

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
МІСЬКОГО ГОСПОДАРСТВА імені О. М. БЕКЕТОВА

МЕТОДИЧНІ РЕКОМЕНДАЦІЇ

до проведення практичних занять
із навчальної дисципліни

«ІНФОРМАЦІЙНЕ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ
ДІЯЛЬНОСТІ ПІДПРИЄМСТВА»

*(для здобувачів першого (бакалаврського) рівня вищої освіти зі спеціальностей
122, F3 – Комп'ютерні науки, освітньо-професійна програма «Комп'ютерні
науки»)*

Харків
ХНУМГ ім. О. М. Бекетова
2025

Методичні рекомендації до проведення практичних занять із навчальної дисципліни «Інформаційне та програмне забезпечення діяльності підприємства» (для здобувачів першого (бакалаврського) рівня вищої освіти зі спеціальностей 122, F3 – Комп'ютерні науки, освітньо-професійна програма «Комп'ютерні науки») / Харків. нац. ун-т міськ. госп-ва ім. О. М. Бекетова ; уклад. : М. А. Мірошник, Ю. В. Пахомов. – Харків : ХНУМГ ім. О. М. Бекетова, 2025. – 50 с.

Укладачі: д-р техн. наук, проф. М. А. Мірошник,
канд. техн. наук, доц. Ю. В. Пахомов

Рецензент

О. В. Прохоров, доктор технічних наук, професор, професор кафедри комп'ютерних наук та інформаційних технологій Національного аерокосмічного університету імені М. Є. Жуковського «Харківський авіаційний інститут»

Рекомендовано кафедрою комп'ютерних наук та інформаційних технологій, протокол № 17 від 02.05.2025

ЗМІСТ

ВСТУП.....	5
ПРАКТИЧНЕ ЗАНЯТТЯ 1 ВПРОВАДЖЕННЯ СИСТЕМИ УПРАВЛІННЯ ПІДПРИЄМСТВОМ.....	7
1.1 Мета роботи.....	7
1.2 Завдання.....	7
1.3 Необхідний теоретичний матеріал.....	7
1.4 Порядок виконання роботи.....	9
1.4.1 Варіанти для виконання завдань (підприємства різних галузей).....	11
1.5 Звіт про виконання завдання.....	12
1.6 Контрольні запитання.....	12
ПРАКТИЧНЕ ЗАНЯТТЯ 2 ВИКОРИСТАННЯ CRM-СИСТЕМ ДЛЯ УПРАВЛІННЯ ВЗАЄМОВІДНОСИНАМИ З КЛІЄНТАМИ.....	14
2.1 Мета роботи.....	14
2.2 Необхідний теоретичний матеріал.....	14
2.2.1 Що таке CRM-системи?.....	14
2.2.2 Переваги використання CRM-систем для масштабування бізнесу....	15
2.2.3 Успішні приклади використання CRM-систем.....	16
2.3 Порядок виконання роботи.....	18
2.4 Детальний опис виконання роботи, оформлення звіту.....	20
2.5 Варіанти для виконання завдань.....	21
2.6 Контрольні запитання.....	21
ПРАКТИЧНЕ ЗАНЯТТЯ 3 ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ СТВОРЕННЯ ТА РОБОТИ З БАЗАМИ ДАНИХ. ІНТЕГРОВАНЕ СЕРЕДОВИЩЕ <i>DELPHI</i>	23
3.1 Мета роботи.....	23
3.2 Необхідний теоретичний матеріал.....	23
3.3 Проєкти в середовищі <i>Delphi</i>	27
3.4 Розташування елементів на сторінці панелі <i>Standard</i>	28
3.4.1 Основні властивості компонентів.....	31
3.5 Порядок виконання роботи.....	32
3.6 Оформлення звіту.....	33
3.7 Контрольні запитання.....	33
ПРАКТИЧНЕ ЗАНЯТТЯ 4 СИСТЕМИ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ НА ПІДПРИЄМСТВІ.....	35
4.1 Мета заняття.....	35
4.2 Необхідний теоретичний матеріал.....	35
4.2.1 Основні характеристики системи підтримки прийняття рішень.....	35

4.2.2 Структура та компоненти системи підтримки прийняття рішень.....	37
4.2.3 Етапи процесу прийняття рішень у системі підтримки прийняття рішень.....	37
4.2.4 Класифікація систем підтримки прийняття рішень.....	38
4.2.5 Переваги використання систем підтримки прийняття рішень.....	38
4.2.6 Застосування систем підтримки прийняття рішень.....	38
4.3 Завдання для виконання.....	39
4.4 Порядок виконання завдання.....	39
4.5 Контрольні запитання.....	41
ПРАКТИЧНЕ ЗАНЯТТЯ 5 СИСТЕМИ УПРАВЛІННЯ ЛАНЦЮГОМ ПОСТАВОК НА ПІДПРИЄМСТВІ.....	
5.1 Мета заняття.....	42
5.2 Необхідний теоретичний матеріал.....	42
5.2.1 Поняття та суть управління ланцюгами поставок.....	42
5.3 Завдання для виконання.....	45
5.4 Порядок виконання завдання.....	45
5.5 Контрольні запитання.....	47
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	48

ВСТУП

Дисципліна «Інформаційне та програмне забезпечення діяльності підприємства» належить до циклу фахової підготовки здобувачів першого (бакалаврського) рівня вищої освіти за спеціальностями 122, F3 – Комп'ютерні науки та відповідає освітньо-професійній програмі «Комп'ютерні науки».

Основна мета дисципліни – формування у здобувачів вищої освіти комплексних знань та навичок, необхідних для ефективної розробки, впровадження та управління інформаційними системами, що підтримують діяльність підприємства, що забезпечить підготовку майбутніх фахівців до успішної кар'єри в галузі *IT* і забезпечить їх необхідними інструментами та знаннями для вирішення складних технічних і управлінських завдань.

Підготовлений фахівець повинен набути певних знань і умінь в галузі *IT*-технологій, що передбачає:

- поглиблене розуміння знань про архітектуру, структуру та функції інформаційних систем, що використовуються в підприємницькій діяльності;
- розвиток практичних навичок програмування, використовуючи для цього сучасні мови програмування та інструменти розробки;
- вивчення методів збору, зберігання, обробки та аналізу даних, які є критично важливими для підтримки прийняття рішень на підприємствах, зокрема роботу з великими даними (*Big Data*) та використання аналітичних інструментів для прогнозування та аналізу;
- набуття навичок інтегрування різних інформаційних систем для забезпечення безперервності та ефективності планування бізнес-процесів;
- вивчення основних принципів інформаційної безпеки, включаючи захист даних, управління доступом, шифрування та протидію кіберзагрозам;
- використання сучасних технологій, таких як хмарні обчислення, штучний інтелект, машинне навчання, інтернет-речі (*IoT*) та блокчейн;
- підготовка до професійної діяльності;

– стратегічне управління інформаційними технологіями для досягнення бізнес-цілей, зокрема впровадження нових технологій, управління проєктами та ресурсами.

Практична частина курсу призначена для отримання практичних навичок щодо використання теоретичних знань на підприємстві, що сприяє забезпеченню глибшого розуміння матеріалу й допоможе поглибити набуті знання про інформаційні системи та програмне забезпечення.

Практичні завдання спрямовані на розвиток ключових навичок програмування, розробки програмного забезпечення, роботи з базами даних та іншими технологіями, що необхідно для професійної діяльності у сфері інформаційних систем.

Практичні завдання зазвичай базуються на реальних кейсах і проєктах, що дозволяє студентам зрозуміти, які теоретичні концепції застосовуються в реальних умовах бізнесу. Це готує їх до вирішення практичних проблем у майбутній професійній діяльності.

Практичні роботи сприяють розвитку креативного мислення та навичок вирішення проблем, що важливо для інноваційної діяльності у сфері інформаційних технологій. Роботи послідовно та логічно впорядковані за складністю та охоплюють всі теми, що вивчаються у цьому курсі.

Матеріал для кожної практичної роботи містить мету, основні теоретичні відомості, загальне завдання, варіанти індивідуальних завдань, список питань для самоперевірки, приклад звіту, а також список рекомендованих джерел для підготовки та виконання практичних робіт.

ПРАКТИЧНЕ ЗАНЯТТЯ 1 ВПРОВАДЖЕННЯ СИСТЕМИ УПРАВЛІННЯ ПІДПРИЄМСТВОМ

1.1 Мета роботи

Мета роботи полягає у формуванні у студентів розуміння сутності *ERP*-систем, їх ролі в автоматизації та оптимізації бізнес-процесів підприємства, а також у набутті практичних навичок планування та організації етапів впровадження *ERP*-систем з урахуванням особливостей конкретного бізнесу. Завдяки цьому студенти зможуть аналізувати бізнес-процеси, визначати цілі впровадження *ERP*, розробляти плани проекту, здійснювати налаштування та тестування системи, а також організовувати навчання користувачів і запуск системи в експлуатацію для підвищення ефективності управління ресурсами підприємства.

1.2 Завдання

1. Вибрати та встановити безкоштовну *ERP*-систему (наприклад, *Odoo*, *ERPNext*).
2. Налаштувати основні модулі системи для конкретного підприємства (наприклад, фінанси, управління складом, продажі).
3. Провести тестування системи та оцінити її ефективність.

1.3 Необхідний теоретичний матеріал

Планування ресурсів підприємства (*ERP*-система) (англ. *Enterprise Resource Planning System* – «корпоративна інформаційна система» (*KIC*)) призначена для автоматизації обліку й керування. Зазвичай *ERP*-системи будуються за модульним принципом і в так чи інакше охоплюють всі ключові процеси діяльності компанії або підприємства.

Класичні системи *ERP* забезпечують:

- керування фінансами;
- керування виробництвом;
- керування формуванням та розподілом запасів;
- керування реалізацією та маркетингом;
- керування утриманням покупців;
- керування постачанням;
- керування проєктами;
- керування сервісним обслуговуванням;
- керування процедурами забезпечення якості продукції.

Історично концепція *ERP* стала розвитком більш ранніх концепцій *MRP I* (*Material Requirement Planning* – *Планування матеріальних потреб*) і *MRP II* (*Manufacturing Resource Planning* – *Планування виробничих ресурсів*). Також в *ERP*-системах використовується програмний інструментарій, який дозволяє проводити виробниче планування, моделювання потоку замовлень, оцінювання можливості їхньої реалізації в службах і підрозділах підприємства в поєднанні його зі збутом.

