

Вінницький національний технічний університет
(повне найменування вищого навчального закладу)

Факультет інтелектуальних інформаційних технологій та автоматизації
(повне найменування інституту, назва факультету (відділення))

Кафедра комп'ютерних наук
(повна назва кафедри (предметної, циклової комісії))

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

«Інформаційна технологія прогнозування обсягу закупівлі квітів»

Виконала: студентка 2-го курсу, групи
ІКН-21м спеціальності 122

«Комп'ютерні науки»

(шифр і назва напрямку підготовки, спеціальності)

Климчук А.М.
(прізвище та ініціали)

Керівник: к.т.н., ст.викл. кафедри КН

Озеранський В.С.
(прізвище та ініціали)

« 15 » 12 2022 р.

Опонент: к.т.н., доц. кафедри КСУ

Юхимчук М.С.
(прізвище та ініціали)

« 15 » 12 2022 р.

Допущено до захисту

Завідувач кафедри КН

д.т.н., проф. Яровий А. А.

(прізвище та ініціали)

« 16 » 12 2022 р.

Вінницький національний технічний університет
 Факультет інтелектуальних інформаційних технологій та автоматизації
 Кафедра комп'ютерних наук
 Рівень вищої освіти II-й (магістерський)
 Галузь знань – 12 Інформаційні технології
 Спеціальність – 122 Комп'ютерні науки
 Освітньо-професійна програма – Системи штучного інтелекту

ЗАТВЕРДЖУЮ


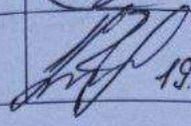
Завідувач кафедри КН
 Д.т.н., проф. Яровий А.А.

14. 09. 2022 року

ЗАВДАННЯ НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТЦІ

Климчук Анастасії Михайлівні
 (прізвище, ім'я, по батькові)

1. Тема роботи «Інформаційна технологія прогнозування обсягу закупівлі квітів»
 керівник роботи к.т.н., ст. викл. каф. КН, Озеранський В. С.
 затверджені наказом вищого навчального закладу від «16» 09 2022 року №203
2. Строк подання студентом роботи 18 листопада 2022 року
3. Вихідні дані до роботи: мова програмування – об'єктно-орієнтована; мінімальний період прогнозування – 4 днів; мінімальна кількість товару – 10 од.; максимально можлива кількість замовлення – 1000 од.; тип інтерфейсу користувача інтуїтивно-зрозумілий.
4. Зміст текстової частини: вступ, обґрунтування доцільності розробки інформаційної технології прогнозування обсягу закупівлі квітів, моделювання інформаційної технології прогнозування обсягу закупівлі квітів, структурна організація та особливості програмної реалізації інформаційної технології прогнозування обсягу закупівлі квітів, економічна частина, висновки, список використаних джерел, додатки.
5. Перелік ілюстративного матеріалу (з точним зазначенням обов'язкових креслень): схема загального алгоритму функціонування системи; структура інформаційної технології; модель функціонування системи прогнозування обсягу закупівлі квітів; схема алгоритму запису інформації; UML-діаграма класів; діаграма діяльності системи при прогнозуванні; приклад роботи програми.

6. Консультанти розділів роботи		Підпис, дата	
Розділ	Прізвище, ініціалита посада консультанта	завдання видав	виконав
1-3 Спеціальна частина	Озеранський В. С. к.т.н., ст. викладач каф. КН		14.09.2022
4 Економічна частина	Нікіфорова Л. О., к. е. н., доц. каф. ЕПВМ		19.09.2022


7. Дата видачі завдання 14.09. 2022 року


КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів магістерської кваліфікаційної роботи	Строк виконання етапів роботи	Примітки
1	Обґрунтування доцільності розробки інформаційної технології прогнозування обсягу закупівлі квітів	14.09.2022р - - 1.10.2022р	Розділ 1
2	Модельовання інформаційної технології прогнозування обсягу закупівлі квітів	2.10.2022р - - 16.10.2022р	Розділ 2
3	Структурна організація та особливості програмної реалізації інформаційної технології	17.10.2022р - - 07.11.2022р	Розділ 3
4	Підготовка економічної частини	08.11.2022р - - 21.11.2022р	Розділ 4
5	Апробація та/або впровадження результатів дослідження	23.11.2022р - 01.12.2022р	Тези доповідей
6	Оформлення пояснювальної записки, графічного матеріалу та презентації	02.12.2022р - - 14.12.2022р	Пояснювальна записка, графічний матеріал, презентація

Студент

Керівник роботи


(підпис)


(підпис)

Климчук А.М.

Озеранський В.С.

АНОТАЦІЯ

УДК 519.87

Климчук А. М. Інформаційна технологія прогнозування обсягу закупівлі квітів. Магістерська кваліфікаційна робота зі спеціальності 122 – Комп'ютерні науки, освітня програма – Комп'ютерні науки. Вінниця: ВНТУ, 2022. 112 с.

На укр. мові. Бібліогр.: 42 назв; рис.: 23; табл. 7.

Дана магістерська кваліфікаційна робота присвячена розробці інформаційної технології прогнозування обсягу закупівлі квітів. Виконано аналіз сучасних програм-аналогів, які використовуються для прогнозування обсягу закупівлі, наведено коротку порівняльну характеристику знайдених програм-аналогів, досліджено методи, які можуть бути використані для реалізації поставленої задачі. Проведено обґрунтування моделі прогнозування обсягу закупівлі квітів. Як базовий метод запропоновано застосувати метод прогнозування часових рядів. Розроблено алгоритм прогнозування обсягу закупівлі квітів та відповідне програмне забезпечення на мові програмування Java, в середовищі IntelliJ IDEA. Аналіз роботи програмного забезпечення показав підвищення точності прогнозу закупівлі квітів.

Графічна частина складається з 7 плакатів із результатами моделювання.

У розділі економічної частини здійснено оцінювання комерційного потенціалу розробки інформаційної технології прогнозування обсягу закупівлі квітів, проведено оцінювання комерційного потенціалу розробки, спрогнозовано витрати на виконання наукової роботи та впровадження результатів, які склали грн, розраховано період окупності – роки.

Ключові слова: інформаційна технологія, метод прогнозування часових рядів, прогноз, ковзне середнє, фактор сезонності.

ABSTRACT

Klymchuk A. M. Information technology for forecasting the volume of flower purchases. Master's degree in specialty 122 - Computer Science, educational program - Computer Science. Vinnytsia: VNTU, 2022. 112 p.

In Ukrainian language. Bibliogr .: 42 titles; fig .: 23; table 7.

This master's qualification work is devoted to the development of information technology for forecasting the volume of flower purchases. The analysis of modern programs-analogues used for forecasting the volume of procurement is performed, a brief comparative description of the found programs-analogues is given, the methods that can be used to implement the task are investigated. The model of forecasting the volume of flower purchases is substantiated. It is proposed to use the method of forecasting time series as a basic method. An algorithm for predicting the purchase of flowers and the corresponding software in the Java programming language in the IntelliJ IDEA environment has been developed. Analysis of the software showed an increase in the accuracy of the forecast of the purchase of flowers.

The graphic part consists of 7 posters with simulation results.

In the section of the economic part the estimation of commercial potential of development of information technology of forecasting of volume of purchase of flowers is carried out, the estimation of commercial potential of development is carried out, expenses for performance of scientific work and introduction of results which have made UAH are forecasted.

Keywords: information technology, time series forecasting method, forecast, moving average, seasonality factor.

ЗМІСТ

ВСТУП	8
1 ОБҐРУНТУВАННЯ ДОЦІЛЬНОСТІ РОЗРОБКИ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПРОГНОЗУВАННЯ ОБСЯГУ ЗАКУПІВЛІ КВІТІВ.....	12
1.1 Особливості ринку квітів в Україні.....	12
1.2 Дослідження принципів прогнозування та планування обсягу закупівлі квітів для квіткових магазинів.....	14
1.3 Існуючі моделі та методи, що застосовуються для реалізації поставленої задачі	14
1.4 Аналіз відомих програмних реалізацій для прогнозування обсягу закупівлі квітів	19
1.5 Постановка задачі.....	24
1.6 Висновок до розділу 1	26
2 МОДЕЛЮВАННЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПРОГНОЗУВАННЯ ОБСЯГУ ЗАКУПІВЛІ КВІТІВ.....	31
2.1 Обґрунтування вибору методу прогнозування обсягу закупівлі квітів.....	31
2.2 Розробка математичної моделі прогнозування обсягу закупівлі квітів	33
2.3 Проектування структури інформаційної технології прогнозування обсягу закупівлі квітів.....	37
2.4 Розробка алгоритму функціонування інформаційної технології.....	40
2.5 Висновок до розділу 2	44
3 СТРУКТУРНА ОРГАНІЗАЦІЯ ТА ОСОБЛИВОСТІ ПРОГРАМНОЇ РЕАЛІЗАЦІЇ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПРОГНОЗУВАННЯ ОБСЯГУ ЗАКУПІВЛІ КВІТІВ	45
3.1 Обґрунтування вибору мови та середовища програмування	45
3.2 Обґрунтування вибору бібліотек та фреймворків	51
3.3 Опис процесу кодування	56
3.4 Опис класів і функцій інформаційної технології прогнозування обсягу закупівлі квітів.....	59

3.5 Тестування інформаційної технології прогнозування обсягу закупівлі квітів	61
3.6 Висновок до розділу 3	69
4 ЕКОНОМІЧНА ЧАСТИНА	70
4.1 Комерційний та технологічний аудит науково-технічної розробки.....	70
4.2 Прогнозування витрат на виконання науково-дослідної (дослідно-конструкторської) роботи.....	74
4.3 Розрахунок економічної ефективності науково-технічної розробки за її можливої комерціалізації потенційним інвестором	81
4.4 Висновок до розділу 4	75
ВИСНОВКИ.....	87
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	89
Додаток А (обов'язковий) Результат перевірки на плагіат в онлайн-системі UNICHECK	93
Додаток Б (обов'язковий) Лістинг програми	89
Додаток В (обов'язковий) Ілюстративна частина.....	120
Додаток Г (довідниковий) Інструкція користувача	123

ВСТУП

Актуальність теми дослідження. На сьогоднішній день прогнозування обсягу закупівлі квітів із застосування сучасних інформаційних технологій зумовлена активним розвитком та ростом обсягів квіткового бізнесу, його затребуваністю та прибутковістю, а також складністю прогнозування доходів та обсягу закупівлі квітів вручну, що потребує розробки ефективних і доступних методів її прогнозування [3].

Діяльність кожної компанії на ринку важлива не лише для її власників, а й для держави в цілому. Оскільки в умовах науково-технічного прогресу вдосконалення системи економічного прогнозування стає одним із вирішальних чинників формування стратегії та тактики розвитку сучасних та майбутніх підприємств. Тому роль прогнозування сьогодні зростає. У свою чергу, планування як первинна функція управління пов'язує всі функції управління. Адже без планування, чіткого та раціонального визначення корпоративних цілей і діяльності це буде неможливо [4].

Квіти – один з головних атрибутів свята. Традиція дарувати квіти і прикрашати ними простір міцно встановилась в Україні. Таким чином, флористи завжди знаходять свого покупця, незважаючи на сезон та економічну кризу. Запит на квіти – своєрідне відображення фінансового стану компанії – чим краще, тим більше квітів ви придбаєте [3].

Планування збуту – це дуже важливий крок, оскільки від цього залежить постачання квіткової продукції. Погано розрахований час виконання може призвести до простою магазину або втрати через пошкодження продукту. Як правило, квіти купуються власниками торгових точок оптом. Планування продажів квітників базується на сезонності квіткової діяльності. Пік продажів припадає на наступні дати – 14 лютого, 14 жовтня, 8 березня, 1 вересня. Сьогодні існує велика кількість оптових баз квітів, які готові працювати з усіма споживачами, оскільки вони завжди мають велику кількість товарів. Іншим

важливим завданням у цьому плані є налагоджена система доставки квітів. Свіжі квіти слід купувати кожні 4–5 днів.

Актуальність інформаційної технології прогнозування обсягу закупівлі квітів полягає у потребі створення інструменту для полегшення роботи працівників та власників квіткових магазинів, складів або мереж квіткових магазинів, що забезпечить збільшення ефективності роботи з таких магазинів, а також збільшення прибутків, що досягнеться точністю прогнозування обсягів закупівлі квітів, зменшивши витрати та виключивши можливість псування великої кількості залишків, що були нереалізовані через відсутність попиту на них. Так як на сьогоднішній день існує проблема, що характеризується недостатньою точністю прогнозування серед існуючих програмних рішень для такого товару як зрізані квіти, розробка інформаційної технології прогнозування обсягу закупівлі квітів є актуальною.

Зв'язок роботи з науковими програмами, планами, темами. Магістерська кваліфікаційна робота виконана відповідно до напрямку наукових досліджень кафедри комп'ютерних наук Вінницького національного технічного університету 22 К1 «Моделі, методи, технології та пристрої інтелектуальних інформаційних систем управління, економіки, навчання та комунікацій» та плану наукової та навчально-методичної роботи кафедри.

Мета та завдання дослідження. Метою дослідження є підвищення точності визначення закупівлі обсягу квітів, що дозволить підвищити ефективність підприємницької діяльності за допомогою використання інформаційної технології прогнозування.

Для досягнення поставленої мети необхідно розв'язати наступні **задачі**:

- провести аналіз програм-аналогів та обґрунтувати доцільності розробки інформаційної технології прогнозування обсягу закупівлі квітів;
- розробити структуру інформаційної технології прогнозування обсягу закупівлі квітів;
- розробити математичну модель прогнозування обсягу закупівлі квітів;

- здійснити програмну реалізацію інформаційної технології прогнозування обсягу закупівлі квітів;
- провести тестування програми та проаналізувати отримані результати;
- економічно обґрунтувати доцільність розробки інформаційної технології прогнозування обсягу закупівлі квітів.

Об’єкт дослідження – це процес прогнозування обсягу закупівлі квітів.

Предмет дослідження – програмні засоби прогнозування обсягу закупівлі квітів.

Методи дослідження. У роботі використано такі методи наукових досліджень: метод системного аналізу для аналізу структури інформаційної системи; методи аналізу часових рядів; метод Байєса, методи експертних оцінок, регресійні методи; методи математичної статистики; методи об’єктно-орієнтованого програмування для автоматизації розрахунків.

Наукова новизна одержаних результатів полягає в наступному: удосконалено інформаційну технологію прогнозування обсягу закупівлі квітів, яка відрізняється від існуючих удосконаленим методом, прорахуванням тренду і сезонності що поєднує в собі метод ковзного середнього для прогнозування обсягу закупівлі сезонних квітів та методу експоненціального згладжування для несезонних квітів, що загалом дозволяє збільшити точність прогнозування необхідної кількості квітів для закупівлі.

Практичне значення одержаних результатів полягає у такому:

1. Удосконалено алгоритм прогнозування обсягу закупівлі квітів.
2. Здійснено програмну реалізацію інформаційної технології прогнозування обсягу закупівлі квітів.

Достовірність теоретичних положень магістерської кваліфікаційної роботи підтверджується строгістю постановки задач, коректним застосуванням математичних методів під час доведення наукових положень, строгим виведенням аналітичних співвідношень, порівнянням результатів з відомими та

збіжністю результатів математичного моделювання з результатами, що отримані під час впровадження розроблених програмних засобів.

Особистий внесок здобувача. Результати даної магістерської кваліфікаційної роботи отримані самостійно. В публікації у співавторстві здобувачу належить дослідження перспектив інформаційної технології прогнозування обсягу поставки квітів [1].

Апробація результатів роботи. Результати досліджень було апробовано на «LI Науково-технічна конференція факультету будівництва, цивільної та екологічної інженерії (2022)»[1].

Публікації. За основними результатами досліджень опубліковано тези доповіді на конференції [1].

1 ОБҐРУНТУВАННЯ ДОЦІЛЬНОСТІ РОЗРОБКИ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПРОГНОЗУВАННЯ ОБСЯГУ ЗАКУПІВЛІ КВІТІВ

1.1 Аналіз діяльності ринку продажів флористичної продукції

Ринок квітів – один із небагатьох ринків в Україні, який характеризується відсутністю великих власників та мережових продажів. Незважаючи на зростання продажів, ринок квітів представлений стихійною торгівлею, відсутністю сервісу та сучасних технологій. Таким чином, вітчизняні компанії займають близько 30% внутрішнього ринку, а все інше – імпортне.

Класично ринок квіткової продукції можна розділити на три основні сегменти: ринок зрізаних квітів; горщикової продукції та посадкового матеріалу. Наразі немає відкритої статистики чи іншої кількісної інформації щодо оцінки ринку зрізаних квітів в Україні. За оцінками експертів, більше половини всіх зрізаних квітів, які продаються в Україні, є імпортними. Менша частка – квіти вітчизняного виробництва, їх вирощують як спеціалізовані підприємства (переважно троянди), так і непрофесійно – у господарствах на невеликих ділянках, які потім продаються переважно в невеликих кіосках і на натуральних базарах. У сегменті зрізаних квітів в Україні переважає виробництво троянд, гербер, гвоздик, цибулин. Найпопулярніші квіти на квітковому ринку в горщиках - фікус, орхідея, драцена, спатифіллум, азалія, бегонія. Найбільше їх імпортують з Нідерландів (90%), Польщі, Данії та Італії. Загалом, аналіз квіткового ринку України показує, що його обсяг значною мірою залежить від фінансового благополуччя населення нашої країни. Так було під час кризи 2014-2015 років, ринок квітів впав на 58%, а в наступні два роки, після початку економічного зростання, збільшився на 40% [2]. Сьогодні спостерігається тенденція до зниження ринку квітів, оскільки війна в Україні може знизити популярність ринку квітів, особливо їх імпорту. Тому очікується розвиток вітчизняного виробництва популярних квітів, особливо троянд. У цьому секторі з'являться нові учасники, щоб конкурувати з провідними галузями сучасності, такими як

Асканія-Флора, Камелія, Фрезія та Тандем. Троянди є основою продажу (до 70%) асортименту зрізаних квітів протягом року: це троянди, які дарують на будь-якому заході, тому вони широко використовуються у флористиці. Найпопулярнішою є червона троянда (майже половина всіх троянд), за нею йде біла троянда (25%), троянди інших кольорів часто складають не більше 25% асортименту [3]. Наявність інших квітів в асортименті залежить від пори року. Український ринок досить консервативний, тому асортимент складається переважно з класичних категорій, новинки дуже рідкісні. Структура продажу свіжих зрізаних квітів сильно варіюється в залежності від різних свят.

Сьогодні більшість українців дарує квіти на свята. Найбільші доходи продавці квітів приносять 8 березня та 14 лютого, на які припадає 13,5% і 9,4% річного обсягу продажів відповідно. Ще 5,6% додає 1 вересня традиція ходити до школи з букетом [4]. Структура продажів зрізаних квітів у святкові дні в Україні продемонстрована на рисунку 1.1.

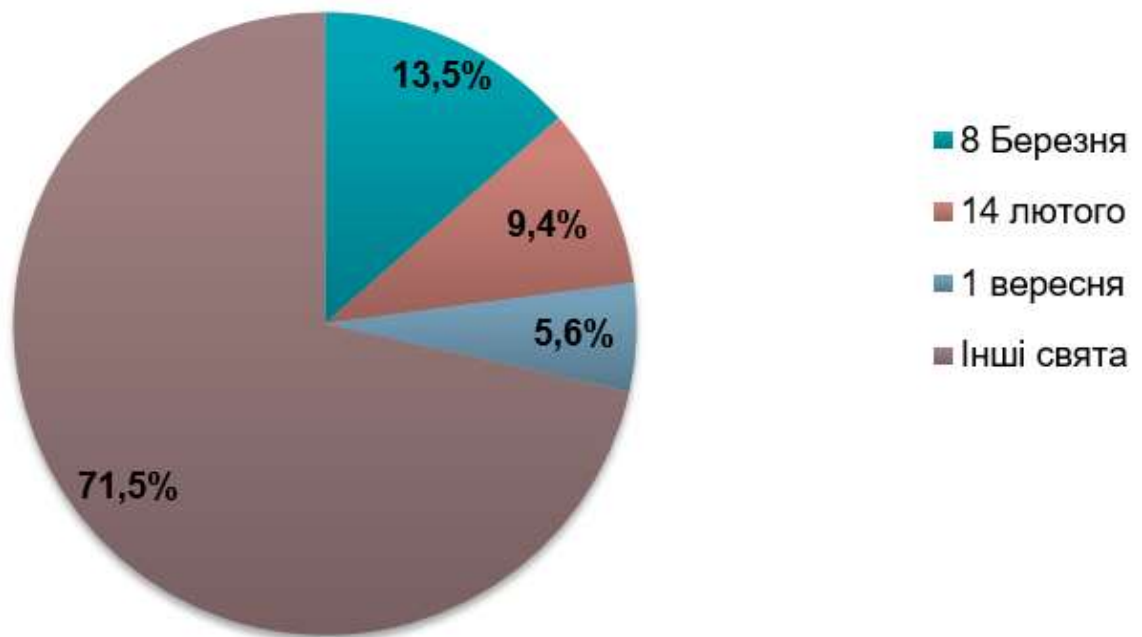


Рисунок 1.1 – Діаграма об'ємів продажу зрізаних квітів у святкові дні

Аналізуючи ринок квітів, слід зазначити, що більшість квітів імпортного походження. Вирощують троянди переважно в українських теплицях. Друге місце в структурі займають тюльпани, але їх частка не перевищує 2%.

Для цього ринку характерна сезонність – кількість імпортованих квітів більше взимку та навесні, а влітку та восени збільшується кількість квітів від вітчизняних виробників.

Аналіз ринку квітів в Україні показує, що найбільш різноманітний він у столиці. Регіональні оптові бази поповнюються квітами приблизно раз на тиждень, і чим менше місто, тим менше точок і асортимент квітів.

Використовуючи аналіз ринку квітів в Україні, можна виділити наступні основні канали збуту [5]:

- продовольчі ринки та невеликі кіоски – на них припадає 60-70% продажів квітів через відносно низькі ціни, свіжість продукції та зручне розташування на контрольно-пропускних пунктах;
- фірмові флористи – підтримують частку продажів на рівні 20-25%; намагаються сформувати конкурентоспроможне співвідношення ціна-якість шляхом прямого контакту з виробниками та додатковими послугами;
- супермаркети та гіпермаркети, бакалія та сегменти DIY – становлять близько 5% від загального обсягу продажів; перевага віддається вже готовим букетам, які покупці купують на додаток до основного подарунка, придбаного тут;
- інтернет-магазини – працюють як окремо, так і як служби доставки для офлайн-магазинів; дозволяють подарувати квіти кур'єром на відстані.

1.2 Дослідження принципів прогнозування та планування обсягу закупівлі квітів для квіткових магазинів

Прогноз продажу – один з найважливіших етапів ведення бізнесу: підприємству повинно бути представлено повідомлення про те, що воно продає на якій суммі з якою рентабельністю. Методи прогнозування продажів здійснюють різні, починаючи від елементарних, закінчуючи тими, які складаються за допомогою складних математичних інструментів [6].

Прогноз продажу є первинним, а план продажу – вторинним, залежним від результатів прогнозування. В кількісному відношенні прогнозів може бути декілька, а планів – один.

Прогноз завжди має під собою визначену базу, він ніколи не робить із передбачуваних, пов'язаних, наприклад, із бажанням підприємця отримати вигоду в якій-небудь період [7].

Щодо форм прогнозування, то слід зазначити, що існують два різновиди:

- прогноз ринку в цілому, що визначає прогноз збуту окремих товарів, що виробляються всіма підприємствами. Далі розраховується частка ринку компанії і, таким чином, розраховується прогнозована вартість реалізації її товарів;

- пряме прогнозування, при якому підприємство безпосередньо визначає вартість прогнозу, виходячи з інформації про власний продаж своєї продукції. У той же час він не прогнозує ринок в цілому. У більшості випадків маркетологи пропонують таку форму прогнозування, хоча цей підхід менш ефективний. Він недостатньо враховує загальний розвиток збуту на ринку.

Достовірність прогнозів продажів певною мірою залежить від використовуваних принципів. Відомо, що прогнозування збуту – відповідальне і складне завдання. Застосування відповідних принципів може забезпечити належну точність відповідних результатів, отриманих у процесі прогнозування.

Основні принципи прогнозування продажів включають [8]:

- об'єктивність прогнозів. Це забезпечує відповідність стану навколишнього середовища, на якому базується прогноз, та його результатів. Реалізація цього принципу прогнозування передбачає:

а) використання достовірної інформації за минулий період;

б) використання в очікуванні найбільш ефективних методів за цих умов.

Нижче буде зазначено, що існують різні методи прогнозування. У кожній конкретній ситуації необхідно вибрати ту, яка враховує характер і зміст середовища, тип прогнозу тощо.

в) одночасне використання кількох методів прогнозування. Дуже часто в бізнесі паралельно використовуються такі методи, як регресійний аналіз і метод

експоненційного згладжування. За допомогою регресійного аналізу досягається більш висока точність прогнозу (коли при прогнозуванні необхідно враховувати різні фактори). Аналогічно передбачення поєднуються з математичними методами, зокрема з такими, як експоненціальне згладжування [9]:

- мінімізація кількості ринкових факторів, що впливають на результати прогнозу. Чим можна пояснити необхідність використання цього принципу? Значна кількість факторів ускладнює процес прогнозування. Перед прогнозістом постає проблема визначення їх реального впливу на величину прогнозу. По-друге, чим більше факторів, тим вище ймовірність дублювання їх впливу на результати прогнозу. І по-третє, модель регресійного аналізу, яка спирається на значну кількість факторів, буде досить складною з точки зору розуміння їх впливу на продажі нерухомості;

- використання різних оцінок результатів прогнозу (оптимістичних, найбільш ймовірних і песимістичних). Це пов'язано з можливістю розвитку різних ринкових ситуацій (сприятливі умови, найгірші ситуації, проміжні прогнози між цими двома крайнощами);

- відповідність прогнозу цілям використання його результатів. Нагадаємо, що з цією метою є обґрунтування поставлених перед збутом товарів цілей. Тому ця кореспонденція в першу чергу стосується термінів, на які робляться прогнози. Вони повинні відповідати умовам, для яких складаються плани збуту. Причому ця відповідність стосується факторів, які впливають як на вартість прогнозу, так і на значення прогнозних цілей продажу товарів. У цьому випадку виходить більша кореляція між прогнозами та планами;

- з урахуванням максимальних можливостей прогнозу. В даному випадку мова йде про точність прогнозів продажів різних товарів. Наприклад, здатність прогнозувати продажі вже існуючих товарів у цьому плані набагато більше, ніж, скажімо, нових товарів. На прогноз продажів останнього можуть вплинути фактори, повний перелік яких важко передбачити. Те саме стосується сезонних продуктів, продуктів, призначених для продажу на зовнішніх ринках. Тому

менеджери, розробляючи прогнози збуту цих товарів, повинні враховувати обмеження можливості прогнозування їх збуту.

Прогнозування обсягу закупівлі та продажу товару, що використовується компаніями, зазвичай різноманітне. Воно зумовлене великою кількістю планів, розроблених менеджерами з маркетингу [10]. Класифікація прогнозування представлена в таблиці 1.1.

Таблиця 1.1 – Класифікація та види прогнозування

№ з/п	Ознаки класифікації прогнозів	Види прогнозів
1	Терміни, на які складаються прогнози	1. Короткострокові 2. Середньострокові 3. Довгострокові 4. Прогнози, що відповідають тривалості сезону
2	Об'єкти прогнозування	1. Прогнозу для всього асортименту 2. Прогноз для окремих товарів
3	Характер ринку	1. Весь ринок 2. Окремі сегменти 3. Внутрішній ринок 4. Зовнішній ринок 5. Торгові зони
4	Характер процесу прогнозування	1. Пошукові 2. Інтуїтивні 3. Нормативні
5	Точність кількісної оцінки результатів	1. З однозначною оцінкою 2. З інтервальними оцінками
6	Ступінь імовірності настання результатів	1. Імовірнісні 2. Інваріативні

З точки зору прогнозування обсягу закупівлі квітів, необхідно звернути увагу на короткострокові прогнози, адже закупка квітів відбувається в основному на період часу від 4 до 10 днів.

1.3 Існуючі моделі та методи, що застосовуються для реалізації поставленої задачі

Для прогнозування обсягу закупівлі квітів доцільно використовувати кількісні методи прогнозування. Розглянемо деякі з них детально.

Кількісні методи, на відміну від якісних методів прогнозування збуту, спираються на точний розрахунок обсягу продажів, який спирається на використання набору статистичних даних. Такі методи зазвичай використовують аналіз часових рядів, що передбачає врахування обсягу закупівлі минулого року в такий же період наступного року.

Їх застосовують у випадках, коли закономірні тенденції розвитку збуту, характерні для попереднього періоду, залишаються незмінними для прогнозного періоду. Іншими словами, доцільно використовувати методи кількісного прогнозування при стабільності економічної ситуації як у попередні роки, так і в прогнозний період. Це дозволяє екстраполювати тенденцію продажів, яка склалася за минулий період, на прогнозні роки.

Серед найбільш поширених методів кількісного прогнозування виділяють:

- прогноз за ковзними середніми;
- експоненційне згладжування;
- прогнозування найменших квадратів;
- графічний метод;
- метод Бокса-Дженкінса.

Метод прогнозування з використанням ковзних середніх є досить поширеним завдяки своїй простоті. За допомогою цього методу прогнозована вартість продажів у майбутньому буде дорівнювати його середньому обсягу минулих років. Такий підхід пояснюється тим, що фактори, які діяли як у прогнозному періоді, так і в майбутньому, незмінні [11].

Однак з часом вони можуть змінюватися. Тому необхідно взяти дані з кількох минулих періодів, усереднювати їх і визначити прогноз продажів на

майбутнє за національною ознакою. Чим більше років минулого періоду буде включено в розрахунок, тим точнішим буде прогноз продажів.

Середня поточна вартість i -го року визначається як частка від ділення суми продажів за останні роки на кількість років, за які продажі враховуються при розрахунку середніх оборотних значень.

