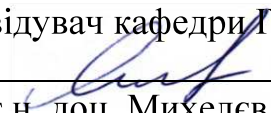


МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ
ІМЕНІ АДМІРАЛА МАКАРОВА

Навчально-науковий інститут комп'ютерних наук та управління проектами
Кафедра інформаційних управляючих систем та технологій

"Допущений до захисту"

Завідувач кафедри ІУСТ


к.т.н, доц. Михелєв І.Л.

"__" "____" 2024 р.

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття ступеня вищої освіти "магістр"

(освітньо-кваліфікаційний рівень)

на тему **Системний аналіз технологій моніторингу та розробка
інформаційної системи аналізу стану агротехнічного підприємства**

Виконав: студент 6 курсу, групи 6145м
напряму підготовки (спеціальності)

124 – "Системний аналіз".

(шифр і назва напряму підготовки, спеціальності)

Сербул Д.В.

(прізвище та ініціали)

Керівник Казимиренко Ю.О.

(прізвище та ініціали)

м. Миколаїв – 2024 р.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ
ІМЕНІ АДМІРАЛА МАКАРОВА

Інститут, факультет Навчально-науковий інститут комп'ютерних наук та управління проектами

Кафедра Інформаційних управляючих систем та технологій

Рівень вищої освіти другий (магістерський)

Спеціальність 124 – "Системний аналіз"

Освітня програма "Системний аналіз"

"ЗАТВЕРДЖУЮ"

Гарант освітньої програми

д.ф-м.н, проф. Ушкац М.В.

"___" _____ 2024 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ
на здобуття ступеня вищої освіти "магістр"

Студенту Сербул Дмитрій Віталійович

1. Тема проекту (роботи) Системний аналіз технологій моніторингу та розробка інформаційної системи аналізу стану агротехнічного підприємства

керівник проекту (роботи) Казимиренко Юлія Олексіївна,

затверджені наказом вищого навчального закладу від "08" 10 2024 року № 1055-уч.

2. Строк подання студентом проекту (роботи) 15 грудня 2024 року.

3. Вихідні дані до проекту (роботи) ДСТУ щодо обробки інформації, літературні джерела, технічна документація на існуючі аналоги систем, матеріали практики.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити):

- Титульний аркуш, завдання на дипломну роботу, анотація (українською, англійською), зміст, перелік умовних означень, символів, одиниць та термінів (при необхідності).

- Вступ.

- Аналіз предметної області і постановка задачі.

- Розробка проекту автоматизованої системи.

- Реалізація проекту автоматизованої системи (технічна документація по проекту).

- Розділ з охорони праці.

- Висновки.
- Список використаної літератури.
- Додатки: загальний опис підсистеми, керівництво користувача, керівництво адміністратора, програма і методика випробувань, графічні інтерфейси, код програми.

5. Перелік графічного матеріалу

- 5.1 Схема організаційної структури аграрного підприємства.
- 5.2 Архітектура автоматизованої ІС.
- 5.3 Організаційно-функціональна структура автоматизованої ІС.
- 5.4 Інформаційна модель програмного забезпечення автоматизованої ІС.
- 5.5 Модель переходів станів програмного забезпечення автоматизованої ІС.
- 5.6 Модель технологічного забезпечення автоматизованої ІС.
- 5.7. Екранні форми.

6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1.	Казимиренко Ю.О.	05.09.2023	22.12.2023
2.	Казимиренко Ю.О.	23.12.2023	15.06.2024
3.	Казимиренко Ю.О.	15.06.2024	25.09.2024
4.	Казимиренко Ю.О.	25.09.2024	14.11.2024
5.	Гурець Н.В.	25.09.2024	14.11.2024
6.	Гурець Н.В.	25.09.2024	14.11.2024

7. Дата видачі завдання 05 вересня 2024 р.

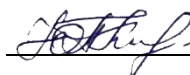
КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту (роботи)	Примітка
1.	Вивчення і опис предметної області	22.12.2024	
2.	Розробка концепції ІС	15.03.2024	
3.	Розробка проектних рішень по ІС	15.06.2024	
4.	Реалізація проекту, розробка робочої документації	25.09.2024	
5.	Вирішення питань з охорони праці	14.11.2024	
6.	Виконання графічної частини роботи	14.11.2024	
7.	Оформлення пояснювальної записки	14.11.2024	
8.	Подання роботи на кафедральний захист	16.11.2024	
9.	Подання роботи рецензенту	20.11.2024	
10.	Подання роботи на державний захист	15.12.2024	

Студент

 Сербул Д.В.

Керівник проекту (роботи)

 д.т.н., професор, Казимиренко Ю.О.

АНОТАЦІЯ

Темою дипломної роботи є "Системний аналіз технологій моніторингу та розробка інформаційної системи аналізу стану агротехнічного підприємства".

В якості предметної області для розробки проекту, проведення дослідження та аналізу було обрано аграрне підприємство, зокрема систему відстеження, фіксації та управління проблемами, що виникають на досліджуваному підприємстві.

В якості засобів реалізації було обрано мову програмування Java, проектування реалізовано за допомогою мови UML.

Метою роботи є оптимізація управління ризиками, зменшення витрат ресурсів, необхідних для забезпечення цих функцій, забезпечення можливості контролю за обробкою поточних заявок та усуненням наявних проблем.

Шляхом реалізації мети є розробка інформаційної системи моніторингу стану аграрного підприємства.

В роботі було проведено дослідження існуючих на сьогодні аналогів, аналіз існуючої організаційної структури підприємства та виявлені існуючі проблеми зі збору та обробки інформації, що потребують автоматизації. Розроблені проектні рішення по системі з різних видів забезпечення, передбачених етапами ескізного та технічного проектування, зокрема загальносистемні рішення, рішення з організаційного забезпечення, рішення з інформаційного забезпечення, рішення з програмного забезпечення. Розроблена робоча документація по спроектованій системі та виконано спеціальний розділ з охорони праці.

Робота виконана українською мовою та викладена на 161 сторінці, містить 21 рисунок, 25 таблиць та 4 додатки, які містять технічне завдання, інструкцію з використання, технічний опис та лістинг коду розроблюваного програмного забезпечення.

Ключові слова: системний аналіз; технології моніторингу; інформаційна система; агротехнічне підприємство; інтелектуальні системи; автоматизація агропідприємств; розробка інформаційних систем.

ABSTRACT

The theme of the current thesis is "System analysis of monitoring technologies and development of an information system for analyzing the state of an agricultural enterprise".

As a subject for the development project, conducting research and analysis was chosen agricultural enterprises, including a tracking system, recording and management of problems arising in the target company.

The aim is to optimize risk management, reduction of cost of resources needed to support these functions enable control over the processing of current applications and eliminating these problems.

Through the realization of the goals is to develop state monitoring system for agricultural enterprises.

In this paper, a study of the existing organizational structure and found the existing problems in gathering and processing information requiring automation. All requirements and restrictions to the developed workstation in terms of reference. The developed design solutions for a system with different types of support provided for the stages of conceptual and technical design, including system-wide solution, the solution of organizational software solutions for information security solutions for software. Developed working documentation system designed and implemented a special section on health.

Work carried out in the Ukrainian language and set out on pages 161, contains 21 pictures, 25 tables, and 4 applications that include the terms of reference, instruction for use, technical description and listing of the code of the developed software.

Keywords: system analysis; monitoring technologies; information system; agrotechnical enterprise; intelligent systems; automation of agricultural enterprises; development of information systems.

ЗМІСТ

Перелік умовних позначень, символів, одиниць, скорочень і термінів	
Вступ	
1. Дослідження методів проектування інформаційних систем агротехнічних підприємств та аналіз їх особливостей	
1.1 Аналіз особливостей проектування інформаційних систем агротехнічних підприємств	
1.2 Аналіз існуючих методів та технологій проектування інформаційних систем аграрних підприємств	
1.3 Особливості проектування інформаційних систем в умовах обмежених ресурсів	
1.4 Підходи і методи розподілення ресурсів на стадії планування та аналізу вимог при проектуванні інформаційних систем	
1.5 Когнітивне моделювання розподілу обмежених ресурсів на стадії планування та аналізу вимог	
2. Формування вимог до ІС, розробка концепції і постановка задачі	
2.1 Опис предметної області	
2.2 Формування вимог до інформаційної системи	
2.3 Наявність існуючих рішень з автоматизації моніторингу підприємства	
2.4 Розробка концепції ІС	
2.5 Постановка задачі	
3. Розробка проектних рішень по ІС	
3.1 Загальносистемні проектні рішення	
3.2 Рішення по організаційному забезпеченню	
3.3 Рішення по інформаційному забезпеченню	
3.4 Рішення по технічному забезпеченню	
3.5 Опис алгоритмів функціонування системи	
3.6 Рішення по програмному забезпеченню	
4. Розробка робочої документації ІС моніторингу стану агротехнічного підприємства	
5. Розрахунок економічної ефективності впровадження інформаційної управляючої системи агротехнічного підприємства	
6. Охорона праці	
7. Охорона навколишнього середовища	
Висновки	
Список використаних джерел	
Додатки	
Додаток А	
Додаток Б	
Додаток В	
Додаток Г	

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

БД – база даних.

БЗ – база знань.

ЕО – експертна оцінка.

ІС – інформаційна система.

ОПР – особа, яка приймає рішення.

ППР – підтримка прийняття рішень.

СППР – система підтримки прийняття рішень.

СУБД – система управління базами даних.

ВСТУП

На сьогодні автоматизація виробництва та впровадження сучасних інформаційних технологій на підприємства дозволяє суттєво підвищити ефективність їх функціонування, оптимізувати та полегшити роботу. Однак, цей процес є доволі складним і вимагає значних зусиль, а іноді і перебудови основної структури.

Завданням даної дипломної роботи є дослідження методів та розробка інформаційної системи моніторингу стану агротехнічного підприємства, тобто для автоматизації процесу відстеження, фіксації та координації вирішення проблем на підприємстві. Функції, що автоматизуються системою, пов'язані з організацією робочого часу працівників, моніторингом, виявленням, усуненням зафіксованих проблем, розподіленням завдань між працівниками для контролю та виконання, а також документообігу, для звітування про стан підприємства.

Метою розробки інформаційної системи є:

- підвищення ефективності управління підприємством за рахунок оптимізації обігу інформації про його стан, наявні проблеми чи ускладнення;
- зменшення ризику помилок при прийнятті рішень через невірогідність інформації про функціонування підприємства;
- зменшення часу усунення проблем на підприємстві;
- зменшення витрат на документообіг, контроль і збереження інформації про виявленні недоліки, роботи з їх усунення, відповідальних осіб;
- подовження строків експлуатації основних засобів за рахунок вчасного їх ремонту, налагодження, виявлення вад та недоліків;
- централізований контроль та оптимізація керування та розподілення робіт з усунення проблем на всіх рівнях підприємства;

Для досягнення поставленої мети були вирішені наступні основні задачі:

- проведений аналіз роботи підрозділів підприємства;
- проведений аналіз існуючої системи улагодження проблем;
- виявлені вади поточної системи;

- проведений аналіз можливих альтернатив, існуючих на інших підприємствах;
- проведене дослідження методів теорії прийняття рішень для аналізу заявок в системі;
- проведений функціонально-вартісний аналіз, і обрана концепція майбутньої системи;
- досліджений економічний ефект від розробки програмного забезпечення;
- проведена оцінка пожежної обстановки у будівлі підприємства;
- проведені дослідження з екології та охорони праці;
- проведений аналіз основних проблемних ділянок підприємства;
- досліджені найбільш ефективні шляхи збору інформації.

Наукова новизна роботи полягає у централізації збору інформації про стан підприємства та оптимізації методів роботи с заявками, що надходять до підсистеми за допомогою методів теорії прийняття рішень, зокрема методом аналізу ієрархій.

Програмний продукт є унікальним і індивідуальним, розробляється на замовлення підприємства. Розробка не має прямих аналогів, однак існує значна кількість програмного забезпечення для моніторингу виробничих процесів різного роду. Втім, більшість з таким розробок також є індивідуальними та закритими, аналіз даних проводиться експертом, тому є уповільненим за рахунок значної кількості заявок, що надходять до системи.

1. ДОСЛІДЖЕННЯ МЕТОДІВ ПРОЕКТУВАННЯ ІНФОРМАЦІЙНИХ СИСТЕМ АГРОТЕХНІЧНИХ ПІДПРИЄМСТВ ТА АНАЛІЗ ЇХ ОСОБЛИВОСТЕЙ

1.1. Аналіз особливостей проектування інформаційних систем агротехнічних підприємств

1.1.1 Склад та структура інформаційних систем аграрних підприємств

В даний час склалася думка про інформаційну технологічну систему як про систему, реалізовану за допомогою комп'ютерної техніки. Хоча в загальному випадку інформаційну систему можна розуміти й у некомп'ютерному варіанті.

Перші інформаційні системи були створені для того, щоб допомогти людям працювати з великими обсягами інформації і складалися зі сховищ документів і персоналу, що займався обробкою інформації і відповідав за її збереження. Сучасне ж розуміння інформаційної системи припускає використання в якості основного технічного засобу переробки інформації персонального комп'ютера. Технічне втілення інформаційної системи саме по собі нічого не буде значити, якщо не врахована роль людини, для якої призначена вироблена інформація і без якої неможливо її одержання і представлення. Комп'ютери, обладнані спеціалізованими програмними засобами, є технічною базою й інструментом для інформаційних систем. Інформаційна система немислима без персоналу, взаємодіючого з комп'ютерами і телекомунікаціями.

Перші технології автоматизування інформаційних систем з'явилися в 50-х рр. У ці роки вони були призначені для обробки рахунків і розрахунку зарплати, а реалізовувалися на електромеханічних бухгалтерських рахункових машинах. Це приводило до деякого скорочення витрат і часу на підготовку паперових документів. 60-і рр. знаменуються зміною відношення до інформаційних систем. Інформація, отримана з них, стала застосовуватися для періодичної звітності по багатьом параметрам. Для цього організаціям було

потрібно комп'ютерне устаткування широкого призначення, здатне обслуговувати безліч функцій, а не тільки обробляти рахунки і розраховувати зарплату, як було раніше. У 70-х - початку 80-х рр. інформаційні системи починають широко використовуватися як засіб управлінського контролю, що підтримує і прискорює процес прийняття рішень. До кінця 80-х рр. концепція використання інформаційних систем знову змінюється. Вони стають стратегічним джерелом інформації і використовуються на всіх рівнях організації будь-якого профілю. Інформаційні системи цього періоду, надаючи вчасно потрібну інформацію, допомагають організації досягти успіху у своїй діяльності, створювати нові товари і послуги, знаходити нові ринки збуту, забезпечувати собі гідних партнерів, організовувати випуск продукції за низькою ціною і багато чого іншого.

Тож сучасні інформаційні технологічні системи стали невід'ємною частиною будь-якого підприємства, яке намагається досягти успіху.

З системного підходу структурні складові комп'ютерної інформаційної системи управління аграрного підприємства такі:

1) керуюча система, або суб'єкт управління — СУ (сукупність взаємодіючих структурних підрозділів підприємства, які здійснюють функції управління);

2) керована частина, або об'єкт управління — ОУ (підсистема матеріальних елементів діяльності (сировина, матеріали, устаткування, працівники) і господарських процесів (виробництво, постачання, збут);

3) інформаційна система — ІС, через яку відбувається зв'язок між СУ та ОУ (сукупність організаційних, технічних, програмних і інформаційних засобів, що об'єднані в єдину систему для реалізації мети і виконання задач).

Загальну схему взаємодії між цими елементами зображено на рисунку 1.1.

Елементами інформаційної системи є:

- інформаційна модель, що є сукупністю правил і алгоритмів функціонування. Інформаційна модель включає всі форми документів, структуру довідників і даних і т.д.:

- регламент розвитку інформаційної моделі і правил по внесенню до неї змін;
- програмний комплекс, конфігурація якого відповідає вимогам інформаційної моделі;
- апаратно-технічна база (комп'ютери, периферійні пристрої, канали зв'язку, системне ПО, СУБД).

Основними цілями інформаційних систем підприємств є збір, обробка, зберігання і представлення даних про діяльність організації і зовнішнє середовище у вигляді, що є зручним для фінансового і будь-якого іншого аналізу і для використання при ухваленні управлінських рішень.

Задачею інформаційної системи є автоматизація функцій управління, які підрозділяються на два класи:

- Загальне управління (встановлення нормативних вимог і політики управління, політики інновацій, прогнозування, планування, організація роботи, мотивація, координація, контроль, облік, аналіз);
- конкретні області управління (маркетинг, фінанси, виробництво, технологічні процеси, постачання, збут, кадри, основні фонди).

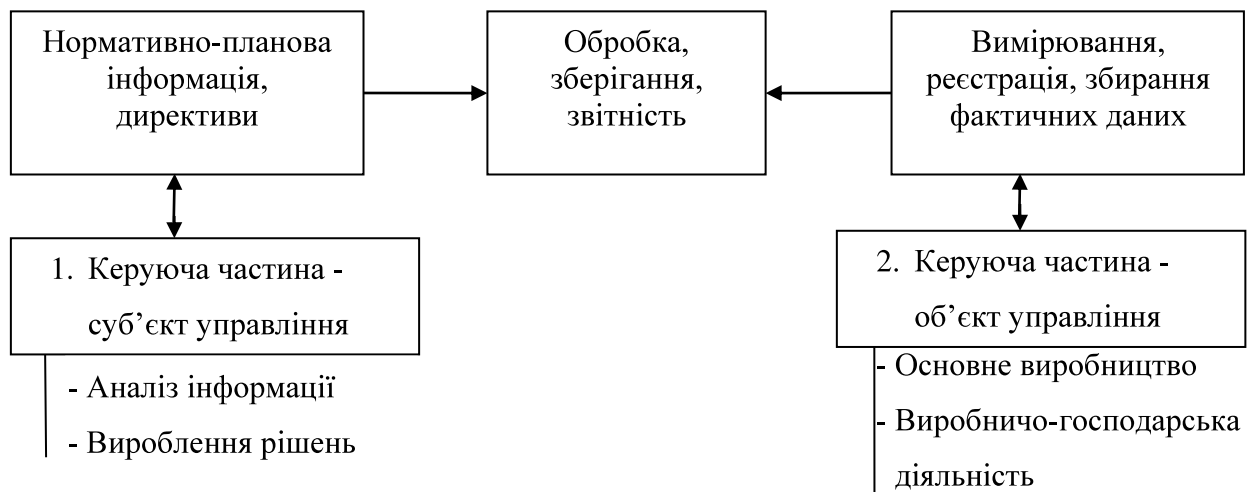


Рисунок 1.1.- Схема взаємодії між елементами системи управління

Різноманітність функцій управління, що реалізуються за допомогою інформаційних систем, привела до появи безлічі однотипних систем, які

відрізняються принципами побудови і обробки інформації. Класифікацію ІС на підприємствах зведемо до загальної таблиці (таблиця 1.1).

Таблиця 1.1. Класифікація інформаційних систем на підприємствах

Параметр класифікації	Типи інформаційних систем	Приклади
Тип даних, що зберігаються	Фактографічні	
	Документальні	
	Іконографічні	
Міра автоматизації інформаційних процесів	Ручні	
	Автоматичні	
	Автоматизовані	
Характер обробки даних	Інформаційно-пошукові	
	Інформаційно-вирішувальні	
Сфера використання	Інформаційні системи організаційного управління	
	Інформаційні системи управління технологічними процесами	
	Інформаційні системи автоматизованого проектування	
	Інтегровані інформаційні системи	
Рівень управління	Системи підтримки прийняття рішень	CORPORATE PLANNER, Project Expert
	Системи управління ефективністю бізнесу	Hyperion Pillar, Adaytum, Comshare MPC, Oracle Financial Analyzer, SyteLine Budgeting
	Інформаційні системи управління	
	Системи обробки даних	SAP R/3, Oracle Applications, BAAN, "Галактика", "Парус", "ИТ-предприятие"
	Системи управління в реальному часі	DCS, SCADA, PLC

Структура автоматизованої інформаційної системи підприємства – це сукупність взаємодіючих інформаційних систем, що забезпечують функції управління, які визначаються процесами виробничого циклу організації.

Система управління підприємств характеризується різноманітністю функцій управління. Для забезпечення їх інформаційною підтримкою потрібне використання різних класів інформаційних систем.

Для автоматизації проектування використовуються системи класу САПР. Вони в свою чергу підрозділяються на два підкласи: САПР В (САПР виробу) і САПР ТП (САПР технологічних процесів).

До підкласу САПР В відносяться системи розрахунків і інженерного аналізу CAE (Computer Aided Engineering) і системи конструкторського проектування CAD (Computer Aided Design). Система CAM (Computer Aided Manufacturing), в якій виконується проектування технологічних процесів, відноситься до класу САПР ТП.

Для забезпечення єдиного інформаційного простору між CAE/CAD/CAM системами використовується система управління проектними даними PDM (Product Data Management).

Інформаційна підтримка етапу виробництва продукції здійснюється класом автоматизованих систем управління підприємством (АСУП). До АСУП відносяться:

- системи планування і управління підприємством ERP (Enterprise Resource Planning);
- планування виробництва та вимог до матеріалів MRPII (Manufacturing Requirement Planning);
- виробнича виконавська система MES (Manufacturing Execution Systems);
- система управління постачаннями SCM (Supply Chain Management);
- система управління взаєминами із замовниками CRM (Customer Requirement Management).

Для розробки планів, розподілу ресурсів, відстежування прогресу і аналізу обсягів робіт використовується клас систем управління проектами (СУП).

До класу автоматизованих систем управління технологічними процесами (АСУТП) відносяться:

- SCADA (Supervisory Control and Data Acquisition), яка виконує збір і обробку даних про стан обладнання, технологічні процеси і розробку ПО для вбудованого обладнання;
- система CNC (Computer Numerical Control) здійснює програмне управління технічним обладнанням.

Серед найважливіших задач, що постають перед підприємством в сучасних умовах, можна виділити: підвищення конкурентної боротьби, необхідність оперативно реагувати на ускладнення у процесі вирощування культур чи не очікуванні погодні умови, необхідність випускати продукцію відповідно до поточних замовлень покупців, а не з довгостроковими перспективними планами, необхідність оперативного прийняття рішень у складній економічній ситуації, зміцнення зв'язків між постачальниками, виробниками і покупцями.