Класичні *ERP*-системи, на відміну від так званого програмного забезпечення «в коробці», належать до категорії «важких» замовних програмних продуктів: вибір, придбання і впровадження зазвичай потребують ретельного планування в рамках тривалого проєкту з участю партнерської компанії-постачальника або консультанта. Оскільки *KIC* будуються за модульним принципом, замовник часто (принаймні, на ранніх стадіях таких проєктів) купує не повний спектр модулів, а їхній обмежений комплект. У процесі впровадження проєктна команда зазвичай протягом декількох місяців (до року) здійснює налаштування модулів, що поставляються замовнику.

Використання *ERP*-систем дозволяє використовувати одну інтегровану програму замість декількох розрізнених. Єдина система може керувати обробкою, логістикою, дистрибуцією, запасами, доставкою, виставлянням рахунків, бухгалтерським обліком, податковим обліком, програмою лояльності.

Реалізована система розмежування доступу до інформації, яка призначена (в комплексі з іншими заходами інформаційної безпеки підприємства) для протидії як зовнішнім загрозам (наприклад, промислового шпигунству), так і внутрішнім (наприклад, розкраданням). Упроваджувані спільно з *CRM*-системою і системою контролю якості, *ERP*-системи спрямовані на максимальне задоволення потреб компаній-замовників у засобах управління бізнесом.

Є деякі обмеження використання *ERP*-систем, а саме:

- невеликі компанії не можуть дозволити собі інвестувати достатньо коштів в *ERP* і адекватно навчити всіх співробітників;
- іноді *ERP*-систему складно або неможливо адаптувати до документообігу компанії і її специфічних бізнес-процесів;
- система може мати проблему «слабої ланки», ефективність всієї системи може бути порушена одним департаментом або партнером;
- опір департаментів у наданні інформації зменшує ефективність системи;
- проблема сумісності з попередніми системами.

Натепер з розвитком хмарних технологій такі системи можуть бути доступні як для малого, так і для середнього бізнесу.

1.4 Порядок виконання роботи

Завдання 1. Вибір та встановлення безкоштовної *ERP*-системи (наприклад, *Odoo*, *ERPNext*).

На першому етапі впровадження *ERP*-системи необхідно обрати відповідну платформу. Серед безкоштовних рішень великої популярності набули такі системи:

Odoo – модульна *ERP*-система з відкритим вихідним кодом, яка включає інструменти для фінансів, продажів, *CRM*, складу, *HR* тощо.

ERPNext – ще одне потужне рішення з відкритим кодом, що орієнтоване на малий і середній бізнес. Має простий інтерфейс та розширену функціональність.

Для встановлення системи потрібно:

- встановити середовище: сервер на базі *Ubuntu / Debian, Docker* або *VPS*;
- завантажити вихідні файли з офіційного репозиторію (*GitHub*);
- налаштувати базу даних (*PostgreSQL* для *Odoo*, *MariaDB* для *ERPNext*);
- запустити сервіс і виконати початкове налаштування через вебінтерфейс.

Завдання 2. Налаштування основних модулів системи для конкретного підприємства.

Після встановлення системи потрібно адаптувати її під потреби підприємства. Для прикладу візьмемо умовне підприємство «Дистриб'ютор електроніки».

Налаштовуються такі модулі:

Фінанси: створення рахунків, планів рахунків, податкових ставок, форм звітності (баланс, прибутки / збитки).

Управління складом: налаштування складів, маршрутів переміщення товарів, рівнів запасів, штрих-кодування.

Продажі: створення клієнтів, товарів, налаштування цін, автоматизація рахунків і замовлень.

CRM (опціонально): для керування потенційними клієнтами та нагадувань про подальші дії.

Також виконується створення користувачів, призначення ролей, налаштування прав доступу та брендуння інтерфейсу.

Завдання 3. Проведення тестування системи та оцінка ефективності.

На завершальному етапі проводиться перевірка коректності роботи системи в умовах, наближених до реальних, а саме:

1. Імітація типових бізнес-процесів (наприклад, оформлення замовлення, проведення платежу, відвантаження зі складу).
2. Виявлення помилок у налаштуванні або логіці роботи модулів.
3. Оцінка зручності інтерфейсу та адаптації персоналу.
4. Визначення часу виконання ключових операцій.

За результатами тестування складається звіт, що включає:

- переваги системи (швидкість, функціональність);

- обмеження або проблеми;
- рекомендації щодо доопрацювання або переходу на інше рішення (за потреби).

Таким чином, практичне впровадження *ERP*-системи потребує технічних знань, аналізу бізнес-процесів підприємства, а також планування і тестування на кожному етапі.

1.4.1 Варіанти завдань (підприємства різних галузей)

1. Мотор Січ (машинобудування).
2. *Nestlé* Україна (харчова промисловість, *FMCG*).
3. ДТЕК Нафтогаз (паливно-енергетичний комплекс).
4. *EVA* (ритейл).
5. Київський завод шампанських вин (харчова промисловість).
6. Ельдорадо (торгівля електронікою).
7. *Carlsberg Ukraine* (пивоварна промисловість).
8. Укрзалізниця (транспорт).
9. *SoftServe* (IT).
10. *Procter & Gamble* Україна (*FMCG*).
11. ДніпроАзот (хімічна промисловість).
12. *Winner Imports* (автомобільна торгівля).
13. *Glencore Agriculture* (агробізнес, торгівля).
14. Таврія Плюс (ритейл).
15. *Sandora* (харчова промисловість).
16. Міжнародні авіалінії України (авіап перевезення).
17. *EPAM Systems* (IT).
18. Ювелірний завод «Золотий Вік» (ювелірна промисловість).
19. Меблева фабрика «Мірт» (меблева промисловість).
20. ТОВ Нестле Україна.
21. *PepsiCola*.

1.5 Звіт про виконання роботи

Основні етапи роботи:

1. Отримати та прокоментувати результати.
2. Усі запити й результати представити як скриншоти із необхідними поясненнями.
3. Зробити висновки за роботою.

1.6 Контрольні запитання

1. Що таке *ERP*-система і які основні завдання вона вирішує на підприємстві?
2. Які ключові функції *ERP*-системи ви знаєте? Наведіть приклади їх застосування.
3. Які основні модулі *ERP*-системи та які бізнес-процеси вони автоматизують?
4. Як *ERP*-система допомагає у плануванні ресурсів підприємства?
5. Яким чином *ERP*-система сприяє підвищенню ефективності виробничих процесів?
6. Які інструменти *ERP*-системи використовуються для управління фінансами підприємства?
7. Як *ERP*-система допомагає у синхронізації роботи різних підрозділів підприємства?
8. Які етапи впровадження *ERP*-системи на підприємстві можна виокремити?
9. Які можливості *ERP*-система надає для автоматизації документообігу та управління проєктами?
10. Як *ERP*-система підтримує роботу з клієнтами (*CRM*-функції) та управління персоналом?
11. Що таке *ERP*-система і які її основні функції?

12. Які основні переваги впровадження *ERP*-системи на підприємстві?
13. Які етапи включає процес впровадження *ERP*-системи?
14. Які ключові фактори успішного впровадження *ERP*?
15. Як аналіз бізнес-процесів допомагає у впровадженні *ERP*?
16. Які основні труднощі можуть виникнути під час впровадження *ERP*-системи?
17. Яку роль відіграє навчання користувачів у процесі впровадження *ERP*?
18. Як оцінити ефективність роботи *ERP*-системи після її впровадження?
19. Які типи *ERP*-систем існують за способом розгортання (on-premise, хмарні тощо)?
20. Як інтеграція *ERP* з іншими інформаційними системами підприємства підвищує ефективність управління?

ПРАКТИЧНЕ ЗАНЯТТЯ 2 ВИКОРИСТАННЯ CRM-СИСТЕМ ДЛЯ УПРАВЛІННЯ ВЗАЄМОВІДНОСИНАМИ З КЛІЄНТАМИ

2.1 Мета роботи

Використання програмного забезпечення для управління взаємовідносинами з клієнтами, для зберігання та аналізу інформації про клієнтів, автоматизації маркетингу, продаж, сервісу і підвищення ефективності бізнесу.

2.2 Необхідний теоретичний матеріал

2.2.1 Що таке CRM-системи?

CRM для послуг, або системи управління відносинами з клієнтами, є інтегрованими програмними продуктами, які спеціалізуються на збереженні, управлінні та аналізі інформації про клієнтів та їх взаємодії з компанією. Вони використовуються для автоматизації та оптимізації таких процесів, як управління контактами, управління продажами, маркетингові кампанії та обслуговування покупців.

Основна мета *CRM*-систем для товарного бізнесу полягає у поліпшенні відносин з клієнтами, збільшенні їхньої лояльності та задоволеності, що, своєю чергою, приводить до збільшення обсягів продажів та прибутковості. Ці системи дозволяють підприємствам збирати інформацію про користувачів з різних джерел, відстежувати їхню взаємодію з компанією, аналізувати ці дані та використовувати їх для прийняття стратегічних рішень.

Однією з ключових функцій *CRM*-систем для інтернет-магазинів є зберігання централізованої бази даних про клієнтів, у якій можна зберігати інформацію про контактні дані покупців, історію їхніх замовлень, взаємодій та інші важливі деталі. Це дозволяє співробітникам різних відділів компанії

отримувати безперешкодний доступ до цієї інформації та ефективно взаємодіяти з користувачами.

Крім того, *CRM* допомагає керувати продажами, включаючи відстеження потенційних та поточних угод, підімкнення платіжних систем до сайту, керування воронкою продажів та прогнозування. Вони надають засоби для створення та відстеження маркетингових кампаній, а також аналізу їх ефективності.

Також однією з переваг українських *CRM*-систем, таких як *SITNIKS*, *Zoho CRM* тощо, є їх можливість інтегрування з *ЛідФорм* та іншими бізнес-системами, такими як системи управління складом, бухгалтерські програми та електронна пошта. Це дозволяє підприємствам створювати повністю інтегровану систему управління бізнесом, що сприяє підвищенню ефективності та оптимізації робочих процесів.

Отже, *CRM* для інтернет-магазинів відіграють важливу роль у сучасному бізнес-світі, допомагаючи підприємствам збільшити ефективність управління відносинами з клієнтами, поліпшити обслуговування і збільшити прибуток. Їхнє використання стає необхідним для компаній, які прагнуть залишатися конкурентоздатними в умовах постійних змін на ринку.

2.2.2 Переваги використання *CRM*-систем для масштабування бізнесу

Використання *CRM* для інтернет-торгівлі та масштабування бізнесу відкриває безліч переваг, які допомагають компаніям рости та розвиватися на ринку. Нижче наведені кілька ключових переваг цього підходу.