Такий підхід до прогнозування збуту доцільний за умови, що його обсяг з роками не змінюється. У випадках, коли спостерігається тенденція до зростання обсягів продажів за рік, навіть незначне, прогноз збуту може бути нижчим за його фактичне значення в окремі роки минулого періоду. Це нелогічно. Більше того, в умовах стабільної економіки закономірно динамічно збільшувати продажі. Щоб уникнути такої ситуації, необхідно змінити підхід до використання ковзних середніх у прогнозі. Прогнозна вартість продажів буде розрахована в такому порядку:

- збирається статистична інформація про фактичні обсяги реалізації минулих років;
- визначаються ковзні середні (бажано три);
- значення цих величин систематично попарно порівнюють і встановлюють їх зростання;
- обчислює середнє збільшення цих значень за досліджуваний період як відношення їх суми до величини;
- шляхом додавання цього збільшення до фактичного обсягу реалізації останнього року досліджуваного періоду та визначає значення прогнозу на наступний рік.

У випадках, коли статистика минулих років «змішана», тобто одні з них більш змістовні та точні, а інші – навпаки, використовується метод прогнозування, який називають експоненційним згладжуванням. Цей метод незалежно відкрили Браун (1958) і Холт (1957) [11]. Цей тип прогнозу використовує значення числового ряду попередніх років. Його суть полягає в застосуванні константи згладжування. У цьому випадку розрахункова вартість продажів розраховується за такою формулою [12]:

$$ОП_{t+1} = ax_t + (1 - a)ОП_t, \quad (1.1)$$

де $ОП_{t+1}$ – прогнозована величина продажу часового періоду $t + 1$; a – константа згладжування; x_t – фактичний обсяг продажу періоду t ; $ОП_t$ – прогнозований обсяг продажу періоду t .

Вибір конкретного значення константи згладжування, який виконується довільно, впливає на значення прогнозу продажів. Якщо він великий, це означає, що фактичний продаж минулого періоду стає більше. Константа меншого значення збільшує вплив на прогнози продажів на більш віддалені періоди. Тому важливу роль у прогнозуванні збуту тут грає правильне визначення значення константи згладжування [13].

На практиці фахівець, який займається прогнозом, приймає свою цінність виходячи з:

- вивчення інформації, що використовується в прогнозі;
- його інтуїція, що сформувалася з часом;
- врахування подібності умов, що мали місце в минулому і прогнозованих періодах;
- умови прогнозу.

Щоб знайти константи згладжування, пропонується використовувати квазіньютонівський алгоритму для максимізації правдоподібності (ймовірності).

Експоненціальне згладжування як метод прогнозування продажів має як переваги, так і недоліки.

Переваги:

- простота використання;
- здатність прогнозиста визначати вплив на точність прогнозу на певний період.

Проте у нього є деякі недоліки. До них належать:

- нездатність передбачити зростання або спад ринку;
- довільний вибір значення константи згладжування, яке прогнозист визначає самостійно.

Незважаючи на ці недоліки, ряд компаній досягли успіху використовуючи експоненціальне згладжування в практиці прогнозування продажів.

Теорії і практиці відомий також такий метод прогнозування збуту, як метод найменших квадратів. Він відноситься до методів екстраполяції [14].

Його суть полягає в мінімізації суми квадратів відхилень між спостережуваними значеннями і розрахунковими значеннями. Прогноз продажів буде точнішим, коли відомі між цими параметрами менші. Прогноз продажів для цього методу визначається за формулою:

$$y_{t+1} = a + bx_n, \quad (1.2)$$

де y_{t+1} – прогнозований обсяг продажу; $t + 1$ – період, на який здійснювався прогноз; a і b – коефіцієнти; x_n – умовне позначення часу (номер періоду, на який здійснювався прогноз);

$$a = \frac{\sum_{i=1}^n x_i^2 \sum_{i=1}^n y_i - \sum_{i=1}^n x_i \sum_{i=1}^n x_i y_i}{n \cdot \sum_{i=1}^n x_i - (\sum_{i=1}^n x_i)^2} \quad (1.3)$$

$$b = \frac{n \cdot \sum_{i=1}^n x_i y_i - \sum_{i=1}^n x_i \sum_{i=1}^n y_i}{n \cdot \sum_{i=1}^n x_i^2 - (\sum_{i=1}^n x_i)^2} \quad (1.4)$$

де n – кількість років, за які взята статистична інформація для прогнозування; x_i – номер i -го року; y_i – обсяг продажу в цьому році.

Можна також використовувати графічний метод з побудовою діаграми для прогнозування продажів. Це досить просто. Суть цього методу полягає в наступному [15]:

- збирається інформація про фактичні обсяги реалізації за попередні роки;
- будується система координат. Періоди часу відкладаються на осі x , а обсяги продажів – на осі y ;
- перетин перпендикулярів, опущених уздовж двох осей, фіксується точкою;

- через ці точки проходить пряма, рівновіддалена від усіх цих точок;
- від точки, розташованої на осі абсцис і відповідної прогнозованому році, перпендикуляр до перетину з проведеною лінією, яка фіксується точкою;
- з цієї точки падає перпендикулярно до осі y . Точка перетину і буде визначати вартість прогнозованих продажів.

Точність прогнозу збуту за допомогою графічного методу залежить від умінь прогнозиста правильно провести (рівновіддалену від усіх точок) пряму лінію.

Наступним розглянемо метод Бокса-Дженкінса. Він поєднує моделі авторегресії та ковзного середнього, щоб утворити змішану модель ARIMA, яка є основною моделлю серед інших моделей цього класу. Модель авторегресії – це модель часового ряду [16].

У ньому значення часового ряду в даний момент часу лінійно залежить від значень того самого ряду, який їм передує. Часовий ряд – це набір даних певного показника, в даному випадку обсягів продажів, за відповідну кількість послідовних періодів. Останні можуть описувати різні часові ряди - як стаціонарні, так і нестаціонарні.

Стаціонарні часові ряди характеризуються постійним середнім значенням, а їх дані коливаються навколо нього з відносно постійною дисперсією. Модель ARIMA відноситься до лінійних моделей класу.

Прогнозування продажів за методом ARIMA не передбачає чіткої моделі. Тут визначається лише їхній загальний клас. А також наявний опис часового ряду. Це дозволяє визначити поточний стан змінної. Після цього алгоритм самостійно вибирає необхідні моделі прогнозування.

Для прогнозування обсягу закупівлі квітів існує потреба у врахуванні тренду та сезонності, однак є квіти, що продаються протягом всього року, а є ті, що можна вважати несезонними товарами. Для цього необхідно поєднати метод ковзного середнього та експонентного згладжування для сезонних та несезонних квітів відповідно.

1.4 Аналіз відомих програмних рішень для прогнозування обсягу закупівлі квітів

Кількість запасів є одним з головних критеріїв, що гарантують стійке функціонування компанії. Крім того, відсутність та надлишок товарних запасів на складі спричиняє підприємство зазнати збитків. Щоб цього не сталося, необхідно правильно організувати систему закупівель, що неможливо без своєчасного планування.

Досвід показує, що найбільш прогресивним варіантом планування закупівель є застосування спеціалізованих програм, наприклад, Forecast NOW! [17]. Розглянемо дану програму детальніше.

На початковому етапі створюється система обліку та контролю руху товару. Для цього група працівників робить повну інвентаризацію товарів на складі. Крім того, при вивченні рахунків-фактур та інших документів враховується надходження та споживання товару на кожну позицію, а також його переміщення всередині компанії. За результатами інвентаризації створюється набір документів, який регулює прогноз закупівель відповідно до потреб бізнесу з урахуванням змін попиту внаслідок сезонних коливань та зміни умов ринку.

Крім того, дані аналізуються для визначення правильної стратегії закупівель. Експерти вважають двовимірну систему аналізу ABC найбільш ефективною. Це дозволяє розділити товар на групи за двома критеріями, наприклад, за рентабельністю та періодичністю продажів. Найвигідніша і найбільш продавана продукція – у групі AA, а найменш прибуткова – при періодичному попиті – у групі CC (рис. 1.2).

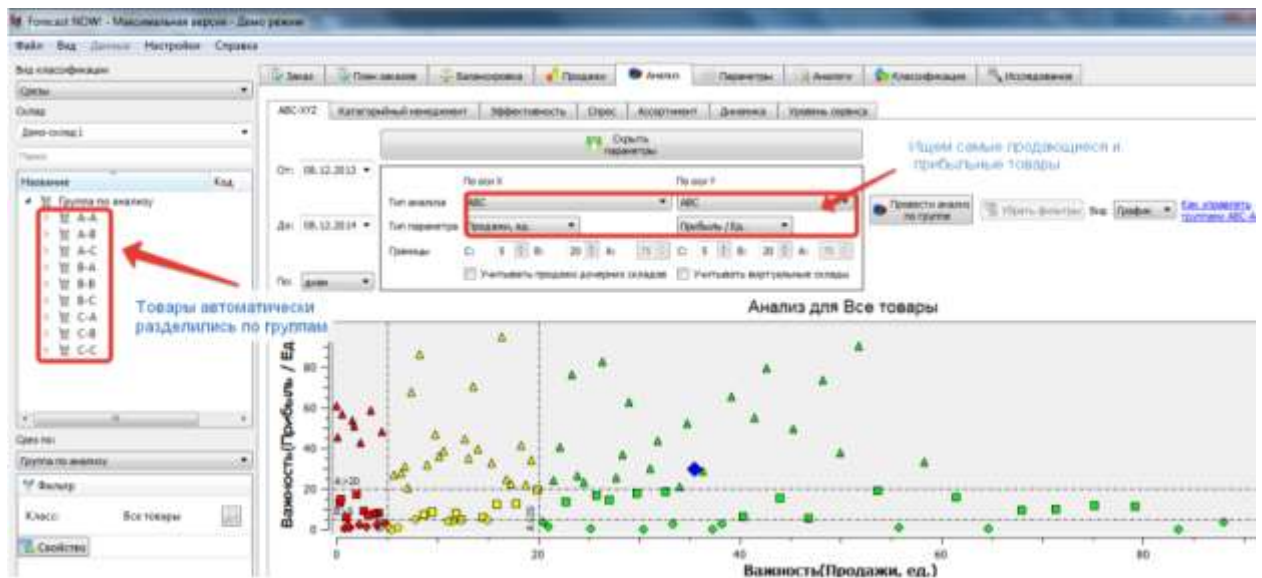


Рисунок 1.2 – Приклад роботи програми Forecast NOW!

Даний програмний продукт дозволяє автоматично формувати план замовлення та отримання, розраховувати прогноз попиту та націлювати оптимальне замовлення для кожного основного товару для кожної точки продажу окремо, для територіальної групи складів або центрального складу. Розрахунок проводиться для кожної товарної позиції, товарної групи із заданим ступенем агрегації.

Інтелектуальна система управління запасами пропонує такі переваги:

- звільнити від 15% до 30-40% оборотних коштів за рахунок зменшення надлишкових запасів та неліквідних активів – дозволити бізнесу розвиватися швидше та ефективніше;
- зменшити дефіцит на 5-30% і збільшити продажі;
- значно зменшить залишки запасів – звільнить приміщення для нового асортименту;
- забезпечити точний прогноз попиту – зменшити страховий запас;
- розширить асортимент доступної продукції без додаткових вкладень коштів;
- дозволить вам позбутися зайвих транспортних витрат завдяки більш точному плануванню необхідних поставок та оптимальному переобладнанню транспорту.

Проте, як і більшість подібних програм, даний продукт не є безкоштовним, А також містить функціонал, що для прогнозування обсягу закупівлі квітів є зайвим, що означає неефективний вклад коштів.

Продукт складського і товарного обліку «ЕКАМ» призначений для планування продаж, управління запасами, створення замовлення постачальникам, відстеження кількості продукції, що надійшла на склад [18]. При досягненні заданого мінімального залишку заявки на поставку формуються програмою автоматично.

Основна мета використання «ЕКАМ» – максимально точно спрогнозувати купівельний попит на найближчий період. Правильні підрахунки дозволять забезпечити магазин достатньою кількістю товарних запасів і уникнути непотрібного надлишку.

Перевагою сервісу є те, що він не обмежений тільки цим функціоналом і дає підприємцю безліч додаткових можливостей:

- аналіз даних про продажі в онлайн-режимі;
- призначення знижок і цін;
- створення персоналізованих програм лояльності;
- контроль над рухом грошових коштів;
- підключення онлайн-каси по 54-ФЗ;
- оптимізація роботи персоналу за рахунок докладної звітності за результатами продажів, руху та списання товарів, а також автоматизації документообігу і багато іншого.

Приклад роботи програми представлено на рисунку 1.3.

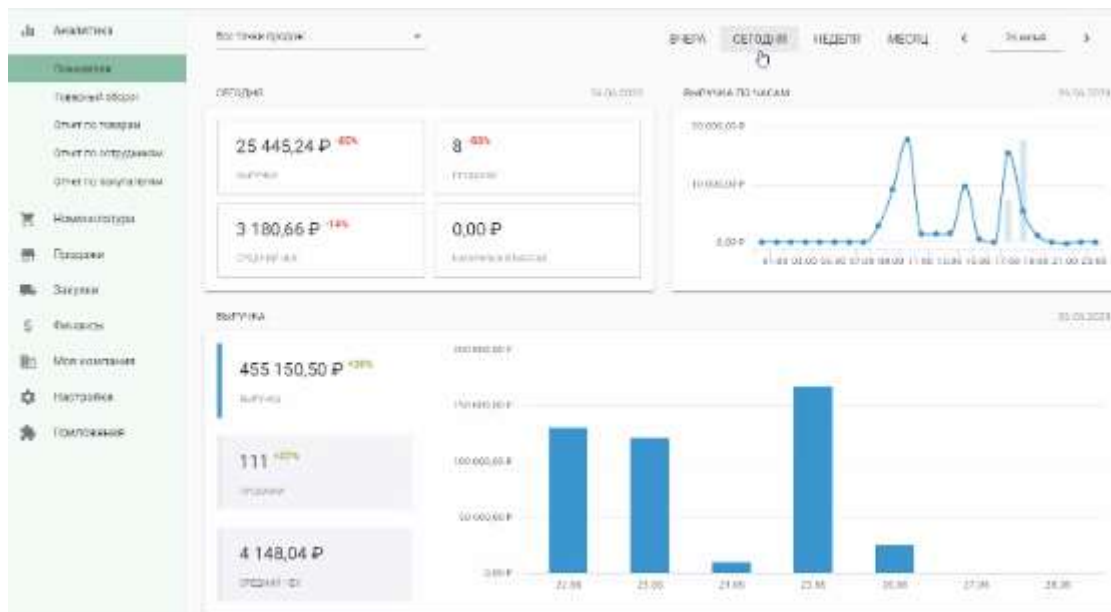


Рисунок 1.3 – Загальний вигляд інтерфейсного вікна програми «ЕКАМ»

Е1-Закупівлі – система, що допомагає швидко створювати, агрегувати, редагувати, узгодити заявки і плани закупівель. Програма допомагає простежити життєвий цикл конкретної позиції від етапу заявки функціонального підрозділу до виконання по специфікації укладеного договору. Така архітектура допомагає провести інтеграцію з корпоративним або загальноросійським довідником матеріалів [19].

Система дозволяє повністю автоматизувати закупівельний процес організації: планування та проведення закупівель, робота з договорами.

Система має ряд функцій, що можна розглядати як переваги для прогнозування обсягу закупівлі квітів:

- створення нової заявки на закупівлю на основі існуючої;
- зв'язка позицій заявки на закупівлю з рядками бюджетних лімітів;
- об'єднання заявок на закупівлю в лоти натисненням однієї кнопки;
- формування електронного плану в системі;
- формування друкованої форми плану за заданим шаблоном.

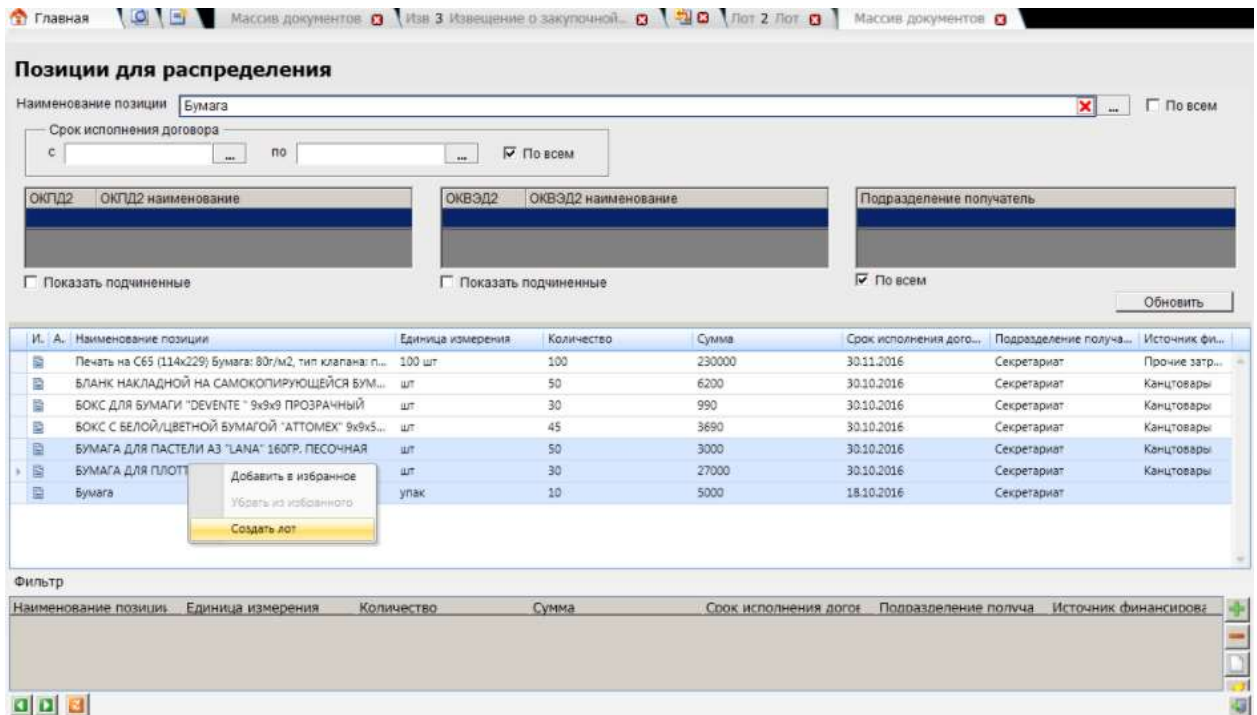


Рисунок 1.4 – Загальний вигляд роботи програми Е1-Закупівлі

Однак багато функцій є зайвими для вирішення проблеми прогнозування закупівлі квітів, що може перевантажити магазин та є недоцільним вкладенням коштів. Також вагомим недоліком для магазину квітів є те, що програма більше орієнтована не на прогнозування обсягу закупівель, а на їх автоматизацію.

«Класс365» – додаток, що дозволяє автоматизувати продажі, вести торговий та складський облік, управляти клієнтською базою, справами та співробітниками [20]. Як і розглянуті вище програми, дана система має багато функціональних можливостей, зайвих для вирішення проблеми прогнозування обсягу закупівлі квітів. Даний комплекс орієнтований на великі підприємства, а для маленького магазину квітів не обґрунтовано використовувати весь його функціонал. Однак він містить короткострокове та довгострокове планування, побудову індивідуальних планів та точне прогнозування обсягу продажу на вибрані періоди, що є підставою для розгляду даної системи в дослідженні.

Серед проаналізованих програмних систем, «ЕКАМ», «Е1-Закупівлі» та «Класс365» являються багатофункціональними та дорогими програмами, що мають вагомі переваги для підприємств у сфері продажу, однак непотрібні для

прогнозування обсягу закупівлі квітів. Дані програми містять інструмент для вирішення нашої задачі, проте їх недоцільно використовувати, зважаючи на ціну.

Анализ продаж 01.12.2013 — 31.01.2014 Сформировать Выгрузить в Excel

Анализ по документам: Группировка:
 Заказы покупателей По сотрудникам

Фильтры не установлен

Контрагент	Заказ покупателя	Сумма	Сумма оплаты	Дата оплаты	Отгружено	Прибыль	Ответственный
▼ Татьяна Леснова 5 документов на сумму 150 000 руб.							
Солнечная поляна	Заказ покупателя № 121 от 10.12.2013	25 000,00	25 000,00 (100%)	20.12.2013	25 000,00 (100%)	5 000,00	Татьяна Леснова
Солнечная поляна	Заказ покупателя № 122 от 10.12.2013	55 000,00	50 000,00 (90,1%)	24.12.2013	50 000,00 (90,1%)	3 000,00	Татьяна Леснова
Интер-Компьютеры	Заказ покупателя № 152 от 10.12.2013	55 000,00	50 000,00 (90,1%)	24.12.2013	50 000,00 (90,1%)	3 000,00	Татьяна Леснова
Грицанкова Валентина	Заказ покупателя № 152 от 10.12.2013	55 000,00	25 000,00 (100%)	24.12.2013	25 000,00 (100%)	3 000,00	Татьяна Леснова
Грицанкова Валентина	Заказ покупателя № 152 от 10.12.2013	55 000,00	25 000,00 (100%)	24.12.2013	25 000,00 (100%)	3 000,00	Татьяна Леснова
▼ Василиса Орлова 3 документа на сумму 550 000 руб.							
Интер-Компьютеры	Заказ покупателя № 152 от 10.12.2013	55 000,00	50 000,00 (90,1%)	24.12.2013	50 000,00 (90,1%)	3 000,00	Василиса Орлова

Рисунок 1.5 – Загальний вигляд інтерфейсного вікна програми «Класс365»

Наближеним по функціональних характеристиках є програма «Forecast NOW!», але вона теж має недолік у вигляді ціни, а також недостатню точність прогнозування для такого товару як зрізані квіти, адже навіть незначні похибки можуть призвести до великих втрат. Саме тому в подальшому обираємо як прототип цю систему, однак вдосконалимо точність прогнозування для запобігання неприємних ситуацій у вигляді неправильного прогнозу та, як наслідок, втрати коштів.

1.5 Постановка задач дослідження

Магістерська кваліфікаційна робота (МКР) присвячена реалізації інформаційної технології прогнозування обсягу закупівлі квітів. Отже, необхідно програмно вирішити задачу прогнозування для магазину квітів. Інформаційна технологія призначена для працівників та власників квіткових магазинів, складів або мереж квіткових магазинів з метою забезпечення збільшення ефективності роботи з таких магазинів, а також збільшення

прибутків, що досягнеться точністю прогнозування обсягів поставок квітів, зменшивши витрати та виключивши можливість псування великої кількості залишків, що були нереалізовані через відсутність попиту на них. Таким чином, інформаційна технологія буде виконувати прогнозування обсягу закупівлі квітів на основі аналізу часових рядів, використовуючи попередню статистику продажу квітів.

Дана інформаційна технологія повинна містити зручну та зрозумілу реалізацію для кінцевого користувача, не потребувати від працівників квітового магазину спеціальних навичок та знань у галузі алгоритмів прогнозування. Задання вхідних даних має здійснюватися автоматично, імпорт даних про попередні продажі та закупівлі здійснюється шляхом запитів до бази даних, а вихідні дані мають характеризуватися чіткістю подання. Користувачеві створеної інформаційної технології важливо максимально можлива точність прогнозування обсягу закупівлі квітів за вказаний період.

Задача цієї роботи полягає у розробці інформаційної технології прогнозування обсягу закупівлі квітів.

Вимоги до клієнту:

- головна сторінка з меню;
- перегляд архіву у вигляді таблиці;
- можливість вибору терміну;
- можливість додавання нових даних;
- завершення сесії (вихід).

Вимоги до менеджера:

- можливість редагування інформації;
- можливість додавання нової інформації (квітів);
- можливість видалення інформації;

Вимоги до серверу:

- зберігання даних користувача через зв'язок з базою даних;
- видача необхідних даних по запиту від клієнта.

Вхідними даними роботи є період прогнозування.

Вихідними даними є прогнозована кількість квітів, необхідна для закупівлі на вибраний період.

Отже, потрібно розробити інформаційну технологію, в якій буде реалізовано алгоритм прогнозування обсягу закупівлі квітів.

1.6 Висновок до розділу 1

У даному розділі було досліджено особливості ринку квітів в Україні. Також розглянуто принципи прогнозування та планування обсягу закупівлі квітів для квіткових магазинів. Було проведено аналіз сучасних програм-аналогів, які використовуються для прогнозування закупівлі і продажу товарів. Було наведено короткий опис основних функцій, які виконують дані програми. Також було досліджено методи, що можуть бути використані для прогнозування обсягів продаж, для чого запропоновано використати методи аналізу часових рядів – метод ковзного середнього та експонентного згладжування. Здійснено постановку задачі.

2 МОДЕЛЮВАННЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПРОГНОЗУВАННЯ ОБСЯГУ ЗАКУПІВЛІ КВІТІВ

2.1 Обґрунтування вибору методу прогнозування обсягу закупівлі квітів

Прогнозування на основі часових рядів є одним із найпопулярніших підходів до прогнозування розвитку економічних процесів, обсягів торгових операцій, обсягів виробництва та накопичення продукції на складах, оцінки альтернативних економічних стратегій, формування бюджетів підприємства та держави, прогнозування та управління економічними та фінансовими ризиками та інші.

Аналіз часового ряду починається з побудови та вивчення його графіка. Якщо нестационарність часового ряду очевидна, то перш за все необхідно виділити нестационарну складову ряду. Процес виявлення тренду та інших складових ряду, які призводять до порушення стаціонарності, може проходити в кілька етапів. На кожному з них розглядається ряд залишків, отриманий в результаті розрахунку з вихідного ряду обраної моделі тренду, або результат диференціальних та інших перетворень ряду. Крім графіків, ознаками нестационарності часового ряду можуть бути автокореляційна функція, яка не прямує до нуля (за винятком дуже великих значень лагів) і наявність на періодограмі яскраво виражених піків на низьких частотах. За допомогою функції автокореляції також досліджуються внутрішні зв'язки між елементами часового ряду [21].

У вибіркових дослідженнях найпростіші числові характеристики описової статистики (середнє, медіана, дисперсія, стандартне відхилення, коефіцієнти асиметрії та ексцесу) зазвичай дають достатньо інформативну картину вибірки. При цьому графічні методи представлення та аналізу вибірок відіграють лише допоміжну роль, дозволяючи краще зрозуміти локалізацію та концентрацію даних, закон їх розподілу. Зовсім інша роль графічних методів в аналізі часових

рядів. Табличне представлення часових рядів і описова статистика в цілому не дозволяють зрозуміти природу процесу, тоді як з графіка часових рядів можна зробити досить багато висновків.

Метод прогнозування з використанням ковзних середніх досить поширений завдяки своїй простоті. При цьому методі очікуваний обсяг закупівлі у майбутньому періоді дорівнюватиме середньому його обсягу попередніх років. Такий підхід пояснюється тим, що фактори, які діяли як у прогнозованому періоді, так і в майбутньому, залишаються незмінними.

Проте з часом вони можуть змінюватися. Тому необхідно взяти дані кількох минулих періодів, усереднити їх і на основі цього визначити прогноз закупівлі квітів на майбутнє. Чим більше років минулого періоду буде включено в розрахунок, тим точніше буде прогноз закупівлі.

Середня ковзна вартість i -го року визначається як частка суми обсягів закупок за останні роки, поділена на кількість років, обсяг закупівлі яких враховується при розрахунку середніх ковзних величин.

Такий підхід до прогнозування закупівлі сезонних квітів є доцільним за умови, що їх обсяги мало змінюються від року до року. У цих випадках, коли спостерігається тенденція зростання обсягу закупівлі сезонних квітів протягом багатьох років, навіть якщо вона незначна, може виявитися, що прогнози закупок будуть нижчими за реальні потреби в окремі минулі періоди. Крім того, в умовах стабільної економіки природно динамічно нарощувати обсяг продажів. Для уникнення такої ситуації, необхідно трохи змінити підхід до використання ковзних середніх в прогнозі. При цьому прогнозне значення закупівлі слід розраховувати в такому порядку:

- збір статистичної інформації про фактичні обсяги закупівлі та продажу минулих років;
- визначення ковзних середніх;
- послідовне порівняння величин цих значень та визначення їх зростання;
- обчислення середнього приросту цих значень за досліджуваний період як відношення їх суми до кількості;

– додавання приросту до фактичного обсягу реалізації за останній рік досліджуваного періоду, визначення значення прогнозу на наступний період.

Отже, метод ковзного середнього доцільно використовувати для прогнозування обсягу закупівлі сезонних квітів. Що стосується несезонних квітів, доречно розглянути метод експоненціального згладжування.

Експоненціальне згладжування – це метод згладжування часових рядів, процедура розрахунку якого включає обробку всіх попередніх спостережень з урахуванням старіння інформації з віддаленням від прогнозного періоду. Іншими словами, чим «старше» спостереження, тим менше воно повинно впливати на значення прогнозної оцінки. Ідея експоненціального згладжування полягає в тому, що в міру старіння даних спостережень їм призначають зменшувальні ваги [22].

Цей метод прогнозування вважається дуже ефективним. Основні переваги методу полягають у можливості врахування ваг вихідної інформації, у простоті розрахункових операцій, у гнучкості опису динаміки різних процесів. Метод експоненціального згладжування дає змогу отримати оцінку параметрів тренду, які характеризують не середній рівень процесу, а тренд, що склався під час останнього спостереження. Найбільшого поширення метод отримав при виконанні середньострокових прогнозів. Для експоненціального методу згладжування основним є вибір параметра згладжування (константи згладжування) і початкових умов.