У конкурентній боротьбі перемагає тільки той, хто швидше за інших реагує на зміни в бізнесі і приймає більш вірні рішення. Саме інформаційні технології допомагають керівникам аграрних підприємств у вирішенні цих складних задач. Країни ринкової економіки мають великий досвід створення і розвитку інформаційних технологій для аграрних підприємств.

У таблиці 1.2 сформуємо основні функції та їх характеристики, які можливо автоматизувати за допомогою певних конкретних модулів інформаційних систем.

Таблиця 1.2. Функції, автоматизація яких можлива за участю інформаційних систем

Назва функції, яку необхідно автоматизувати за допомогою ІС	Детальний опис
Виробниче планування і управління	Проблеми обмеженості ресурсів часто стають стимулом для впровадження систем, що дозволяють ці ресурси заощаджувати. Головне завдання - ощадливе використання ресурсів і орієнтація на інтереси клієнтів.
Маркетинг і збут	Маркетинг - нова сфера діяльності, що

	потребує фахівців, що вміють просувати продукцію компанії, аналізувати продажі за різними критеріями і робити правильні висновки. Збут - це не обов'язковий розподіл продукції. Головне - забезпечення договірних обсягів, номенклатури і термінів постачань за погодженими цінами.
Управління запасами	Найчастіше планування поповнення запасів необґрунтовано ведеться по встановленим нормативам, незалежно від конкретної ситуації. У результаті треба встановлювати свідомо завищений рівень запасів, що по суті означає невиправдане відволікання фінансових ресурсів і зниження їхньої оборотності. В сучасних АІС управління запасами розглядається у нерозривному зв'язку з іншими ланками управління, що дозволяє визначати необхідний рівень запасів і стратегію їхнього поповнення у залежності від виробничої ситуації.
Облік та управління фінансами	Найбільша проблема - постановка методик калькуляції собівартості незавершеного виробництва та напівфабрикатів. Ці методики складні, вимагають трудомістких розрахунків. Ці питання також можуть бути вирішені на основі автоматизації, що розглядає фінансові і виробничі аспекти як частину цілісної системи планування.
Інформованість керівника	Якість управління підприємством визначається ефективністю рішень керівників. Гостро проблеми вірогідності, оперативності, регулярності управлінської інформації постають перед великими підприємствами. Вони мають досить велику кількість дочірніх структур, територіально вилучених від головної компанії і працюючих відносно автономно. У цих умовах ІС - єдина можливість збирати, обробляти дані і представляти їх у вигляді, зручному для прийняття управлінських рішень.

Можна умовно виділити наступні *класи ІС*:

Локальні системи (системи для малого бізнесу). Такі системи призначені для ведення обліку по одному або декількох напрямках (бухгалтерія, збут, склади, облік кадрів і т.д.). Системами цієї групи може скористатися практично будь-яке підприємство, якому необхідне керування фінансовими потоками й автоматизація облікових функцій.

За багатьма критеріями вони універсальні, хоча найчастіше розробниками пропонуються рішення галузевих проблем. Універсальність приводить до того, що цикл впровадження таких систем невеликий, іноді можна скористатися "коробковим" варіантом.

Фінансово-управлінські системи. Такі системи дуже гнучко і легко набудовуються на потреби конкретного підприємства, добре інтегрують діяльність підприємства і призначені, у першу чергу, для обліку і керування ресурсами невиробничих компаній. Ці системи, як правило, універсальні, однак може з'являтися необхідність відображення специфіки діяльності конкретної компанії, оскільки їхні функціональні можливості ширше, ніж в локальних. Багато які з цих систем мають базові можливості управління виробництвом.

Середні інтегровані системи. Ці системи призначені для управління виробничим підприємством і інтегрованим плануванням виробничого процесу. Облікові функції, хоча і глибоко пророблені, виконують допоміжну роль і часом неможливо виділити модуль бухгалтерського обліку, тому що інформація в бухгалтерію надходить автоматично з інших модулів.

Такі системи більш складні в установці (цикл упровадження може займати від 6-9 місяців до півтора року і більш). Це обумовлено тим, що система покриває потреби підрозділів і цілком інтегрує виробниче підприємство, що вимагає значних спільних зусиль співробітників підприємства, постачальника програмного забезпечення або консалтингової компанії, що здійснює впровадження.

Великі інтегровані системи. Ці системи відрізняються від середніх інтегрованих систем набором вертикальних ринків і глибиною підтримки процесів управління великих багатофункціональних груп підприємств.

Такі системи відрізняє широта охоплення, включаючи управління виробництвом, управління складними фінансовими потоками, корпоративну консолідацію, глобальне планування, бюджетування й ін. Подібні функції властиві і багатьом фінансово-управлінським (за винятком виробництва), і середнім інтегрованим системам, однак з більш низьким ступенем пророблення. Термін упровадження великих інтегрованих систем звичайно більше року.

Спеціалізовані системи. Такі системи призначені в основному для одержання корпоративної консолідованої звітності, для корпоративного планування і бюджетування, корпоративної консолідації, одержання управлінської звітності, аналізу даних за технологією OLAP.

Безумовно, ніякі інформаційні технології не здатні вирішити весь комплекс різноманітних управлінських проблем. Проте, є багато областей, у яких вони є незамінним помічником.

Класифікація ринку інформаційних систем зведена у таблицю 1.3.

Таблиця 1.3. Класифікація ринку інформаційних систем

Локальні системи	Малі інтегровані системи	Середні інтегровані системи	Великі інтегровані системи
БЭСТ, Инотек, Инфософт, Супер-Менеджер, Турбо-Бухгалтер, Инфо-Бухгалтер	Concorde XAL, Exact, NS-2000, Platinum PRO/MIS, Scala SunSystems, БЭСТ-ПРО, 1С-Предприятие, БОСС-Корпорация, Галактика, Парус, Ресурс, Эталон	Microsoft-Business Solutions - Navision, Axapta, J D Edwards (Robertson & Blums), MFG-Pro (QAD/BMS), SyteLine (COKAP/SYMIX)	SAP/R3 (SAP AG), Baan (Baan), BPCS (ITS/SSA), OEBS (Oracle E-Business Suite)

1.2 Аналіз існуючих методів та технологій проектування інформаційних систем аграрних підприємств

1.2.1 Склад компонентів технології проектування інформаційних систем аграрних підприємств

Проектування інформаційної системи визначається як процес перетворення вхідної інформації про задачі і функції управління, про методи проектування в проект інформаційної системи.

Наявність різноманітних та не взаємозв'язаних задач і функцій управління на аграрному підприємстві, робить процес проектування інформаційної системи досить трудомістким.

Проектування інформаційних систем здійснюється за допомогою конкретної технології проектування, яка визначає, що, як і в якій послідовності, повинно бути виконано для створення проекту.

Технологія проектування складається з трьох основних компонентів:

- *методологія проектування*, яка містить концепції, принципи проектування, які реалізуються методами проектування;
- *засоби проектування*, що забезпечують реалізацію методів проектування;
- *модель життєвого циклу проектування*, що визначає роботи та послідовність їх виконання при розробці інформаційної системи.

Отже, вибір технології проектування передбачає вибір методології, засобів проектування, моделі життєвого циклу. Критерії вибору залежать від особливостей проекту інформаційної системи.

Проекти інформаційних систем аграрних підприємств можуть бути таких типів:

- проект інформаційної системи на базі готових програмних технічних компонентів;
- проект розробки програмних компонентів інформаційної системи;
- проект реконструкції програмних компонентів, що є на підприємстві.

Здійснення одного або усіх вищезазначених проектів залежить від вимог до інформаційної системи конкретного підприємства і визначається на етапі аналізу. В разі наявності інформаційних систем, що здатні автоматизувати усі необхідні функції управління, проектування системи виконується з готових елементів. Відсутність програмних компонентів або їх невідповідність задачам управління призводить до реалізації другого й третього пунктів відповідно.

У таблиці 1.4 приведені характеристики технологій проектування на основі методів, які використовуються в них.

Таблиця 1.4. Характеристики технологій проектування

Клас технології проектування	Методи проектування		
	Залежно від порядку виконання етапів	По мірі використання типових рішення	По мірі автоматизації
Канонічна	Низхідний і висхідний методи	Оригінальне	Ручне
Індустріальна автоматизована	Низхідний і паралельний метод	Оригінальне	Комп'ютерне
Індустріальна типова	Низхідний і паралельний метод	Типове	Комп'ютерне

Для проектування інформаційних систем на аграрних підприємствах може використовуватися як одна технологія проектування, так і декілька.

Для проектування інформаційної системи на базі готових програмних і технічних компонентів доцільно використовувати індустріальну типову технологію. Для розробки програмних компонентів інформаційної системи – канонічну або індустріальну автоматизовану; для реконструкції програмних компонентів, що існують на підприємстві, – поєднання канонічної, індустріальної автоматизованої і типової.

1.2.2 Методи проектування інформаційних систем

Методологія побудови інформаційних систем складається з трьох компонентів:

1. Набір моделей (типів моделей) для опису вимог до інформаційної системи, проектних та програмних рішень. Кожна модель звичайно містить як визначення конструкцій (нотацію), так і правила їх використання (синтаксис).

2. Метод застосування набору моделей для побудови інформаційної системи. Метод звичайно використовує фіксований набір моделей і визначає послідовність їх побудови для опису різних аспектів системи, що створюється.

3. Процес організації проектних робіт, який включає різноманітні технології – планування, управління проектом тощо.

Кожен проект можна розглядати як реалізацію конкретного процесу застосування метода. В залежності від обмежень по строкам і вартості, в конкретну реалізацію можуть бути включені лише окремі частини повного методу і процесу. Крім послідовності етапів методи проектування відрізняються об'єктами дослідження і синтезу.

Класифікація методів проектування показана на рисунку 1.2.

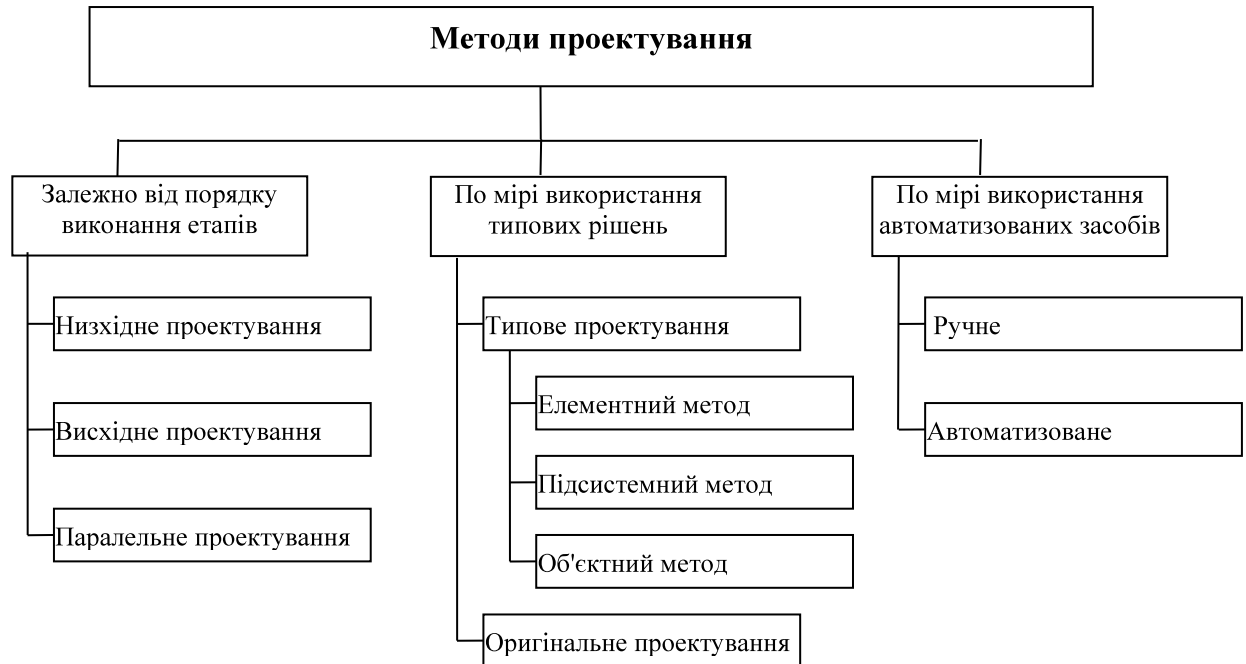


Рисунок 1.2 - Класифікація методів проектування

По мірі використання автоматизованих засобів методи проектування розділяються на методи ручного і автоматизованого проектування.

Методи ручного проектування застосовуються для проектування компонентів інформаційної системи без використання спеціальних інструментальних програмних засобів, а програмування здійснюється на алгоритмічних мовах.

Методи автоматизованого проектування передбачають використання спеціалізованих програмних засобів.

Очевидно, що при ручному проектуванні можливі високі тимчасові витрати й недостатня точність проектних рішень. Однак дані методи завжди використовуються разом при проектуванні, тому що кожен етап проектування

має свою власну задачу, яку не завжди можна вирішити за допомогою методу автоматизованого проектування. Наприклад, визначення вимог до інформаційної системи, що створюється, оформлення техніко-економічного обґрунтування неможливо здійснити за допомогою автоматизованих засобів. В той же час метод ручного проектування для побудови логічної і фізичної структури інформаційної системи істотно ускладнює задачу проектувальника та негативно впливає на подальші етапи проектування, які реалізують свої задачі на основі цих даних.

Залежно від порядку виконання етапів проектування розрізняють методи низхідного (зверху-вниз), висхідного (знизу-вгору) і паралельного проектування.

При *висхідному проектуванні* задачі нижчих ієрархічних рівнів вирішуються раніше, ніж задачі вищих ієрархічних рівнів. Даний метод доцільно застосовувати при проектуванні інформаційних систем з невеликою кількістю функцій. При проектуванні інформаційних систем на підприємствах із складною ієрархією, метод висхідного проектування може привести до такого результату, як "клаптева автоматизація". При такому підході основна увага спрямована на автоматизацію функцій найнижчого рівня організації, які, по-перше, згодом достатньо проблематично буде пов'язати з функціями більш високого рівня, а по-друге буде потрібно постійне доопрацювання деяких компонентів інформаційної системи. Елементи інформаційної системи визначені, проте відомості про властивості і структуру інформаційної системи носять гаданий характер.

Застосування *низхідного методу проектування* інформаційних систем дозволяє раніше вирішувати задачі, які пов'язані з вищими ієрархічними рівнями. Рівні управління на підприємстві взаємозв'язані, оброблена інформація на оперативному рівні поступає на тактичний, а потім і на стратегічний рівні управління. Також складність задач на стратегічному рівні значно більше, ніж на нижчих, тактичному і оперативному. Низхідний метод проектування надає можливість створювати інформаційну систему, враховуючи стратегію організації. Це в свою чергу дозволяє зв'язати задачі нижчого рівня з головною

задачею підприємства. Сучасні виробники інформаційних систем при проектуванні використовують метод низхідного проектування.

Проте, як в низхідному, так і у висхідному проектуванні, присутня невизначеність, яка може привести до значних помилок, збільшити терміни і собівартість проектування. При низхідному проектуванні слабо враховуються потреби нижніх рівнів, а при висхідному – верхніх рівнів. Дану проблему здатний вирішити *метод паралельного проектування*.

Використання цього методу дозволяє проводити паралельно проектування інформаційної системи на трьох рівнях: стратегічному, тактичному і оперативному. Учасники робіт мають можливість обмінюватися інформацією на всіх рівнях, внаслідок чого знижується ймовірність появи якої-небудь ризикової події, забезпечуються вимоги, як стратегічних цілей організації, так і тактичних, і оперативних. Перевагою паралельного проектування є те, що на конкретному етапі проектування враховуються критерії з інших етапів.

По мірі використання типових проектних рішень розрізняють наступні методи проектування: оригінального і типового проектування.

Методи оригінального проектування використовуються тоді, коли проектні рішення розробляються "з нуля" відповідно до вимог до інформаційної системи.

Оригінальне проектування інформаційної системи на аграрному підприємстві здійснюється у разі, коли на ринку програмних продуктів відсутні засоби, які здатні забезпечити потреби організацій, або існуючі програмні рішення на підприємстві не мають в своєму складі компонента, який здатний забезпечити інформаційну підтримку необхідної функції управління.

Методи типового проектування застосовуються для створення інформаційної системи на основі готових типових проектних рішень (програмних модулів).

Дані методи застосовуються, коли підприємство ставить перед собою наступні задачі:

- створити інтегровану інформаційну систему з існуючих програмних комплексів;
- створити інформаційну систему з існуючих проектних рішень, що спрямована на реалізацію конкретної функції на певному рівні управління.

Методи типового і оригінального проектування часто перетинаються між собою, якщо підприємство створює інтегровану інформаційну систему (типове проектування) і паралельно доопрацьовує деякі модулі програмних продуктів (оригінальне проектування) відповідно до своїх задач.

Проектування інформаційної системи з використанням методів типового проектування декомпонується на безліч складових елементів. На кожному рівні декомпозиції застосовується певний метод типового проектування: елементний, підсистемний і об'єктний.

При елементному методі як типовий елемент системи використовується типове рішення по завданню. Застосування цього методу доцільне тоді, коли необхідно створити інформаційну систему, яка вирішує невелику кількість задач, або створити один модуль. У інших випадках можливі великі затрати часу на з'єднання різнорідних елементів системи.

Типовим елементом в *підсистемному методі* є окремі підсистеми, розроблені з урахуванням функціональної повноти і мінімізації зовнішніх зв'язків. Даний метод використовується у разі потреби проектування інформаційної системи, що забезпечує окрему функцію управління. Наприклад, фінансове управління, матеріально-технічне постачання, бухгалтерський облік. Проблема при використанні даного методу може виникнути в області інтеграції інформаційної системи, що розробляється, із програмними рішеннями, що існують на підприємстві. До готових типових рішень можна віднести наступні системи: "1С предприятие", Project Planner, Oracle Financial Services Applications.

При об'єктному методі у якості типового елемента використовується повний набір функціональних підсистем для об'єктів управління певної галузі. Даний метод слід застосовувати при необхідності автоматизації комплексу взаємозв'язаних функцій управління. Наприклад, для комплексної автоматизації служб підприємства по управлінню виробництвом, ресурсами, фінансами.

Однак при використанні даного методу також можуть виникнути труднощі, тому що необхідно враховувати специфіку предметної області при проектуванні. Застосування готових типових рішень (Парус, Галактика, ІТ-предприятие) у більшості випадків потребує зміни структури підприємства, послідовності виробничих процесів, функціональних обов'язків співробітників.

Ознаки вибраних методів дозволяють визначити характер необхідної технології проектування для конкретної інформаційної системи. Розрізняють канонічну та індустріальну технології проектування.

Канонічне проектування ґрунтується на застосуванні виконавцями ручних методів проектування, без використання інструментальних засобів. Дана технологія реалізується в проектуванні невеликих локальних інформаційних систем.

Індустріальна технологія проектування ґрунтується на використанні вже готових засобів розробки інформаційних систем.

Таблиця 1.5 містить порівняння існуючих методів, визначення недоліків та переваг кожного з них.

Таблиця 1.5. Порівняння існуючих типових методів проектування інформаційних систем

Методи		Елемент системи	Приклади ІС	Переваги	Недоліки
Типове використання рішень	Елементарний	Типове рішення по завданню		Дозволяє якісно та швидко створити один модуль, що вирішує нескладну задачу	При складній ІС треба багато часу для з'єднання різноманітних елементів системи
	Підсистемний	Окремі підсистеми, розроблені з урахуванням повноти і мінімізації зовнішніх зв'язків	"ІС підприємства", Project Planner, Oracle Financial Services Applications	Дозволяє спроектувати окрему функцію управління	Проблема інтеграції з програмними рішеннями, що вже існують на підприємстві

					тві
	Об'єктний	Повний набір функціональних підсистем для об'єктів управління галузі	Парус, Галактика, ІТ-предприємство	Використання для автоматизації комплексу взаємозв'язаних функцій управління	Необхідно враховувати специфіку предметної області, потребує зміни структури підприємства

Існують і інші підходи до визначення методів проектування інформаційних систем. Наведемо ці підходи та класифікацію методів.

Підхід, запропонований Клещевим і Романовим, в якому методи умовно згруповані як: структурно-функціональні, віртуальні (універсальні), функціонально-технологічні, об'єктні. Очевидно, що таким чином вилічені групи сильно взаємопов'язані і суттєво перетинаються по змісту. Однак структурно-функціональний аналіз є методологічною основою практично у всіх методах проектування інформаційних систем.

Віртуальні (універсальні) методи дають змогу описати абстрактний набір структур розрахункових процесів і каналів взаємодії, які відображають оригінальну структуру об'єкта, що автоматизується. Опис предметів на віртуальному рівні визначає внутрішню структуру систем, але не враховує програмних і інформаційних взаємозв'язків між процесами, задачами, що їх реалізують і об'єктами автоматизації.

Віртуальний рівень проектування отримав широке розповсюдження для інформаційних систем, орієнтованих на рішення простих задач або на операції обробки типових даних. Для інформаційних систем, що реалізують постійно змінюючи функції організаційного управління, як сама сукупність задач, так і, особливо, їх параметри у великій мірі характеризуються невизначеністю не лише на початкових стадіях проектування, але і в процесі експлуатації. В період усього періоду створення і розвитку таких інформаційних систем, як

організаційна структура самого об'єкта автоматизації, так і функціонально-технологічні задачі, що вирішується, знаходяться в процесі розвитку.

Структурно-функціональні методи проектування і розробки полягають у декомпозиції структури інформаційної системи на окремі підсистеми і модулі в цілях аналізу їх технічного, системного і прикладного складу і наступного синтезу структури системи для реалізації заданих функцій, що потребуються.

Ці методи широко використовують вже відпрацьовані алгоритми і промодельовані задачі і гарно себе зарекомендували при проектуванні універсальних і спеціалізованих систем, окремих машин і пристроїв. Якщо зміни і нові функції незначно відображаються на структурі цих пристроїв, машин і систем, то їх адаптація до нових умов функціонування не викликає надто великих фінансових втрат.