Централізоване зберігання даних. *CRM*-системи для оптової торгівлі надають можливість зберігати всю інформацію про клієнтів, контакти, угоди та історію взаємодії в одній централізованій базі даних. Це полегшує доступ до неї для всіх відділів компанії та сприяє зменшенню розпорошеності даних.

Поліпшення обслуговування клієнтів. *CRM* дозволяє підприємствам більш ефективно відстежувати та реагувати на потреби своїх клієнтів. Завдяки аналізу

даних компанії можуть надавати персоналізовані послуги та продукти, що підвищує їхню лояльність та задоволеність.

Збільшення продуктивності працівників. Автоматизація з *Facebook* та оптимізація інших рутинних завдань, таких як обробка заявок, відстеження контактів та планування зустрічей, дозволяє співробітникам більше часу приділяти важливим стратегічним завданням та взаємодії з клієнтами.

Ефективне управління продажами. CRM-системи допомагають відстежувати угоди, прогнозувати продажі та визначати потенційні можливості для росту. Це дозволяє оптимізувати процес продажів та збільшити їхні обсяги.

Аналіз та оптимізація маркетингових кампаній. CRM для бізнесу дозволяє відстежувати ефективність маркетингових кампаній, а також ідентифікувати найбільш перспективні канали комунікації з клієнтами. Це допомагає підприємствам зосередитися на стратегіях, які найбільш ефективні для їхньої аудиторії.

Масштабування бізнесу. Завдяки усуненню рутинних завдань та поліпшенню ефективності управління українські CRM-системи дозволяють компаніям приділяти більше уваги стратегічному росту та масштабуванню бізнесу. Це може включати розширення ринків, розробку нових продуктів або послуг, а також виявлення нових можливостей для розвитку.

Використання кращих CRM-систем для малого бізнесу з метою масштабування організацій дозволяє підприємствам оптимізувати взаємодію з клієнтами, підвищити продуктивність та ефективність управління, що сприяє загальному росту та успіху компанії.

2.2.3 Успішні приклади використання CRM

Нині існує чимало прикладів успішного використання CRM-систем в різних сферах бізнесу. Такими є компанії «Нова Пошта», *Rozetka* та *SoftServe*. Завдяки використанню систем вони ефективно поліпшують взаємодію з клієнтами та оптимізують свої послуги.

«Нова Пошта». Використовує CRM-систему для зберігання та обробки інформації про клієнтів, що дозволяє компанії ефективно взаємодіяти на всіх етапах послуг. Основними аспектами використання CRM-систем *«Новою Поштою»* є управління клієнтською базою даних завдяки зберігання інформації про своїх користувачів, включаючи контактні дані, історію замовлень та попередній контакт. Це дозволяє оператору швидко надавати інформацію та реагувати на їхні запити.

Крім цього, *«Нова Пошта»* може надавати персоналізовані послуги своїм клієнтам й адаптувати пропозиції та рекомендації відповідно до індивідуальних потреб. Також за допомогою автоматизації *«Нової Пошти»* спрощується виконання багатьох рутинних процесів, таких як ведення журналу замовлень, програм для обліку товару, відстеження вантажів та надання звітів. Це полегшує роботу персоналу та підвищує ефективність.

Дані, які компанія збирає завдяки CRM-системам, дозволяють вдосконалювати свої послуги та пристосовуватися до змін на ринку, а завдяки персоналізованому підходу та якісному обслуговуванню *«Нова Пошта»* підвищує лояльність своїх клієнтів. Враховуючи успішний досвід цієї компанії щодо використання CRM-систем, можна зробити висновок, що вона стала важливим інструментом для оптимізації та розвитку її бізнесу.

Rozetka. Відома компанія використовує CRM для ведення бази даних клієнтів та управління продажами. Організація зберігає дані про покупців у CRM-системі, включаючи їхні контактні дані, історію покупок та поведінку на сайті. Це дозволяє надавати персоналізовані пропозиції та послуги. За допомогою CRM для послуг, *Rozetka* може розділити свою клієнтську базу на сегменти за різними критеріями, такими як поведінка, відповідність щодо певних категорій товарів тощо. Це дозволяє компанії створювати цільові маркетингові кампанії та пропозиції для кожного сегменту.

Крім того, *Rozetka* використовує CRM для автоматизації процесів маркетингу, включаючи розсилку рекламних електронних листів, SMS-повідомлень та сповіщень в мобільному додатку. Завдяки цьому компанія

ефективно комунікує з клієнтами через послугу «смарт-месенджер» та стимулює їх до повторних покупок. Також *CRM* допомагає прогнозувати майбутні продажі на основі історичних даних та тенденцій ринку, відстежувати якість обслуговування клієнтів і реагувати на їхні запити та скарги.

Завдяки успішному використанню *CRM*-системи *Rozetka* змогла підвищити ефективність своїх продажів та забезпечити високий рівень задоволеності клієнтів. Цей приклад показує, як використання *CRM* для товарного бізнесу може стати ключовим фактором успіху для українських компаній у висококонкурентному середовищі.

SoftServe. Використовує *CRM*-систему для ефективного управління відносинами з клієнтами та збільшення продажів. Завдяки *CRM* компанія відстежує поточних та потенційних клієнтів, керує угодами та переговорами, а також планує та прогнозує продажі. *SoftServe* розділяє свою базу користувачів на різні сегменти за допомогою *CRM* для бізнесу, що дозволяє компанії створювати спеціалізовані пропозиції та послуги для кожного сегменту.

CRM-система дозволяє організації ефективно взаємодіяти з клієнтами через різні канали комунікації, включаючи електронну пошту, телефон, боти для спілкування в *Telegram*, чати та особистий кабінет. Крім цього, компанія використовує аналітичні засоби *CRM* для аналізу даних про користувачів та їхню взаємодію з ними, що дозволяє підвищити ефективність маркетингових та продажних стратегій.

2.3 Порядок виконання роботи

Завдання 1. Аналіз потреб підприємства у *CRM*-системах.

Мета: визначити, які бізнес-процеси підприємства потребують автоматизації за допомогою *CRM*.

Для виконання цього завдання необхідно:

– вивчити структуру підприємства, основні канали взаємодії з клієнтами;

- визначити ключові проблеми у роботі з клієнтами (наприклад, дублювання контактів, втрата лідів, неефективна комунікація);
- скласти перелік функцій, які повинна вирішувати CRM-система для певного підприємства.

Завдання 2. Вибір CRM-системи.

Мета: ознайомитися з пропозиціями ринку та обґрунтувати вибір CRM для підприємства.

Порядок виконання:

- провести огляд популярних CRM-систем (наприклад, *Bitrix24*, *Creatio*, *Salesforce*, *amoCRM*);
- порівняти їх функціональні можливості, ціну, зручність інтерфейсу, інтеграції;
- вибрати найбільш відповідну CRM-систему та обґрунтувати свій вибір з урахуванням потреб підприємства.

Завдання 3. Створення та наповнення клієнтської бази.

Мета: навчитися заносити дані про клієнтів у CRM та структурувати інформацію.

Порядок виконання:

- створити нових клієнтів у системі, заповнити їх контактні дані, додати історію взаємодій;
- сегментувати клієнтів за заданими критеріями (наприклад, за типом, регіоном, обсягом покупок);
- перевірити можливість імпорту даних із зовнішніх джерел (*Excel*, *Google Contacts* тощо).

Завдання 4. Управління продажами та комунікаціями.

Мета: опанувати роботу з лідами, угодами та автоматизованими комунікаціями.

Порядок виконання:

- додати новий лід (потенційного клієнта), створити угоду та провести її через етапи продажу (контакт, переговори, укладення угоди);

- запланувати та зафіксувати комунікації (дзвінки, імейл, зустрічі);
- використати шаблони для автоматичної розсилки повідомлень клієнтам.

Завдання 5. Аналіз ефективності роботи з клієнтами.

Мета: навчитися використовувати аналітичні інструменти *CRM* для оцінки результатів.

Порядок виконання:

- побудувати звіти щодо кількості нових клієнтів, закритих угод, ефективності маркетингових кампаній;
- проаналізувати причини втрати клієнтів і запропонувати шляхи поліпшення роботи з ними;
- підготувати коротку презентацію результатів аналізу.

2.4 Детальний опис виконання роботи, оформлення звіту

Оформлення звіту передбачає такі етапи:

1. *Підготовка*: ознайомитися з теоретичними засадами CRM, переглянути інтерфейс обраної системи.
2. *Аналіз потреб*: провести аналіз бізнес-процесів підприємства, визначити, які з них можна автоматизувати.
3. *Вибір CRM*: оцінити декілька CRM-систем, зробити вибір і обґрунтувати його.
4. *Реєстрація клієнтів*: створити клієнтську базу, заповнити профілі, сегментувати дані.
5. *Управління продажами*: внести потенційних клієнтів, провести їх по воронці продажів, зафіксувати всі комунікації.
6. *Автоматизація*: налаштувати автоматичні розсилки, нагадування, шаблони листів.
7. *Аналітика*: згенерувати звіти, проаналізувати показники, підготувати висновки щодо ефективності використання CRM.

8. *Захист роботи*: представити результати виконаних завдань, відповісти на контрольні запитання.

2.5 Варіанти для виконання завдань

1. Мотор Січ (машинобудування).
2. ДТЕК (енергетика).
3. Київстар (телекомунікації).
4. Епіцентр К (ритейл).
5. Укрзалізниця (транспорт).
6. «Нова Пошта» (логістика).
7. Кернел (агробізнес).
8. Оболонь (харчова промисловість).
9. ПриватБанк (банківська сфера).
10. Інтерпайп (металургія).
11. АТБ-Маркет (ритейл).
12. МХП (Миронівський хлібопродукт, агроіндустрія).
13. *SoftServe (IT)*.
14. Фармак (фармацевтика).
15. *Carlsberg Ukraine* (харчова промисловість).
16. *Rozetka (e-commerce)*.
17. Укрнафта (нафтогазова промисловість).
18. *Comfy* (торгівля електронікою).
19. *Winner Imports Ukraine* (автомобільна торгівля).

2.6 Контрольні запитання

1. Що таке *CRM*-система? Які завдання вона виконує на підприємстві?
2. Які переваги впровадження *CRM*-системи для підприємства?
3. Опишіть основні функції *CRM*-системи.

