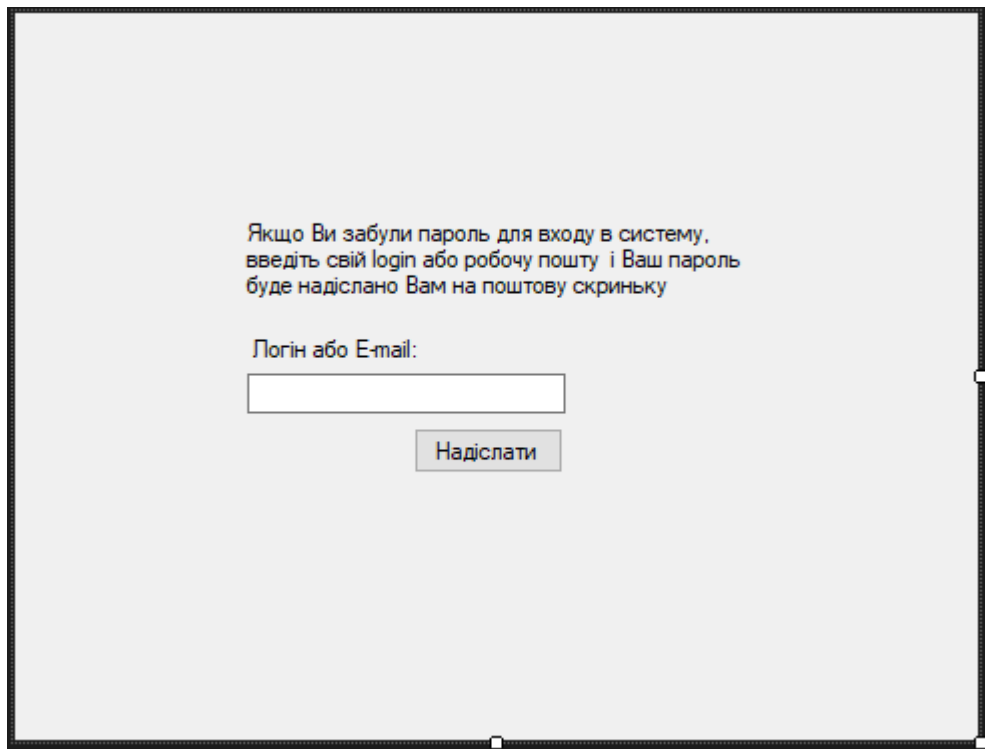


Рис. 2.9. User Control для запита забутого пароля

Кнопка «Надіслати» при її натисненні створює нову заявку, яка буде містити введений користувачем логін або електронну пошту, тип заявки – пароль та статус виконання «Виконується», SQL код запиту створення заявки:

```
INSERT INTO Заявка(Код_типу_заявки, Заявка, Статус_виконання)
VALUES (1," + logmail + ", " + vuk + " ),
```

де logmail – введений користувачем текст, а vuk – змінна, в якій зберігається статус виконання, який автоматично присвоюється заявці.

Передбачено, що заявка не може бути створена і надіслана, якщо поле введення є пустим. Код цієї обробки наведено в додатку Г.

Вибірка даних виконується за допомогою запитів. Запити формуються за допомогою створення нового TableAdapter і Конструктору запитів всередині нього. Наприклад, запит, який відображається на User Control-і на рис. 2.3, має такий вигляд :

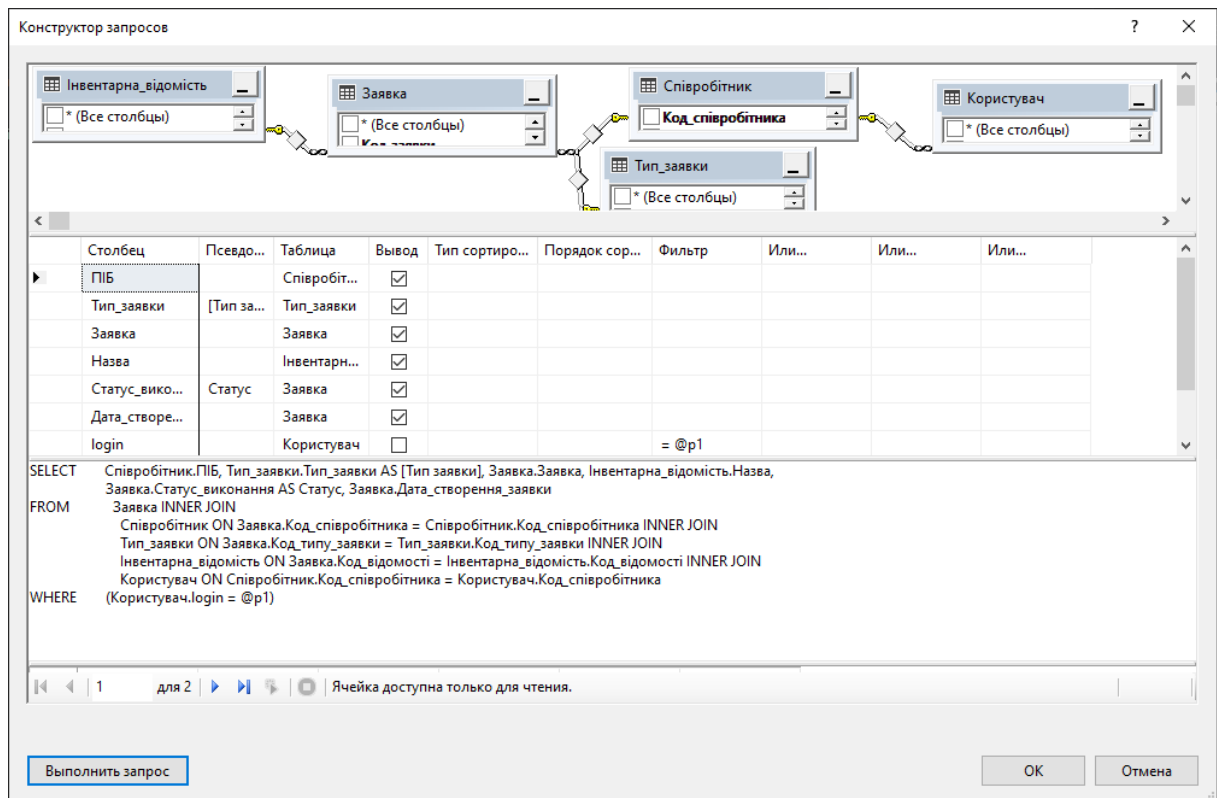


Рис. 2.10. Запит для відображення заявок від конкретного користувача

В якості параметра для цього запиту виступає логін користувача, під яким він здійснив вхід в систему.

Створені запити - TableAdapter-и стають Джерелами даних і їх можна як таблицю помістити в User Control для перегляду даних.

Приклади інших запитів наведено в додатку В.

Також була створена можливість пошуку необхідних даних в межах однієї таблиці. Адміністратори (працівники ІТ-відділу) мають можливість переглядати і змінювати дані зі всіх таблиць, для полегшення пошуку необхідного запису в деяких таблицях були створені додаткові елементи, в яких міститься перелік всіх значень одного поля в межах однієї таблиці, обравши необхідне значення, адміністратор побачить всі дані, які відносяться до обраного значення. Наприклад, таблиця «Користувач» містить додатковий елемент ComboBox, до якого підставлені всі значення поля login цієї таблиці.

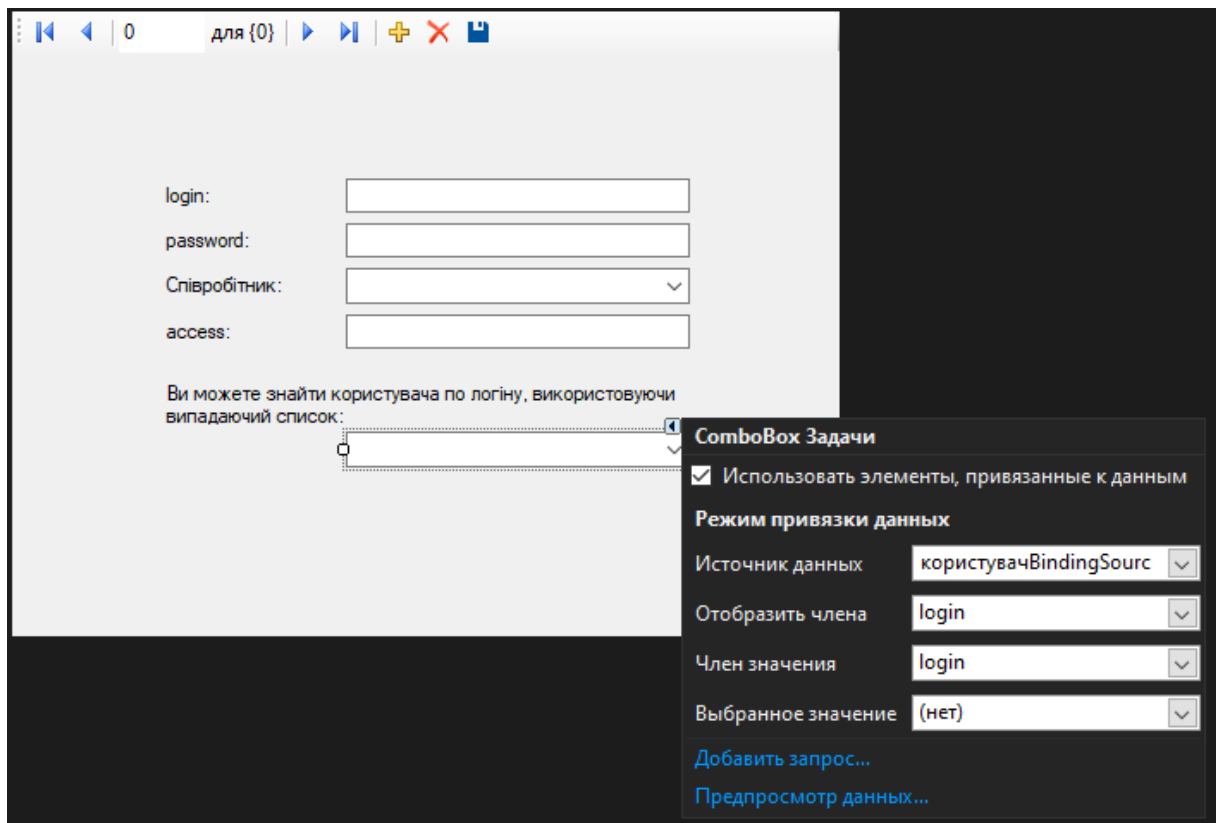


Рис. 2.11. User Control з випадаючим списком для пошуку даних

Звіти створюються за допомогою майстра звітів, який викликається шляхом додання нового елементу до проекту. В майстрі звітів обираємо запит, по якому хочемо отримати звіт та відповідні поля.

Для перегляду звітів використовуємо User Control та елемент ReportViewer до якого підключаємо створений раніше звіт [19], наприклад:

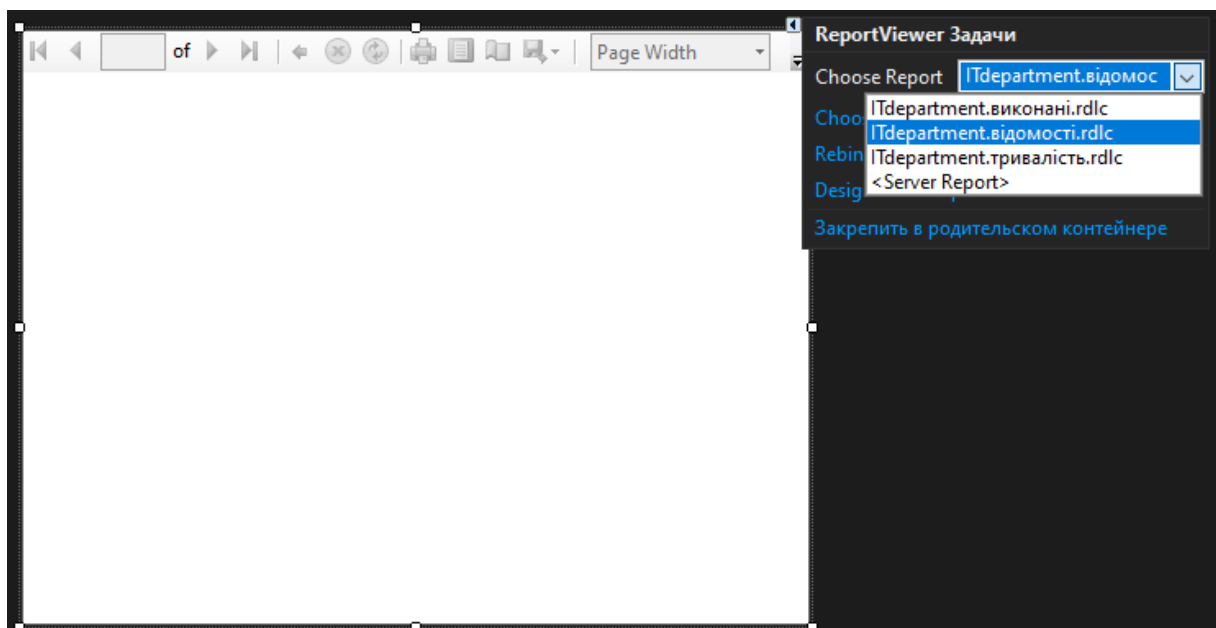


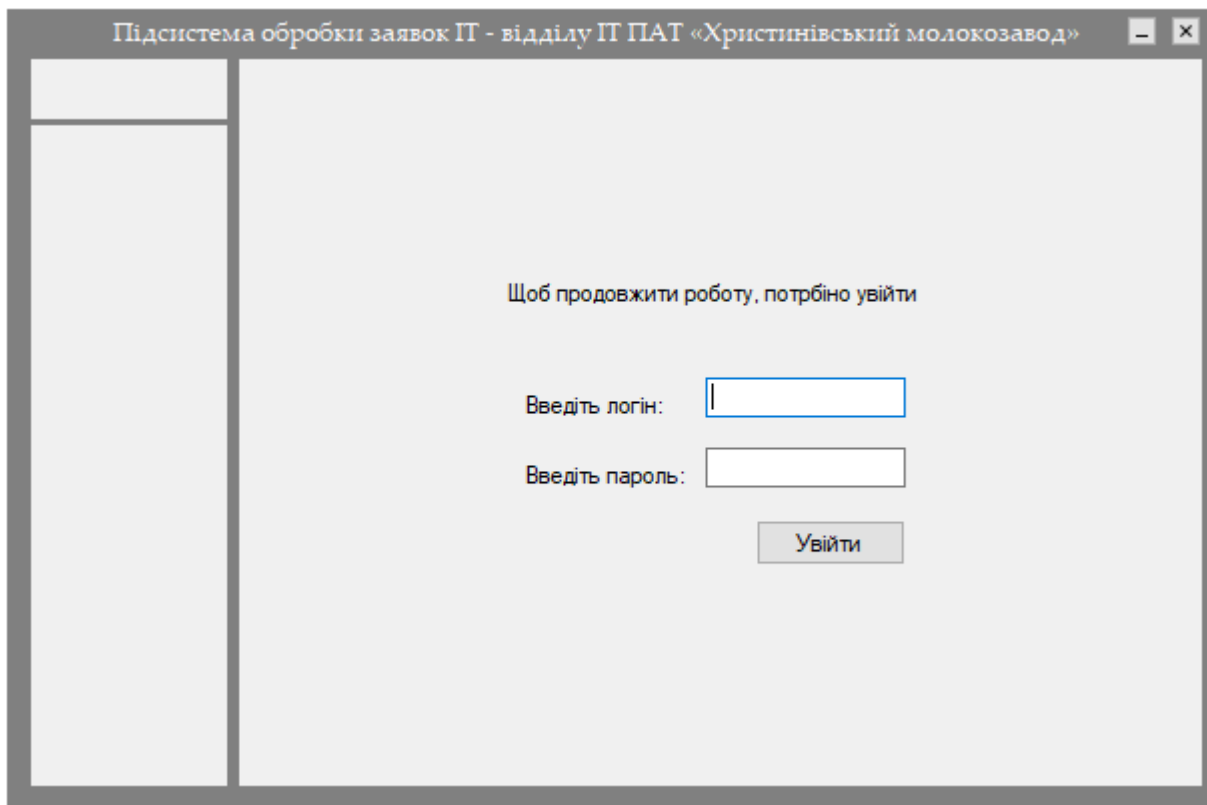
Рис. 2.12. Розміщення звіту всередині User Control

2.4. Інструкція користувача

Так як створена система передбачає роботу двох типів користувачів: працівників ІТ-відділу (адміністраторів) та працівників інших відділів підприємства, інструкція буде написана окремо для двох цих типів користувачів. До моменту роботи з системою, системний адміністратор повинен встановити систему на комп'ютери у всіх відділах підприємства.