Функціонально-технологічні методи проектування інформаційних систем забезпечують аналіз неперервно змінного спектра організаційних, управлінських функцій і технологій.

Існуючою перевагою функціонально-технологічних методів є формування і застосування динамічних типових конструкторських рішень при створенні ІС: типових прикладних рішень, типових технологічних підсистем і типових операцій по обробці інформації, а також пов'язаних з ними програмно-технологічних рішень.

Застосування типових конструкторських рішень забезпечує високу динаміку проробки технічних рішень для всіх видів процесів і взаємозв'язок їх по трьом рівням проектування: концептуальному, на якому здійснюється аналіз і синтез системних архітектур; рівні функціональних об'єктів, де формуються рішення по створенню і вдосконаленню інформаційної системи; рівні елементів ІС – створення програмно-технічної інфраструктури об'єктів, що функціонують.

Об'єктні методи забезпечують єдину концепцію для передпроектного аналізу, проектування, програмування функціональних задач.

Концептуальне єдність об'єктних методів забезпечується за рахунок використання базових елементів: об'єктів, класів і властивостей наслідування.

Абстрактний об'єкт складається з трьох частин: імені об'єкта, стану (змінних стану) і методів (операцій).

Об'єктний аналіз представляє собою метод аналізу предметної області, заснований на переважному виявленні об'єктів цієї предметної області і встановленні взаємних зв'язків між ними.

Об'єктне проектування представляє собою метод проектування, що поєднує в собі процес об'єктної декомпозиції, прийоми представлення моделей проекрованої інформаційної системи як на логічному та фізичному рівнях, так і в статичній і динамічній формах. В цілому концепція об'єктних методів спирається на декомпозицію об'єкта, декларований опис об'єктів класами, а також покрокове програмування з використанням механізму наслідування.

Типові технологічні підсистеми, створені за допомогою об'єктних методів, є розподіленими системами розрахунків, так як пам'ять розподілена між об'єктами і кожна операція виконується лише в пам'яті даного об'єкта.

Перевагами об'єктних методів є їх орієнтованість на задачі аналізу, проектування і програмування інформаційних систем з урахування динаміки інформаційних вимог користувачів, реалізація яких традиційними методами вкрай трудомістка.

На рисунку 1.3 запропоновано класифікацію методів за цим підходом і визначені сфери застосування.



Рисунок 1.3 - Класифікація методів за підходом, запропонованим Клещевим і Романовим

Підхід, запропонований Прозоровим, класифікація методів в якому здійснюється на підставі поняття життєвого циклу інформаційної системи, підходів до суттєвої та об'єктної декомпозиції і локалізації інформаційної системи. Від представлений наступними методами: метод OMT (Object Modeling Technique), методи сімейства (Icam DEFinition) і метод COMET (Concurrent Object Modeling and Architectural Design Method).

У *методі OMT* інформаційна система, що проєктується представляється у вигляді трьох взаємопов'язаних моделей:

- об'єктної моделі, яка визначає статичні аспекти системи, в основному пов'язані з даними (статична структура);

- динамічної моделі, яка описує роботу окремих частин системи (динамічна структура);
- функціональної моделі, яка описує взаємодію окремих частин системи, яке виникає у процесі її роботи (взаємодія описується як у ростині даних, так і у ростині управління).

Ці три моделі дозволяють отримати три ортогональні представлення системи в одній нотації. Сукупність моделей системи може бути проінтерпретована на комп'ютері, що дозволяє продемонструвати характер роботи з майбутньою системою і суттєво спрощує узгодження попереднього проекту системи.

Цей метод обмежується двома фазами життєвого циклу ІС: аналізом вимог спільно з побудовою об'єктної моделі і реалізацією програмної системи.

Сімейство методів проектування нотацій IDEF базується на методології ICAM (Intergated Computer Aided Manufacturing) і складається з наступних представників:

- IDEF0 – метод і нотація опису бізнес-процесів;
- IDEF1 – метод і нотація опису взаємозв'язків між інформаційними потоками;
- IDEF1X – метод і нотації розробки реляційних баз даних;
- IDEF3 – метод і нотація опису технологічних процесів;
- IDEF5 – метод і нотація опису онтологічних досліджень.

Метод COMET заснований на методиці COMADM (Concurrent Object Modeling and Architectural Design Method), маючи з ним незначні відмінності, обумовлені використанням у методі COMET нотації UML. Вказані два методи розроблені професором кафедри програмо техніки університету Джорджа Мейсона Хасаном Гома, авторитетним ідеологом у області проектування розподілених додатків і додатків реального часу.

У загальному вигляді метод COMET представляє собою метод об'єктного моделювання і архітектурного проектування паралельних систем, в основі якого лежить створення об'єктно-орієнтованого програмного

забезпечення. Життєвий цикл даного методу характеризується значним числом ітерацій.

Основними етапами метода COMET є:

- Етап моделювання функціональних потреб – основна увага приділяється збору і класифікації функцій до системи, в той час, як сама система розглядається як "чорна скринька"
- Етап аналітичного моделювання виконується в термінах сутностей моделі. Основна увага на даному етапі приділяється предметній області, при цьому структуру сутностей моделі описується за допомогою статичного представлення моделі, а характер поведінки – за допомогою динамічного представлення моделі. Статистичне представлення моделі виконується у термінах класів (об'єктів) предметної області відношень між ними, в той час як динамічне представлення моделі виконується в термінах взаємодії між об'єктами;
- Етап архітектурного (імітаційного) моделювання виконується у термінах структури імітаційної моделі (класи і відношення між ними. На даному етапі основна увага приділяється об'єктній і часовій декомпозиції сутностей моделей, формуються базові критерії розбиття системи на основні частини (підсистеми, модулі та ін.);
- Етап програмного моделювання виконується в термінах програмної моделі (атрибути і операції класів). На даному етапі особлива увага приділяється програмній реалізації імітаційної моделі.

На рисунку 1.4 приведена умовна схема класифікації методів, яка була запропонована Прозоровим.

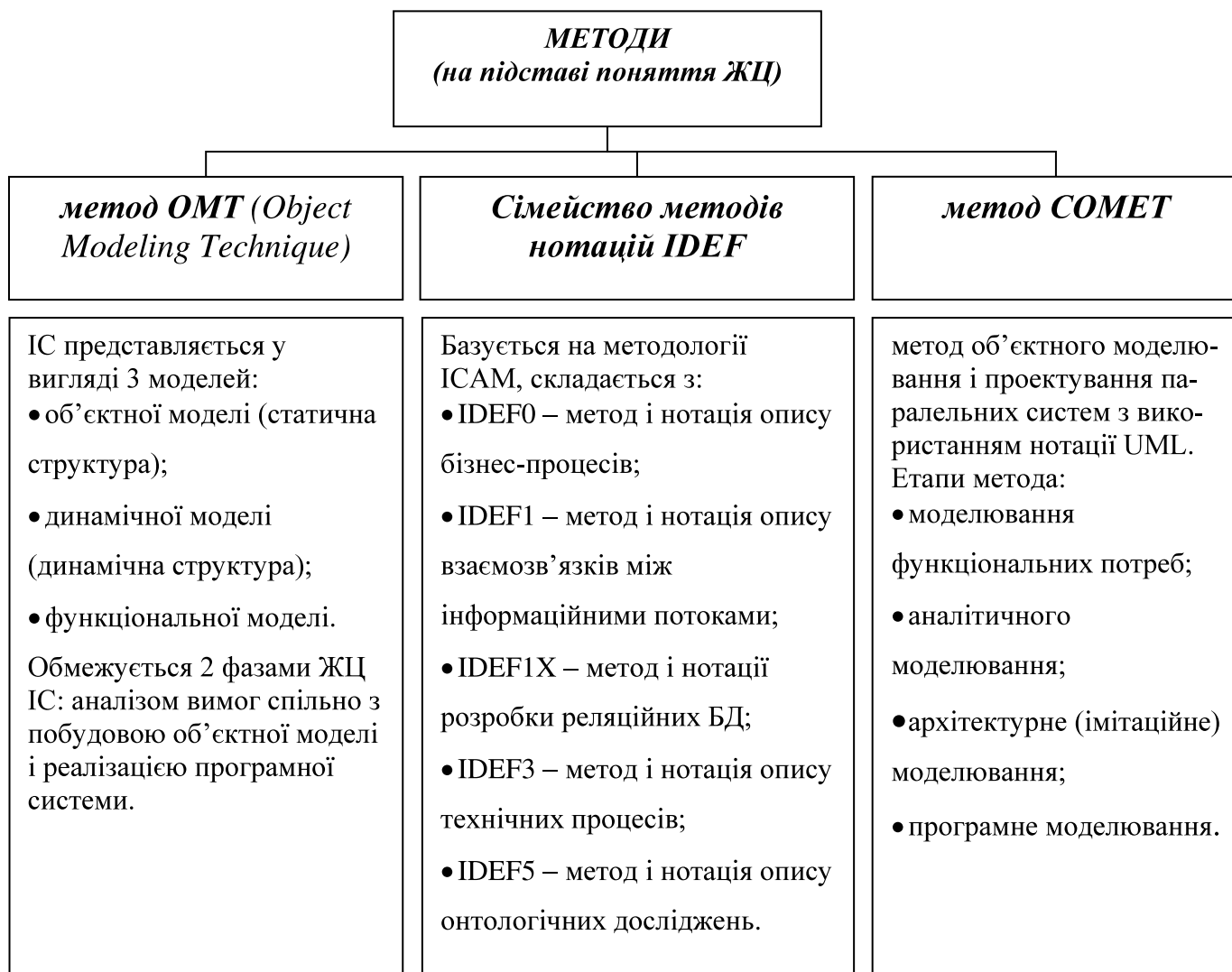


Рисунок 1.4 - Класифікація методів за підходом, запропонованим Прозоровим

1.2.3 Досвід впровадження інформаційних систем та визначення результатів впровадження

В даний час практично будь-який керівник компанії розуміє, що успішне ведення бізнесу без підтримки сучасної корпоративної інформаційної системи (КІС) дуже важке. Проте впровадження корпоративної інформаційної системи на підприємстві, як і будь-яке серйозне перетворення, є складним і часто болючим процесом. Тим не менш, деякі проблеми, що виникають при впровадженні КІС, досить добре вивчені, формалізовані і мають ефективні методології рішення. Завчасне вивчення цих проблем і підготовка до них значно

полегшують процес впровадження і підвищують ефективність подальшого використання КІС.

Основні проблеми та завдання, що вимагають особливої уваги при їх вирішенні:

- залучення топ-менеджменту;
- відсутність постановки задачі менеджменту на підприємстві;
- необхідність у частковій або повній реорганізації структури підприємства;
- необхідність зміни технології бізнесу в різних аспектах;
- розмежування завдань бухгалтерського та управлінського обліку;
- опір співробітників підприємства;
- тимчасове збільшення навантаження на співробітників під час впровадження КІС;
- необхідність у формуванні кваліфікованої групи впровадження та супроводження КІС, вибір сильного керівника групи.

Також існують наступні технології впровадження інформаційних систем на підприємствах:

- експрес-впровадження;
- типове впровадження;
- проектне впровадження.

Експрес-впровадження. Галузеве впровадження з фіксованим терміном, обсягом і вартістю робіт. Застосовується для малого бізнесу або як пілотний етап комплексного проекту автоматизації.

Основні етапи впровадження галузевих програмних продуктів:

- Проведення експрес-обстеження.
- Вибір та постачання програмних продуктів.
- Налаштування та конфігурування системи під вимоги Замовника (без зміни функціоналу).
- Навчання користувачів роботі з програмним продуктом.

- Впровадження системи: інсталяція програмного забезпечення та налаштування робочих місць; консультації співробітників Замовника з введення початкових залишків та заповнення нормативно-довідкової інформації.

- Дослідна експлуатація системи: консультації співробітників Замовника.

- Далі за бажанням Замовника здійснюється розвиток системи:

- Формалізація вимог з доопрацювання системи під індивідуальні потреби Замовника. Узгодження Технічного завдання (ТЗ).

- Реалізація доробок системи відповідно до затвердженого ТЗ.

- Консультації співробітників Замовника по проведеним змінам в системі.

- Супровід системи в період аграрної експлуатації.

Перевага даної технології полягає в тому, що Ви отримуєте швидкий результат в короткі терміни. При цьому існує можливість доопрацьовувати систему під специфічні потреби та бізнес-процеси Вашої організації паралельно з впровадженням і в період подальшої експлуатації системи.

Типове впровадження. Впровадження типових програмних продуктів без зміни функціоналу, з мінімальною вартістю і терміном виконання робіт. Застосовується для малого та середнього бізнесу з типовими бізнес-процесами.

Основні етапи типового впровадження:

- Поставка та інсталяція типового програмного забезпечення на робочих місцях.

- Мінімальна настройка програмного продукту (без зміни функціоналу).

- Навчання користувачів роботі з програмним продуктом.

- Запуск програмного продукту в експлуатацію: консультації співробітників Замовника з введення початкових залишків та заповнення нормативно-довідкової інформації.

- Стандартне інформаційно-технологічний супровід (ІТС).

Перевага даної технології полягає в тому, що Ви отримуєте готовий до експлуатації програмний продукт, фактично оплачуючи тільки вартість ліцензій. Умовою вибору даної технології є відповідність бізнес-процесів Вашої організації стандартному функціоналу програмного продукту або можливість зміни бізнес-процесів організації під стандарти, закладені в типовому програмному продукті.

Проектне впровадження. Впровадження зі значною зміною типового функціоналу. Максимальна деталізація і послідовне поетапне впровадження. Застосовується для середнього та великого бізнесу.

Проектне впровадження засноване на класичному підході і міжнародних стандартах управління проектами PMBook.

Основні етапи проектного впровадження:

- **Предпроектне обстеження підприємства:** в результаті обстеження складається схема бізнес-процесів "як є" і здійснюється вибір програмного забезпечення, найбільш відповідного до існуючих бізнес-процесів Замовника.
 - Постачання та установка обраного програмного забезпечення.
 - Формалізація вимог по доопрацюванням програмного забезпечення під потреби Замовника. Узгодження Технічного завдання (ТЗ).
 - Реалізація доробок програмного забезпечення згідно ТЗ.
 - Попереднє тестування працездатності програмного продукту.
 - Підготовка користувальницької та технічної документації для роботи з програмним продуктом.
 - Навчання користувачів роботі з програмним продуктом.
 - Попереднє Впровадження: консультації з перенесення початкових залишків та заповнення нормативно-довідковою інформацією; розгортання та налаштування робочих місць.
 - Дослідно-аграрна експлуатація: оперативна підтримка та консультації користувачів, можливі доопрацювання програмного продукту, тестування, виправлення помилок.
 - Супровід системи в період аграрної експлуатації.

Перевага даної технології полягає в тому, що впровадження здійснюється відповідно до міжнародних стандартів, що дозволяє Вам на будь-якому етапі проекту контролювати виконання проекту, бачити проміжні результати і відслідковувати строки та бюджет проекту. Передумовами застосування даної технології є масштаби впровадження, спрямовані на автоматизацію різних функціональних підрозділів організації, наявність регіональної мережі і територіальна віддаленість об'єктів, невизначеність методів досягнення цілей і завдань на стадії ініціації проекту, створення унікального продукту з можливістю тиражування на всіх об'єктах Замовника.

Це найбільш витратний і тривалий спосіб ведення проектів, тому його недоцільно застосовувати на підприємствах малого бізнесу, а також при автоматизації окремих функціональних підрозділів.

На рисунку 1.5 схематично показані технології впровадження інформаційних систем, визначені сфери застосування і основні переваги та недоліки.

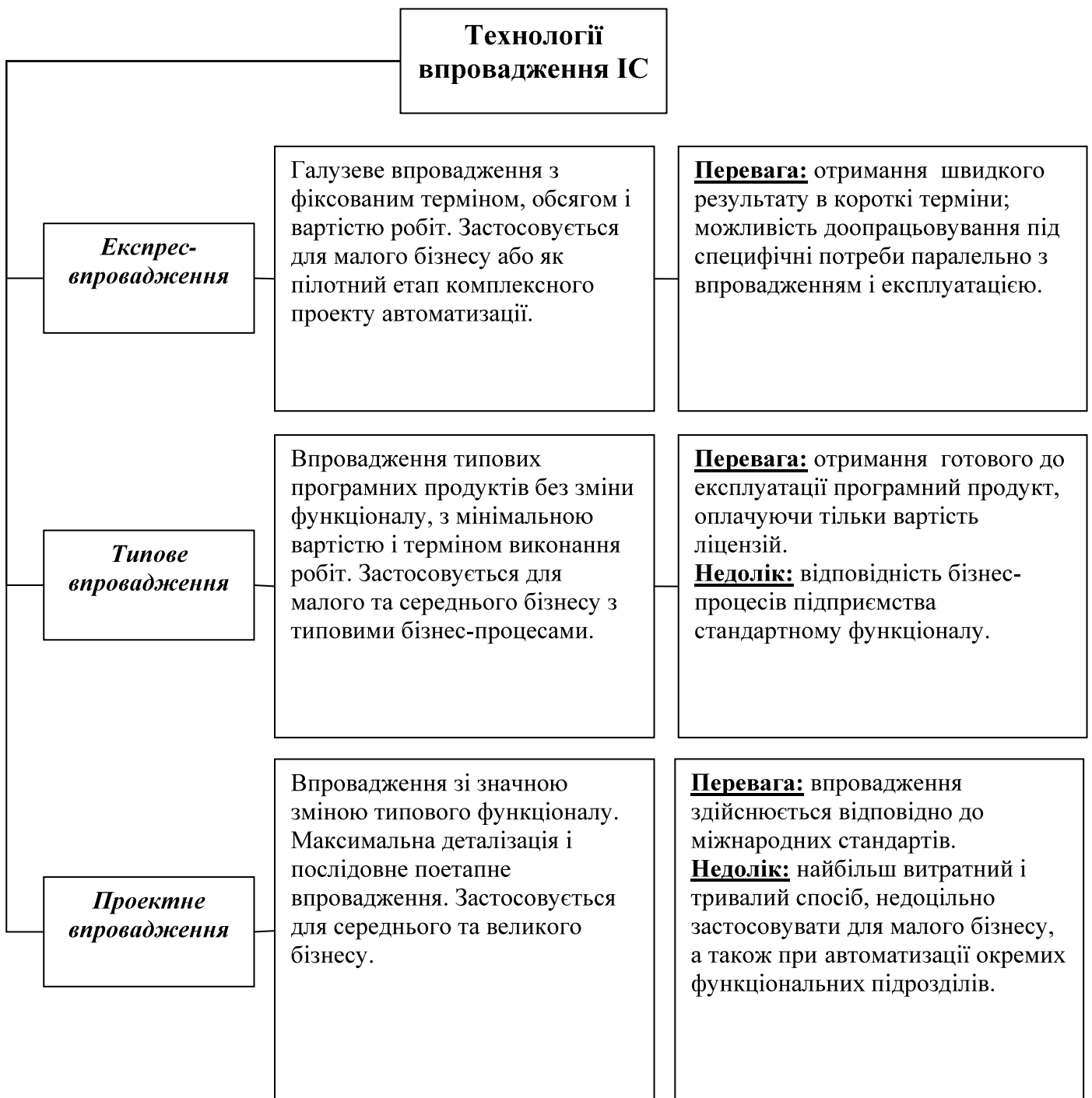


Рисунок 1.5 - Технології впровадження інформаційних систем

Перш за все слід відзначити той факт, що впровадження ІС призводить зазвичай до серйозних наслідків для всієї структури управління компанією. Тому оцінювати ефективність впровадження системи без оцінки ефективності зміни роботи компанії неможливо. Оцінка тих чи інших змін при впровадженні системи принципово залежить від загальних результатів змін (реінжинірингу), супутнього впровадженню системи.

Більше 65% впроваджень інформаційних систем виявляються провальними. Дуже часто це трапляється через те, що спочатку не приділяється увага передпроектному етапу роботи - не проводиться комплексне обстеження організації та аналіз інформаційного простору.

П'ять причин невдалого впровадження ERP систем

Невдалі приклади впровадження ERP систем, будь то збільшення бюджету, система, яка не підтримує потреби бізнесу або відсутність підтримки співробітників, тягнуть за собою великі проблеми в бізнесі підприємства. Так чому ж більше 65% впроваджень ERP систем виявляються провальними?

Досвід консалтингової фірми "Панорама", що надає допомогу щодо виправлення наслідків подібних невдалих впроваджень, дозволяє їй узагальнити та проаналізувати причини провалів. Нижче наведено п'ять основних причин, які зустрічалися їм найчастіше.

1. Невідповідний софт. На ринку є сотні програм, дуже важливо вибрати ту, яка найбільш повно підходить до потреб вашого підприємства.
2. Завищені очікування від впровадження. Компанії, що планують впровадження, часто не закладають в проект достатню кількість часу, грошей і ресурсів для забезпечення успіху проекту. Крім того, що провалилися проекти часто не включають такі ключові моменти, як менеджмент змін, визначення бізнес процесів і документообігу та пілотні впровадження.
3. Недолік підтримки та управління з боку керівництва компанії. Управління проектом з боку вищого керівництва компанії включає в себе не тільки оплату рахунків і передачу проекту команді виконавців. Воно також включає визначення завдань впровадження, створення керуючого органу проекту, а також участь у всіх рішеннях при необхідності. Цей пункт - найважливіший, бо він зачіпає і інші чотири.
4. Перевага кастомізувати софт, відмова від використання стандартних функцій. Практично кожне невдале впровадження містить спробу кастомізувати софт. Це відбувається через невдалого вибору софта спочатку.
5. Брак досвіду впровадження. У невдалих випадках впроваджень часто фігурує недосвідчена команда інтеграторів, яка не знає, чого вона не знає.

На рисунку 1.6 показана діаграма, в якій зазначено основні причини невдалого впровадження інформаційних систем (за результатами інтернет-опитування "Інформаційні системи управління проектами – вибір російських компаній").

