

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний університет кораблебудування імені адмірала Макарова

Навчально-науковий інститут комп'ютерних наук та управління проектами

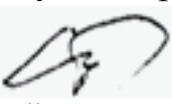
Кафедра програмного забезпечення автоматизованих систем

Спеціальність 121 «Інженерія програмного забезпечення»

Освітня програма «Інженерія програмного забезпечення»

«Допущений до захисту»

Завідувач кафедри

—  проф. Приходько С.Б.

«___» _____ 2025 р.

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття ступеня вищої освіти «бакалавр»

на тему: **Розробка програмного забезпечення для агентства з підбору кадрів**

«Новий шанс»

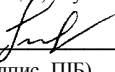
Виконав: студент групи 4151

—  Бобух М.В.
(підпис, ПІБ)

Керівник роботи:

доцент кафедри ПЗАС, к.т.н., доцент

(посада, науковий ступень вчене звання)

—  Макарова Л. М.
(підпис, ПІБ)

Миколаїв – 2025 р.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет кораблебудування імені адмірала Макарова

Навчально-науковий інститут комп'ютерних наук та управління проєктами
 Кафедра програмного забезпечення автоматизованих систем
 Спеціальність 121 «Інженерія програмного забезпечення»
 Освітня програма «Інженерія програмного забезпечення»

«ЗАТВЕРДЖУЮ»



Гарант освітньої програми

доц. Макарова Л.М.

« 08 » 04 2025 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ
на здобуття ступеня вищої освіти «бакалавр»

Студенту _____ Бобух Максима Володимировичу _____
 (Прізвище, ім'я, по батькові)

1. Тема роботи: Розробка програмного забезпечення для агентства з підбору кадрів «Новий шанс»

Керівник роботи _____ доцент кафедри ПЗАС, к.т.н., доцент Макарова Л. М

Затверджені наказом ректора №153 від 08.04.2025 р.

2. Термін подання роботи: 02.06.2025 р.

3. Вихідні дані по роботі: _____

4. Перелік питань, що належать до розробки (найменування розділів) _____

Титульний аркуш; Завдання на кваліфікаційну роботу; Анотація (українською та англійською мовами); Зміст; Перелік умовних позначень, символів, одиниць та термінів (при необхідності); Вступ; Аналіз практичної задачі інженерії програмного забезпечення; Проєкт програмного забезпечення; Результати розробки програмного забезпечення; Розділ з охорони праці; Висновки; Список використаних джерел; Додатки (технічне завдання, текст програми, опис програми, інструкція користувача, програма та методика випробувань програмного забезпечення).

5. Перелік презентаційних матеріалів _____

Титульний слайд, Мета та завдання кваліфікаційної роботи, Аналіз вимог до ПЗ, Діаграми діяльності ПЗ, Концептуальна модель даних, Діаграма класів ПЗ, Діаграми взаємодії ПЗ, Логічна модель даних ПЗ, Результати розробки ПЗ, Висновки, Заключний слайд.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
4	Гурець Н.В., ст. викладач		

7. Дата видачі завдання 08.04.2025 р.

КАЛЕНДАРНИЙ ПЛАН

Номер	Назва етапів роботи	Терміни виконання	Примітка
1.	Підготовка розділу ВСТУП	31.03.2025	ПП*
2.	Аналіз практичної задачі інженерії ПЗ	07.04.2025	ПП*
3.	Аналіз вимог до ПЗ	14.04.2025	ПП*
4.	Розробка архітектури ПЗ	21.04.2025	
5.	Розробка проекту ПЗ	05.05.2025	
6.	Реалізація та тестування ПЗ	12.05.2025	
7.	Підготовка розділу Результати розробки ПЗ	16.05.2025	
8.	Підготовка розділу з охорони праці	19.05.2025	ПП*
9.	Підготовка розділу ВИСНОВКИ	23.05.2025	
10.	Оформлення списку використаних джерел та додатків	26.05.2025	
11.	Подання на кафедрі ПЗАС тексту остаточного варіанту роботи, підписаного її керівником, у роздрукованому та електронному форматі разом із заявами щодо самостійності виконання роботи та ідентичності друкованої та електронної версій роботи (Додатки 1 і 2 «Порядку здійснення заходів з перевірки робіт на наявність текстових збігів/ідентичності/схожості із використанням програмно-технічних засобів», який введений в дію наказом ректора НУК за №20 від 20.01.2020 р.)	02.06.2025	не пізніше, ніж за 14 діб до захисту (згідно п.4.1 зазначеного Порядку
12.	Підготовка презентації та доповіді	06.06.2025	
13.	Попередній захист роботи на засіданні кафедри ПЗАС	09.06.2025	
14.	Подання на кафедрі ПЗАС електронних версій наступних документів у форматі pdf: кваліфікаційної роботи; файлу-опису кваліфікаційної роботи (згідно Додатку до наказу ректора НУК за №287-уч від 19.05.2020 р.); презентації доповіді, а також Авторського договору з додатком	16.06.2025	

* - за результатами переддипломної практики (ПП), яка була з 03.02.2025 до 23.03.2025 р.

Студент

(підпис)

Бобух М.В.

(ПІБ)

Керівник роботи

(підпис)

Макарова Л.М.

(ПІБ)

АНОТАЦІЯ

Кваліфікаційна робота присвячена вирішенню практичної задачі інженерії програмного забезпечення, що характеризується комплексністю та невизначеністю умов – розробці програмного забезпечення для агентства з підбору кадрів «Новий шанс», яке забезпечує можливість рекрутингу персоналу та пошук вакансій шукачами.

У кваліфікаційній роботі проведено аналіз практичної задачі інженерії програмного забезпечення з розробки вказаного програмного забезпечення, розглянуто існуючі аналогічні програмні продукти, сформульовано постановку задачі на розробку програмного забезпечення для агентства з підбору кадрів «Новий шанс». Представлено проєкт програмного забезпечення для агентства з підбору кадрів «Новий шанс», результати розробки програмного забезпечення для агентства з підбору кадрів «Новий шанс» та розділ з охорони праці.

Кваліфікаційна робота викладена на 85 сторінках друкованого тексту, містить 20 рисунків, 24 таблиці, список використаних джерел з 19 найменувань та 5 додатків.

ABSTRACT

The qualification work is dedicated to solving the practical task of software engineering, characterized by complexity and uncertainty of conditions – software development for recruitment agency "New Chance", which provides the ability to recruit staff and search for vacancies by applicants.

In the qualification work, the analysis of the practical task of software engineering for the development of the specified software is carried out, the existing similar software products are considered, the statement of the task for the development of software for the recruitment agency "New Chance" were formulated. The software project for the recruitment agency "New Chance," the results of software development for the recruitment agency "New Chance" and the section on labor protection were presented.

The qualification work is presented on 85 pages of typewritten text and contains: 20 figures, 24 tables, list of references from 19 names, 5 appendices.

ЗМІСТ

ВСТУП	7
1 АНАЛІЗ ПРАКТИЧНОЇ ЗАДАЧІ ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ТА ПОСТАНОВКА ЗАДАЧІ НА РОЗРОБКУ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ АГЕНТСТВА З ПІДБОРУ КАДРІВ «НОВИЙ ШАНС»	9
1.1 Аналіз практичної задачі інженерії програмного забезпечення	9
1.2 Аналіз існуючих програмних продуктів	12
1.3 Постановка задачі на розробку програмного забезпечення для агентства з підбору кадрів «Новий шанс»	15
2 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ АГЕНТСТВА З ПІДБОРУ КАДРІВ «НОВИЙ ШАНС»	18
2.1 Аналіз вимог до програмного забезпечення для агентства з підбору кадрів «Новий шанс»	18
2.1.1 Побудова моделі варіантів використання	18
2.1.2 Розробка специфікації варіантів використання	20
2.2 Архітектура програмного забезпечення для агентства з підбору кадрів «Новий шанс»	22
2.3 Проєкт програмного забезпечення для агентства з підбору кадрів «Новий шанс»	24
2.3.1 Ескізний проєкт програмного забезпечення	24
2.3.2 Технічний проєкт програмного забезпечення	28
2.3.3 Робочий проєкт програмного забезпечення	33
3 РЕЗУЛЬТАТИ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ АГЕНТСТВА З ПІДБОРУ КАДРІВ «НОВИЙ ШАНС»	43
4 ОХОРОНА ПРАЦІ	46

4.1 Вступ	46
4.2 Аналіз шкідливих та небезпечних факторів при роботі з персональним комп'ютером	47
4.3 Розробка заходів щодо зменшення впливу шкідливих та небезпечних факторів при роботі з персональним комп'ютером	49
ВИСНОВКИ	52
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	53
Додаток А Технічне завдання на розробку програмного забезпечення для агентства з підбору кадрів «Новий шанс»	55
Додаток Б Код програмного забезпечення для агентства з підбору кадрів «Новий шанс»	61
Додаток В Опис програмного забезпечення для агентства з підбору кадрів «Новий шанс»	73
Додаток Г Інструкція користувача програмного забезпечення для агентства з підбору кадрів «Новий шанс»	76
Додаток Д Програма та методика випробування програмного забезпечення для агентства з підбору кадрів «Новий шанс»	82

ВСТУП

Щоденно у своїй роботі агентства з підбору кадрів оброблює велику кількість документів: заяви шукачів роботи, формує масиви шукачів за ознаками професій, освіти, віку, пільг. В зв'язку з тим, що інформаційні потоки виробничих процесів агентства з підбору кадрів є відносно великими, то їх обробка та отримання даних вимагає значних витрат трудових ресурсів.

З іншого боку, надходить інформація від юридичних осіб, які подають перелік вакансій з вимогами до претендентів. На сьогодні агентство з підбору кадрів має базу даних претендентів та пропозицій. Автоматизація такої системи – доволі складний процес, який охоплює усі аспекти роботи агентства з підбору кадрів, тому вона повинна проводитися обережно та ретельно.

В українському сегменті існує кілька платформ та систем для підбору персоналу, таких, як: Hurma System, SMART HCM, PeopleForce, PersiaHR та деякі інші [1, 2]. Проте, вони мають низку проблем, які вимагають вирішення, наприклад, мають не потрібні агентства з підбору кадрів функції, або ж навпаки не мають функцій яких агентство потребує. Більшість таких систем мають платну підписку, або безкоштовний обмежений набір функцій. З метою здешевлення та оптимізації діяльності агентства з підбору кадрів «Новий шанс» було винесено рішення про розробку програмного забезпечення для підбору персоналу.

Метою кваліфікаційної роботи є розв'язання практичної задачі інженерії програмного забезпечення, що характеризується комплексністю та невизначеністю умов, а саме розробка програмного забезпечення для агентства з підбору кадрів «Новий шанс», для чого потрібно вирішити наступні задачі:

- провести аналіз практичної задачі інженерії програмного забезпечення, яка стосується вказаної мети;

- провести аналіз існуючих програмних продуктів;
- зробити висновок про доцільність розробки програмного забезпечення для агентства з підбору кадрів «Новий шанс»;
- сформулювати вимоги до програмного забезпечення та розробити технічне завдання;
- розробити проєкт програмного забезпечення для агентства з підбору кадрів «Новий шанс», який включає в себе ескізний, технічний та робочий проєкти вказаного програмного забезпечення;
- провести тестування та випробування розробленого програмного забезпечення для агентства з підбору кадрів «Новий шанс»;
- розробити програмну документацію до вказаного програмного забезпечення.

В кваліфікаційній роботі буде розглянуто засоби вирішення задачі з реєстрації шукачів вакансій, пропозицій роботодавців вакансій та виконанням добору претендентів на наявні вакансії. Кваліфікаційна робота містить детальний опис процесу розробки відповідного програмного забезпечення.

Впровадження розробки дозволить автоматизувати низку функцій працівника агентства з підбору кадрів «Новий шанс». Користувачами програмної розробки можуть бути працівники агентства з підбору кадрів «Новий шанс» та керівник агентства. Розробка дозволить отримати необхідну інформацію за запитами користувачів та отримання аналітичних звітів.

Об'єктом кваліфікаційної роботи є процес розробки програмного забезпечення для агентства з підбору кадрів «Новий шанс».

1 АНАЛІЗ ПРАКТИЧНОЇ ЗАДАЧІ ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ТА ПОСТАНОВКА ЗАДАЧІ НА РОЗРОБКУ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ АГЕНТСТВА З ПІДБОРУ КАДРІВ «НОВИЙ ШАНС»

1.1 Аналіз практичної задачі інженерії програмного забезпечення

Основною метою діяльності агентства з підбору кадрів «Новий шанс» є надання послуг населенню та роботодавцям відповідно до вимог чинного законодавства, а також забезпечення ефективної взаємодії між шукачами роботи та роботодавцями на ринку праці.

У своїй роботі агентство керується рядом ключових принципів, серед яких:

- застосування комплексного підходу до надання послуг, спрямованих на активізацію безробітних та підвищення їх конкурентоспроможності шляхом пошуку та підбору роботи;
- уніфікованість та стандартизація процедур, операцій і виробничих процесів;
- надання клієнтам гарантованих інформаційно-консультаційних послуг, а також допомоги у пошуку та підборі вакансій;
- розвиток партнерських відносин з клієнтами для знаходження найбільш ефективних шляхів працевлаштування;
- узгодженість організації праці персоналу агентства зі змістом послуг, що надаються на кожному етапі супроводу клієнта;
- встановлення взаємовигідної співпраці з роботодавцями;
- підвищення рівня відповідальності співробітників агентства за досягнення кінцевих результатів та створення умов для ефективної праці [3, 4].

У процесі співбесіди клієнт спільно з консультантом формує індивідуальні рекомендації, що сприяють його подальшому працевлаштуванню.

У випадку потреби клієнт може бути направлений до фахівця з профорієнтації. Консультант також інформує клієнта про необхідність участі в заходах активної підтримки.

Одним із ключових елементів технології обслуговування є стимулювання самостійної активності шукачів у процесі пошуку роботи. Ефективним інструментом у цьому виступають індивідуальні рекомендації, розроблені за результатами співпраці з консультантом.

З метою підвищення ефективності супроводу клієнта, перші та подальші зустрічі з ним проводить один і той самий спеціаліст, що сприяє формуванню відповідального партнерства між клієнтом і агентством. Однак у випадку відсутності результатів упродовж тривалого часу (наприклад, 3–6 місяців), директор агентства може ухвалити рішення про заміну консультанта на основі аналізу ситуації [5].

Загальні обов'язки консультантів включають:

- оцінку професійних навичок і особистісних якостей клієнта;
- інформування про актуальні вакансії;
- ретельний підбір вакансій із урахуванням відповідності характеристик клієнта вимогам роботодавців;
- складання індивідуального плану дій для прискорення працевлаштування.

Інформація про клієнта фіксується у персональній картці особи, яка є основним документом для обліку осіб, що звернулися до агентства з метою працевлаштування. Ця картка заповнюється фахівцем після реєстрації клієнта, друкується та підписується як співробітником агентства, так і клієнтом.

Зміст персональної картки особи включає:

- загальні відомості, необхідні для працевлаштування;
- інформацію про трудову діяльність;
- дані щодо реєстрації;

- резюме клієнта.

Наповнення окремих блоків персональної картки особи залежить від статусу особи – безробітна чи така, що шукає роботу.

У разі потреби вноситься інформація до закладок:

- відношення до квоти;
- згода на роботу в іншому населеному пункті.

Заповнення персональної картки особи та ведення персональної справи проводиться за допомогою відповідного програмного забезпечення. Заповнення відповідних блоків у процесі формування персональної картки особи здійснюється спеціалістом агентства з підбору кадрів за допомогою інтерфейсних форм.

1.2 Аналіз існуючих програмних продуктів

Як вже було відмічено у вступі, існує багато платформ та систем для рекрутингу персоналу, які мають великий обсяг функцій та можливостей, але не зовсім задовільняють декільком критичним вимогам до розроблюваної системи [6]. Розглянемо деякі платформи більш детально.

Однією з таких платформ є HURMA System [7]. Це комплексна HRM-система, яка охоплює рекрутинг, управління персоналом, адаптацію, OKR та HR-аналитику. Її особливості:

- автоматизація усіх процесів: від розміщення вакансій до адаптації нових співробітників;
- модулі для моніторингу настрою співробітників та управління відсутностями;
- інтеграція з українськими job-порталами та LinkedIn;
- чат-бот AI Partner для Telegram, Slack та Viber.

Платформа підходить для як для малого, так і для середнього бізнесу, але більше вона орієнтована для великих компаній. Зовнішній вигляд платформи наведено на рисунку 1.1.