4. Назвіть типи *CRM*-систем, які ви знаєте? Чим вони відрізняються?
5. Як формується і ведеться клієнтська база у *CRM*-системі?
6. Перелічіть етапи впровадження *CRM*-системи на підприємстві.
7. Назвіть можливості автоматизації *CRM*-системи для відділу продажу.
8. Як *CRM*-система допомагає аналізувати ефективність роботи з клієнтами?
9. Які дані про клієнтів доцільно зберігати у *CRM*-системі?
10. Які інструменти *CRM*-системи використовуються для комунікації з клієнтами?
11. Як *CRM*-система сприяє підвищенню лояльності клієнтів?
12. Які проблеми можуть виникати під час упровадження *CRM*-системи?
13. Як здійснюється інтеграція *CRM*-системи з іншими інформаційними системами підприємства?
14. Які можливості аналітики та звітності надає *CRM*-система?
15. Які заходи безпеки необхідно враховувати під час роботи з *CRM*-системою?

ПРАКТИЧНЕ ЗАНЯТТЯ З ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ СТВОРЕННЯ ТА РОБОТИ З БАЗАМИ ДАНИХ. ІНТЕГРОВАНЕ СЕРЕДОВИЩЕ DELPHI

3.1 Мета роботи

Знайомство з інтегрованим середовищем *Delphi* та принципами роботи проєктів, які були створені у цьому середовищі.

3.2 Необхідний теоретичний матеріал

Інтегроване середовище розробки *Delphi* (*Delphi IDE*) є багатовіконною системою. Вона включає все необхідне для швидкої розробки Windows-додатків і може швидко налаштовуватися.

Delphi має певний стандартний, передбачений розробниками вигляд, у якому вона з'являється при першому запуску. У такому «стандартному» варіанті середовище *Delphi* має такі вікна (рис. 3.1).

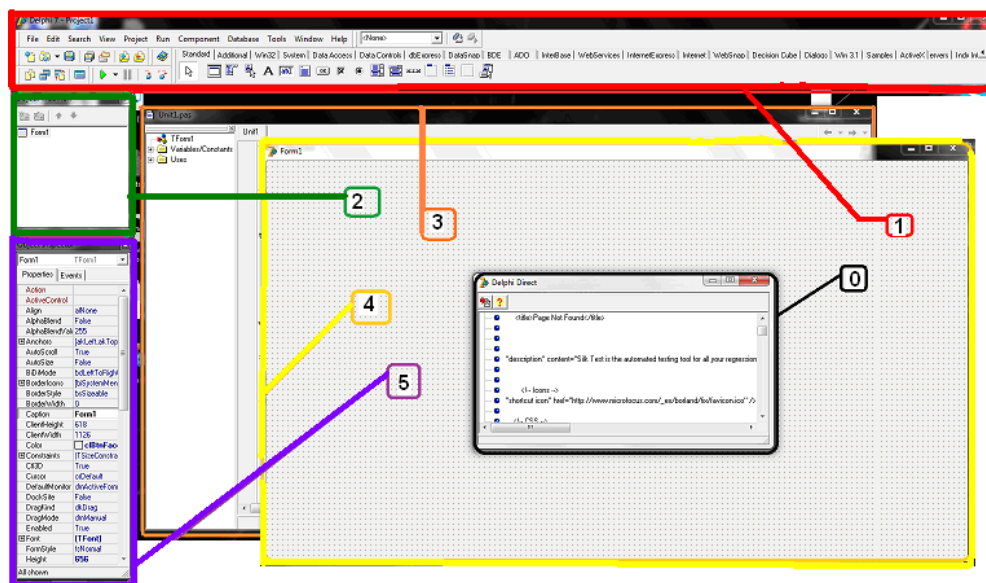


Рисунок 3.1 – Вигляд *Delphi 7 IDE* за замовчуванням

Нижче наведена характеристика та функціональні можливості вікон, які ми бачимо на рисунку 3.1:

0 – вікно зв'язку с *Inprise (Borland)*.

1 – головне вікно (*Delphi 7 – Project1*).

2 – вікно дерева об'єктів (*Object TreeView*).

3 – поєднане вікно редактора коду та провідника коду (на задньому плані, за формою *Form1*).

4 – вікно конструктора форм (*Form1*).

5 – вікно інспектора об'єктів (*Object Inspector*).

Вікно зв'язку с Inprise (Borland) (0). При першому запуску *Delphi* поверх всіх вікон з'явиться вікно зв'язку з розробником *Inprise*, що забезпечує доступ до його вебсторінок.

Головне вікно (1). Здійснює основні функції управління проєктом створюваної програми. Це вікно завжди є на екрані і займає його верхню частину. Воно містить елементи, які завжди повинні бути під рукою програміста. У головному вікні розташовуються головне меню *Delphi*, панелі інструментів та палітра компонентів (рис. 3.2).

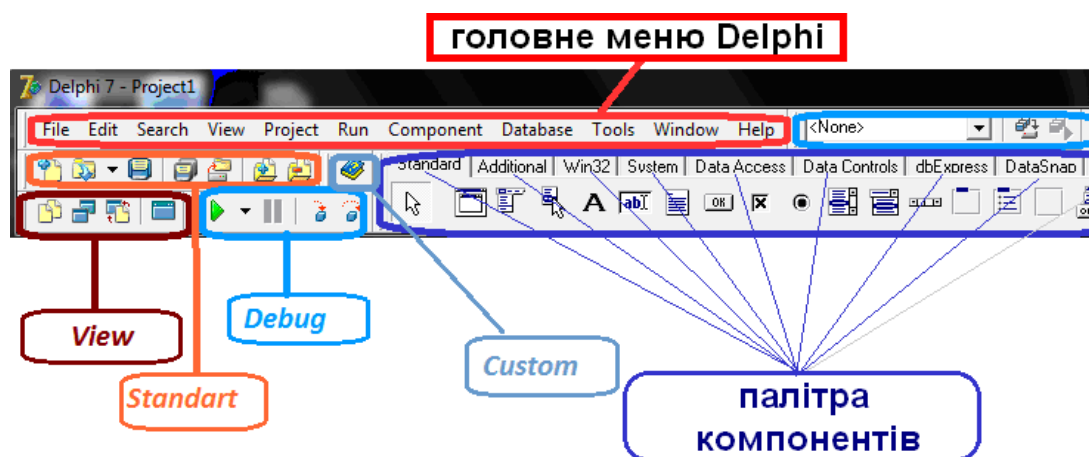


Рисунок 3.2 – Головне меню (*Delphi 7 – Project1*)

Палітра компонентів (*Component palette*) займає праву верхню частину головного вікна і має закладки (сторінки) для забезпечення доступу до потрібного компонента. Під компонентом розуміють функціональний елемент, що містить певні властивості, які розміщуються програмістом у вікні форми. За допомогою компонентів створюється каркас програми, зокрема її видимі на екрані зовнішні прояви – вікна, кнопки, списки вибору тощо.

Усі компоненти згруповані за вкладками, кількість та склад яких дещо відрізняються залежно від версії та варіантів поставки. Наприклад, *Delphi 7 Enterprise* має 33 вкладки, що містять компоненти, які належать до тієї чи іншої групи *VCL* (*VCL* – це бібліотека-сховище компонентів).

Вікно дерева об'єктів (2). Призначене для наочного відображення зв'язків між окремими компонентами, розміщеними на активній формі або активному модулі даних.

Поєднане вікно редактора коду та провідника коду (3). Вікно редактора коду (або просто вікно коду) призначене для створення та редагування тексту програми (рис. 3.3).

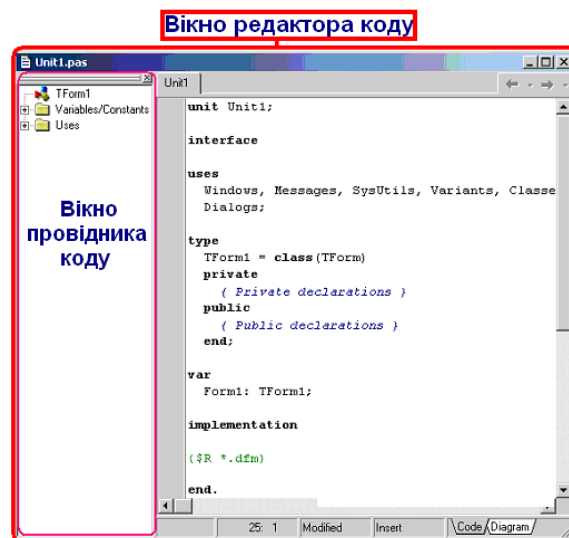


Рисунок 3.3 – Вікно редактора коду і провідника коду

Цей текст складається за спеціальними правилами та описує алгоритм роботи програми. Сукупність правил запису тексту називається мовою

програмування. У системі *Delphi* використовується мова програмування *Object Pascal*.

Спочатку вікно коду містить мінімальний початковий текст, що забезпечує нормальне функціонування порожньої форми як повноцінного Windows-вікна. Під час роботи над проектом програміст вносить до нього необхідні доповнення, щоб надати програмі необхідну функціональність.

Вікно конструктора форм (вікно форми) (4). Вікно форми є проектом Windows-вікна майбутньої програми. Спочатку це вікно порожнє, його робоча область заповнена крапками координатної сітки, яка потрібна для впорядкування розміщення на формі компонентів.