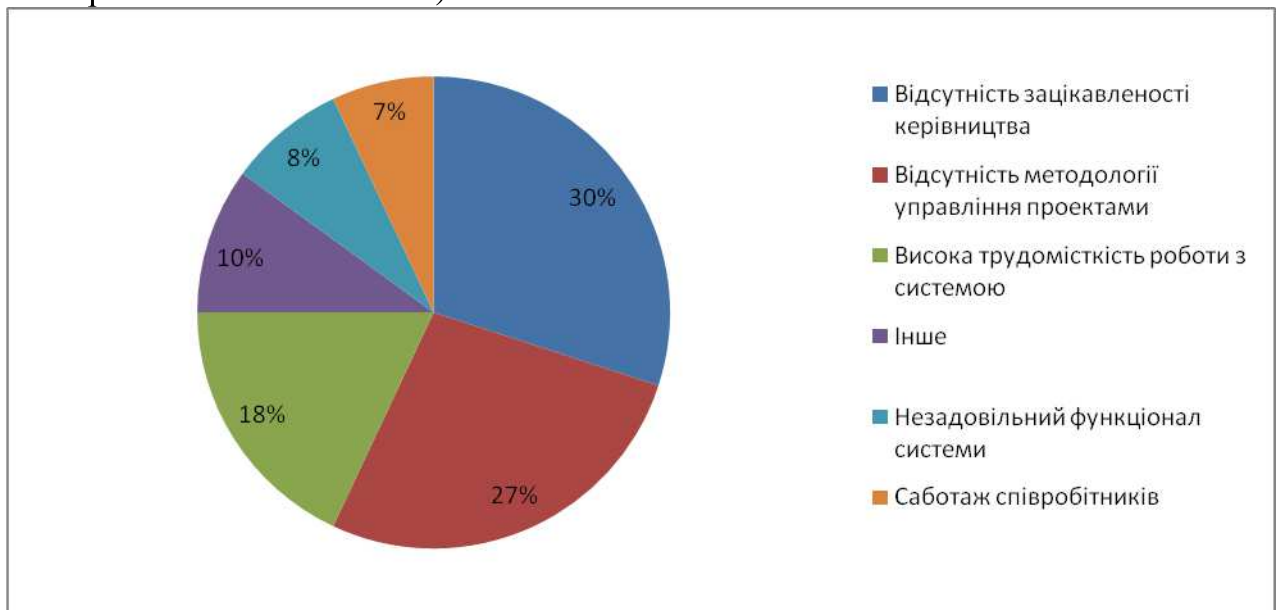


Рисунок 1.6 - Причини невдалого впровадження інформаційних систем

Параметри успіху процесу впровадження

Найважливішим параметром успіху впровадження ІС є відпрацьована методологія впровадження, що дозволяє жорстко контролювати і документувати цей процес.

Крім цього, можна навести такі чинники, що гарантують успіх процесу впровадження:

- Наявність досвіду успішних комплексних впроваджень на аграрних підприємствах.
- База перевірених на практиці рішень.
- Кваліфікація впроваджувальних колективів. Наявність корпоративної мережі впроваджувальних центрів і центрів підтримки. Внутрішні стандарти на впровадження та супровід.
- Оперативна підтримка розробником.
- База організаційних документів щодо впровадження (статут проекту, плани організаційно-технічних заходів, плани навчання, проекти наказів, ...).

- Наявність повної документації на систему, включаючи посібники, керівництва по налаштуванню, керівництво адміністратора, керівництво програміста, повний опис структур таблиць інформаційної бази, опис.

Тривалість проекту впровадження системи

Як показує досвід впровадження автоматизованих систем на підприємствах, найбільш реальними термінами впровадження комплексної системи є 0,5-2 року (окремі модулі можна впровадити за 1-3 місяці).

На рисунку 1.7 показана діаграма, на якій зазначено основні чинники, що можуть забезпечити ефективне впровадження інформаційної системи.

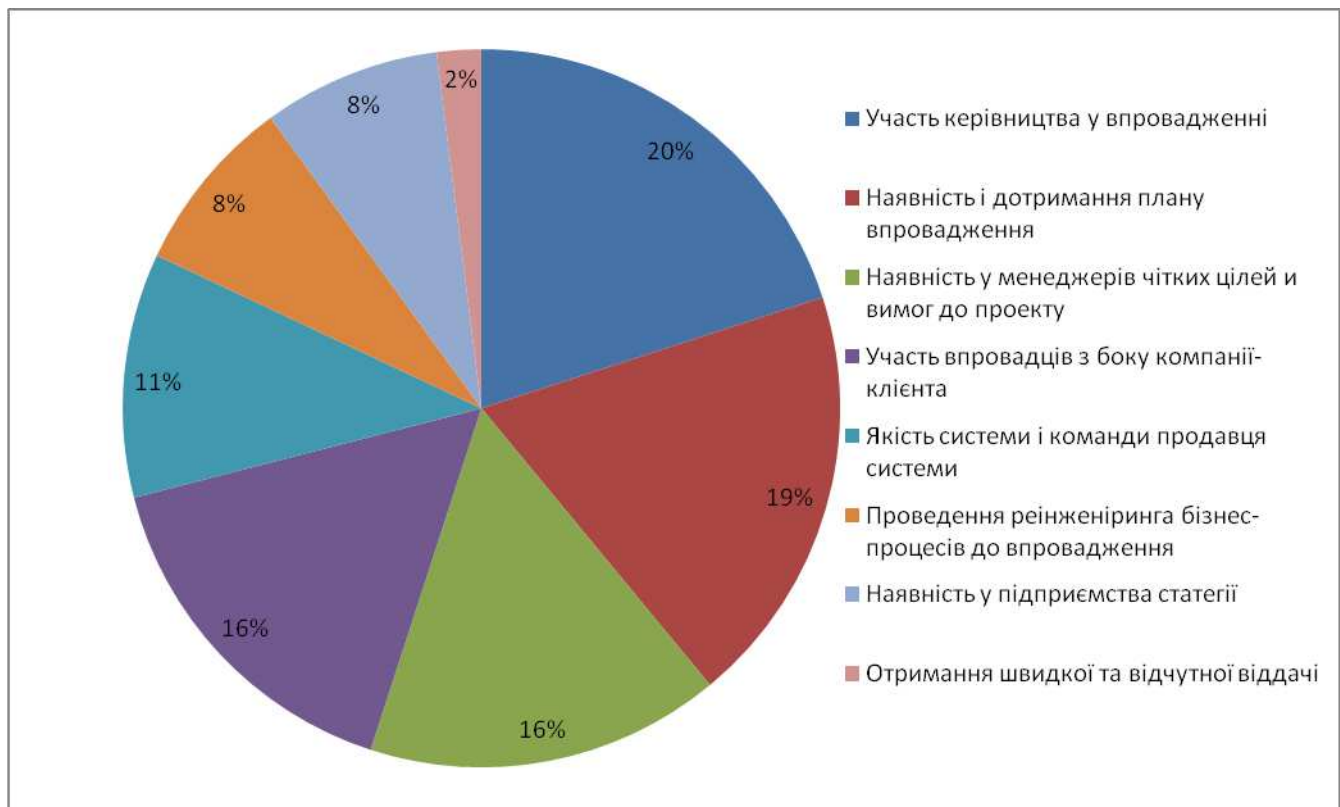


Рисунок 1.7 – Фактори успіху впровадження інформаційних систем

1.3 Особливості проектування інформаційних систем в умовах обмежених ресурсів

У сучасних економічних умовах основним показником роботи підприємств є рівень конкуренції. Конкурентне середовище виробництва – це сукупність умов і факторів, які впливають на функціонування підприємства і вимагають прийняття управлінських рішень, які були б направлені на забезпечення конкурентоспроможності продукції і підприємства. Рівень

конкурентоспроможності підприємства визначають такі фактори, як рентабельність виробництва, характер інноваційної діяльності, рівень виробничої праці, ефективність управління підприємства та ін. Саме для вдосконалення рівня ефективності управління та збереження своїх позицій на ринку підприємства є сенс розвивати та вдосконалювати сучасні інформаційні технології.

ІС підприємства – сукупність організаційних, технічних, програмних і інформаційних засобів, що об'єднані в єдину систему для реалізації мети і виконання задач.

Основними цілями ІС аграрних підприємств є збір, обробка, зберігання і представлення даних про діяльність організації і зовнішнє середовище у вигляді, що є зручним для фінансового і будь-якого іншого аналізу і для використання при ухваленні управлінських рішень.

Проектування ІС визначається як процес перетворення вхідної інформації про задачі і функції управління та про методи проектування в проект ІС. Так як проектування ІС є дуже важливим, то є сенс вже на стадії планування і аналізу вимог проаналізувати усі можливі впливи факторів. Також у процесі проектування ІС вже на початковій стадії може виникнути проблема, яка пов'язана з розподілення ресурсів. Все це необхідно робити, щоб у подальшому попередити переробку та, відповідно, матеріальні витрати на це.

ІС аграрного підприємства розробляється як деякий проект. Багато особливостей управління проектами і фази розробки проекту є загальними. Проект – це обмежена за часом цілеспрямована зміна окремої системи з початковими чітко визначеними цілями, досягнення яких позначає завершення проекту, а також з встановленими вимогами по строкам, результатам, ризику, кількості використаних ресурсів, організаційній структурі.

1.3.1 Опис метода Total cost of ownership (TCO), вплив прямих та непрямих витрат на проектування ІС

Total Cost of Ownership (Сукупна вартість власності) (TCO) - це метод, що можливо використовувати для того, щоб прослідкувати, що усі супутні

витрати враховані на певний період часу. ТСО можна представити, як усі витрати, пов'язані з володінням і експлуатацією активу впродовж певного періоду часу. ТСО не лише відбиває витрати купівлі, а й включає усі інші аспекти подальшого використання і обслуговування активу. Сукупна вартість володіння є найбільш ефективним механізмом миттєвої оцінки загального об'єму витрат підприємства на ІТ -інфраструктуру. Метод ТСО перше був розроблений у кінці 1980-х дослідницькою компанією Gartner для того, щоб визначити витрати на комп'ютерну техніку.

Методика ТСО передбачає оцінку витрат на ІТ - інфраструктуру і на окремі її компоненти. Сукупні витрати включають як прямі або "бюджетні", які беруться з бухгалтерських документів, так і "непрямі" - фінансові витрати, які підприємство несе з причини неефективності роботи інформаційної системи і сервісних ІТ - служб.

Наприклад, при аналізі ІТ - інфраструктури в цілому "непрямі" витрати включатимуть:

- самонавчання користувачів роботі з ІС підприємства;
- самообслуговування користувачем свого комп'ютера і встановленого програмного забезпечення, допомога колегам в цьому;
- використання службового комп'ютера і ПО для неслужбової діяльності - ігри, Web - серфінг, чат і ICQ, робота на сторону;
- корупційні схеми при купівлі устаткування, комплектуючих і витратних матеріалів, замовленні послуг;
- простої і збої в роботі ІС в цілому і окремих її частин.

Як видно, нехтувати подібними витратами не можна. Їх треба враховувати при проектуванні ІС, оскільки вони є не випадковими або не стихійними, а постійними і закономірними. І іноді на реалізацію подібних витрат йде досить велика сума грошей.

За своєю суттю, методика ТСО дуже добре лягає на механізми аналізу операційних і технологічних ризиків. Більше того вона є основою або

складовою частиною значного числа методик якісної і кількісної оцінки ефективності ІТ – інфраструктури підприємства.

1.3.2 Аналіз існуючих моделей життєвого циклу інформаційних систем та принцип розподілу ресурсів на кожній з них

Життєвий цикл - період створення і використання інформаційної системи, який охоплює її різноманітні стани, починаючи з моменту виникнення необхідності в даній інформаційній системі і закінчуючи моментом її повного виходу з експлуатації.

Системне проектування (розробка) і використання інформаційної системи має пройти наступний життєвий цикл інформаційної системи:

- передпроектний аналіз (досвід створення інших аналогічних систем, відмінності і особливості та інше), аналіз зовнішніх проявів системи;
- внутрішньосистемний аналіз, внутрішній аналіз (аналіз підсистем системи);
- системне представлення системи (опис системної цілі, системних відношень і зв'язків із зовнішнім середовищем, інших системам і системних ресурсів - матеріальних, енергетичних, інформаційних, організаційних, людських, часових і просторових);
- визначення критеріїв адекватності, ефективності і стійкості (надійності);
- функціональний опис підсистем системи (опис моделей, алгоритмів функціонування підсистем);
- макетний опис системи, оцінка взаємодії підсистем системи, при цьому можливе використання "макетів" критеріїв адекватності, стійкості, ефективності;
- "збір" і тестування системи – реалізація повноцінних функціональних підсистем і критеріїв, оцінка моделі по сформульованим критеріям;
- функціонування системи;

- визначення цілей подальшого розвитку системи і її додатків;
- уточнення, модифікація, розширення можливостей системи в режимі її функціонування (з ціллю її еволюціонування).

Життєвий цикл носить ітеративний характер: реалізовані етапи життєвого циклу, починаючи з самих ранніх, циклічно повторюються відповідно до нових вимог і змін зовнішніх умов. На кожному етапі життєвого циклу формується набір документів і технічних рішень, які є початковими для наступних рішень.

Найбільш відомими моделями життєвого циклу є каскадна модель, ітераційна модель, спіральна модель та модель Microsoft Solutions Framework.

1. Каскадна модель (рисунок 1.8) передбачає розробку інформаційної системи протягом декількох послідовних етапів, причому перехід на наступний етап відбувається лише тоді, коли робота на попередньому етапі була повністю завершена. Дана модель задовольняла потреби проектування невеликих і нескладних інформаційних систем в 70-х роках. Проте через звичку та недолік знань каскадна модель широко використовується і зараз.

Переваги каскадного підходу виявляються при побудові інформаційних систем, для яких на самому початку розробки можна сформулювати всі вимоги. В цьому випадку для кожного етапу можна визначити точні терміни завершення і вартість робіт.

В разі використання каскадної моделі для реалізації складних проектів виникає велика кількість труднощів: неможливість виправити помилки на попередніх етапах; неточності в документації і звітності; значна тривалість проектування, оскільки сформульовані вимоги до проекту можна змінити лише в кінці розробки.

Розподілення ресурсів при каскадному підході відбувається відразу на першому етапі, після чого відбувається перехід на наступний етап проектування. Це може призвести к тому, що виникне проблема неправильного розподілення ресурсів, яка в свою чергу виявиться на наступних етапах, а повернутися назад проектувальник вже не матиме змогу. І в результаті, початкове розподілення ресурсів буде некоректним та хибним, буде потребувати змін або детальнішого

аналізу, а кінцевий результат характеризуватиметься витратами, матеріальними збитками та неефективністю роботи всієї інформаційної системи. Треба також зазначити, що не всі ресурси можна буде врахувати відразу, деякі можуть з'явитися через проміжок часу, а при каскадному підході стає неможливим їх включення в систему та подальший аналіз і розподіл.

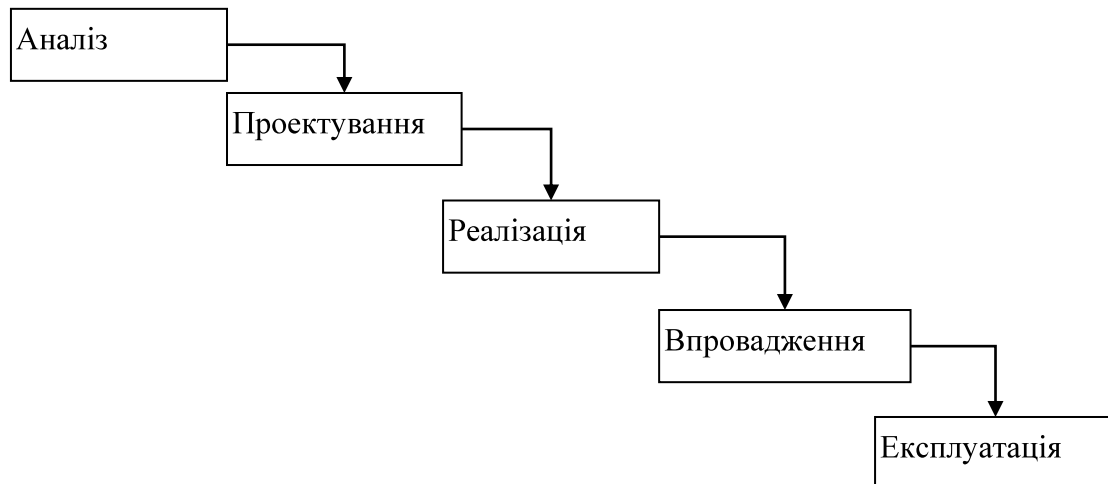


Рисунок 1.8 - Каскадна модель життєвого циклу

2. Ітераційна модель, зображена на рисунку 1.9, з'явилася в 70-80 роки, і є вдосконаленою каскадною моделлю. Для усунення такого недоліку своєї попередниці, як відсутність можливості виправлення помилок на попередніх етапах, в ній реалізується зворотний зв'язок. Проте, використання цієї моделі також можливо лише в разі нескладних проектів інформаційних систем.

Ітераційне повернення на попередні етапи дозволяє виправити невеликі недоліки в проектуванні. Інакше може виникнути потреба в перегляді раніше сформульованих вимог, а це може призвести до перепроєктування всієї інформаційної системи. При використанні цієї моделі для реалізації складних проектів виникає величезна кількість ітерацій, що призводить до заплутаності в документації і збільшує тривалість життєвого циклу.

Розподілення ресурсів при ітераційному підході відбувається аналогічно каскадному, з однією лише відмінністю – присутність зворотного зв'язку дозволяє повернутися на попередній етап, якщо на наступному виявлені неточності чи помилки розподілення. Але ця модель не вирішує питання розподілення ресурсів, які можуть з'явитися у процесі розробки, при цьому

додається суттєва проблема – велика кількість можливих переходів та повернень. Тобто, цей підхід характеризується великою кількістю дій при переході з етапу на етап. І при цьому ж, кількість цих дій не покращую та не вирішує проблему економічного та вигідного розподілення ресурсів при проектуванні інформаційної системи.

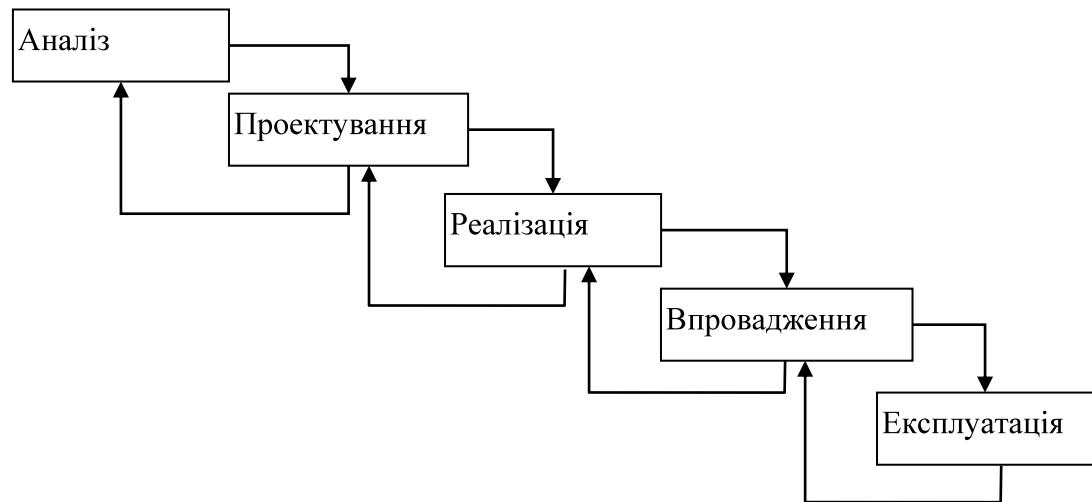


Рисунок 1.9 - Ітераційна модель життєвого циклу

3. Спіральна модель передбачає циклічне виконання всіх етапів каскадної моделі, внаслідок чого рівень реалізації технічних рішень перевіряється за допомогою прототипів. Кожен виток спіралі відповідає створенню фрагмента інформаційної системи, на ньому уточнюються цілі і характеристики проекту, визначається його якість і плануються роботи.

У основі спірального життєвого циклу (рисунок 1.10) лежить застосування RAD-технології. Інформаційна система розробляється шляхом розширення програмних прототипів, повторюючи шлях від деталізації вимог до деталізації програмного коду на кожному етапі. Даний життєвий цикл не має недоліків двох попередніх: модель дозволяє виконувати повернення до попередніх етапів, а невелике число ітерацій зменшує вірогідність виникнення помилок та невідповідностей. Вище перелічені позитивні сторони моделі говорять про те, що вона повністю відповідає особливостям сучасних проектів інформаційних систем.

Проте вдосконалення спіральної моделі ще не завершено, оскільки вона також має ряд недоліків: через високу адаптивність вона непридатна у випадках

контрактів з фіксованою ціною; великий обсяг проектної документації; відсутність чітких критеріїв виконання проекту.

Розподілення ресурсів у спіральній моделі відбувається спіралью, тобто на кожному з витків (етапів) уточнюються умови, економічність розподілення та перевіряється наявність нових ресурсів, відповідність та адекватність їх розподілу. За допомогою такої особливості є можливість корекції та перерозподілу ресурсів, повернення на попередні етапи, що дає змогу більш економічно та вигідно підходити до задачі розподілу та оптимізації. При цьому кількість ітерацій значно менша, це впливає і на швидкість проектування.

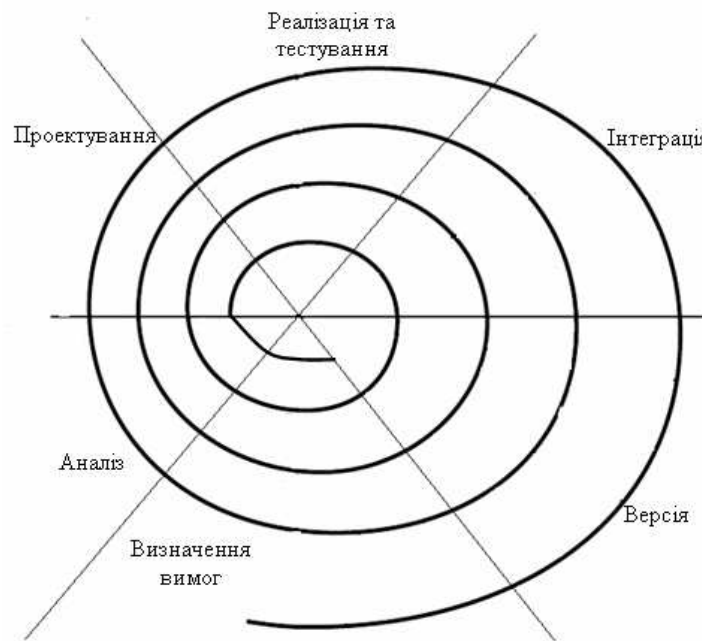


Рисунок 1.10 - Спіральна модель життєвого циклу

4. Модель Microsoft Solutions Framework (MSF) була розроблена компанією Microsoft для своїх власних цілей, але отримала доволі велику популярність і серед інших розробників. Модель, яка зображена на рисунку 1.11, увібрала в себе краще з трьох моделей, що описано вище, - спіральної, ітераційної та каскадної.