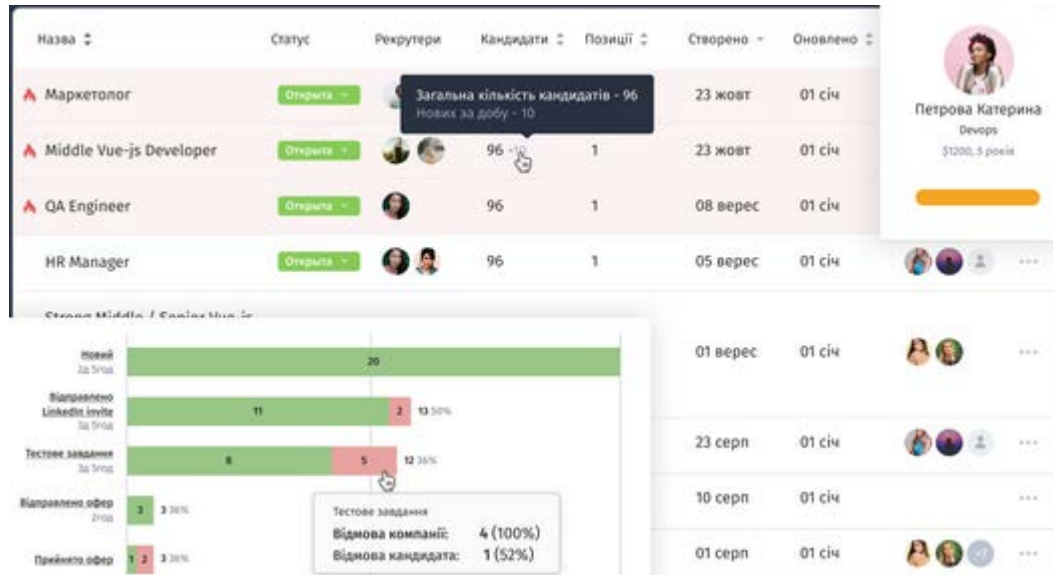


Рисунок 1.1 – Зовнішній вигляд платформи HURMA System

Інший програмний продукт PeopleForce – це модульна HRM-платформа для автоматизації HR-процесів, включаючи рекрутинг, облік робочого часу, оцінку ефективності та управління цілями [8]. Її особливості:

- оцінка персоналу за методом Performance Review;
- кейс-менеджмент та автоматизація онбордингу;
- можливість налаштування політик відпусток та проведення опитувань;
- інтеграція з популярними сервісами: Google, Slack, Zoom, Teams.

Програмний продукт підходить для середніх та великих компаній. Зовнішній вигляд платформи наведено на рисунку 1.2.

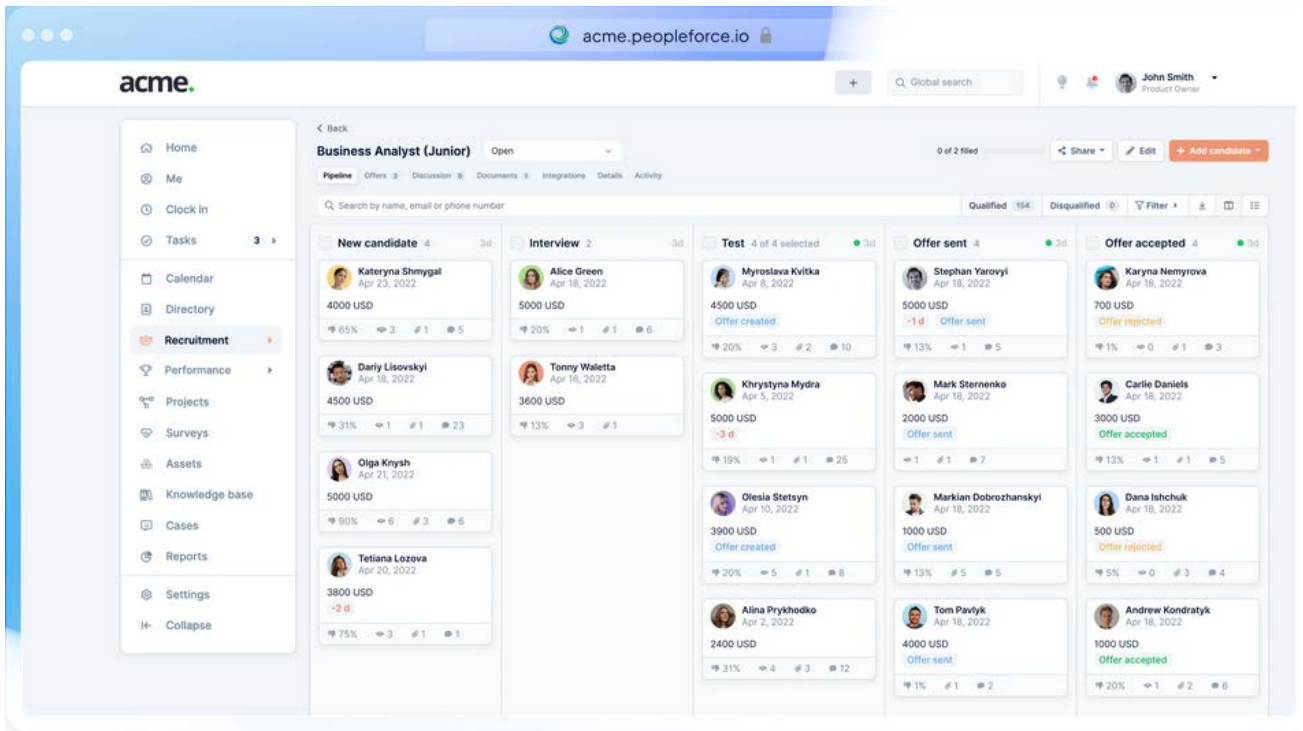


Рисунок 1.2 – Зовнішній вигляд платформи PeopleForce

Ще однією платформою є PersiaHR. Це онлайн-сервіс автоматизації роботи команди рекрутерів в Україні, розроблений для спрощення рутинних процесів підбору персоналу [9]. Її особливості:

- функції для управління кандидатами та вакансіями;
- інтерфейс, орієнтований на рекрутерів.

Платформа підходить для команд, які зосереджені на рекрутингу. Зовнішній вигляд платформи наведено на рисунку 1.3.

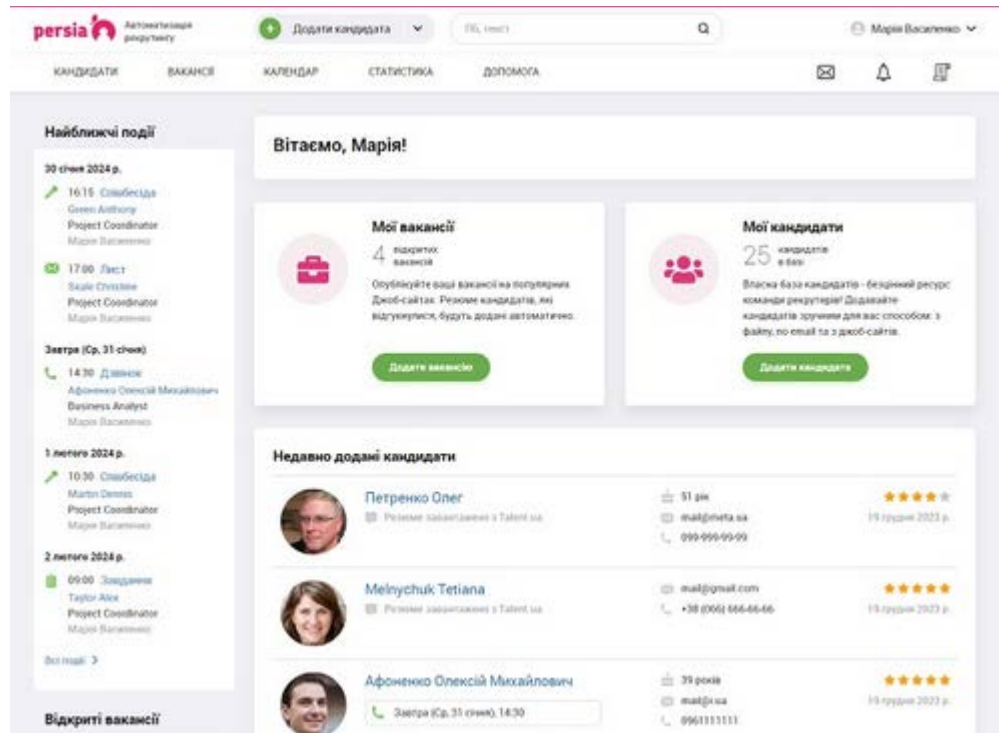


Рисунок 1.3 – Зовнішній вигляд платформи PersiaHR

Підсумовуючи, можна зробити загальний висновок, що програмні продукти, які були розглянуті у цьому підрозділі або орієнтовані більше на великі підприємства та мають надлишкову функціональність для роботи агентства з підбору кадрів «Новий шанс», а відповідно, і завищеної ціни, або є онлайн-сервісом, що також не зовсім підходить для роботи агентства з підбору кадрів «Новий шанс». Тому автоматизація роботи агентства з підбору кадрів «Новий шанс» завдяки розробці відповідного програмного забезпечення є актуальною.

1.3 Постановка задачі на розробку програмного забезпечення для агентства з підбору кадрів «Новий шанс»

Виходячи з аналізу практичної задачі інженерії програмного забезпечення та існуючих програмних продуктів була виконана постановка задачі на розробку програмного забезпечення для агентства з підбору кадрів «Новий шанс». Розробка є доцільною, тому програмне забезпечення повинно забезпечити автоматизацію процесу рекрутингу, що полегшить процес підбору персоналу для роботодавців і процес пошуку вакансій для шукачів.

Дана розробка призначена для зберігання, перегляду, пошуку, редагування та видалення інформації шукачів, вакансій, роботодавців у базі даних агентства з підбору кадрів «Новий шанс» та співставлення наявних вакансій та заяв від шукачів з метою їх подальшого працевлаштування.

Функціональним призначенням програми є автоматизація підбору вакансій для агентства з підбору кадрів «Новий шанс», яка покращить та полегшить обробку потрібної інформації.

Вимоги до функціональних характеристик

Програмне забезпечення повинно виконувати ряд наступних функцій:

- авторизація;
- перегляд, введення та редагування даних;
- виконання адміністративних функцій (підтримка та збереження цілісності бази даних);
- отримання списків роботодавців, пропозицій, шукачів, вакансій;
- формування звіту про наявність вакансій по критерію (наприклад, вакансії для інвалідів);
- формування звіту про працевлаштованих;
- формування звіту про роботодавців.

Вимоги до організації вхідних та вихідних даних

Введення даних здійснюється за допомогою клавіатури. Данні вводяться вручну або обираються з випадаючих списків.

Обмін інформацією між програмним забезпеченням і базою даних відбувається за допомогою файлів з довільним доступом, з розширенням .mdb.

Вхідні дані:

- інформація про шукачів;
- інформація про вакансії;
- інформація про роботодавців.

Вихідні дані:

- список шукачів;
- список пропозицій;
- список роботодавців;
- звіт про наявність вакансій по критерію;
- звіт про працевлаштованих;
- звіт про роботодавців.

Довідник «Пропозиції» повинен характеризуватись такими даними: код роботодавця, код вакансії, заробітна плата, режим роботи, кількість пропозицій, дата пропозиції, потрібна освіта, потрібна спеціальність, потрібний вік, потрібна стать, інвалідність.

Довідник «Працевлаштування» повинен характеризуватись такими даними: код шукача, дата заявки, відмова, код вакансії.

Довідник «Роботодавці» повинен характеризуватись такими даними: код роботодавця, назва підприємства, інформація, керівник.

Довідник «Шукачі» повинен характеризуватись такими даними: код шукача, ПІБ, дата народження, адреса, телефон, освіта, спеціальність, стать, інвалідність.

Довідник «ДКП» повинен характеризуватись такими даними: код спеціальності, назва.

Вимоги до складу та параметрів технічних засобів

Комп'ютер кінцевого користувача повинен задовольняти наступним мініміальним вимогам:

- CPU: Intel(R) i5 2,16 МГц або аналогічний за параметрами AMD;
- RAM: 4 GB;
- HDD: 10 GB вільного місця.

Вимоги до інформаційної та програмної сумісності

Для повноцінного функціонування програми необхідно використовувати операційну систему Windows версії не нижче 7. Для роботи системи необхідно СУБД Access версії не нижче Access 2003.

Детальна постановка задачі на розробку програмного забезпечення для агентства з підбору кадрів «Новий шанс» представлена у технічному завданні, яке наведено у додатку А.

2 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ АГЕНТСТВА З ПІДБОРУ КАДРІВ «НОВИЙ ШАНС»

2.1 Аналіз вимог до програмного забезпечення для агентства з підбору кадрів «Новий шанс»

2.1.1 Побудова моделі варіантів використання

Головною метою розробки будь-якого програмного забезпечення є створення продукту, що полегшує виконання повсякденних завдань користувача. На початковому етапі проектування визначаються вимоги, яким повинно відповідати майбутнє програмне забезпечення.

Для чіткого розуміння очікуваної поведінки програмне забезпечення застосовується опис функціональних можливостей за допомогою варіантів використання - Use Case або прецедентів. Діаграми варіантів використання дозволяють візуалізувати функціональне призначення програмного забезпечення та відобразити його основні дії з точки зору користувача.

Діаграма варіантів використання програмного забезпечення для агентства з підбору кадрів «Новий шанс» наведена на рисунку 2.1.

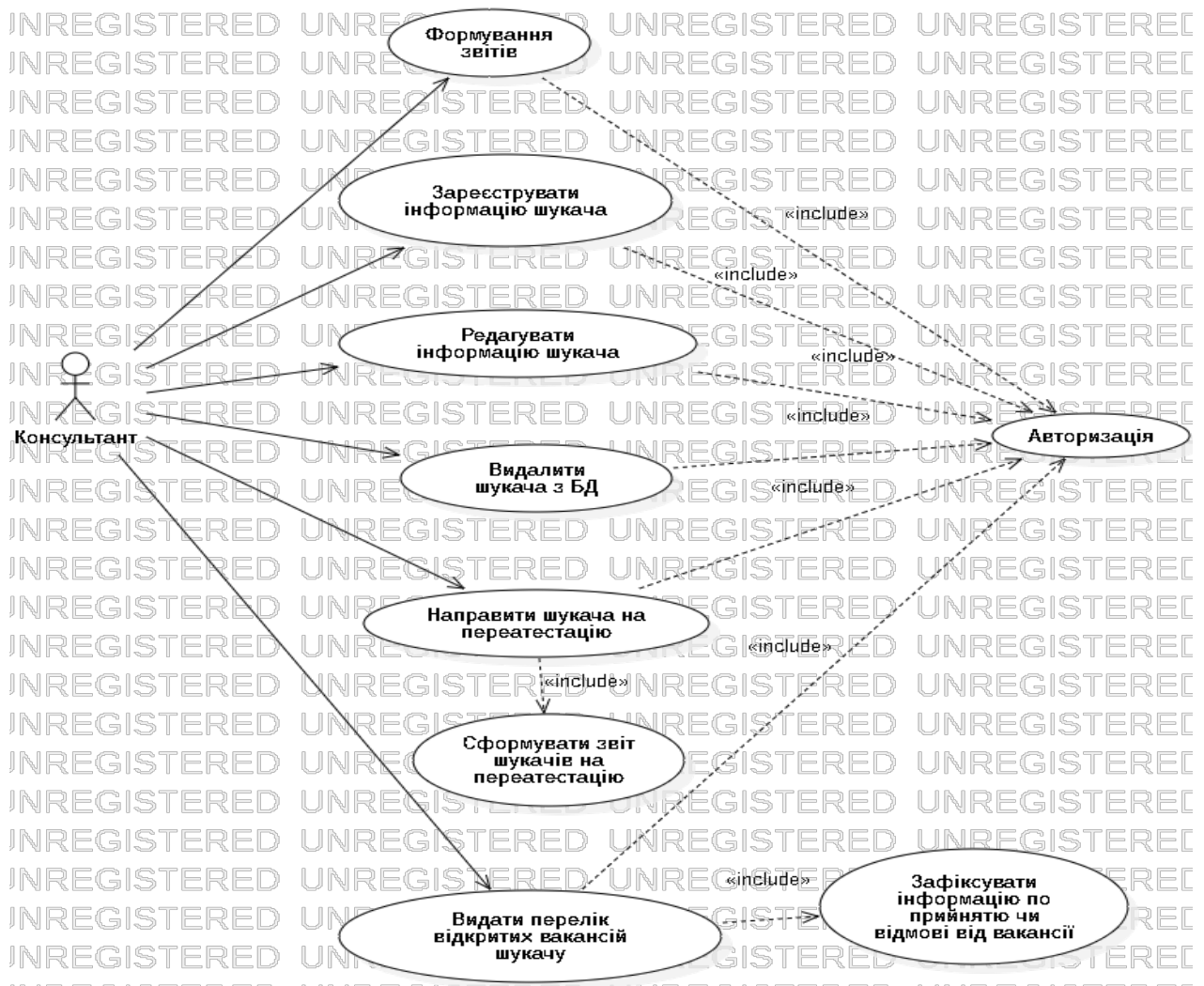


Рисунок 2.1 – Діаграма варіантів використання програмного забезпечення для агентства з підбору кадрів «Новий шанс»

Основним користувачем програмного забезпечення є консультант агентства з підбору кадрів «Новий шанс». Діаграма варіантів використання є основою для визначення архітектури програми та інформаційних потоків.

2.1.2 Розробка специфікації варіантів використання

Специфікації основних варіантів використання програмного забезпечення агентства з підбору кадрів «Новий шанс» представлені нижче у таблицях 2.1 – 2.5.

Таблиця 2.1 – Специфікація варіанту використання «Авторизація»

Складова	Опис
Короткий опис	Даний варіант використання дозволяє дійовій особі пройти перевірку доступу.
Учасники	Консультант
Стартова точка	Запуск програми
Передумови	Користувач повинен мати ідентифікатор (ім'я та пароль) доступу в систему
Основний потік подій	Користувач вводить ім'я та пароль Користувач натискає кнопку ОК
Альтернативний потік подій	-
Постумови	Перехід до головного меню

Таблиця 2.2 – Специфікація варіанту використання «Зареєструвати інформацію шукача»

Складова	Опис
Короткий опис	Даний варіант використання дозволяє занести в базу даних інформацію подану шукачем
Учасники	Консультант
Стартова точка	Обрано довідник «Шукачі»
Передумови	Виконання варіанта використання «Авторизація».
Основний потік подій	1. Система переходить до головного меню; 2. Користувач обирає кнопку «Обробка оперативних даних»; 3. Користувач обирає кнопку «Шукачі»; 4. Ввід даних шукача у відповідні поля на формі; 5. Натиснути кнопку «Додати».
Альтернативний потік подій	-
Постумови	Дані шукача заносяться до бази даних.