2.2 Розробка математичної моделі прогнозування обсягу закупівлі квітів

Нехай існують часові ряди обсягу продажу квітів $x_i(t) \in R$ – продажі i -ого товару в магазині за період t , причому значення продажу відомі при $t_0 \leq t \leq t_1$. Також заданий товарний класифікатор, завдяки якому товар можна розбити на групи. Необхідно для всіх товарів спрогнозувати обсяги продажу на наступні після t_1 періоди, тобто знайти значення:

$$y_i = \sum_{t=t_1+1}^{t_1+n} x_i(t). \quad (2.1)$$

Для оцінки якості прогнозів слід використати ковзний контроль, помістивши в навчальну вибірку значення $x_i(t)$ при $t \in [t_0, t_{max}]$, $t_{max} < t_1$. Як функцію якості використаємо:

$$Q_m(Y, \hat{Y}) = \sum_i |y_i - \hat{y}_i|, \quad (2.2)$$

або

$$Q_s(Y, \hat{Y}) = \sum_i |y_i - \hat{y}_i|^2. \quad (2.3)$$

Обрано базовий алгоритм – ковзне середнє по кожному товару за минулі місяці:

$$y_i^0 = \frac{n}{w} s_i(w). \quad (2.4)$$

Де w – ширина вікна (кількість часових проміжків, на основі яких робиться прогноз),

$$s_i(w) = \sum_{t=t_1-w+1}^{t_1} x_i(t). \quad (2.5)$$

Далі за алгоритмом групового обліку:

1. Розбити множину товарів I на підмножини $I_k \subset I$.
2. Повторювати пункти 3-4 для всіх I_k .
3. Знайти сумарні продажі товарів з I_k :

$$S_w = \sum_{i \in I_k} s_i(w). \quad (2.6)$$

4. Визначити долі продажу окремих квітів з групи:

$$D_w(i) = \frac{1}{S_w}. \quad (2.7)$$

5. Обрахувти за методом ковзного середнього прогноз продажу товарів з I_k на наступний період n :

$$S_v(I_k) = \frac{n}{v} \sum_{i \in I_k} s_i(v). \quad (2.8)$$

6. Розподілити спрогнозоване число товарів згідно величини $D_w(i)$:

$$\hat{y}_i = S_v(I_k) D_w(i). \quad (2.9)$$

Вищенаведений алгоритм використовує усереднення два рази - при обчисленні $D_w(i)$ і $S_v(I_k)$. Довжина вікна, яка при цьому використовується, не зобов'язана збігатися - в загальному випадку $v \neq w$. Цьому можна дати просте пояснення: покупці можуть змінювати переваги в межах групи товарів повільніше або, навпаки, швидше, ніж змінюється сумарний попит на товари з групи [23]. Обидві довжини v і w найкраще визначити за допомогою ковзного контролю.

У випадку, коли $x_i(t)$ ціле число, алгоритм потребує модифікації, яку проведемо наступним чином:

1. З $[S_v(I_k)]$ товарів певну частину розподілити детерміновано за формулою

$$y_i^{fix} = [\alpha [S_v(I_k)] D_w(i)], \quad (2.10)$$

де $0 \leq \alpha \leq 1$.

2. Обрахувати кількість товарів, що залишились:

$$S'(I_k) = [S_v(I_k)] - \sum_{i \in I_k} y_i^{fix} \quad (2.11)$$

3. Перерахувати розподіл товарів:

$$D'(i) = \frac{[S_v(I_k)]D_w(i) - y_i^{fix}}{S'(I_k)}. \quad (2.12)$$

4. Товари, що залишились, розподілити згідно ймовірностей $D'(i)$.

Далі необхідно провести оцінку якості. Як зазначено раніше, для оцінки якості пропонується використовувати суму модулів або квадратів відхилень прогнозу від дійсного продажу за формулами (2.2), (2.3).

Прогнозування обсягу закупівлі несезонних квітів відбувається завдяки методу експотенціального згладжування. У даному випадку доцільно використати потрійне експотенційне згладжування (метод Холта-Вінтерса).

Рівняння для мультиплікативного методу Холта-Вінтерса має наступний вигляд:

$$\hat{y}_{t+h|t} = (l_t + hb_t)S_{t+h-m(k+1)}; \quad (2.13)$$

$$l_t = \alpha \frac{y_t}{s_{t-m}} + (1 - \alpha)(l_{t-1} - b_{t-1}); \quad (2.14)$$

$$b_t = \beta^*(l_t - l_{t-1}) + (1 - \beta^*)b_{t-1}; \quad (2.15)$$

$$s_t = \gamma \frac{y_t}{(l_t - b_{t-1})} + (1 - \gamma)s_{t-m}, \quad (2.16)$$

де α – параметр згладжування даних, $0 < \alpha < 1$, β – параметр згладжування тренду, $0 < \beta < 1$, та γ – параметр згладжування сезонності, $0 < \gamma < 1$.

Метод експотенціального згладжування добре працює в ситуаціях, коли змінна, що цікавить, поводить ся стаціонарно, а її відхилення від постійного

значення спричинені випадковими факторами та не є регулярними за своєю природою [24]. Але: незалежно від значення параметра α , метод експоненціального згладжування не зможе передбачити дані, що монотонно зростають або зменшуються (прогнозовані значення завжди будуть нижчими або вищими, ніж спостережувані відповідно). Також можна показати, що моделі з сезонними змінами не зможуть отримати задовільні прогнози за допомогою цього методу.

2.3 Проектування структури інформаційної технології прогнозування обсягу закупівлі квітів

Прогнозування обсягу закупівлі квітів потребує великої обчислювальної потужності. Вхідними даними інформаційної технології є період часу, на який необхідно зробити прогноз, враховуючи архівні дані замовлень закупівлі квітів. Оскільки імпорт вхідних даних виконується через базу даних, користувачеві потрібно лише визначити термін прогнозування. Після отримання вхідних даних – періоду прогнозування – відбувається встановлення зв'язку з базою даних, де міститься інформація про обсяги закупівлі попередніх місяців та відповідного періоду часу минулих років. Завдяки цій інформації формується обсяг закупівлі квітів та виводиться прогнозований результат. Інформаційна технологія здатна автоматично опрацьовувати всю доступну вхідну інформацію, введену користувачем та наявну в базі даних, а також вивести результат у доступному форматі, забезпечивши зручність та зрозумілість подання інформації. Структура інформаційної технології прогнозування обсягу закупівлі квітів зображено на рис. 2.1.



Рисунок 2.1 – Структурна схема інформаційної технології прогнозування обсягу закупівлі

В інформаційній технології будуть представлені наступні модулі: модуль інтерфейсу, модуль прогнозування обсягу закупівлі квітів, модуль роботи з базою даних. Модель функціонування інформаційної технології прогнозування обсягу закупівлі квітів наведена на рисунку 2.2.

Завдяки графічному інтерфейсу, що пов'язаний з модулями прогнозування та роботи з базою даних користувач отримує всю необхідну йому інформацію. Він відповідає за коректне відображення на екрані результатів прогнозування, а також за введення та виведення відповідної інформації. Користувач може додавати інформацію, переглядати архіви, вводити проміжки часу, на які

здійснюватиметься прогноз, переглядати прогнозовані значення завдяки модулю інтерфейсу.

Модуль прогнозування пов'язаний з модулем роботи з базою даних. Цей модуль є головним модулем програми та містить інтелектуальний алгоритм прогнозування обсягу закупівлі квітів. Прогнозування відбується при застосуванні методу аналізу часових рядів.

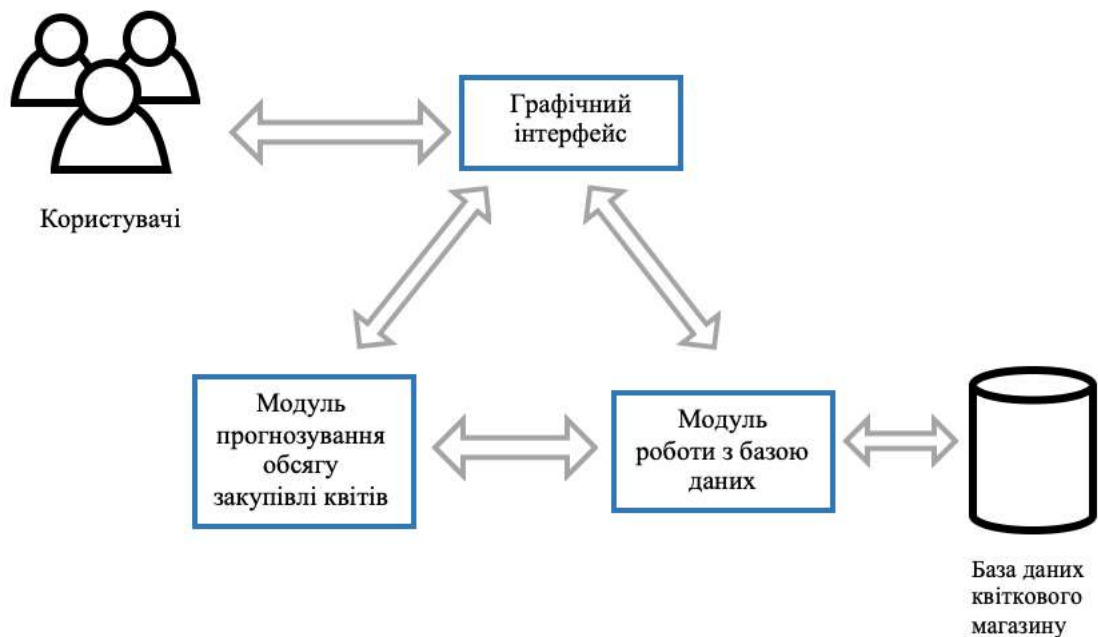


Рисунок 2.2 – Схема моделі функціонування інформаційної технології прогнозування обсягу закупівлі

Модуль роботи з базою даних пов'язаний з модулем прогнозування та модулем інтерфейсу. Даний модуль забезпечує зв'язок з базою даних, в якій містяться дані квітового магазину. Цей модуль містить запити на мові SQL, які дозволяють переглядати архів, що містить інформацію про закупку квітів минулих місяців. SQL є основою багатьох баз даних, оскільки відповідає за фізичну структуру і запис даних на диск, а також читання даних з диска, дозволяє отримувати SQL-запити від інших компонентів бази даних і користувацьких програм [25]. Таким чином, SQL є потужним інструментом, який надає користувачам, програмам і комп'ютерним системам доступ до інформації в

реляційних базах даних. База даних повинна бути синхронізованою з інформаційною технологією прогнозування обсягу закупівлі квітів та забезпечувати швидкий імпорт та автоматичний експорт усієї інформації, необхідної для прогнозування.

2.4 Розробка алгоритму функціонування інформаційної технології

Розглянемо загальний алгоритм роботи додатку, що реалізує інформаційну технологію прогнозування обсягу закупівлі квітів (рис. 2.3).

Даний алгоритм складається з наступних кроків:

1. На першому кроці відбувається встановлення зв'язку з базою даних.
2. На другому етапі відбувається виведення головного меню програми, в якому відображено три варіанти подальших дій: показ архіву, додавання інформації та прогнозування.
 3. Перевірка на необхідність відображення архіву:
 - 3.1. Введення початкової та кінцевої дати для показу архіву.
 - 3.2. На даному етапі виконується запит до бази даних щодо виведення обсягів закупівлі квітів в обраний період.
 - 3.3. Відображення результату. Перехід до пункту 8.
 4. Перевірка на необхідність додавання інформації для здійснення прогнозування:
 - 4.1. Введення даних щодо обсягу закупівлі квітів певної дати у запропоноване вікно.
 - 4.2. Запис введених даних до бази даних.
 5. Введення початкової та кінцевої дати – часового проміжку, на який має здійснюватися прогнозування.
 6. Прогнозування обсягу закупівлі квітів на вказаний період.
 7. Виведення результату прогнозування.
 8. Закінчення діалогу з системою.

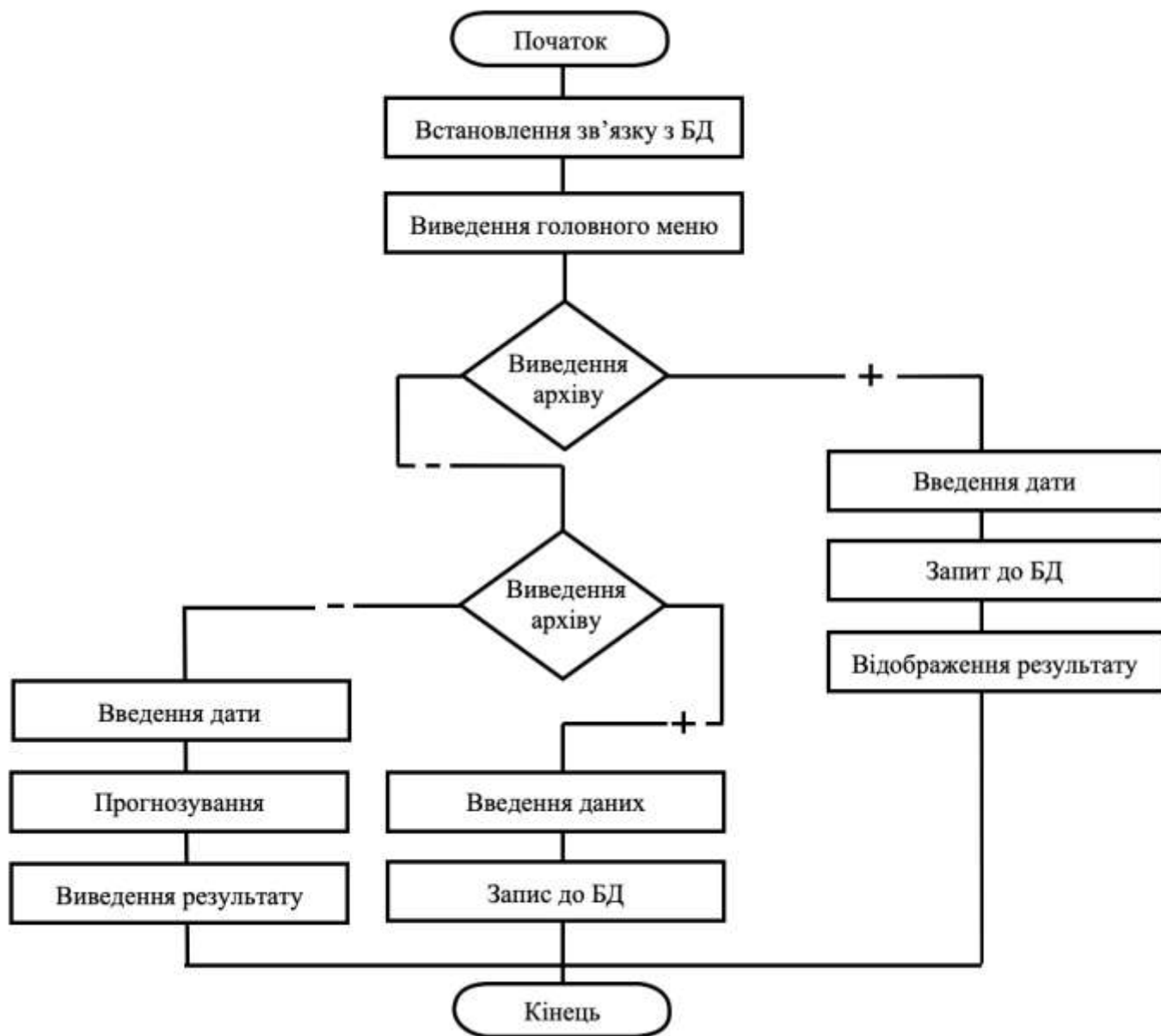


Рисунок 2.3 – Схема загального алгоритму функціонування інформаційної технології прогнозування обсягу закупівлі

На рисунку 2.4 зображено алгоритм виведення архіву, що містить інформацію про попередні закупівлі квітів.

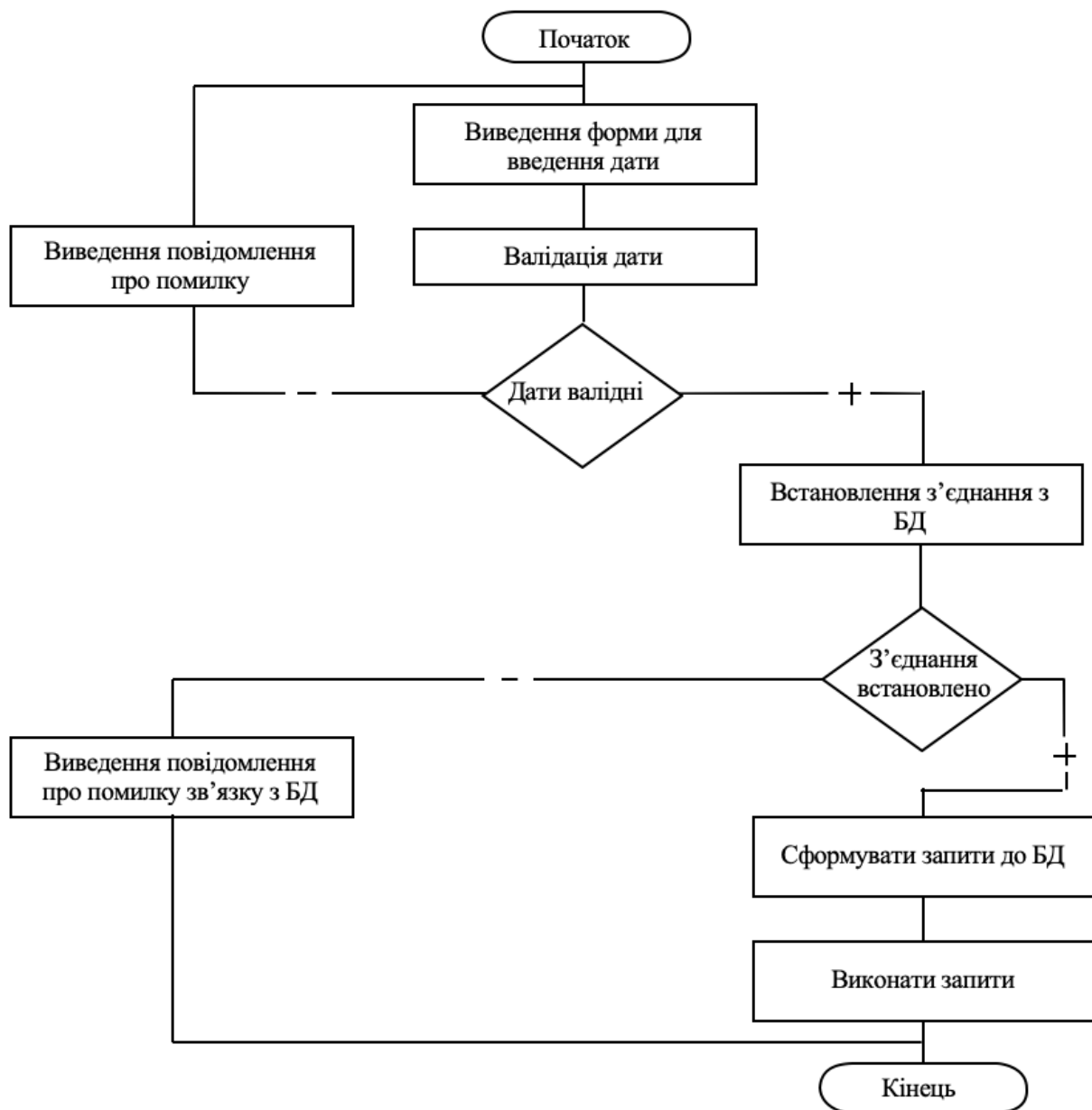


Рисунок 2.4 – Схема алгоритму виведення архіву прогнозування обсягу закупівлі

Даний алгоритм містить наступні кроки:

1. Виведення форми, в яку користувач вводить дату початку та дату кінця періоду, на який необхідно здійснити прогноз;
2. На даному етапі відбувається валідація дат;
3. Перевірка валідності дат. Якщо дати валідні – перехід до кроку 5.

4. Якщо дані не пройшли перевірку на валідність – виводиться повідомлення про помилку. Після цього користувач має змогу пройти заново алгоритм, перейшовши до його першого кроку.

5. Якщо дані валідні – відбувається встановлення зв'язку з базою даних.

6. Перевірка на встановлення зв'язку. При успішному з'єднанні з базою даних відбувається перехід до кроку 8.

7. Виведення повідомлення про відсутність зв'язку з базою даних, завершення діалогу з системою.

8. Формування запиту до бази даних.

9. Виконання запитів, завершення діалогу з системою.

На рисунку 2.5 зображено алгоритм запису інформації щодо поточної закупівлі квітів в архів.

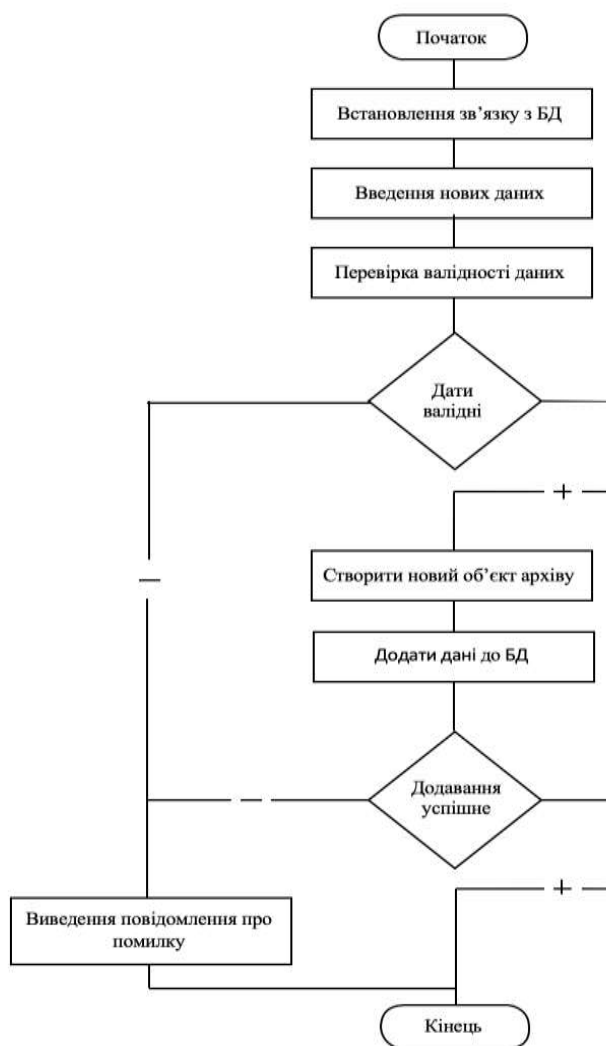


Рисунок 2.5 – Схема алгоритму запису інформації даних замовлення квітів

Даний алгоритм містить наступні кроки:

1. Встановлення зв'язку з базою даних.
2. Введення нових даних.
3. Валідація введених даних.
4. Перевірка на валідність даних, якщо дані невалідні – перехід до б.
5. Створення нового об'єкту архіву.
6. Додавання введених даних до бази даних.
7. Перевірка на успішність додавання даних. Якщо додавання успішне – завершення діалогу.
8. Виведення повідомлення про помилку. Завершення діалогу з системою.

2.5 Висновок до розділу 2

У другому розділі обґрунтовано використання методів прогнозування часових рядів, перевагами яких є те, що вони не накладають обмежень на довжину часового ряду, швидко адаптуються до динаміки змін часового ряду, не вимагають складних математичних розрахунків. На основі вибраних методів було розроблено математичну модель, що поєднує в собі алгоритми прогнозування часових рядів з прорахуванням тренду і сезонності та методу ковзного середнього для прогнозування обсягу закупівлі сезонних квітів та методу експоненціального згладжування для несезонних квітів, що дозволяє максимально задовольнити потреби та максимально точно спрогнозувати потрібну кількість квітів для закупівлі. Спроековано структуру інформаційної технології прогнозування обсягу закупівлі квітів, наведено модель функціонування інформаційної технології. Розроблено алгоритми функціонування інформаційної технології, а також наведено алгоритм виведення архіву та алгоритму запису інформації в базу даних.

3 СТРУКТУРНА ОРГАНІЗАЦІЯ ТА ОСОБЛИВОСТІ ПРОГРАМНОЇ РЕАЛІЗАЦІЇ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПРОГНОЗУВАННЯ ОБСЯГУ ЗАКУПІВЛІ КВІТІВ

3.1 Обґрунтування вибору мови та середовища програмування

Розглянемо популярні мови програмування для реалізації інформаційної технології обсягу закупівлі квітів. Java і Python є двома найпопулярнішими мовами програмування сьогодні. Хоча Java була найпопулярнішою мовою програмування з моменту її випуску в 1995 році, Python також набирає популярності з кожним роком. Розглянемо більш детально кожен з них.

Мова програмування Java – це універсальна, узгоджена, суворо типізована, об'єктно-орієнтована мова на основі класів. Зазвичай вона компілюється в набір інструкцій байт-коду та двійковий формат, визначений у специфікації віртуальної машини Java [26].

Хорошими прикладами програмного забезпечення мовою Java є банківські системи – великі розподілені програми, які успішно працюють протягом десятиліть. Програма, написана 10 років тому, буде успішно працювати на новітній версії платформи Java (JVM – Java Virtual Machine), не вимагаючи великих ресурсів та інвестицій. Платформа Java спочатку була розроблена для роботи на сильно завантажених системах з мільйонами користувачів, і досі досягає успіху. Доказом цього є активне використання Java такими гігантами ІТ-індустрії, як Google, Twitter і Amazon.

Об'єктно орієнтована мова програмування Java надзвичайно портативна. Одна й та сама програма Java працюватиме однаково на будь-якому комп'ютері, незалежно від апаратних можливостей чи операційної системи, якщо в ній є інтерпретатор Java. Крім портативності, ще однією з вагомих переваг Java є набір функцій безпеки, що захищають ПК, на якому запущена програма Java, не лише від проблем, спричинених помилковим кодом, але й від шкідливих програм (наприклад, вірусів). Ви можете безпечно запускати аплет Java, завантажений з

Інтернету, оскільки функції безпеки Java перешкоджають цим типам аплетів отримати доступ до жорсткого диска ПК або мережевих підключень. Аплет, як правило, є невеликою програмою на Java, яка вбудована в сторінку HTML.

Мову Java можна називати як компільованою, так і інтерпретованою мовою, так як її вихідний код спершу компілюється у двійковий байт-код. Цей байт-код виконується на віртуальній машині Java (JVM), що частіше за все є програмним інтерпретатором. Використання скомпільованого байт-коду дозволяє інтерпретатору (віртуальній машині) бути невеликим і ефективним (і майже таким же швидким, як і процесор, який виконує власний скомпільований код). Крім того, цей байт-код надає Java переносимість: він працюватиме на будь-якій JVM, яка правильно реалізована, незалежно від конфігурації апаратного чи програмного забезпечення комп'ютера. Більшість веб-браузерів (наприклад, Microsoft Internet Explorer або Netscape Communicator) містять JVM для запуску Java-аплетів.

Наведемо переваги мови Java [27].

1. Простота. Java є простою мовою програмування, тому що її легко вивчити і легко зрозуміти. Його синтаксис заснований на C++ і використовує автоматичне збирання сміття; тому нам не потрібно видаляти з пам'яті об'єкти без посилання. Java також вилучила такі функції, як явні покажчики, перевантаження операторів тощо, що полегшує читання та запис.

2. Об'єктно-орієнтованість. Java використовує об'єктно-орієнтовану парадигму, що робить її більш практичною. Усе в Java є об'єктом, який піклується як про дані, так і про поведінку. Java використовує об'єктно-орієнтовані концепції, такі як об'єкт, клас, успадкування, інкапсуляція, поліморфізм та абстракція.

3. Безпечність. Java є безпечною мовою програмування, оскільки не використовує явних покажчиків. Крім того, програми Java запускаються в пісочниці віртуальної машини. JRE також надає завантажувач класів, який використовують для динамічного завантаження класу в JVM. Він відокремлює пакунки класів з локальної файлової системи від імпортованих з мережі.

4. Надійність. Java є надійною мовою програмування, оскільки вона використовує потужне управління пам'яттю. Ми також можемо обробляти винятки за допомогою коду Java. Крім того, ми можемо використовувати перевірку типів, щоб зробити наш код безпечнішим. Він не надає явних покажчиків, щоб програміст не міг отримати доступ до пам'яті безпосередньо з коду.

5. Незалежність від платформи. Java-код може працювати безпосередньо на кількох платформах, тобто нам не потрібно компілювати його щоразу. Це лише один раз, запустіть в будь-якому місці (WORA) мову, яку можна перетворити на байтовий код під час компіляції. Байт-код – це незалежний від платформи код, який може виконуватися на кількох платформах.

6. Багатопотоковість. Java використовує багатопотокове середовище, де більшу задачу можна перетворити на кілька потоків і виконати окремо. Основна перевага багатопоточності полягає в тому, що нам не потрібно надавати пам'ять кожному запущеному потоку.

Далі розглянемо мову Python. Python – це високорівнева, універсальна і дуже популярна мова програмування. Мова програмування Python використовується у веб-розробці, програмах машинного навчання, а також у всіх передових технологіях в індустрії програмного забезпечення [28]. Мова програмування Python дуже добре підходить для початківців, а також досвідчених програмістів з іншими мовами програмування, такими як C++ і Java.

На даний момент Python є найбільш широко використовуваною багатоцільовою мовою програмування високого рівня. Python дозволяє програмувати в об'єктно-орієнтованих і процедурних парадигмах. Програми на Python, як правило, менші, ніж інші мови програмування, такі як Java. Програмістам доводиться вводити відносно менше, а вимога до відступів мови робить їх читаними весь час. Мова Python використовується майже всіма технологічними гігантами, такими як Google, Amazon, Facebook, Instagram, Dropbox, Uber тощо. Найбільшою перевагою Python є величезна колекція стандартних бібліотек.

Python використовується у двох основних сферах: наукових обчисленнях та науковій інформації. Він має широкий вибір бібліотек, ресурсів і ресурсів, корисних для дослідження.

Python має бути однією з найпростіших мов, оскільки він підкреслює читабельність коду. Його бібліотеки та база коду для читання допомагають програмістам оновлювати й підтримувати інформацію про програмне забезпечення без додаткових зусиль і витрат на курси розробки програмного забезпечення.

Концепція машинного навчання включає в себе класифікацію кількох речей, таких як фінансові послуги, поведінка розпізнавання голосу і навіть вибір місць на Netflix. Python використовується для машинного навчання через свої зрозумілі бібліотеки та фреймворки машинного навчання PyTorch, які включають дві з найпопулярніших – TensorFlow і scikit learn для кластеризації та вибору моделі.