2.4.1. Інструкція для працівників не ІТ-відділу

Коли користувач відкриває систему, в першу чергу він бачить сторінку входу в систему, і повідомлення про те, що для продовження роботи в системі, йому потрібно здійснити вхід.



Підсистема обробки заявок ІТ - відділу ІТ ПАТ «Христинівський молокозавод»

Щоб продовжити роботу, потрібно увійти

Введіть логін:

Введіть пароль:

Рис. 2.13. Сторінка для входу в систему

Реєструє користувачів в системі адміністратор, дані для входу кожен співробітник отримує на робочу електронну пошту.

Користувач вводить дані для входу у відповідно підписані поля і натискає кнопку «Увійти» для входу.

Після здійснення входу, користувача одразу з'являється на сторінці створення нової заявки:

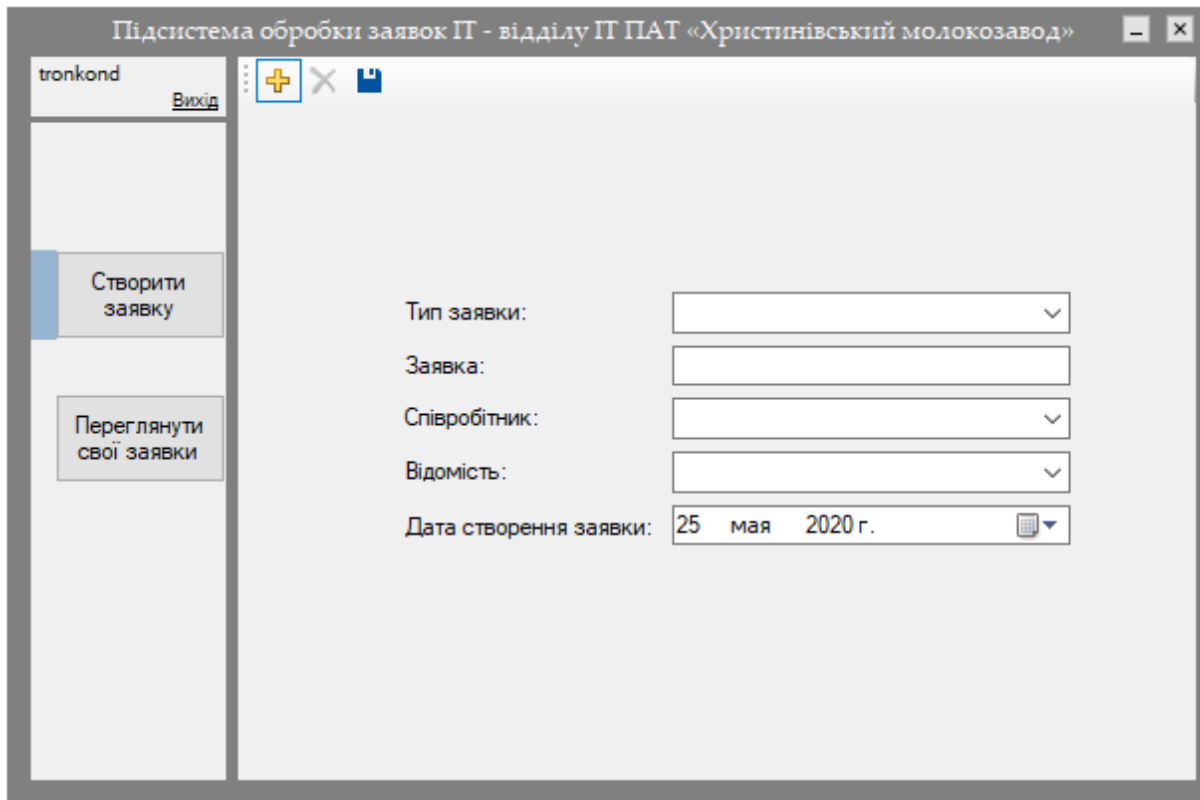


Рис. 2.14. Сторінка створення нової заявки

На цій сторінці розміщені елементи вводу даних, в які користувач вручну вводить необхідні дані, а також елементи вибору можливих даних, наприклад «Тип заявки», в якому розміщені всі можливі типи заявок, які користувач може створити.

У верхній частині сторінки знаходиться елемент управління даними з трьома кнопками: жовта кнопка «+» дає можливість створити заявку та почати вводити дані, наступна кнопка стає активною тільки якщо натиснута попередня і вона відповідає за видалення створеної заявки, остання кнопка зберігає створену заявку.

На рис. 2.12. з лівої сторони можемо побачити користувацьке меню, яке складається з двох пунктів.

Другий пункт меню «Переглянути свої заявки» дозволяє переглянути дані про заявки, які були створені користувачем, який зайшов в систему.

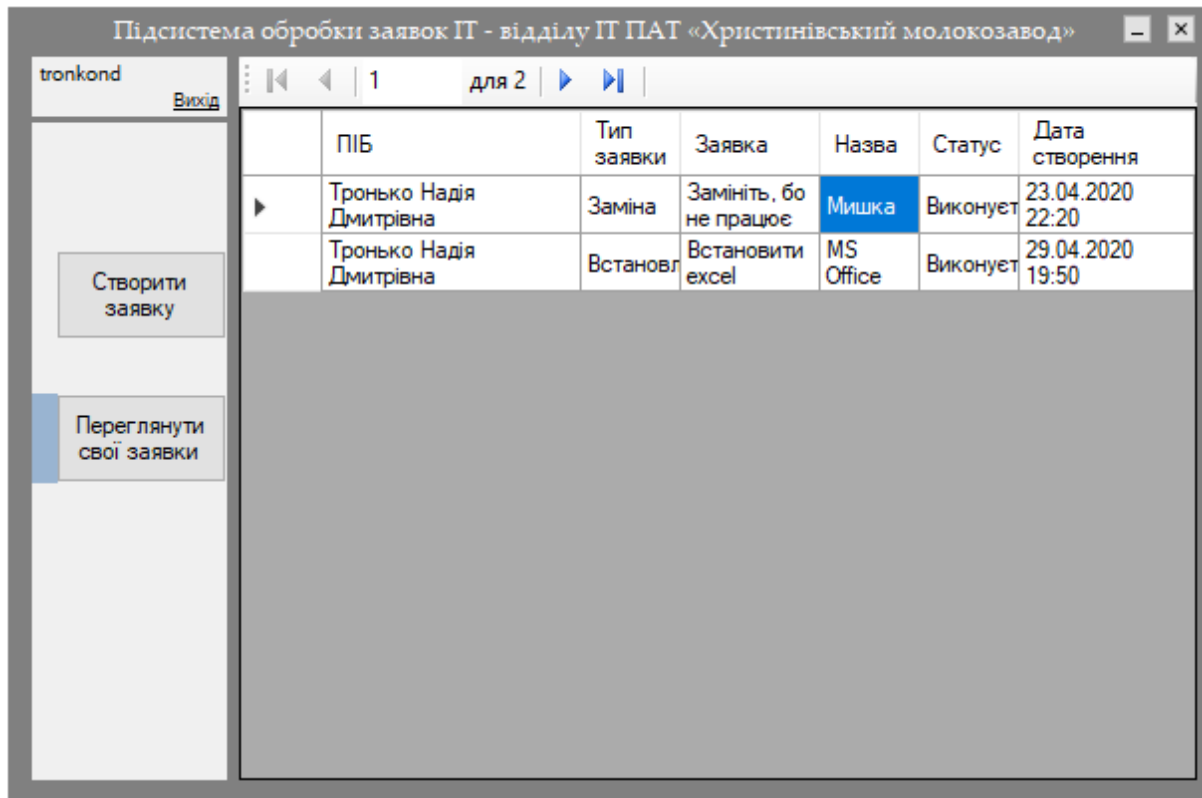


Рис. 2.15. Сторінка перегляду своїх заявок

У верхній частині сторінки знаходиться елемент управління даними, який забезпечує переміщення між створеними заявками. Таблиця, в якій зображено дані призначена тільки для перегляду, змінити чи додати дані тут не можна.

На цьому основні функції закінчуються. Якщо користувач закінчив роботу з системою, він може натиснути кнопку «Вихід» та повернутись до початкової сторінки входу.

2.4.2. Інструкція для працівників ІТ-відділу

Після здійснення входу, користувача одразу потрапляє на сторінку для вводу, перегляду, редагування та видалення даних таблиці «Співробітник».

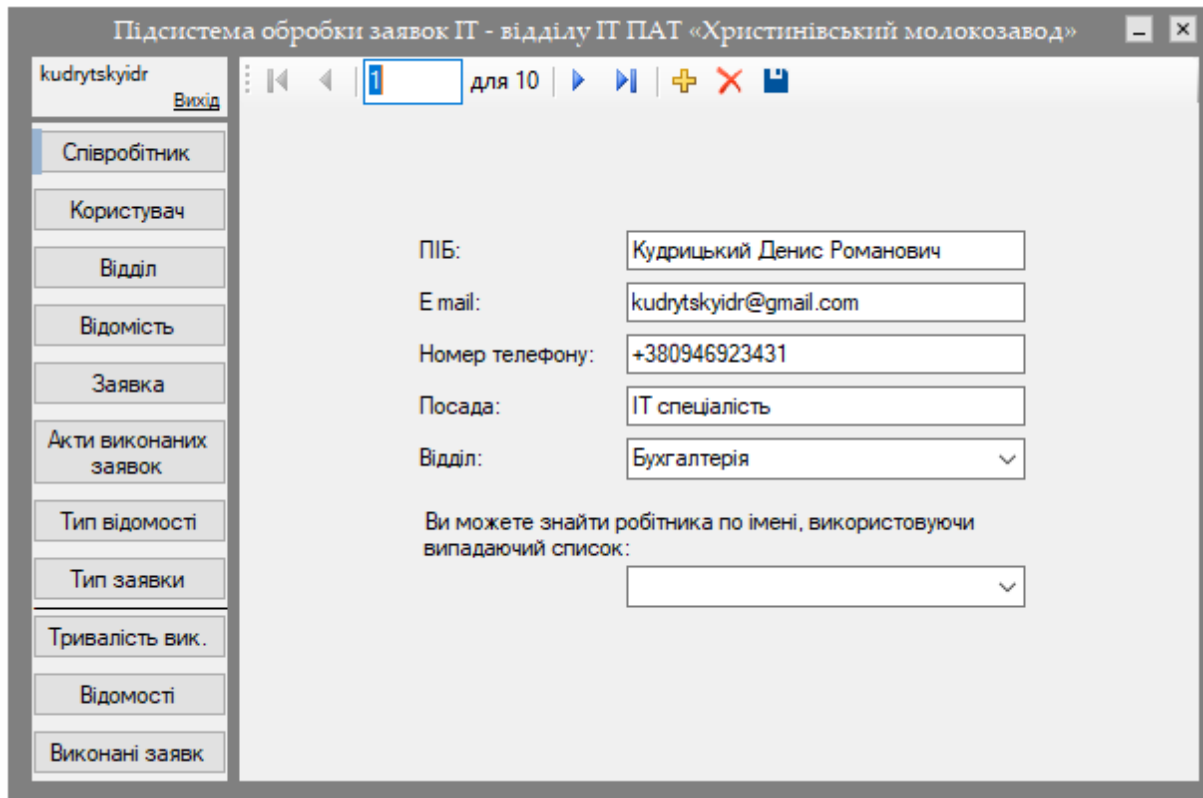


Рис. 2.16. Сторінка для вводу, перегляду, редагування та видалення даних таблиці «Співробітник»

Всі інші ілюстрації пунктів меню, які призначені для перегляду чи зміни даних із таблиць бази даних знаходяться в додатку В.

На всіх сторінках перегляду даних (див. рис.2.13) у верхній частині знаходиться елемент управління даними таблиці. За допомогою стрілочок відбувається переміщення між записами таблиці. Натиснувши на жовту кнопку «+» можна вводити новий запис до таблиці. За допомогою червоного «X» відбувається видалення обраного запису з таблиці. Синя дискета зберігає введенні або змінені дані, якщо її натиснути [20].

На деяких сторінках перегляду даних передбачена функція пошуку потрібного запису в таблиці, наприклад таблиця «Користувач» має таку функцію:

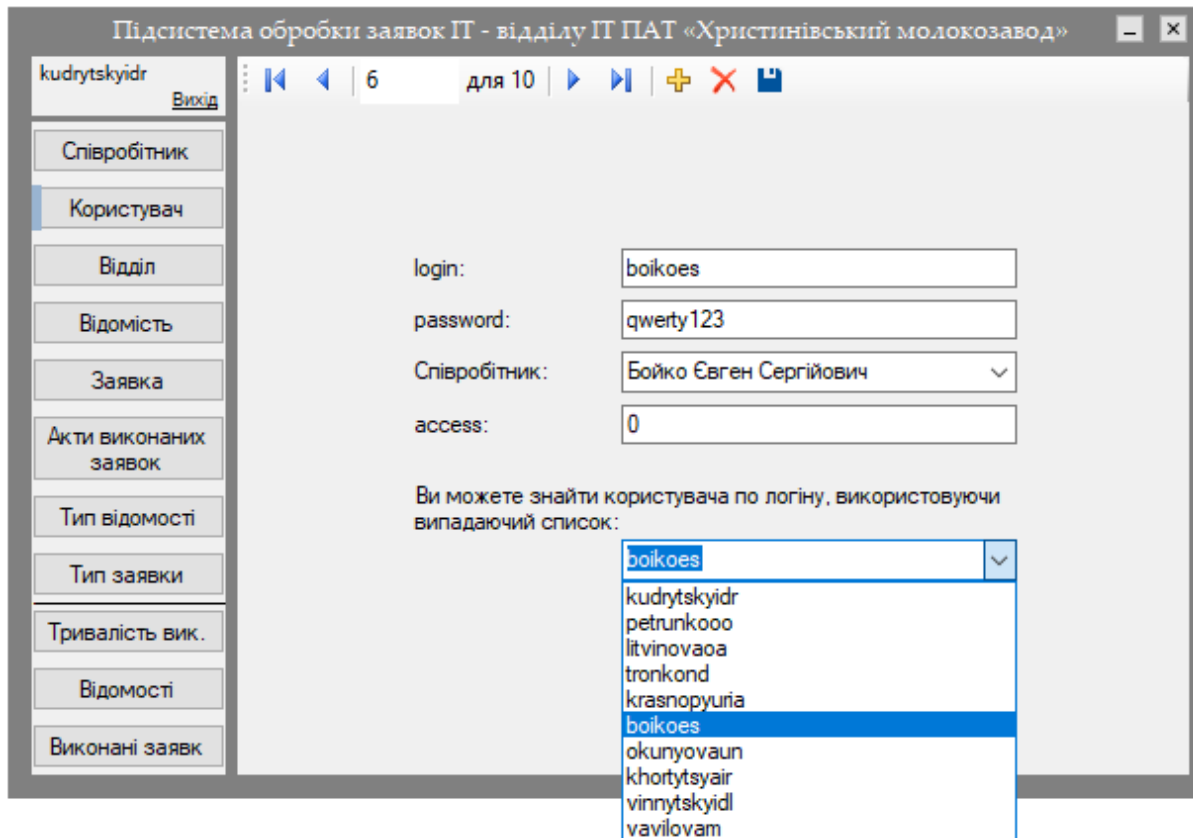


Рис. 2.17. Реалізована функція пошуку по логіну на сторінці перегляду таблиці «Користувач»

Користувач обирає потрібний йому логін з випадаючого списку і система автоматично показує йому запис в таблиці, пов'язаний з цим логіном.