Проект реалізується поетапно, з наявністю відповідних контрольних точок, а сама послідовність етапів може повторюватися по спіралі. Складається з чотирьох етапів: аналіз, планування, розробка, стабілізація, розгортання. Кожен етап закінчується досягненням певного результату, наприклад, після аналізу проекту йде схвалення продуманої концепції. У результаті створюється

певний документ, в якому записується результат виконання даного етапу, для того, щоб кожен з розробників чітко розумів свою роль в проекті і задачі, які йому потрібно вирішити. Модель ітераційна і при проходженні всіх етапів, проект можна допрацювати з урахуванням попередньої ітерації. Але, оскільки, закінчення кожного етапу чітко вказане, то відсутнє нескінченне повторення одного й того ж процесу, як у спіральній моделі.

В цій моделі розподілення ресурсів відбувається найефективніше через те, що розробка йде по спіралі, враховує зміни та дає можливість перерозподілу в залежності від потреб та вимог, що виникають у процесі розробки інформаційної системи. У моделі MSF чітко визначено, хто яку роботу виконує, тому процес займає менше часу на узгодження виконавців на кожному з етапів. Треба зазначити, що ця модель має ітераційний характер, тому при розробці є змогу повернутися на потрібний етап за виконати зміни та корекцію попереднього помилкового розподілу. При цьому відсутня проблема постійного повтору ітерацій, так як кожен етап чітко окреслений по часу та має фіксований початок та кінець.



Рисунок 1.11 - Модель Microsoft Solutions Framework

Немає необхідності при проектуванні прив'язуватися до конкретних моделей життєвого циклу і процесів. Реалізація проекту передбачає вирішення ряду завдань, специфіка яких і визначає потрібні процеси проектування і послідовність їх вирішення. Отже, при проектуванні ІС на аграрних

підприємствах для підвищення ефективності процесу розробки слід використовувати різні життєві цикли і процеси. Каскадну і ітераційну модель життєвого циклу слід застосовувати в канонічній технології проектування, інші моделі - в індустріальній автоматизованій і типовій технологіях.

Враховуючи усі переваги та недоліки моделей, для проектування ІС на аграрному підприємстві доцільним є використання спіральної моделі життєвого циклу. Треба зазначити, що і модель MSF є придатною для виконання подібного завдання, проте надамо перевагу саме спіральній, через ряд причин. По-перше, саме у цій моделі першочергово робиться упор на предпроектну стадію, саме на той етап, де відбувається аналіз вимог та безпосереднє розподілення ресурсів. По-друге, легко можна вирішити основну проблему цієї моделі - визначення моменту переходу на наступний етап. Для необхідно ввести тимчасові обмеження на кожного з етапів життєвого циклу. Перехід здійснюється відповідно до плану, навіть якщо не уся запланована робота закінчена. План складається на основі статистичних даних, отриманих в попередніх проектах, і особистого досвіду розробників. По-третє, хоча у моделі MSF є доволі багато переваг, її реалізація складніша, займає більш часу, потребує більш глибокого аналізу та більшої кількості людей, вона представляє собою загальну методологію, тому для її використання треба проводити детальний аналіз актуальності, вибір методу. Все це займає час, ресурси, можливо і фінанси, а результат не буде настільки відмінним від результату спіральної моделі, щоб вибір саме цієї моделі визначити доцільнішим.

Від якості робіт, що проектуються, залежить ефективність функціонування системи, тому кожна стадія розділяється на ряд етапів і передбачає складання документації, що відображає результати робіт.

На **предпроектній стадії** можна виділити наступні етапи:

1. Збір матеріалів для проектування – передбачає розробку і вибір варіанта концепції системи, вияв всіх характеристик об'єкту і управлінської діяльності, потоків внутрішніх і зовнішніх інформаційних зв'язків, складу задач і спеціалістів, які будуть працювати в нових технологічних умовах, рівень їх підготовки.

На цьому етапі відбувається збір та зведення усіх ресурсів, які необхідні для проектування інформаційної системи та які у подальшому будуть потребувати аналізу та розподілу. Визначається найменування ресурсів, їх вплив на подальший процес проектування, збираються інші необхідні дані та характеристики.

2. Аналіз матеріалів і формування документації – складання завдання на проектування, затвердження техніко-економічного забезпечення.

На етапі аналізу вимог проходить структуризація вже зібраних раніше вимог. Ціль етапу – надати чіткий список не дубльованих вимог до системи, які мають бути виділені із надлишкових і частково дубльованих сценаріїв і користувацьких історій, які були отримані на попередньому етапі. Правильно згруповані вимоги допоможуть обійтися мінімальною кількістю функціонала для задоволення максимально більшої кількості цілей, а це, в свою чергу, допоможе зекономити бюджет і не дасть розповзтися рамкам проекту.

На цьому етапі відбувається чітке визначення характеру розподілу ресурсів, відбувається їх групування та аналіз, зазначається порядок розподілу та формуються основні вимоги, які будуть застосовуватися при розподілі цих ресурсів. Далі формується основний порядок. Чітко та правильно сформульовані вимоги, згруповані ресурси, коректно зазначені цілі розподілу та необхідна мета, яку треба досягнути, дозволяють на етапі предпроектної стадії окреслити майбутній проект та вплинути на ефективність інформаційної системи, яка проектується, зберегти кошти, час, людські ресурси та досягнути максимального прибутку та конкурентоспроможності. Тобто, саме на стадії планування та аналізу вимог необхідно розподіляти ресурси з урахуванням тієї моделі життєвого циклу, яка використовується при проектування певної інформаційної системи.

Стадія проектування поділяється на:

1. Етап технічного проектування – формуються проектні рішення по частинам, що забезпечують функціонування інформаційної системи, відбувається моделювання виробничих, господарських, фінансових ситуацій, здійснюється постановка задачі і блок-схеми, їх розв’язок. При цьому

фіксуються алгоритми та схеми, за якими відбуватиметься розподіл ресурсів, здійснюється основна постановка та мета розподілу, визначаються методи для її досягнення на практиці, обґрунтовуються характерні особливості того чи іншого методу, визначаються обмеження та фіксуються фактори, які впливають на систему.

2. Етап робочого проектування – здійснюється розробка, корегування структури, створення різноманітної документації: на поставку, на установку технічних засобів, інструкції по експлуатації, посадові інструкції.

Стадія впровадження інформаційної системи припускає:

1. Підготовку до вводу в експлуатацію – на цьому етапі відбувається встановлення технічних засобів, налаштування системи, навчання персоналу, пробне використання.

2. Проведення дослідів всіх компонентів системи перед запуском.

3. Здача в аграрну експлуатацію, яка оформлюється актом здачі-прийому.

На етапі функціонування інформаційної системи в робочому режимі не виключається корегування функцій і управляючих параметрів. Також здійснюється оперативне обслуговування і адміністрування.

Типи існуючих ресурсів підприємства, що використовуються на стадіях життєвого циклу, зведемо до таблиці 1.6.

Таблиця 1.6. Типи ресурсів підприємств

Типи ресурсу	Ресурси
Трудові	Працездатна частина населення, що наділена фізичними та інтелектуальними можливостями
Фінансові	Грошові кошти, що належать суб'єкту
Капітальні	Техніка та технології
Матеріальні	Засоби праці, відходи, паливо, полуфабрикати, сировина тощо
Науково-інформаційні	Знання, освіта людей, стан законодавства тощо

1.4 Підходи і методи розподілення ресурсів на стадії планування та аналізу вимог при проектуванні інформаційних систем

Проблему розподілення обмежених ресурсів можна вирішити декількома способами. Для математичної формалізації та дослідження цієї проблеми застосовують нечітку логіку, теорію графів та теорію матриць. Безпосередньо моделювання ситуації представляє собою ітеративний процес.

В процесі розвитку, а також по мірі зміни економічних умов підприємства можуть зіткнутися з необхідністю вдосконалення своїх економічних структур. Підприємства переглядають існуючі системи управління, застосовують нові ІС управління, проводять реорганізацію бізнесу. Ситуація зумовлює необхідність формування нових методичних основ і розробки практичних рекомендацій по побудові систем управління фінансами, в особливості інвестиційної діяльності як однієї з важливіших умов розвитку вітчизняних підприємств і систем, факторів збільшення ефективності виробництва. Для досягнення поставленої цілі необхідно ефективне управління матеріальними (основними фондами, виробничими запасами) і фінансовими ресурсами підприємства.

1.4.1 Постановка типових задач розподілення ресурсів

Проблема розподілення ресурсів відноситься до розряду "вічних": ресурси, на відміну від потреб, завжди обмежені. Їх, так або інакше, приходится розподіляти на різноманітні нужди постійно і на всіх рівнях. Прикладами таких задач розподілення ресурсів є динамічна задача оптимізації портфеля проектів, задача оптимізації фінансування ряду багатостадійних інвестиційних проектів в рамках деякої цільової програми з достатньо довгим строком реалізації. Динамічне програмування є одним з найбільш ефективних методів рішення подібних задач.

Загальна постановка подібної задачі виглядає наступним чином: для розвитку виділені капітальні вклади. Є кілька об'єктів вкладу, по кожному з яких відомий очікуваний прибуток, що отримується від вкладу певної суми коштів. Необхідно розподілити капітальні вклади між певною кількістю об'єктів

(підприємствами, проектами) таким чином, щоб отримати максимально можливий сумарний прибуток.

При такій постановці необхідно скласти математичну модель, яка у подальшому буде розв'язуватися методами динамічного програмування. Для цього виділяються наступні положення:

- прибуток від кожного підприємства не залежить від вкладу коштів в інші підприємства;
- прибуток від кожного підприємства виражається в одних умовних одиницях;
- сумарний прибуток дорівнює сумі прибутків, отриманих від кожного підприємства.

Дана постановка є спрощеною моделлю реального процесу розподілення інвестицій, і в "чистому" виді не зустрічається, так як не враховується деякі фактори, а саме: присутність "неформальних" критеріїв, тобто тих, які неможливо виміряти кількісно, у зв'язку з чим проекти можуть мати різний пріоритет; рівень ризику проектів; інші фактори.

У зв'язку з необхідністю врахування рівня ризику при проектуванні з'явилося стохастичне динамічне програмування, яке має справу з ймовірнісними величинами. Воно знайшло застосування в різноманітних областях, серед яких однією з найбільш широко досліджуваних є управління ризиковими фінансовими інвестиціями на підприємстві при проектуванні та використанні ІС.

1.4.2 Опис існуючих методів розподілення ресурсів, виявлення їх переваг та недоліків

В різноманітних виробничо-економічних системах значна кількість задач, що розв'язуються, тісно пов'язані з ефективним використанням обмежених ресурсів, що необхідні для нормального функціонування таких систем.

Серед методів розв'язання подібного роду задач виділяють наступні альтернативні методи та підходи, що дозволяють оптимально розподілити обмежені ресурси:

- Метод невизначених множників Лагранжа.
- Метод виключення змінних.
- Пошукові методи, в том числі "штучні нейронні мережі" та ін.
- Метод на базі нечіткої логіки.
- Метод на базі відкритого управління (метаігровий синтез).
- Методи на базі оптимального агрегування "виробничих функцій".

Метод невизначених множників або **метод невизначених множників Лагранжа** - метод знаходження умовного оптимуму, запропонований італійським математиком Жозефом-Луї Лагранжем. Він застосовується для вирішення задач з аналітичним виразами для критерію оптимальності і наявності обмежень на незалежні змінні типу рівності. Для отримання аналітичного рішення вимагається, щоб обмеження мали аналітичний вигляд. Застосування невизначених множників Лагранжа дозволяє звести завдання оптимізації з обмеженнями до завдання, що вирішується методами дослідження функцій класичного аналізу. В цьому випадку порядок системи рівнянь, що вирішується для знаходження екстремуму критерію оптимізації, підвищується на число обмежень. Застосування методу ефективно при кількості змінних три і менш. Метод використовується і при кількості змінних більше трьох, якщо процес описується кінцевими рівняннями.

Метод Гауса - **метод послідовного виключення** змінних, коли за допомогою елементарних перетворень система рівнянь приводиться до рівносильної системи ступінчастого (чи трикутного) виду, з якого послідовно, починаючи з останніх (по номеру) змінних, знаходяться усі інші змінні.

Метод повного виключення складається з кінцевого числа ітерацій і полягає в зведенні матриці коефіцієнтів до одиничного виду. Метод ґрунтується на двох операціях: 1) один з рядків розширеної матриці множать на множник, відмінний від нуля; 2) з кожного рядка віднімають один рядок, помножений на

деяке число. Кожне з таких елементарних перетворень призводить до нової системи лінійних рівнянь, що еквівалентна початковій системі.

Інтелектуальні системи на основі **штучних нейронних мереж** дозволяють з успіхом вирішувати проблеми розпізнавання образів, виконання прогнозів, оптимізації, розподілення ресурсів, керування тощо. Традиційні підходи до рішення цих проблем не завжди надають необхідну гнучкість.

На відміну від традиційної математики, що вимагає на кожному кроці моделювання точних і однозначних формулювань закономірностей, **нечітка логіка** пропонує абсолютно інший рівень мислення, завдяки якому творчий процес моделювання відбувається на найвищому рівні абстракції, при якому постулюється лише мінімальний набір закономірностей. Нечіткі числа, що отримуються, вільні від властивих останнім недоліків: мала кількість придатних до аналізу функцій розподілу, необхідність їх примусової нормалізації, дотримання вимог адитивності, трудність обґрунтування адекватності математичній абстракції для опису поведінки фактичних величин.

Недоліками нечітких систем є: відсутність стандартної методики конструювання нечітких систем; неможливість математичного аналізу нечітких систем існуючими методами; застосування нечіткого підходу в порівнянні з імовірнісним не призводить до підвищення точності розрахунку.

Метод оптимального агрегування має унікальні особливості - він мало чуттєвий до розмірності задачі і його можна використовувати для довільних виробничих функцій елементів.

Кожен з цих методів має свої переваги і недоліки. Якщо обирати з цих методів найкращий, то треба обрати усі методи та застосовувати їх комплексно. Однак Але при всьому цьому, усі дослідженні методи для вирішення задачі розподілення обмежених ресурсів, мають один суттєвий недолік. Їх головною особливістю є те, що жоден з них не враховує багатфакторний вплив зовнішнього середовища, який є неминучим при реальних умовах, тобто при вирішенні поставленої задачі не відбувається аналіз факторів, їх зміна та взаємний вплив кожного з них на усі інші. Все це дає підставу стверджувати, що жоден з цих методів, при використанні їх для проектування ІС не підійде, бо він

не зможе забезпечити необхідну надійність та не зможе на початковій стадії врахувати усі фактори, які так чи інакше будуть впливати на подальшу роботу аграрного підприємства.

Тому постає питання про те, щоб застосувати для цього іншу методику, яка б дозволила ще на початковій стадії аналізу і планування вимог визначити взаємовплив факторів один на одний, та врахувати ці зміни при розподіленні обмежених ресурсів при проектуванні інтегрованої ІС.

1.5 Когнітивне моделювання розподілу обмежених ресурсів на стадії планування та аналізу вимог

В наш час розподілення усіх типів ресурсів є актуальною проблемою на підприємстві, а на розробку та проектування інформаційних систем впливають безпосередньо, формуючи подальший процес роботи, визначаючи можливі збитки та прибутки, та, найголовніше, роблять підприємство конкурентоспроможним серед інших.

Саме тому темі розподілу ресурсів необхідно приділяти особливу увагу. Треба зазначити, що зробити це на етапі аналізу та планування вимог не так просто. Складнощі виникають насамперед через те, що необхідно враховувати вплив факторів зовнішнього середовища, а також робити аналіз, враховуючи, що і самі ресурси впливають один на одний або прямо, або через інші. Це суттєво погіршує та робить складнішим процес розподілу, збільшуючи можливість помилки та усіх наслідків від неї у подальшому.

Як було визначено у таблиці 1.6, на підприємстві мають місце основні 5 типів ресурсів: трудові, фінансові, капітальні, матеріальні та науково-інформаційні. Очевидно, що розподілити їх, не врахувавши взаємовплив – не виявляється можливим, а якщо і зробити це, то у результаті вийде хибні результати, які не нестимуть ніякої корисної інформації.

Трудові ресурси неможливо розподілити без врахування освіти, рівня знань, кваліфікації, тобто науково-інформаційних ресурсів. Їх вплив взаємопов'язаний та прямопропорційний, так як при більш високому рівні знань, більш вищій кваліфікації працівник має більші уміння, навички та інтелектуальні можливості.

Фінансові ресурси залежать від трудових, так як гроші, які необхідно витратити на зарплати працівникам, залежать від тієї посади, яку вона займає. В свою чергу, фінансові витрати на виробництво залежать від матеріальних ресурсів, а саме засобів праці, сировини та ін. В залежності від того, якими засобами праці виконується робота, які матеріали використовуються, формуються фінансові витрати підприємства. Теж саме можна сказати і про такий вид ресурсів, як капітальний. Ясно, що в залежності від програмних засобів, інформаційних вимог, видів та моделей техніки, яка використовується при роботі над проектом, залежить дуже багато. В тому числі і фінанси.

Для роботи з сучасними науково-технічними засобами потрібні кваліфіковані спеціалісти, які б були обізнані в основних та нових технологіях, вміли вправно працювати з технікою. Також тут можна врахувати кошти на проведення семінарів, навчання персоналу та підвищення рівня його кваліфікації.

Тобто, як видно, усі види ресурсів взаємопов'язані та мають вплив один на одний. Не врахувати при розподілі їх, неможливо. Тому стандартні методи оптимізації не підходять через те, що в них не вдосконалений процес врахування багатофакторного впливу. Саме тому є сенс при проектуванні інтегрованої ІС аграрного підприємства використовувати інший підхід, який дозволить врахувати проблему взаємовпливу. Це дозволяє зробити саме когнітивний підхід, який надає засоби для реалізації задачі оптимізації при розподіленні ресурсів, враховуючи при цьому, що ресурси, що розподіляються, впливають безпосередньо один на одний.

1.5.1 Аналіз факторів впливу при проектуванні інформаційної системи

Для прийняття рішень у процесі проектування ІС необхідно враховувати вплив багатьох факторів, аналізувати специфічні особливості підприємства та існуючі на ньому проблеми виробництва. На стадії аналізу та планування вимог це успішно дозволяють зробити засоби пізнавального (когнітивного) моделювання, засновані на когнітивному підході. Специфіка застосування

засобів когнітивного моделювання полягає в їх орієнтованості на конкретні умови розвитку ситуації на тому чи іншому підприємстві. За їх допомогою можна ще на початковій стадії планування успішно розподілити обмежені ресурси, що значно полегшить подальший процес проектування інформаційної системи на аграрному підприємстві.

Когнітивні технології дозволяють враховувати такі людські показники, як психічний стан і процеси реагування на оточуюче середовища. Головною перевагою когнітивного моделювання є можливість систематичного якісного, а не кількісного врахування наслідків рішень, що приймаються у процесі проектування, та можливість виявлення побічних ефектів, які можуть заважати реалізації навіть очевидних рішень. Як правило, побічні ефекти важко оцінити інтуїтивно саме через велику кількість факторів та наявність багаточисельних траєкторій взаємодії між ними.

Ще одною перевагою є можливість пошуку рішень в умовах різкої зміни навколишнього середовища та при недостатній інформації про фактори моделі і ступеня їх взаємодії. При формальному виділенні факторів і зв'язків між ними неминуче виявляються раніше не враховані аспекти ситуації, зв'язки, що здавалися на перший погляд несуттєвими. Також в процесі аналізу формується система понять, в термінах якої навіть неформальне обговорення проблеми стає більш чітким і обґрунтованим.

Треба зазначити, що у когнітивного моделювання є і свій суттєвий недолік: треба застерігатися завищених очікувань при використанні когнітивного підходу у проектуванні ІС, навіть у тому випадку, коли побудована модель визнана адекватною. Це спричинено тим, що на кожному етапі проектування когнітивної моделі треба приймати рішення, які впливають на подальший хід процесу проектування. І від сукупності цих рішень залежить адекватність побудованої моделі. Тобто, необхідно дуже обережно та відповідально підходити до прийняття кожного з рішень, навіть проміжних. На кожному з етапів аналізувати та лише після цього робити висновок і остаточно формувати наступний крок.

1.5.2 Опис методу когнітивного моделювання розподілу обмежених ресурсів

Під когнітивний підходом розуміється рішення традиційних для цієї науки проблем методами, що враховують когнітивні аспекти, в які включаються процеси сприйняття, мислення, пізнання, пояснення і розуміння. Когнітивний підхід в будь-якій предметній області акцентує увагу на "знаннях", вірніше, на процесах їх представлення, зберігання, обробки, інтерпретації і виробництві нових знань. Когнітивний підхід має десятки ракурсів. Велика увага приділяється питанням розуміння природної мови, комп'ютерного перекладу, проблемам комп'ютеризації суспільства і теорії штучного інтелекту.

Для складання когнітивної моделі предметної області необхідно: виділити список значущих чинників; побудувати матрицю взаємовпливу; визначити початкові тенденції зміни чинників.

Маючи, таким чином, складену модель можна вирішити такі завдання, як: моделювати саморозвиток ситуації; моделювати керований розвиток ситуації; шукати необхідні управління.