Python є досконалою мовою в індустрії інвестиційного банкінгу та фінансових хедж-фондів. Він не має вбудованої підтримки алгебри лінійної попередньої обробки, але натомість має досить простий спосіб написання коду. Аналітики хедж-фондів використовують найбільш зрозумілий спосіб представлення набору матриць і векторів у вигляді списків і вкладених списків.

Існує багато переваг використання Python як мови програмування, тому наведемо деякі з них [29]:

1. Прискорення роботи програми за допомогою спеціальних моделей виконання.
2. Підтримка модулів і пакетів.
3. Модулі використовують Інтернет-протоколи та надають допомогу в супутніх технологіях.
4. Заохочує модульність програми та повторне використання коду.
5. Забезпечує підвищення ефективності продуктів або проєктів.
6. Підтримує функціональну та процедурну мову програмування.

7. Відкритий вихідний код та його розповсюдження можна вільно поширювати.

8. Сумісний з платформами, базою даних, продуктами та програмами

У порівнянні з Python, Java начебто стоїть між C++ і Python. Програми написані на Java зазвичай працюють швидше, ніж відповідні програми на Python, і повільніше, ніж C++. Як і C++, Java виконує статичну перевірку типів, але Python ні. Розглянувши мови програмування Python та Java, можна зробити висновок, що завдяки перевагам Java саме її необхідно вибрати для розробки інформаційної технології прогнозування обсягу закупівлі квітів. Тому доцільно розглянути середовище розробки мовою програмування Java.

Для написання програми мовою програмування Java необхідно розглянути деякі середовища, де розробник може реалізувати коди та програми. Для цього використовують інтегроване середовище розробки Java (Java IDE). Потреба в Java IDE виникла, коли у розробників виникли проблеми з кодуванням величезної програми.

Великі програми будуть мати багато класів і файлів, тому їх важко налаштувати. Використовуючи IDE можна підтримувати належне керування проектами. Він надає вказівки щодо завершення коду, синтаксичних помилок тощо.

Інтегроване середовище розробки (IDE) – це програмний застосунок, що дозволяє розробникам використовувати платформу з великою кількістю функцій та інструментів для розробки комп'ютерних програм, веб-сторінок, інструментів, служб тощо. Інструмент IDE включає текстові редактори, налагоджувачі, компілятори спільно з деякими функціями та інструментами, що допомагають автоматизувати, тестувати та аналізувати робочий процес розробки додатків.

З метою розробки серверних програм Java зазвичай використовують три середовища розробки, а саме IntelliJ IDEA, Eclipse і NetBeans.

IntelliJ IDEA – це середовище розробки для створення програмних застосунків за допомогою Java. IntelliJ IDEA розроблялася компанією JetBrains

[30]. Він доступний у вигляді ліцензованої версії спільноти Apache 2 і власної комерційної версії. Обидва середовища можна використовувати для прогнозування обсягу закупівлі квітів. Однак саме IntelliJ IDEA було вибрано для реалізації інформаційної технології обсягу закупівлі квітів. Приклад роботи в IntelliJ IDEA наведено на рисунку 3.1.

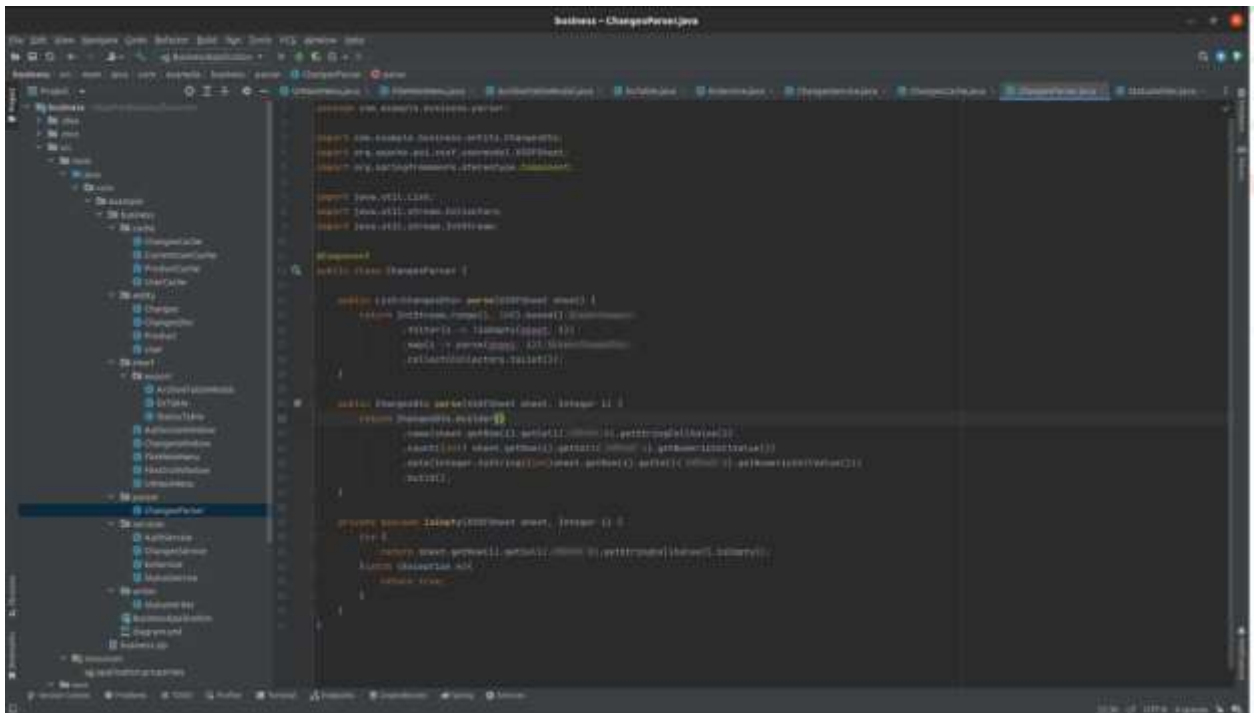


Рисунок 3.1 – Загальний вигляд середовище розробки IntelliJ IDEA

IntelliJ IDEA надає рекомендації щодо завершення коду, аналізу коду та перевірених інструментів рефакторингу. Середовище має такі інструменти, як контроль версій, зв'язок з багатьма мовами і фреймворками. Воно може стежити за контекстом розробника і самостійно запускати потрібні інструменти.

Особливості IntelliJ IDEA [31]:

- Розумний висновок: у ньому перераховані найбільш релевантні символи, які стосуються поточного контексту. Він безперервно переміщує новітні класи, методи тощо, збільшує список пропозицій. Відповідно, заповнення коду проводиться швидше.
- Аналіз потоку даних: дане середовище може аналізувати потік даних та передбачати можливий символ при виконанні.

- Мовне введення: можна легко інтегрувати фрагменти іншої мови у свій код Java, наприклад SQL.
- IntelliJ рекомендує глибокий та ефективний рефакторинг, так як він має необхідні знання про використання символів.
- Середовище пропонується з широким набором вбудованих інструментів, як наприклад GIT, контроль версій, декомпілятор, покриття, база даних SQL тощо.
- Найвний потужний компілятор, що може виявляти дублікати, запах коду тощо.
- Має надійну інтеграцію з серверами додатків.

Отже, завдяки наведеним перевагам, було вибрано мову програмування Java та середовище розробки IntelliJ IDEA.

3.2 Обґрунтування вибору бібліотек та фреймворків для інформаційної технології

Програмування інформаційної технології прогнозування обсягу закупівлі квітів здійснюється мовою програмування Java. Розглянемо деякі бібліотеки, що використовуються в розробленій інформаційній технології.

Swing – інструмент для створення графічного інтерфейсу користувача (GUI) на мові програмування Java. Це частина бібліотеки класів Java Foundation (JFC) [32].

Swing був розроблений, щоб надати більш функціональний набір програмних компонентів для створення графічного інтерфейсу користувача, ніж попередні інструменти AWT. Компоненти Swing підтримують специфічні модулі зовнішнього вигляду, які динамічно підключаються. Вони дозволяють емулювати графічний інтерфейс платформи (тобто компонент може бути динамічно підключений до іншого, специфічного для цього типу та поведінки операційної системи). Вагомим недоліком таких компонентів є відносно повільна робота, хоча це нещодавно не було підтверджено через зростаючу

потужність персональних комп'ютерів. Перевага - універсальність інтерфейсу створюваних програм на всіх платформах.

У наведеному нижче коді наведено приклад використання Swing, що відображає вікно початкової активності – головне вікно програми, яке містить кнопки «Показати архів», «Додати інформацію», «Прогнозування».

```
package java1.interf;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class MainMenu {
    private ImageIcon image;
    private JFrame frame;
    private JLabel label;

    public MainMenu() {
        image = new ImageIcon(getClass().getResource("1.jpg"));
    }

    public void createWindow() {
        frame = new JFrame();
        frame.setTitle("Головне меню");
        frame.setSize(400, 350);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setLocationRelativeTo(null);

        Font font1 = new Font("TimesRoman", Font.BOLD, 14);
        label = new JLabel();
        label.setLayout(new GridBagLayout());
        label.setVisible(true);
        label.setBackground(Color.BLUE);
        label.setIcon(image);
    }
}
```

```

        JButton exFlowers = new JButton();
        exFlowers.setText("Показати архів");
        exFlowers.setFont(font1);
        label.add(exFlowers, new GridBagConstraints(0, 0, 1, 1, 0.0, 0.9,
GridBagConstraints.CENTER,
                GridBagConstraints.HORIZONTAL, new Insets(30, 10, 10, 10), 0, 0));
        exFlowers.addActionListener(new exFlowersListener());

        JButton newFlowers = new JButton();
        newFlowers.setText("Додати інформацію");
        newFlowers.setFont(font1);
        label.add(newFlowers, new GridBagConstraints(0, 1, 1, 1, 0.0, 0.9,
GridBagConstraints.CENTER,
                GridBagConstraints.HORIZONTAL, new Insets(30, 10, 10, 10), 0, 0));
        newFlowers.addActionListener(new newFlowersListener());

        JButton newForecast = new JButton();
        newForecast.setText("Прогнозування");
        newForecast.setFont(font1);
        label.add(newForecast, new GridBagConstraints(0, 2, 1, 1, 0.0, 0.9,
GridBagConstraints.CENTER,
                GridBagConstraints.HORIZONTAL, new Insets(30, 10, 10, 10), 0, 0));
        newForecast.addActionListener(new newForecastListener());

        frame.add(label);
        frame.setVisible(true);
    }

    public void closeWindow() {
        frame.setVisible(false);
        frame.remove(label);
    }
}

```

Java DataBase Connectivity (JDBC) – це інтерфейс програмування Java, який визначає методи, за допомогою яких програмне забезпечення на Java отримує доступ до бази даних. JDBC – це незалежний від платформи галузевий стандарт

для взаємодії програм Java з різними базами даних, реалізований у формі пакета `java.sql`, який є частиною Java SE [33].

Перевагами JDBC є:

- простота розробки: розробник може не знати особливостей бази даних, з якою працює;
- при переході компанії до іншої бази даних код не змінюється;
- немає необхідності встановлювати громіздку клієнтську програму;
- можливе підключитися до будь-якої бази даних за допомогою легко описаної URL-адреси.

Java DataBase Connectivity – це стандартний API для самостійного підключення мови програмування Java до різних баз даних [34].

JDBC вирішує такі завдання: створення підключення до бази даних, створення виразів SQL, виконання запитів SQL, перегляд та модифікація отриманих записів. Загалом, JDBC – це бібліотека, яка забезпечує цілий набір інтерфейсів для доступу до різних баз даних.

Для доступу до кожної конкретної бази даних потрібен спеціальний драйвер JDBC, який є адаптером програми Java до бази даних.

Бібліотека JDBC, що забезпечує цілий набір інтерфейсів для доступу до різних баз даних використовується в наступному фрагменті коду:

```
package java1.db;

import java.sql.*;
import java.util.ArrayList;

public class FlowerDAO {

    private static final String mySQL = "INSERT INTO flowers (dateS, rrl, rrm, rye, rw, rr, ro,
ow, flowers.or, oy, ch) " +
        "VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?)";
    private static final String myGetSQL = "SELECT * FROM flowers where dateS > ? and
dateS < ?";
```

```

public void addNewFl(db.Flower flower){
    PreparedStatement pstmt = null;
    ResultSet rs = null;
    Connection con = null;
    DBManager dbManager = DBManager.getInstance();
    try {
        con = dbManager.getConnection();
        pstmt = con.prepareStatement(mySQL,
Statement.RETURN_GENERATED_KEYS);
        pstmt.setString(1, flower.getDate());
        pstmt.setInt(2, flower.getRoseRedLong());
        pstmt.setInt(3, flower.getRoseRedMini());
        pstmt.setInt(4, flower.getRoseYellow());
        pstmt.setInt(5, flower.getRoseWhite());
        pstmt.setInt(6, flower.getRosePink());
        pstmt.setInt(7, flower.getRoseOrange());
        pstmt.setInt(8, flower.getOrchidWhite());
        pstmt.setInt(9, flower.getOrchidPink());
        pstmt.setInt(10, flower.getOrchidYellow());
        pstmt.setInt(11, flower.getChrysanthemums());
        System.out.println(pstmt);
        pstmt.executeUpdate();
        rs = pstmt.getGeneratedKeys();
        if (rs.next()) {
            flower.setId(rs.getInt(1));
        }
    } catch (SQLException throwables) {
        throwables.printStackTrace();
    } finally {
        close(rs);
        close(pstmt);
        close(con);
    }
}

```

3.3 Опис процесу програмної розробки

Розробимо діаграму діяльності інформаційної технології прогнозування обсягу закупівлі квітів при введенні нових даних. Наведений нижче код описує зображену на рисунку 3.2 діаграму діяльності системи інформаційної технології при введенні нових даних :

```

JButton newForecast = new JButton();
    newForecast.setText("Прогнозування");
    newForecast.setFont(font1);
    label.add(newForecast, new GridBagConstraints(0, 2, 1, 1, 0.0, 0.9,
GridBagConstraints.CENTER,
        GridBagConstraints.HORIZONTAL, new Insets(30, 10, 10, 10), 0, 0));
    newForecast.addActionListener(new newForecastListener());
    frame.add(label);
    frame.setVisible(true);
}
public void closeWindow() {
    frame.setVisible(false);
    frame.remove(label);
}
private class newFlowersListener implements ActionListener {
    @Override
    public void actionPerformed(ActionEvent JCom) {
        closeWindow();
        new PageForNewFlowers().createWindow();
    }
}
private class exFlowersListener implements ActionListener {
    @Override
    public void actionPerformed(ActionEvent JCom) {
        closeWindow();
        new PageShowOldFlower().createWindow();
    }
}
private class newForecastListener implements ActionListener {

```

```

@Override
public void actionPerformed(ActionEvent JCom) {
    closeWindow();
    new PageForNewForecast().createWindow();
}
}

```

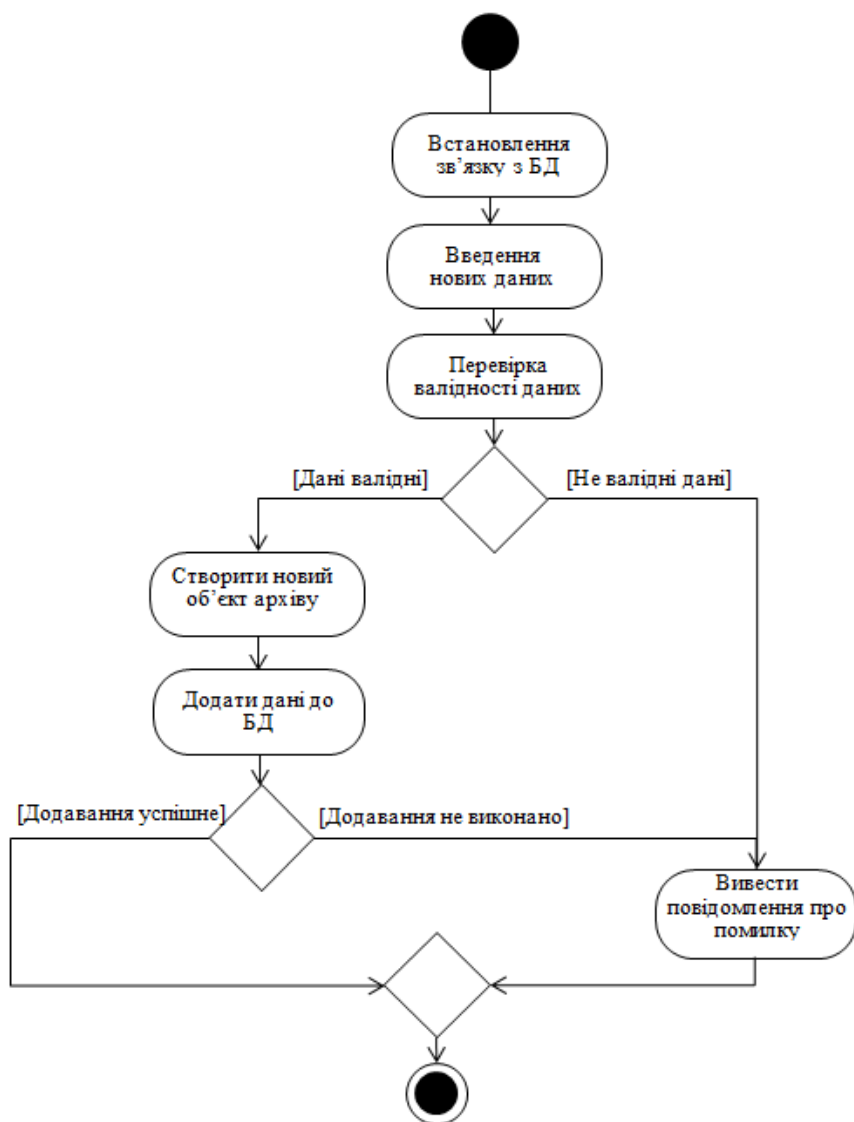


Рисунок 3.2 – Діаграма діяльності системи інформаційної технології при введенні нових даних

Діаграма діяльності інформаційної технології при прогнозуванні обсягу закупівлі квітів зображена на рисунку 3.3.

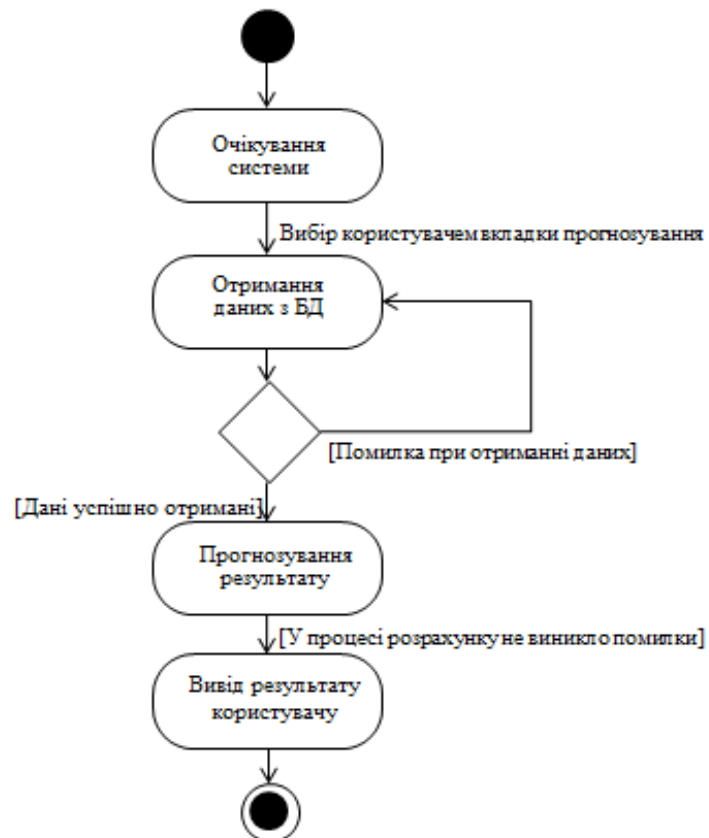


Рисунок 3.3 – Діаграма діяльності системи інформаційної технології при прогнозуванні обсягу закупівлі квітів

Діаграма діяльності інформаційної технології при прогнозуванні обсягу закупівлі квітів відображає алгоритм прогнозування закупівлі квітів. Після вибору користувачем роботи з прогнозуванням відбувається встановлення зв'язку з базою даних та отримання даних на період, вказаний у запиті користувача. При отриманні необхідних даних відбувається перехід до розрахунку прогнозованого результату. В протилежному випадку відбувається перехід до попереднього стану. Якщо прогнозування пройшло успішно і при його обрахунку не виникло помилок, користувач отримує результат прогнозування обсягу закупівлі квітів на період часу, який зацікавив користувача.

Клас Flower призначений для отримання даних про квіти, що знаходяться в базі даних.

За допомогою класу FlowerDAO відбувається виконання запитів до бази даних.

Клас DBManager призначений для здійснення доступу до бази даних. В даному класі здійснюється підключитися до розробленої бази даних за допомогою легко описаної URL-адреси:

```
private static final String URL = "jdbc:mysql://localhost:3306/flowersdb";
```

Клас CountForecast – клас з основною логікою програми, в ньому відбувається прогнозування обсягу закупівлі квітів на основі методу аналізу часових рядів.

Класи PageForNewFlowers, PageForNewForecast, PageShowOldFlower, TableForForecast відповідають за відображення графічного інтерфейсу програми з допомогою інструменту для створення графічного інтерфейсу користувача Swing. Кожен клас відповідає за відображення та функціонування окремих вікон.

3.5 Тестування інформаційної технології прогнозування обсягу закупівлі квітів

Тестування інформаційної технології прогнозування обсягу закупівлі квітів заключається у процесі технічного дослідження, призначеного для виявлення інформації про якість продукту відносно контексту, в якому його мають використовувати. За допомогою тестування необхідно оцінити відповідність поставленим вимогам, правильність відповіді для всіх можливих вхідних даних, швидкість виконання функцій, практичність, переваги над аналогічними рішеннями.

Провівши 100 запусків додатку, було протестовано можливості його роботи, здійснено оцінку функціональних можливостей. Тестування

реалізованої інформаційної технології прогнозування обсягу закупівлі квітів підтвердило коректність її роботи.

Початок роботи з додатком починається з відкриття головного меню інформаційної технології. Дане меню включає наступні пункти – «Показати архів», «Додати інформацію», «Прогнозування» (рис. 3.5). Натиснувши на кожен з пунктів відбувається перехід до наступного функціонального вікна. Отже, на першому кроці необхідно натистути кнопку у вікні початкової активності, що відповідає пункту меню, який цікавить користувача.

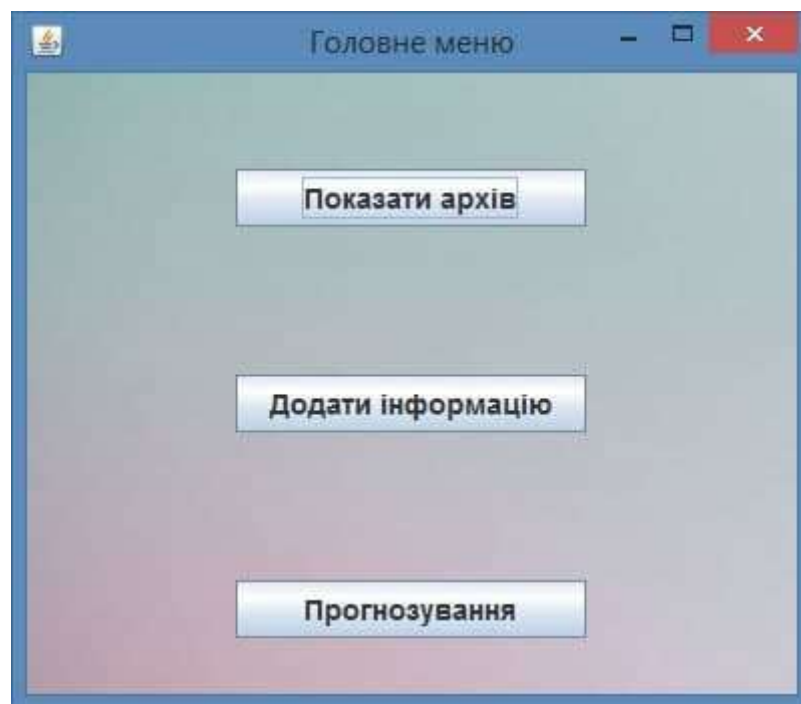


Рисунок 3.5 – Загальний вигляд вікна головного меню

Якщо користувач вибрав перший пункт головного меню програми – «Показати архів» – відбувається перехід до вікна архіву. Дане вікно містить поля для введення початкової та кінцевої дати, тобто конкретний проміжок часу, за який було здійснено закупівлі квітів, що цікавить користувача в даний момент (рис. 3.6).

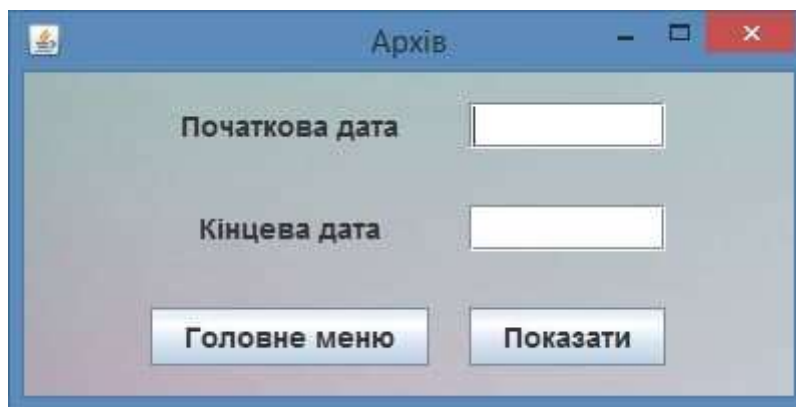


Рисунок 3.6 – Загальний вигляд вікна введення проміжку часу

Дане вікно містить також кнопки «Головне меню» та «Показати». Натискання кнопки «Головне меню» приводить до попереднього вікна з вибором пунктів для подальшої роботи. При введенні необхідного проміжку часу та натискання кнопки «Показати» відбувається перехід до вікна архіву, де міститься таблиця з закупками квітів на вказаний користувачем проміжок часу (рис. 3.7). Таблиця формується шляхом імпорту значень з бази даних.

Дата	Троянда черво.	Троянда черво.	Троянда жовта	Троянда біла	Троянда рожева	Троянда оран.	Орхідея біла	Орхідея рожева	Орхідея жовта	Хризантема ас.
2019-07-05	9	13	14	12	9	12	11	4	5	3
2019-07-06	6	12	16	9	9	7	9	6	5	4
2019-07-07	7	13	9	13	13	11	14	6	4	5
2019-07-08	11	8	13	8	13	8	12	6	4	5
2019-07-09	7	12	12	10	15	10	15	4	5	4
2019-07-10	9	13	10	13	11	10	11	3	5	3
2019-07-11	9	8	12	9	11	7	9	4	5	5
2019-07-12	10	16	13	9	15	8	17	3	4	3
2019-07-13	10	11	18	12	13	11	15	3	3	3
2019-07-14	12	8	16	8	10	6	16	4	5	3

Рисунок 3.7 – Загальний вигляд вікна «Архів»

Вікно архіву містить таблицю з атрибутами «Дата» та назвами квітів, що закуповує магазин. В таблицю виводяться дати, коли здійснювалась закупка

квітів обраного періоду, а також кількість квітів різних видів, що були закуплені відповідного дня.

У розглянутому вікні присутні кнопки, що забезпечують перехід до попередніх вікон. Завдяки кнопці «Назад» відбувається перехід до вінка введення періоду часу, необхідного для перегляду архівних даних. Кнопка «Головне меню» повертає користувача до вибору пунктів головного меню та подальшої роботи з ними.

При виборі пункту головного меню «Додати інформацію» виводиться вікно для введення поточної закупівлі, що має бути збережена в базу даних (рис. 3.8).



Дата	
Троянда червона метрова	<input type="text"/>
Троянда червона міні	<input type="text"/>
Троянда жовта	<input type="text"/>
Троянда біла	<input type="text"/>
Троянда рожева	<input type="text"/>
Троянда оранжева	<input type="text"/>
Орхідея біла	<input type="text"/>
Орхідея рожева	<input type="text"/>
Орхідея жовта	<input type="text"/>
Хризантеми асорті	<input type="text"/>

Рисунок 3.8 – Загальний вигляд вікна введення нових даних

Наведене на рисунку 3.8 вікно містить поля для введення дати закупівлі та кількості квітів відповідно до їх виду. Вікно містить кнопку «Додати дані», що

забезпечує запис введених значень, заповнених у відповідні поля користувачем, для їх подальшого збереження в базі даних. З даного вікна також можливий перехід до вікна початкової активності – головного меню, що здійснюється за допомогою кнопки «Головне меню».

При виборі останнього пункту головного меню – «Прогнозування» – відбувається перехід до вікна прогнозування. Дане вікно містить поля для введення початкової та кінцевої дати, тобто конкретний проміжок часу, на який потрібно здійснити прогноз закупівлі квітів, що цікавить користувача в даний момент (рис. 3.9).