Інші прикладу пошуку даних на сторінці показано в додатку В.

Також у системі реалізовано можливість перегляду трьох типів звітів. При натисненні, наприклад пункту меню «Тривалість виконання» відкривається сторінка зі звітом з розрахованою тривалістю виконання всіх виконаних заявок.

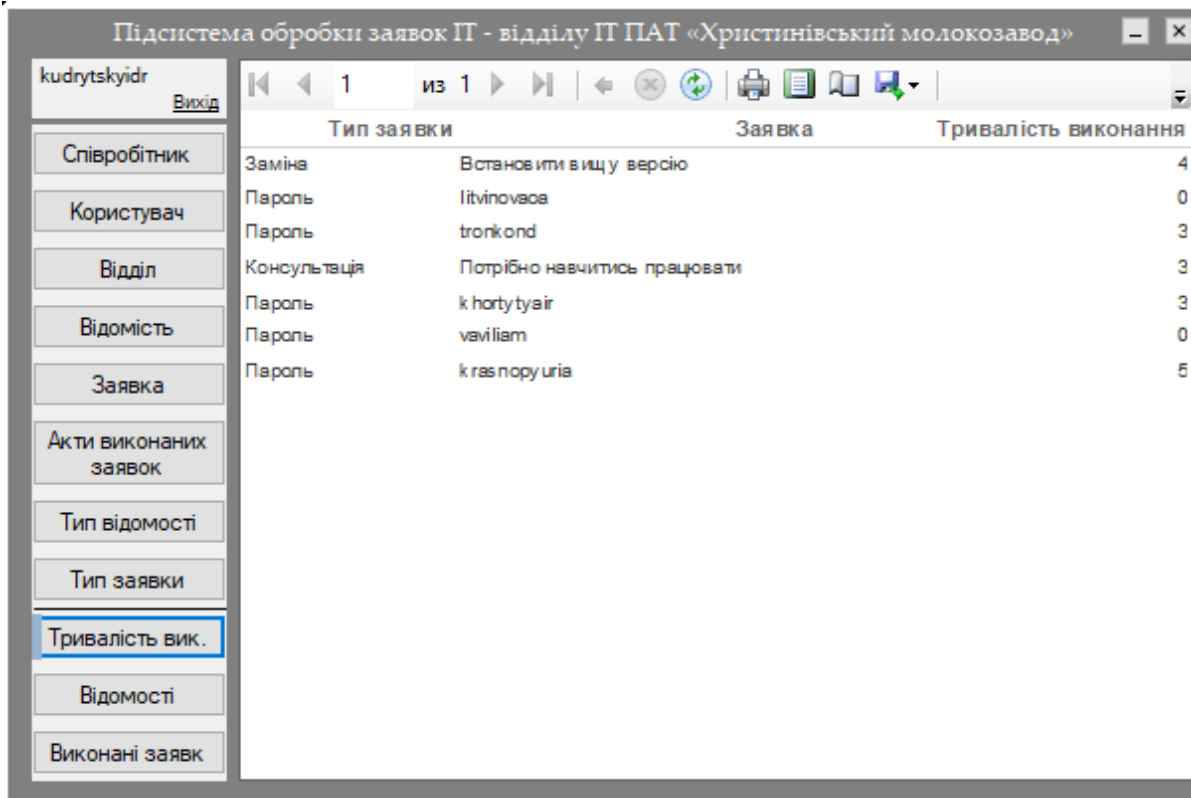


Рис. 2.18. Сторінка для перегляду звіту «Тривалість виконання»

На всіх сторінках зі звітами у верхній частині знаходиться елемент керування, який дозволяє (кнопки по - порядку): за допомогою стрілок переміщуватись між сторінками, оновляти дані у звіті, друкувати дані, змінювати вигляд перегляду даних, змінювати параметри сторінки, експортувати звіт у: Excel, Word та PDF, зміна масштабу сторінки, кнопка в правому кінці при її натисненні показує: поле для введення даних для пошуку по звіті, кнопка пошуку даних у звіті, кнопка переходу до наступного елементу при пошуку.

Дві інші сторінки перегляду звітів наведені в додатку В.

Якщо користувач закінчив роботу з системою, він може натиснути кнопку «Вихід» та повернутись до початкової сторінки входу.

2.4.3. Загальна інструкція для ситуації, коли користувач забув пароль

Якщо користувач забув свій логін для входу в систему або ввів його неправильно, на сторінці входу з'явиться відповідне повідомлення:

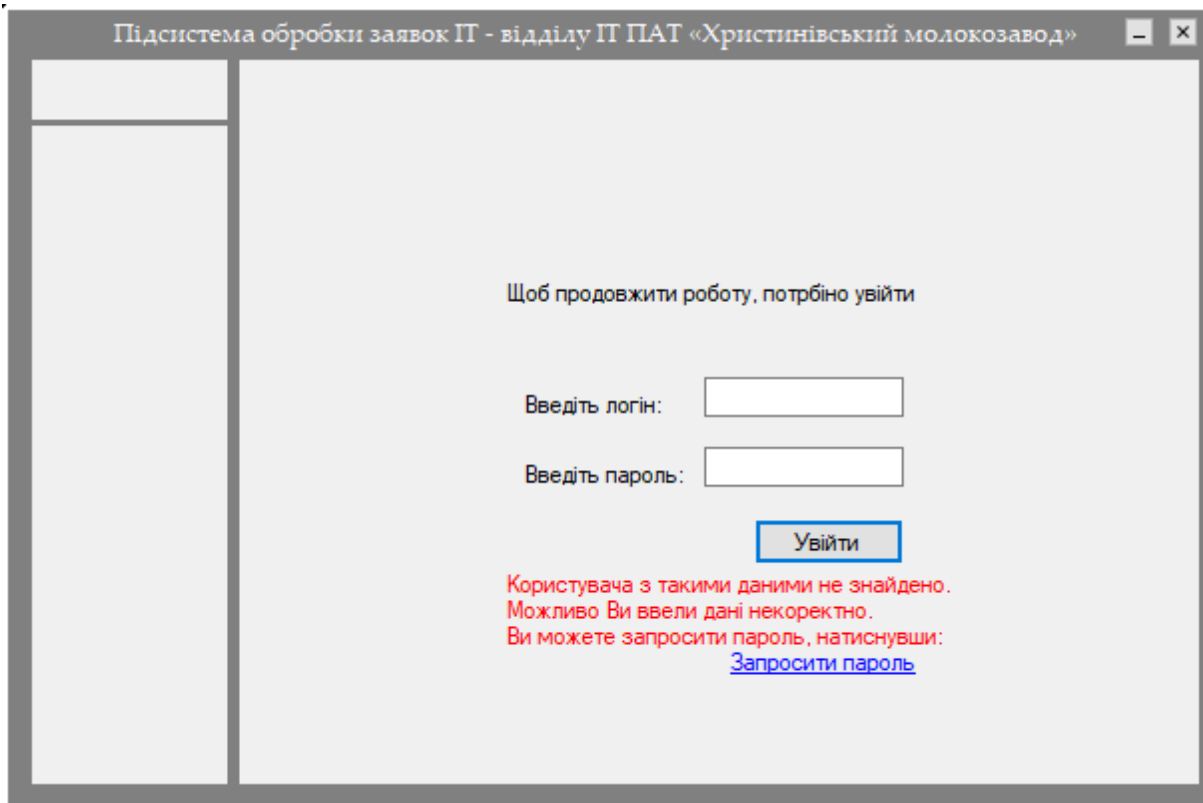


Рис. 2.19. Сторінка входу в систему, коли дані для входу були введені неправильно

Якщо ж користувач забув свій пароль, він може натиснути на відповідний синій напис внизу сторінки «Запросити пароль».

При натисненні відкривається наступна сторінка:

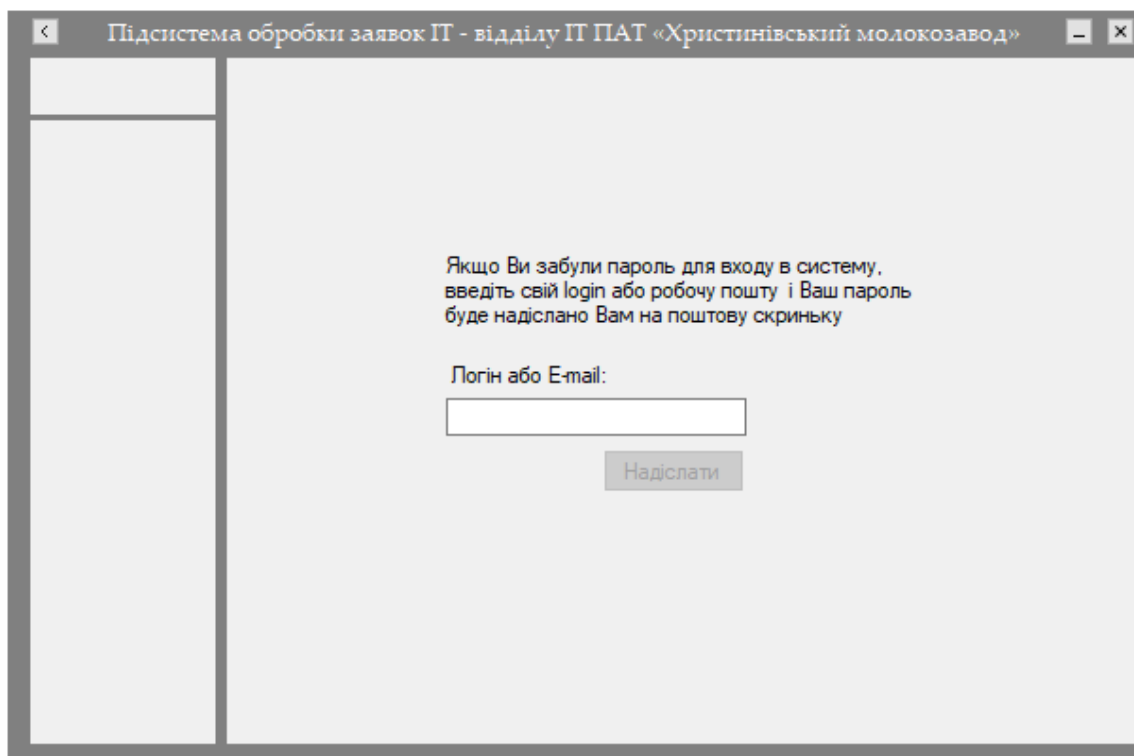


Рис. 2.20. Сторінка запиту забутого пароля

В відповідне поле введення користувач вводить свій логін або робочу електронну пошту, на яку буде надіслано забутий пароль.

Спочатку кнопка «Надіслати» є неактивною, але як тільки користувач щось вводить в поле, кнопка стає активною і запит можна надіслати, як тільки запит надісланий, з'являється відповідне повідомлення і після цього поле введення та кнопка стає недоступними.

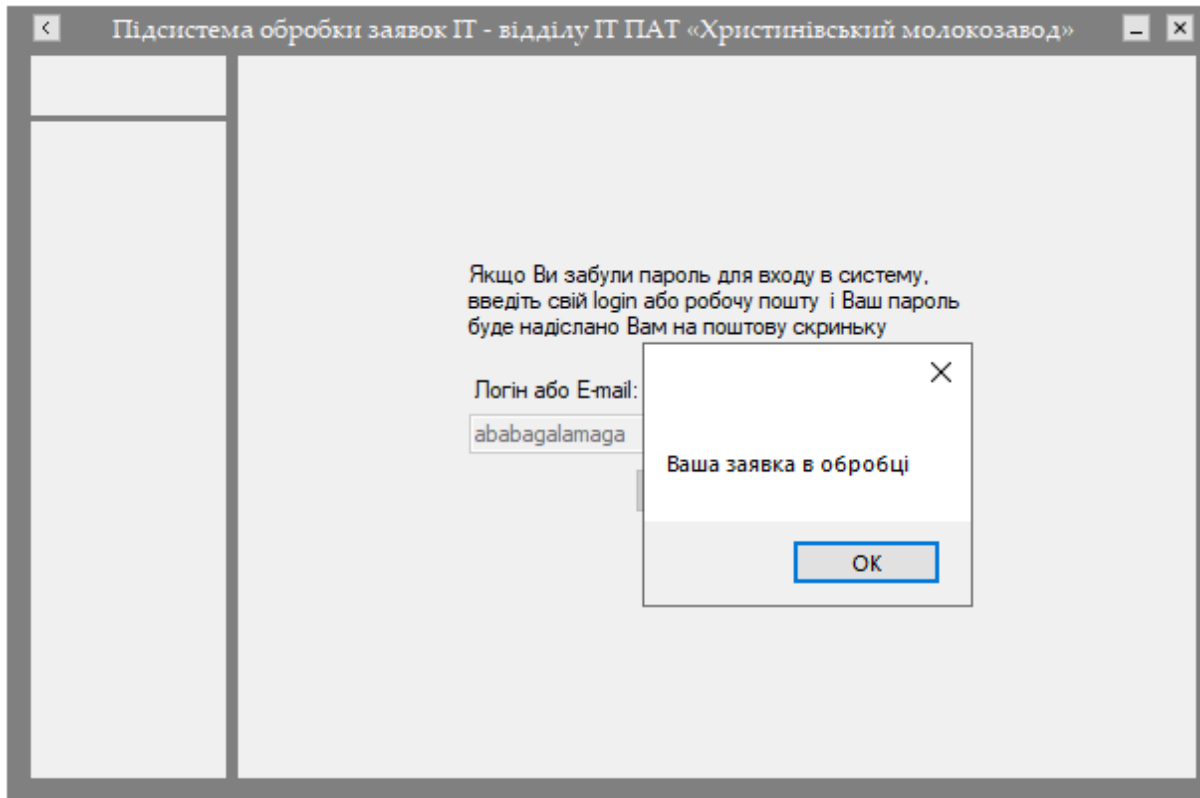


Рис. 2.21. Успішно надісланий запит на забутий пароль

Для повернення на початкову сторінку користувач може скористатись кнопкою «<» у верхньому лівому куті.