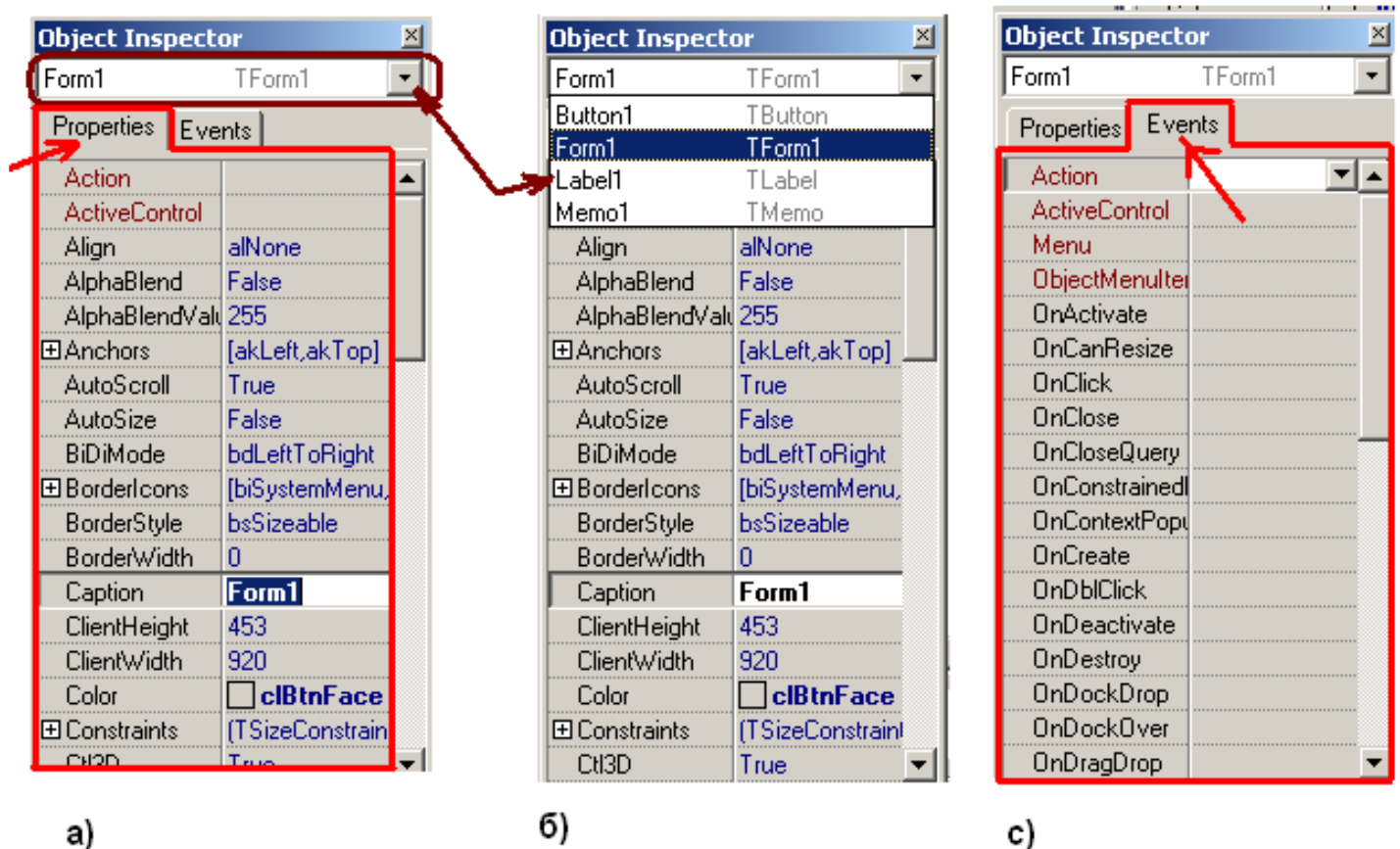


Рисунок 3.4 – Вікно інспектора об'єктів

Вікно інспектора об'єктів (5). Будь-який компонент, розміщений на формі, характеризується набором властивостей: положенням, розміром, кольором тощо. Для зміни значення цих властивостей призначене вікно

інспектора об'єктів. Це вікно містить дві сторінки: *Properties* – властивості та *Events* – події (рис. 3.4).

У верхній частині вікна інспектора об'єктів розташовується список, що містить всі розташовані на формі компоненти (див. рис. 3.4, а, б). Оскільки форма є компонентом, її ім'я також є у цьому списку.

Сторінка *Properties* необхідна для встановлення потрібних властивостей компонентів. Сторінка *Events* дозволяє визначити реакцію компонентів на ту чи іншу подію. Якщо вікно інспектора об'єктів опиниться не в полі зору, натисніть F11 або оберіть опцію *View → Object Inspector*, щоб воно знову з'явилося на екрані.

3.3 Проєкти у середовищі *Delphi*

Проєкти у середовищі *Delphi* допомагають реалізувати вирішення певного завдання за допомогою комп'ютера. Отже, проєкт – це досить складний елемент, який складається з різних файлів, кожен з яких виконує певні функції. Сукупність файлів, необхідних для створення програми, називається проєктом. Компілятор послідовно обробляє файли проєкту та будує програму (exe-файл).

Файли проєкту поділяються на типи, а саме:

- *.dpr* – файл проєкту;
- *.pas* – модулі програми, що містять код на *Object Pascal*;
- *.dfm* – модулі програми, що містять інформацію про вікно форми;
- *.res* – файли ресурсів (наприклад, іконки);
- *.obj* – файли, що містять готовий для компіляції програмний код;
- *.cfg, .dof, .dsk* – службові файли проєкту *Delphi*.

Перед тим як відкрити середовище *Delphi* та розпочати розробку нового проєкту, необхідно створити особисту папку, в якій зберігатиметься ваш проєкт. Нагадаємо, що перше збереження ще порожнього проєкту виконується одразу після відкриття середовища *Delphi* командою *File → SaveProjectAs*.

Після збереження проєкту у вашій папці утворюється одразу шість файлів, а після компіляції проєкту їх стане вісім (див. рис. 3.5).

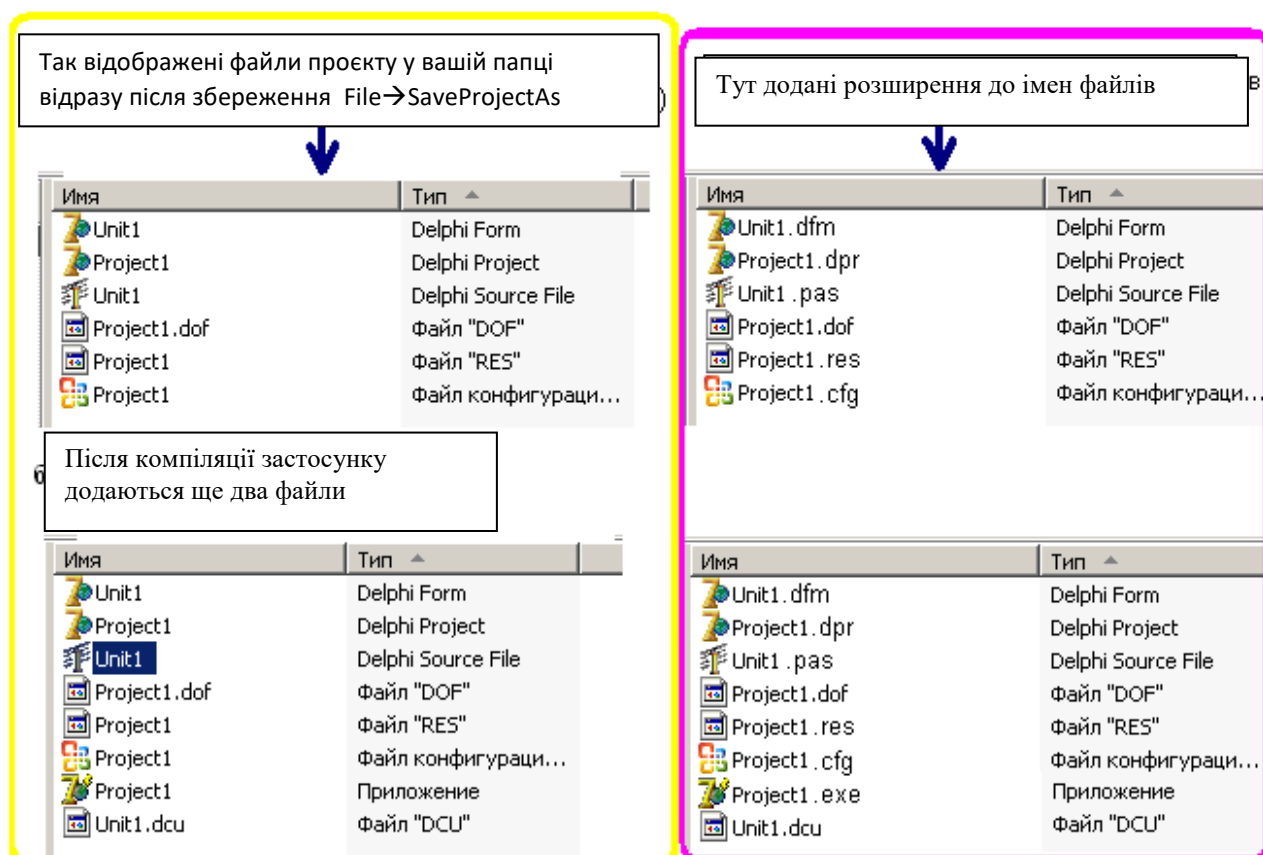


Рисунок 3.5 – Файли збереженого проєкту

Компіляція програми – це переклад вихідної програми в еквівалентну їй результуючу програму мовою машинних команд. Інакше кажучи, компілятор перетворює код програми, який написаний «людською» мовою, в об'єктний код, зрозумілий комп'ютеру.

3.4 Розташування елементів на сторінці панелі *Standard*

З назви вкладки зрозуміло, що компоненти, представлені в ньому, є стандартними та системними (рис. 3.6).

Label – текстова мітка (напис) на формі. Використовується для відображення будь-якого тексту у вікні. Текст є значенням властивості *Caption*. Властивість *Font* дозволяє налаштувати шрифт.

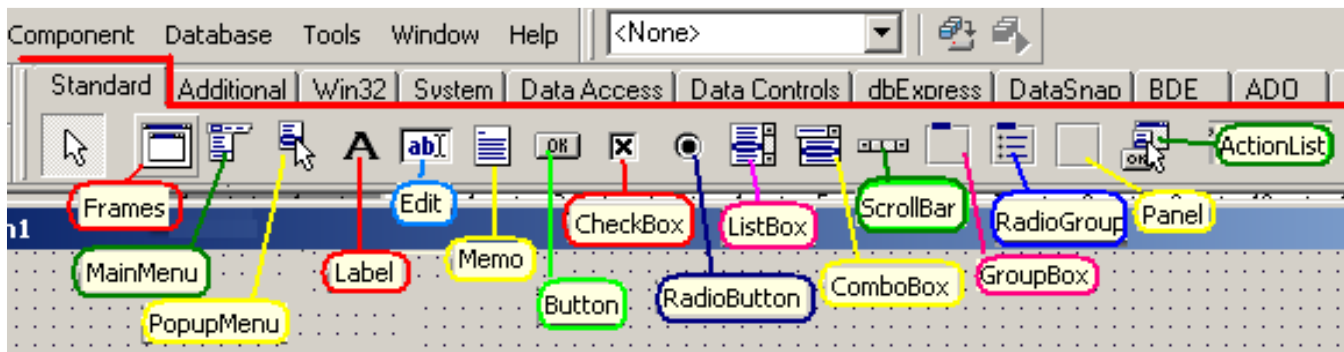


Рисунок 3.6 – Сторінка *Standard* палітри компонентів

Edit – поле введення тексту. Використовується для введення будь-яких даних (тексту, числа тощо), представлених одним рядком. Властивість *Read Only* дозволяє заборонити редагування тексту в полі. Текст зберігається як істю *Text*. Властивість *MaxLength* дозволяє вказати максимальну кількість символів, які можуть бути введені у поле. Значення 0 означає, що обмеження не встановлено.