В цілому когнітивні технології сприяють рішення широкого круга завдань і проблем, таких як:

- Моделі і методи інтелектуальних інформаційних технологій і систем для створення геополітичних, національних і регіональних стратегій соціально-економічного розвитку.
- Моделі виживання "м'яких" систем в середовищах, що змінюються, при дефіциті ресурсів.
- Ситуаційний аналіз і управління розвитком подій в кризових середовищах і ситуаціях.
- Інформаційний моніторинг соціально-політичних, соціально-економічних і військово-політичних ситуацій.
- Розробка принципів і методології проведення комп'ютерного аналізу проблемних ситуацій.
- Вироблення аналітичних сценаріїв розвитку проблемних ситуацій і управління ними.

- Підготовка рекомендацій за рішенням першочергових стратегічних проблем на основі комп'ютерної системи аналізу проблемних ситуацій.
- Моніторинг проблем в соціально-економічному розвитку корпорації, регіону, міста, держави.
- Технологія когнітивного моделювання цілеспрямованого розвитку регіону.
- Аналіз розвитку регіону і моніторинг проблемних ситуацій при цілеспрямованому розвитку регіону.
- Моделі для формування державного регулювання і саморегулювання споживчого ринку.
- Аналіз і управління розвитком ситуації на споживчому ринку.

Ключовим поняттям в когнітивному моделюванні є "когнітивна карта", що є зважений орієнтований граф, в якому вершини взаємно-однозначного відповідають чинникам, в термінах яких описується предметна область, а дуги відображують безпосередні зв'язки (взаємовплив) між чинниками. Взаємовплив може бути позитивним, (збільшення/зменшення одного чинника приводить до збільшення/зменшення іншого) і негативними (збільшення/зменшення одного чинника приводить до зменшення/збільшення іншого). Для відображення міри впливу використовують сукупність лінгвістичних змінних і відповідну їй сукупність числових значень з інтервалу $[0, 1]$: "дуже слабе" - 0,1, "помірне" - 0,3, "істотне" - 0,5, "сильне" - 0,7 і "дуже сильне" - 0,9 (допустимі і проміжні значення).

Когнітивні карти - це ментальні репрезентації (представлення) людини про просторову організацію довкілля. Помилки і спотворення в когнітивних картах розглядаються з точки зору психофізичного і функціонального підходу.

Когнітивні карти відносяться до ранніх компонент пам'яті, вони є схемою або ескізом для формування інших предметних віддзеркалень. У загальному випадку, когнітивна карта - це суб'єктивна картина, яка створюється в результаті активних дій в довкіллі і має просторові координати, вона визначає місце розташування окремих сприйманих предметів.

Види когнітивних карт: карта-шлях (послідовне представлення зв'язків між об'єктами по певному маршруту); карта-огляд (одночасне представлення просторового розташування об'єктів).

Когнітивні карти дають можливість людині орієнтуватися в просторі, рухатися і здійснювати свою діяльність. Процес створення когнітивних карт тривалий, він проходить ряд психологічних трансформацій, завдяки яким людина отримує, зберігає, згадує і використовує інформацію про просторове оточення. Інформація для роботи когнітивної карти збирається за результатами спеціального тестування і роботи контрольних апаратних засобів. Використання когнітивних карт можливе при розробці програмних комплексів підтримки ухвалення рішень по управлінню вищих навчальних закладів, підприємств і різних систем.

1.5.3 Карти взаємовпливу факторів

Ключовим поняттям у когнітивному моделюванні є когнітивна карта, яка представляє собою орієнтований граф, в якому вершини взаємно однозначно відповідають факторам, а дуги відображають зв'язки між цими факторами.

Для побудови когнітивної моделі у галузі суднобудування необхідно виділити фактори, які будуть впливати на подальше проектування інформаційної системи та проаналізувати їх взаємовплив.

Врахуємо наступні фактори, які впливають на проектування інформаційної системи:

X_1 – надійність – структура апарату, яка забезпечує достовірне відображення фактичного стану виробництва і результатів прийнятих рішень;

X_2 – рівень виробництва, який визначає швидкість виконання операцій у системі;

X_3 – ефективність – рівень перебільшення прибутків над витратами на виробництво;

X_4 – економічність – процес виконання поставлених задач та вимог з найменшим рівнем витрат;

X_5 – кваліфікація співробітників – рівень освіти персоналу, що працюють на підприємстві;

X_6 – підтримка системи, яка полягає в сервісному обслуговуванні системи в процесі експлуатації;

X_7 – модернізація, що характеризує придатність системи до змін, які підвищують характеристики функціонування;

X_8 – можливість тестування, що забезпечує моніторинг функціонування системи;

X_9 – кількість співробітників – кількість робітників, що працюють на підприємстві або у певній структурі цього підприємства;

X_{10} – якість функціонування системи;

X_{11} – кількість рівнів управління.

Видно, що усі названі фактори є взаємопов'язаними, і зміна того чи іншого з них впливає на усі інші, а значить і на загальний рівень роботи усього підприємства. Саме тому важливим є не тільки врахування зміни кожного окремого з факторів, але і реакції інших на цю зміну. Необхідно проаналізувати рівень впливу кожного з факторів один на один та встановити ймовірність, з якою кожний фактор буде впливати на усі інші. Це дасть у подальшому можливість більш ефективно спроектувати інформаційну систему, яка матиме змогу забезпечити підприємству прибуток, більшу активність та збільшить його конкурентоспроможність.

У наступному розділі детально розглянемо принципи побудови когнітивних матриць, карт и моделей, та на прикладі представимо схему урахування впливу факторів у когнітивній моделі на конкретні математичні моделі розподілу ресурсів.

2. ФОРМУВАННЯ ВИМОГ ДО ІС, РОЗРОБКА КОНЦЕПЦІЇ І ПОСТАНОВКА ЗАДАЧІ

2.1 Опис предметної області

2.1.1 Аналіз структури агротехнічного підприємства

Агротехнічне підприємство має складну структуру та складається з багатьох підсистем різних напрямів та виробничих рівнів. Так, можна виділити наступні рівні існуючої системи:

- Директорат

Директори та керівники вищого рівня

- Підсистема загального обслуговування

До цієї системи можна віднести економічний, юридичний, інформаційний та інші системи, що забезпечують загальне існування підприємства, однак не задіяні безпосередньо у процесі виробництва

- Підсистема виготовлення кінцевої продукції

Підсистема охоплює забезпечення роботи заводу, що з сировини, врожаю, виготовляє кінцевий продукт

- Транспортна підсистема

Поточна частина роботи підприємства не може розглядатися як складова якоїсь підсистеми, більш того, її важливість не можна не дооцінювати: до неї входить координування та управління машинно-транспортним парком, розподілення навантажень на техніку, її ремонт та обслуговування.

- Агрономічна підсистема

До підсистеми можливо віднести комплекс робіт, що здійснюють агрономи та спеціалізовані працівники для догляду за врожаєм, аналізом стану рослин, плануванням сільськогосподарських робіт.

- Допоміжні підсистеми

Охоронна система, відділ забезпечення, роботи з догляду за приміщенням можна віднести до цієї групи як необов'язкові, однак потрібні для повноцінної та комфортної роботи підприємства.

Загальний вигляд структури підприємства зображено на рисунку 2.1.

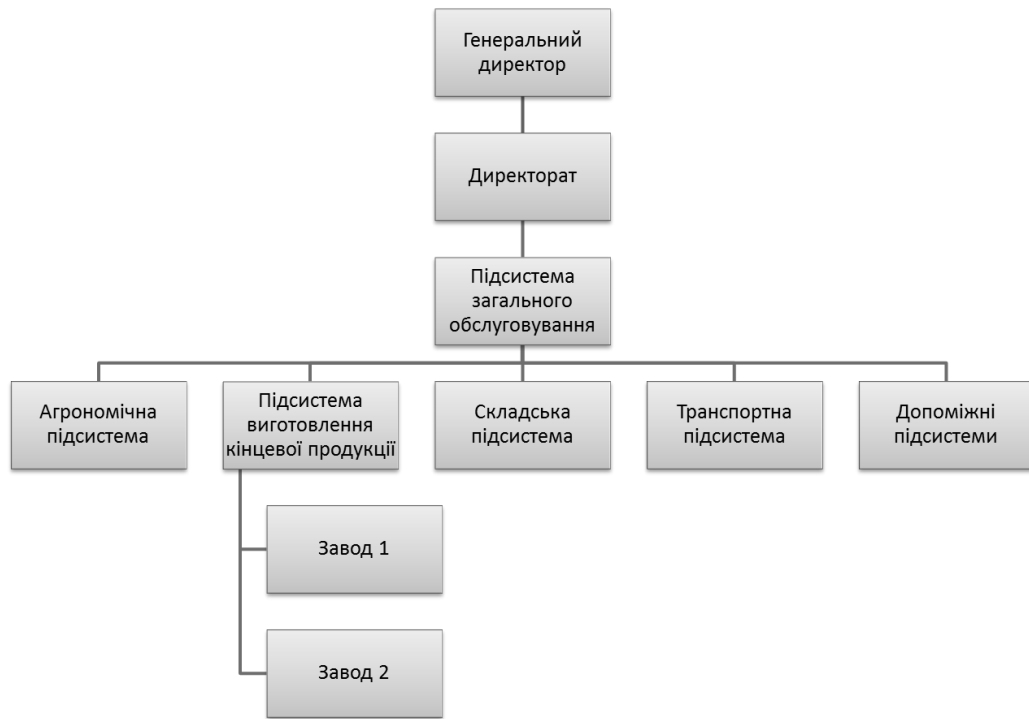


Рисунок 2.1. Структури агротехнічного підприємства

2.1.2 Аналіз наявних проблем

Оскільки агротехнічне підприємство виступає як складна система, що доведено у попередньому підрозділі, кожна її складова функціонує як окреме ціле. Таким чином, на кожному рівні функціонування підприємства можуть виникати певні проблеми та недоліки, що можуть знижати якість виробництва, а іноді і просто зривати графіки роботи, таким чином призводячи до зниження доходів, підвищення розходів, а, в результаті всього, до зниження основного економічного показника ефективності роботи підприємства — зниження прибутку.

До проблем, що виникають у процесі виробництва та можуть бути зафіксовані виробниками можуть належати як розбите вікно, так і незначні відхилення у роботі техніки(що також потрібно проаналізувати), зламаний верстат, чи вада кваліфікованої робочої сили для виконання певних робіт.

Іноді робітники пишуть службові записки, у яких обґрунтовують необхідність тих чи інших заходів, іноді скарги та побажання надходять у усній формі. Тим не менш, написання записки є процесом, що потребує певного часу,

присутності людей, яким її можна віддати, а повідомлення в усній формі нерідко забуваються через велику кількість поточних завдань.

Більш того, навіть при надходженні та розгляді завдань можуть виникати ускладнення: проблеми надходять, найчастіше, до підсистеми, у якій вони виникають, та для їх вирішення необхідне втручання інших відділів. Координація, домовлення також може займати певний час.

Таким чином, основною проблемою системи моніторингу стану підприємства, є її децентралізованість, відсутність єдиного центру збору та обробки інформації про проблеми, перерозподілення робіт з їх виправлення по підрозділам.

2.1.4 Формування вимог до ІС

Після виконання опису предметної області та аналізу основних проблем на підприємстві, можна дійти висновку, що розроблювальна ІС місце повинна мати такі властивості, як:

- Централізованість

Обробкою та розподіленням проблем повинна займатися конкретна спеціально назначена людина - диспетчер

- Зручність засобів подання заявок

Так, необхідно насамперед проаналізувати засоби, що будуть зручними для працівників всіх рівнів для подання повідомлень. Крім того, ці засоби також повинні подавати на обробку найбільш повні необхідні для фіксації проблеми дані. Опис подібних засобів наведено у таблиці 2.1.

Таблиця 2.1. Можливі засоби збору інформації про проблеми на підприємстві

Поштові скриньки на заводах	На заводах деякі працівники не мають часу писати службові записки, деякі погано розбираються в техніці (телефон, комп'ютер, тощо), тому для подібних людей набагато зручнішими будуть поштові скриньки, встановлені на заводах, куди вони зможуть кидати зафіксовані ними проблеми. Раз на тиждень, чи будь-який інший зазначений термін оператор відкриватиме скриньки та обробляти подані заявки, вбиваючи їх в систему завдань
-----------------------------	--

Телефон гарячої лінії	Іноді доповідати про проблему набагато швидше за допомогою телефону гарячої лінії. Це потрібно, наприклад, якщо проблема повинна мати, наприклад, додатковий опис, деталі, що потребують уточнення.
Електронна пошта	Заявку, що доповідає про проблеми можна буде подати також за допомогою електронної пошти. На відміну від працівників заводів, працівники офісу найчастіше використовують у роботі для надсилання повідомлень електронну пошту. За цих обставин слід передбачити створення спеціальної електронної скрині для отримання повідомлень про проблеми.
Подання заявки через форму на сайті	Оскільки підприємство має свій сайт – на сайті також можна передбачити форму зворотнього зв'язку для скарг та побажань.
Мобільний додаток	Деякі робітники підприємства, наприклад агрономи та трактористи, мають робочі телефони з встановленими на них програмами, що необхідні для роботи, контролю своєї роботи, складання звітів. В результаті цього ці працівники звикли пов'язувати свою роботу з мобільним пристроєм. Відповідно, набагато зручніше для них буде доповідати за допомогою спеціально розробленого додатку для смартфона.

- Зручність обробки даних у системі;
- Зручність системи накопичення даних про проблеми та відстеження їх виконання

Розроблювана система насамперед призначена для автоматизації роботи з моніторингу та вирішення проблем наявних на підприємстві. Функції, що автоматизуються системою, пов'язані з організацією робочого часу працівників, розподіленням робіт, отриманням завдань з улагодження проблем та складанням документообігу, як то звіт та статистичні показники ефективності роботи ІС.

Система є середовищем, складовими елементами якого є комп'ютери, об'єднані в комп'ютерні мережі, програмні продукти, бази даних, люди, різного роду технічні і програмні засоби зв'язку тощо. Основна мета ІС — організація збереження, обробки і передачі інформації. ІС являє собою людино-комп'ютерне оточення (систему) для обробки інформації.

ІС, що розробляється, є розподіленою, тобто для її функціонування, крім програм-клієнтів, необхідна наявність принаймні одного сервера програм. Програмне забезпечення сучасних ІС має наступні логічні рівні:

- Рівень користувача (інтерфейс і сервіси користувача);
- Прикладний рівень (прикладні сервіси);
- Рівень даних (сервіси доступу до даних і сховище даних).

Опишемо функції кожного рівня:

- Рівень користувача: відображає дані і дозволяє користувачу редагувати їх. Існує два основних типи інтерфейсу: віконний (реалізується засобами підсистеми інтерфейсу користувача операційної системи) і на основі Web. Web-інтерфейси засновані на HTML, DHTML, XHTML чи XML, у результаті чого вони можуть відображатися будь-яким Web-оглядачем на будь-якій платформі.

- Прикладний рівень: тут реалізовані бізнес-правила й обмеження на дані (перевірка коректності даних). І хоча його сервіси використовуються презентаційним рівнем, він не прив'язаний до якого-небудь клієнта – сервіси прикладного рівня доступні будь-якому клієнту. Бізнес-правила виражаються у формі прикладних алгоритмів, корпоративних правил і т.д. Обмеження на дані гарантують точність і цілісність інформації, що зберігається. Бізнес-правила звичайно реалізуються окремим модулем на централізованому сервері, що дає можливість доступу до нього відразу декільком клієнтам.

- Рівень даних: прикладний рівень не має інформації про те, як і де зберігаються дані, які він обробляє. У цьому питанні він покладається на сервіси доступу до даних, що виконують всю роботу з отримання і передачі даних. Сервіси доступу до даних також реалізуються у виді ізольованих модулів, які знають про місце збереження інформації. Кожен модуль доступу до даних, як правило, відповідає і за цілісність сховища (наприклад, реляційної бази даних). Для багаторівневих програм у якості сховища інформації підходять сервера баз даних (MS SQL Server, Oracle, Sybase, MySQL, PostgreSQL і ін.), необхідні для обслуговування даних у таблицях і високопродуктивної виборки інформації.

Реалізація функцій ІС неможлива без знання орієнтованих на нього інформаційних технологій. Нижче представлений опис технологій, які використовуються на кожному рівні інформаційної системи, що розробляється.

Технології рівня користувача

Web-оглядач і Web-інтерфейс

Web-оглядач чи Web-браузер – це програма, що формує стандартний запит для Web-сервера відповідно до правил протоколу HTTP (Hyper Text Transfer Protocol), одержує документи від Web-сервера, інтерпретує і відображає їх на екрані користувача.

Фактично Web-оглядач є єдиною програмою, яка необхідна користувачу для роботи з інформаційною системою; він надає відповідний Web-інтерфейс.

Web-інтерфейс широко розповсюджений і пропонує готові засоби відображення. Web-оглядачі, що вільно розповсюджуються, мають прості інтерфейси, що збільшує ефективність розгортання програм. Найпоширеніші Web-оглядачі: MS Internet Explorer, Safari, Mozilla FireFox, Opera.

HTML

Найпоширеніший і простий спосіб створення Web-інтерфейсу – використання HTML (HyperText Markup Language, Мова розмітки гіпертексту). Web-інтерфейс складається з набору HTML-документів, статичних чи генерованих динамічно. HTML являє собою набір команд, що називаються тегами, які розташовуються між фрагментами тексту й усередині них. Команди HTML дозволяють структурувати документ, виділяючи в ньому частини тексту (заголовки різних рівнів, абзаци, перерахування і т.д.), які логічно розрізняються. У результаті кожен Web-оглядач може формувати текст документа таким чином, щоб найкращим способом відобразити його на конкретному дисплеї. Для надання документам більшої виразності текст звичайно форматується з використанням збільшених розмірів шрифту заголовків, застосуванням напівжирного і курсивного накреслень для важливих термінів, виділенням пунктів списків і т.д. Мова HTML дозволяє також включати в документи ілюстративну графіку, що може бути відображена

програмами перегляду, що основані на використанні графічного інтерфейсу користувача

Однією з найважливіших властивостей HTML є можливість включення в документ гіпертекстових посилань. Ці посилання дозволяють користувачу завантажити новий документ на свій комп'ютер, просто клацнувши вказівником миші в тій місці екрана, де розташоване посилання. Будь-який документ може містити посилання на інші документи. Областю документа, що використовується як посилання, може служити слово, група слів, графічне зображення чи навіть заданий фрагмент зображення.

Іншою не менш важливою властивістю HTML є форми. HTML-форми дають розробнику Web-інтерфейсу можливість змішування традиційних елементів, властивих друкованим сторінкам, з інтерфейсом користувача сучасної віконної системи. Механізм форм дозволяє розміщати серед форматowanego тексту з ілюстраціями кнопки, меню, що розкриваються, текстові поля й інше. Ці елементи інтерфейсу дають користувачам, що переглядають сторінки, можливість передачі відгуку, перетворюючи їх із простих глядачів в учасників, що мають право голосу. Головне обмеження HTML-форм пов'язано з тим, що Web-оглядачі змушені здійснювати нове з'єднання з сервером при кожному отриманні HTML-сторінки чи передачі даних форми.

HTML сам по собі не дозволяє створювати інтерактивні документи, тобто такий Web-інтерфейс є статичним. Існує ще одна технологія (DHTML), що дозволяє Web-сторінкам бути більш динамічними. DHTML (Dynamic HTML, динамічний HTML) дозволяє змінювати вид сторінки Web і її стиль після того, як документ завантажився.

Ключовим поняттям DHTML є шари. У звичайному HTML-документі, всі елементи так чи інакше впливають один на одного при формуванні його зовнішнього вигляду. Тобто абзац тексту "зміщує" наступні елементи на деяку відстань униз, картинка, вирівняна вправо, "змушує" текст обтікати її і так далі. Всі елементи разом утворюють так званий HTML-потік чи потік документа (document flow). Шари у свою чергу – це частини сторінки, що не входять в

основний HTML-потік. Для кожного з них створюється свій потік, що ніяк не впливає на основний. При цьому вміст шарів відображається, як правило, над (зверху) основним змістом сторінки. Шари можна пересувати і робити невидимими. Таким чином, за допомогою шарів можна створювати різні інтерактивні ефекти (наприклад, деревоподібне меню як у провіднику Windows).

Одне з обмежень DHTML полягає в тому, що сторінки, створені з застосуванням технології DHTML, можна переглядати тільки в четвертій і більш пізніх версіях Web-оглядачів MS Internet Explorer і Netscape Navigator.

DHTML заснована на технологіях CSS, JavaScript і HTML.

CSS

CSS (Cascading Style Sheets, каскадні таблиці стилів) – ще один стандарт, розроблений W3C. Існують CSS Level 1 і CSS Level 2. Далі мова йде про CSS2.

CSS визначив правила опису стилів візуального відображення елементів HTML-документів. Основні особливості CSS:

- CSS — це мова, що дозволяє приєднувати стилі до будь-яких структурованих документів. На сьогодні такими є HTML-документи і XML-документи.
- CSS поширив поняття стилю відображення на друкувальні пристрої, синтезатори мови й інші пристрої відображення документів.

Ціль створення каскадних таблиць стилів полягала в тому, щоб відокремити структуру документа (описану мовою HTML чи XML) від правил його відображення на різних пристроях (що задаються таблицями CSS). За допомогою стилів можна визначити властивості елементів (текст, гіпертекстове посилання, абзац, заголовок, рисунок, таблиця й інші) у HTML-документі.

Існує три способи завдання стилів у HTML-документі: зовнішні таблиці стилів, внутрішні таблиці стилів і таблиці стилів елементів.

JavaScript

Мова JavaScript – інтерпретована об'єктно-базована мова високого рівня, що дозволяє писати програми, які виконуються в середовищі Web-оглядача. В даний час JavaScript є найбільш широко розповсюдженою мовою в Web для написання сценаріїв, що виконуються на стороні клієнта.

JavaScript дозволяє вбудовувати команди в HTML-документи. Команди JavaScript можуть виконуватися при завантаженні документа, при заповненні полів форм, при наведенні вказівника миші на елемент документа й інше. Мова JavaScript має досить широкі можливості. Вона не дозволяє працювати на рівні машинних кодів, однак дозволяє отримати доступ до багатьох можливостей Web-оглядачів, HTML-документів, а іноді й операційної системи, у якій працює Web-оглядач. На відміну від Java чи C, програми на JavaScript не потрібно компілювати, а оглядачу не потрібно завантажувати віртуальну машину для виконання програмного коду.