2.5. Техніко-економічне обґрунтування

Економічний ефект від впровадження ІС визначається шляхом співвідношення прибутку від впровадження системи і витрат, які йдуть на розробку системи [21].

Витрати на розробку проекту розраховуються за формулою:

$$V_{\Sigma} = V_1 + V_2 + V_3 + V_4 \quad (1)$$

де V_1 - витрати на розробку, V_2 – витрати на придбання ПК, V_3 – витрати на підготовку приміщення, V_4 – витрати на навчання персоналу

Витрати на розробку ІС

Річний час роботи ПК ≈ 1900 годин.

Якщо враховувати, що на розробку дипломної роботи студент витрачає ≈ 300 годин машинного часу, то час корисної роботи ПК буде дорівнювати

$$T_{\text{ПК}} = 1900 * \frac{300}{2002} = 285 \text{ годин.} \quad (2)$$

Витрати на електроенергію, затрачену на роботу ПК під час розробки ІС визначається за формулою:

$$V_{\text{ЕЛ}} = P_{\text{ПК}} * T_{\text{ПК}} * C_{\text{ЕЛ}} * A \quad (3)$$

де $P_{\text{ПК}}$ – потужність ПК, $T_{\text{ПК}}$ – час корисної роботи ПК, $C_{\text{ЕЛ}}$ – ціна 1 кВт електроенергії (0,9 грн/кВт), A – коефіцієнт інтенсивного використання ПК (0,9)

Тому витрати на електроенергію за формулою (3) дорівнюють

$$V_{\text{ЕЛ}} = 0,12 * 285 * 0,9 * 0,9 = 27,7 \text{ грн.}$$

Нехай витрати на ремонт та технічне обслуговування ПК на підприємстві будуть дорівнювати 1200 грн.

Якщо враховувати оплату, яка була б виплачена програмісту (студенту) за роботу, то вона б склала 5000 грн.

Отже, витрати на розробку ІС (V_1) складають $27,7 + 1200 + 5000 = 6\,227,7$ грн.

Витрати на придбання та установку ПК на підприємстві (V_2)

Так як потреби в закупівлі комп'ютерів нема, то витрати на це дорівнюють 0 грн.

Витрати на підготовку приміщення (V_3)

Не у всіх підрозділах підприємства однаково добре організовані місця для роботи з ПК, тому витрати на це будуть становити 1500 грн.

Витрати на навчання персоналу (V_4)

Не всі працівники підприємства мають однаковий рівень роботи з ПК, тому деякі співробітники потребують детальнішого пояснення роботи з створеною ІС. Можна вважати, що витрати на це будуть становити 300 грн.

Витрати на розробку проекту розраховуються за (1):

$$V_{\Sigma} = 6\,227,7 + 0 + 1500 + 300 = 8\,027,7 \text{ грн.}$$

Для обрахунку річного економічного ефекту слід брати величину з урахуванням норми амортизаційних витрат, а для комп'ютерних систем ця норма $H_A = 5$, тому за формулою

$$V_p = \frac{V_{\Sigma}}{H_A} \quad (4)$$

За (4), економічний ефект дорівнює:

$$V_p = \frac{8\,027,7}{5} = 1\,605,54 \text{ грн.}$$

Річний прибуток P_p від впровадження ІС буде досягнутий за рахунок зменшення збитків від аварійних ситуацій внаслідок своєчасного проведення технічного обслуговування обладнання, та зменшення збитків від простою окремих структурних підрозділів підприємства у зв'язку з неможливістю виконання якоїсь роботи і орієнтовно складатиме 5000.

Коефіцієнт економічної ефективності розробки розраховується за формулою:

$$K_{\text{ЕФ}} = \frac{P_p}{V_p} \quad (5)$$

За (5), коефіцієнт економічної ефективності розробки $K_{\text{ЕФ}}$ дорівнює:

$$K_{\text{ЕФ}} = \frac{5000}{1\,605,54} = 3,1$$

Термін окупності впровадження системи визначається за формулою:

$$T_{\text{ОК}} = \frac{1}{K_{\text{ЕФ}}} = \frac{1}{3,1} = 0,32$$

Отже, термін окупності ІС становить **4 місяці та 5 днів**.

Однією з причин низьких витрат на розробку ІС є використання безкоштовного програмного забезпечення.

РОЗДІЛ 3. ОХОРОНА ПРАЦІ

Відділ інформаційних технологій ПАТ «Христинівський молокозавод» забезпечує роботу і підтримку мережевого обладнання, інформаційних мереж, програмно-технічного комплексу, допомагає іншим структурним підрозділам вирішувати питання пов'язані з несправністю техніки чи з некоректною роботою програмного забезпечення. Дана робота стосується одного конкретного процесу – обробка заявок від всіх відділів підприємства на ремонт та обслуговування програмно-технічного комплексу.

Далі будуть розкриті основні чинники, які можуть створювати небезпеку або завдавати шкоду працівникам в межах вищевказаного процесу, джерела їх виникнення, який вплив ці чинники здійснюють на працівників та можливі заходи передбачення їх виникнення [22].

Під час процесу обробки та виконання заявки можна виявити такі фактори, які можуть завдавати шкоду здоров'ю або нести загрозу життю співробітників, які причасні до цього процесу та джерела їх виникнення [23]:

- Місце для роботи з ПК не відповідає вимогам. Робочий стіл може не відповідати сучасним вимогам ергономіки і не забезпечувати достатньої кількості робочої поверхні для розміщення на ньому клавіатури, миші, монітору. Сюди ж відноситься місце для сидіння. Воно може бути просто табуретом.
- Освітленість робочого місця є недостатньою. Це може бути викликане недостатньою кількістю штучних джерел світла на периметр робочого приміщення та відсутністю або недостатньою кількістю вікон в приміщення. Якщо вікон достатньо, але світла проникає мало, це може бути викликане тим, що біля приміщення ростуть дерева, які своєю кроною заважають світлу проникати в приміщення.
- Незадовільний мікроклімат в робочих приміщеннях. Це може бути викликане відсутністю або незадовільним станом вентиляції та

системи опалення. Сюди можна віднести недостатню або підвищену вологість повітря [24].

- Небезпека ураження електрострумом. Джерелом може бути несправний електроприлад чи пошкоджений дрiт [25].

- Можливість займання електроприладiв чи елементiв примiщення. Це може трапитись через несправнiсть або пошкодження електроприладiв та дротiв, через стрибок напруги або через людську необережнiсть або байдужiсть.

- Рiвень шуму в примiщеннi вище норми. Джерелом шуму можуть бути: транспорт поза межами примiщень, якщо це виробничий цех, то шум може створюватись вiд роботи устаткування, люди, ПК [26].

- Нервово-психiчне навантаження. Це бiльше стосується працiвникiв IT-вiддiлу. Джерелами цього можуть бути: однотипна монотонна робота, напруженiсть очей пiд час роботи за ПК, незручне робоче мiсце, однотипнi запитання та завдання, якi надходять у виглядi заявок [27].

Вплив, який цi фактори можуть здiйснювати на працiвникiв:

- Пiд час роботи за таким робочим мiсцем працiвник може вiдчувати дискомфорт та бiль у спинi, шиї.

- Погана освiтленiсть може сприяти зниженню працездатностi очей та викликати втому всього органiзму, що може вiдобразитися на працездатностi робiтника.

- Незадовiльний мiкроклiмат може впливати на теплове самопочуття спiвробiтникiв. При високiй температурi повітря в примiщеннi робiтник буде почувати себе некомфортно, а низька температура може викликати переохолодження органiзму.

- Ураження електричним струмом може викликати непритомнiсть, опiки та смерть через зупинку дихання та серця.

- Займання електроприладiв чи елементiв примiщення може зашкодити як i здоров'ю чи життю спiвробiтникiв, так i нанести вагомих збиткiв пiдприємству.

- Підвищений рівень шуму може спричиняти: зниження слуху, зниження виробничих показників, поганий сон, зниження уваги, головний біль, відчуття важкості, тривожність тощо.
- Нервово-психічне навантаження може викликати: навантаження на очі, головний біль, втому, роздратованість, стрес.

Можливі заходи передбачення виникнення небезпечних чинників:

- Організація робочого місця відповідно до норм. Це передбачає купівлю робочого столу, який відповідав би сучасним ергономічним вимогам і міг би забезпечувати достатньою кількістю робочої поверхні для розміщення на ньому клавіатури, миші, монітору, та робочої поверхні для сидіння: стільця чи крісла, які б відповідали усіх вимогам.
- Освітленість робочого місця можна підвищити, збільшивши кількість штучних джерел освітлення, або замінивши лампи на більш потужні в тих джерелах, які вже встановлені. Також, якщо це можливо та доцільно, в робочому приміщенні можна встановити додаткове вікно або вікна. Встановлення додаткових вікон також допоможе передбачити виникнення наступного чинника.
- Незадовільний мікроклімат в робочих приміщеннях. Це можна виправити за рахунок встановлення вентиляції, якщо вона відсутня та системи опалення. Так як в різні пори року комфортні температурні межі різні, то для теплих періодів потрібно передбачити встановлення кондиціонерів або вентиляторів, але все це повинно відповідати нормам. Вологість повітря можна підвищувати шляхом вологого прибирання з певною частотою або використовуючи відповідні технічні засоби.
- Небезпека ураження електрострумом. З цим пунктом повинні бути ознайомлені всі працівники підприємства, оскільки вони повинні проходити інструктажі, і повинні знати, як діяти в тих чи інших небезпечних ситуаціях, пов'язаних з електрострумом. Використання електричних приборів повинно відбуватись згідно правил та вимог.

- Цей пункт схожий з попереднім, правила тут подібні. На підприємстві та у відділах повинна бути налагоджена пожежна охоронна система, яка в разі небезпеки забезпечить найшвидше ліквідування проблеми.
- Рівень шуму в приміщенні вище норми. Якщо мова йдеться про робоче приміщення, яке знаходиться біля виробничих устаткувань, то є доцільним говорити про обладнання цього місця засобами, які поглинають шум. Якщо шум створює ПК чи інша техніка, то є доцільність перевірити цю техніку на випадок поломки.
- Нервово-психічне навантаження. До основних заходів запобігання цього чинника можна віднести: зміна режиму праці та відпочинку, виконання фізичних вправ для спини, очей та кровообігу протягом робочого дня, розділення роботи між кількома працівниками.

ВИСНОВКИ

Ця робота була присвячена проблемі обробки заявок на підприємстві «Христинівський молокозавод». Після аналізу роботи відділу та аналізу процесу обробки заявок було виявлено, що рівень автоматизації процесу є недостатнім, та були поставлені задачі автоматизації, які потребують реалізації.

Було проведено аналіз існуючих на ринку ІС, які могли б впоратись з поставленими задачами. Їх порівняння показало, що виростання цих систем на підприємстві є недоцільним. Тому було прийнято рішення розроблення нової ІС, яка б виконувала поставлені задачі та задовільнила вимоги відділу. Була виконана постановка задачі, під час якої були поставлені функції, які система повинна виконувати.

В другому розділі роботи було описано процес розробки підсистеми з самого початку. Розробка почалась з проектування бази даних: створення логічної та фізичної моделей даних та генерація фізичної моделі в готову БД в СУБД Microsoft SQL Server. На основі розробленої БД було створено інтерфейс користувача.

Розроблена підсистема та її функції дозволяє реалізувати задачі автоматизації з першого розділу. Підсистема надала користувачам нових можливостей, які були відсутні до цього. Це, в першу чергу, можливість збереження даних про створені заявки в БД, можливість створювати та переглядати заявки безпосередньо зі свого робочого місця, збереження даних дає змогу в подальшому їх обробляти, переглядати звіти, мати повну картину про роботу, яку виконують працівники ІТ відділу та про те, з чим частіше всього працівники мають проблеми.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Система обробки заявок [Електронний ресурс] – Режим доступу до ресурсу: <https://okdesk.ru/blog/communications>.
2. Христинівський молокозавод [Електронний ресурс] – Режим доступу до ресурсу: <https://www.stockworld.com.ua/ru/profile/company/543297375dfc9db7410073f0>.
3. Про ПАТ "Христинівський молокозавод" [Електронний ресурс] – Режим доступу до ресурсу: <https://khrystynivskiy-molokozavod.business-guide.com.ua/>.
4. Ситуація на підприємстві "Христинівський молокозавод" [Електронний ресурс] – Режим доступу до ресурсу: <https://procherk.info/news/7-cherkassy/63356-hristinivskij-molokozavod-nevdovzi-ogolosjat-bankrutom>.
5. Проектування інформаційних систем [Електронний ресурс]: лабораторний практикум для здобувачів освітнього ступеня "Бакалавр" спеціальності 122 "Комп'ютерні науки" денної та заочної форм навчання Модуль 1. "Функціональне моделювання та моделювання баз даних" уклад.: О. М. М'якило, О. В. Харкянен: НУХТ, 2018. – 60 с.
6. OTRS [Електронний ресурс] – Режим доступу до ресурсу: <https://ru.wikipedia.org/wiki/OTRS>.
7. Переваги OTRS [Електронний ресурс] – Режим доступу до ресурсу: <https://otrs.com/lp/sme-software/>.
8. OTRS Official documentation [Електронний ресурс] – Режим доступу до ресурсу: <https://doc.otrs.com/doc/manual/user/8.0/en/content/index.html>.
9. Опис підсистеми технічної підтримки на базі [Електронний ресурс] – Режим доступу до ресурсу: https://xn--90afdtkhdeabaxvge.net/funkcionalnie_vozmozhnosti/helpdesk/opisanie_podsistemi_teh_podderzhki.html.

10. Автоматизація технічної підтримки [Електронний ресурс] – Режим доступу до ресурсу: <https://1c-helpdesk.ru/>.
11. OTRS cost [Електронний ресурс] – Режим доступу до ресурсу: <https://www.getapp.com/resources/alternatives-to-otrs/>.
12. 1С Підприємство 8 Документообіг КОРПІ для України [Електронний ресурс] – Режим доступу до ресурсу: <http://bookkeeping.com.ua/1c-enterprise-8-document-circulation-ukr.html>.
13. Проектування інформаційних систем [Електронний ресурс] – Режим доступу до ресурсу: https://elearning.sumdu.edu.ua/free_content/lectured:de1c9452f2a161439391120eef364dd8ce4d8e5e/20151118130027/183252/index.html.
14. Проектування інтерфейсу користувача [Електронний ресурс] – Режим доступу до ресурсу: <http://lib.chdu.edu.ua/pdf/posibnuku/313/7.pdf>.
15. Проектування інформаційних систем. [Електронний ресурс]: лабораторний практикум для студентів освітнього ступеню "бакалавр" спец. 122 "Комп'ютерні науки " денної і заочної форм навчання. Частина 2 "Проектування клієнтського додатку" / Уклад.: О.М. М'яшило, О.В. Харкянен – К.: НУХТ, 2017 – 33 с.
16. UserControl Class [Електронний ресурс] – Режим доступу до ресурсу: <https://docs.microsoft.com/enus/dotnet/api/system.windows.forms.usercontrol?view=netcore-3.1>.
17. Panel Class [Електронний ресурс] – Режим доступу до ресурсу: <https://docs.microsoft.com/enus/dotnet/api/system.windows.forms.panel?view=netcore-3.1>.
18. Using COLLATE in SQL Statements [Електронний ресурс] – Режим доступу до ресурсу: <https://dev.mysql.com/doc/refman/5.7/en/charset-collate.html>.
19. VS 2019 RDLC Report Designer [Електронний ресурс] – Режим доступу до ресурсу:

<https://developercommunity.visualstudio.com/content/problem/418987/vs-2019-rdlc-report-designer.html>.