GroupBox – «контейнер» для компонентів. Використовується для об'єднання елементів окремої області. При переміщенні *GroupBox* всі компоненти, розташовані на ньому, також переміщуються. У *GroupBox* можна створити текст заголовка – властивість *Caption*.

Panel – практично такий же контейнер, як і *GroupBox*, проте без заголовка та з можливістю відімкнення рельєфних кордонів.

Зазвичай застосовується під час створення панелей інструментів.

Увага! Щоб при наведенні миші на компонент з'являвся напис (підказка) з назвою компонента, повинен бути виставлений прапорець в пункті *ShowHints* у контекстному меню, що викликається на палітрі компонентів (див. рис. 3.7).

Далі більш детально наведені дії налаштування середовища розробки:

1. Увімкнути показ підказок у середовищі розробки:

- на панелі компонентів клацніть правою кнопкою миші;
- у контекстному меню оберіть пункт *Show Hints* (показувати підказки);
- поставте прапорець навпроти цього пункту, якщо він ще не встановлений.

Це дозволить відображати підказки при наведенні курсора на компонент.

2. Налаштувати властивість *Hint* компонента:

- виділіть потрібний компонент на формі;
- в інспекторі об'єктів знайдіть властивість *Hint*;
- у полі цієї властивості введіть текст підказки, наприклад, назву компонента або короткий опис його призначення.

3. Увімкнути властивість *ShowHint* для конкретного компонента:

- також в інспекторі об'єктів знайдіть властивість *ShowHint* для цього компонента;
- встановіть її значення в *True* – це активує відображення підказки на формі під час виконання програми.

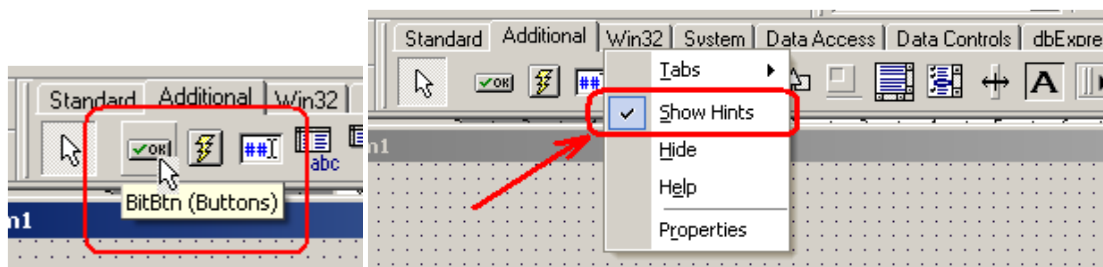


Рисунок 3.7 – Увімкнення «підказки, що спливає»

Властивості є атрибутами компонента, що визначають його зовнішній вигляд та поведінку.

3.4.1 Основні властивості компонентів

Властивості компонента відображаються на сторінці властивостей (*Properties*) в інспекторі об'єктів.

Основні властивості компонентів:

Name – ідентифікатор об'єкта, ім'я компонента, яким до нього можна звертатися з програми.

Caption – напис на об'єкті.

Text – рядок, пов'язаний із компонентом.

Font – шрифт, складна властивість із безліччю вкладених у нього властивостей.

Color – колір у форматі *RRGGBB*.

Visible – видимість об'єкта (відображення компонента).

Alignment – визначає вирівнювання тексту по горизонталі області мітки. Може набувати значення *taLeftJustify*, *taRightJustify* та *taCenter*.

Layout – визначає вирівнювання тексту по вертикалі. Може набувати значення *tlTop*, *tlCenter* та *tlBottom*.

Align – спосіб вирівнювання компонента.

– *AlNone* – не використовувати;

– *AlTop* – по верхньому краю, ширина на весь контейнер;

– *AlBotton* – по нижньому краю, ширина на весь контейнер;

– *AlLeft* – по лівому краю, висота на весь контейнер;

– *AlRight* – по правому краю, висота на весь контейнер;

– *AlClient* – на весь контейнер;

– *AlCustom* – розміри та положення визначаються розробником;

Enabled – доступність об'єкта, активність (здатність реагувати на дії користувача);

Height – висота компонента;

Width – ширина компонента;

Top – координата зверху;

Left – координата зліва;

AutoSize – визначає, чи повинен компонент змінювати свої розміри залежно від змісту (наприклад, тексту);

Transparent – визначає прозорість тла мітки;

WordWrap – визначає перенесення напису на наступні рядки.

3.5 Порядок виконання роботи

Eman 1. Створення першого проєкту.

1. Створити новий проєкт.

2. Розмістити на формі такі компоненти:

Panel – панель для групування елементів.

GroupBox – контейнер з рамкою та заголовком.

Label – для відображення статичного тексту.

PopupMenu – контекстне меню для форми або компонентів.

ScrollBar – горизонтальна або вертикальна прокрутка.

Image – для відображення зображення.

Memo – багаторядкове текстове поле.

Button – кнопка для взаємодії з користувачем.

Shape – графічна фігура (коло, прямокутник тощо).

3. Налаштувати компоненти на формі.

– розташувати компоненти зручно для візуального сприйняття;

– додати деякий початковий текст до *Label* та *Memo*;

– встановити зображення для компонента *Image*;

– додати пункти до *PopupMenu*;

– встановити події для *Button* (наприклад, змінити текст у *Label*);

– визначити тип і розмір *Shape*.

Eman 2. Створення копії проєкту.

1. Зберегти перший проєкт.

2. Створити його копію під іншою назвою (наприклад, «Проект_змінений»).

3. У копії змінити основні властивості кожного компонента через інспектор об'єктів:

- *Panel* – змінити колір тла, ширину або висоту;
- *GroupBox* – змінити заголовок, шрифт;
- *Label* – змінити текст, колір тексту, шрифт;
- *PopupMenu* – додати або змінити пункти меню;
- *ScrollBar* – змінити напрям, максимальне значення;
- *Image* – змінити зображення або його розмір;
- *Memo* – змінити колір тла, шрифт тексту;
- *Button* – змінити заголовок, колір, розмір кнопки;
- *Shape* – змінити тип фігури, колір заливки або рамки.

3.6 Оформлення звіту

1. У звіті показати два окремі проекти: базовий – із типовими налаштуваннями компонентів; змінений – з оновленими властивостями компонентів через інспектор об'єктів.

2. Пояснити роль кожного компонента та способи його налаштування.

3. Пояснити практику роботи з інспектором об'єктів та візуальним редактором форми.

4. Під яас захисту практичної роботи знати файли, що становлять структуру проекту та основні властивості компонентів, з яких складається проект.

3.7 Контрольні запитання

1. Для чого призначений компонент *Panel* у графічному інтерфейсі користувача?

2. Чим відрізняється *GroupBox* від *Panel*?
3. Для чого використовується компонент *Label* і які його основні властивості?
4. Для чого призначений *PopupMenu* і як він пов'язується з іншими компонентами?
5. Які формати зображень підтримує компонент *Image*?
6. Для чого використовується *Memo* і чим він відрізняється від *Edit*?
7. Які події обробляє компонент *Button*?
8. Які типи фігур можна створити за допомогою компонента *Shape*?
9. Як за допомогою інспектора об'єктів змінити властивості компонента (наприклад, колір, шрифт, розміри)?
10. Які основні властивості впливають на розташування компонента на формі?
11. Як за допомогою *Panel* згрупувати інші компоненти?
12. Які властивості дозволяють змінити зовнішній вигляд кнопки (наприклад, *Caption*, *Font*, *Color*)?
13. Як реалізувати взаємодію між *ScrollBar* і компонентом *Memo* (наприклад, прокручування тексту)?
14. Яким є призначення властивості *Enabled* будь-якого компонента?
15. Як продублювати проєкт і змінити властивості компонентів у копії?
16. Які компоненти потрібно використовувати для виведення інформації користувачу, а які – для введення?

ПРАКТИЧНЕ ЗАНЯТТЯ 4 СИСТЕМИ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ НА ПІДПРИЄМСТВІ

4.1 Мета заняття

Вивчення принципів побудови та функціонування систем підтримки прийняття рішень (*СППР/DSS*), застосовування моделей та методів для вирішення типових управлінських завдань із використанням *DSS*.

4.2 Необхідний теоретичний матеріал

Система підтримки прийняття рішень (*СППР*, англ. *Decision Support System, DSS*) – це комп'ютеризована інформаційна система, яка допомагає менеджерам, керівникам і фахівцям приймати рішення в умовах невизначеності, браку інформації, складності або багатокритеріальності завдань. Головна мета *СППР* – підвищення ефективності управлінських рішень шляхом збору, аналізу та обробки великих обсягів даних, а також моделювання різних сценаріїв розвитку подій.

4.2.1 Основні характеристики системи підтримки прийняття рішень

Робота з неструктурованими та слабоструктурованими завданнями: призначені для підтримки прийняття рішень у складних ситуаціях, коли неможливо чітко формалізувати всі правила або алгоритми. Такі завдання зазвичай мають багато невизначеностей, багатокритеріальні, а також включають суб'єктивні оцінки. На відміну від традиційних систем, що працюють із структурованими даними, *СППР* допомагають аналізувати і приймати рішення тоді, коли стандартні алгоритми не діють.

Використання даних і моделей для аналізу ситуації *СППР* інтегрують два ключові складники:

- бази даних, які містять оперативну, історичну та зовнішню інформацію;
- бази моделей, що включають математичні, статистичні, оптимізаційні та імітаційні моделі для оцінки альтернатив, прогнозування та оптимізації рішень.

Така комбінація дозволяє системі не лише зберігати інформацію, а й виконувати складний аналіз і симуляції.

Підтримка, а не заміна процесу прийняття рішень людиною. СППР не приймають рішення замість користувача, а лише допомагають йому обґрунтувати вибір, надаючи необхідну інформацію, аналітику та інструменти для моделювання. Відповідальність за прийняття рішення залишається за людиною, що дозволяє враховувати інтуїцію, досвід і контекст, які важко формалізувати.

Гнучкість і адаптивність. СППР мають бути гнучкими, щоб адаптуватися до змін у зовнішньому середовищі, організаційній структурі та вимогах користувачів. Це забезпечує довготривалу ефективність системи і можливість її модифікації без значних витрат. Гнучкість також означає підтримку різних стилів прийняття рішень і можливість роботи як індивідуально, так і в групах.

Індивідуальне та групове використання. СППР можуть бути зорієнтовані як на одного користувача (наприклад, менеджера середньої ланки), так і на групу користувачів, які спільно приймають рішення. У групових СППР підтримуються механізми координації, обміну інформацією та колективного аналізу.