Таблиця 2.3 – Специфікація варіанту використання «Формування звітів»

Складова	Опис
Короткий опис	Даний варіант використання дозволяє користувачу працювати зі звітами «Переатестація», «Статус вакансій», «Вакансії для інвалідів», «Вакансії для учасників АТО»
Учасники	Консультант
Стартова точка	Обрано пункт «Звіти»
Передумови	Виконання варіанта використання «Авторизація».
Основний потік подій	1. Система переходить до головного меню; 2. Користувач обирає кнопку «Звіти» 3. Користувач обирає потрібний звіт. 4. Система відкриває форму звіту. 5. Користувач вводить параметри відбору даних. 6. Система формує обраний звіт.
Альтернативний потік подій	
Постумови	Сформовано обраний користувачем звіт

Таблиця 2.4 – Специфікація варіанту використання «Видати перелік відкритих вакансій шукачу»

Складова	Опис
Короткий опис	Даний варіант використання дозволяє консультанту отримати список вакансій підібраних для вказаного шукача системою.
Учасники	Консультант
Стартова точка	Виконання варіанта використання «Авторизація».
Передумови	Виконання варіанта використання «Авторизація».
Основний потік подій	1. Система переходить до головного меню; 2. Користувач обирає пункт меню «Запити» 3. Система відкриває перелік запитів до бази даних. 3. Користувач обирає пункт меню «Пошук збіжностей» 4. Система відкриває форму запиту. 5. Користувач вводить дані потрібного шукачу у відповідні поля на формі. 6. Система виводить список вакансій на які може претендувати шукач.
Альтернативний потік подій	
Постумови	Не визначені.

Таблиця 2.5 – Специфікація варіанту використання системи «Зафіксувати інформацію по прийняттю чи відмові від вакансії»

Складова	Опис
Короткий опис	Даний варіант використання дозволяє підтвердити прийняття вакансії, чи відмову від неї
Учасники	Консультант
Стартова точка	Обрано запит «Пошук збіжностей»
Передумови	Виконання варіанта використання «Видати перелік відкритих вакансій шукачу».
Основний потік подій	1. Відкривається форма з детальною інформацією щодо обраної вакансії. 2. Користувач фіксує згоду з вакансією або відмову від неї через відповідні поля у таблиці.
Альтернативний потік подій	
Постумови	У разі згоди чи відмови від вакансії користувач заносяться до таблиці «Працевлаштування» з відповідним ідентифікатором.

Далі після розробки діаграми варіантів використання та специфікації варіантів використання програмного забезпечення для агентства з підбору кадрів «Новий шанс» можемо переходити до розробки архітектури програмного забезпечення.

2.2 Архітектура програмного забезпечення для агентства з підбору кадрів «Новий шанс»

Архітектура програмного забезпечення охоплює всі ключові проєктні рішення, що стосуються структури програми та взаємодії між її складовими елементами, які разом формують систему. Такі рішення визначають набір необхідних властивостей, яких повинно дотримуватися програмне забезпечення для досягнення успішного функціонування. Вони також формують

концептуальну базу, що слугує основою для розробки, подальшої підтримки та супроводу програмного продукту.

Для розробки програмного забезпечення для агентства з підбору кадрів «Новий шанс» був обраний паттерн проєктування MVC (Model-View-Controller).

Model-View-Controller (MVC, «Модель-Подання-Контролер») – це архітектурний паттерн, що передбачає розділення структури застосунку на три окремі компоненти: модель, представлення та контролер. Такий поділ дозволяє ізолювати бізнес-логіку (Model), інтерфейс користувача (View) та логіку обробки подій (Controller), що забезпечує можливість незалежного оновлення або зміни кожного з компонентів без впливу на інші.

Патерн MVC дозволить спростити процес розробки шляхом поділу частини дизайну і логіки. Тим же способом вирішується проблема масштабування. Поділ бізнес-логіки й обробки даних, в модулях взаємодії з користувачем, в модулі моделі і контролерів, дозволяє зменшити навантаження. Так само даний патерн проєктування, є дуже гнучким і дає змогу вносити зміни в існуючий функціонал розроблюваного програмного забезпечення дуже легко і швидко [10]. Схема патерну MVC наведена на рисунку 2.2.

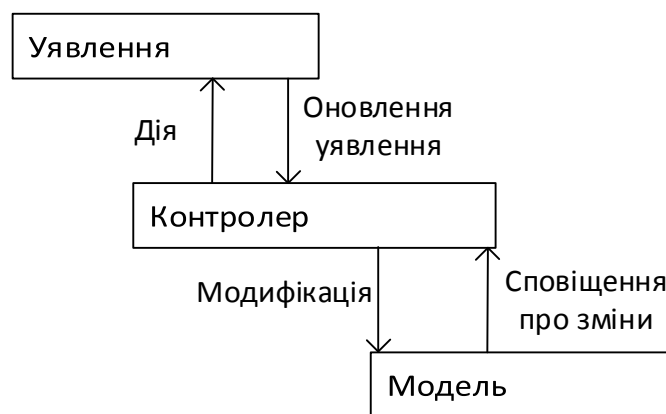


Рис. 2.2 – Схема патерну MVC

Опис компонентів патерну MVC

Компонент View здійснює взаємодію між користувачем і системою. Компонент презентує і відображає дані отримані з бази даних у вигляді інтерфейсу користувача. Через даний компонент команди користувача надсилаються до компонента Controller, отримуються та відображаються дані з компонента Model.

Компонент Controller здійснює взаємодію користувачем і компонентом який надає дані. Компонент отримує командами користувача з компонента View та здійснює дії для зміни даних у компоненті Model, згідно цих команд.

Компонент Model здійснює взаємодію між компонентом, який надає команди на зміну даних та інтерфейсом користувача. Компонент отримує команди на зміну даних з компонента Controller, запитує дані у компонента View і надсилає потрібні дані до компонента View.

2.3 Проект програмного забезпечення для агентства з підбору кадрів «Новий шанс»

2.3.1 Ескізний проект програмного забезпечення

Побудова діаграми діяльності

Діаграма діяльності (діаграма активності) дозволяє моделювати послідовності бізнес-процесів або дій, реалізованих методами класів. Діаграми діяльності є аналогом блок-схеми будь-якого алгоритму. Ці діаграми широко використовуються в описі поведінки, що включає велику кількість паралельних процесів. Кожний стан на діаграмі діяльності відповідає виконанню деякої елементарної операції, а перехід в наступний стан виконується тільки після завершення цієї операції [11].

Діаграма діяльності для основних варіантів використання програмного забезпечення для агентства з підбору кадрів «Новий шанс» наведена на рисунках 2.3–2.4.

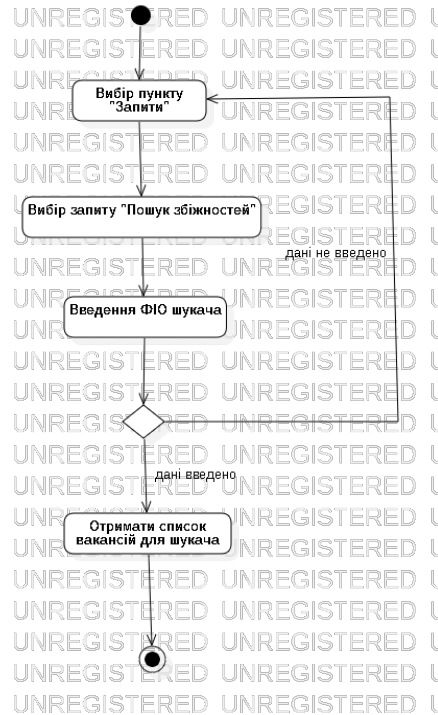


Рисунок 2.3 – Діаграма діяльності для варіанту використання «Видати перелік відкритих вакансій шукачу»

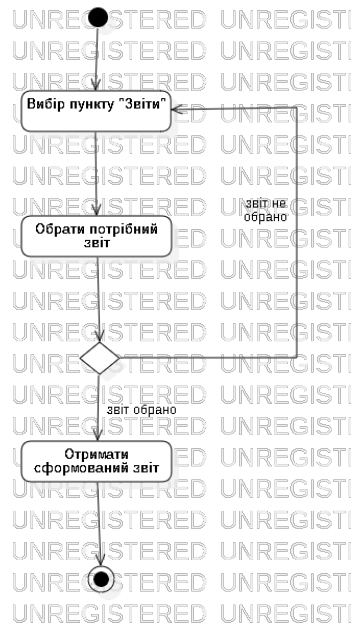


Рисунок 2.4 – Діаграма діяльності для варіанту використання «Формування звітів»

Розробка концептуальної моделі

З метою визначення зв'язків між потоками даних, на основі вимог до функціональних характеристик визначених в технічному завданні, розробляємо концептуальну модель даних, яка представлена у вигляді діаграми на рисунку 2.5. Вона дозволяє визначити сутності, їх атрибути, а також зв'язки між сутностями.

Сутність являє собою безліч екземплярів реальних, чи абстрактних об'єктів, що володіють загальними атрибутами чи характеристиками. Любий об'єкт системи може бути представлений тільки однією сутністю, що унікально ідентифікована. Відношення, в самому загальному вигляді являються зв'язком між двома, або більше сутностями [12].

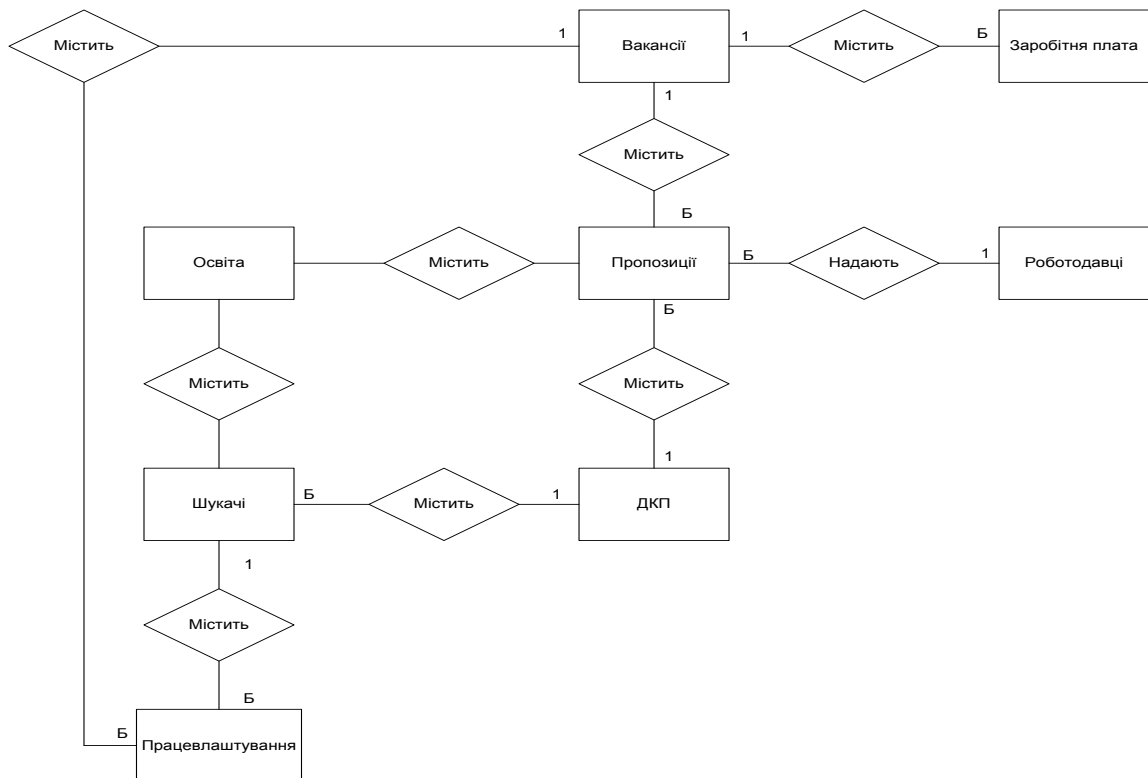


Рисунок 2.5 – Концептуальна модель даних програмного забезпечення для агентства з підбору кадрів «Новий шанс»

Розробка прототипу інтерфейсу користувача

Інтерфейс користувача – сукупність засобів для обробки та відображення інформації, максимально пристосованих для зручності користувача. У графічних системах реалізовується багатовіконним режимом, змінами кольору, розміру, видимості вікон, їх розташуванням, сортуванням елементів вікон, гнучкими налаштуваннями як самих вікон, так і окремих їх елементів, доступністю багатокористувацьких налаштувань [13].

Створення максимально ефективного прототипу інтерфейсу є надзвичайно важливою задачею. Прототип повинен бути максимально повним, дешевим та повинен легко оновлюватися.

На рисунках 2.6 – 2.7 представлено прототип інтерфейсу основних форм програмного забезпечення для агентства з підбору кадрів «Новий шанс».

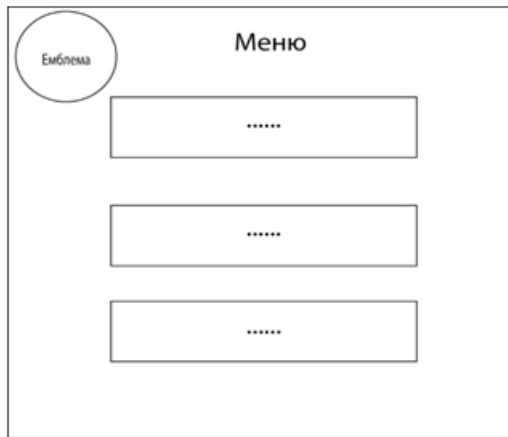


Рисунок 2.6 – Прототип головної форми програмного забезпечення для агентства з підбору кадрів «Новий шанс»

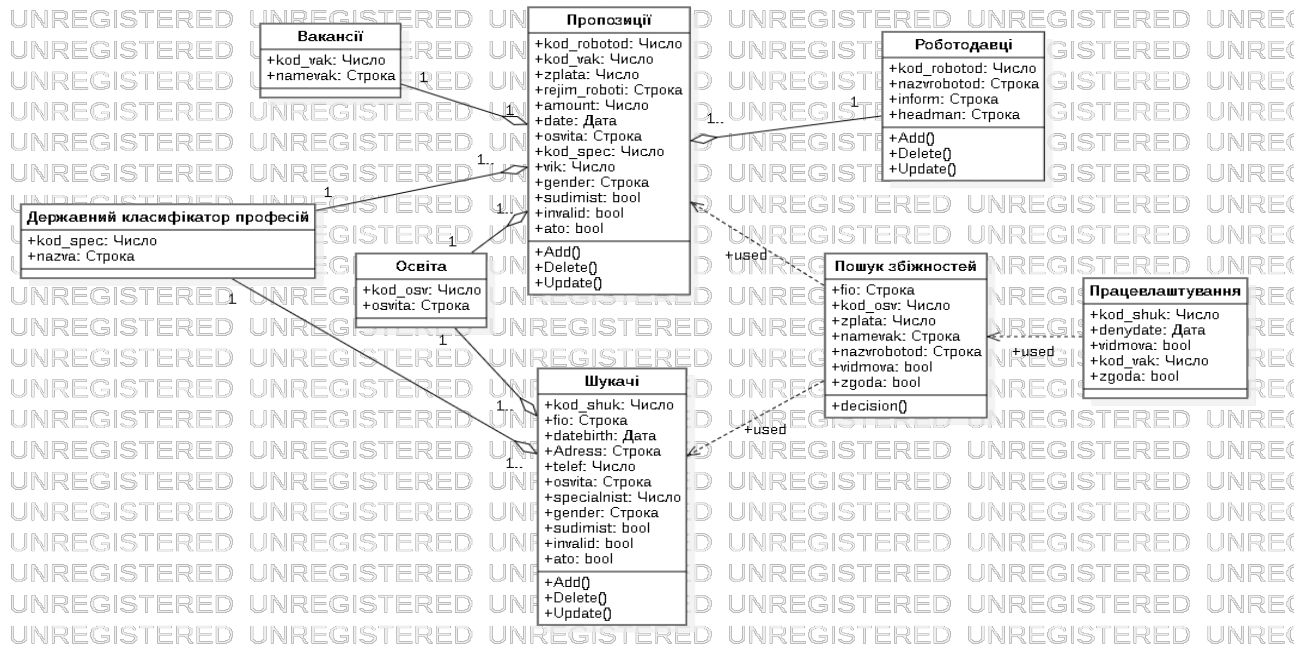


Рис. 2.8 – Діаграма класів програмного забезпечення для агентства з підбору кадрів «Новий шанс»

Побудова діаграми взаємодії

Діаграма взаємодії в UML є засобом моделювання поведінки системи шляхом опису процесу обміну повідомленнями між об'єктами, що беруть участь у певному варіанті використання. На такій діаграмі відображаються конкретні екземпляри об'єктів та послідовність повідомлень між ними [15]. Як правило, кожна діаграма взаємодії моделює поведінку системи в контексті одного окремого сценарію (use case), що дозволяє детально відстежити логіку взаємодії між компонентами.