Одна з причин, згідно якої Web-розробники використовують JavaScript, – можливість виконання на стороні клієнта багатьох функцій, що раніше виконувалися винятково на стороні сервера. Кращим прикладом є перевірка HTML-форм. За допомогою JavaScript елементи форми можна перевірити до того, як користувач передасть інформацію Web-серверу. Це приводить до зменшення обсягу транзакцій HTTP.

JavaScript широко використовується для створення DHTML-документів. JavaScript дозволяє переміщатися по дереву (ієрархії) елементів HTML-документа, змінювати його властивості і багато чого іншого.

Крім JavaScript існує ще одна мова для вирішення аналогічних задач – VBScript. Однак вона, на відміну від JavaScript, не отримала такого ж широкого поширення.

Технології прикладного рівня

Web-сервер

Web-сервер (HTTP-сервер) – програма, що сприймає запити клієнтів (наприклад, Web-оглядачів), які приходять, як правило, на стандартний порт TCP/IP (порт 80), і виконує в найпростішому випадку пересилання HTML-документів клієнту. Web-сервер є програмою, яка дозволяє службі Internet, що називається WWW (World Wide Web, Всесвітня інформаційна мережа), знаходити і використовувати гіпертекстові і графічні документи на різних серверах.

Фактично Web-сервер передає інформацію користувачу, який знаходиться на віддаленому комп'ютері. Користувач переглядає інформацію за допомогою Web-оглядача (MS Internet Explorer, Netscape Navigator і інші) і відсилає нові запити Web-серверу.

Web-сервер як правило має вбудовану систему аутентифікації по логіну та пароллю, яку можна використовувати для обмеження доступу до ресурсів сервера.

Найбільш розповсюджені Web-сервера: Apache Server, MS IIS, Netscape Enterprise Server.

CGI

CGI (Common Gateway Interface) – це набір правил, згідно з якими програми на стороні сервера можуть через Web-сервер пересилати дані клієнтам. CGI є перевіреним і універсальним способом створення розподілених систем у Internet. Головна задача, яку він вирішує, – це забезпечення викликів віддалених процедур між клієнтом і сервером з використанням сокетів TCP/IP.

Крім CGI існують ще кілька подібних технологій – ISAPI/NSAPI, Servlet і JSP. Ці чотири технології призначені для вирішення приблизно тих самих задач і концептуально дуже схожі. Всі вони дозволяють розширити можливості Web-серверів шляхом розробки програм, зареєстрованих на цьому сервері.

Servlet і JSP

Технологія Servlet була розроблена компанією Sun Microsystems, і була призначена для того щоб серверна частина отримуючи запит від клієнта у якості відповіді повернула згенеровану HTML сторінку яка і відображає дані. Servlet – це спеціальний Java клас, який реалізує необхідні API для отримання запитів від WEB клієнта та після обробки надання результату. Сам по собі Servlet не чого робити не вміє він є частиною системи яка знаходиться у контейнері (Java web сервер) який вміє отримати запит і передати його необхідному додатку. Технологія JSP (JSPx) є „надбудовою” над сервлетами і призначена для більш наглядного конструювання інтерфейсу. JSP є наслідником Servlet і являє собою комбінацію Java коду та HTML, на зразок PHP.

Java web сервери

Існує декілька Java web серверів зокрема такі, JBoss, Tomcat, Glassfish та ін. Майже усі вони виконують одні і ті ж самі задачі:

- компіляція JSP сторінок;
- зв'язок з базою даних
- обмін даними між програмою та клієнтом;
- обмін даними між програмою та БД;

Технології рівня даних

Мова SQL

SQL – це скорочена назва структурованої мови запитів (Structured Query Language). SQL – це мова програмування, яка використовується для виконання запитів і модифікації даних, а також для управління СУБД (системами управління базами даних). SQL працює тільки з базами даних одного певного типу, які називаються реляційними.

Стандарти SQL затверджені ANSI і ISO. Хоча в кожній СУБД використовується власний діалект SQL, більшість цих діалектів задовольняють стандарту ANSI SQL. Таким чином, для сумісності програмного забезпечення з різними СУБД, необхідно строго дотримуватися стандарту ANSI SQL.

Мова SQL містить оператори, які підрозділяються на дві категорії: мова опису даних (data definition language, DDL) і мова маніпулювання даними (data manipulation language, DML). DDL використовується для створення і управління такими об'єктами, як бази даних, таблиці і представлення. Оператори DDL містять команди CREATE, ALTER і DROP для кожного з об'єктів. DML використовується для управління даними в об'єктах баз даних за допомогою таких операторів, як SELECT, INSERT, UPDATE і DELETE.

Сервер баз даних

Сервер баз даних – це окрема програма, яка виконується як окремий процес (сервіс). Сервер передає обрану з бази інформацію по каналу клієнту. Саме сервер працює з даними, піклується про їхнє розміщення на диску.

Найрозповсюджені сервера баз даних: MS SQL Server, Oracle, Sybase, MySQL, PostgreSQL.

EJB

EJB – ще одна технологія від Sun яка призначена для реалізації Моделі у Java програмі. Вона реалізує зв'язок з БД не залежно від реалізації, та представлення у вигляді ява об'єктів. EJB поділяється на дві складові

- EntityBeans (класи які є об'єктним представленням інформацією з БД)
- SessionBeans (класи та інтерфейси які контролюють, змінюють, і тд EntityBeans)

Динаміка роботи ІС

Маючи загальне уявлення про технології, які використовуються для розробки ІС, можна описати динаміку роботи системи:

- Користувач запускає Web-оглядач на віддаленому комп'ютері-клієнті і вводить адресу Web-сервера інформаційної системи.
- Web-оглядач надсилає запит Web-серверу.
- Web-серверу надсилає запит Java-серверу.
- Java-сервер передає запит відповідному додатку (системі).
- Система отримує запит і передає його відповідному сервлету (JSP).
- Servlet(JSP) обробив запит звертається (якщо необхідно) до SessionBean
- SessionBean звертається до Java-сервера з проханням надати зв'язок з необхідним сервером БД та проводить аутентифікацію користувача по логіну і пароллю та вибирає потрібну базу.
- Отримавши зв'язок SessionBean проводить необхідний запит і отримавши результати повертає їх контролеру (Servlet або JSP)
- Контролер отримав результат генерує необхідну HTML сторінку та повертає її і передає її Web-серверу.
- Web-сервер додає HTTP-заголовок до отриманого документа і відправляє його Web-оглядачу.
- Web-оглядач приймає HTML-документ і відображає його на екрані користувача.
- Користувач отримує Web-інтерфейс і необхідну інформацію. Він може надіслати новий запит Web-серверу.

Таким чином, вимальовується схема взаємодії технологій (рисунок 2.2)

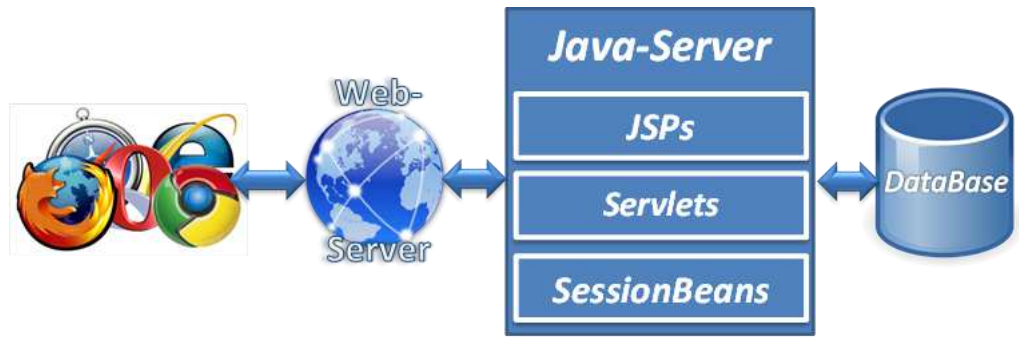


Рисунок 2.2 – Схема взаємодії технологій ІС

2.2 Формування вимог до інформаційної системи

2.2.1 Характеристика об'єкта автоматизації

Об'єктом автоматизації є комплексна система моніторингу стану агротехнічного підприємства. Підприємство має потребу у створенні єдиної бази даних моніторингу та збору даних, у тому числі про наявні ускладнення, а також систему управління, розподілення, створення та отримання завдань, збору інформації про проблеми. Крім того, необхідно забезпечити можливість проводити статистичний аналіз та зберігати історію зафіксованих проблем, переглядати статус заявок, що надійшли до системи.

2.2.2 Мета, критерії та обмеження створення ІС

Метою розробки ІС є збільшення продуктивності та ефективності роботи підприємства, підвищення прибутку, поліпшення умов праці працівників шляхом автоматизації та оптимізації системи моніторингу стану підприємства.

Основні критерії створення ІС:

- **Соціальний критерій.** ІС розрахована на користувачів з різною комп'ютерною підготовкою. Необхідний зручний і інтуїтивно зрозумілий інтерфейс для роботи з нею.

- **Економічний критерій.** ІС створюється для підприємства безкоштовно. Вартість ІС буде складатися з витрат на придбання необхідного програмного, технічного й іншого видів забезпечення.

- **Критерій мобільності.** Користувач повинен мати можливість працювати із системою як з комп'ютера локальної мережі, так і з будь-якого комп'ютера, що має доступ до Internet.

- **Критерій безпеки.** ІС моніторингу стану підприємства повинна бути безпечною з погляду збереження й обробки даних. Втрата даних може призвести до серйозних збоїв у роботі підприємства. Необхідно забезпечити захист системи від несанкціонованого втручання.

2.2.3 Функції та задачі ІС

Задачею ІС, що створюється, є автоматизація наступних функцій:

- Обробка інформації про наявні на підприємстві проблеми;
- Облік робіт по улагодженню проблем та інформації про їхній статус;
- Формування звітів;
- Усі інші вимоги не є головними і можуть змінюватися під час розробка або експлуатації.

Для створення ІС необхідно розробити єдину базу даних для збереження інформації про відділи та їх співробітників, про отриманні ними завдання та інше, створити базу даних, до якої будуть добавлятися відомості про стани робіт, ускладнення та їхні статуси. Повинно бути передбачене програмне забезпечення для адміністрування системи, надсилання до системи заявок. Необхідно також внести кадрові зміни та найняти людину на посаду диспетчера відділу.

2.2.4 Висновки і пропозиції

Таким чином, необхідно розробити ІС, що дозволить підвищити ефективність та якість роботи підприємства, поліпшити умови праці, термін експлуатації ОЗ. ІС є глобальною і вирішує задачі, що пов'язані з контролем стану підприємства, його моніторингом, а також обробкою та улагодженням проблем, що існують на виробництві. Система повинна включати базу даних, програмне забезпечення працівника, що надсилає заявки до системи та

програмне забезпечення адміністратора, а також відповідне організаційне, технічне та інші види забезпечення.

У перспективі, база даних системи може використовуватися різними підрозділами підприємства. Начальник відділу або менеджер може одержувати коректну і своєчасну інформацію про зафіксовані проблеми, стан кожного завдання, перерозподіляти завдання, що направлені до його відділу.

2.3 Наявність існуючих рішень з автоматизації моніторингу підприємства

Для динамічного розвитку сільськогосподарського підприємства та економіки в цілому необхідне створення інформаційної інфраструктури як на окремому підприємстві, так і в державі в цілому. Особливу роль при підвищенні рівня автоматизації та інформатизації виробничих і управлінських процесів на підприємстві відіграють інформаційні системи. Широко використовуючись під час прийняття раціональних рішень в управлінні, проведенні інженерно-технічних, економіко-статистичних, математичних розрахунків, плануванні і контролі господарської діяльності, формуванні банків даних у галузі селекції, сучасних засобів хімізації, захисту рослин і забезпечення відповідного технічного стану складних машин і устаткування, використання вторинних матеріальних ресурсів. Інформаційні системи дозволяють мінімізувати витрати й підвищити оперативність керування підприємством у цілому.

ІС класифікуються за: масштабом, сферою застосування, способом організації, рівнями управління. Для розуміння автоматизації основних спеціалізованих завдань агронома, наведемо загальну класифікацію за рівнями управління. Залежно від цього обслуговування виробничих процесів на підприємстві може бути віднесено до різних класів:

Клас А: системи (підсистеми) керування технологічними об'єктами й/або процесами.

Клас В: системи (підсистеми) підготовки й обліку виробничої діяльності підприємства.

Клас С: системи (підсистеми) планування й аналізу виробничої діяльності підприємства.

Виокремимо основні напрями використання ІС в сільському господарстві:

1. Засоби обробки великих масивів неструктурованої інформації забезпечують доступ та обробку інформації, яка, як правило, надходить із макрооточення підприємства, дозволяючи здійснювати доступ до віддалених баз даних, інформаційно-довідкових і пошукових систем. До таких можна віднести довідкові системи типу "Агроном", геоінформаційні системи, системи обробки даних дистанційного зондування тощо.

2. Засоби автоматизації процесів підприємства дозволяють його працівникам виконувати роботу більш якісно і ефективно. До них належать засоби обробки великих масивів структурованих даних (бази даних, електронні архіви), системи управління машинами, системи фітомінторингу тощо.

3. Засоби автоматизації праці агрономів в якості управлінців, менеджерів дають можливість використовувати напрацьований світовий досвід, закладений в інформаційні продукти для управління підприємством. До них віднесемо: системи автоматизованої обробки статистичних та економічних даних, роботу в режимі реального часу з великим обсягом різнопланової інформації, використання компонентів, пов'язаних із плануванням і аналізом, наявність незапланованих зовнішніх впливів тощо.

4. Автоматизовані системи наукових (агрономічних) досліджень - програмно-апаратні комплекси, призначені для наукових досліджень, польових випробувань.

5. Інтегровані інформаційні системи - це забезпечення автоматизації більшості функцій підприємства. Вони можуть включати в себе підсистеми (модулі) всіх вищезначених систем.

У країнах світу, зокрема у США, Канаді, Австралії, країнах ЄС, для інформаційного забезпечення АПК широко використовують наступну класифікацію:

- системи моніторингу стану агроресурсів і прогнозування врожайності сільськогосподарських культур;

- системи забезпечення контролю якості сільськогосподарської продукції;
- системи оперативного управління й оптимізації продукційних процесів;
- інформаційно-довідкові системи маркетингової спрямованості;
- аналітичні та моделюючі системи відстеження розвитку надзвичайних ситуацій, їх впливу на виробництво та якість сільськогосподарської продукції;
- спеціалізовані інформаційні системи різноманітної спрямованості та рівня деталізації;
- засоби математичного моделювання, кореляційно-регресійне моделювання, імітаційне моделювання, створення оптимізаційних моделей.

Опрацювання даних (інформаційних ресурсів) здійснюється на основі побудованої інфраструктури інформаційних технологій та описаних математичних моделей. Результатом роботи ІС можна вважати отримання інформаційного продукту, що відбиває стан, інформаційну модель визначеної предметної.

2.4 Розробка концепції ІС

2.4.1 Опис результатів вивчення об'єкта автоматизації

ІС розробляється як централізований засіб моніторингу стану агротехнічного підприємства, оперативного управління завданнями з вирішення ускладнень, що виникають. Для створення ІС необхідна розробка бази даних інтерфейсу користувача та адміністратора(диспетчера), створення та отримання завдання його перенаправлення та отримання інформації про нього, також створення статистичного аналізу.

2.4.2 Аналіз вимог до ІС

Крім функціональних вимог до ІС (облік наявних проблем на підприємстві, організація роботи з завданнями, перегляд історії, тощо), необхідно врахувати вимоги до системи в цілому:

- Простий і зручний інтерфейс користувача.

- Доступ до системи через Internet з будь-якого комп'ютера глобальної мережі.
- Висока продуктивність.
- Надійність.
- Конфіденційність інформації.

Аналізуючи зазначені вище вимоги, можна зробити висновок, що інформаційна система повинна мати/підтримувати:

- Web-інтерфейс.
- Багаторівневу архітектуру.
- Сучасні стандарти в області інформаційних технологій (протоколи передачі даних, стандарти збереження даних і т.д.).
- Сучасні серверні платформи і програмне забезпечення.
- Сучасні рішення в області інформаційних технологій по розробці розподілених систем.
- Ефективні методи доступу до даних.
- Забезпечити простоту модернізації, можливість повторного використання програмного забезпечення (коду).

2.4.3 Вибір логічної архітектури програмного забезпечення ІС

Логічна архітектура програмного забезпечення ІС являє собою концептуальний опис структури програмного забезпечення системи. Кожна система має три рівні: рівень користувача, прикладний і рівень даних. Крім того, в усіх системах мається презентаційне програмне забезпечення (ПЗ), ПЗ обробки бізнес-правил і даних, а також ПЗ, що відповідає за збереження інформації. Відрізняються ж архітектури тільки організацією програмного забезпечення (програмного коду), індивідуальної для конкретної системи.

Багато сучасних інформаційних систем побудовані на базі дворівневої архітектури "клієнт/сервер". При цьому клієнтське програмне забезпечення відповідає за обробку і відображення даних. Самі ж дані централізовано зберігаються на серверах, до яких підключаються клієнти. Часто час використання системи обмежується тривалістю такого з'єднання.

Клієнт-серверні системи добре працюють тільки в контрольованих середовищах, коли число користувачів передбачуване (це необхідно для виділення ресурсів). Однак ця архітектура стає неефективною, при великому числі користувачів або якщо їхнє число невідомо. Крім цього, масштабуємість таких систем невисока, тому що кількість безпосередніх з'єднань із сервером обмежена. Невеликі і можливості повторного застосування, оскільки користувачі прив'язані до визначеного формату бази даних. А тому що клієнтське програмне забезпечення містить блок обробки інформації, загальний обсяг програмного забезпечення досить великий (Такий тип клієнтських програм іноді називають "товстим" клієнтом). У цьому випадку при зміні алгоритмів обробки даних приходиться встановлювати нове програмне забезпечення на кожен клієнтський комп'ютер.

Можна отримати невелике поліпшення шляхом переміщення прикладних алгоритмів і блоків обробки даних на сервери даних (наприклад, за допомогою збережених процедур сервера баз даних). Таку архітектуру іноді називають 2,5-рівневою. Масштабуємість подібних систем трохи краща, але все рівно мала для виконання вимог інформаційної системи. Крім того, можливість повторного використання програмного забезпечення (коду) залишається на невисокому рівні.

Масштабуємість і ступінь повторного використання можна помітно поліпшити, додавши в архітектуру системи третій рівень. У такій багаторівневій архітектурі (її також називають N-рівневою) усі рівні – рівень користувача, прикладний і рівень даних – логічно розділені (рисунк 2.3). Багаторівнева архітектура задовольняє вимогам до інформаційної системи, що розробляється.

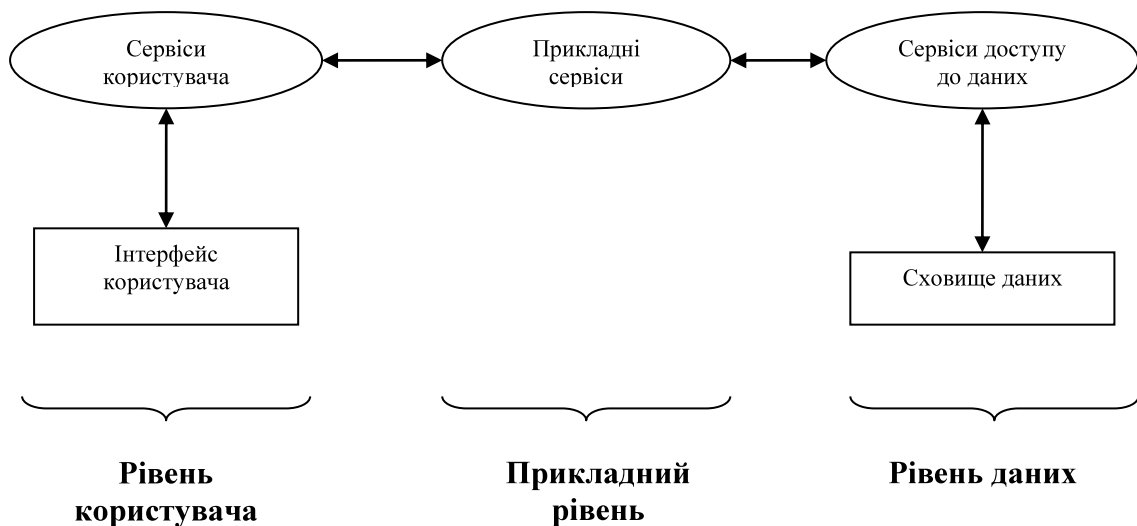


Рисунок 2.3 - Логічна архітектура програмного забезпечення ІС

Опишемо функції кожного рівня:

- Рівень користувача: відображає дані і дозволяє користувачу редагувати їх. Існує два основних типи інтерфейсу: віконний (реалізується засобами підсистеми інтерфейсу користувача операційної системи) і на основі Web. Web-інтерфейси засновані на HTML, DHTML, XHTML чи XML, у результаті чого вони можуть відображатися будь-яким Web-оглядачем на будь-якій платформі.
- Прикладний рівень: тут реалізовані бізнес-правила й обмеження на дані (перевірка коректності даних). І хоча його сервіси використовуються презентаційним рівнем, він не прив'язаний до якого-небудь клієнта – сервіси прикладного рівня доступні будь-якому клієнту. Бізнес-правила виражаються у формі прикладних алгоритмів, корпоративних правил і т.д. Обмеження на дані гарантують точність і цілісність інформації, що зберігається. Бізнес-правила звичайно реалізуються окремим модулем на централізованому сервері, що дає можливість доступу до нього відразу декільком клієнтам.
- Рівень даних: прикладний рівень не має інформації про те, як і де зберігаються дані, які він обробляє. У цьому питанні він покладається на сервіси доступу до даних, що виконують всю роботу з отримання і передачі даних. Сервіси доступу до даних також реалізуються у виді ізольованих модулів, які

знають про місце збереження інформації. Кожен модуль доступу до даних, як правило, відповідає і за цілісність сховища (наприклад, реляційної бази даних). Для багаторівневих програм у якості сховища інформації підходять сервера баз даних (MS SQL Server, Oracle, Sybase, MySQL, PostgreSQL і ін.), необхідні для обслуговування даних у таблицях і високопродуктивної виборки інформації.