Інтерактивність і простота використання. Для ефективної підтримки процесу прийняття рішень СППР повинні мати зручний інтерфейс, що дозволяє користувачеві легко формулювати запити, отримувати результати, вносити зміни і проводити аналіз без наявності глибоких технічних знань.

Підтримка різних фаз процесу прийняття рішень. СППР допомагають на всіх етапах: від виявлення проблеми (інтелектуальна фаза), через формування і аналіз альтернатив (проектування), до вибору оптимального рішення (фаза вибору).

Таким чином, основні характеристики СППР забезпечують комплексну, гнучку і ефективну підтримку менеджерів у складних, багатокритеріальних та,

зазвичай, неструктурованих ситуаціях, поєднуючи потужність комп'ютерного аналізу з інтуїцією і досвідом людини.

4.2.2 Структура та компоненти системи підтримки прийняття рішень

Типова архітектура *СППР* включає такі основні компоненти:

- база даних: містить всю необхідну інформацію для аналізу (оперативні, історичні, зовнішні дані тощо);
- база моделей: набір математичних, статистичних, імітаційних моделей, які дозволяють аналізувати альтернативи, прогнозувати наслідки, оптимізувати вибір;
- інтерфейс користувача: забезпечує зручний доступ до даних і моделей, дозволяє формувати запити, отримувати результати, візуалізувати дані;
- підсистему управління знаннями (у розширених *СППР*): містить експертні правила, логіку, евристику, що підвищують інтелектуальні можливості системи.

4.2.3 Етапи процесу прийняття рішень у системі підтримки прийняття рішень

Згідно з класичною моделлю Герберта Саймона, процес прийняття рішень у *СППР* складається з чотирьох фаз:

1. Інтелектуальна фаза (*Intelligence*): пошук і виявлення проблеми, що потребує рішення.
2. Проєктування (*Design*): розробка і аналіз можливих альтернатив, моделювання наслідків.
3. Вибір (*Choice*): оцінка альтернатив і вибір оптимального рішення.
4. Упровадження (*Implementation*): реалізація обраного рішення в практиці.

4.2.4 Класифікація систем підтримки прийняття рішень

СППР можна класифікувати за різними ознаками:

1. За способом організації: індивідуальні, групові, корпоративні.
2. За типом підтримки: персональні, групові, організаційні.
3. За технологічною архітектурою: текст-орієнтовані, бази даних, електронні таблиці, моделюючі, експертні, комбіновані (*compound DSS*).
4. За рівнем інтелектуалізації: традиційні та інтелектуальні (на основі штучного інтелекту, *IDSS*).

4.2.5 Переваги використання систем підтримки прийняття рішень

СППР мають такі переваги:

1. Підвищення якості та обґрунтованості управлінських рішень.
2. Зменшення часу на аналіз ситуації та вибір альтернативи.
3. Можливість роботи з великими обсягами даних і складними багатокритеріальними завданнями.
4. Підвищення прозорості та аргументованості прийнятих рішень.

4.2.6 Застосування систем підтримки прийняття рішень

СППР використовуються у різних сферах: бізнесі, медицині, фінансах, логістиці, екології, державному управлінні тощо. Вони особливо корисні для завдань прогнозування, оптимізації, планування, оцінки ризиків, вибору постачальників, управління запасами тощо.

Системи підтримки прийняття рішень – це сучасний інструмент для підвищення ефективності управління організацією, що поєднує можливості комп'ютерної обробки даних, моделювання та експертних знань для вирішення складних управлінських завдань в умовах невизначеності.

4.3 Завдання для виконання

1. Вивчити основні поняття, структуру та функції *СППР*.
2. Обрати приклад управлінського завдання (наприклад, вибір постачальника, оптимізація запасів, планування виробництва).
3. Сформулювати завдання прийняття рішення: визначити цілі, критерії, альтернативи, обмеження.
4. Побудувати модель прийняття рішення (наприклад, матриця рішень, дерево рішень, метод аналізу ієрархій).
5. Використати *DSS* (або її імітацію в *Excel* чи іншому *ПЗ*) для аналізу альтернатив і вибору оптимального рішення.
6. Проаналізувати отримані результати, зробити висновки щодо ефективності використання *СППР*.

4.4 Порядок виконання завдання

Завдання 1. Ознайомтеся з теоретичними засадами *СППР*: визначення, призначення, типи завдань, які вирішуються за допомогою *DSS*, основні компоненти (інтерфейс користувача, база моделей, база даних тощо).

Вивчіть типову структуру процесу прийняття рішень: ідентифікація проблеми, формування альтернатив, вибір критеріїв, оцінка альтернатив, вибір рішення.

Завдання 2. Формулювання прикладного завдання:

- оберіть реальне або умовне управлінське завдання (наприклад, вибір постачальника для підприємства);
- визначте ціль завдання (наприклад, мінімізувати витрати, забезпечити якість, скоротити терміни постачання);
- складіть перелік альтернатив (наприклад, постачальник А, Б, В);
- визначте критерії оцінки (ціна, якість, термін доставки, надійність).

У таблиці 4.1 наведено приклад вибору постачальника за критеріями.

Таблиця 4.1 – Вибір постачальника для підприємства

Постачальник (альтернатива)	Ціна 40 %	Якість 30 %	Термін доставки 20 %	Надійність 10 %
А	8	7	9	6
Б	6	9	8	8
В	7	8	7	9

Оцінки розраховуються за 10-бальною шкалою. Виконується розрахунок зваженої оцінки для кожного постачальника. Постачальник обирається виходячи з найвищої сумарної оцінки.

Завдання 3. Побудова моделі прийняття рішень.

Як уже зазначалося, матриця прийняття рішень є корисним інструментом, коли важко вибрати один варіант. Однак різні варіанти мають бути порівнюваними. Якщо вам потрібно ухвалити рішення, наприклад таке: «У якому напрямі має розвиватися маркетингова стратегія нашої компанії протягом найближчих двох років?», то за допомогою сітки рішень ви нічого не зможете зробити.

Використовуйте аналіз матриці рішень, якщо це потрібно:

- порівняти кілька схожих варіантів;
- звузити деякі варіанти до остаточного рішення;
- оцінити декілька важливих факторів;
- знайти логічне та раціональне рішення.

Таким чином, матриця рішень – це правильний вибір, якщо наявні багато критеріїв ухвалення рішення та декілька варіантів, які можна порівняти.

Оберіть відповідну модель (наприклад, матриця рішень або метод аналізу ієрархій).

Побудуйте матрицю рішень, де рядки – альтернативи, стовпці – критерії, а значення – оцінки альтернатив за кожним критерієм.

Встановіть значущість критеріїв (наприклад, ціна – 40 %, якість – 30 %, термін доставки – 20 %, надійність – 10 %).

Завдання 4. Аналіз даних за допомогою DSS:

1. Введіть дані у вибрану систему *DSS* або використайте *Excel* для обчислення зважених сум із кожної альтернативи.

2. Проведіть аналіз чутливості: змініть значущість критеріїв чи оцінки, проаналізуйте, як це вплине на вибір рішення.

3. Визначте оптимальну альтернативу за результатами аналізу.

Завдання 5. Висновки:

1. Оцініть ефективність застосованої *DSS*: допомогла система структурувати вибір чи забезпечила прозорість і аргументованість рішення?

2. Запропонуйте рекомендації щодо поліпшення моделі чи використання інших методів у майбутньому.

4.5 Контрольні запитання

1. Що таке система підтримки прийняття рішень? Які її основні функції?

2. Які етапи процесу прийняття рішень ви знаєте?

3. Які моделі використовуються в *СППР* для вибору альтернатив?

4. Як формулюються критерії та альтернативи у завданнях прийняття рішень?

5. Якими є переваги використання *СППР* порівняно з інтуїтивним прийняттям рішень?

6. Як можна оцінити ефективність упровадження *СППР* на підприємстві?

7. Які основні компоненти має сучасна *СППР*?

8. Які існують обмеження у процесі прийняття рішень, що передбачають застосування *СППР*?

ПРАКТИЧНЕ ЗАНЯТТЯ 5 СИСТЕМИ УПРАВЛІННЯ ЛАНЦЮГОМ ПОСТАВОК НА ПІДПРИЄМСТВІ

5.1 Мета заняття

Ознайомитися із засадами систем управління ланцюгом поставок, їх структурою, функціями та роллю в оптимізації логістичних процесів підприємства. Отримати практичні навички моделювання та аналізу ланцюга поставок із використанням *SCM*-систем.

5.2 Необхідний теоретичний матеріал

5.2.1 Поняття та суть управління ланцюгами поставок

Управління ланцюгами поставок (*SCM*) – це комплексна система стратегічного управління, яка охоплює планування, організацію, координацію та контроль усіх процесів, пов'язаних із рухом матеріальних, інформаційних і фінансових потоків від постачальників сировини до кінцевих споживачів. Мета *SCM* – забезпечити своєчасне, ефективне та економічно вигідне постачання товарів і послуг, задовольняючи потреби ринку та підвищуючи конкурентоздатність підприємства.

SCM включає інтеграцію і координацію діяльності різних учасників ланцюга поставок – постачальників, виробників, логістичних операторів, дистриб'юторів, роздрібних продавців і споживачів:

1. Основні компоненти та потоки ланцюга поставок.

Компоненти ланцюга поставок:

- постачальники сировини та комплектуючих;
- виробничі підприємства;
- логістичні центри та склади;
- транспортні компанії;

- дистриб'ютори та роздрібні мережі;
- кінцеві споживачі.

2. Види потоків:

- матеріальні – рух сировини, напівфабрикатів, готової продукції;
- інформаційні – замовлення, звіти, планування, обмін даними між учасниками;
- фінансові – платежі, кредитування, інвестиції.

3. Основні цілі та завдання *SCM*:

- забезпечення високої якості обслуговування клієнтів;
- оптимізація запасів і зниження логістичних витрат;
- скорочення часу доставки та підвищення гнучкості ланцюга;
- підвищення прозорості та контролю над процесами;
- координація дій між учасниками ланцюга для уникнення конфліктів і дублювань;
- підвищення стійкості ланцюга поставок до зовнішніх ризиків і змін ринку;

4. Основні функції управління ланцюгами поставок:

- *планування*: визначення джерел постачання, прогнозування попиту, планування виробництва і запасів;
- *закупівлі*: вибір постачальників, ведення переговорів, оформлення замовлень;
- *виробництво*: організація виробничих процесів відповідно до планів;
- *логістика*: управління зберіганням, транспортуванням, розподілом продукції;
- *обслуговування клієнтів*: забезпечення своєчасної доставки, підтримка замовлень і повернень;
- *контроль і моніторинг*: аудит, аналіз ефективності, управління якістю і ризиками.