20. BindingNavigator Class [Електронний ресурс] – Режим доступу до ресурсу: <https://docs.microsoft.com/en-us/dotnet/api/system.windows.forms.bindingnavigator?view=netcore-3.1>.

21. Управління ІТ проектами [Електронний ресурс]: методичні рекомендації до виконання курсової роботи для студентів напряму підготовки 6.050101«Комп'ютерні науки» денної та заочної форм навч. / уклад. М. В. Гладка, О. А. Хлобистова. – К. : НУХТ, 2014.– 91 с. – Режим доступу: <http://library.nuft.edu.ua/ebook/file/51.13.pdf>.

22. Охорона праці [Електронний ресурс] – Режим доступу до ресурсу: <https://bit.ly/2XM7oD0>.

23. Навчання з електробезпеки для працівників ІТ-відділу [Електронний ресурс] – Режим доступу до ресурсу: <https://esop.mcfrr.ua/588685?from=id2&token=24cd39e4-bcaa-11a0-e17d-2d01a254f112&ttl=7776000&ustp=W>.

24. Основи фізіології та гігієни праці [Електронний ресурс] – Режим доступу до ресурсу: <https://bit.ly/36IDIE6>.

25. Небезпека ураження електрострумом [Електронний ресурс] – Режим доступу до ресурсу: <http://www.ztoe.com.ua/news.php?id=502>.

26. Шум на виробництві та його вплив на організм людини [Електронний ресурс] – Режим доступу до ресурсу: <http://oppb.com.ua/news/shum-na-vyrobnyctvi-ta-yogo-vplyv-na-organizm-lyudyny>.

27. Психофізіологічне перевантаження на роботі [Електронний ресурс] – Режим доступу до ресурсу: <https://bit.ly/3gCRt2B>.