На рисунках 2.9 – 2.10 зображені діаграми взаємодії для декількох основних варіантів використання програмного забезпечення для агентства з підбору кадрів «Новий шанс».

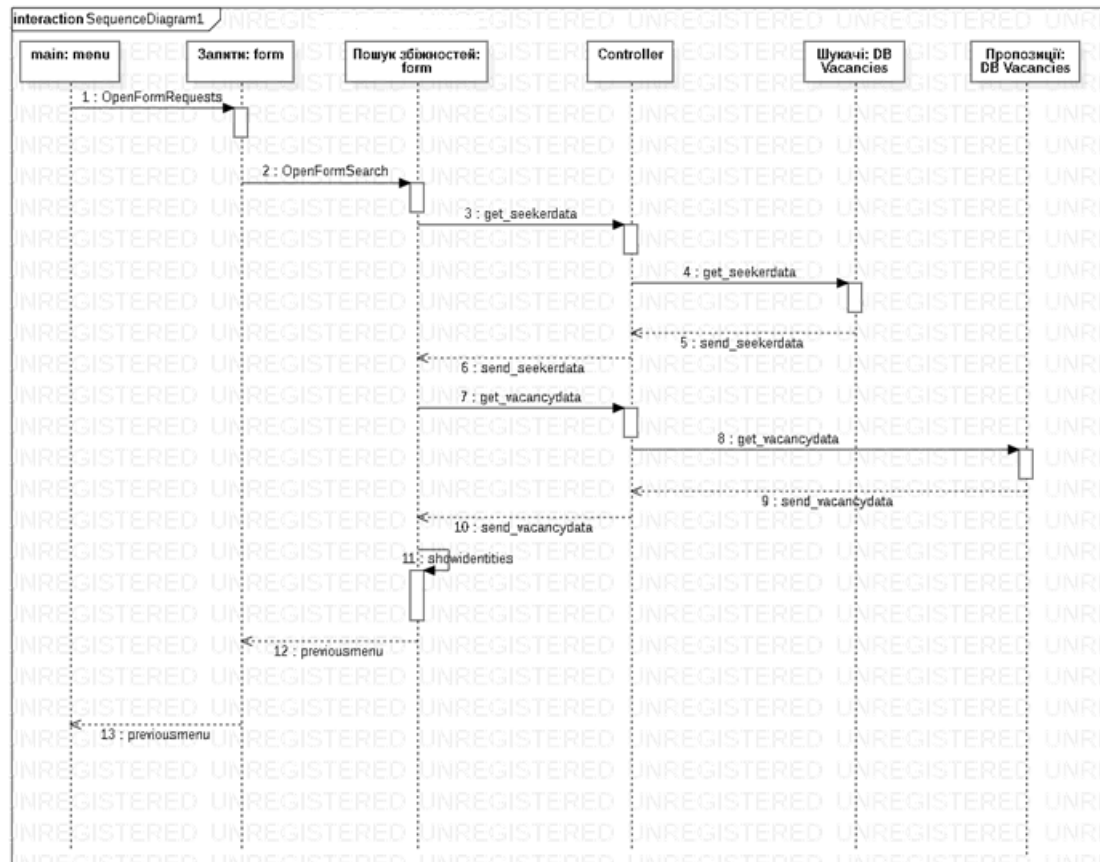


Рис. 2.9 – Діаграма взаємодії для варіанту використання «Видати перелік відкритих вакансій шукачу»

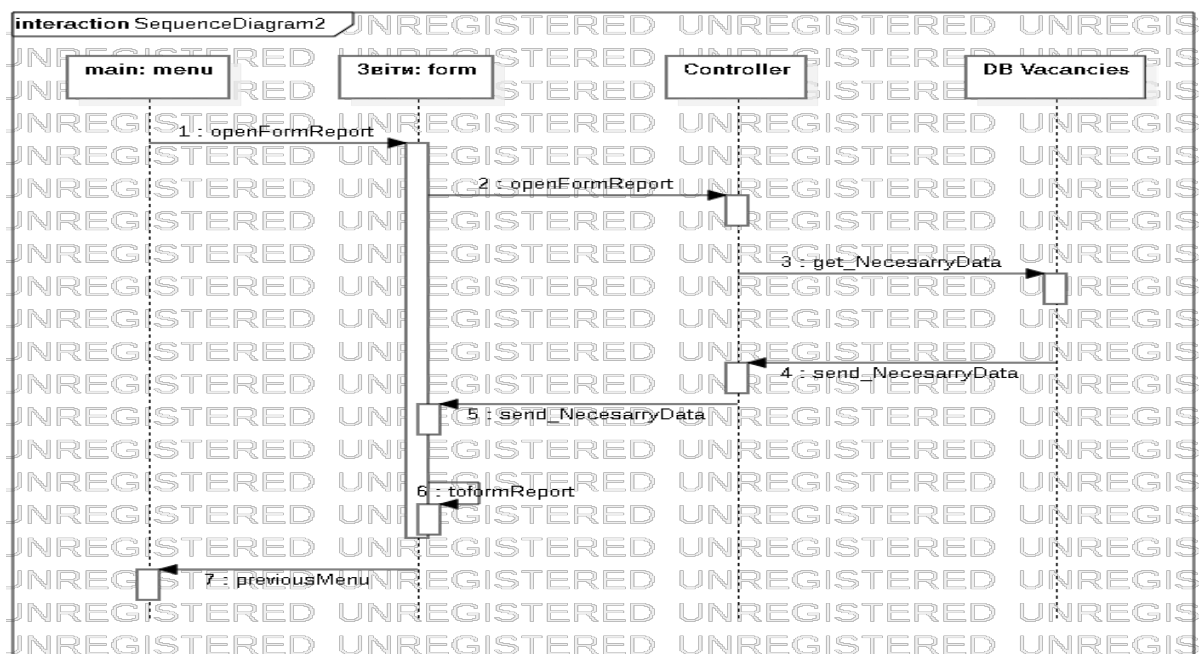


Рис. 2.10 – діаграма взаємодії для варіанту використання «Формування звітів»

Розробка логічної моделі даних

Для того щоб побудувати логічну модель даних, використаємо концептуальну модель даних, яка розроблена раніше. Мета створення логічної моделі даних – більш точне уявлення про склад даних, які будуть використовуватися у програмному забезпеченні.

Логічна модель даних програмного забезпечення для агентства з підбору кадрів «Новий шанс» наведена на рис.2.11.

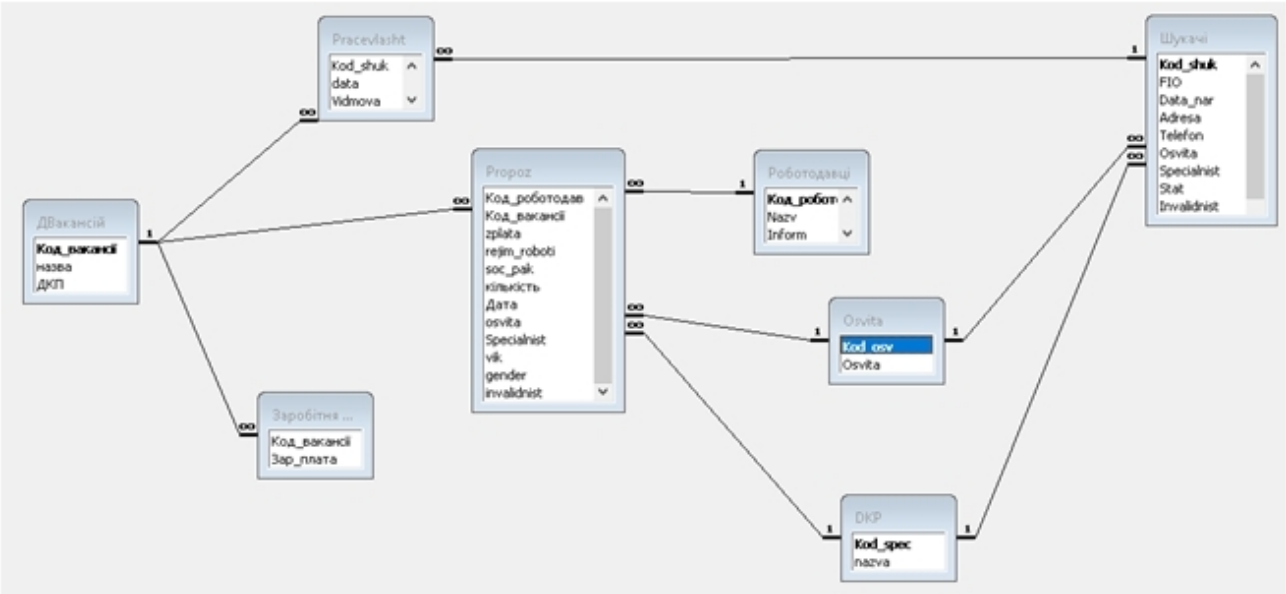


Рисунок 2.11 – Логічна модель даних програмного забезпечення для агентства з підбору кадрів «Новий шанс»

В таблицях 2.6 – 2.13 представлені структури таблиць логічної моделі даних програмного забезпечення для агентства з підбору кадрів «Новий шанс».

Таблиця 2.6 – Логічна модель таблиці „Пропозиції”

Поле	Тип	Первинний ключ
1	2	3
Код пропозиції	Числовий	+
Код роботодавця	Числовий	
Код вакансії	Числовий	
Заробітня плата	Числовий	

Продовження таблиці 2.6

1	2	3
Режим роботи	Текстовий	
Соціальний пакет	Логічний	
Кількість пропозицій	Числовий	
Дата пропозиції	Дата/Час	
Потрібна освіта	Текстовий	
Потрібна спеціальність	Числовий	
Потрібний вік	Числовий	
Потрібна стаття	Текстовий	
Інвалідність	Логічний	

Таблиця 2.7 – Логічна модель таблиці „Працевлаштування”

Поле	Тип	Первинний ключ
Код шукача	Числовий	
Дата заявки	Дата/Час	
Відмова	Числовий	
Код вакансії	Числовий	

Таблиця 2.8 – Логічна модель таблиці „Роботодавці”

Поле	Тип	Первинний ключ
Код роботодавця	Числовий	+
Назва підприємства	Текстовий	
Інформація	Текстовий	
Керівник	Текстовий	

Таблиця 2.9 – Логічна модель таблиці „Шукачі”

Поле	Тип	Первинний ключ
Код шукача	Числовий	+
ПІБ	Текстовий	
Дата народження	Дата	
Адреса	Текстовий	
Телефон	Текстовий	
Освіта	Текстовий	
Спеціальність	Текстовий	
Стать	Текстовий	
Інвалідність	Логічний	

Таблиця 2.10 – Логічна модель таблиці „Освіта”

Поле	Тип	Первинний ключ
Код освіти	Числовий	+
Освіта	Текстовий	

Таблиця 2.11 – Логічна модель таблиці „ДКР”

Поле	Тип	Первинний ключ
Код спеціальності	Числовий	+
Назва	Текстовий	

Таблиця 2.12 – Логічна модель таблиці „ДВакансій”

Поле	Тип	Первинний ключ
Код вакансії	Числовий	+
Назва	Текстовий	

Таблиця 2.13 – Логічна модель таблиці „Заробітня плата”

Поле	Тип	Первинний ключ
Код вакансії	Числовий	
Заробітня плата	Числовий	

Правильно спроектована логічна модель даних дозволить далі виконати розробку фізичної моделі даних, використовуючи конкретну обрану СУБД.

2.3.3 Робочий проєкт програмного забезпечення

Обґрунтування вибору використаних технологій, мов програмування та СКБД

Програмне забезпечення для агентства з підбору кадрів «Новий шанс» повинно реалізовувати функції створення, збереження, пошуку даних в масивах інформації . Інформаційні масиви повинні утримувати в собі довідкові та

оперативні дані. За запитами користувача програмне забезпечення повинно формувати та видавати йому конкретні дані оперативного та довідкового характеру.

З огляду на наведені вимоги, виконується вибір мови програмування для розробки програмного забезпечення для агентства з підбору кадрів «Новий шанс» – C++.

Багато компонентів C++ мають своє візуальне зображення. Перевагою також є те, що розміщення компонентів на екрані, а також завдання початкових значень їхніх властивостей (розміру, кольору, виду бордюру й ін.) C++ дозволяє здійснювати на етапі конструювання форми без написання якої-небудь програми. Для цієї мети передбачене спеціальне вікно, назване Інспектором об'єктів, у якому перераховані всі доступні в режимі проєктування властивості виділеного компонента і їхні поточні значення. Іншими словами, можна до запуску програми бачити, як буде виглядати на екрані проєктована форма.

Такий спосіб роботи з об'єктами, що мають графічне представлення, прийнято називати візуальним програмуванням. До C++ додається ціла бібліотека стандартних візуальних компонентів – Visual Component Library [16]. Дуже важливо, що бібліотеку цю кожен користувач може поповнити, додаючи в неї власні компоненти.

В якості середовища розробки було обрано C++Builder 12 Community Edition. Це повнофункціональне інтегроване середовище розробки призначене для створення застосунків як для Windows, так і для iOS, Android, macOS з використанням єдиної бази коду C++ (обмежена ліцензія на використання в комерційних цілях). C++Builder Community Edition включає в себе редактор коду, потужні інструменти налагодження, вбудовану функцію доступу до популярних локальних баз даних, що містять живі дані, безпосередньо під час розробки, можливості Bluetooth і IoT, а також інструмент для розробки

візуальних інтерфейсів користувача, який підтримує ідеальний дизайн пікселів для конкретної платформи [17].

Для реалізації програмного забезпечення для агентства з підбору кадрів «Новий шанс» також була обрана мова структурованих запитів SQL у СУБД Access для розробки бази даних.

SQL (Structured Query Language) – це декларативна мова програмування, призначена для виконання запитів, модифікації даних та управління реляційними базами даних. Більшість сучасних СУБД підтримують стандарт SQL із власними розширеннями. Основу SQL становить набір команд, що забезпечують виконання базових операцій – таких як вибірка, вставка, оновлення та видалення даних – а також адміністрування структур і доступу до бази даних.

Фізична модель бази даних

Для побудови фізичної моделі бази даних програмного забезпечення для агентства з підбору кадрів «Новий шанс» використовуємо уже розроблену логічну модель даних. В таблицях 2.14 – 2.21 представлені структури таблиць фізичної моделі бази даних програмного забезпечення для агентства з підбору кадрів «Новий шанс».

Таблиця 2.14 – Фізична модель таблиці „Пропозиції”

Логічна назва поля	Фізична назва поля	Тип	Довжина поля
1	2	3	4
Код пропозиції	Kod_propoz	Числовий	Довге ціле
Код роботодавця	Kod_robotodav	Числовий	Довге ціле
Код вакансії	Kod_vacansii	Числовий	Довге ціле
Заробітня плата	Zplata	Числовий	Ціле
Режим роботи	Rejim_roboti	Текстовий	30
Соціальний пакет	Soc_paket	Логічний	

Продовження таблиці 2.14

1	2	3	4
Кількість пропозицій	Number	Числовий	Ціле
Дата пропозиції	Data	Дата/Час	
Потрібна освіта	Osvita	Текстовий	30
Потрібна спеціальність	Specialnist	Текстовий	30
Потрібний вік	Vik	Числовий	Ціле
Потрібна стать	Gender	Текстовий	15
Інвалідність	Invalidnist	Логічний	

Таблиця 2.15– Фізична модель таблиці „Працевлаштування”

Логічна назва поля	Фізична назва поля	Тип	Довжина поля
Код шукача	Kod_shuk	Числовий	Довге ціле
Дата заявки	Data	Дата/Час	
Відмова	Vidmova	Числовий	Ціле
Код вакансії	Kod_vacansii	Числовий	Довге ціле

Таблиця 2.16 – Фізична модель таблиці „Роботодавці”

Логічна назва поля	Фізична назва поля	Тип	Довжина поля
Код роботодавця	Kod_robotodav	Числовий	Довге ціле
Назва підприємства	Nazva	Текстовий	30
Інформація	Inform	Текстовий	60
Керівник	Head	Текстовий	30

Таблиця 2.17 – Фізична модель таблиці „Шукачі”

Логічна назва поля	Фізична назва поля	Тип	Довжина поля
1	2	3	4
Код шукача	Kod_shuk	Числовий	Довге ціле
ПІБ	FIO	Текстовий	30
Дата народження	Data_nar	Дата	
Адреса	Adresa	Текстовий	30
Телефон	Telefon	Текстовий	30

Продовження таблиці 2.17

1	2	3	4
Освіта	Osvita	Текстовий	30
Спеціальність	Specialnist	Текстовий	30
Стать	Gender	Текстовий	15
Інвалідність	Invalidnist	Логічний	

Таблиця 2.18 – Фізична модель таблиці „ Osvita ”

Логічна назва поля	Фізична назва поля	Тип	Довжина поля
Код освіти	Kod_osv	Числовий	Довге ціле
Освіта	Osvita	Текстовий	30

Таблиця 2.19 – Фізична модель таблиці „ DKP ”

Логічна назва поля	Фізична назва поля	Тип	Довжина поля
Код спеціальності	Kod_spec	Числовий	Довге ціле
Назва	Nazva	Текстовий	60

Таблиця 2.20 – Фізична модель таблиці „ ДВакансій ”

Логічна назва поля	Фізична назва поля	Тип	Довжина поля
Код вакансії	Kod_vacansii	Числовий	Довге ціле
Назва	Nazva	Текстовий	30

Таблиця 2.21 – Фізична модель таблиці „ Заробітня плата ”

Логічна назва поля	Фізична назва поля	Тип	Довжина поля
Код вакансії	Kod_vacansii	Числовий	Довге ціле
Заробітня плата	Zplata	Числовий	Ціле

Основна частина коду програмного забезпечення для агентства з підбору кадрів «Новий шанс» представлена в додатку Б.