5. Моделі та концепції *SCM*:

– *SCOR-модель (Supply Chain Operations Reference)*: включає п'ять основних процесів: планування, постачання, виробництво, доставку, повернення; дозволяє стандартизувати управління ланцюгом поставок і оцінювати ефективність;

– *GSCF-модель (Global Supply Chain Forum)*: фокусується на інтеграції ключових бізнес-процесів від кінцевого споживача до постачальника, включаючи продажі, обслуговування, планування, виробництво, постачання і повернення;

– *інтеграція бізнес-процесів*: *SCM* передбачає тісну співпрацю і обмін інформацією між усіма учасниками ланцюга для досягнення спільних цілей.

6. Вплив зовнішніх і внутрішніх факторів.

Управління ланцюгами поставок має враховувати:

- змінювання споживчого попиту та ринкові тенденції;
- глобалізацію і розвиток нових ринків;
- технологічні інновації і цифровізацію;
- економічні та політичні ризики;
- екологічні та соціальні вимоги (сталий розвиток, корпоративна відповідальність тощо).

7. Сучасні інформаційні технології в *SCM*:

– *ERP-системи (Enterprise Resource Planning)*. Призначені для інтеграції внутрішніх процесів;

– *WMS (Warehouse Management Systems)*. Використовуються для управління складами;

– *TMS (Transportation Management Systems)*. Призначені для планування і контролю транспортування.

– *системи електронного обміну даними (EDI)*. Використовуються для швидкого обміну інформацією між партнерами;

– *аналітичні інструменти і Big Data*. Призначені для прогнозування і оптимізації.

8. Значення *SCM* для бізнесу.

Ефективне управління ланцюгом поставок дозволяє підприємствам:

- знизити загальні витрати на виробництво і логістику;
- підвищити якість обслуговування клієнтів і швидкість реакції на ринок;
- зменшити ризики перебоїв і збільшити стійкість бізнесу;
- забезпечити конкурентні переваги через гнучкість і інновації.

5.3 Завдання для виконання

1. Вивчити теоретичні основи *SCM*: поняття, цілі, складові ланцюга поставок.
2. Проаналізувати структуру ланцюга поставок конкретного підприємства (реального або умовного).
3. Визначити основні матеріальні, інформаційні та фінансові потоки в ланцюгу поставок.
4. Сформувати модель ланцюга поставок із урахуванням ключових учасників і процесів.
5. Виконати оптимізацію логістичних процесів (наприклад, вибір постачальника, планування запасів, маршрутизація доставки) із застосуванням *SCM*-підходів.
6. Оцінити ефективність запропонованих рішень.
7. Підготувати звіт із результатами аналізу та рекомендаціями.

5.4 Порядок виконання завдання

Завдання 1. Теоретичне ознайомлення:

1. Вивчити основні поняття *SCM*: що таке ланцюг поставок, які функції виконує *SCM*-система, які компоненти входять до ланцюга поставок (постачальники, виробники, дистриб'ютори, роздрібні продавці, клієнти).
2. Ознайомитися з типами потоків у ланцюгу поставок: матеріальними, інформаційними, фінансовими.

3. Вивчити роль *SCM* для підвищення ефективності бізнесу, зниження витрат, поліпшення обслуговування клієнтів.

Завдання 2. Аналіз ланцюга поставок підприємства:

1. Обрати підприємство (реальне або умовне) для аналізу.
2. Визначити основних учасників ланцюга поставок і їхні функції.
3. Описати потоки матеріалів (сировина, комплектуючі, готова продукція), інформації (замовлення, звіти, плани) і фінансів (платежі, кредити).

Завдання 3. Моделювання ланцюга поставок:

1. Побудувати схему ланцюга поставок із зазначенням учасників і напрямів потоків.
2. Визначити ключові процеси: закупівлі, виробництво, зберігання, транспортування, розподіл.
3. Визначити параметри кожного процесу (терміни, обсяги, витрати).

Завдання 4. Оптимізація логістичних процесів:

1. Визначити проблему для оптимізації (наприклад, мінімізація витрат на транспортування, зменшення часу доставки, оптимізація запасів).
2. Застосувати методи *SCM* (наприклад, вибір оптимального постачальника за критеріями, планування запасів за моделлю *EOQ*, розробка маршруту доставки).
3. Використати таблиці, діаграми або спеціалізоване програмне забезпечення (за наявності) для розрахунків.

Завдання 5. Оцінка результатів:

1. Порівняти початкові і оптимізовані показники (витрати, час, рівень обслуговування).
2. Проаналізувати вплив запропонованих змін на ефективність ланцюга поставок.
3. Визначити можливі ризики та обмеження.

Завдання 6. Підготовка звіту:

1. Описати мету і завдання роботи.
2. Представити теоретичні засади *SCM*.

3. Надати аналіз ланцюга поставок підприємства.
4. Показати модель і результати оптимізації.
5. Викласти висновки та рекомендації.

5.5 Контрольні запитання

1. Що таке система управління ланцюгом поставок (*SCM*) і які її основні цілі?
2. Які основні компоненти входять до ланцюга поставок?
3. Які види потоків існують у ланцюгу поставок?
4. Яку роль відіграє *SCM* у підвищенні ефективності підприємства?
5. Які основні функції виконує *SCM*-система?
6. Як можна оптимізувати логістичні процеси в ланцюзі поставок?
7. Які методи планування запасів використовуються в *SCM*?
8. Як інформаційні технології сприяють управлінню ланцюгом поставок?
9. Які ризики можуть виникати в процесі управління ланцюгом поставок?
10. Як оцінити ефективність упровадження *SCM*-системи на підприємстві?
11. Що таке ланцюг поставок? Які основні елементи він включає?
12. Які основні цілі та завдання систем управління ланцюгом поставок ?
13. Які види потоків існують у ланцюгу поставок? Наведіть приклади.
14. Які функції виконує *SCM*-система на підприємстві?
15. Які основні учасники ланцюга поставок і які їхні ролі?
16. Що таке матеріальні, інформаційні та фінансові потоки в *SCM*?
17. Як *SCM* впливає на ефективність бізнес-процесів підприємства?
18. Які сучасні інформаційні технології використовуються в управлінні ланцюгом поставок?

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Долбнєва Д. В. Основи роботи і функціональні можливості системи IT-Enterprise з ведення обліку, звітності та оподаткування : навч.-метод. посіб. / Д. В. Долбнєва, Є. М. Романів. – Львів : Ліга-Прес, 2021. – 149 с. – Існує електрон. версія. (Режим доступу: <https://financial.lnu.edu.ua/wp-content/uploads/2020/02/Posibnyk-IT-Bukhhalteriia-Dolbnieva-Romaniv.pdf>, вільний).
2. Новінський В. Реалізація «Enterprise Resource Planning» засобами «Business Process Management System» : навч. посіб. / В. Новінський, С. Пустовіт ; Нац. техн. у-т «КПІ ім. Ігоря Сікорського». – Київ : КПІ ім. Ігоря Сікорського, 2024. – 105 с.
3. Павленко Т. В. Маркетинг персоналу та HR-брендинг [Електрон. ресурс] : конспект лекцій / Т. В. Павленко ; КПІ ім. Ігоря Сікорського. – Електрон. текст. дані. – Київ : КПІ ім. Ігоря Сікорського, 2022. – 97 с. – Режим доступу: <https://ela.kpi.ua/server/api/core/bitstreams/7664a234-4e87-4753-bdf6-e377cb6d68bf/content>, вільний (дата звернення: 15.05 2025). – Назва з екрана.
4. IT-Enterprise [Електрон. ресурс] : сайт. – Електрон. текст. дані. – Київ, 2025. – Оновлюється постійно. – Режим доступу: <https://www.it.ua/>, вільний (дата звернення: 12.05.2025). – Назва з екрана.
5. Що таке CRM: різновиди та застосування у бізнесі [Електрон. ресурс] : сайт. – Електрон. текст. дані. – Харків, 2025. – Оновлюється постійно. – Режим доступу: <https://torgsoft.ua/articles/stati/crm-systema-dlja-biznesu/>, вільний (дата звернення: 10.04.2025). – Назва з екрана.
6. Задачі аналізу даних в теорії прийняття рішень. Практикум. Частина 1 : навч. посіб. [Електрон. ресурс] / КПІ ім. Ігоря Сікорського ; уклад. ; [О. В. Гавриленко, В. В. Онищенко]. – Електрон. текст. дані. – Київ : КПІ ім. Ігоря Сікорського, 2024. – 110 с. – Режим доступу: <https://ela.kpi.ua/server/api/core/bitstreams/d47e0af0-c86e-4356-bcd3-a06ee3050743/content>, вільний (дата звернення: 12.05 2025). – Назва з екрана.

7. Тоценко В.Г. Методи та системи підтримки прийняття рішень. Алгоритмічний аспект / В.Г. Тоценко. – Київ : Наукова думка, 2022. – 381 с.

8. Луценко І.С. Управління ланцюгами поставок : навч. посіб. / І.С. Луценко ; Електронне мережне навчальне видання. – Київ : КПІ ім. Ігоря Сікорського, 2022. – 175 с. – Існує електрон. версія. (Режим доступу: <https://ela.kpi.ua/server/api/core/bitstreams/69cafc6d-f794-4275-a563-f8e6156c7bf1/content>, (дата звернення: 12.05 2025)). – Назва з екрана.

Електронне навчальне видання

Методичні рекомендації
до проведення практичних занять
із навчальної дисципліни

**« ІНФОРМАЦІЙНЕ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ
ДІЯЛЬНОСТІ ПІДПРИЄМСТВА »**

*(для здобувачів першого (бакалаврського) рівня вищої освіти зі спеціальностей
122, F3 – Комп'ютерні науки, освітньо-професійна програма «Комп'ютерні
науки»)*

Укладачі: **МІРОШНИК** Марина Анатоліївна,
ПАХОМОВ Юрій Васильович

Відповідальний за випуск *М. В. Булаєнко*
Редактор *О. А. Норик*
Комп'ютерне верстання *Ю. В. Пахомов*

План 2024, поз. 289М

Підп. до друку 20.05.2025. Формат 60 × 84/16.
Ум. друк. арк. 3,0.

Видавець і виготовлювач:
Харківський національний університет
міського господарства імені О. М. Бекетова,
вул. Чорноглазівська, (Маршала Бажанова), 17, Харків, 61002.
Електронна адреса: office@kname.edu.ua
Свідоцтво суб'єкта видавничої справи:
ДК № 5328 від 11.04.2017.