ДОДАТКИ

Додаток А. Функціональні моделі

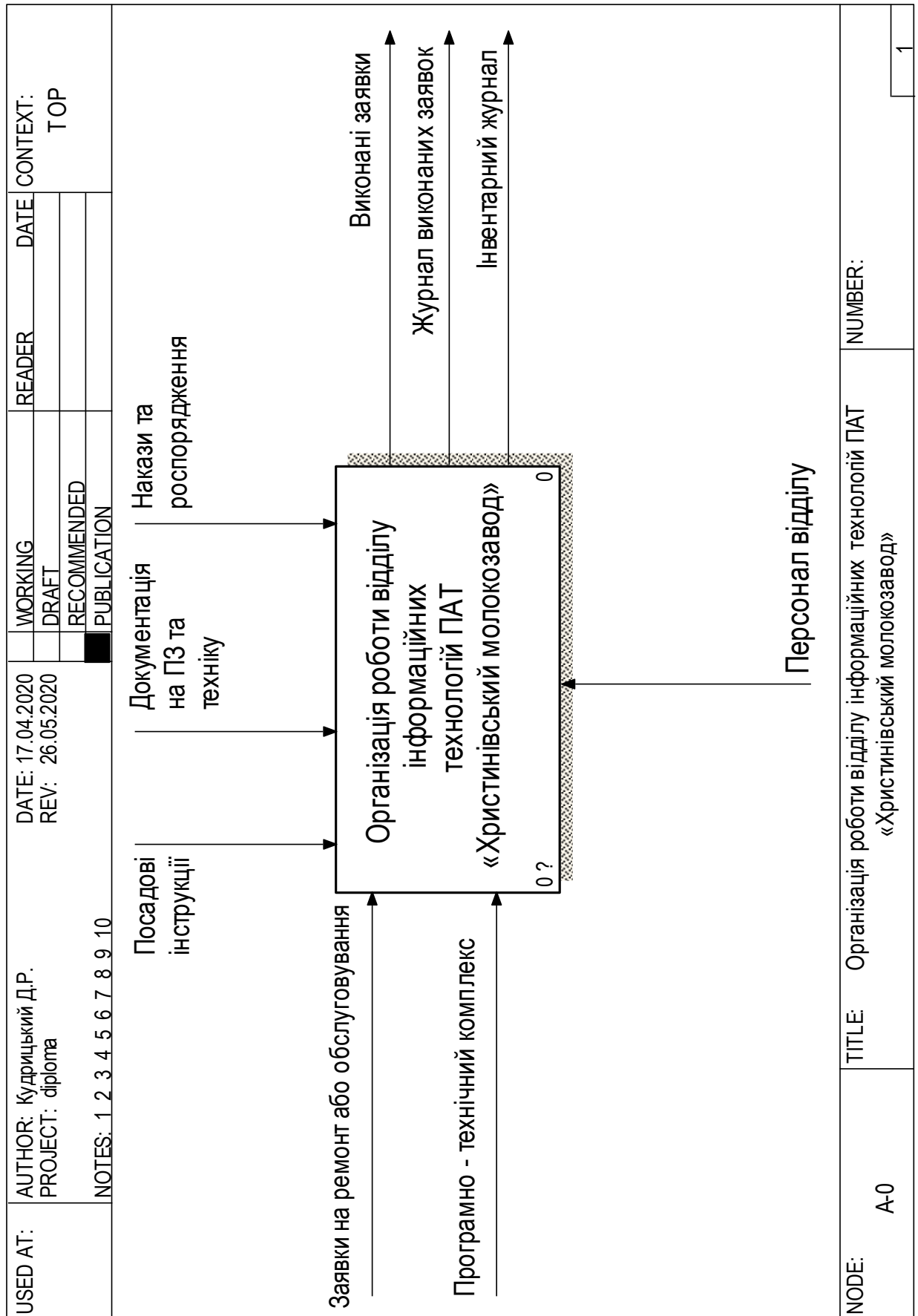


Рис. 1. Контекстна діаграма функціональної моделі

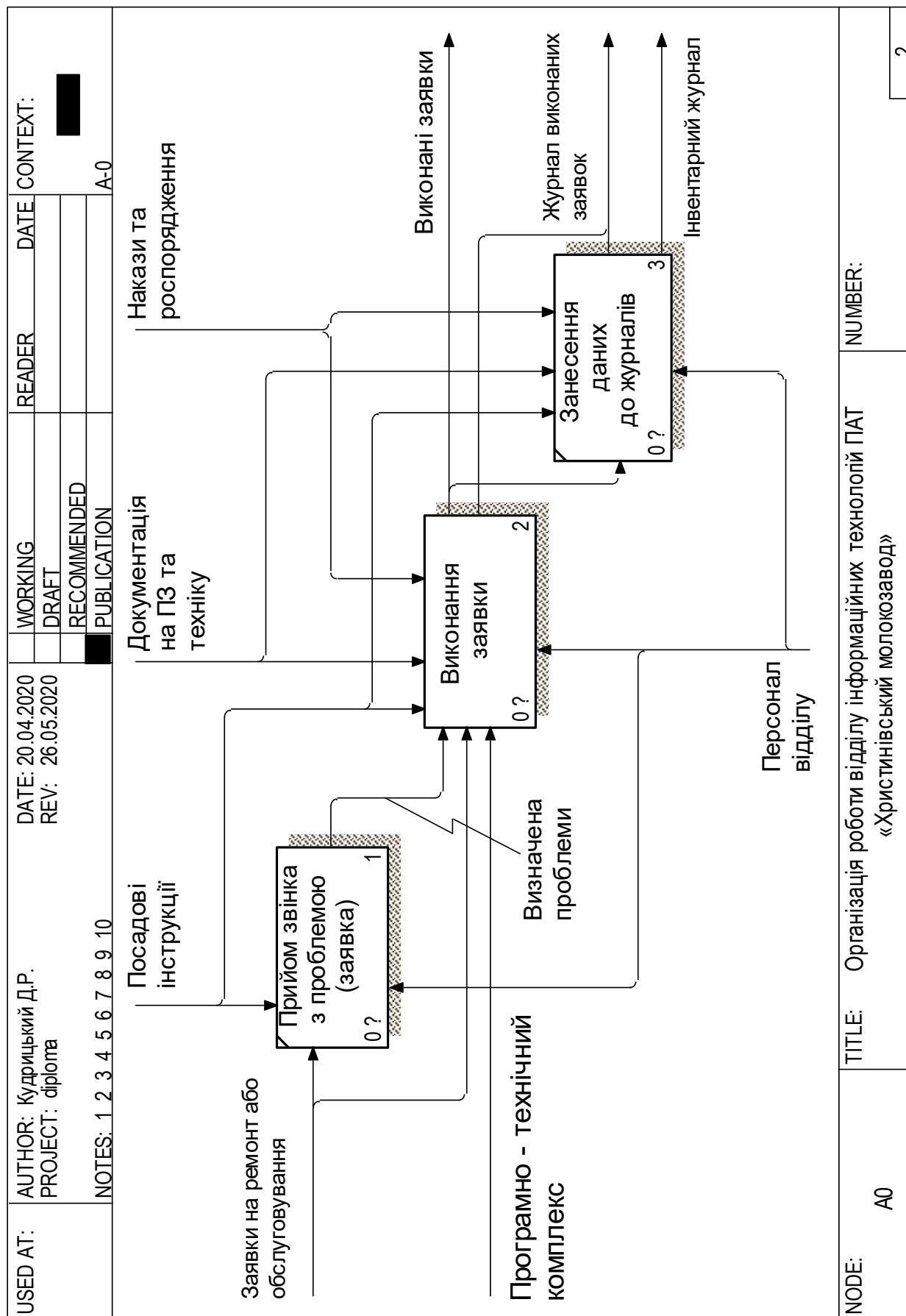


Рис. 2. Перший рівень декомпозиції контекстної діаграми

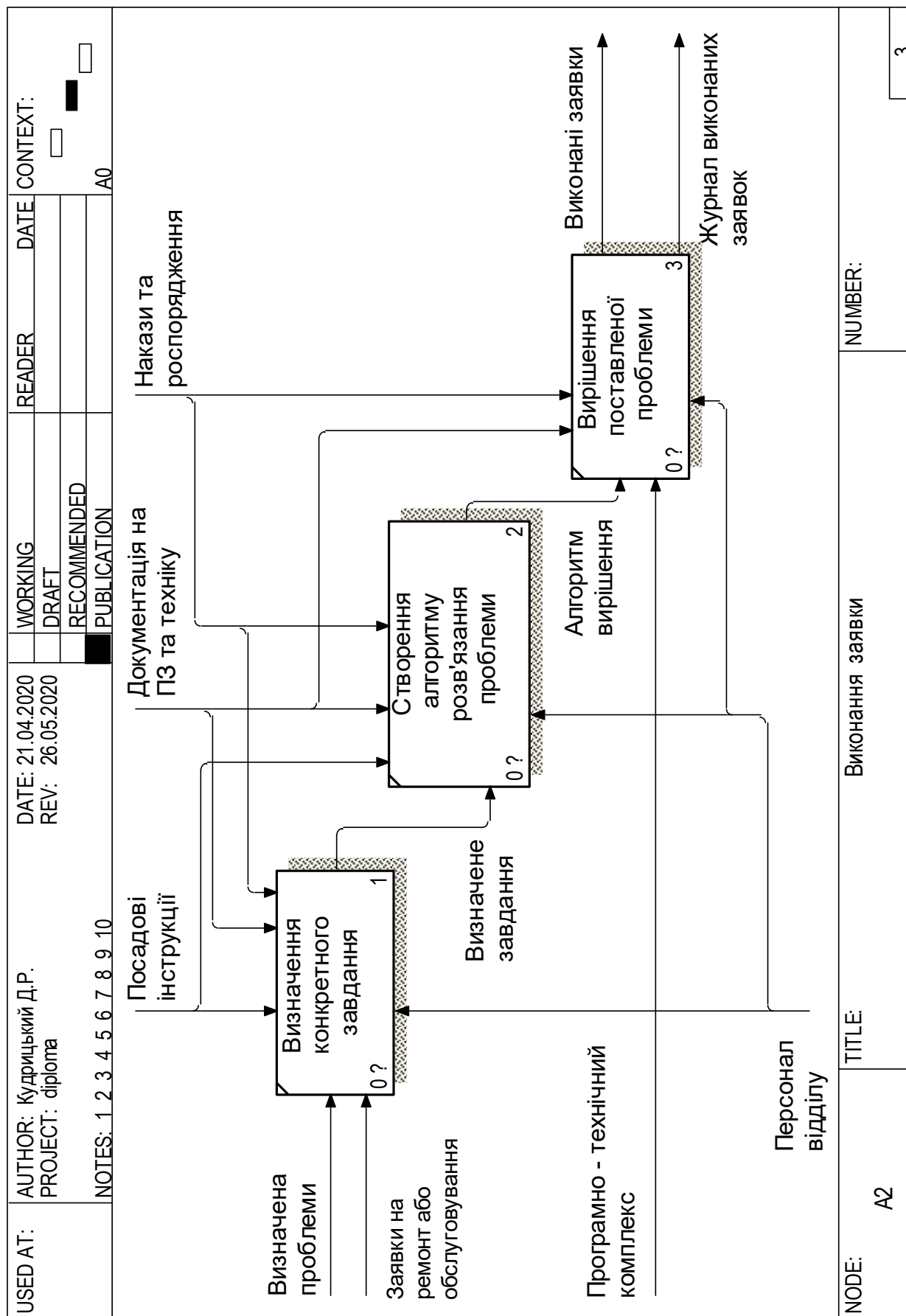


Рис. 3. . Другий рівень декомпозиції контекстної діаграми

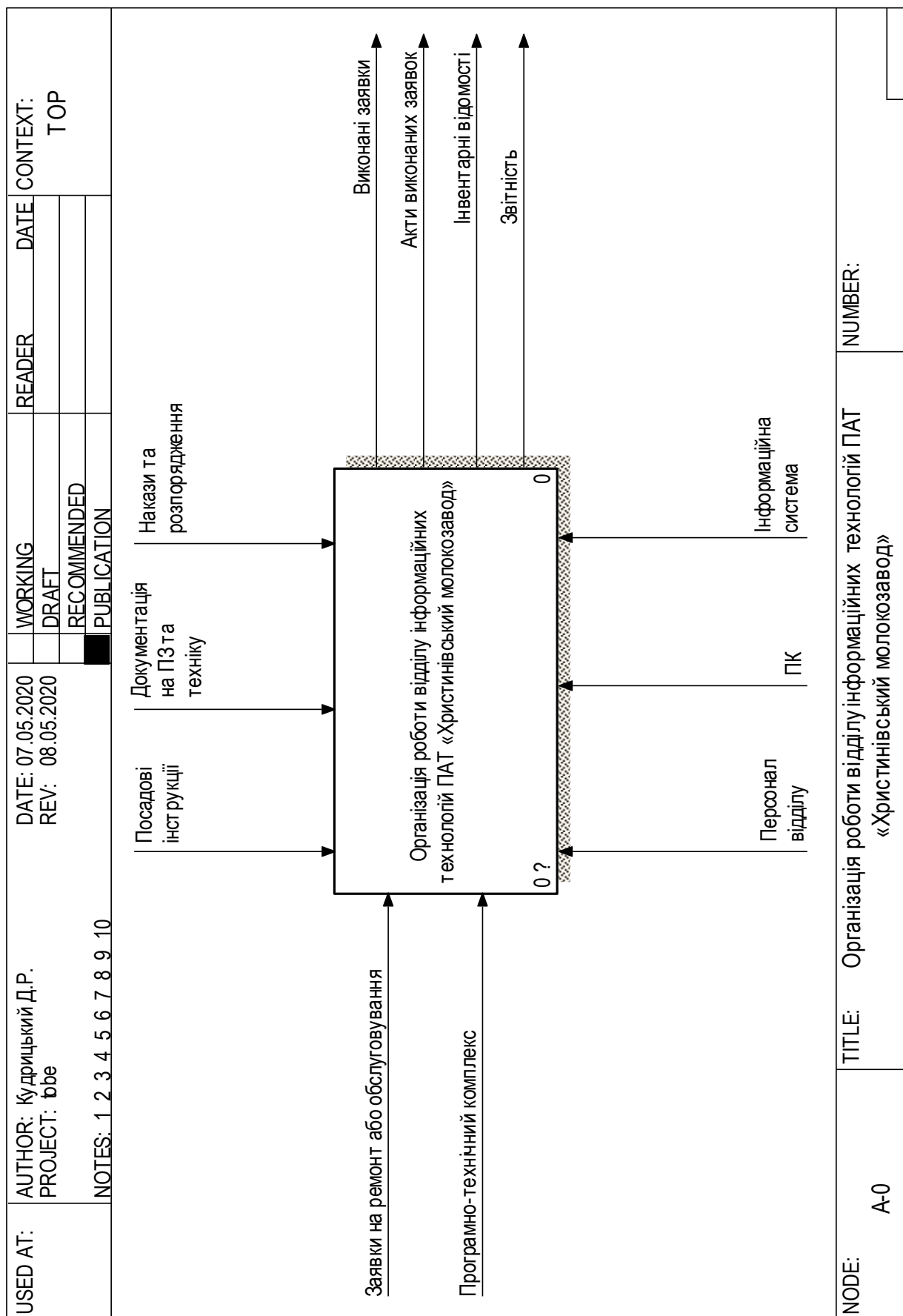


Рис. 4. Концептуальна модель

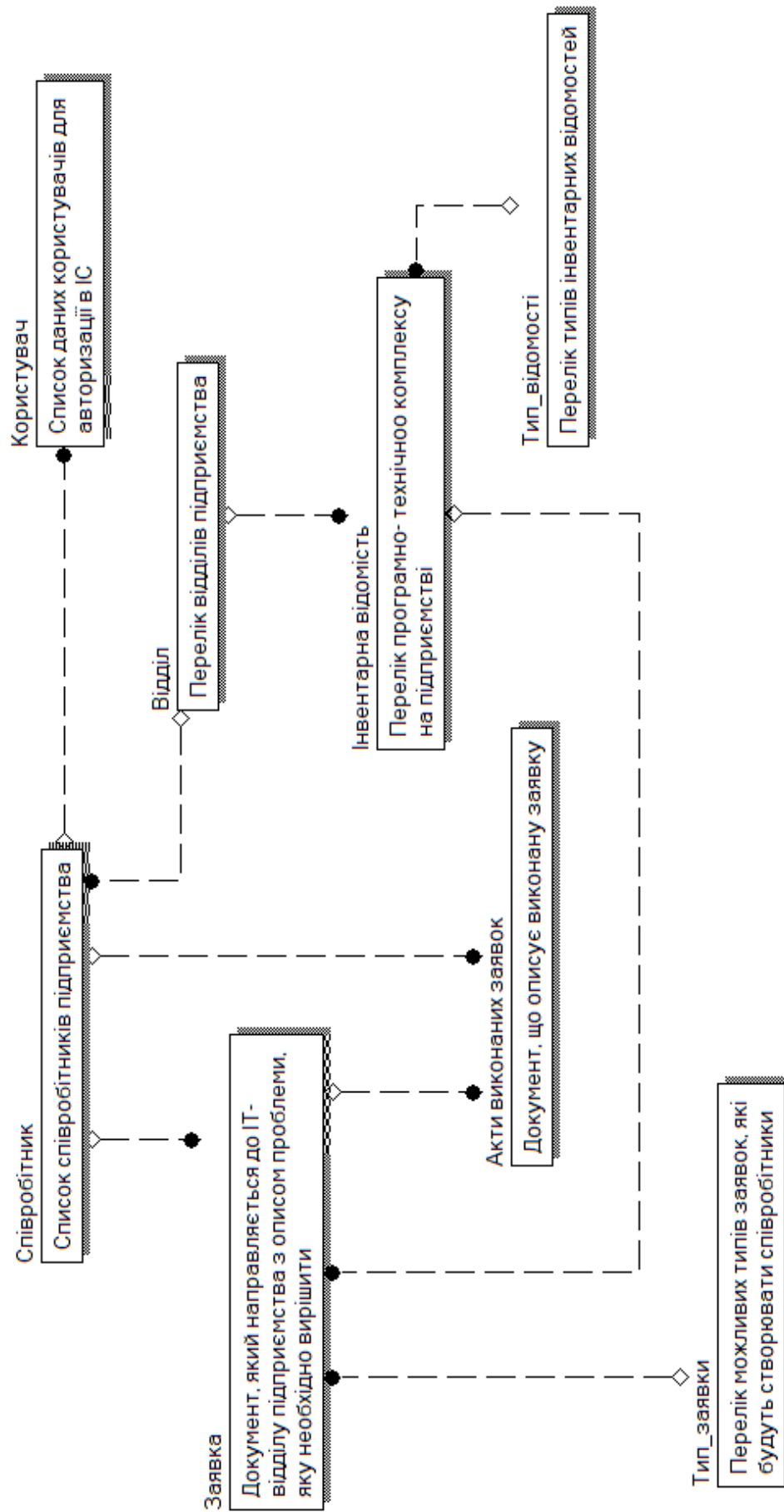


Рис. 6. Логічна модель даних

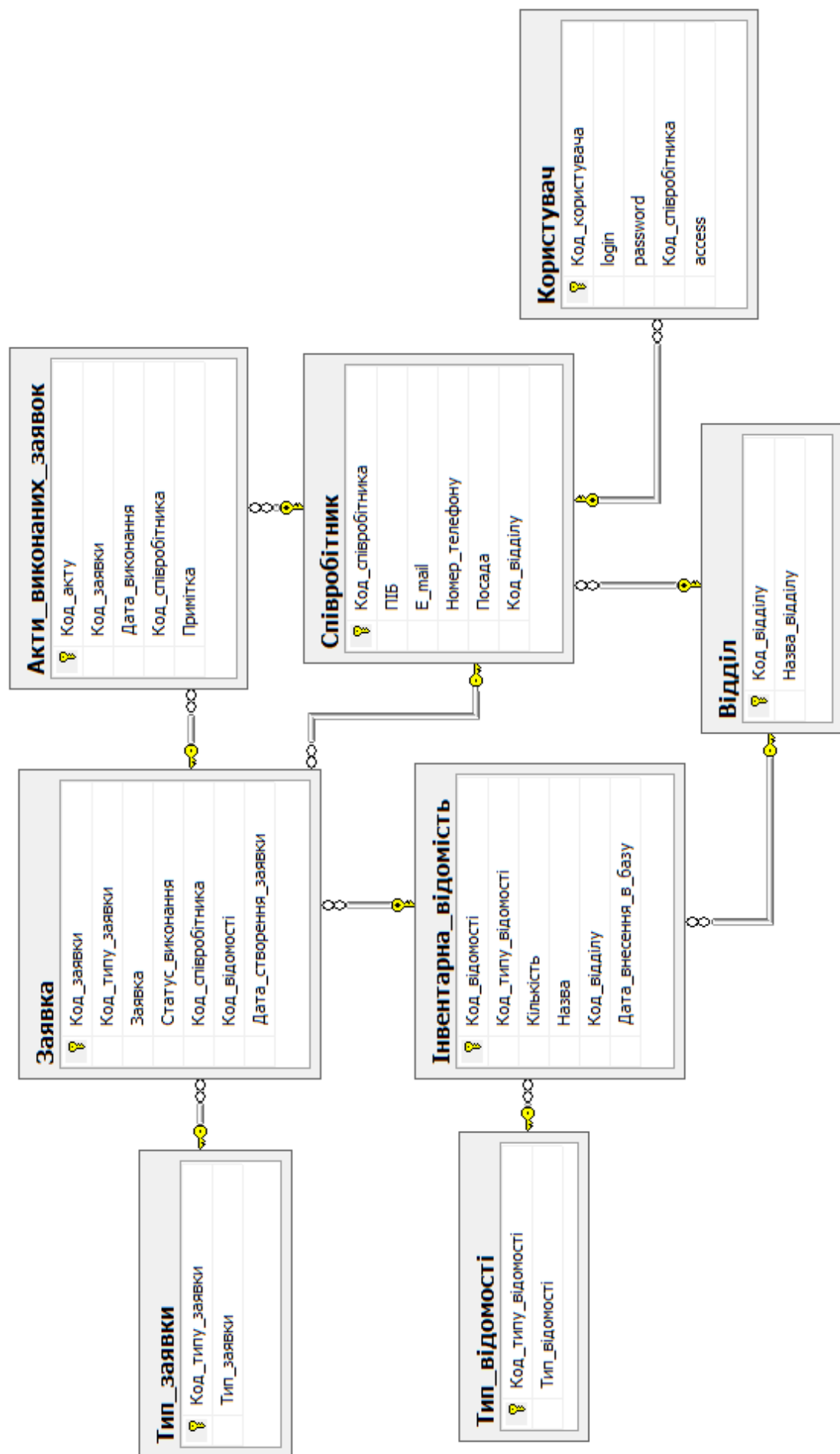


Рис. 8. Згенерована БД в MS SQL Server

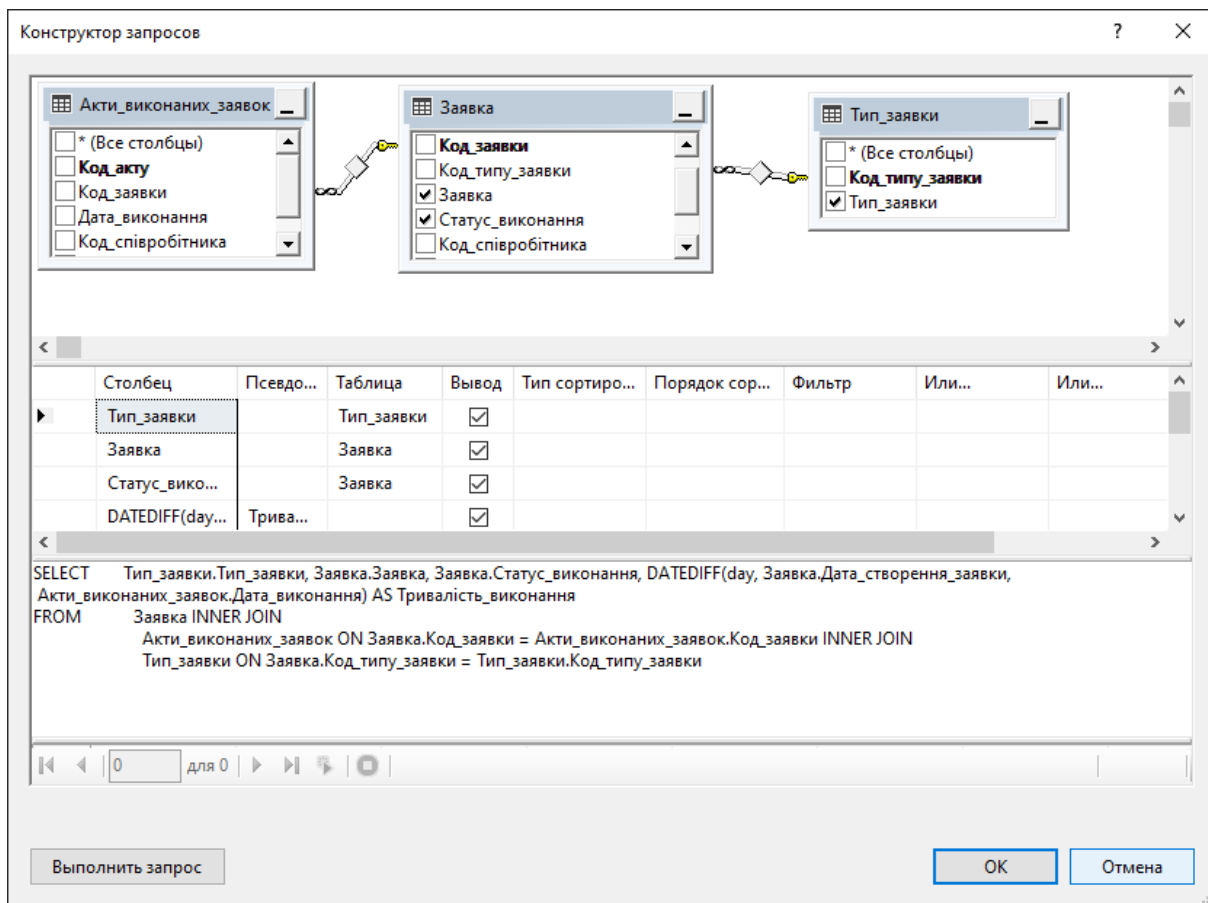


Рис. 9. Запит для обчислення тривалості виконання заявок

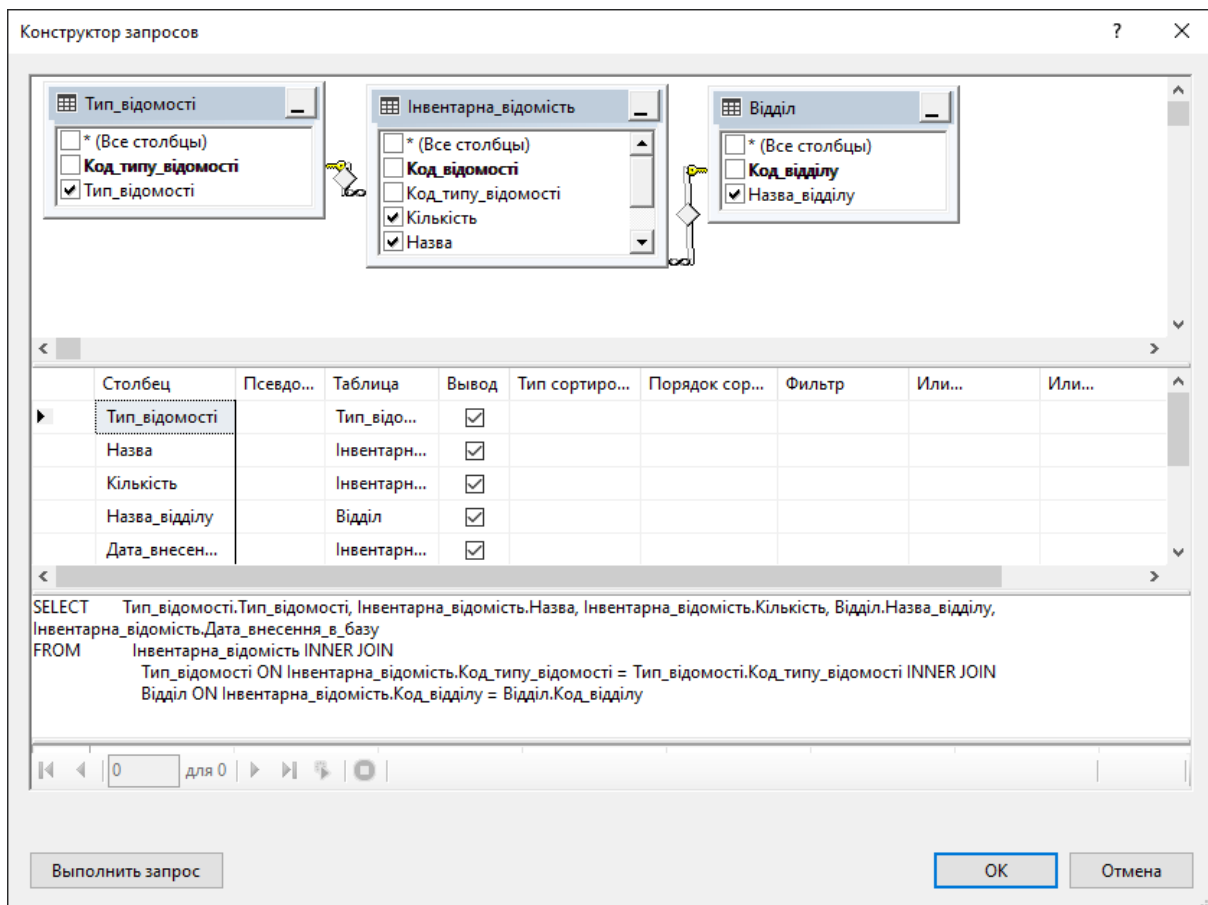


Рис. 10. Запит для відображення всіх інвентарних відомостей

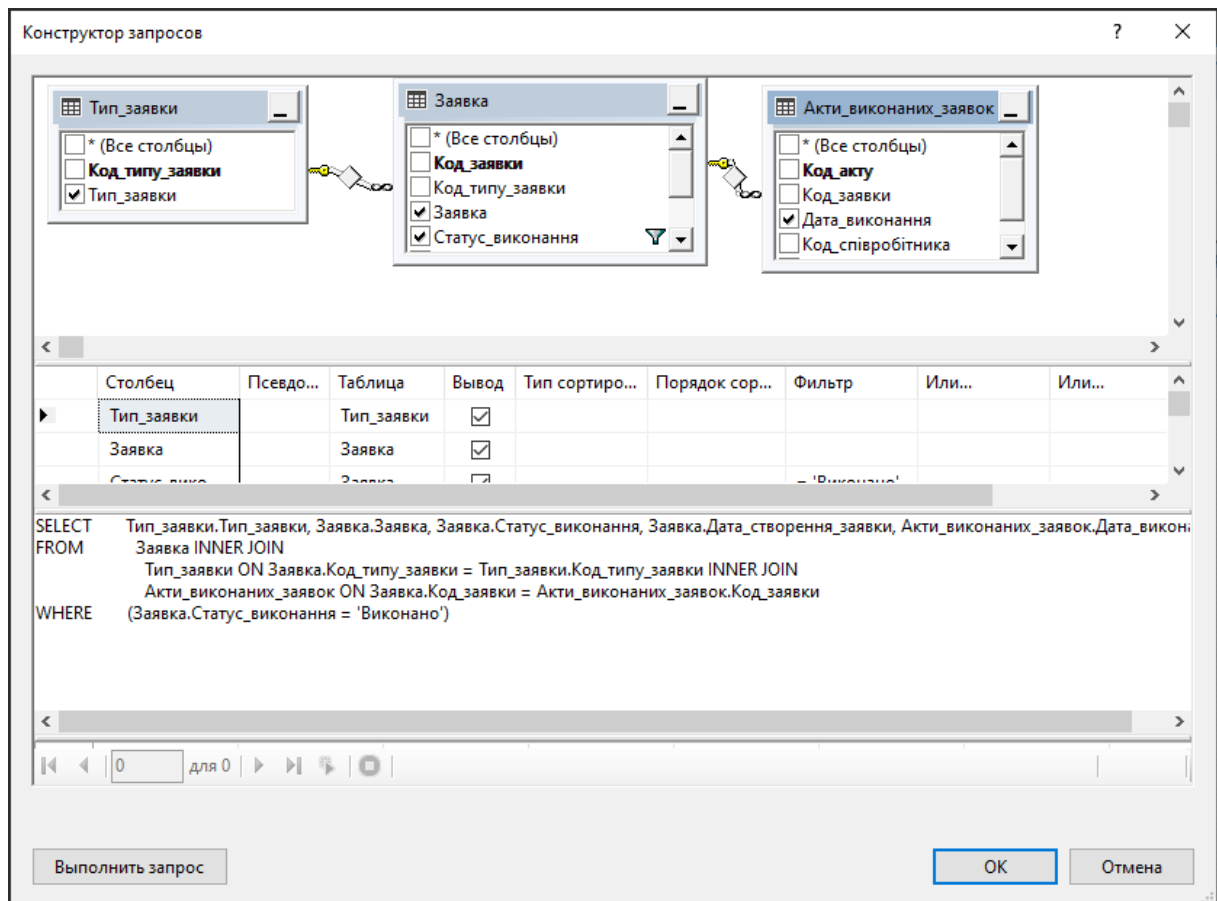


Рис. 11. Запит для відображення всіх виконаних заявок

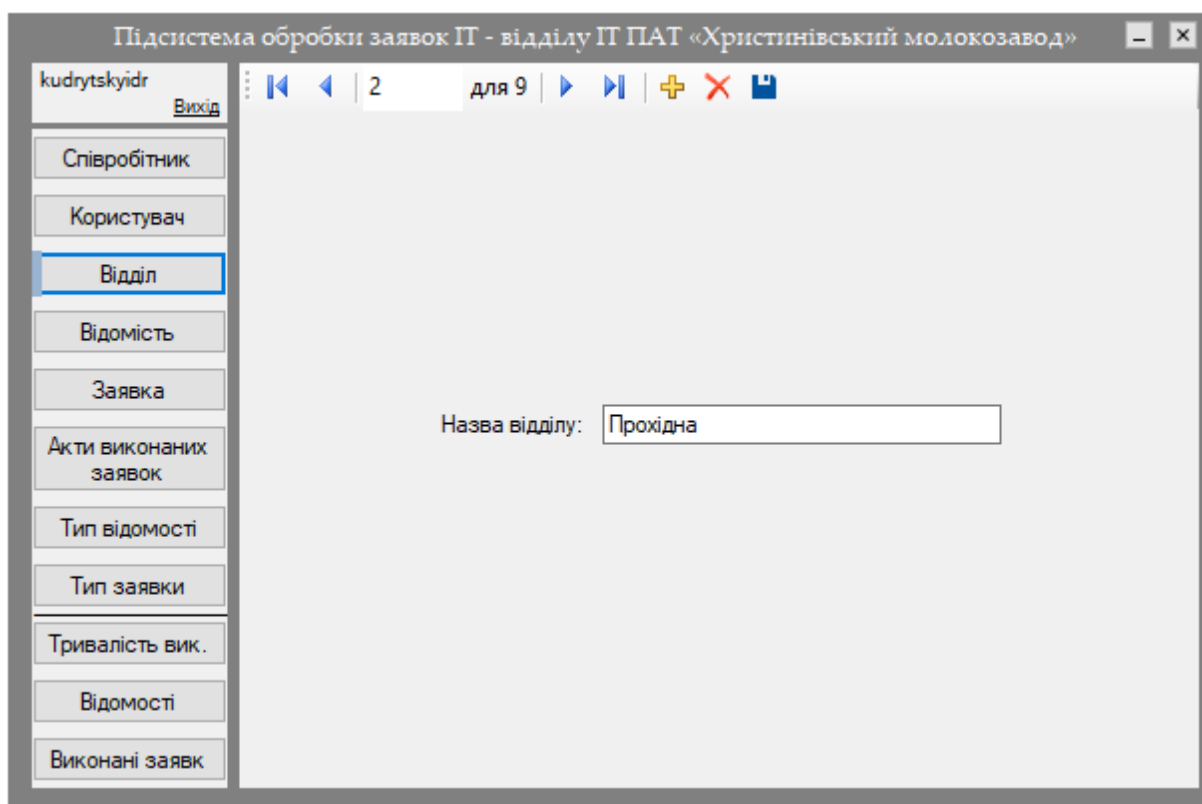


Рис. 12. Сторінка для вводу, перегляду, редагування та видалення даних таблиці «Відділ»

Підсистема обробки заявок ІТ - відділу ІТ ПАТ «Христинівський молокозавод»

kudrytskyidr [Вихід](#)

Співробітник

Користувач

Відділ

Відомість

Заявка

Акти виконаних заявок