Реалізація та тестування основних класів еквівалентності

Тестування програмного забезпечення — це процес перевірки відповідності розробленого продукту заданим вимогам. У випадку агентства з підбору кадрів «Новий шанс» тестування проводилося методом «чорного ящика», що передбачає подачу тестових наборів даних і подальший аналіз отриманих результатів без доступу до внутрішньої структури програми.

Для перевірки функції «Зареєструвати інформацію шукача» було застосовано метод розбиття на еквівалентні класи, який дозволяє систематизувати тестові випадки та підвищити ефективність тестування.

Класи еквівалентності функції «Зареєструвати інформацію шукача» представлені в таблиці 2.22.

Почнемо з тестування правильних класів еквівалентності, які наведені в таблиці 2.23. За результатами тестування програмне забезпечення працює правильно. Результат тестування представлено на рисунку 2.13.

Таблиця 2.22 – Класи еквівалентності функції «Зареєструвати інформацію шукача»

Вхідні дані	Правильні класи еквівалентності	Неправильні класи еквівалентності
ПІБ	Рядок (символи)	Цифри, графічні зображення
Дата народження	Дата	Символи, графічні зображення
Адреса	Рядок (символи)	Графічні зображення
Телефон	Цифри	Символи, графічні зображення
Освіта	Рядок (символи)	Графічні зображення
Спеціальність	Рядок (символи)	Графічні зображення

Таблиця 2.23 – Тести з правильних класів еквівалентності функції «Зареєструвати інформацію шукача»

№ тесту	Вхідні дані	Вихідні дані
1	В поле «ПІБ» введено значення Тодоренко А.І.	Результат вірний.
2	В поле «Дата народження» введено значення 6/7/1991	Результат вірний.
3	В поле «Адреса» введено значення вул. Дачна, 12	Результат вірний.
4	В поле «Телефон» введено значення 443332	Результат вірний.
5	В поле «Освіта» введено значення Повна середня	Результат вірний.
6	В поле «Спеціальність» введено значення Бухгалтер	Результат вірний.

Тепер протестуємо неправильні класи еквівалентності, які представлені у таблиці 2.24. За результатами тестування програмне забезпечення працює правильно.

Таблиця 2.24 – Тести з неправильних класів еквівалентності функції «Зареєструвати інформацію шукача»

№ тесту	Вхідні дані	Вихідні дані
1	В поле «ПІБ» введено значення 2356874	Програма виводить повідомлення про помилку. Результат підтверджено
2	В поле «Дата народження» введено будь-який графічний символ ASCII	Програма виводить повідомлення про помилку. Результат підтверджено
3	В поле «Адреса» введено графічний елемент-смайлик	Програма виводить повідомлення про помилку. Результат підтверджено
4	В поле «Телефон» введено значення капримпе	Програма виводить повідомлення про помилку. Результат підтверджено
5	В поле «Освіта» введено будь-який графічний символ ASCII	Програма виводить повідомлення про помилку. Результат підтверджено
6	В поле «Спеціальність» введено графічний елемент-смайлик	Програма виводить повідомлення про помилку. Результат підтверджено

Випробування програмного забезпечення

Випробування програмного забезпечення для агентства з підбору кадрів «Новий шанс» були виконані згідно з програмою та методикою випробувань, яка знаходиться у додатку Г.

1) Перевірка правильності авторизації у програмі

Після введення вірних даних в поля для авторизації, які вказані на рисунку 2.12, відображається форма головного меню програмного забезпечення, яка наведена на рисунку 3.1.

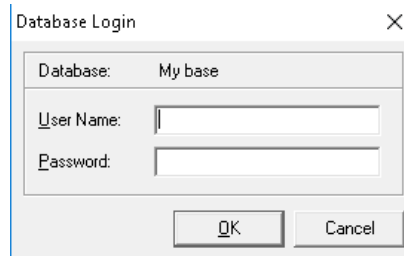


Рисунок 2.12 – Вікно авторизації у програмному забезпеченні для агентства з підбору кадрів «Новий шанс»

2) Перевірка правильності додавання запису шукача

Додамо новий запис шукача. Для цього, після перевірки прав доступу обираємо пункт головного меню «Обробка оперативних даних». Відкривається форма для вибору наборів даних. На ній обираємо пункт «Шукачі». Відкривається форма з інформацією про шукачів. Для додання нового запису шукача потрібно заповнити усі необхідні дані (див. рис. 2.13) і натиснути кнопку «Додати». Як бачимо новий запис додався, тому перевірка пройдена.

Код шукача	ПІБ	Дата народження	Адреса	Телефон	Освіта	Спеці
1	Ольжич В.О.	12/2/1986	вул.Перемиська 18	587126	Повна середня	
2	Вербицкий Р.О.	3/15/1991	вул.Олейника 20а	513220	Базова середня	
3	Молочко Г.В.	6/30/1993	вул.Мирзана 198	321345	Повна середня	
4	Житник Г.Р.	2/23/1988	вул.Прежана 78	845264	Вища 3.4 катего	
5	Млещанский В.С.	5/11/1979	вул.Перев'язи 65	437345	Вища освіта 1.2	
6	Кирило Л.М.	9/8/1995	вул.Мирони 91	668952	Вища 3.4 катего	
7	Теренко Л.О.	2/15/1993	вул. Снярнова 22	232423	Базова середня	
8	Тараненко Г.Д.	6/24/1990	вул. Переяславська 95	564322	Повна середня	
9	Назаренко В.І.	12/6/1994	вул. Дача 12	443332	Вища освіта 1.2 категорія	
10	Тодоренко А.І.	6/7/1991	вул. Заріччя	321122	Вища освіта 1.2 категорія	

10

Тодоренко А.І.

6/7/1991

вул. Заріччя

321122

Вища освіта 1.2 кате

4

Код шукача

ПІБ

Дата народження

Адреса

Телефон

Освіта

Спеціальність

Ж

False

Стать

Невалідність

Додати

Видалити

Редагувати

Очистити поля

Close

Рисунок 2.13 – Додано новий запис шукача

3) Перевірка правильності видалення даних шукача

Видаємо дані про шукача. Для цього, після перевірки прав доступу обираємо пункт головного меню «Обробка оперативних даних». Відкривається форма для вибору наборів даних. На ній обираємо пункт «Шукачі». Відкривається форма з інформацією про шукачів. Для видалення запису шукача потрібно обрати запис (див. рис. 2.14) і натиснути кнопку «Видалити». Як бачимо новий запис видалився, тому перевірка пройдена.

Код шукача	ПІБ	Дата народження	Адреса	Телефон	Освіта	Спеці
1	Ольжич В.О.	12/2/1986	вул.Перемиська 18	587126	Повна середня	
2	Вербицкий Р.О.	3/15/1991	вул.Олейника 20а	513220	Базова середня	
3	Молочко Г.В.	6/30/1993	вул.Мирзана 198	321345	Повна середня	
4	Житник Г.Р.	2/23/1988	вул.Прежана 78	845264	Вища 3.4 катего	
5	Млещанский В.С.	5/11/1979	вул.Перев'язи 65	437345	Вища освіта 1.2	
6	Кирило Л.М.	9/8/1995	вул.Мирони 91	668952	Вища 3.4 катего	
7	Теренко Л.О.	2/15/1993	вул. Снярнова 22	232423	Базова середня	
8	Тараненко Г.Д.	6/24/1990	вул. Переяславська 95	564322	Повна середня	
9	Назаренко В.І.	12/6/1994	вул. Дача 12	443332	Вища освіта 1.2 категорія	

9

Назаренко В.І.

12/6/1994

вул. Дача 12

443332

Вища освіта 1.2 кате

3

Код шукача

ПІБ

Дата народження

Адреса

Телефон

Освіта

Спеціальність

Ч

False

Стать

Невалідність

Додати

Видалити

Редагувати

Очистити поля

Close

Рисунок 2.14 – Запис шукача видалено

4) Перевірка правильності формування звітів

Підприємству для ведення звітності необхідна функція формування звітів. Давайте сформуємо такий звіт. Для цього, після перевірки прав доступу обираємо пункт головного меню «Звіти». Відкривається форма зі звітами. На ній обираємо один з звітів які може сформувати програма. Наприклад звіт «Вакансії для інвалідів». Відкривається форма звіту (див. рис. 2.15). Як бачимо, звіт сформовано, тобто перевірка пройдена успішно.

Вакансії для інвалідів		
Назва	Інвалідність	Кількість
ТОВ "Інди"	True	2
Назва вакансії	Системний адміністратор	
ТОВ "Ікар"	True	4
Назва вакансії	Системний адміністратор	
ТОВ "Транзит"	True	3
Назва вакансії	Прибиральник	
ТОВ "Перетин"	True	6
Назва вакансії	Прибиральник	

Рисунок 2.15 – Звіт з вакансіями для інвалідів

В результаті проведеного тестування та випробування можна зробити висновок, що розроблене програмне забезпечення для агентства з підбору кадрів «Новий шанс» працює правильно.

3 РЕЗУЛЬТАТИ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ АГЕНТСТВА З ПІДБОРУ КАДРІВ «НОВИЙ ШАНС»

У межах даної кваліфікаційної роботи була вирішена практична задача інженерії програмного забезпечення, що характеризується комплексністю та невизначеністю умов, а саме виконана розробка програмного забезпечення для агентства з підбору кадрів «Новий шанс». Програмне забезпечення дозволяє зберігання, перегляд, пошук, редагування та видалення інформації шукачів, вакансій, роботодавців у базі даних агентства з підбору кадрів «Новий шанс» та співставлення наявних вакансій та заяв від шукачів з метою їх подальшого працевлаштування. Для збирання та накопичення відомостей про фізичну особу чи групу фізичних осіб кожна фізична особа особисто дає згоду на обробку даних згідно Закону України «Про захист персональних даних».

Проведено огляд літературних джерел, нормативних документів, Internet-джерел для аналізу роботи та методів можливого автоматизування підбору вакансій для агентства з підбору кадрів «Новий шанс».

Проведено пошук та аналіз вже існуючих аналогічних програмних продуктів для вирішення практичної задачі інженерії програмного забезпечення з розробки програмного забезпечення для агентства з підбору кадрів «Новий шанс». Здійснено постановку задачі для розробки відповідного програмного забезпечення.

Проведено аналіз вимог до програмного забезпечення для агентства з підбору кадрів «Новий шанс», побудована модель та розроблені специфікації варіантів використання. Виконано ескізний, технічний та робочий проекти. В рамках ескізного проєкту побудовано діаграми діяльності для основних варіантів використання засобами мови моделювання UML, розроблено концептуальну модель даних та прототип інтерфейсу користувача.

В рамках технічного проєкту розроблено діаграму класів програмного забезпечення та діаграми взаємодії, побудовано логічну модель бази даних.

Програмне забезпечення для агентства з підбору кадрів «Новий шанс» було розроблено з використанням мови програмування C++ в середовищі розробки C++Builder 12 Community Edition. Для створення бази даних було обрано СУБД Microsoft Access. Зовнішній вигляд головного меню програмного забезпечення для агентства з підбору кадрів «Новий шанс» наведено на рисунку 3.1, а форма підбору вакансій для шукачів – на рисунку 3.2.

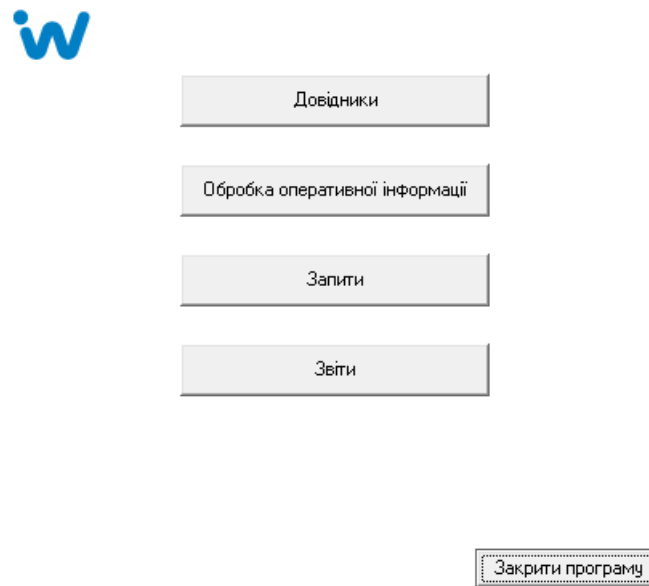


Рисунок 3.1 – Вікно головного меню програмного забезпечення для агентства з підбору кадрів «Новий шанс»

Розроблене програмне забезпечення використовує незначну кількість ресурсів на комп'ютері користувача.

Було виконано тестування та випробування програмного забезпечення, яке показало повну працездатність та відповідність вимогам, які представлені у Додатку А – Технічне завдання на розробку програмного забезпечення для агентства з підбору кадрів «Новий шанс».

ФІО Шукача	Назва вакансії	Рівень освіти	Заробітна плата	Назва підприємства
▶ Назаренко В.І.	Прибиральник	Вища освіта 1,2 категорія	6500	ТОВ "Перетин"
Назаренко В.І.	Агент страховий	Вища освіта 1,2 категорія	15500	ТОВ "Лерика"
Назаренко В.І.	Акушерка	Вища освіта 1,2 категорія	16000	ТОВ "Позенкарц"
Назаренко В.І.	Секретар	Вища освіта 1,2 категорія	14000	ТОВ "Лерика"
Назаренко В.І.	Акушерка	Вища освіта 1,2 категорія	16000	ТОВ "Транзит"
Назаренко В.І.	Секретар	Вища освіта 1,2 категорія	14000	Управління ЖКГ Миколаївської облдержадміністрації

Введіть прізвище та ініціали людини
Наприклад: Петров І.Г.

Рисунок 3.2 – Форма підбору вакансій для шукачів програмного забезпечення для агентства з підбору кадрів «Новий шанс»

Код програми наведено у додатку Б – Код програмного забезпечення для агентства з підбору кадрів «Новий шанс». Опис програми наведено у додатку В – Опис програмного забезпечення для агентства з підбору кадрів «Новий шанс». Для експлуатації програмного забезпечення також було розроблено інструкцію користувача, яку наведено у додатку Г – Інструкція користувача програмного забезпечення для агентства з підбору кадрів «Новий шанс». Програму та методику випробувань програмного забезпечення наведено у додатку Д – Програма та методика випробування програмного забезпечення для агентства з підбору кадрів «Новий шанс».

4 ОХОРОНА ПРАЦІ

4.1 Вступ

Охорона праці є комплексною системою правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних та лікувально-профілактичних заходів, спрямованих на забезпечення безпеки, збереження життя, здоров'я і працездатності працівників під час виконання трудових функцій. Основною метою охорони праці є створення безпечних і комфортних умов праці для забезпечення ефективної і стабільної трудової діяльності [18].

Як міждисциплінарна галузь, охорона праці вивчає питання запобігання виробничому травматизму, професійним захворюванням, аваріям, пожежам та вибухам на підприємствах. Вона формує у спеціалістів знання з правових, організаційних, санітарних та технічних аспектів забезпечення безпеки праці, а також підкреслює пріоритет захисту життя і здоров'я працівників, що закріплено в Конституції України.

Об'єктом дослідження охорони праці є виробнича система, що включає людину, обладнання і виробниче середовище. Методологічно охорона праці базується на системному підході, який охоплює аналіз взаємодії «людина – машина – середовище» з урахуванням фізичних, хімічних, біологічних, психологічних і соціальних факторів, а також нормативно-правового забезпечення безпеки виробництва.

Для ефективної роботи застосовуються методи статистики (для аналізу травматизму і захворювань), економіки (обґрунтування витрат на заходи безпеки), фізики, хімії і біології (оцінка параметрів виробничого середовища і допустимих норм шкідливих факторів). Охорона праці не лише використовує

законодавчу базу, а й адаптує її до сучасних технологічних та економічних умов [19].

4.2 Аналіз шкідливих та небезпечних факторів при роботі з персональним комп'ютером

Працівники, які виконують свою діяльність за персональними комп'ютерами, зазнають впливу комплексу шкідливих і небезпечних виробничих факторів, які можуть негативно позначатися на їхньому здоров'ї і працездатності. Розглянемо детальніше основні категорії таких факторів.

Фізичні фактори

Електрична напруга в мережах робочих місць може призводити до ураження електрострумом у разі несправностей, коротких замикань чи порушення заземлення.

Випромінювання комп'ютерної техніки, зокрема рентгенівське, ультрафіолетове та інфрачервоне, хоч і знаходиться на низькому рівні, але тривале і безперервне впливання може викликати стомлення та подразнення органів зору і шкіри.

Статична електрика, що накопичується на поверхнях обладнання, може викликати дискомфорт та мікроудари, а також сприяти притягненню пилу і забруднень.

Якість повітря робочого приміщення часто порушується через запиленість, наявність аероіонів, які впливають на загальне самопочуття та концентрацію.

Освітлення – нерівномірний розподіл яскравості, пульсації світлового потоку, відблиски на екрані призводять до зорової втоми і головного болю.