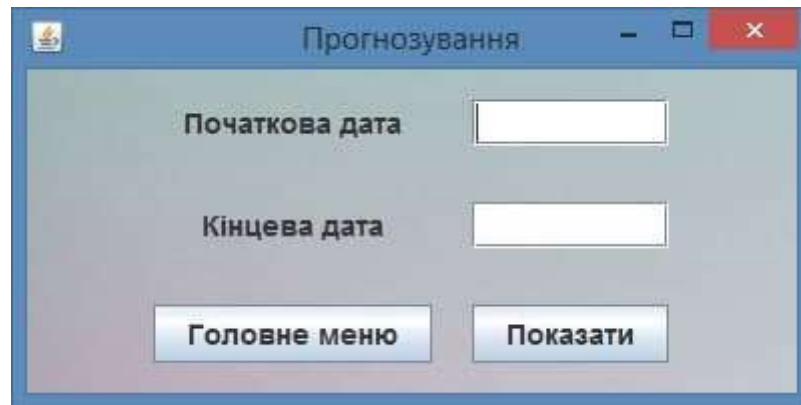
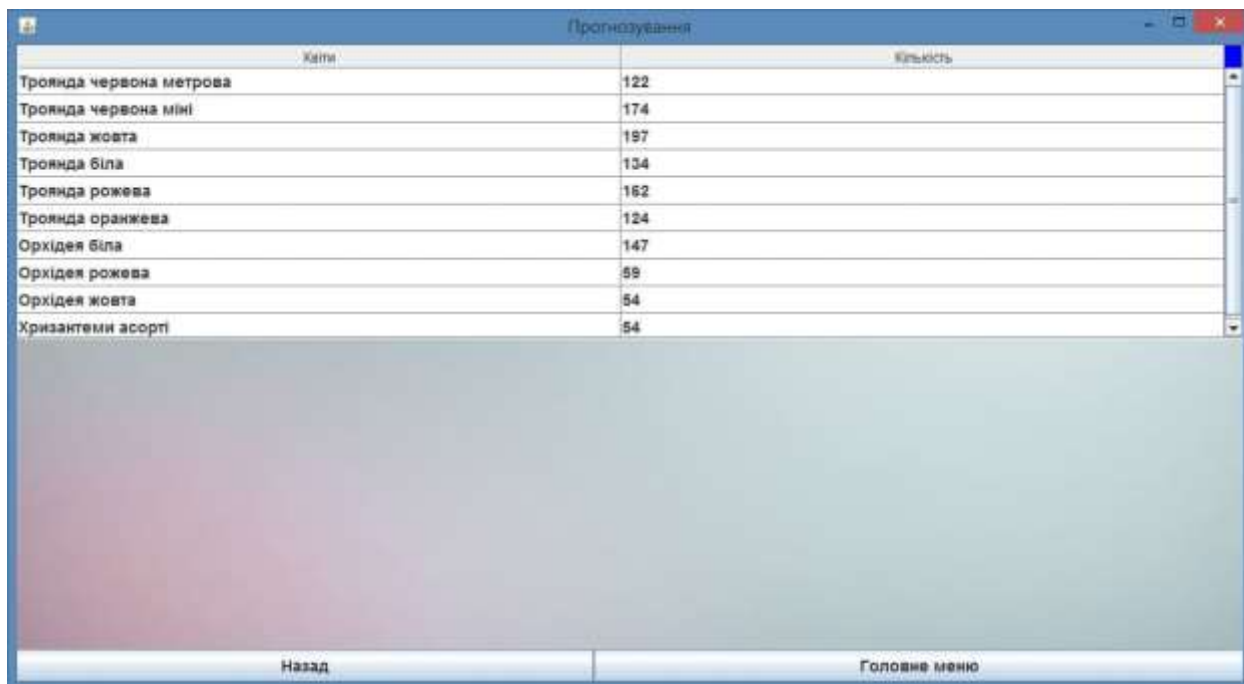


Рисунок 3.9 – Загальний вигляд вікна введення проміжку часу для прогнозування

Дане вікно містить кнопку для повернення до вікна початкової активності, якщо користувач на попередньому кроці помилково вибрав пункт «Прогнозування».

Після введення початкової та кінцевої дати користувач натискає кнопку «Показати». Після цього програма аналізує дати на присутність сезонних чинників та здійснює прогнозування обсягу закупівлі сезонних квітів методом ковзного середнього. Для несезонних квітів прогнозування відбувається завдяки методу експотенціального згладжування. Після проведення інформаційною технологією необхідних обчислень відбувається перехід до вікна, що містить таблицю прогнозованих значень закупівлі квітів на обраний користувачем період часу (рис. 3.10).



Квіти	Кількість
Троянда червона метрова	122
Троянда червона міні	174
Троянда жовта	197
Троянда біла	134
Троянда рожева	162
Троянда оранжева	124
Орхідея біла	147
Орхідея рожева	89
Орхідея жовта	84
Хризантеми асорті	84

Рисунок 3.10 – Загальний вигляд вікно виведення результатів прогнозування

Таблиця результатів прогнозування містить атрибути «Квіти» та «Кількість». В ній наведені прогнозована кількість квітів певного виду, що доцільно закупити на обраний період часу.

Отже, розглянувши приклад роботи додатку можна зробити висновок, що інформаційна технологія задовільняє наступні вимоги:

Вимоги до клієнтів:

- головна сторінка з меню;
- перегляд архіву у вигляді таблиці;
- можливість вибору терміну;
- можливість додавання нових даних;
- завершення сесії (вихід).

Вимоги до менеджера:

- можливість редагування інформації;
- можливість додавання нової інформації (квітів);
- можливість видалення інформації;

Вимоги до серверу:

- зберігання даних користувача через зв'язок з базою даних;
- видача необхідних даних по запиту від клієнта.

Проведемо тестування інформаційної технології на відповідність прогнозованих результатів реальним значенням продажу, тобто оцінимо точність прогнозування. Для цього обираємо проміжок часу з 02 червня 2022 року до 11 червня 2022 року (рис. 3.11).

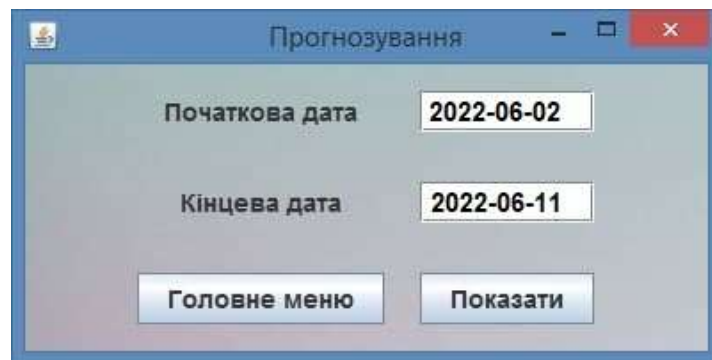
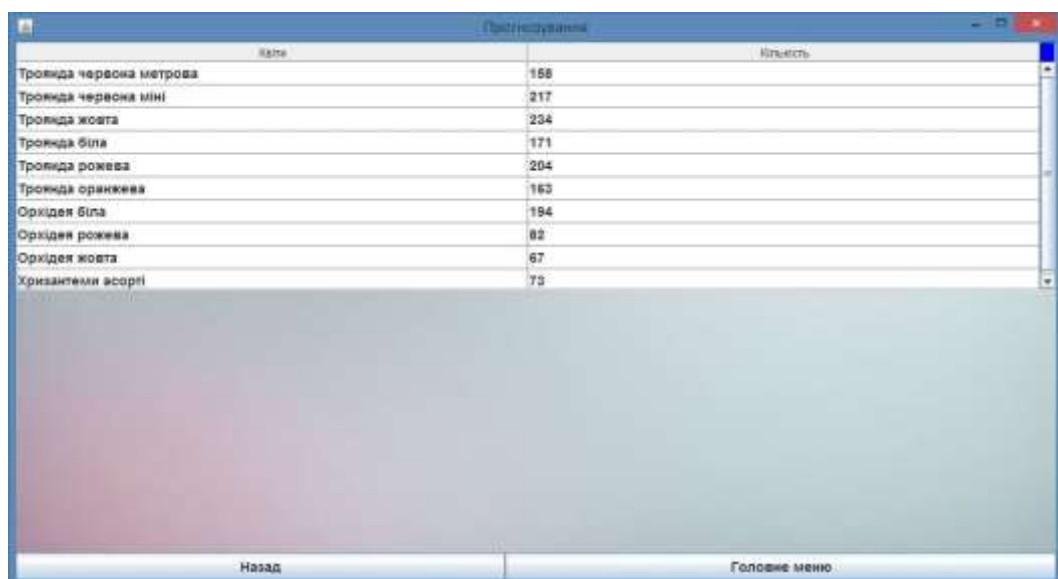


Рисунок 3.11 – Загальний вигляд вікна введення проміжку часу для прогнозування

Після натискання кнопки «Показати» отримаємо результати прогнозування, обраховані за допомогою інформаційної технології (рис. 3.12).



Назва	Кількість
Троїнда червона метрова	156
Троїнда червона міні	217
Троїнда жовта	234
Троїнда біла	171
Троїнда рожева	204
Троїнда оранжева	163
Орхідея біла	184
Орхідея рожева	82
Орхідея жовта	67
Хризантеми асорті	72

Рисунок 3.10 – Загальний вигляд вікна виведення результатів прогнозування

У таблиці 3.1 наведемо реальні значення продажу квітів, здійснені в проміжок часу з 02 червня 2022 року до 11 червня 2022 року та прогнозовані значення розробленим додатком.

Таблиця 3.1 – Тестування розробленого додатку.

Тип квітів	Пронозована кількість (шт.)	Реальна кількість (шт.)	Похибка результату
Троянда червона метрова	158	150	5%
Троянда червона міні	217	205	5,6%
Троянда жовта	234	200	15%
Троянда біла	171	165	4%
Троянда рожева	204	198	3%
Троянда оранжева	163	152	7%
Орхідея біла	194	180	7,3%
Орхідея рожева	82	50	39%
Орхідея жовта	67	60	10,5%
Хризантеми асорті	73	70	4,2%

З наведених даних у таблиці 3.1 видно, що похибка розрахунку в окремому випадку досягає 39%, однак, зважаючи на те, що в інших випадках вона не досягає 15% можна зробити висновок, що інформаційна технологія прогнозування обсягу закупівлі квітів працює коректно та з достатньою точністю.

Аналогічні розрахунки були проведено за допомогою аналогів «Forecast NOW!» та «ЕКAM». У таблиці 3.2 наведено результати порівняння роботи розробленого додатку з наявними рішеннями.

Таблиця 3.2 – Порівняння результатів роботи програм

Програмний продукт	Точність прогнозування
Розроблений додаток	89,94%
Forecast NOW!	86,83%
ЕКAM	83,28%

Таким чином, із таблиці 3.2 видно, що розроблена інформаційна технологія має на 3,11% кращу точність прогнозування ніж кращий аналог, отже, мета роботи досягнута.

Результати тестування роботи програмного додатку, відповідають результатам, які очікувались.

3.6 Висновок до розділу 3

В третьому розділі було здійснено обґрунтування вибору мови програмування для реалізації інформаційної технології прогнозування обсягу закупівлі квітів. Інформаційну технологію розроблено на об'єктно-орієнтованій мові програмування Java, враховуючи її кросплатформність, об'єктно-орієнтованість та простоту розробки, в середовищі розробки IntelliJ IDEA, що має переваги у вигляді глибокого розуміння коду, задоволенні усіх вимог розробника, розумного доповнення коду.

Розроблено та описано діаграму класів, розглянуто методи та атрибути розроблених класів. Програмно реалізовано інформаційну технологію прогнозування обсягу закупівлі квітів.

Проведено тестування роботи інформаційної технології прогнозування обсягу закупівлі квітів та проаналізовано її результати. Тестування програми підтвердило очікувані результати та правильність роботи програми, довівши підвищення точності прогнозування на 3,11%.

4 ЕКОНОМІЧНА ЧАСТИНА

4.1 Проведення комерційного та технологічного аудиту науково-технічної розробки

Метою проведення комерційного і технологічного аудиту є оцінювання науково-технічного рівня та рівня комерційного потенціалу інформаційної технології прогнозування обсягу закупки квітів, створеної в результаті науково-технічної діяльності, тобто під час виконання магістерської кваліфікаційної роботи.

Аналогом може бути система «Poster», вартість використання якої складає 475 грн/місяць, тобто за три роки вартість використання даної програми складе 17100 грн.

Для проведення комерційного та технологічного аудиту залучають не менше 3-х незалежних експертів, якими можуть бути провідні викладачі випускової або спорідненої кафедри чи інші відомі фахівці. Не рекомендується залучати експертами керівника магістерської кваліфікаційної роботи та завідувача відповідної випускової кафедри.

Оцінювання науково-технічного рівня розробки та її комерційного потенціалу рекомендується здійснювати із застосуванням п'ятибальної системи оцінювання за 12-ма критеріями, наведеними в табл. 4.1.

Таблиця 4.1 – Рекомендовані критерії оцінювання науково-технічного рівня і комерційного потенціалу розробки та бальна оцінка

Бали (за п'ятибальною шкалою)					
	0	1	2	3	4
1	2	3	4	5	6
<i>Технічна здійсненість концепції</i>					
1	Достовірність концепції не підтверджена	Концепція підтверджена експертними висновками	Концепція підтверджена розрахунками	Концепція перевірена на практиці	Перевірено роботоздатність продукту в реальних умовах

Продовження таблиці 4.1

1	2	3	4	5	6
<i>Ринкові переваги (недоліки)</i>					
2	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Кілька аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому ринку
3	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінам аналогів	Ціна продукту дещо нижча за ціни аналогів	Ціна продукту значно нижча за ціни аналогів
4	Технічні та споживчі властивості продукту значно гірші, ніж в аналогів	Технічні та споживчі властивості продукту трохи гірші, ніж в аналогів	Технічні та споживчі властивості продукту на рівні аналогів	Технічні та споживчі властивості продукту трохи кращі, ніж в аналогів	Технічні та споживчі властивості продукту значно кращі, ніж в аналогів
5	Експлуатаційні витрати значно вищі, ніж в аналогів	Експлуатаційні витрати дещо вищі, ніж в аналогів	Експлуатаційні витрати на рівні експлуатаційних витрат аналогів	Експлуатаційні витрати трохи нижчі, ніж в аналогів	Експлуатаційні витрати значно нижчі, ніж в аналогів
<i>Ринкові перспективи</i>					
6	Ринок малий і не має позитивної динаміки	Ринок малий, але має позитивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивною динамікою
7	Активна конкуренція великих компаній на ринку	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкурентів немає
<i>Практична здійсненність</i>					
8	Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витратити значні кошти та час на навчання наявних фахівців	Необхідне незначне навчання фахівців та збільшення їх штату	Необхідне незначне навчання фахівців	Є фахівці з питань як з технічної, так і з комерційної реалізації ідеї
9	Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні	Потрібні незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування

Продовження таблиці 1.4

1	2	3	4	5	6
10	Необхідна розробка нових матеріалів	Потрібні матеріали, що використовуються у військово-промисловому комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві
11	Термін реалізації ідеї більший за 10 років	Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5ти років. Термін окупності інвестицій більший 5ти років	Термін реалізації ідеї менший 3-х років. Термін окупності інвестицій від 3-х до 5-ти років	Термін реалізації ідеї менший 3-х років. Термін окупності інвестицій менший 3-х років
12	Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію продукту	Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту, що потребує значних коштів та часу	Процедура отримання дозвільних документів для виробництва та реалізації продукту потребує незначних коштів та часу	Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту	Відсутні будь-які регламентні обмеження на виробництво та реалізацію продукту

Результати оцінювання науково-технічного рівня та комерційного потенціалу науково-технічної розробки потрібно звести до таблиці 4.2.

Таблиця 4.2 – Результати оцінювання науково-технічного рівня і комерційного потенціалу розробки

Критерії	Експерт (ПІБ, посада)		
	1	2	3
	Бали:		
1. Технічна здійсненність концепції	4	3	4
2. Ринкові переваги (наявність аналогів)	2	2	3
3. Ринкові переваги (ціна продукту)	4	4	4

Продовження таблиці 4.2

4. Ринкові переваги (технічні властивості)	3	4	4
5. Ринкові переваги (експлуатаційні витрати)	3	3	2
6. Ринкові перспективи (розмір ринку)	3	4	3
7. Ринкові перспективи (конкуренція)	3	3	2
8. Практична здійсненність (наявність фахівців)	4	4	4
9. Практична здійсненність (наявність фінансів)	3	3	3
10. Практична здійсненність (необхідність нових матеріалів)	4	3	3
11. Практична здійсненність (термін реалізації)	4	3	3
12. Практична здійсненність (розробка документів)	3	2	3
Сума балів	40	38	38
Середньоарифметична сума балів $СБ_c$	$СБ_c = \frac{\sum_i^3 СБ_i}{3} = 38,6$		

За результатами розрахунків, наведених в таблиці 4.2, робиться висновок щодо науково-технічного рівня і рівня комерційного потенціалу розробки. При цьому використовують рекомендації, наведені в табл. 4.3.

Таблиця 4.3 – Науково-технічні рівні та комерційні потенціали розробки

Середньоарифметична сума балів $СБ$, розрахована на основі висновків експертів	Науково-технічний рівень та комерційний потенціал розробки
41...48	Високий
31...40	Вищий середнього
21...30	Середній
11...20	Нижчий середнього
0...10	Низький

Встановивши досягнутий науково-технічний рівень розробки та її комерційний потенціал, одним-двома короткими реченнями обов'язково потрібно пояснити, за рахунок чого такий рівень було досягнуто:

- покращення та/або розширення функціональних можливостей нової науково-технічної розробки порівняно з аналогічними розробками, існуючими в цей час на ринку;
- значно вища якість, конкурентоспроможність нової науковотехнічної розробки за рахунок ... (вказати за рахунок чого);
- суттєве зростання продуктивності, ефективності нової науковотехнічної розробки за встановленими показниками (вказати ці показники) та пояснити, за рахунок чого саме це було досягнуто;
- значне зменшення витрат ресурсів і часу на виконання поставлених завдань (вказати яких саме) та за рахунок чого саме це відбувається;
- суттєве покращення соціальних показників розвитку суспільства (збереження здоров'я населення, покращення якості життя тощо) та екологічних показників (зменшення забруднення навколишнього середовища, зменшення витрат енергії і сировини тощо).

4.2 Розрахунок витрат на здійснення науково-дослідної роботи

Витрати, пов'язані з проведенням науково-дослідної, дослідноконструкторської, конструкторсько-технологічної роботи, створенням дослідного зразка і здійсненням виробничих випробувань, під час планування, обліку і калькулювання собівартості науково-дослідної роботи групуються за такими *статтями*:

- витрати на оплату праці;
- відрахування на соціальні заходи;
- матеріали;
- паливо та енергія для науково-виробничих цілей;
- витрати на службові відрядження;
- спецустаткування для наукових (експериментальних) робіт;
- програмне забезпечення для наукових (експериментальних) робіт;

- витрати на роботи, які виконують сторонні підприємства, установи і організації;
- інші витрати;
- накладні (загальновиробничі) витрати.

4.2.1 Витрати на оплату праці

До статті «Витрати на оплату праці» належать витрати на виплату основної та додаткової заробітної плати керівникам відділів, лабораторій, секторів і груп, науковим, інженерно-технічним працівникам, конструкторам, технологам, лаборантам, робітникам, студентам, аспірантам та іншим працівникам, безпосередньо зайнятим виконанням конкретної теми, обчисленої за посадовими окладами, відрядними розцінками, тарифними ставками згідно з чинними в організаціях системами оплати праці, також будь-які види грошових і матеріальних доплат, які належать до елемента «Витрати на оплату праці».

Основна заробітна плата дослідників

Витрати на основну заробітну плату дослідників (Z_o) розраховують відповідно до посадових окладів працівників, за формулою:

$$Z_o = \sum_i = 1 \frac{k \times M_{ni} \times t_i}{T_p}, \quad (4.1)$$

де k – кількість посад дослідників, залучених до процесу досліджень;

M_{ni} – місячний посадовий оклад конкретного дослідника, грн;

t_i – кількість днів роботи конкретного дослідника, дн.;

T_p – середня кількість робочих днів в місяці, $T_p=21...23$ дні. Проведені розрахунки бажано звести до таблиці.

Оберемо кількість робочих днів в місяці 22 дні.

Таблиця 4.4 – Витрати на заробітну плату дослідників

Найменування посади	Місячний посадовий оклад, грн	Оплата за робочий день, грн	Кількість днів роботи	Витрати на заробітну плату, грн
<i>Керівник проекту</i>	38 000	1727,3	40	69092
<i>Науковий співробітник</i>	30 000	1363,6	40	54544
Всього				123636

Так як в даному випадку розробляється програмний продукт, то розробник виступає одночасно і основним робітником, і тестувальником розроблюваного програмного продукту.

Додаткова заробітна плата дослідників та робітників

Додаткова заробітна плата розраховується як 10 ... 12% від суми основної заробітної плати дослідників та робітників за формулою:

$$Z_d = (Z_o + Z_p) \times \frac{H_{\text{дод}}}{100\%}, \quad (4.2)$$

де $H_{\text{дод}}$ – норма нарахування додаткової заробітної плати.

$$Z_d = (123636 * 12 \% / 100 \%) = 14\ 836,3 \text{ (грн.)}$$

4.2.2 Відрахування на соціальні заходи

До статті «Відрахування на соціальні заходи» належать відрахування внеску на загальнообов'язкове державне соціальне страхування та для здійснення заходів щодо соціального захисту населення (ЄСВ – єдиний соціальний внесок).

Нарахування на заробітну плату дослідників та робітників розраховується як 22% від суми основної та додаткової заробітної плати дослідників і робітників за формулою:

$$H_3 = (Z_o + Z_p + Z_{\text{дод}}) \times \frac{H_{\text{зп}}}{100\%}, \quad (4.3)$$

де $H_{\text{зп}}$ – норма нарахування на заробітну плату.

$$H_3 = (123636 + 14\,836,3) \cdot 22\% / 100\% = 30463,91 \text{ (грн.)}$$

4.2.3 Амортизація обладнання, програмних засобів та приміщень

До статті «Амортизація обладнання, програмних засобів та приміщень» відносять амортизаційні відрахування по кожному виду обладнання, устаткування та інших приладів і пристроїв, а також програмного забезпечення для проведення науково-дослідної роботи, за його наявності в дослідній організації або на підприємстві.

В спрощеному вигляді амортизаційні відрахування по кожному виду обладнання, приміщень та програмному забезпеченню тощо можуть бути розраховані з використанням прямолінійного методу амортизації за формулою:

$$A_{\text{обл}} = \frac{Ц_б}{T_в} \times \frac{T_{\text{вик}}}{12}, \quad (4.4)$$

$$A_{\text{обл}} = \frac{40\,000}{2} \times \frac{1,5}{12} = 2500$$

де $Ц_б$ – балансова вартість обладнання, програмних засобів, приміщень тощо, які використовувались для проведення досліджень, грн;

$t_{\text{вик}}$ – термін використання обладнання, програмних засобів, приміщень під час досліджень, місяців;

$T_в$ – строк корисного використання обладнання, програмних засобів, приміщень тощо, років.

Проведені розрахунки необхідно звести до таблиці 4.5.

Таблиця 4.5 – Амортизаційні відрахування по кожному виду обладнання

Найменування обладнання	Балансова вартість, грн	Строк корисного використання, років	Термін використання обладнання, місяців	Амортизаційні відрахування, грн
Комп'ютер (Ноутбук ASUS ROG Strix G15)	39999	2	1,82	3033,26
Приміщення	750000	20	1,82	5687,5
Офісне обладнання (стіл 3500, крісло 3750)	7250	5	1,82	219,91
Ліцензійна ОС, та спеціалізовані ліцензійні нематеріальні ресурси (Intellij Idea – 16467 грн., MySQL Workbench – 58500 грн.)	74967	10	1,82	1137
Всього				10 077,67

4.2.4 Паливо та енергія для науково-виробничих цілей

До статті «Паливо та енергія для науково-виробничих цілей» належать витрати на придбання у сторонніх підприємств, установ і організацій будь-якого палива, що витрачається з технологічною метою на проведення досліджень. Стаття формується у разі виконання енергоємних наукових досліджень за методом прямого внесення витрат і досягає значної питомої ваги у собівартості досліджень.

Витрати на силову електроенергію (B_e) розраховують за формулою:

$$B_e = \sum_{\eta_{i=1}} \frac{W_{yi} \times t_i \times \text{Ц}_e \times K_{\text{впі}}}{\eta_{i=1}}, \quad (4.5)$$

$$B_e = \sum_{\eta_{i=1}} \frac{0,096 \times 320 \times 6,10 \times 0,9}{\eta_{i=1}} = 450,53,$$

$$B_e = \sum_{\eta_{i=1}} \frac{0,036 \times 320 \times 6,1 \times 0,9}{\eta_{i=1}} = 168,95$$

$$B_e = \sum_{\eta_{i=1}} \frac{0,024 \times 320 \times 6,1 \times 0,9}{\eta_{i=1}} = 37,54,$$

де W_{yi} – встановлена потужність обладнання на певному етапі розробки, кВт; t_i – тривалість роботи обладнання на етапі дослідження, год;

C_e – вартість 1 кВт-години електроенергії, грн; Тарифи на розподіл електроенергії для всіх енергорозподільних компаній встановлює Національна комісія з регулювання енергетики і комунальних послуг (НКРЕКП), вартість для юридичних осіб $B = 6,10$ грн/кВт;

K_{eni} – коефіцієнт, що враховує використання потужності, $K_{eni} < 1$; η_i – коефіцієнт корисної дії обладнання, $\eta_i < 1$.

Проведені розрахунки необхідно звести до таблиці.

Таблиця 2.14 – Витрати на електроенергію

Найменування обладнання	Встановлена потужність, кВт	Тривалість роботи, год	Сума, грн
Ноутбук ASUS ROG Strix G15	0.096	320	168,65
Монітор (DELL P2219H)	0.036	320	63,24
Освітлення	0.024	320	42,16
Всього			274,05

4.2.5 Інші витрати

До статті «Інші витрати» належать витрати, які не знайшли відображення у зазначених статтях витрат і можуть бути віднесені безпосередньо на собівартість досліджень за прямими ознаками.

Витрати за статтею «Інші витрати» розраховуються як 50...100% від суми основної заробітної плати дослідників та робітників за формулою:

$$I_B = (Z_o + Z_p) \times \frac{H_{iB}}{100\%}, \quad (4.7)$$

де H_{iB} – норма нарахування за статтею «Інші витрати».

$$I_B = 123636 * 60\% / 100\% = 74181,6 \text{ (грн.)}$$

4.2.6 Накладні

До статті «Накладні (загальновиробничі) витрати» належать: витрати, пов'язані з управлінням організацією; витрати на винахідництво та раціоналізацію; витрати на підготовку (перепідготовку) та навчання кадрів; витрати, пов'язані з набором робочої сили; витрати на оплату послуг банків; витрати, пов'язані з освоєнням виробництва продукції; витрати на науково-технічну інформацію та рекламу та ін.

Витрати за статтею «Накладні (загальновиробничі) витрати» розраховуються як 100...150% від суми основної заробітної плати дослідників та робітників за формулою:

$$B_{\text{нзв}} = (Z_o + Z_p) \times \frac{H_{\text{нзв}}}{100\%}, \quad (4.8)$$

де $H_{\text{нзв}}$ – норма нарахування за статтею «Накладні (загальновиробничі) витрати».

$$H_{\text{нзв}} = 123\,636 * 120\% / 100\% = 148\,363,2 \text{ (грн.)}$$

Витрати на проведення науково-дослідної роботи розраховуються як сума всіх попередніх статей витрат за формулою:

$$B_{\text{заг}} = Z_o + Z_p + Z_{\text{дод}} + Z_n + M + K_v + B_{\text{спец}} + B_{\text{прг}} + A_{\text{обл}} + B_e + B_{\text{св}} + B_{\text{сн}} + I_v + B_{\text{нзв}}. \quad (4.9)$$

$$B_{\text{заг}} = 123\,636 + 14\,836,3 + 30\,463,91 + 10\,077,67 + 274,05 + 74\,181,6 + 148\,363,2 = 401\,832,73 \text{ грн.}$$

Загальні витрати ZB на завершення науково-дослідної (науковотехнічної) роботи та оформлення її результатів розраховуються за формулою:

$$ЗВ = \frac{В_{заг}}{\eta}, \quad (4.10)$$

де η – коефіцієнт, який характеризує етап (стадію) виконання науководослідної роботи. Так, якщо науково-технічна розробка знаходиться на стадії: науково-дослідних робіт, то $\eta=0,1$; технічного проектування, то $\eta =0,2$; розробки конструкторської документації, то $\eta=0,3$; розробки технологій, то $\eta=0,4$; розробки дослідного зразка, то $\eta=0,5$; розробки промислового зразка, то $\eta=0,7$; впровадження, то $\eta=0,9$.

$$ЗВ = 401\,832,73 / 0,5 = 803\,665,46 \text{ грн.}$$

4.3 Розрахунок економічної ефективності науково-технічної розробки за її можливої комерціалізації потенційним інвестором

В ринкових умовах узагальнювальним позитивним результатом, що його може отримати потенційний інвестор від можливого впровадження результатів цієї чи іншої науково-технічної розробки, є збільшення у потенційного інвестора величини чистого прибутку. Саме зростання чистого прибутку забезпечить потенційному інвестору надходження додаткових коштів, дозволить покращити фінансові результати його діяльності, підвищить конкурентоспроможність та може позитивно вплинути на ухвалення рішення щодо комерціалізації цієї розробки.

Для того, щоб розрахувати можливе зростання чистого прибутку у потенційного інвестора від можливого впровадження науково-технічної розробки необхідно:

а) вказати, з якого часу можуть бути впроваджені результати науково-технічної розробки;

б) зазначити, протягом скількох років після впровадження цієї науково-технічної розробки очікуються основні позитивні результати для потенційного інвестора (наприклад, протягом 4-х років після її впровадження);

в) кількісно оцінити величину існуючого та майбутнього попиту на цю або аналогічні чи подібні науково-технічні розробки та назвати основних суб'єктів (зацікавлених осіб) цього попиту;

г) визначити ціну реалізації на ринку науково-технічних розробок з аналогічними чи подібними функціями.

При розрахунку економічної ефективності потрібно обов'язково враховувати зміну вартості грошей у часі, оскільки від вкладення інвестицій до отримання прибутку минає чимало часу.

При оцінюванні ефективності інноваційних проектів передбачається розрахунок таких важливих показників:

- ✓ абсолютного економічного ефекту (чистого дисконтованого доходу);
- ✓ внутрішньої економічної дохідності (внутрішньої норми дохідності);
- ✓ терміну окупності (дисконтованого терміну окупності).

Аналізуючи напрямки проведення науково-технічних розробок, розрахунок економічної ефективності науково-технічної розробки за її можливої комерціалізації потенційним інвестором можна об'єднати, враховуючи визначені ситуації з відповідними умовами.