Тип відомості

Тип заявки

Тривалість вик.

Відомості

Виконані заявок

3 для 12

Тип відомості: Програмне забезпечення

Кількість: 1

Назва: Роутер

Відділ: Бухгалтерія

Дата внесення в базу: 19 января 2020 г.

Рис. 13. Сторінка для вводу, перегляду, редагування та видалення даних таблиці «Відомість»

Підсистема обробки заявок ІТ - відділу ІТ ПАТ «Христинівський молокозавод»

kudrytskyidr [Вихід](#)

Співробітник

Користувач

Відділ

Відомість

Заявка

Акти виконаних заявок

Тип відомості

Тип заявки

Тривалість вик.

Відомості

Виконані заявок

4 для 15

Тип заявки: Ремонт

Заявка: Не працює

Статус виконання: В роботі

Хто створив заявку: Красножор Іван Андрійович

Відомість: Монітор

Дата створення заявки: 29 апреля 2020 г.

Ви можете знайти потрібну заявку, обравши її з випадального списку:

Не працює

Рис. 14. Сторінка для вводу, перегляду, редагування та видалення даних таблиці «Заявка»

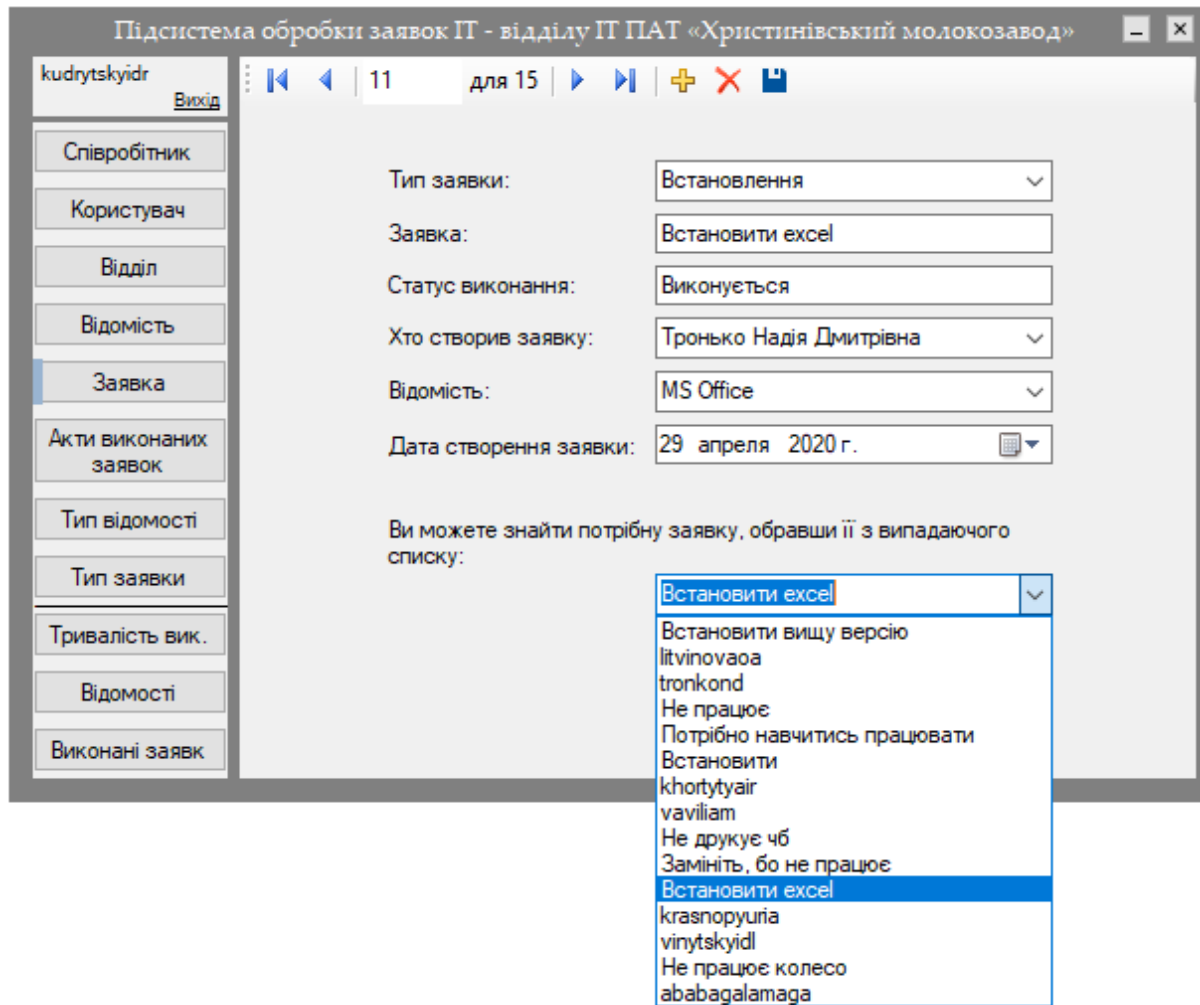


Рис. 15. Реалізована функція пошуку по заявці на сторінці перегляду таблиці «Заявка»

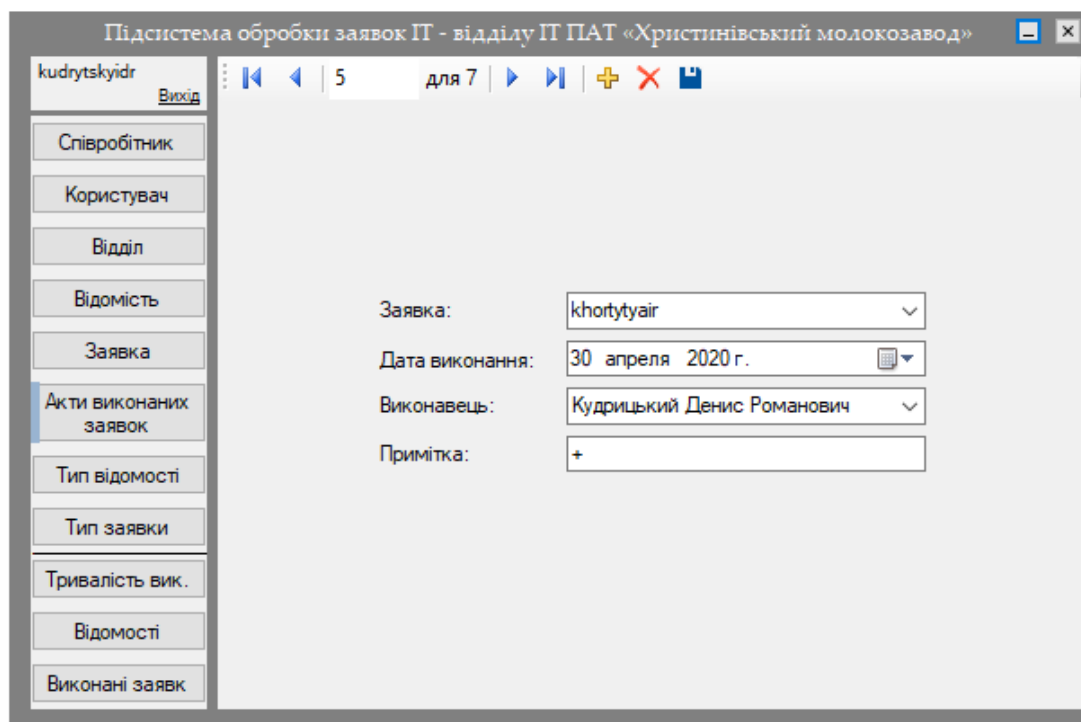


Рис. 16. Сторінка для вводу, перегляду, редагування та видалення даних таблиці «Акти виконаних заявок»