Хімічні фактори

Підвищений рівень таких речовин у повітрі, як вуглекислий газ, озон, аміак, формальдегід, фенол та інші, виникає через матеріали офісного устаткування, чистячі засоби або недостатню вентиляцію. Вони здатні викликати подразнення дихальних шляхів, алергічні реакції, зниження працездатності.

Психофізіологічні фактори

Робота з персональними комп'ютерами пов'язана з високою концентрацією уваги, інтенсивною обробкою інформації, що викликає психоемоційне напруження. Серед головних проблем – напруга зору, втома, стрес, монотонність роботи, що може призвести до зниження мотивації, емоційного вигорання і навіть розвитку неврозів.

Основні шкідливі фактори, пов'язані з ергономікою

Тривале сидіння в одній позі викликає застій крові, перевантаження м'язів шиї, спини, рук, що сприяє розвитку остеохондрозу, сколіозу у дітей, а також проблем із судинами малого тазу (простатит, геморої).

Електромагнітне випромінювання комп'ютерної техніки, хоч і незначне, в довготривалій перспективі може мати негативний вплив на центральну нервову систему.

Перевантаження зорового аналізатора спричиняє синдром «сухого ока», зниження гостроти зору, головні болі, що посилюється неправильним освітленням.

Навантаження на кистьові суглоби веде до розвитку тунельного синдрому, артритів, що знижує працездатність і викликає хронічний біль.

Вплив виробничого середовища

Мікроклімат робочих приміщень, в яких знаходяться персональні комп'ютери, визначається температурою, вологістю, рухом повітря і рівнем теплового випромінювання. Нормальні параметри мікроклімату забезпечують

комфортний тепловий баланс організму і запобігають перевантаженню систем терморегуляції.

Значна увага приділяється також складу повітря: у процесі роботи за персональними комп'ютерами відбувається різка зміна концентрації іонів повітря – зниження негативно заряджених іонів та зростання позитивних, що сприяє виникненню втоми, дратівливості, зниженню імунітету [19].

Для збереження здоров'я працівників необхідне регулярне провітрювання, застосування іонізаторів повітря, систем кондиціонування та зволоження. Усі ці заходи сприяють підтримці оптимального мікроклімату і зменшують негативний вплив шкідливих факторів на організм.

4.3 Розробка заходів щодо зменшення впливу шкідливих та небезпечних факторів при роботі з персональним комп'ютером

Для ефективного захисту працівників від шкідливого впливу при роботі з персональними комп'ютерами, необхідно впроваджувати комплекс організаційних, медичних, технічних та гігієнічних заходів.

Організаційні заходи

Всі працівники перед початком роботи проходять медичний огляд із обов'язковою консультацією терапевта, невропатолога, офтальмолога для виявлення протипоказань і профілактики захворювань. Регулярний медичний контроль повинен здійснюватися не рідше одного разу на 2 роки.

Встановлення регламентованих перерв протягом робочого дня: рекомендовано 10–15 хвилин відпочинку кожную годину або дві, в залежності від складності роботи. Час безперервної роботи за комп'ютером не має перевищувати 4 годин.

Організація спеціального приміщення для відпочинку працівників, де вони можуть відновити сили, зняти нервові напруження, що сприяє збереженню працездатності.

Ергономічні заходи

Робоче місце повинно відповідати стандартам ДСТУ 12.2.032-78: стіл із регульованою висотою (725 мм), шириною від 600 до 1400 мм, глибиною 800–1000 мм; підставка для ніг з регульованою висотою до 150 мм і кутом нахилу до 20°.

Забезпечення правильної пози користувача персонального комп'ютера: ноги повинні стояти на підлозі або підставці, стегна горизонтальні, передпліччя вертикальні, лікті зігнуті під кутом 70–90°, зап'ястя — не більше 20° відносно горизонталі, голова нахилена під кутом 15–20°.

Відстань від екрану до очей — не менше 600 мм, клавіатура повинна розташовуватися на опорі з антиковзним покриттям із регульованим кутом нахилу 5–15° і відстанню 100–300 мм від краю столу.

Технічні заходи

Використання захисних екранів, світлофільтрів для зменшення впливу електромагнітного випромінювання і шкідливого світлового навантаження.

Забезпечення належної вентиляції та кондиціонування приміщень, застосування генераторів негативних іонів і зволожувачів повітря для підтримки оптимального мікроклімату.

Регулярне технічне обслуговування обладнання для запобігання перегріву і зниження випромінювання.

Гігієнічні рекомендації

Навчання працівників правильній організації робочого процесу, використанню ергономічного обладнання, дотриманню режиму праці і відпочинку.

Рекомендації щодо виконання спеціальних вправ для зняття м'язового напруження, гімнастики для очей, що зменшує зорове навантаження.

Контроль за дотриманням чистоти робочого місця, регулярне провітрювання приміщень.

Впровадження цих заходів значно підвищує рівень безпеки і комфорту праці, знижує ризики виникнення професійних захворювань і підвищує загальну ефективність роботи співробітників.

ВИСНОВКИ

В ході виконання кваліфікаційної роботи було вирішено практичну задачу інженерії програмного забезпечення, що характеризується комплексністю та невизначеністю умов, а саме розроблено програмне забезпечення для агентства з підбору кадрів «Новий шанс», яке дозволяє автоматизувати виконання таких функцій: реєстрація шукачів вакансій, пропозицій роботодавців вакансій та виконання добору претендентів на наявні вакансії. З цією метою розроблено відповідне програмне забезпечення, яке включає базу даних та додаток до неї для розв'язання наведених питань.

В кваліфікаційної роботи наведено опис процесу розробки програмного забезпечення для агентства з підбору кадрів «Новий шанс», включаючи аналіз практичної задачі інженерії програмного забезпечення, аналіз вимог до програмного забезпечення, проєкт програмного забезпечення та результати розробки програмного забезпечення. Також виконано розділ з охорони праці.

Впровадження розробки дозволило автоматизувати низку функцій працівників агентства з підбору кадрів «Новий шанс». Розробка дозволяє отримати необхідну інформацію за запитами користувачів та отримання аналітичних звітів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Топ 10: HRM системи для України. URL: <https://www.livebusiness.com.ua/ua/tools/hrm/> (дата звернення: 17.03.2025).
2. Вибір правильної системи HRM: оцінка кращих гравців на ринку. URL: <https://vagomo.com/vybir-pravylnoyi-systemy-hrm-oczinka-krashhyh-gravcziv-na-rynku.html> (дата звернення: 17.03.2025).
3. Сочинська-Сибірцева І.М., Доренська А.О., Тушевська Т.В. HR-менеджмент: навч. посіб. Кропивницький : ЦНТУ, 2022. 278 с.
4. Кушнерик О. HR-менеджмент: інноваційний підхід до управління персоналом. *Підприємництво та інновації*, 2020. 12. С. 125-129. <https://doi.org/10.37320/2415-3583/12.21>
5. Малиновська Л. Енциклопедія рекрутингу або як знайти чорну кішку у темній кімнаті. К.: Мультимедійне видавництво Стреблицького, 2013. 210 с.
6. Автоматизація рекрутингу: що враховувати при виборі софту. URL: <https://cleverstaff.net/blog/uk/avtomatyzatsiya-rekrutynhu/> (дата звернення: 17.03.2025).
7. HURMA. ATS система та HRM платформа в одному рішенні. URL: <https://hurma.work/> (дата звернення: 17.03.2025).
8. PeopleForce. Автоматизація HR процесів з CoreHR. URL: <https://peopleforce.io/uk/products/peoplehr> (дата звернення: 17.03.2025).
9. PersiaHR. Онлайн-рішення для автоматизації роботи команди рекрутерів. URL: <https://persiahr.com/uk> (дата звернення: 17.03.2025).
10. MVC: шаблон проєктування архітектури додатку. URL: <https://brander.ua/technologies/mvc> (дата звернення: 29.03.2025).

11. Діаграма діяльності в UML: символ, компоненти та приклад. URL: <https://www.guru99.com/uk/uml-activity-diagram.html> (дата звернення: 29.03.2025).

12. Проєктування та розробка корпоративних інформаційних систем. Концептуальні моделі та схеми даних. URL: http://citforum.ck.ua/cfin/prcorpsys/infersistpr_09.shtml (дата звернення: 29.03.2025).

13. Інтерфейс користувача. URL: <http://sven.ua/ua/service/glossary/ukr/11/> (дата звернення: 29.03.2025).

14. Діаграма класів. URL: <http://moodle.ipo.kpi.ua/moodle/mod/resource/view.php?id=28088> (дата звернення: 16.04.2025).

15. Діаграма взаємодії. URL: <http://moodle.ipo.kpi.ua/moodle/mod/resource/view.php?r=16538> (дата звернення: 16.04.2025).

16. Introducing the Visual Component Library (VCL). URL: [https://docwiki.embarcadero.com/RADStudio/Athens/en/Introducing_the_Visual_Component_Library_\(VCL\)](https://docwiki.embarcadero.com/RADStudio/Athens/en/Introducing_the_Visual_Component_Library_(VCL)) (дата звернення: 16.04.2025).

17. C++Builder 12 Community Edition. URL: <https://www.embarcadero.com/products/cbuilder/starter> (дата звернення: 16.04.2025).

18. Про охорону праці: Закон України від 21.11.2002 р. № 229-IV. Дата оновлення: 04.02.2021. URL: <https://zakon.rada.gov.ua/laws/main/2694-12#Text>. (дата звернення: 17.03.2025).

19. Гандзюк М.П. Основи охорони праці. Підручник для вищих навчальних закладів. К.: Каравела, 2004. 408 с.

Додаток А Технічне завдання на розробку програмного забезпечення для
агентства з підбору кадрів «Новий шанс»

Вступ

Повна назва продукту: програмного забезпечення для агентства з підбору кадрів «Новий шанс».

Область застосування: агентства з підбору кадрів «Новий шанс».

1 Підстава для розробки

Підставою для розробки програмного забезпечення є наказ на затвердження тем кваліфікаційних робіт бакалаврів зі спеціальності 121 Інженерія програмного забезпечення у Національному університеті кораблебудування імені адмірала Макарова.

2 Призначення розробки

Експлуатаційне призначення

Експлуатаційним призначенням програмного забезпечення для агентства з підбору кадрів «Новий шанс» є автоматизація зберігання, перегляду, пошуку, редагування та видалення інформації шукачів, вакансій, роботодавців у базі даних агентства з підбору кадрів «Новий шанс» та співставлення наявних вакансій та заяв від шукачів з метою їх подальшого працевлаштування.

Функціональне призначення

Функціональним призначенням програмного забезпечення для агентства з підбору кадрів «Новий шанс» є автоматизація підбору вакансій для агентства з підбору кадрів «Новий шанс», яка покращить та полегшить обробку потрібної інформації.

3 Вимоги до програмного забезпечення

3.1 Вимоги до функціональних характеристик

3.1.1 Вимоги до складу виконуваних функцій

Програмне забезпечення повинно виконувати ряд наступних функцій:

- авторизація;
- перегляд, введення та редагування даних;
- виконання адміністративних функцій (підтримка та збереження цілісності бази даних);
- отримання списків роботодавців, пропозицій, шукачів, вакансій;
- формування звіту про наявність вакансій по критерію (наприклад, вакансії для інвалідів);
- формування звіту про працевлаштованих;
- формування звіту про роботодавців.

3.1.2 Вимоги до організації вхідних та вихідних даних

Введення даних здійснюється за допомогою клавіатури. Данні вводяться вручну або обираються з випадających списків.

Обмін інформацією між програмним забезпеченням і базою даних відбувається за допомогою файлів з довільним доступом, з розширенням .mdb.

Вхідні дані:

- інформація про шукачів;
- інформація про вакансії;
- інформація про роботодавців.

Вихідні дані:

- список шукачів;
- список пропозицій;
- список роботодавців;
- звіт про наявність вакансій по критерію;

- звіт про працевлаштованих;
- звіт про роботодавців.

Довідник «Пропозиції» повинен характеризуватись такими даними: код роботодавця, код вакансії, заробітна плата, режим роботи, кількість пропозицій, дата пропозиції, потрібна освіта, потрібна спеціальність, потрібний вік, потрібна стать, інвалідність.

Довідник «Працевлаштування» повинен характеризуватись такими даними: код шукача, дата заявки, відмова, код вакансії.

Довідник «Роботодавці» повинен характеризуватись такими даними: код роботодавця, назва підприємства, інформація, керівник.

Довідник «Шукачі» повинен характеризуватись такими даними: код шукача, ПІБ, дата народження, адреса, телефон, освіта, спеціальність, стать, інвалідність.

Довідник «ДКП» повинен характеризуватись такими даними: код спеціальності, назва.

3.2 Вимоги до надійності

Програмне забезпечення повинно нормально функціонувати при безперебійній роботі комп'ютера. При виникненні збоїв в роботі, відновлення нормальної роботи повинне проводитися після перезавантаження програми.

Для забезпечення надійності інформації повинна використовуватися СКБД, що забезпечує цілісність транзакцій і несе відповідальність за цілісність інформації.

Система повинна продовжувати коректне функціонувати при втраті частини інформації. У випадку неможливого продовження коректної роботи має бути повідомлення про це.

У разі введення користувачем некоректної інформації система повинна повідомити про помилку і надати можливість виправити її.

3.3 Вимоги до умов експлуатації

Кліматичні умови експлуатації, при яких повинні забезпечуватися задані характеристики, повинні відповідати вимогам, що пред'являються до технічних засобів в частині умов їх експлуатації.

Користувач повинен мати базові навички роботи з ноутбуком або комп'ютером.

3.4 Вимоги до складу та параметрів технічних засобів

Комп'ютер кінцевого користувача повинен задовольняти наступним мінімімальним вимогам:

- CPU: Intel(R) i5 2,16 МГц або аналогічний за параметрами AMD;
- RAM: 4 GB;
- HDD: 10 GB вільного місця.

3.5 Вимоги до інформаційної та програмної сумісності

Для повноцінного функціонування програми необхідно використовувати операційну систему Windows версії не нижче 7. Для роботи системи необхідно СУБД Access версії не нижче Access 2003.

3.6 Вимоги до маркування та пакування

Продукт буде упакований в електронний архів і мати найменування New_Chance.rar та містити:

- виконуваний файл New_Chance.exe;
- файл з описом ReadMe.txt;
- файл, який містить структуру БД New_Chance.mdb.

4 Вимоги до програмної документації

Склад програмної документації повинен включати в себе:

- технічне завдання на розробку програмного забезпечення;
- код програмного забезпечення;
- опис програмного забезпечення;
- інструкцію користувача програмного забезпечення;
- програму та методику випробування програмного забезпечення.

5 Техніко-економічні показники

Техніко-економічні показники не розраховуються у зв'язку з тим, що програмного забезпечення розробляється в рамках кваліфікаційної роботи.

6 Стадії та етапи розробки

Стадії та етапи розробки програмного забезпечення представлені у таблиці А.1

Таблиця А.1 – Стадії та етапи розробки програмного забезпечення

№ п/п	Назва етапів роботи	Кінцевий термін виконання
1.	Аналіз практичної задачі інженерії ПЗ	17.02.2025
2.	Аналіз вимог до ПЗ	28.03.2025
3.	Розробка архітектури ПЗ	08.04.2025
4.	Розробка проєкту ПЗ	15.04.2025
5.	Реалізація та тестування ПЗ	28.04.2025

7 Порядок контролю та приймання

Контроль за аналізом та проєктуванням кожної окремої частини програмного забезпечення відбувається на кожному етапі з урахуванням вимог, визначених у технічному завданні. Кожна стадія розробки повинна бути представлена в зазначені строки та узгоджена із замовником.

Приймання проводяться відповідно до програми і методики випробувань і документують за допомогою протоколу проведення випробувань. У разі знаходження помилок під час прийому програмного забезпечення складається акт про знайдені помилки, який підписуються представниками замовника і розробника і затверджуються керівником організації – замовника та організації – розробника. Розробник повинен на протязі не більше ніж 2 тижнів виправити зазначені помилки і оповістити замовника про повторне проведення перевірки.