4.3.1 Розробка чи суттєве вдосконалення інформаційної технології для використання масовим споживачем

В цьому випадку майбутній економічний ефект буде формуватися на основі таких даних:

$$\Delta\Pi_i = (\pm\Delta C_0 \cdot N + C_0 \cdot \pm\Delta N)_i \cdot \lambda \cdot \rho \cdot \left(1 - \frac{\vartheta}{100}\right), \quad (4.11)$$

де $\pm\Delta C_0$ – зміна вартості програмного продукту (зростання чи зниження) від впровадження результатів науково-технічної розробки в аналізовані періоди часу;

N – кількість споживачів які використовували аналогічний продукт у році до впровадження результатів нової науково-технічної розробки;

C_o – основний оціночний показник, який визначає діяльність підприємства у даному році після впровадження результатів наукової розробки, $C_o = C_o \pm \Delta C_o$;

C_b – вартість програмного продукту у році до впровадження результатів розробки;

ΔN – збільшення кількості споживачів продукту, в аналізовані періоди часу, від покращення його певних характеристик;

λ – коефіцієнт, який враховує сплату податку на додану вартість. Ставка податку на додану вартість дорівнює 20%, а коефіцієнт $\lambda = 0,8333$.

p – коефіцієнт, який враховує рентабельність продукту;

ϑ – ставка податку на прибуток, у 2022 році $\vartheta = 18\%$.

Припустимо, що при прогнозованій ціні 4000 грн. за програмний продукт, термін збільшення прибутку складе 3 роки. Після завершення розробки і її вдосконалення, можна буде підняти її ціну на 400 грн. Кількість проданих підписок на програмний продукт також збільшиться: протягом першого року – на 2000 шт., протягом другого року – на 2500 шт., протягом третього року на 3000 шт. До моменту впровадження результатів наукової розробки реалізації продукту не було:

$$\Delta\Pi_1 = (0*400 + (4000 + 400)*2000)*0,8333*0,3) * (1 - 0,18) = 1\,803\,927,8 \text{ грн.}$$

$$\Delta\Pi_2 = (0*400 + (4000 + 400)*(2000 + 2500)*0,8333*0,3) * (1 - 0,18) = 4\,058\,837,64 \text{ грн.}$$

$$\Delta\Pi_3 = (0*400 + (4000 + 400)*(2000 + 2500 + 3000)*0,8333*0,3) * (1 - 0,18) = 6\,764\,729 \text{ грн.}$$

Отже, комерційний ефект від реалізації результатів розробки за три роки складе 12 627 495 грн.

Розраховуємо приведену вартість збільшення всіх чистих прибутків ПП, що їх може отримати потенційний інвестор від можливого впровадження та комерціалізації науково-дослідної розробки:

$$ПП = \sum_1^T \frac{\Delta\Pi_i}{(1+\tau)^t}, \quad (4.12)$$

де $\Delta\Pi_i$ збільшення чистого прибутку у кожному із років, протягом яких виявляються результати виконаної та впровадженої науково-дослідної роботи, грн;

T – період часу, протягом якого виявляються результати впровадженої науково-дослідної роботи, роки;

τ – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні, $\tau = 0,05 \dots 0,15$;

t – період часу (в роках).

Збільшення прибутку ми отримаємо починаючи з першого року:

$$ПП = (1\,803\,927,8 / (1 + 0,1)^1) + (4\,058\,837,64 / (1 + 0,1)^2) + (6\,764\,729 / (1 + 0,1)^3) = 1\,639\,934,4 + 3\,354\,411,27 + 5\,082\,441 = 10\,076\,78,67 \text{ грн.}$$

Далі розраховують величину початкових інвестицій PV , які потенційний інвестор має вкласти для впровадження і комерціалізації науково-технічної розробки. Для цього можна використати формулу:

$$PV = k_{\text{інв}} \times ЗВ, \quad (4.13)$$

де $k_{\text{інв}}$ – коефіцієнт, що враховує витрати інвестора на впровадження науково-технічної розробки та її комерціалізацію. Це можуть бути витрати на підготовку приміщень, розробку технологій, навчання персоналу, маркетингові заходи тощо; зазвичай $k_{\text{інв}} = 2 \dots 5$, але може бути і більшим;

$ЗВ$ – загальні витрати на проведення науково-технічної розробки та оформлення її результатів, грн.

$$PV = 2 * 800525,24 = 1\,601\,050,5 \text{ грн.}$$

Тоді абсолютний економічний ефект $E_{абс}$ або чистий приведений дохід (NPV, Net Present Value) для потенційного інвестора від можливого впровадження та комерціалізації науково-технічної розробки становитиме:

$$E_{абс} = ПП - PV, \quad (4.14)$$

$$E_{абс} = 10\,076\,786,67 - 1\,601\,050,5 = 8\,475\,736,2 \text{ грн.}$$

Оскільки $E_{абс} > 0$ то вкладання коштів на виконання та впровадження результатів даної науково-дослідної (науково-технічної) роботи може бути доцільним.

Для остаточного прийняття рішення з цього питання необхідно розрахувати внутрішню економічну дохідність або показник внутрішньої норми дохідності (IRR, Internal Rate of Return) вкладених інвестицій та порівняти її з так званою бар'єрною ставкою дисконтування, яка визначає ту мінімальну внутрішню економічну дохідність, нижче якої інвестиції в будь-яку науково-технічну розробку вкладати буде економічно недоцільно.

Розрахуємо відносну (щорічну) ефективність вкладених в наукову розробку інвестицій E_B . Для цього використаємо формулу:

$$E_B = \sqrt[T_{ж}]{1 + \frac{E_{абс}}{PV}} - 1, \quad (4.15)$$

$T_{ж}$ життєвий цикл наукової розробки, роки.

$$E_B = \sqrt[3]{(1 + 8\,475\,736,2 \div 1\,601\,050,5)} - 1 = 0,846$$

Визначимо мінімальну ставку дисконтування, яка у загальному вигляді визначається за формулою:

$$\tau = d + f, \quad (4.16)$$

де d – середньозважена ставка за депозитними операціями в комерційних банках; в 2022 році в Україні $d = (0,23 \dots 0,25)$;

f – показник, що характеризує ризикованість вкладень; зазвичай, величина $f = (0,05 \dots 0,5)$.

$$\tau_{min} = 0,25 + 0,05 = 0,29.$$

Так як $E_B > \tau_{min}$, то інвестор може бути зацікавлений у фінансуванні даної наукової розробки.

Розрахуємо термін окупності вкладених у реалізацію наукового проекту інвестицій за формулою:

$$T_{ок} = \frac{1}{E_B}, \quad (4.17)$$

$$T_{ок} = 1 / 0,846 = 1,18 \text{ р.}$$

Оскільки < 3 -х років, а саме термін окупності рівний 1,18 роки, то фінансування даної наукової розробки є доцільним.

4.4 Висновок

Економічна частина даної роботи містить розрахунок витрат на розробку інформаційної технології прогнозування обсягу закупки квітів, сума яких складає 803 665,46 гривень. Було спрогнозовано орієнтовану величину витрат по кожній з статей витрат. Також розраховано чистий прибуток, який може отримати виробник від реалізації нового технічного рішення та економічний ефект при використанні даної розробки. В результаті аналізу розрахунків можна зробити висновок, що розроблений програмний продукт за ціною дешевший за аналог і є високо конкурентоспроможним.

Висновки

У ході виконання магістерської кваліфікаційної роботи реалізовано інформаційну технологію прогнозування обсягу закупівлі квітів. Для досягнення мети були поставлені такі завдання: провести аналіз програм аналогів, систем-аналогів та обґрунтувати доцільність розробки інформаційної технології прогнозування обсягу закупівлі квітів; розробити математичну модель прогнозування обсягу закупівлі квітів; розробити структуру інформаційної технології прогнозування обсягу закупівлі квітів; здійснити програмну реалізацію інформаційної технології прогнозування обсягу закупівлі квітів; провести тестування програми та проаналізувати отримані результати; економічно обґрунтувати доцільність розробки інформаційної технології прогнозування обсягу закупівлі квітів, які виконані в повному обсязі.

Обґрунтовано доцільність розробки інформаційної технології прогнозування обсягу закупівлі квітів. Проведено аналіз сучасних аналогів та систем-аналогів, які виконують функції організації системи закупівель та управління товарами на складі. Виявлено проблему відсутності інтуїтивно зрозумілого інтерфейсу, високої ціни за користування додатком, а також недостатньо високу точність прогнозування для такого товару як зрізані квіти, адже навіть незначні похибки можуть призвести до великих втрат, що довело актуальність розробки інформаційної технології прогнозування обсягу закупівлі квітів.

На основі аналізу існуючих моделей та методів, що застосовуються для вирішення поставленої задачі було обґрунтовано використання в інформаційній технології алгоритмів прогнозування часових рядів, поєднавши методи ковзного середнього та експонентного згладжування для сезонних та несезонних квітів відповідно, які увійшли в розроблену математичну модель прогнозування обсягу закупівлі квітів, що забезпечить підвищення точності прогнозування.

Було розроблено структуру інформаційної технології прогнозування обсягу закупівлі квітів. Розроблено алгоритми функціонування інформаційної технології.

Здійснено проектування та розробку інформаційної технології прогнозування обсягу закупівлі квітів. Обґрунтовано вибір мови програмування. Розробка інформаційної технології велася на об'єктно-орієнтованій мові програмування Java в середовищі розробки IntelliJ IDEA. Здійснено програмну реалізацію інформаційної технології прогнозування обсягу закупівлі квітів, детально описано класи та функції, які використовуються в інформаційній технології.

Проведено тестування роботи інформаційної технології прогнозування обсягу закупівлі квітів та проаналізовано її результати. Тестування програми підтвердило очікувані результати та правильність роботи програми, довівши підвищення точності прогнозування на 3,11%.

Здійснено економічне обґрунтування доцільності розробки інформаційної технології прогнозування обсягу закупівлі квітів. Було спрогнозовано орієнтовану величину витрат по кожній з статей витрат. Також розраховано чистий прибуток, який може отримати виробник від реалізації нового технічного рішення, знайдено термін окупності витрат для виробника та економічний ефект для споживача при використанні такої розробки. В результаті аналізу розрахунків можна зробити висновок, що розробка у виробництві та використанні дешевша за аналог і є висококонкурентоспроможною. Період окупності складе близько 1,18 роки. Загальні витрати становлять 803 665,46 грн.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Климчук А.М., Мельник О.Д. «Software module of accounting and forecasting for solving the problem of purchasing flowers» в Матеріали конференції «LI Науково-технічна конференція факультету будівництва, цивільної та екологічної інженерії (2022)», Вінниця, 2022. [Електронний ресурс]. Режим доступу: <https://conferences.vntu.edu.ua/index.php/all-fbtegr/all-fbtegr-2022/author/submission/16067>. Дата звернення: травень 2022.
2. Квітковий бізнес в Україні. [Електронний ресурс]. Режим доступу: https://bankchart.com.ua/biznes/biznes_start/statti/kvitkoviy_biznes_v_ukrayin_i_z_chogo_pochati_i_yak_vidkriti_kvitkoviy_magazin. Дата звернення: травень 2022.
3. Антонова О. А. Ринок квітів та декоративних рослин в Україні. Проект. К.: Кристал, 2019. 26 с.
4. Глівенко С. В. Економічне прогнозування : навч посіб. / С. В. Глівенко, М. О. Соколов, О.М. Теліженко. – Суми : Університетська книга, 2001. 207с.
5. Дудар Т. Г. Аналіз стану маркетингу органічної продукції та шляхи розвитку каналів її збуту на агропродовольчому ринку. Вісник Хмельницького національного університету, 2009, №5. С. 66-70.
6. Савицька Н.Л. Управління попитом: навч.-метод посібник / Н.Л. Савицька, О.М. Прядко. – Харків : ХДУХТ, 2016. 197 с.
7. Лігоненко Л.О., Гуляєва Н.М., Гриню Н.А., Ситник Г.В., Докієнко Л.М. Фінанси підприємства: Підручник/Київський національний торговельно-економічний ун-т. – К., 2007. – 491 с.
8. Бідюк П. І., Половцев О. В. Аналіз та моделювання економічних процесів перехідного періоду. – Київ: НТУУ «КПІ», 1999. – 230 с.
9. Бідюк П. І, Гожий О. П. Ймовірно-статистичні методи моделювання і прогнозування : монографія. Миколаїв : ЧДУ ім. П. Могили, 2014, 440 с.

10. Кузмішин П. Підприємницьке середовище як фактор потужності і конкурентоспроможності економіки та регіонів. Науковий вісник Ужгородського університету. Серія Економіка, Спецвипуск 22. Частина 3, 2007. С. 12-21.
11. Бідюк П. І. Системний підхід до прогнозування на основі моделей часових рядів. Системні дослідження та інформаційні технології. 2003. №3. С. 88 – 110.
12. Грін В. Г. Економетричний аналіз : навч. посіб Київ: Основи, 2005. 1198 с.
13. Бідюк П.І., Савенков О.І., Баклан І.В. Часові ряди: моделювання і прогнозування : монографія. Київ: ЕКМО, 2003. 144 с.
14. Лук'яненко І. Г., Городніченко Ю. О. Сучасні економічні методи у фінансах : навч. посіб. Київ : Літера ЛТД, 2002. 352 с.
15. Томашевський О. В., Рисіков В. П. Комп'ютерні технології статистичної обробки даних : навч. посіб. Запоріжжя : ЗНТУ, 2015. 175 с.
16. Цеслів О. В. Коломієць А. С. Технологія проектування та адміністрування баз даних та сховищ даних : навч. посіб. Київ : КПІ ім. І. Сікорського, 2017. 290 с.
17. Forecast NOW – [Електронний ресурс]. – Режим доступу: <https://fnow.ru/>
Дата звернення: травень 2022.
18. Облачная система учета продаж, товаров и клиентов – [Електронний ресурс].
– Режим доступу: <https://www.ekam.ru/>. Дата звернення: травень 2022.
19. E1-Закупівлі – [Електронний ресурс]. – Режим доступу:
<https://e1zakupki.ru/>. Дата звернення: травень 2022.
20. Автоматизация учета в магазине [Електронний ресурс] – режим доступа:
<https://class365.com.ua/>. Дата звернення: травень 2022.
21. Aaker, David A. and Day George S. Marketing Research. — 4th ed. — New York: John Wiley and Sons, 1990. — Chapter 22 «Forecasting».
22. Kress G.J., Shyder J. Forecasting and Market Analysis Techniques: A Practical Approach. — Hardcover, 1994.

23. Wheelwright, S. and Makridakis, S. *Forecasting Methods for Management*. — 4th ed. — John Wiley & Sons, Canada, 1985.
24. Кириленко О. П. *Планування та управління фінансовими ресурсами територіальної громади* / О.П. Кириленко, Б.С. Малиняк, В.В. Письменний, В.М. Русін / Асоціація міст України : навч. посіб. Київ, ТОВ «ВІ ЕН ЕЙ», 2015. 396 с.
25. Приймак В. М. *Прийняття управлінських рішень* : навч. посіб. Київ, Атіка, 2008. 241 с.
26. Мулеса О. Ю. *Основи мови запитів SQL* : навч. посіб. Ужгород, 2015. 48 с.
27. Кадомський К. К., Ніколюк П. К. *Java. Теорія і практика* : навч. посіб. Вінниця : Донну, 2019. 197 с.
28. Копитко М. Ф., Іванків К. С. *Основи програмування мовою Java* : тексти лекцій. Львів : ЛНУ ім. Івана Франка, 2002. 83с.
29. Яковенко А. В. *Основи програмування. Python. Частина 1* : навч. посіб. Київ : КПІ ім. І. Сікорського, 2018. 195 с.
30. Юрченко І. В., Сікора В. С. *Програмування мовою Python* : навч. посіб. Чернівці : ЧНУ, 2022. 104 с.
31. IntelliJ Idea – середовище розробки для Java. [Електронний ресурс]. – Режим доступу: <https://ualinux.com/uk/ubuntu-apps-programming/idea-ic>
Дата звернення: серпень 2022.
32. *Java-програмування: комп'ютерний практикум* : навч. посіб. Київ: КПІ ім. І. Сікорського, 2021. 95 с.
33. *Java Swing Project: Properties, Advantages & Frameworks*. [Електронний ресурс]. – Режим доступу: <https://www.upgrad.com/blog/java-swing-project/>
Дата звернення: серпень 2022.
34. Рисований О. М. *Системне програмування. Графічний інтерфейс користувача GUI* : навч. посіб. Харків : ХПІ, 2018. 160 с.
35. *Java JDBC*. [Електронний ресурс]. – Режим доступу: <https://www.javatpoint.com/java-jdbc>. Дата звернення: вересень 2022.

- 36.Методичні вказівки до виконання економічної частини магістерських кваліфікаційних робіт. [Електронний ресурс]. – Режим доступу: https://epvm.vntu.edu.ua/wp-content/uploads/2021/09/Ekonom_magister_tehn_2021.pdf Дата звернення: вересень 2022.
- 37.Розрахунок тарифів на електроенергію в навчальних роботах [Електронний ресурс]. – Режим доступу: <https://epvmvntu.wordpress.com/2022/09/16/> Дата звернення: вересень 2022.
- 38.Економіка підприємства. [Електронний ресурс]. – Режим доступу: https://web.posibnyky.vntu.edu.ua/fmib/24nikiforova_ekonomika_pidpriyemstva/ Дата звернення: вересень 2022.

Результат перевірки на плагіат в онлайн-системі UNICHECK



Імя користувача:
Озеранський В.С. КН

ID перевірки:
1013270014

Дата перевірки:
11.12.2022 19:12:57 EET

Тип перевірки:
Doc vs Internet + Library

Дата звіту:
11.12.2022 19:14:02 EET

ID користувача:
62038

Назва файлу: 122МКР-КлимчукАМ2022

Кількість сторінок: 63 Кількість слів: 10424 Кількість символів: 81448 Розмір файлу: 1.07 MB ID файлу: 1013028796

Обнаружены модификации текста (могут влиять на процент совпадений)

10.7% Совпадения

Наибольшее совпадение: 10.7% с источником из Библиотеки (ID файла: 1011191147)

Не найдено источников из Интернета

10.7% Источники из Библиотеки 1

Страница 65

0% Цитат

Исключение цитат выключено

Исключение списка библиографических ссылок выключено

38.5% Исключений

Некоторые источники исключены автоматически (фильтры исключения: количество найденных слов меньш...

2.87% Исключений из Интернета 22

Страница 66

36.1% Исключенного текста из Библиотеки 227

Страница 66

Модификации

Обнаружены модификации текста. Подробная информация доступна в онлайн-отчете.

Замененные символы 10

Подозрительное форматирование 11 страниц

Лістинг програми

Клас MainMenu

```

package java1.interf;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class MainMenu {
    private ImageIcon image;
    private JFrame frame;
    private JLabel label;

    public MainMenu() {
        image = new ImageIcon(getClass().getResource("1.jpg"));
    }

    public void createWindow() {
        frame = new JFrame();
        frame.setTitle("Головне меню");
        frame.setSize(400, 350);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setLocationRelativeTo(null);

        Font font1 = new Font("TimesRoman", Font.BOLD, 14);
        label = new JLabel();
        label.setLayout(new GridBagLayout());
        label.setVisible(true);
        label.setBackground(Color.BLUE);
        label.setIcon(image);

        JButton exFlowers = new JButton();
        exFlowers.setText("Показати архів");
        exFlowers.setFont(font1);
        label.add(exFlowers, new GridBagConstraints(0, 0, 1, 1, 0.0, 0.9,
GridBagConstraints.CENTER,
        GridBagConstraints.HORIZONTAL, new Insets(30, 10, 10, 10), 0, 0));
        exFlowers.addActionListener(new exFlowersListener());

        JButton newFlowers = new JButton();

```

```

newFlowers.setText("Додати інформацію");
newFlowers.setFont(font1);
label.add(newFlowers, new GridBagConstraints(0, 1, 1, 1, 0.0, 0.9,
GridBagConstraints.CENTER,
    GridBagConstraints.HORIZONTAL, new Insets(30, 10, 10, 10), 0, 0));
newFlowers.addActionListener(new newFlowersListener());

JButton newForecast = new JButton();
newForecast.setText("Прогнозування");
newForecast.setFont(font1);
label.add(newForecast, new GridBagConstraints(0, 2, 1, 1, 0.0, 0.9,
GridBagConstraints.CENTER,
    GridBagConstraints.HORIZONTAL, new Insets(30, 10, 10, 10), 0, 0));
newForecast.addActionListener(new newForecastListener());

frame.add(label);
frame.setVisible(true);
}

public void closeWindow() {
    frame.setVisible(false);
    frame.remove(label);
}

private class newFlowersListener implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent JCom) {
        closeWindow();
        new PageForNewFlowers().createWindow();
    }
}

private class exFlowersListener implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent JCom) {
        closeWindow();
        new PageShowOldFlower().createWindow();
    }
}

private class newForecastListener implements ActionListener {

    @Override

```

```

    public void actionPerformed(ActionEvent JCom) {
        closeWindow();
        new PageForNewForecast().createWindow();
    }
}
}

```

Клас Flower

```
package db;
```

```

public class Flower {
    private int id;
    private String date;
    private int roseRedLong;
    private int roseRedMini;
    private int roseYellow;
    private int roseWhite;
    private int rosePink;
    private int roseOrange;
    private int orchidWhite;
    private int orchidPink;
    private int orchidYellow;
    private int chrysanthemums;

```

```

    public Flower() {
    }

```

```

    public Flower(String date, int roseRedLong, int roseRedMini, int roseYellow, int
    roseWhite, int rosePink, int roseOrange, int orchidWhite, int orchidPink, int orchidYellow, int
    chrysanthemums) {

```

```

        this.date = date;
        this.roseRedLong = roseRedLong;
        this.roseRedMini = roseRedMini;
        this.roseYellow = roseYellow;
        this.roseWhite = roseWhite;
        this.rosePink = rosePink;
        this.roseOrange = roseOrange;
        this.orchidWhite = orchidWhite;
        this.orchidPink = orchidPink;
        this.orchidYellow = orchidYellow;
        this.chrysanthemums = chrysanthemums;
    }

```

```

    public void setId(int id) {
        this.id = id;
    }

```

```
}

public String getDate() {
    return date;
}

public void setDate(String date) {
    this.date = date;
}

public int getRoseRedLong() {
    return roseRedLong;
}

public void setRoseRedLong(int roseRedLong) {
    this.roseRedLong = roseRedLong;
}

public int getRoseRedMini() {
    return roseRedMini;
}

public void setRoseRedMini(int roseRedMini) {
    this.roseRedMini = roseRedMini;
}

public int getRoseYellow() {
    return roseYellow;
}

public void setRoseYellow(int roseYellow) {
    this.roseYellow = roseYellow;
}

public int getRoseWhite() {
    return roseWhite;
}

public void setRoseWhite(int roseWhite) {
    this.roseWhite = roseWhite;
}

public int getRosePink() {
    return rosePink;
}
```

```
public void setRosePink(int rosePink) {
    this.rosePink = rosePink;
}

public int getRoseOrange() {
    return roseOrange;
}

public void setRoseOrange(int roseOrange) {
    this.roseOrange = roseOrange;
}

public int getOrchidWhite() {
    return orchidWhite;
}

public void setOrchidWhite(int orchidWhite) {
    this.orchidWhite = orchidWhite;
}

public int getOrchidPink() {
    return orchidPink;
}

public void setOrchidPink(int orchidPink) {
    this.orchidPink = orchidPink;
}

public int getOrchidYellow() {
    return orchidYellow;
}

public void setOrchidYellow(int orchidYellow) {
    this.orchidYellow = orchidYellow;
}

public int getChrysanthemums() {
    return chrysanthemums;
}

public void setChrysanthemums(int chrysanthemums) {
    this.chrysanthemums = chrysanthemums;
}
}
```

Клас FlowerDAO


```

package java1.db;

import java.sql.*;
import java.util.ArrayList;

public class FlowerDAO {

    private static final String mySQL = "INSERT INTO flowers (dateS, rrl, rrm, rye, rw, rr, ro,
ow, flowers.or, oy, ch) " +
        "VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?)";
    private static final String myGetSQL = "SELECT * FROM flowers where dateS > ? and
dateS < ?";

    public void addNewFl(db.Flower flower){
        PreparedStatement pstmt = null;
        ResultSet rs = null;
        Connection con = null;
        DBManager dbManager = DBManager.getInstance();
        try {
            con = dbManager.getConnection();
            pstmt = con.prepareStatement(mySQL,
Statement.RETURN_GENERATED_KEYS);
            pstmt.setString(1, flower.getDate());
            pstmt.setInt(2, flower.getRoseRedLong());
            pstmt.setInt(3, flower.getRoseRedMini());
            pstmt.setInt(4, flower.getRoseYellow());
            pstmt.setInt(5, flower.getRoseWhite());
            pstmt.setInt(6, flower.getRosePink());
            pstmt.setInt(7, flower.getRoseOrange());
            pstmt.setInt(8, flower.getOrchidWhite());
            pstmt.setInt(9, flower.getOrchidPink());
            pstmt.setInt(10, flower.getOrchidYellow());
            pstmt.setInt(11, flower.getChrysanthemums());
            System.out.println(pstmt);
            pstmt.executeUpdate();
            rs = pstmt.getGeneratedKeys();
            if (rs.next()) {
                flower.setId(rs.getInt(1));
            }
        } catch (SQLException throwables) {
            throwables.printStackTrace();
        } finally {
            close(rs);
            close(pstmt);
            close(con);
        }
    }
}

```

```

}

public ArrayList<db.Flower> getFloverList(String arg1, String arg2){
    ArrayList<db.Flower> result = new ArrayList<>();
    PreparedStatement pstmt = null;
    ResultSet rs = null;
    Connection con = null;
    try {
        con = DBManager.getInstance().getConnection();
        pstmt = con.prepareStatement(myGetSQL);
        pstmt.setString(1, arg1);
        pstmt.setString(2, arg2);
        System.out.println(pstmt);
        rs = pstmt.executeQuery();
        while (rs.next()) {
            db.Flower flower = new db.Flower();
            flower.setDate(rs.getString("dateS"));
            flower.setRoseRedLong(rs.getInt("rrl"));
            flower.setRoseRedMini(rs.getInt("rrm"));
            flower.setRoseYellow(rs.getInt("rye"));
            flower.setRoseWhite(rs.getInt("rw"));
            flower.setRosePink(rs.getInt("rr"));
            flower.setRoseOrange(rs.getInt("ro"));
            flower.setOrchidWhite(rs.getInt("ow"));
            flower.setOrchidPink(rs.getInt("or"));
            flower.setOrchidYellow(rs.getInt("oy"));
            flower.setChrysanthemums(rs.getInt("ch"));
            result.add(flower);
        }
    } catch (SQLException throwables) {
        throwables.printStackTrace();
    } finally {
        close(rs);
        close(pstmt);
        close(con);
    }
    return result;
}

private void close(AutoCloseable forClose) {
    if (forClose != null) {
        try {
            forClose.close();
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}

```

```

    }
  }
}

```

Класс DBManager

```

package java1.db;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;

public class DBManager {

    private static DBManager instance;
    private static final String URL = "jdbc:mysql://localhost:3306/flowersdb";
    private static final String USERNAME = "root";
    private static final String PASSWORD = "root";

    public static synchronized DBManager getInstance() {
        if (instance == null)
            instance = new DBManager();
        return instance;
    }

    private DBManager() {
    }

    public Connection getConnection() throws SQLException {
        Connection con = null;
        try {
            con = DriverManager.getConnection(URL, USERNAME, PASSWORD);
        } catch (SQLException e) {
            e.printStackTrace();
        }
        return con;
    }

    public void commitAndClose(Connection con) {
        try {
            if (con != null) {
                con.commit();
                con.close();
            }
        }
    }
}

```

```

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

```

Клас PageForNewFlowers

```

package java1.interf;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

import db.Flower;
import java1.db.FlowerDAO;

public class PageForNewFlowers {
    private ImageIcon image;
    private JFrame frame;
    private JLabel label;

    private JTextField tf1, tf2, tf3, tf4, tf5, tf6, tf7, tf8, tf9, tf10, tf11;

    public PageForNewFlowers() {
        image = new ImageIcon(getClass().getResource("1.jpg"));
    }

    public void createWindow() {
        frame = new JFrame();
        frame.setTitle("Нові дані");
        frame.setSize(400, 700);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setLocationRelativeTo(null);

        Font font1 = new Font("TimesRoman", Font.BOLD, 14);
        label = new JLabel();
        label.setLayout(new GridBagLayout());
        label.setVisible(true);
        label.setBackground(Color.BLUE);
        label.setIcon(image);

        JLabel dateLabel = new JLabel();
        dateLabel.setText("Дата");
        dateLabel.setFont(font1);
    }
}

```

```

        label.add(dateLabel, new GridBagConstraints(0, 0, 1, 1, 0.0, 0.9,
GridBagConstraints.CENTER,
        GridBagConstraints.CENTER, new Insets(10, 10, 10, 10), 0, 0));

        tf1 = new JTextField();
        tf1.setFont(font1);
        label.add(tf1, new GridBagConstraints(1, 0, 1, 1, 0.0, 0.9, GridBagConstraints.CENTER,
        GridBagConstraints.HORIZONTAL, new Insets(10, 10, 10, 10), 0, 0));

        JLabel roseLabel = new JLabel();
        roseLabel.setText("Троянда червона метрова");
        roseLabel.setFont(font1);
        label.add(roseLabel, new GridBagConstraints(0, 2, 1, 1, 0.0, 0.9,
GridBagConstraints.CENTER,
        GridBagConstraints.CENTER, new Insets(10, 10, 10, 10), 0, 0));

        tf2 = new JTextField();
        tf2.setFont(font1);
        label.add(tf2, new GridBagConstraints(1, 2, 1, 1, 0.0, 0.9, GridBagConstraints.CENTER,
        GridBagConstraints.HORIZONTAL, new Insets(10, 10, 10, 10), 0, 0));

        JLabel roseLabel1 = new JLabel();
        roseLabel1.setText("Троянда червона міні");
        roseLabel1.setFont(font1);
        label.add(roseLabel1, new GridBagConstraints(0, 3, 1, 1, 0.0, 0.9,
GridBagConstraints.CENTER,
        GridBagConstraints.CENTER, new Insets(10, 10, 10, 10), 0, 0));

        tf3 = new JTextField();
        tf3.setFont(font1);
        label.add(tf3, new GridBagConstraints(1, 3, 1, 1, 0.0, 0.9, GridBagConstraints.CENTER,
        GridBagConstraints.HORIZONTAL, new Insets(10, 10, 10, 10), 0, 0));

        JLabel roseLabel2 = new JLabel();
        roseLabel2.setText("Троянда жовта");
        roseLabel2.setFont(font1);
        label.add(roseLabel2, new GridBagConstraints(0, 4, 1, 1, 0.0, 0.9,
GridBagConstraints.CENTER,
        GridBagConstraints.CENTER, new Insets(10, 10, 10, 10), 0, 0));

        tf4 = new JTextField();
        tf4.setFont(font1);
        label.add(tf4, new GridBagConstraints(1, 4, 1, 1, 0.0, 0.9, GridBagConstraints.CENTER,
        GridBagConstraints.HORIZONTAL, new Insets(10, 10, 10, 10), 0, 0));

        JLabel roseLabel3 = new JLabel();