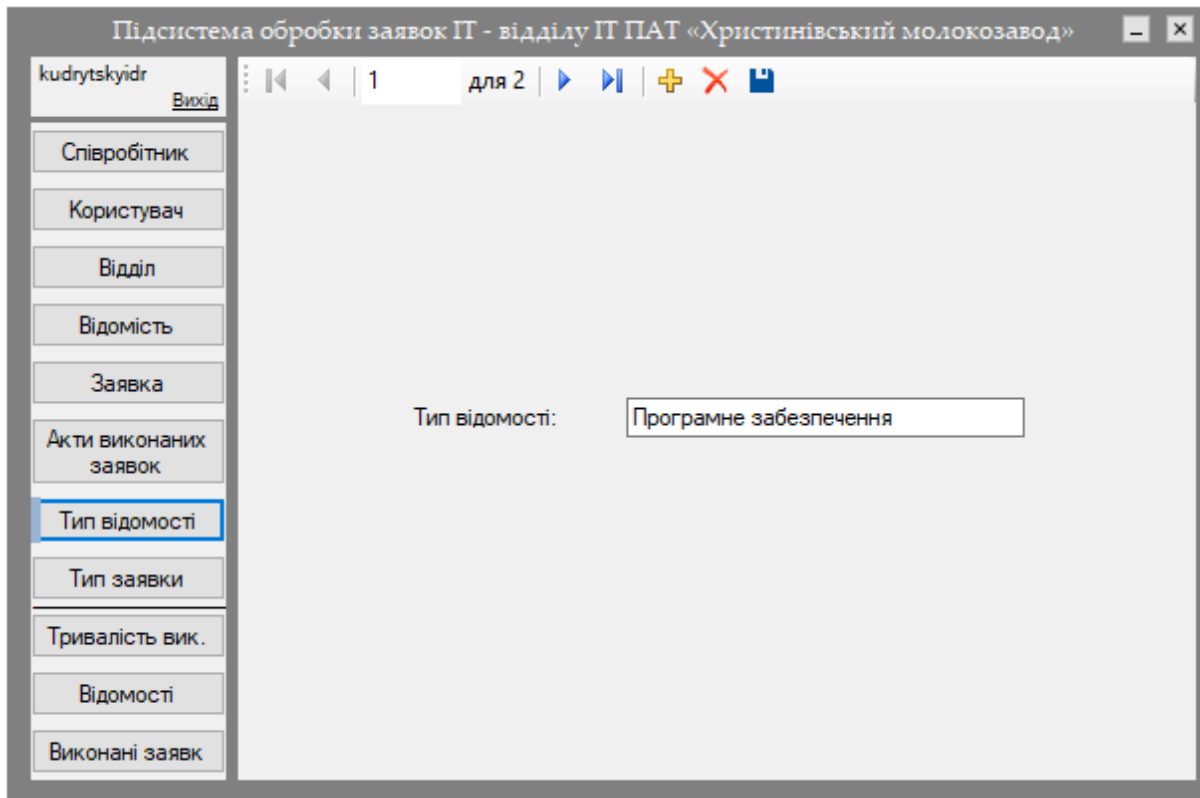


Рис. 17. Сторінка для вводу, перегляду, редагування та видалення даних таблиці «Тип відомості»

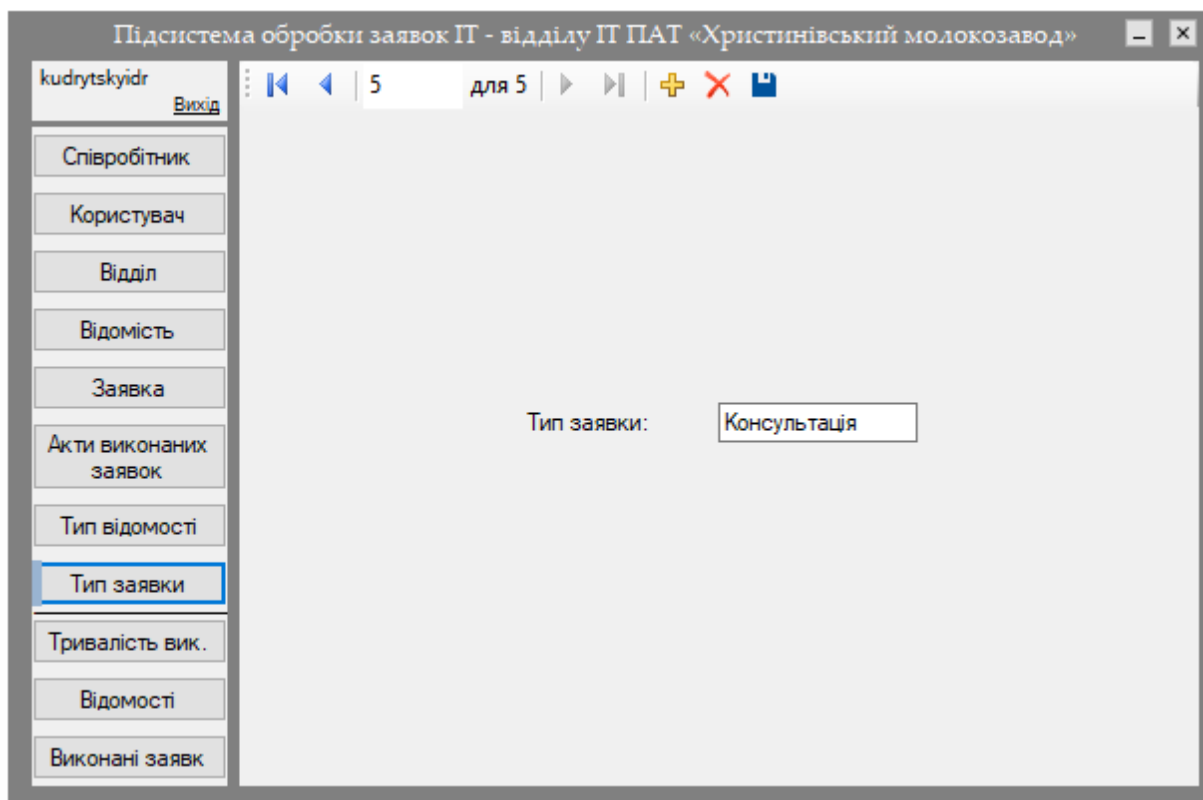


Рис. 18. Сторінка для вводу, перегляду, редагування та видалення даних таблиці «Тип відомості»

Підсистема обробки заявок ІТ - відділу ІТ ПАТ «Христинівський молокозавод»				
kudrytskyidr	Вихід			
Співробітник				
Користувач				
Відділ				
Відомість				
Заявка				
Акти виконаних заявок				
Тип відомості				
Тип заявки				
Тривалість вик.				
Відомості				
Виконані заявк				

Тип відомості	Назва	Кількість	Назва відділу	Дата внесення в базу
Програмне забезпечення	MS Office	1	Бухгалтерія	18.01.2020 0:00:00
Техніка	Мишка	2	Бухгалтерія	18.01.2020 0:00:00
Програмне забезпечення	Роутер	1	Бухгалтерія	19.01.2020 0:00:00
Техніка	Монітор	1	Прохідна	16.01.2020 0:00:00
Програмне забезпечення	MS Office	2	Відділ кадрів	24.01.2020 0:00:00
Техніка	Принтер	1	Відділ ІТ	22.01.2020 0:00:00
Техніка	Клавіатура	2	Оклад	
Техніка	Системний блок	1	Маслоцех	20.01.2020 0:00:00
Техніка	Роутер	1	Управлюючий відділ	03.02.2020 0:00:00
Програмне забезпечення	Expert PRO 2020	2	Лабораторія	05.02.2019 0:00:00
Техніка	Hub	2	Серверна	07.02.2020 0:00:00
Програмне забезпечення	IT super pro	1	Відділ ІТ	10.02.2020 0:00:00

Рис. 19. Сторінка для перегляду звіту «Відомості»

Підсистема обробки заявок ІТ - відділу ІТ ПАТ «Христинівський молокозавод»				
kudrytskyidr	Вихід			
Співробітник				
Користувач				
Відділ				
Відомість				
Заявка				
Акти виконаних заявок				
Тип відомості				
Тип заявки				
Тривалість вик.				
Відомості				
Виконані заявк				

Тип заявки	Заявка	Статус виконання	Дата створення заявки	Дата виконання
Заміна	Встановити вищу версію	Виконано	07.02.2020 0:13:02	11.02.2020 0:00:00
Пароль	litvinovoa	Виконано	22.02.2020 0:13:02	22.02.2020 0:00:00
Пароль	tronkond	Виконано	06.03.2020 0:13:02	09.03.2020 0:00:00
Консультація	Потрібно навчитись працювати	Виконано	01.05.2020 18:35:29	04.05.2020 0:00:00
Пароль	khortytysair	Виконано	27.04.2020 22:20:27	30.04.2020 0:00:00
Пароль	aviliam	Виконано	16.04.2020 22:20:27	16.04.2020 0:00:00
Пароль	kras poruyria	Виконано	03.04.2020 19:50:17	08.04.2020 0:00:00

Рис. 20. Сторінка для перегляду звіту «Виконані заявки»

Код процедури для можливості переміщення вікна по робочому столу:

```
public const int WM_NCLBUTTONDOWN = 0xA1;
public const int HT_CAPTION = 0x2;
[DllImportAttribute("user32.dll")]
public static extern int SendMessage(IntPtr hWnd,
int Msg, int wParam, int lParam);
[DllImportAttribute("user32.dll")]
public static extern bool ReleaseCapture();
private void Form1_MouseDown(object sender,
MouseEventArgs e)
{
    if (e.Button == MouseButtons.Left)
    {
        ReleaseCapture();
        SendMessage(Handle, WM_NCLBUTTONDOWN,
HT_CAPTION, 0);
    }
}
```

Процедури натиску кнопки в користувацькому меню:

```
private void button2_Click(object sender, EventArgs e)
{
    SidePanel.Height = button2.Height;
    SidePanel.Top = button2.Top;
    if
(!Form1.Instance.PnlContainer.Controls.ContainsKey("adm_vid_en
ter"))
    {
        adm_vid_enter ave = new adm_vid_enter();
        ave.Dock = DockStyle.Fill;

Form1.Instance.PnlContainer.Controls.Add(ave);
    }

Form1.Instance.PnlContainer.Controls["adm_vid_enter"].BringToF
ront();
}
```

Повний код обробки натискання кнопки «Увійти»:

```
private void enterbtn_Click(object sender, EventArgs e)
{
    if (logintb.Text.Length > 0 &
passtb.Text.Length > 0)
    {
        string login = logintb.Text;
```

```

        string pass = passtb.Text;
        string myConnectionString = @"Data Source
=(local)\SQLDEN; Initial Catalog = itdep; Integrated
Security=True";

        string mySelectQuery = "SELECT
[login],[password],[access] FROM [Копистываю] WHERE [login]
COLLATE Cyrillic_General_CS_AI = '" + login + "'and [password]
COLLATE Cyrillic_General_CS_AI ='" + pass + "'";
        using (SqlDataAdapter dataAdapter = new
SqlDataAdapter(mySelectQuery, myConnectionString))
        {
            DataTable table = new DataTable();
            dataAdapter.Fill(table);
            if (table.Rows.Count == 0) //т.е. если
не нашел совпадающего по логину и паролю в Table будет 0 строк
            {
                //MessageBox.Show("нечего");
                label5.Visible = false;
                label4.Visible = true;
                linkLabel1.Visible = true;
                linkLabel1.BringToFront();
                logintb.Text = passtb.Text = "";
            }
            else
            {
                string lgg =
table.Rows[0][0].ToString();
                Form1.Instance.lglabel.Visible =
true;
                Form1.Instance.lglabel.Text =
lgg;

                Form1.Instance.LogOutButton.Visible = true;
                string aaa =
table.Rows[0][2].ToString();
                int acs = Int32.Parse(aaa);
                if (acs == 1)
                {
                    if
(!Form1.Instance.PnlContainer2.Controls.ContainsKey("adm_panel
2"))
                    {
                        adm_panel2 ap = new
adm_panel2();
                        ap.Dock = DockStyle.Fill;

                        Form1.Instance.PnlContainer2.Controls.Add(ap);
                    }
                }
            }
        }
    }
}

```

```

Form1.Instance.PnlContainer2.Controls["adm_panel2"].BringToFront();

        if
        (!Form1.Instance.PnlContainer.Controls.ContainsKey("adm_spivr_user_enter"))
        {
            adm_spivr_user_enter asue
= new adm_spivr_user_enter();
            asue.Dock
=
DockStyle.Fill;

Form1.Instance.PnlContainer.Controls.Add(asue);
        }

Form1.Instance.PnlContainer.Controls["login"].Dispose();

Form1.Instance.PnlContainer.Controls["adm_spivr_user_enter"].BringToFront();
        }
        else
        {
            if
            (!Form1.Instance.PnlContainer2.Controls.ContainsKey("panel2_users"))
            {
                panel2_users pu = new
panel2_users();
                pu.Dock = DockStyle.Fill;

Form1.Instance.PnlContainer2.Controls.Add(pu);
            }

Form1.Instance.PnlContainer2.Controls["panel2_users"].BringToFront();

            if
            (!Form1.Instance.PnlContainer.Controls.ContainsKey("users_create"))
            {
                users_create uc = new
users_create();
                uc.Dock = DockStyle.Fill;

Form1.Instance.PnlContainer.Controls.Add(uc);
            }

```

```

Form1.Instance.PnlContainer.Controls["login"].Dispose();

Form1.Instance.PnlContainer.Controls["users_create"].BringToFront();

        }
    }
}
else
{
    logintb.Text = passtb.Text = "";
    linkLabel1.Visible = false;
    label4.Visible = false;
    label5.Visible = true;
    label5.Text = "Потрібно ввести логін та
пароль";
}
}

```

Код обробки того, що заявка не може бути створена і надіслана, якщо поле введення є пустим:

```

private void textBox1_TextChanged(object sender, EventArgs e)
{
    if (textBox1.Text.Length == 0)
    {
        button1.Enabled = false;
    }
    else
        button1.Enabled = true;
}
}

```

Процедура натиснення кнопки «Вихід»:

```

private void logoutbtn_LinkClicked(object sender,
LinkLabelLinkClickedEventArgs e)
{
    var vs =
Form1.Instance.PnlContainer.Controls["users_create"];
    login lg = new login();
    lg.Dock = DockStyle.Fill;
    panell1.Controls.Add(lg);
    if
(!Form1.Instance.PnlContainer2.Controls.ContainsKey("login"))
{
        panell1.Controls["login"].BringToFront();
    }
}

```

```

        }
        if
(Form1.Instance.PnlContainer2.Controls.ContainsKey("panel2_use
rs") )
        {
            panel2.Controls["panel2_users"].Dispose();
        }
        if
(Form1.Instance.PnlContainer2.Controls.ContainsKey("adm_panel2
"))
        {
            panel2.Controls["adm_panel2"].Dispose();
        }
        logoutbtn.Visible = false;
        loginlb.Visible = false;
    }

```