Додаток Б Код програмного забезпечення для агентства з підбору кадрів «Новий шанс»

```

#include <vcl.h>
#pragma hdrstop
#include <Dblogdlg.hpp>
//-----
#pragma package(smart_init)
#pragma resource "*.dfm"
TMain *Main;
//-----
__fastcall TMain::TMain(TComponent* Owner)
    : TForm(Owner)
{
}
//-----
void __fastcall TMain::BtCloseClick(TObject *Sender)
{
    Close();
}
//-----
void __fastcall TMain::BtPropozClick(TObject *Sender)
{
    FormPropoz->Show();
}
//-----
void __fastcall TMain::BtShukClick(TObject *Sender)
{
    FormShuk->Show();
}
//-----
void __fastcall TMain::BtRobotodClick(TObject *Sender)
{
    FormRobotod->Show();
}
//-----
void __fastcall TMain::BtVakClick(TObject *Sender)
{
    FormVak->Show();
}
//-----
void __fastcall TMain::BtPodDovClick(TObject *Sender)
{
    FormPodDov->Show();
}
//-----
void __fastcall TMain::BtPodObrobkaClick(TObject *Sender)
{
    FormPodObrobka->Show();
}
//-----
void __fastcall TMain::FormShow(TObject *Sender)
{
    Main->Top=261;
}

```

```

Main->Left=474;
AnsiString una, pwd;
label: if (LoginDialog("My base", una,pwd))
{
if ((una=="Admin") && (pwd=="123qwe"))
Main->Show();
else {
ShowMessage("Невірна інформація");
goto label;
};
}
else Close();
}
//-----
void __fastcall TMain::BtPodZapClick(TObject *Sender)
{
FormPodQuery->Show();
}
//-----
void __fastcall TMain::BtPodOtchClick(TObject *Sender)
{
FormPodReport->Show();
}
//-----
void __fastcall TMain::FormCreate(TObject *Sender)
{
}
//-----

```

Модуль Довідники

```

#include <vcl.h>
#pragma hdrstop
//-----
#pragma package(smart_init)
#pragma resource "*.dfm"
TFormPodDov *FormPodDov;
//-----
__fastcall TFormPodDov::TFormPodDov(TComponent* Owner)
: TForm(Owner)
{
}
//-----
void __fastcall TFormPodDov::BtDkpClick(TObject *Sender)
{
FormDkp->Show();
}
//-----
void __fastcall TFormPodDov::BtClose8Click(TObject *Sender)
{
Close();
}
//-----
void __fastcall TFormPodDov::FormShow(TObject *Sender)
{
FormPodDov->Top=261;
FormPodDov->Left=474;
}
//-----
void __fastcall TFormPodDov::BtVakanClick(TObject *Sender)

```

```

{
FormVak->Show();
}
//-----
void __fastcall TFormPodDov::FormCreate(TObject *Sender)
{
}
//-----

Модуль Вакансії
//-----
#include <vcl.h>
#pragma hdrstop
//-----
#pragma package(smart_init)
#pragma resource "*.dfm"
TFormVak *FormVak;
//-----
__fastcall TFormVak::TFormVak(TComponent* Owner)
: TForm(Owner)
{
}
//-----
void __fastcall TFormVak::BtClose6Click(TObject *Sender)
{
Close();
}
//-----
void __fastcall TFormVak::FormShow(TObject *Sender)
{
FormVak->Top=261;
FormVak->Left=254;
}
//-----
void __fastcall TFormVak::DBGrid1CellClick(TColumn *Column)
{
TedKodVak->Text=ADOVak->Fields->Fields[0]->AsString;
TedNazva->Text=ADOVak->Fields->Fields[1]->AsString;
TedDKP->Text=ADOVak->Fields->Fields[2]->AsString;
}
//-----
void __fastcall TFormVak::BtCloseClick(TObject *Sender)
{
Close();
}
//-----
void __fastcall TFormVak::BtAddClick(TObject *Sender)
{
TLocateOptions Opts;
Opts.Clear();
Opts << loPartialKey;
{
if ((TedKodVak->Text=="") || (TedNazva->Text==""))
{ ShowMessage("Не всі поля заповнено");
TedKodVak->SetFocus();
}
else
if (ADOVak->Locate("Код_вакансії", StrToInt(TedKodVak->Text), Opts))
{ ShowMessage("Код дублюється! Будь ласка, введіть унікальний код!");
TedKodVak->SetFocus(); }
}
}

```

```

else
if (MessageDlg("Додати? Так,
Hi", mtInformation, TMsgDlgButtons() << mbYes << mbNo, 0) == mrYes)
ADOVak->InsertRecord (ARRAYOFCONST ((StrToInt (TedKodVak->Text),
TedNazva->Text, TedDKP->Text)));
}
TedKodVak->Clear();
TedNazva->Clear();
TedDKP->Clear();
}
//-----

void __fastcall TFormVak::BtDeleteClick(TObject *Sender)
{
if (MessageDlg("Видалити? Так,
Hi", mtInformation, TMsgDlgButtons() << mbYes << mbNo, 0) == mrYes)
ADOVak->Delete();
}
//-----

void __fastcall TFormVak::BtEditClick(TObject *Sender)
{
ADOVak->Edit();
ADOVak-> SetFields (ARRAYOFCONST ((StrToInt (TedKodVak->Text),
TedNazva->Text, TedDKP->Text)));
ADOVak->Post();
}
//-----

void __fastcall TFormVak::BtClearClick(TObject *Sender)
{
for(int i=0; i<ComponentCount; i++)
{ TEdit* TedKodVak=dynamic_cast<TEdit*>(Components[i]);
if (TedKodVak)
TedKodVak->Text=" ";
}
}
//-----

void __fastcall TFormVak::FormCreate(TObject *Sender)
{
}
//-----

```

Модуль Обробка оперативної інформації

```

//-----
#include <vcl.h>
#pragma hdrstop
//-----
#pragma package(smart_init)
#pragma resource "*.dfm"
TFormPodObrobka *FormPodObrobka;
//-----

__fastcall TFormPodObrobka::TFormPodObrobka(TComponent* Owner)
: TForm(Owner)
{
}
//-----

void __fastcall TFormPodObrobka::BtPropozClick(TObject *Sender)
{
FormPropoz->Show();
}

```

```

}
//-----
void __fastcall TFormPodObrobka::BtShukClick(TObject *Sender)
{
FormShuk->Show();
}
//-----
void __fastcall TFormPodObrobka::BtVakClick(TObject *Sender)
{
FormVak->Show();
}
//-----
void __fastcall TFormPodObrobka::BtRobotodClick(TObject *Sender)
{
FormRobotod->Show();
}
//-----
void __fastcall TFormPodObrobka::BtCloseClick(TObject *Sender)
{
Close();
}
//-----
void __fastcall TFormPodObrobka::FormShow(TObject *Sender)
{
FormPodObrobka->Top=261;
FormPodObrobka->Left=474;
}
//-----
void __fastcall TFormPodObrobka::FormCreate(TObject *Sender)
{
}
//-----

```

Модуль Пропозиції

```

//-----
#include <vcl.h>
#pragma hdrstop
//-----
#pragma package(smart_init)
#pragma resource "*.dfm"
TFormPropoz *FormPropoz;
//-----
__fastcall TFormPropoz::TFormPropoz(TComponent* Owner)
: TForm(Owner)
{
}
//-----
void __fastcall TFormPropoz::BtClose7Click(TObject *Sender)
{
Close();
}
//-----
void __fastcall TFormPropoz::FormShow(TObject *Sender)
{
FormPropoz->Top=261;
FormPropoz->Left=254;
}
//-----
void __fastcall TFormPropoz::GPropozCellClick(TColumn *Column)

```

```

{
TedKodRob->Text=ADOPropoz->Fields->Fields[0]->AsString;
TedKodVak->Text=ADOPropoz->Fields->Fields[1]->AsString;
TedZplata->Text=ADOPropoz->Fields->Fields[2]->AsString;
TedRejim->Text=ADOPropoz->Fields->Fields[3]->AsString;
TedSocPak->Text=ADOPropoz->Fields->Fields[4]->AsString;
TedKilkist->Text=ADOPropoz->Fields->Fields[5]->AsString;
TedData->Text=ADOPropoz->Fields->Fields[6]->AsString;
TedOsvita->Text=ADOPropoz->Fields->Fields[7]->AsString;
TedSpec->Text=ADOPropoz->Fields->Fields[8]->AsString;
TedVik->Text=ADOPropoz->Fields->Fields[9]->AsString;
TedGender->Text=ADOPropoz->Fields->Fields[10]->AsString;
TedInval->Text=ADOPropoz->Fields->Fields[12]->AsString;
}
//-----
void __fastcall TFormPropoz::BtAddClick(TObject *Sender)
{
    {
        if ((TedData->Text=="") || (TedGender->Text=="") || (TedInval->Text=="") ||
            (TedKilkist->Text=="") || (TedKodRob->Text=="") || (TedKodVak->Text=="") ||
            (TedOsvita->Text=="") || (TedRejim->Text=="") || (TedSocPak->Text=="") ||
            (TedSpec->Text=="") || (TedVik->Text=="") || (TedZplata->Text==""))
        { ShowMessage("Не всі поля заповнено");
          TedKodRob->SetFocus();
        }
        else
        if (MessageDlg("Додати? Так,
Hi",mtInformation,TMsgDlgButtons()<<mbYes<<mbNo,0)==mrYes)
        ADOPropoz->InsertRecord(ARRAYOFCONST((StrToInt(TedKodRob->Text),
StrToInt(TedKodVak->Text),StrToInt(TedZplata->Text), TedRejim->Text,
TedSocPak->Text, StrToInt(TedKilkist->Text), TedData->Text, TedOsvita->Text,
StrToInt(TedSpec->Text), StrToInt(TedVik->Text), TedGender->Text,
TedInval->Text,)));
        }
        TedData->Clear();
        TedGender->Clear();
        TedInval->Clear();
        TedKilkist->Clear();
        TedKodRob->Clear();
        TedKodVak->Clear();
        TedOsvita->Clear();
        TedRejim->Clear();
        TedSocPak->Clear();
        TedSpec->Clear();
        TedVik->Clear();
        TedZplata->Clear();
    }
    //-----
void __fastcall TFormPropoz::BtDeleteClick(TObject *Sender)
{
    if (MessageDlg("Видалити? Так,
Hi",mtInformation,TMsgDlgButtons()<<mbYes<<mbNo,0)==mrYes)
    ADOPropoz->Delete();
}
//-----
void __fastcall TFormPropoz::BtClearClick(TObject *Sender)
{
    for(int i=0; i<ComponentCount; i++)
    { TEdit*TedKodRob=dynamic_cast<TEdit*>(Components[i]);

```

```

if (TedKodRob)
TedKodRob->Text=" ";
}
}
//-----
void __fastcall TFormPropoz::BtEditClick(TObject *Sender)
{
ADOPropoz->Edit();
    ADOPropoz-> SetFields (ARRAYOFCONST ((StrToInt (TedKodRob->Text),
        StrToInt (TedKodVak->Text), StrToInt (TedZplata->Text), TedRejim->Text,
        TedSocPak->Text, StrToInt (TedKilkist->Text), TedData->Text,
        TedOsvita->Text, StrToInt (TedSpec->Text), StrToInt (TedVik->Text),
        TedGender->Text, TedInval->Text)));
ADOPropoz->Post();
}
//-----
void __fastcall TFormPropoz::DBGrid2CellClick(TColumn *Column)
{
TedRejim->Text=ADOREjim->Fields->Fields[0]->AsString;
DBGrid2->Visible=false;
}
//-----
void __fastcall TFormPropoz::TedRejimEnter(TObject *Sender)
{
DBGrid2->Visible=true;
    DBGrid3->Visible=false;
    DBGrid4->Visible=false;
}
//-----
void __fastcall TFormPropoz::TedKodRobEnter(TObject *Sender)
{
DBGrid2->Visible=false;
    DBGrid3->Visible=false;
    DBGrid4->Visible=false;
}
//-----
void __fastcall TFormPropoz::TedKodVakEnter(TObject *Sender)
{
DBGrid2->Visible=false;
    DBGrid3->Visible=false;
    DBGrid4->Visible=false;
}
//-----
void __fastcall TFormPropoz::TedZplataEnter(TObject *Sender)
{
DBGrid2->Visible=false;
    DBGrid3->Visible=false;
    DBGrid4->Visible=false;
}
//-----
void __fastcall TFormPropoz::TedSocPakEnter(TObject *Sender)
{
DBGrid2->Visible=false;
    DBGrid3->Visible=false;
    DBGrid4->Visible=false;
}
//-----
void __fastcall TFormPropoz::TedKilkistEnter(TObject *Sender)
{

```

```

DBGrid2->Visible=false;
    DBGrid3->Visible=false;
    DBGrid4->Visible=false;
}
//-----
void __fastcall TFormPropoz::TedDataEnter(TObject *Sender)
{
    DBGrid2->Visible=false;
    DBGrid3->Visible=false;
    DBGrid4->Visible=false;
}
//-----
void __fastcall TFormPropoz::TedOsvitaEnter(TObject *Sender)
{
    DBGrid2->Visible=false;
    DBGrid3->Visible=true;
    DBGrid4->Visible=false;
}
//-----
void __fastcall TFormPropoz::TedVikEnter(TObject *Sender)
{
    DBGrid2->Visible=false;
    DBGrid3->Visible=false;
    DBGrid4->Visible=false;
}
//-----
void __fastcall TFormPropoz::TedGenderEnter(TObject *Sender)
{
    DBGrid2->Visible=false;
    DBGrid3->Visible=false;
    DBGrid4->Visible=false;
}
//-----
void __fastcall TFormPropoz::TedInvalEnter(TObject *Sender)
{
    DBGrid2->Visible=false;
    DBGrid3->Visible=false;
    DBGrid4->Visible=false;
}
//-----
void __fastcall TFormPropoz::DBGrid3CellClick(TColumn *Column)
{
    TedOsvita->Text=ADOOsv->Fields->Fields[0]->AsString;
    DBGrid3->Visible=false;
}
//-----
void __fastcall TFormPropoz::DBGrid4CellClick(TColumn *Column)
{
    TedSpec->Text=ADOSpec->Fields->Fields[0]->AsString;
    DBGrid4->Visible=false;
}
//-----
void __fastcall TFormPropoz::TedSpecEnter(TObject *Sender)
{
    DBGrid2->Visible=false;
    DBGrid3->Visible=false;
    DBGrid4->Visible=true;
}
//-----

```

```

void __fastcall TFormPropoz::FormCreate(TObject *Sender)
{
}
//-----

Модуль Шукачі
//-----
#include <vcl.h>
#pragma hdrstop
//-----
#pragma package(smart_init)
#pragma resource "*.dfm"
TFormShuk *FormShuk;
//-----
__fastcall TFormShuk::TFormShuk(TComponent* Owner)
    : TForm(Owner)
{
}
//-----
void __fastcall TFormShuk::BtClose2Click(TObject *Sender)
{
    Close();
}
//-----
void __fastcall TFormShuk::FormShow(TObject *Sender)
{
    FormShuk->Top=261;
    FormShuk->Left=254;
}
//-----
void __fastcall TFormShuk::DBGrid1CellClick(TColumn *Column)
{
    TedKodShuk->Text=ADOShuk->Fields->Fields[0]->AsString;
    TedFio->Text=ADOShuk->Fields->Fields[1]->AsString;
    TedDataNar->Text=ADOShuk->Fields->Fields[2]->AsString;
    TedAdresa->Text=ADOShuk->Fields->Fields[3]->AsString;
    TedTelefon->Text=ADOShuk->Fields->Fields[4]->AsString;
    TedOsvita->Text=ADOShuk->Fields->Fields[5]->AsString;
    TedSpec->Text=ADOShuk->Fields->Fields[6]->AsString;
    TedGender->Text=ADOShuk->Fields->Fields[7]->AsString;
    TedInval->Text=ADOShuk->Fields->Fields[8]->AsString;
}
//-----
void __fastcall TFormShuk::BtAddClick(TObject *Sender)
{
    TLocateOptions Opts;
    Opts.Clear();
    Opts << loPartialKey;
    {
        if ((TedAdresa->Text=="") || (TedDataNar->Text=="") || (TedFio->Text=="") ||
            (TedGender->Text=="") || (TedInval->Text=="") || (TedKodShuk->Text=="") ||
            (TedOsvita->Text=="") || (TedSpec->Text=="") || (TedTelefon->Text==""))
        { ShowMessage("Не всі поля заповнено");
          TedKodShuk->SetFocus();
        }
        else
        if (ADOShuk->Locate("Kod_shuk", StrToInt(TedKodShuk->Text), Opts))
        { ShowMessage("Код дублюється! Будь ласка, введіть унікальний код!");
          TedKodShuk->SetFocus();
        }
        else

```

```

if(MessageDlg("Додати? Так,
Hi",mtInformation,TMsgDlgButtons()<<mbYes<<mbNo,0)==mrYes)
ADOShuk->InsertRecord(ARRAYOFCONST((StrToInt(TedKodShuk->Text),
    TedFio->Text, TedDataNar->Text, TedAdresa->Text, TedTelefon->Text,
    TedOsvita->Text, TedSpec->Text, TedGender->Text, TedInval->Text)));
}
TedAdresa->Clear();
TedDataNar->Clear();
TedFio->Clear();
TedGender->Clear();
TedInval->Clear();
TedKodShuk->Clear();
TedOsvita->Clear();
TedSpec->Clear();
TedTelefon->Clear();
}
//-----
void __fastcall TFormShuk::BtDeleteClick(TObject *Sender)
{
if(MessageDlg("Видалити? Так,
Hi",mtInformation,TMsgDlgButtons()<<mbYes<<mbNo,0)==mrYes)
ADOShuk->Delete();
}
//-----
void __fastcall TFormShuk::BtEditClick(TObject *Sender)
{
ADOShuk->Edit();
    ADOShuk-> SetFields(ARRAYOFCONST((StrToInt(TedKodShuk->Text),
    TedFio->Text, TedDataNar->Text, TedAdresa->Text, TedTelefon->Text,
    TedOsvita->Text, TedSpec->Text, TedGender->Text, TedInval->Text)));
ADOShuk->Post();
}
//-----
void __fastcall TFormShuk::BtClearClick(TObject *Sender)
{
    for(int i=0; i<ComponentCount; i++)
    { TEdit*TedKodRob=dynamic_cast<TEdit*>(Components[i]);
    if (TedKodRob)
    TedKodRob->Text=" ";
    }
}
//-----
void __fastcall TFormShuk::DBGrid2CellClick(TColumn *Column)
{
TedOsvita->Text=ADOShukOsv->Fields->Fields[0]->AsString;
DBGrid2->Visible=false;
}
//-----
void __fastcall TFormShuk::TedOsvitaEnter(TObject *Sender)
{
    DBGrid2->Visible=true;
    DBGrid3->Visible=false;
}
//-----
void __fastcall TFormShuk::DBGrid3CellClick(TColumn *Column)
{
TedSpec->Text=ADOShukSpec->Fields->Fields[0]->AsString;
DBGrid3->Visible=false;
}