```

```

roseLabel3.setText("Троянда біла");
roseLabel3.setFont(font1);
label.add(roseLabel3, new GridBagConstraints(0, 5, 1, 1, 0.0, 0.9,
GridBagConstraints.CENTER,
    GridBagConstraints.CENTER, new Insets(10, 10, 10, 10), 0, 0));

tf5 = new JTextField();
tf5.setFont(font1);
label.add(tf5, new GridBagConstraints(1, 5, 1, 1, 0.0, 0.9, GridBagConstraints.CENTER,
    GridBagConstraints.HORIZONTAL, new Insets(10, 10, 10, 10), 0, 0));

JLabel roseLabel4 = new JLabel();
roseLabel4.setText("Троянда рожева");
roseLabel4.setFont(font1);
label.add(roseLabel4, new GridBagConstraints(0, 6, 1, 1, 0.0, 0.9,
GridBagConstraints.CENTER,
    GridBagConstraints.CENTER, new Insets(10, 10, 10, 10), 0, 0));

tf6 = new JTextField();
tf6.setFont(font1);
label.add(tf6, new GridBagConstraints(1, 6, 1, 1, 0.0, 0.9, GridBagConstraints.CENTER,
    GridBagConstraints.HORIZONTAL, new Insets(10, 10, 10, 10), 0, 0));

JLabel roseLabel5 = new JLabel();
roseLabel5.setText("Троянда оранжева");
roseLabel5.setFont(font1);
label.add(roseLabel5, new GridBagConstraints(0, 7, 1, 1, 0.0, 0.9,
GridBagConstraints.CENTER,
    GridBagConstraints.CENTER, new Insets(10, 10, 10, 10), 0, 0));

tf7 = new JTextField();
tf7.setFont(font1);
label.add(tf7, new GridBagConstraints(1, 7, 1, 1, 0.0, 0.9, GridBagConstraints.CENTER,
    GridBagConstraints.HORIZONTAL, new Insets(10, 10, 10, 10), 0, 0));

JLabel roseLabel6 = new JLabel();
roseLabel6.setText("Орхідея біла");
roseLabel6.setFont(font1);
label.add(roseLabel6, new GridBagConstraints(0, 8, 1, 1, 0.0, 0.9,
GridBagConstraints.CENTER,
    GridBagConstraints.CENTER, new Insets(10, 10, 10, 10), 0, 0));

tf8 = new JTextField();
tf8.setFont(font1);
label.add(tf8, new GridBagConstraints(1, 8, 1, 1, 0.0, 0.9, GridBagConstraints.CENTER,
    GridBagConstraints.HORIZONTAL, new Insets(10, 10, 10, 10), 0, 0));

```

```

JLabel roseLabel7 = new JLabel();
roseLabel7.setText("Орхідея рожева");
roseLabel7.setFont(font1);
label.add(roseLabel7, new GridBagConstraints(0, 9, 1, 1, 0.0, 0.9,
GridBagConstraints.CENTER,
    GridBagConstraints.CENTER, new Insets(10, 10, 10, 10), 0, 0));

tf9 = new JTextField();
tf9.setFont(font1);
label.add(tf9, new GridBagConstraints(1, 9, 1, 1, 0.0, 0.9, GridBagConstraints.CENTER,
    GridBagConstraints.HORIZONTAL, new Insets(10, 10, 10, 10), 0, 0));

JLabel roseLabel8 = new JLabel();
roseLabel8.setText("Орхідея жовта");
roseLabel8.setFont(font1);
label.add(roseLabel8, new GridBagConstraints(0, 10, 1, 1, 0.0, 0.9,
GridBagConstraints.CENTER,
    GridBagConstraints.CENTER, new Insets(10, 10, 10, 10), 0, 0));

tf10 = new JTextField();
tf10.setFont(font1);
label.add(tf10, new GridBagConstraints(1, 10, 1, 1, 0.0, 0.9,
GridBagConstraints.CENTER,
    GridBagConstraints.HORIZONTAL, new Insets(10, 10, 10, 10), 0, 0));

JLabel roseLabel9 = new JLabel();
roseLabel9.setText("Хризантеми асорті");
roseLabel9.setFont(font1);
label.add(roseLabel9, new GridBagConstraints(0, 11, 1, 1, 0.0, 0.9,
GridBagConstraints.CENTER,
    GridBagConstraints.CENTER, new Insets(10, 10, 10, 10), 0, 0));

tf11 = new JTextField();
tf11.setFont(font1);
label.add(tf11, new GridBagConstraints(1, 11, 1, 1, 0.0, 0.9,
GridBagConstraints.CENTER,
    GridBagConstraints.HORIZONTAL, new Insets(10, 10, 10, 10), 0, 0));

JButton menuBtn = new JButton();
menuBtn.setText("Головне меню");
menuBtn.setFont(font1);
label.add(menuBtn, new GridBagConstraints(0, 12, 1, 1, 0.0, 0.9,
GridBagConstraints.CENTER,
    GridBagConstraints.HORIZONTAL, new Insets(10, 10, 10, 10), 0, 0));

```

```

        JButton addBtn = new JButton();
        addBtn.setText("Додати дані");
        addBtn.setFont(font1);
        label.add(addBtn, new GridBagConstraints(1, 12, 1, 1, 0.0, 0.9,
GridBagConstraints.CENTER,
                GridBagConstraints.HORIZONTAL, new Insets(10, 10, 10, 10), 0, 0));

        menuBtn.addActionListener(new MenuListener());
        addBtn.addActionListener(new AddListener());

        frame.add(label);
        frame.setVisible(true);

    }

    public void closeWindow() {
        frame.setVisible(false);
        frame.remove(label);
    }

    private class MenuListener implements ActionListener {

        @Override
        public void actionPerformed(ActionEvent JCom) {
            closeWindow();
            new MainMenu().createWindow();
        }
    }

    private class AddListener implements ActionListener {

        @Override
        public void actionPerformed(ActionEvent JCom) {
            closeWindow();

            Flower flower = new Flower(
                tf1.getText(),
                Integer.parseInt(tf2.getText()),
                Integer.parseInt(tf3.getText()),
                Integer.parseInt(tf4.getText()),
                Integer.parseInt(tf5.getText()),
                Integer.parseInt(tf6.getText()),
                Integer.parseInt(tf7.getText()),
                Integer.parseInt(tf8.getText()),
            );
        }
    }

```



```

        Integer.parseInt(tf9.getText()),
        Integer.parseInt(tf10.getText()),
        Integer.parseInt(tf11.getText())
    );

    new FlowerDAO().addNewFl(flower);
}
}
}

```

Клас PageShowOldFlower

```

package java1.interf;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class PageShowOldFlower {
    private ImageIcon image;
    private JFrame frame;
    private JLabel label;

    private JTextField startDateField, finishDateField;

    public PageShowOldFlower() {
        image = new ImageIcon(getClass().getResource("1.jpg"));
    }

    public void createWindow() {
        frame = new JFrame();
        frame.setTitle("Архив");
        frame.setSize(400, 200);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setLocationRelativeTo(null);

        Font font1 = new Font("TimesRoman", Font.BOLD, 14);
        label = new JLabel();
        label.setLayout(new GridBagLayout());
        label.setVisible(true);
        label.setBackground(Color.BLUE);
        label.setIcon(image);

        JLabel date1Label = new JLabel();
    }
}

```

```

date1Label.setText("Початкова дата");
date1Label.setFont(font1);
label.add(date1Label, new GridBagConstraints(0, 0, 1, 1, 0.0, 0.9,
GridBagConstraints.CENTER,
GridBagConstraints.CENTER, new Insets(10, 10, 10, 10), 0, 0));

startDateField = new JTextField();
startDateField.setFont(font1);
label.add(startDateField, new GridBagConstraints(1, 0, 1, 1, 0.0, 0.9,
GridBagConstraints.CENTER,
GridBagConstraints.HORIZONTAL, new Insets(10, 10, 10, 10), 0, 0));

JLabel date2Label = new JLabel();
date2Label.setText("Кінцева дата");
date2Label.setFont(font1);
label.add(date2Label, new GridBagConstraints(0, 1, 1, 1, 0.0, 0.9,
GridBagConstraints.CENTER,
GridBagConstraints.CENTER, new Insets(10, 10, 10, 10), 0, 0));

finishDateField = new JTextField();
finishDateField.setFont(font1);
label.add(finishDateField, new GridBagConstraints(1, 1, 1, 1, 0.0, 0.9,
GridBagConstraints.CENTER,
GridBagConstraints.HORIZONTAL, new Insets(10, 10, 10, 10), 0, 0));

JButton menuBtn = new JButton();
menuBtn.setText("ГОЛОВНЕ МЕНЮ");
menuBtn.setFont(font1);
label.add(menuBtn, new GridBagConstraints(0, 5, 1, 1, 0.0, 0.9,
GridBagConstraints.CENTER,
GridBagConstraints.HORIZONTAL, new Insets(10, 10, 10, 10), 0, 0));

JButton showBtn = new JButton();
showBtn.setText("Показати");
showBtn.setFont(font1);
label.add(showBtn, new GridBagConstraints(1, 5, 1, 1, 0.0, 0.9,
GridBagConstraints.CENTER,
GridBagConstraints.HORIZONTAL, new Insets(10, 10, 10, 10), 0, 0));

menuBtn.addActionListener(new MenuListener());
showBtn.addActionListener(new ShowListener());

frame.add(label);
frame.setVisible(true);

```

```

    }

    public void closeWindow() {
        frame.setVisible(false);
        frame.remove(label);
    }

    private class MenuListener implements ActionListener {

        @Override
        public void actionPerformed(ActionEvent JCom) {
            closeWindow();
            new MainMenu().createWindow();
        }
    }

    private class ShowListener implements ActionListener {

        @Override
        public void actionPerformed(ActionEvent JCom) {
            closeWindow();
            new TableForOldFlower(startDateField.getText(),
finishDateField.getText()).createWindow();
        }
    }
}

```

Клас TableForForecast

```

package java1.interf;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class TableForForecast {
    public static ImageIcon image;
    public static JFrame frame;
    private JLabel label;
    private String arg1, arg2;

    public TableForForecast(String arg1, String arg2) {
        this.arg1 = arg1;
        this.arg2 = arg2;
    }
}

```

```

    image = new ImageIcon(getClass().getResource("1.jpg"));
}

public void createWindow() {
    frame = new JFrame();
    frame.setTitle("Прогнозування");
    frame.setSize(1100, 600);
    frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    frame.setLocationRelativeTo(null);
    Font font = new Font("TimesRoman", Font.BOLD, 14);
    label = new JLabel();
    label.setLayout(new GridBagLayout());
    label.setVisible(true);
    label.setIcon(image);
    frame.add(label);
    frame.setVisible(true);

    TableModalForNewForecast tmq2 = new TableModalForNewForecast();
    JTable table = new JTable(tmq2);

    JScrollPane tableScrollPane = new JScrollPane(table);
    tmq2.addData(arg1, arg2);
    int shur = 0;
    if ((19 + tmq2.getRowCount() * 24) > 500) {
        shur = 500;
    } else {
        shur = 19 + tmq2.getRowCount() * 24;
    }
    tableScrollPane.setPreferredSize(new Dimension(1080, shur));
    System.out.println(tmq2.getRowCount());
    tableScrollPane.setBackground(Color.BLUE);
    table.setFont(font);
    table.setRowHeight(24);

    label.add(tableScrollPane, new GridBagConstraints(0, 0, 4, 1, 1, 1,
GridBagConstraints.NORTH,
        GridBagConstraints.HORIZONTAL, new Insets(0, 0, 0, 0), 0, 0));

    JButton backBtn = new JButton();
    backBtn.setFont(font);
    backBtn.setText("Назад");
    label.add(backBtn, new GridBagConstraints(0, 0, 2, 1, 1, 1,
GridBagConstraints.SOUTH,
        GridBagConstraints.HORIZONTAL, new Insets(0, 0, 0, 0), 0, 0));

    JButton menuBtn = new JButton();

```

```

        menuBtn.setFont(font);
        menuBtn.setText("Головне меню");
        label.add(menuBtn, new GridBagConstraints(2, 0, 2, 1, 1, 1,
GridBagConstraints.SOUTH,
        GridBagConstraints.HORIZONTAL, new Insets(0, 0, 0, 0), 0, 0));

        backBtn.addActionListener(new BackListener());
        menuBtn.addActionListener(new MenuListener());

    }

    public void closeWindow() {
        frame.setVisible(false);
        frame.remove(label);
    }
    private class BackListener implements ActionListener {

        @Override
        public void actionPerformed(ActionEvent JCom) {
            closeWindow();
            new PageForNewForecast().createWindow();
        }
    }
    private class MenuListener implements ActionListener {

        @Override
        public void actionPerformed(ActionEvent JCom) {
            closeWindow();
            new MainMenu().createWindow();
        }
    }
}

```

Клас CountForecast

```

package java1.interf;

import db.Flower;
import java.util.ArrayList;
public class CountForecast {

    public static db.Flower count(ArrayList<db.Flower> f2018, ArrayList<db.Flower> f2019,
ArrayList<db.Flower> f2020) {
        db.Flower flower = new Flower();
        for (Flower flow : f2018) {
            flower.setRoseRedLong(flower.getRoseRedLong() + flow.getRoseRedLong());
        }
    }
}

```

```

    flower.setRoseRedMini(flower.getRoseRedMini() + flow.getRoseRedMini());
    flower.setRoseYellow(flower.getRoseYellow() + flow.getRoseYellow());
    flower.setRoseWhite(flower.getRoseWhite() + flow.getRoseWhite());
    flower.setRosePink(flower.getRosePink() + flow.getRosePink());
    flower.setRoseOrange(flower.getRoseOrange() + flow.getRoseOrange());
    flower.setOrchidWhite(flower.getOrchidWhite() + flow.getOrchidWhite());
    flower.setOrchidPink(flower.getOrchidPink() + flow.getOrchidPink());
    flower.setOrchidYellow(flower.getOrchidYellow() + flow.getOrchidYellow());
    flower.setChrysanthemums(flower.getChrysanthemums()
flow.getChrysanthemums());
    }
    for (Flower flow : f2019) {
        flower.setRoseRedLong(flower.getRoseRedLong() + flow.getRoseRedLong()*2);
        flower.setRoseRedMini(flower.getRoseRedMini() + flow.getRoseRedMini()*2);
        flower.setRoseYellow(flower.getRoseYellow() + flow.getRoseYellow()*2);
        flower.setRoseWhite(flower.getRoseWhite() + flow.getRoseWhite()*2);
        flower.setRosePink(flower.getRosePink() + flow.getRosePink()*2);
        flower.setRoseOrange(flower.getRoseOrange() + flow.getRoseOrange()*2);
        flower.setOrchidWhite(flower.getOrchidWhite() + flow.getOrchidWhite()*2);
        flower.setOrchidPink(flower.getOrchidPink() + flow.getOrchidPink()*2);
        flower.setOrchidYellow(flower.getOrchidYellow() + flow.getOrchidYellow()*2);
        flower.setChrysanthemums(flower.getChrysanthemums()
flow.getChrysanthemums()*2);
    }
    for (Flower flow : f2020) {
        flower.setRoseRedLong(flower.getRoseRedLong() + flow.getRoseRedLong()*4);
        flower.setRoseRedMini(flower.getRoseRedMini() + flow.getRoseRedMini()*4);
        flower.setRoseYellow(flower.getRoseYellow() + flow.getRoseYellow()*4);
        flower.setRoseWhite(flower.getRoseWhite() + flow.getRoseWhite()*4);
        flower.setRosePink(flower.getRosePink() + flow.getRosePink()*4);
        flower.setRoseOrange(flower.getRoseOrange() + flow.getRoseOrange()*4);
        flower.setOrchidWhite(flower.getOrchidWhite() + flow.getOrchidWhite()*4);
        flower.setOrchidPink(flower.getOrchidPink() + flow.getOrchidPink()*4);
        flower.setOrchidYellow(flower.getOrchidYellow() + flow.getOrchidYellow()*4);
        flower.setChrysanthemums(flower.getChrysanthemums()
flow.getChrysanthemums()*4);
    }
    flower.setRoseRedLong(flower.getRoseRedLong() / 7);
    flower.setRoseRedMini(flower.getRoseRedMini() / 7);
    flower.setRoseYellow(flower.getRoseYellow() / 7);
    flower.setRoseWhite(flower.getRoseWhite() / 7);
    flower.setRosePink(flower.getRosePink() / 7);
    flower.setRoseOrange(flower.getRoseOrange() / 7);
    flower.setOrchidWhite(flower.getOrchidWhite() / 7);
    flower.setOrchidPink(flower.getOrchidPink() / 7);
    flower.setOrchidYellow(flower.getOrchidYellow() / 7);

```

```

        flower.setChrysanthemums(flower.getChrysanthemums() / 7);
        return flower;
    }
}

```

Клас TableModalForNewForecast

```

package java1.interf;

import java1.db.FlowerDAO;
import javax.swing.table.AbstractTableModel;
import java.util.ArrayList;

public class TableModalForNewForecast extends AbstractTableModel {
    public ArrayList<String[]> tableData;
    public int columnQ1;

    public TableModalForNewForecast() {
        tableData = new ArrayList<String[]>();
        for (int i = 0; i < tableData.size(); i++) {
            tableData.add(new String[getColumnCount()]);
        }
        this.columnQ1 = 2;
    }

    @Override
    public int getColumnCount() {
        return columnQ1;
    }

    @Override
    public int getRowCount() {
        return tableData.size();
    }

    @Override
    public Object getValueAt(int arg0, int arg1) {
        String[] rows = tableData.get(arg0);
        return rows[arg1];
    }

    @Override
    public String getColumnName(int colomnIndex) {
        switch (colomnIndex) {
            case 0:
                return "Квіти";

```

```

        case 1:
            return "Кількість";
        }
        return null;
    }

    public void addData(String[] row) {
        String[] rowTable = new String[getColumnCount()];
        rowTable = row;
        tableData.add(row);
    }

    public void addData(String arg1, String arg2) {
        StringBuilder sbS2018 = new StringBuilder("2018-");
        sbS2018.append(arg1.substring(5));
        StringBuilder sbS2019 = new StringBuilder("2019-");
        sbS2019.append(arg1.substring(5));
        StringBuilder sbS2020 = new StringBuilder("2020-");
        sbS2020.append(arg1.substring(5));

        StringBuilder sbF2018 = new StringBuilder("2018-");
        sbF2018.append(arg2.substring(5));
        StringBuilder sbF2019 = new StringBuilder("2019-");
        sbF2019.append(arg2.substring(5));
        StringBuilder sbF2020 = new StringBuilder("2020-");
        sbF2020.append(arg2.substring(5));

        ArrayList<db.Flower>          arrayList2018          =          new
        FlowerDAO().getFloverList(sbS2018.toString(), sbF2018.toString());
        ArrayList<db.Flower>          arrayList2019          =          new
        FlowerDAO().getFloverList(sbS2019.toString(), sbF2019.toString());
        ArrayList<db.Flower>          arrayList2020          =          new
        FlowerDAO().getFloverList(sbS2020.toString(), sbF2020.toString());

        db.Flower rez = CountForecast.count(arrayList2018, arrayList2019, arrayList2020);

        String[] row = new String[]{"Троянда червона метрова",
        Integer.toString(rez.getRoseRedLong())};
        addData(row);

        row = new String[]{"Троянда червона міні", Integer.toString(rez.getRoseRedMini())};
        addData(row);

        row = new String[]{"Троянда жовта", Integer.toString(rez.getRoseYellow())};
        addData(row);
    }

```



```

row = new String[]{"Троянда біла", Integer.toString(rez.getRoseWhite())};
addData(row);

row = new String[]{"Троянда рожева", Integer.toString(rez.getRosePink())};
addData(row);

row = new String[]{"Троянда оранжева", Integer.toString(rez.getRoseOrange())};
addData(row);

row = new String[]{"Орхідея біла", Integer.toString(rez.getOrchidWhite())};
addData(row);

row = new String[]{"Орхідея рожева", Integer.toString(rez.getOrchidPink())};
addData(row);

row = new String[]{"Орхідея жовта", Integer.toString(rez.getOrchidYellow())};
addData(row);

row = new String[]{"Хризантеми асорті",
Integer.toString(rez.getChrysanthemums())};
addData(row);
}
}

```

Клас TableModalForOldFlowers

```

package java1.interf;

import java1.db.FlowerDAO;

import javax.swing.table.AbstractTableModel;
import java.util.ArrayList;

public class TableModalForOldFlowers extends AbstractTableModel {
    public ArrayList<String[]> tableData;
    public int columnQ1;

    public TableModalForOldFlowers() {
        tableData = new ArrayList<String[]>();
        for (int i = 0; i < tableData.size(); i++) {
            tableData.add(new String[getColumnCount()]);
        }
        this.columnQ1 = 11;
    }

    @Override

```

```
public int getColumnCount() {
    return columnQ1;
}

@Override
public int getRowCount() {
    return tableData.size();
}

@Override
public Object getValueAt(int arg0, int arg1) {
    String[] rows = tableData.get(arg0);
    return rows[arg1];
}

@Override
public String getColumnName(int columnIndex) {
    switch (columnIndex) {
        case 0:
            return "Дата";
        case 1:
            return "Троянда червона метрова";
        case 2:
            return "Троянда червона міні";
        case 3:
            return "Троянда жовта";
        case 4:
            return "Троянда біла";
        case 5:
            return "Троянда рожева";
        case 6:
            return "Троянда оранжева";
        case 7:
            return "Орхідея біла";
        case 8:
            return "Орхідея рожева";
        case 9:
            return "Орхідея жовта";
        case 10:
            return "Хризантеми асорті";
    }
    return null;
}

public void addData(String[] row) {
    String[] rowTable = new String[getColumnCount()];
```

```

        rowTable = row;
        tableData.add(row);
    }

    public void addData(String arg1, String arg2) {
        ArrayList<db.Flower> flowerList = new FlowerDAO().getFlowerList(arg1, arg2);
        for (db.Flower flower : flowerList) {
            String[] row = new String[]{flower.getDate(),
                Integer.toString(flower.getRoseRedLong()),
                Integer.toString(flower.getRoseRedMini()),
                Integer.toString(flower.getRoseYellow()),
                Integer.toString(flower.getRoseWhite()),
                Integer.toString(flower.getRosePink()),
                Integer.toString(flower.getRoseOrange()),
                Integer.toString(flower.getOrchidWhite()),
                Integer.toString(flower.getOrchidPink()),
                Integer.toString(flower.getOrchidYellow()),
                Integer.toString(flower.getChrysanthemums())
            };
            addData(row);
        }
    }
}

```

Клас TableForOldFlower

```

package java1.interf;
import javax.swing.*.*;
import java.awt.*.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class TableForOldFlower {
    public static ImageIcon image;
    public static JFrame frame;
    private JLabel label;
    private String arg1, arg2;

    public TableForOldFlower(String arg1, String arg2) {
        this.arg1 = arg1;
        this.arg2 = arg2;
        image = new ImageIcon(getClass().getResource("1.jpg"));
    }

    public void createWindow() {
        frame = new JFrame();
        frame.setTitle("Аpxив");
        frame.setSize(1100, 600);
    }
}

```

```

frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
frame.setLocationRelativeTo(null);
Font font = new Font("TimesRoman", Font.BOLD, 14);
label = new JLabel();
label.setLayout(new GridBagLayout());
label.setVisible(true);
label.setIcon(image);
frame.add(label);
frame.setVisible(true);
TableModalForOldFlowers tmq2 = new TableModalForOldFlowers();
JTable table = new JTable(tmq2);

JScrollPane tableScrollPane = new JScrollPane(table);
tmq2.addData(arg1, arg2);
int shur = 0;
if ((19 + tmq2.getRowCount() * 24) > 500) {
    shur = 500;
} else {
    shur = 19 + tmq2.getRowCount() * 24;
}
tableScrollPane.setPreferredSize(new Dimension(1080, shur));
System.out.println(tmq2.getRowCount());
tableScrollPane.setBackground(Color.BLUE);
table.setFont(font);
table.setRowHeight(24);

label.add(tableScrollPane, new GridBagConstraints(0, 0, 4, 1, 1, 1,
GridBagConstraints.NORTH,
    GridBagConstraints.HORIZONTAL, new Insets(0, 0, 0, 0), 0, 0));
JButton backBtn = new JButton();
backBtn.setFont(font);
backBtn.setText("Назад");
label.add(backBtn, new GridBagConstraints(0, 0, 2, 1, 1, 1,
GridBagConstraints.SOUTH,
    GridBagConstraints.HORIZONTAL, new Insets(0, 0, 0, 0), 0, 0));

JButton menuBtn = new JButton();
menuBtn.setFont(font);
menuBtn.setText("ГОЛОВНЕ МЕНЮ");
label.add(menuBtn, new GridBagConstraints(2, 0, 2, 1, 1, 1,
GridBagConstraints.SOUTH,
    GridBagConstraints.HORIZONTAL, new Insets(0, 0, 0, 0), 0, 0));

backBtn.addActionListener(new BackListener());
menuBtn.addActionListener(new MenuListener());
}

```

```
public void closeWindow() {
    frame.setVisible(false);
    frame.remove(label);
}

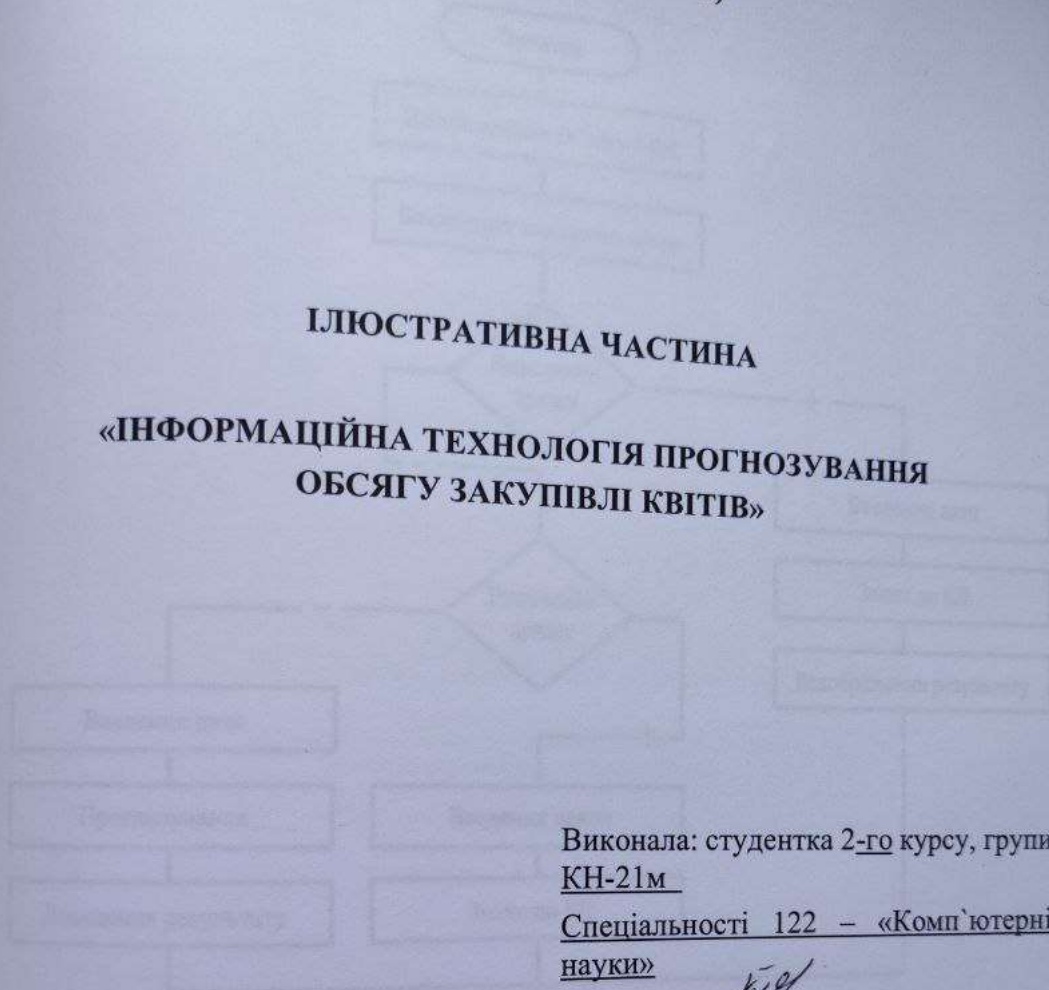
private class BackListener implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent JCom) {
        closeWindow();
        new PageShowOldFlower().createWindow();
    }
}

private class MenuListener implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent JCom) {
        closeWindow();
        new MainMenu().createWindow();
    }
}
}
```

ІЛЮСТРАТИВНА ЧАСТИНА
«ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ ПРОГНОЗУВАННЯ
ОБСЯГУ ЗАКУПІВЛІ КВІТІВ»



Виконала: студентка 2-го курсу, групи
КН-21м

Спеціальності 122 – «Комп'ютерні
науки»

Климчук А.М.

(прізвище та ініціали)

Керівник: д.т.н., ст викл. кафедри КН

Озеранський В.С.

(прізвище та ініціали)

« 15 » 12 2022 р.