```

```

//-----
void __fastcall TFormShuk::TedSpecEnter(TObject *Sender)
{
    DBGrid3->Visible=true;
    DBGrid2->Visible=false;
}
//-----
void __fastcall TFormShuk::TedKodShukEnter(TObject *Sender)
{
    DBGrid3->Visible=false;
    DBGrid2->Visible=false;
}
//-----
void __fastcall TFormShuk::TedFioEnter(TObject *Sender)
{
    DBGrid3->Visible=false;
    DBGrid2->Visible=false;
}
//-----
void __fastcall TFormShuk::TedDataNarEnter(TObject *Sender)
{
    DBGrid3->Visible=false;
    DBGrid2->Visible=false;
}
//-----
void __fastcall TFormShuk::TedAdresaEnter(TObject *Sender)
{
    DBGrid3->Visible=false;
    DBGrid2->Visible=false;
}
//-----
void __fastcall TFormShuk::TedTelefonEnter(TObject *Sender)
{
    DBGrid3->Visible=false;
    DBGrid2->Visible=false;
}
//-----
void __fastcall TFormShuk::TedGenderEnter(TObject *Sender)
{
    DBGrid3->Visible=false;
    DBGrid2->Visible=false;
}
//-----
void __fastcall TFormShuk::TedInvalEnter(TObject *Sender)
{
    DBGrid3->Visible=false;
    DBGrid2->Visible=false;
}
//-----
void __fastcall TFormShuk::FormCreate(TObject *Sender)
{
}
//-----

Модуль Запити
//-----
#include <vcl.h>
#pragma hdrstop
//-----

```

```

#pragma package(smart_init)
#pragma resource "*.dfm"
TFormPodQuery *FormPodQuery;
//-----
__fastcall TFormPodQuery::TFormPodQuery(TComponent* Owner)
    : TForm(Owner)
{
}
//-----
void __fastcall TFormPodQuery::FormShow(TObject *Sender)
{
    FormPodQuery->Top=261;
    FormPodQuery->Left=474;
}
//-----
void __fastcall TFormPodQuery::BtSopostClick(TObject *Sender)
{
    FormQuerySopost->Show();
}
//-----
void __fastcall TFormPodQuery::BtCloseClick(TObject *Sender)
{
    Close();
}
//-----
void __fastcall TFormPodQuery::BtPretendSpecClick(TObject *Sender)
{
    FormQueryPretendSpec->Show();
}
//-----
void __fastcall TFormPodQuery::BtVakOsvClick(TObject *Sender)
{
    FormQueryVakOsv->Show();
}
//-----
void __fastcall TFormPodQuery::BtVakInvalClick(TObject *Sender)
{
    FormQueryVakInval->Show();
}
//-----
void __fastcall TFormPodQuery::BtKstVidmovClick(TObject *Sender)
{
    FormQueryKstVidmov->Show();
}
//-----
void __fastcall TFormPodQuery::FormCreate(TObject *Sender){
}

```

Додаток В Опис програмного забезпечення для агентства з підбору кадрів «Новий шанс»

1 Загальні відомості

Розроблене програмне забезпечення для агентства з підбору кадрів «Новий шанс» представляє собою програму для співробітників агентства, які виконують роботу по працевлаштуванню клієнтів агентства.

Експлуатаційним призначенням програмного забезпечення для агентства з підбору кадрів «Новий шанс» є автоматизація зберігання, перегляду, пошуку, редагування та видалення інформації шукачів, вакансій, роботодавців у базі даних агентства з підбору кадрів «Новий шанс» та співставлення наявних вакансій та заяв від шукачів з метою їх подальшого працевлаштування.

2 Функціональне призначення

Функціональним призначенням програмного забезпечення для агентства з підбору кадрів «Новий шанс» є автоматизація підбору вакансій для агентства з підбору кадрів «Новий шанс», яка покращить та полегшить обробку потрібної інформації.

Програмне забезпечення виконує наступні функції:

- авторизація;
- перегляд, введення та редагування даних;
- виконання адміністративних функцій (підтримка та збереження цілісності бази даних);
- отримання списків роботодавців, пропозицій, шукачів, вакансій;
- формування звіту про наявність вакансій по критерію (наприклад, вакансії для інвалідів);
- формування звіту про працевлаштованих;

– формування звіту про роботодавців.

3 Опис логічної структури

Програма розроблена на модульній основі, тобто кожний компонент (модуль) програми відповідає за конкретну дію. Дані, необхідні для роботи програми, зберігаються у нормалізованих таблицях бази даних.

4 Технічні та програмні засоби

Комп'ютер кінцевого користувача повинен задовольняти наступним мініміальним вимогам:

- CPU: Intel(R) i5 2,16 МГц або аналогічний за параметрами AMD;
- RAM: 4 GB;
- HDD: 10 GB вільного місця.

Для повноцінного функціонування програми необхідно використовувати операційну систему Windows версії не нижче 7. Для роботи системи необхідно СУБД Access версії не нижче Access 2003.

5 Виклик та завантаження

Програмне забезпечення для агентства з підбору кадрів «Новий шанс» встановлюється безпосередньо на комп'ютер користувача. Програмне забезпечення відкривається або за допомогою ярлика, який знаходиться на робочому столі, або за допомогою вибору відповідного пункту меню «Пуск» – «Програми». Програма має один режим використання.

Після запуску програми з'являється вікно авторизації користувача, де пропонується ввести логін та пароль для входу в систему. Якщо дані користувача введені вірно, відкривається головна форма програми.

6 Вхідні та вихідні дані

Введення даних здійснюється за допомогою клавіатури. Данні вводяться вручну або обираються з випадаючих списків.

Обмін інформацією між програмним забезпеченням і базою даних відбувається за допомогою файлів з довільним доступом, з розширенням .mdb.

Вхідні дані:

- інформація про шукачів;
- інформація про вакансії;
- інформація про роботодавців.

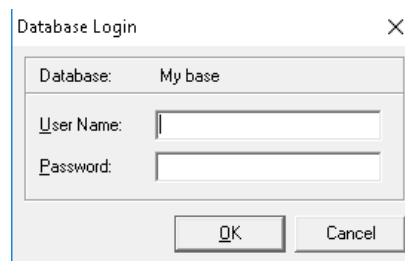
Вихідні дані:

- список шукачів;
- список пропозицій;
- список роботодавців;
- звіт про наявність вакансій по критерію;
- звіт про працевлаштованих;
- звіт про роботодавців.

Додаток Г Інструкція користувача програмного забезпечення для агентства з підбору кадрів «Новий шанс»

Програмне забезпечення для агентства з підбору кадрів «Новий шанс» встановлюється безпосередньо на комп'ютер користувача. Програмне забезпечення відкривається або за допомогою ярлика, який знаходиться на робочому столі, або за допомогою вибору відповідного пункту меню «Пуск» – «Програми». Програма має один режим використання.

Після запуску програми з'являється форма авторизації (див.рис. Г.1)



The image shows a standard Windows-style dialog box titled "Database Login". Inside the dialog, there is a label "Database:" followed by a text box containing "My base". Below this are two more text boxes, one labeled "User Name:" and the other "Password:". At the bottom right of the dialog are two buttons: "OK" and "Cancel".

Рисунок Г.1 – Форма авторизації

Якщо дані авторизації введено не вірно або якісь поля не заповнені, то система виводить повідомлення про помилку.

При отриманні доступу користувач повинен вибрати режим роботи (див. рис Г.2).

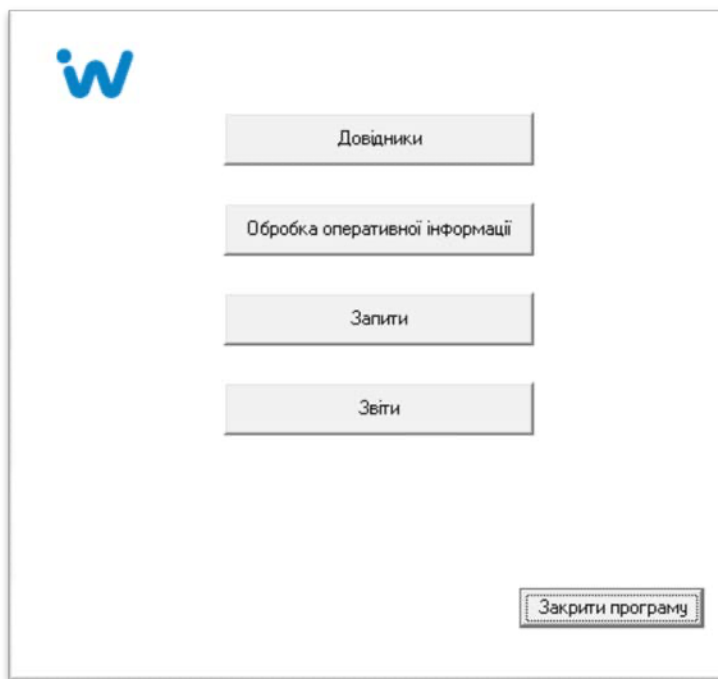


Рисунок Г.2 – Режими роботи програми

Якщо вибрано "Обробка оперативної інформації", з'являється форма вибору наборів даних, яка містить такі кнопки: "Пропозиції", "Шукачі", "Роботодавці".

При спробі занести в БД неповну інформацію видається повідомлення про помилку

Якщо вибрано "Роботодавці", з'являється форма для заповнення (див. рис. Г.3).

Код роботодавця	Назва	Інформація	Керівник
1	ТОВ "Інди"	вул.Чигиринська 12 58-43-56	Охрипний Р.А.
2	ТОВ "Рогин"	вул.Монетська 31 32-61-50	Шоприн Д.А.
3	ТОВ "Ікар"	вул.Оливна 41 53-60-54	Новин Р.Г.
4	ТОВ "Женева"	вул.Пропускна 89 89-61-70	Шепард Л.П.
5	ТОВ "Транзит"	вул.Робоча 23 60-87-32	Гревно Т.А.
6	ТОВ "Перетин"	вул. Переяславська 116 51-98-11	Дожина К.О.
7	ТОВ "Торенторен"	вул. Слободянська 120 43-21-12	Левченко Л.І.
8	ТОВ "Лерика"	вул. Заруби 31 30-22-41	Горич О.В.
9	Управління ЖКГ Миколаївської облдержадміністрації	вул. Ординська 40 21-85-43	Летов Е.Г.
10	ТОВ "Селікарс"	вул. Грибоїдова 93 81-79-85	Перитська О.К.
11	ТОВ "Позенкарц"	вул. Тривала 12 12-81-25	Залежан В.В.

Код роботодавця Назва Інформація Керівник

Рисунок Г.3 – Форма для заповнення набору даних "Роботодавці"

Якщо вибрано "Довідники", з'являється форма вибору довідників: "Державний класифікатор професій" та "Вакансії".

Якщо вибрано "Вакансії", з'являється форма для заповнення (див. рис. Г.4).

Код вакансії	Назва
1	Системний адміністратор
2	Секретар
3	Продавець
4	Бухгалтер
5	Прибиральник
6	Агент комерційний
7	Агент страховий
8	Адвокат
9	Адміністратор
10	Акушерка
11	Аналітик комп'ютерного банку даних
12	Аналітик комп'ютерних систем
13	Верхолаз
14	Підрильник
15	Ведучий програми

Код вакансії Назва

Додати Видалити Редагувати Очистити поля

Close

Рисунок Г.4 – Форма для заповнення довідника "Вакансії"

При виборі функції “Запити” відкривається наступна форма (див. рис. Г.5).

Пошук збіжностей
 Претендент за спеціальністю
 Вакансії за освітою
 Вакансії для інвалідів
 Кількість відмов

Закрити

Рисунок Г.5 – Форма запитів

При виборі запиту "Пошук збіжностей" відкривається форма, в якій необхідно ввести ПІБ шукача. Вона має такий вигляд (див. рис. Г.6).

ФІО Шукача	Назва вакансії	Рівень освіти	Заробітна плата	Назва підприємства
► Назаренко В.І.	Прибиральник	Вища освіта 1,2 категорія	6500	ТОВ "Перетин"
Назаренко В.І.	Агент страховий	Вища освіта 1,2 категорія	15500	ТОВ "Лерика"
Назаренко В.І.	Акушерка	Вища освіта 1,2 категорія	16000	ТОВ "Позенкарц"
Назаренко В.І.	Секретар	Вища освіта 1,2 категорія	14000	ТОВ "Лерика"
Назаренко В.І.	Акушерка	Вища освіта 1,2 категорія	16000	ТОВ "Транзит"
Назаренко В.І.	Секретар	Вища освіта 1,2 категорія	14000	Управління ЖКГ Миколаївської облдержадміністрації

Введіть прізвище та ініціали людини
Наприклад: Петров І.Г.

Рисунок Г.6 – Запит “Пошук збіжностей”

При виборі функції “Звіти” відкривається форма з переліком звітів, до яких відносяться: "Вакансії для інвалідів" та "Переатестація" (див. рис. Г.7).



Вакансії для інвалідів

Переатестація

Рисунок Г.7 – Меню “Звіти”

Фрагмент звіту “Вакансії для інвалідів” наведено на рис. Г.8.

Вакансії для інвалідів

Назва	Інвалідність	Кількість
ТОВ "Імди"	True	2
Назва вакансії	Системний адміністратор	
ТОВ "Ікар"	True	4
Назва вакансії	Системний адміністратор	
ТОВ "Транзит"	True	3
Назва вакансії	Прибиральник	
ТОВ "Перетин"	True	6
Назва вакансії	Прибиральник	

Рисунок Г.8 – Фрагмент вакансій для інвалідів

Завдяки розробленому максимально ефективному інтерфейсу користувача, працювати з програмним забезпеченням може користувач, який має базові навички роботи з ноутбуком або комп’ютером.

Додаток Д Програма та методика випробування програмного забезпечення для агентства з підбору кадрів «Новий шанс»

Об'єкт випробування

Об'єктом випробування є програмне забезпечення для агентства з підбору кадрів «Новий шанс», розроблене в рамках даної кваліфікаційної роботи.

Мета випробування

Метою проведення випробування є перевірка працездатності даного програмного продукту, перевірка відповідності характеристик та вимог розробленої програми, викладених у постановці задачі, приведення прикладу роботи та отримання відповідних навичок.

Вимоги до програми

При проведенні випробувань функціональні характеристики (можливості) програми підлягають перевірці на відповідність вимогам, викладеним у постановці задачі.

Вимоги до програмної документації

1) Склад програмної документації, пропонованої на випробування

Програмна документація повинна включати в себе наступні документи:

- технічне завдання;
- код програми;
- опис програми;
- інструкцію користувача;
- програму та методика випробування.

2) Спеціальні вимоги

Спеціальні вимоги до програмної документації не висуваються.

Засоби і порядок випробувань

1) Технічні засоби, що використовуються під час випробувань

Склад технічних засобів, що використовувалися під час випробувань:

CPU: Intel(R) i5 2,16 МГц або аналогічний за параметрами AMD;

RAM: 4 GB;

HDD: 10 GB вільного місця.

Роздільна здатність екрану – 1280 x 1024 пікселів;

Клавіатура 101/102-х клавішна рус / лат;

Маніпулятор «миша».

2) Програмні засоби, що використовуються під час випробувань

Для повноцінного функціонування програми необхідно використовувати операційну систему Windows версії не нижче 7. Для роботи системи необхідно СУБД Access версії не нижче Access 2003.

3) Порядок проведення випробувань

Порядок проведення випробувань складається з перевірки вимог до функціональних характеристик програмного продукту та перевірки вимог до програмної документації.

Методи випробувань

1) Методика проведення перевірки вимог до програмної документації

Перевірка дотримання вимог програмної документації на програмний продукт виконується візуально представником служби, відповідальної за експлуатацію. У ході перевірки зіставляється склад і комплектність програмної документації, представленої розробником, з переліком програмної документації,

наведеним у пункті «Склад програмної документації, пропонованої на випробування» цього документа.

Перевірка вважається завершеною у випадку відповідності складу та комплектності програмної документації, представленої розробником, переліком програмної документації, наведеному у зазначеному вище пункті.

2) Методика проведення перевірки вимог до функціональних характеристик програмного продукту

Перевірка працездатності програми виконується згідно з пунктом «Вимоги до функціональних характеристик» технічного завдання.

Перевірка вважається завершеною у разі відповідності складу і послідовності дій, при виконанні даної перевірки, вказаною вище пункту технічного завдання.

В процесі тестування потрібно перевірити функціональність програмного забезпечення для агентства з підбору кадрів «Новий шанс». У таблиці Д.1, що наведена нижче, вказано перелік випробувань основних функціональних можливостей.

Таблиця Д.1 – Методика випробування

№	Дія	Очікуваний результат	Результат перевірки	Зауваження
1	2	3	4	5
1	Перевірка правильності авторизації у програмі	Авторизацію пройдено при вводі правильного логіну та паролю		
2	Перевірка правильності переходів на вкладку довідників	Відкривається вікно з довідниками		
3	Перевірка правильності додання запису про шукача	Програма додає запис про шукача		

Продовження таблиці Д.1

1	2	3	4	5
4	Перевірка правильності видалення даних шукача	Програма видаляє потрібний запис про шукача		
5	Перевірка правильності формування звітів	Програма формує звіт з даними на тему, яку обрав користувач		
6	Перевірка правильності функції пошуку вакансій для шукачів	Програма виводить список вакансій, на які може претендувати шукач		