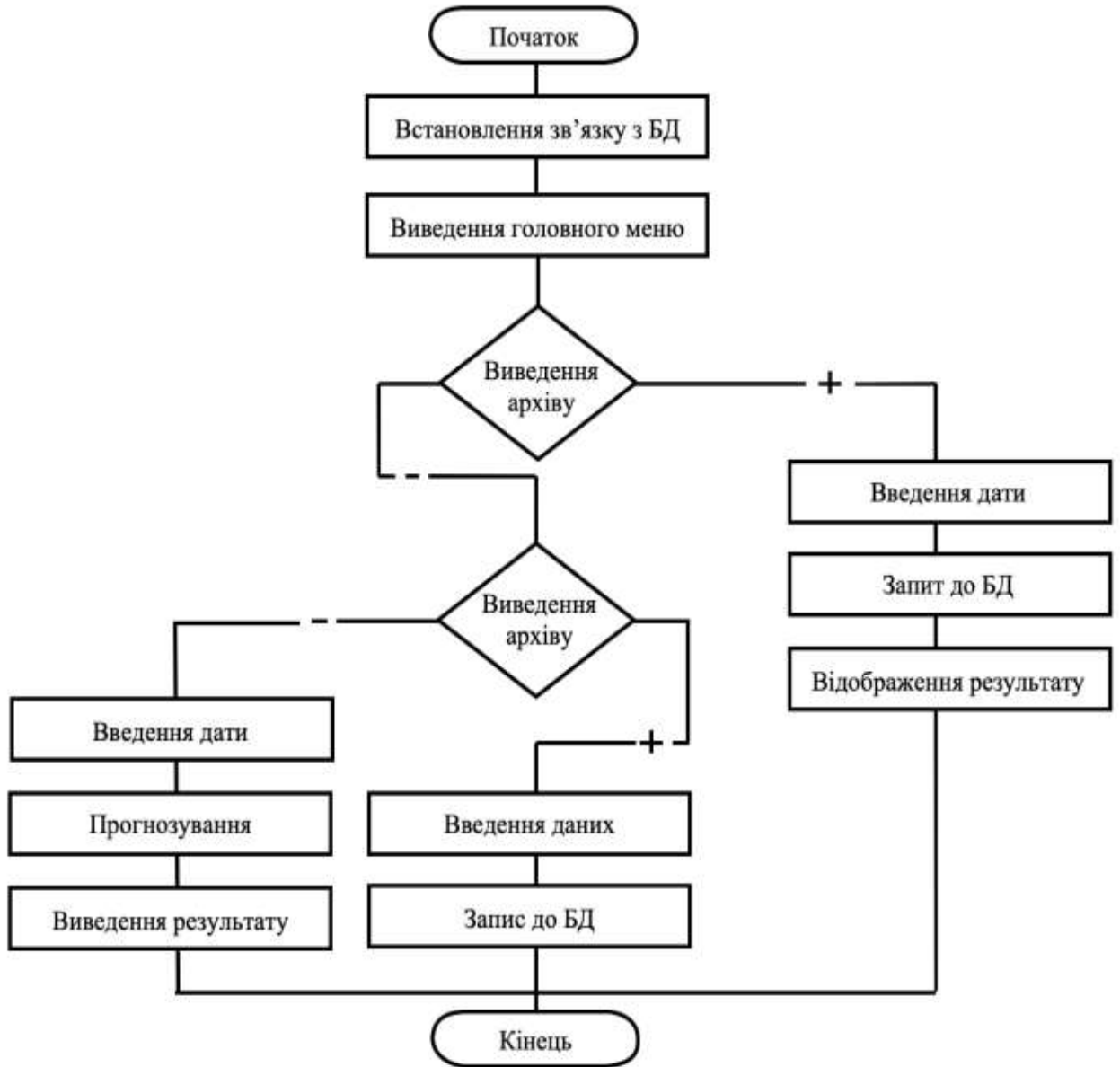


Рисунок В.1 – Схема загального алгоритму функціонування системи



Рисунок В.2 – Структурна схема інформаційної технології

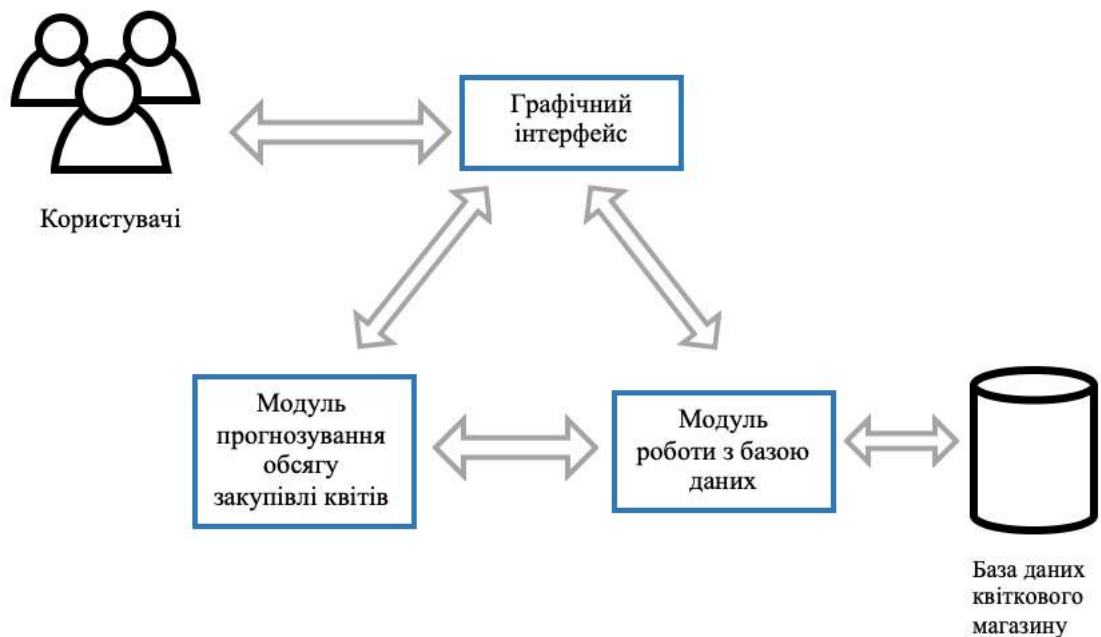


Рисунок В.3 – Модель функціонування системи прогнозування обсягу закупівлі квітів

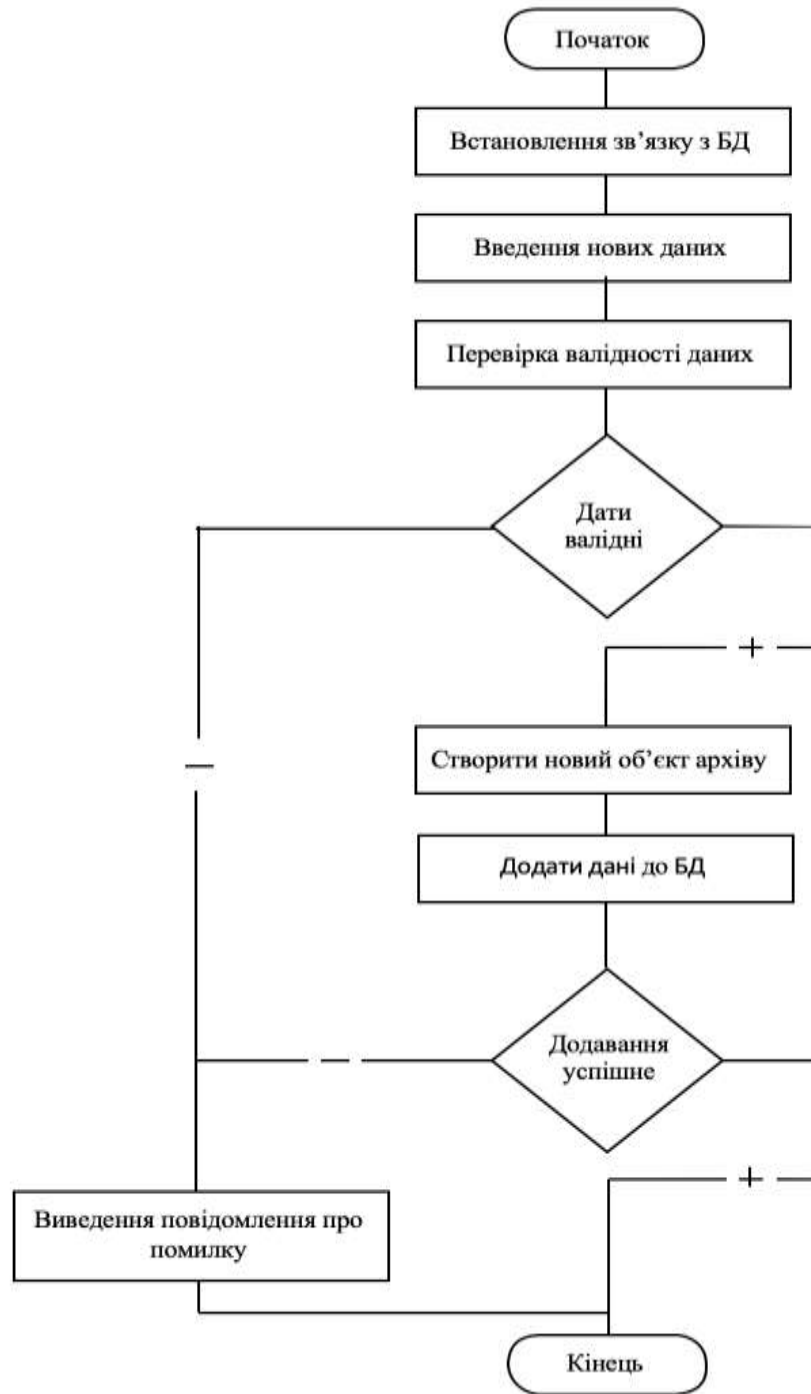


Рисунок В.4 – Схема алгоритму запису інформації

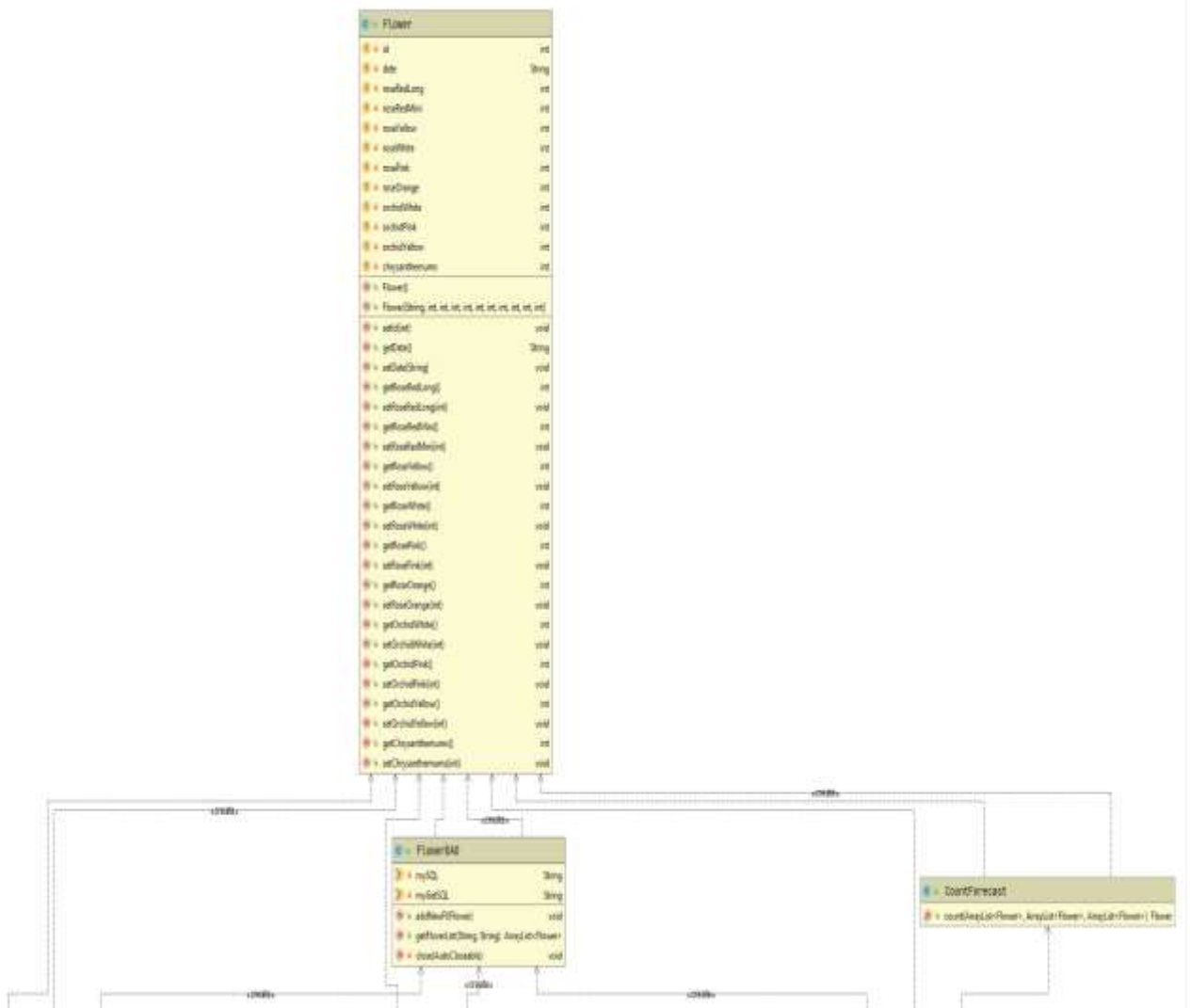


Рисунок В.5 – UML-діаграма класів (частина 1)

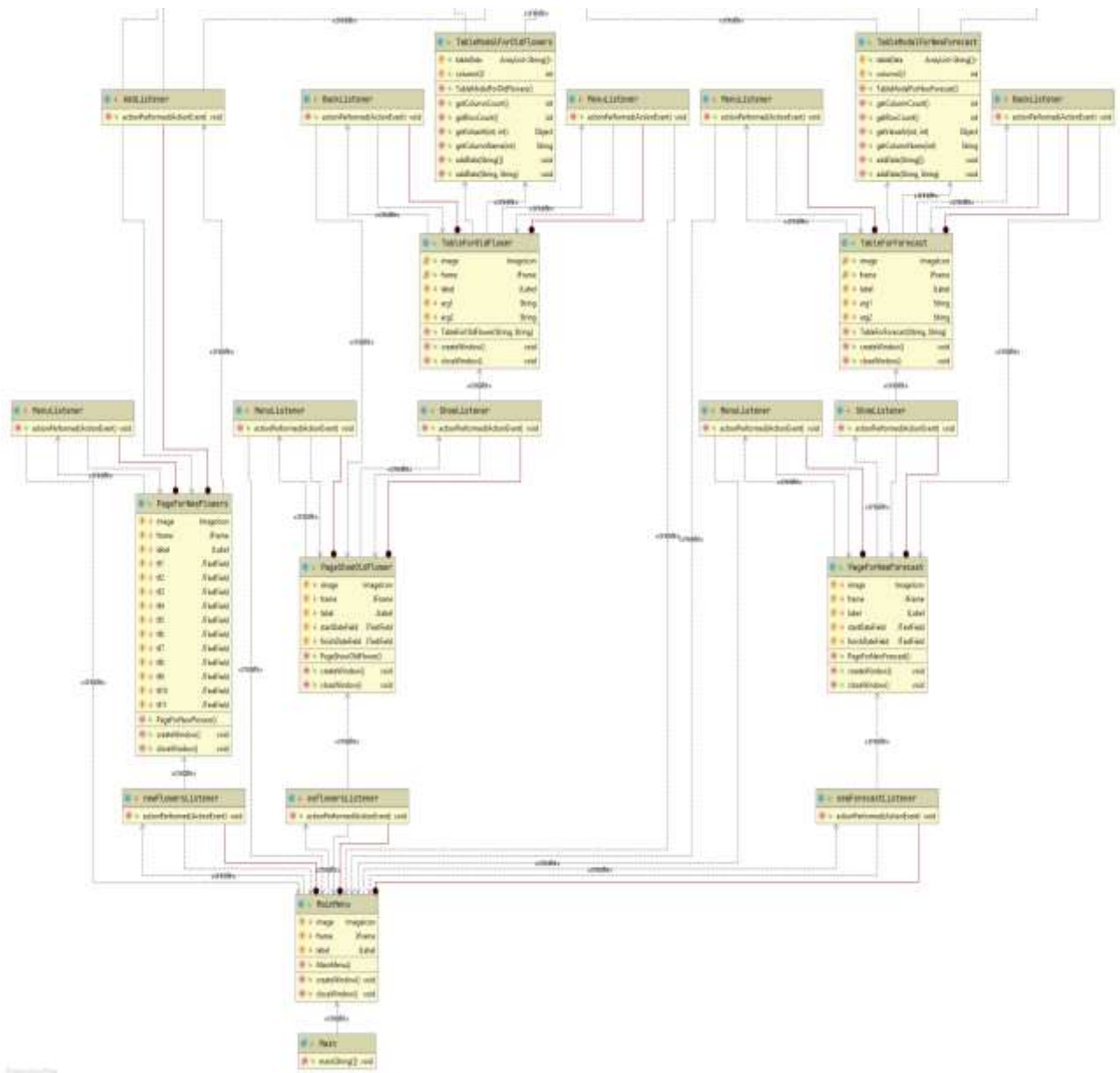


Рисунок В.5 – UML-діаграма класів (частина 2)

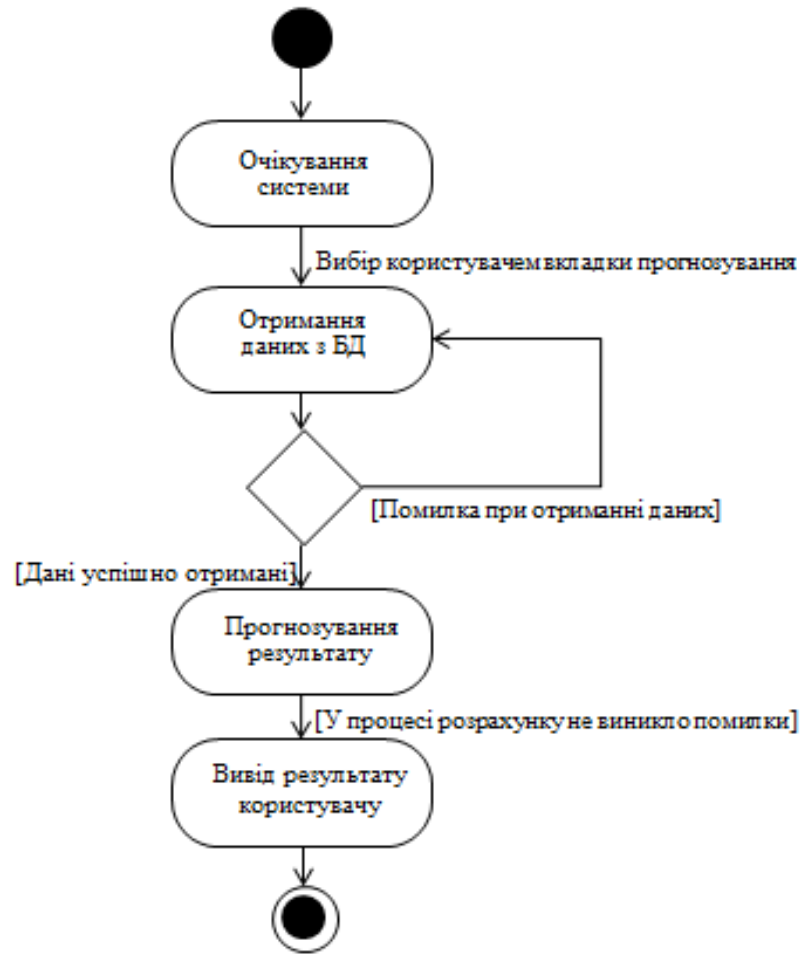
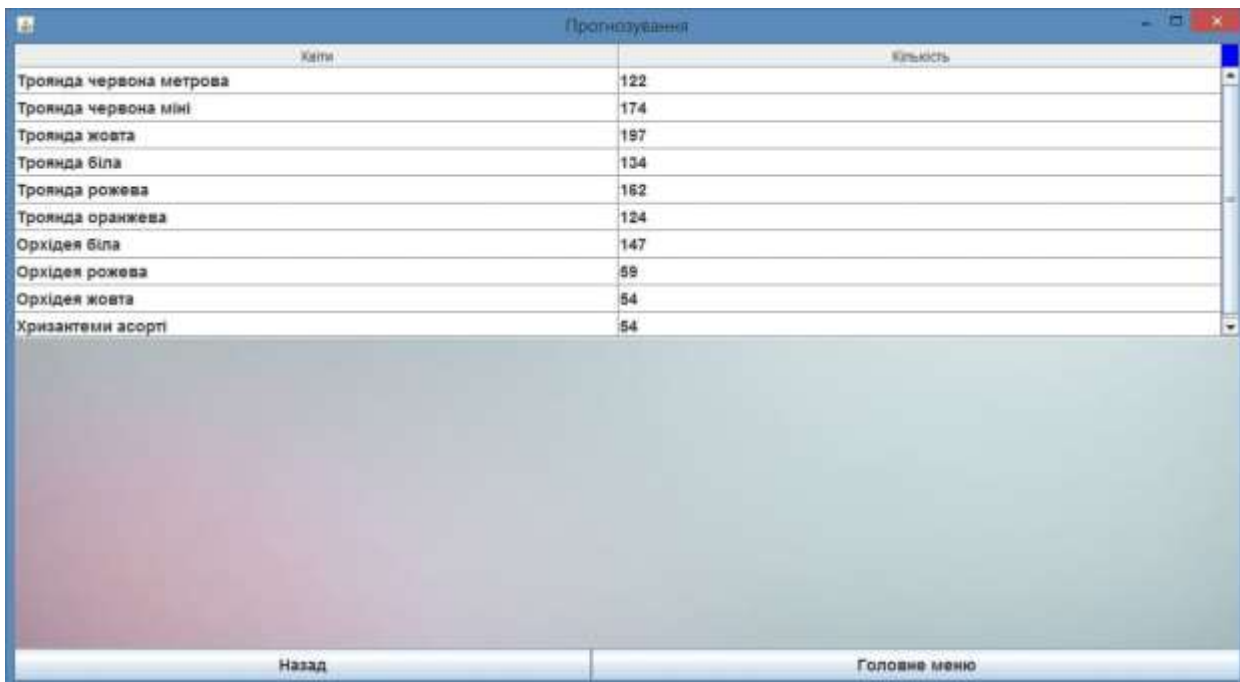


Рисунок В.6 – Діаграма діяльності системи при прогнозуванні



The screenshot shows a window titled "Прогнозування" (Forecasting) with a table of data. The table has two columns: "Категорія" (Category) and "Кількість" (Quantity). The data is as follows:

Категорія	Кількість
Троянда червона метрова	122
Троянда червона міні	174
Троянда жовта	197
Троянда біла	134
Троянда рожева	162
Троянда оранжева	124
Орхідея біла	147
Орхідея рожева	89
Орхідея жовта	54
Хризантеми асорті	54

At the bottom of the window, there are two buttons: "Назад" (Back) and "Головне меню" (Main menu).

Рисунок В.7 – Загальний вигляд прикладу роботи програми

Інструкція користувача

Після запуску програми відкривається вікно початкової активності – головне меню інформаційної технології (рисунок Г.1). Воно містить кнопки «Показати архів», «Додати інформацію» та «Прогнозування». На даному етапі необхідно обрати пункт меню, який цікавить користувача.

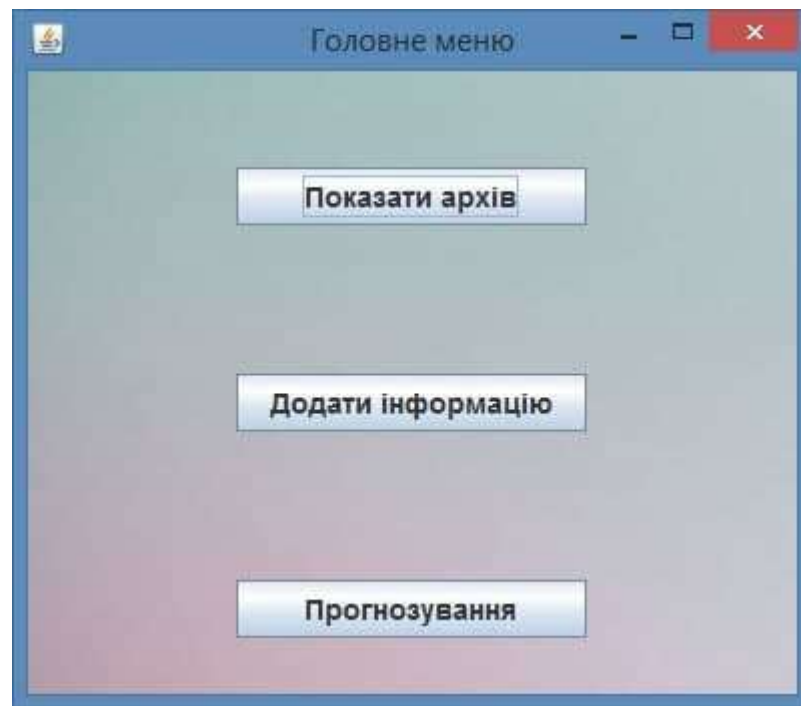
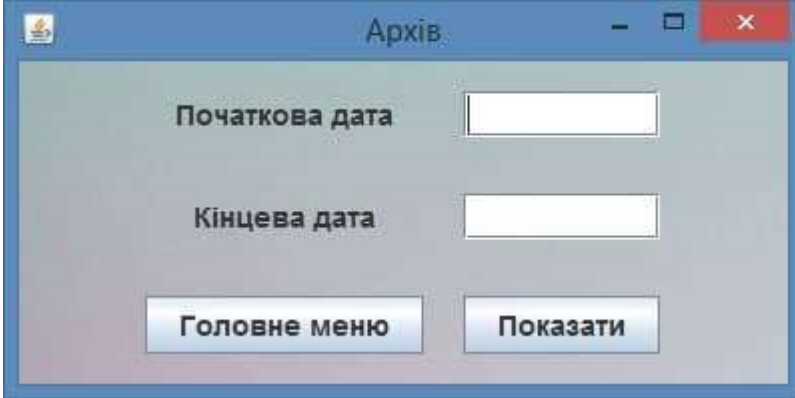


Рисунок Г.1 – Головне меню програми

При виборі користувачем першого пункту – «Показати архів» – виводиться вікно, в якому необхідно вказати проміжок часу, який цікавить користувача (рисунок Г.2) в спеціальні поля для початкової та кінцевої дати. Після введення поміжку користувачеві необхідно натиснути кнопку «Показати», після чого відбувається перехід до вікна архіву, що містить таблицю з закупками квітів на період, вказаний користувачем (рисунок Г.3).



Архів

Початкова дата

Кінцева дата

Головне меню Показати

Рисунок Г.2 – Вікно введення проміжку часу



Дата	Троянда черво.	Троянда черво.	Троянда жовта	Троянда біла	Троянда рожева	Троянда оран.	Орхідея біла	Орхідея рожева	Орхідея жовта	Хризантеми ас.
2019-07-05	9	13	14	12	8	12	11	4	5	3
2019-07-06	6	12	18	9	9	7	9	6	5	4
2019-07-07	7	13	9	13	13	11	14	6	4	5
2019-07-08	11	8	13	8	13	8	12	6	4	5
2019-07-09	7	12	12	10	15	10	15	4	5	4
2019-07-10	9	13	10	13	11	10	11	3	5	3
2019-07-11	9	8	12	9	11	7	9	4	5	5
2019-07-12	10	16	13	9	15	8	17	3	4	3
2019-07-13	10	11	18	12	13	11	15	3	3	3
2019-07-14	12	8	16	8	10	6	15	4	5	3

Назад Головне меню

Рисунок Г.3 – Вікно архіву

При виборі користувачем пункту «Додати інформацію» з головного вікна програми (рисунок Г.1) виводиться вікно для введення нових даних (рисунок Г.4). Для додавання даних користувач заповнює необхідну інформацію в спеціальні поля та натискає кнопку додати дані.



Тип квітки	Введення даних
Дата	<input type="text"/>
Троянда червона метрова	<input type="text"/>
Троянда червона міні	<input type="text"/>
Троянда жовта	<input type="text"/>
Троянда біла	<input type="text"/>
Троянда рожева	<input type="text"/>
Троянда оранжева	<input type="text"/>
Орхідея біла	<input type="text"/>
Орхідея рожева	<input type="text"/>
Орхідея жовта	<input type="text"/>
Хризантеми асорті	<input type="text"/>

Головне меню Додати дані

Рисунок Г.4 – Вікно введення нових даних

При виборі пункту «Прогнозування» з головного меню програми (рисунок Г.1) виводиться вікно для введення проміжку часу, на який необхідно здійснити прогноз (рисунок Г.5). Після введення проміжку часу користувачеві необхідно натиснути кнопку «Показати», після чого відбувається прогнозування (рисунок Г.6).

Прогнозування

Початкова дата

Кінцева дата

Рисунок Г.5 – Вікно введення проміжку часу для прогнозування

Прогнозування

Кіт	Кількість
Троянда червона метрова	122
Троянда червона міні	174
Троянда жовта	197
Троянда біла	134
Троянда рожева	162
Троянда оранжева	124
Орхідея біла	147
Орхідея рожева	59
Орхідея жовта	54
Хризантеми асорті	54

Рисунок Г.6 – Вікно виведення результатів прогнозування



МАЛЕ НАУКОВО-ВИРОБНИЧЕ ПІДПРИЄМСТВО "ТОВ "ІТІ"
Україна, 21021, м.Вінниця, вул. Келецька, 56

№ ____ від " __ " _____ 20__ р.

ДОВІДКА

Дана Климчук Анастасії Михайлівні в тому, що результати, одержані ним в процесі виконання магістерської кваліфікаційної роботи, а саме алгоритми та програмні засоби прогнозування обсягу закупівлі квітів, планується використати в розробках ТОВ «ІТІ».

Зам. директора ТОВ «ІТІ»

Бодяк В.М.