2.4.4 Вибір технологій для розробки ІС

Відповідно до обраної фізичної архітектури для ІС моніторингу стану підприємства, система є розподіленою і має чотири ланки: сервер баз даних, сервер доступу до даних, Web-сервер і клієнт.

Існують наступні технології для створення розподілених систем:

- DCOM/COM+, CORBA (об'єктно-орієнтовані технології).
- CGI, ISAPI/NSAPI, Servlets і інші.

Всі ці технології тим чи іншим способом організують виклик віддалених процедур (методів) чи його аналог.

Взаємодію між клієнтами і Web-сервером раціонально організувати за допомогою CGI, ISAPI/NSAPI і/або Servlets, тому що ці технології орієнтовані на Web-інтерфейс. Використання DCOM/COM+, CORBA чи RMI у цьому випадку серйозно збільшило б вимоги до програмного забезпечення клієнтських машин, що суперечить вимогам до ІС (доступ до системи через Internet з будь-якого комп'ютера глобальної мережі). Усі зазначені технології (CGI, ISAPI/NSAPI, Servlets) вирішують однакові задачі і мають приблизно однакові характеристики по продуктивності. Servlets є найбільш гнучкою та одною з найсучасніших технологій, а також реалізується на мові високого рівня Java яка не залежить від операційної системи тому і обираємо цю технологію.

Для взаємодії серверів та доступа до БД потрібно застосовувати більш розвинуті технології.

Найбільш перспективною й універсальною технологією є CORBA, але вона теж має недоліки. CORBA – технологія досить молода і перебуває в стадії доведень і ліквідації "дитячих хвороб". Вона "сира" просто в силу свого надзвичайно швидкого розвитку. Не встигають розробники виправити помилки в поточній версії, потрібно створювати нову внаслідок появи нових

можливостей. Також, гідним образом реалізовані не всі стандартні сервіси CORBA. Навіть найкраща специфікація не врятує положення при відсутності її доступної реалізації. Крім того, за CORBA (на відміну від того ж COM) треба платити.

Основний недолік DCOM/COM+ пов'язаний з необхідністю використовувати Windows-платформу. DCOM/COM+ на практиці довела свою придатність для розробки інформаційних систем. Це безкоштовна технологія і на ній засновано багато інших технологій (наприклад, OLE DB – найефективніший /найпродуктивніший доступ до будь-яких даних). Ця технологія цілком задовольняє вимогам до інформаційної системи, перевірена на практиці, є безкоштовною, тому раціонально використовувати її для взаємодії серверів. Так як ми обрали Servlets то логічно обрати технологію яка реалізується на Java тому наш вибір це EJB, крім цього EJB має багато переваг серед своїх конкурентів.

Як говорилося раніше, на рівні користувача необхідно використовувати Web-інтерфейс (HTML/DHTML, CSS, JavaScript, DOM, Java-аплети). Клієнтське програмне забезпечення – Web-оглядач.

2.4.5 Порівняльний аналіз можливих концепцій системи

Виходячи з вимог до системи, можна розробити концепцію системи.

Основними компонентами системи є:

- серверна платформа і серверне програмне забезпечення;
- клієнтська платформа і клієнтське програмне забезпечення;
- система керування базами даних;
- мережне середовище;
- засоби розробки.

Від вибору перерахованих компонентів багато в чому залежить ефективність системи. Очевидно, що для рішення подібної задачі може існувати кілька варіантів концепції системи.

Таким чином, ґрунтуючись на ступені поширеності і необхідних функціональних можливостей, визначаємо варіанти використовуваної операційної системи (табл. 2.2).

Таблиця 2.2 - Можливі варіанти впровадження

Варі- ант	Серверна платформа	СУБД	Засоби розробки	Клієнтська платформа
1	Linux Mint 17, WEB server NGINX	Firebird 2.5	Java JDK 7	Android 4.x
2	Windows 2008 Server WEB server NGINX	MS SQL	Visual C++	
3	Windows 2008 Server WEB server NGINX	Firebird 2.5	Borland Delphi 7.0.	

Окрім цього, кожен варіант концепції системи має відповідне технічне забезпечення. Вимоги до швидкодії центральних процесорів, оперативної пам'яті, дискових підсистем серверної та клієнтських ПЕОМ визначаються вимогами програмних засобів по кожному з варіантів і приймається однаковими для усіх варіантів, оскільки обране інформаційне та програмне забезпечення має приблизно однакові вимоги до апаратних.

Щодо мережного середовища сучасним рішенням є використання крученої пари 5 категорії яка через простіший комутатор (так званий Міпі-8ЎІсп) пов'язує усі клієнтські місця, забезпечуючи номінальну пропускну здатність 100 Мб/с.

Отже технічні засоби повинні відповідати наступним вимогам:

- серверна частина з параметрами, що рекомендуються:

CPU Intel Core i5 2 ГГц; RAM 4 - 16 Гб, HDD 2 Тб, мережева карта Ethernet 100 Мб/с, джерело неперервного живлення; маніпулятор „миша”; клавіатура; монітор.

- клієнтська частина з параметрами, що рекомендуються:

CPU Intel Core i3 1,7 ГГц, RAM 2 - 4 Гб, HDD 1 Тб, мережева карта Ethernet 100 Мб/с; маніпулятор „миша”; клавіатура; монітор.

- локальна мережа: комутатор, мережне середовище - кручена пара 5 категорії.

Таблиця 2.3 - Експертні оцінки властивостей системи

Властивість системи, F_i	Варіант	Оцінка властивості кожним експертом, 0...10 балів					
		1	2	3	4	5	B_i
Продуктивність	1	8	9	8	8	8	8,2
	2	9	8	8	9	8	8,4
	3	8	9	8	8	8	8,2
Надійність	1	6	8	6	8	9	7,4
	2	6	8	7	8	7	7,2
	3	9	6	6	8	6	7,0
Захищеність	1	8	8	6	7	8	7,4
	2	7	7	7	8	7	7,2
	3	8	8	7	9	6	7,6
Легкість підтримки	1	7	7	7	6	8	7,0
	2	6	8	8	7	7	7,2
	3	7	8	7	8	7	7,4
Документованість	1	9	8	7	8	9	8,2
	2	8	9	9	7	7	8,0
	3	7	8	9	7	8	7,8

Тепер необхідно провести аналіз переваг і недоліків запропонованих варіантів концепцій і оцінити кожний окремо, з погляду досягнення поставлених перед системою цілей, задоволення висунутим критеріям і обмеженням.

Для вибору найбільш ефективного варіанта концепції системи використовується метод функціонально-вартісного аналізу.

Функціонально-вартісний аналіз складається з двох основних етапів: функціональний аналіз; вартісний аналіз.

На першому етапі функціонального аналізу необхідно визначити ступінь відповідності кожного варіанта концепції системи основним характеристикам даної системи.

Спочатку необхідно оцінити основні параметри системи по кожному з варіантів. Для цього експертами даються експертні оцінки кожного параметра системи для кожного варіанта (таблиця 2.3).

На другому етапі функціонального аналізу потрібно визначити вагові коефіцієнти для кожного з параметрів. У даному випадку вони також вибираються експертним шляхом. Значення вагових коефіцієнтів для кожного з параметрів приведені в таблиці 2.4.

Таблиця 2.4 - Вагові коефіцієнти для параметрів системи

Параметр, F_i	Коефіцієнт, ϕ_i
Надійність технічних засобів	0,45
Надійність програмних засобів	0,45
Захищеність даних	0,50
Продуктивність технічних засобів	0,20
Продуктивність програмних засобів	0,10
Документованість	0,05

Після визначення вагових коефіцієнтів необхідно визначити коефіцієнти технічного рівня для кожного з параметрів по кожному варіанті по формулі

$$K_{TY}[F_i] = B_i \phi_i$$

Результати розрахунку коефіцієнтів технічного рівня приведені в таблиці 2.5.

Таблиця 2.5 - Розрахунок показників технічного рівня

Властивість, F_i	Варіант	Оцінка параметра в балах, B_i	Ваговий коефіцієнт, ϕ_i	Показник технічного рівня. $K_{TP}[F_i]$
Продуктивність	1	8,2	0,3	2,46
	2	8,4		2,52
	3	8,2		2,46
Надійність	1	7,4	0,3	2,22
	2	7,2		2,16
	3	7,0		2,10
Захищеність	1	7,4	0,2	1,48
	2	7,2		1,44
	3	7,6		1,52

Легкість підтримки	1	7,0	0,15	0,7
	2	7,2		0,72
	3	7,4		0,74
Документованість	1	8,0	0.05	0,8
	2	8,1		0,81
	3	8,0		0,8

За даними таблиці 2.5 розраховується коефіцієнт технічного рівня K_{TY} для кожного з варіантів по формулі

$$K_{TY}[F_i] = \sum_i K_{TY}[F_i]$$

Результати розрахунку приведені в таблиці 2.6.

Таблиця 2.6 - Коефіцієнти технічного рівня

Варіант	Коефіцієнт технічного рівня
1	14
2	7.06
3	6.78

Найкращим на етапі функціонального аналізу є варіант, що має максимальне значення коефіцієнта технічного рівня K_{TY} , тобто варіант 1.

Аналіз функцій, що реалізуються системою, повинний бути доповнений вартісним аналізом. Для цього розглядаються зведені витрати Z_j по кожному варіанту:

$$Z_j = C_j + E_n K_j$$

де C_j – поточні витрати по j -му варіанту;

K_j – капітальні вкладення по j -му варіанту;

E_n – нормативний коефіцієнт економічної ефективності капітальних вкладень, що дорівнює 0,15.

Зведені витрати Z_j розраховують у сфері виробництва (у розроблювача) або в сфері експлуатації (у користувача).

Доцільно розраховувати витрати в сфері виробництва. Поточні витрати при розробці C_j – це собівартість створення системи C_c ;

$$C_c = C_z + C_{OTЧ} + C_M + C_H$$

де C_z – заробітна плата розроблювачів;

$C_{OTЧ}$ – відрахування на соціальні заходи (38% від фонду оплати праці).

C_M – вартість машинного часу; необхідного для розробки і налаштування системи,

C_H – накладні витрати в розмірі 50% від витрат на оплату праці (містять оплату праці управлінському персоналові з нарахуваннями оплат службових відряджень; оплату консультативно-інформаційних послуг оплату ремонту і техобслуговування основних фондів, крім ПК; оплату оренди приміщення).

Витрати на оплату праці розробників

$$C_z = C_z T$$

де C_z – погодинна оплата труда,

T – трудомісткість розробки.

Вартість машинного часу

$$C_M = t_M * C_{M-г},$$

де t_M – тривалість машинного часу, який необхідний для розробки системи, год.;

$C_{M-г}$ – собівартість одного часу роботи машини

$$C_{M-г} = \frac{\sum_i U_i}{T_{EF}}$$

де U_i – річні експлуатаційні витрати при обслуговуванні ПК, що включають:

- зарплату фахівця, що обслуговує машину (з виправленням на коефіцієнт зайнятості працівника обслуговуванням ПК);
- нарахування на зарплату (38%);
- витрати на електроенергію, що визначаються як добуток вартості 1 кВт/год. електроенергії на потужність, споживаної ПК, і на ефективний фонд часу ПК (за рік);

- накладні втрати — 50 % від втрат на оіагу праці;

$T_{\text{ПФ}}$ – ефективний фонд часу роботи ПК за рік, що визначається виходячи з календарного річного фонду часу, крім вихідних і святкових днів,

$C_{\text{м-г}} = (300 + 300 * 0,375 + 300 * 0,5 + 328,86) / 1512 = 0,60 \text{ грн./год.}$ (для усіх варіантів);

$$C_{\text{з1}} = 920 * 4 = 3680 \text{ грн};$$

$$C_{\text{отч1}} = 3680 * 0,475 = 1748 \text{ грн};$$

$$C_{\text{м1}} = 0,35 * 700 = 245 \text{ грн}$$

$$C_{\text{н1}} = 3680 * 0,5 = 1840 \text{ грн};$$

$$C_{\text{с1}} = 3680 + 1748 + 245 + 1840 = 7513 \text{ грн.}$$

Для другого варіанта собівартість створення системи:

$$C_{\text{з2}} = 940 * 4 = 3760 \text{ грн};$$

$$C_{\text{отч2}} = 3760 * 0,475 = 1786 \text{ грн};$$

$$C_{\text{м2}} = 0,35 * 700 = 245 \text{ грн};$$

$$C_{\text{н2}} = 3760 * 0,5 = 1880 \text{ грн};$$

$$C_{\text{с2}} = 3760 + 1786 + 245 + 1880 = 7671 \text{ грн.}$$

Для третього варіанта собівартість створення системи:

$$C_{\text{з3}} = 980 * 4 = 3920 \text{ грн};$$

$$C_{\text{отч3}} = 3920 * 0,475 = 1862 \text{ грн};$$

$$C_{\text{м3}} = 0,35 * 700 = 245 \text{ грн}$$

$$C_{\text{н3}} = 3920 * 0,5 = 1960 \text{ грн};$$

$$C_{\text{с3}} = 3920 + 1862 + 245 + 1960 = 7987 \text{ грн}$$

Капітальні вкладення K_j мають одноразовий характер. Віднесемо до них вартість придбання ліцензованого програмного забезпечення, яке необхідне для створення та проектування, вартість експлуатації системи, вартість технічного забезпечення, а також вартість впровадження системи. Вартість необхідного програмного забезпечення наведена в таблиці 2.8.

Таблиця 2.7 - Загальні витрати на впровадження по варіантах

Варіант	Загальні витрати, грн.
1	900
2	1700
3	2000

Таблиця 2.8 - Вартість програмного забезпечення

Вар.	Найменування	Вартість, грн.	Кількість, од.	Разом, грн.	Разом за варіантом, грн.
1	Linux Mint 17	20	1	20	5020
	Firebird 2.5	0	1	0	
	Android with libraries	1000	5	5000	
	Java	0	1	0	
2	Windows 2008 Server	5000	1	5000	24000
	MS SQL	9000	1	9000	
	Android with libraries	1000	5	5000	
	Visual C++	5000	1	5000	
3	Windows 2008 Server	5000	1	5000	21000
	Firebird 2.5	0	1	0	
	Android with libraries	1000	5	5000	
	Borland Delphi 7.0.	10000	1	10000	

Вартість технічного забезпечення клієнтських робочих станцій складає 3500 грн. для всіх варіантів.

Вартість нового серверного забезпечення дорівнює 4000 грн. (за даними прайс-ліста фірми Technomir за станом на час розробки).

Загальні витрати на впровадження підсистеми, що складаються з оплати інсталяції ПЗ та навчання персоналу по варіантах наводиться в таблиці 2.7.

Отримані значення необхідних капітальних вкладень за кожним з варіантів наведені в таблиці 2.9.

Таблиця 2.9 - Капітальні вкладення по варіантах

Варіант	Розмір капітальних вкладень, грн.
1	13420
2	31700
3	29000

Знаючи розміри капітальних вкладень, можна обчислити зведені затрати. Відповідні результати розрахунків для всіх варіантів наведені в таблиці 2.10.

Таблиця 2.10 - Зведені витрати по варіантах

Варіант	Зведені витрати, грн..
1	1980
2	39371
3	36987

Узагальнені показники ефективності варіантів систем визначаються через коефіцієнт техніко-економічного рівня по кожному з варіантів за формулою

$$K_{TEU} = \frac{K_{Tij}}{Z_j}$$

Результати розрахунків представлені в таблиці 2.11.

Таблиця 2.11 — Коефіцієнти техніко-економічного рівня по варіантах

Варіант	Коефіцієнт техніко-економічного рівня
1	4,32*10-4
2	2,18*10-4
3	1,27*10-4

Найкращий варіант визначається за максимальним значенням коефіцієнта K_{TEU} . Таким чином, визначаємо, що найкращий варіант - це варіант 1.

Опис запропонованого варіанту концепції.

Для вибору найкращого варіанта було виконано функціонально-вартісний аналіз. Отримання для кожного варіанта коефіцієнтів техніко-економічного рівня дозволило порівняти варіанти між собою. Найкращий варіант визначається за максимальним значенням коефіцієнта K_{TEU} , тому обираємо варіант 1.

Таким чином, програмними компонентами оцінки кредитоспроможності позичальника будуть:

- СУБД: Firebird 2.5;
- серверна платформа: Linux Mint 17, веб сервер NGINX;

- засоби розробки: Java ;
- клієнтська платформа: Android, 1С

2.5 Постановка задачі

На основі аналізу структури підприємства, моделей управління проектами, методик контролю за виконанням завдань та з метою розробки автоматизованої інформаційної системи моніторингу стану агротехнічного підприємства, визначимо основну мету даного диплому.

Для досягнення мети необхідно створити інформаційну систему яка повинна забезпечувати:

- Аналіз введених даних про проблеми, що виникають у процесі функціонування підприємства, шляхом порівняння з експертними оцінками з бази даних а також власною базою історій.
- На основі поданої інформації про проблему та іншої інформації з урахуванням ризику автоматично розраховувала час вирішення проблеми.
- На основі експертних оцінок автоматично розставляти "важливість" завдання». Рішення повинно прийматися з урахуванням інформації про проблему (технічні проблеми, погодні умови, проблеми зв'язку тощо), по яких проходить оцінка та границя на допустимий ризик.

3. РОЗРОБКА ПРОЕКТНИХ РІШЕНЬ ПО ІС

3.1 Загальносистемні проектні рішення

3.1.1 Організаційна і функціональна структура

У процесі свого функціонування ІС взаємодіє з користувачами з різних підрозділів для виконання ними різних функцій. Схема цієї взаємодії, побудована в нотації варіантів використання ІС (use case) показана на рисунку 3.1.

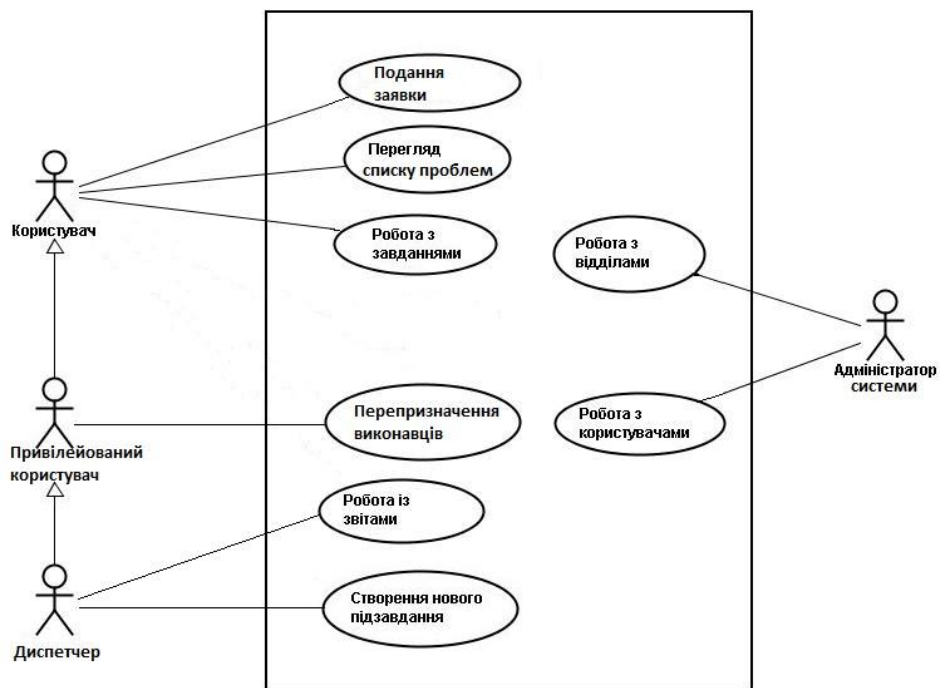


Рисунок 3.1. Схема організаційно-функціональної структури

У ході своєї роботи, працівники підрозділів використовують наступні функції.

Користувач – перегляд наявних проблем та їх станів, створення нових заявок, перегляд минулих проблем, перегляд статистики.

Якщо у користувача є підлеглі, тобто він є *привілейованим користувачем* тоді він може переглянути їхні завдання та статистику, пере направити завдання, що поступають до нього на них.

Диспетчер може здійснювати як функції користувача та привілейованого користувача, так і моніторинг системи, обробку заявок, що надходять до

системи, формування звітів, контроль за виконанням завдань.

Адміністратор системи виконує функції адміністрування та підтримки роботи системи.

3.1.2 Опис функцій, що автоматизуються

Задачею ІС, що створюється, є автоматизація наступних функцій:

- Обробка інформації про стан підприємства та наявні на підприємстві ускладнення.
- Облік робіт по улагодженню проблем та інформації про статус завдань.
- Формування звітів.
- Усі інші вимоги не є головними і можуть змінюватися під час розробки або експлуатації.

Для створення ІС необхідно розробити єдину базу даних для збереження інформації про відділи та їх співробітників, про отриманні ними завдання та інше, створити базу даних, до якої будуть добавляться відомості про проблеми та їхні статуси. Також повинно бути передбачене програмне забезпечення для адміністрування системи, надсилання до системи заявок. Крім того, необхідно внести кадрові зміни та найняти людину на посаду диспетчера відділу.

3.1.3 Опис постановки задачі

На підставі аналізу вимог і обмежень, а також попередніх концепцій реалізації ІС системи моніторингу проблем, можна сформулювати опис постановки задачі. Необхідно розробити ІС моніторингу стану підприємства, що буде задовольняти перерахованим вище функціональним вимогам. Необхідно також забезпечити надійність збереження і безпеку доступу до інформації ІС. Вона повинна обслуговувати одночасну відправку заявок багатьма користувачами.

3.1.4 Загальний опис ІС

ІС є інструментом управління процесом обробки інформації про стан підприємства, заявок щодо існуючих проблем на агротехнічному підприємстві. Основними функціональними характеристиками ІС є:

- ведення загальної інформаційної бази з розподіленим доступом;
- авторизований доступ до даних на основі прав доступу;

- відображення змін, внесених з будь-якого завдання;
- можливість створення і редагування завдань, оперативного контролю ходу їхнього виконання;

ІС має звичний графічний інтерфейс користувача, що знижує витрати на навчання персоналу, і побудована з максимальним використанням існуючого на підприємстві в даний момент устаткування і програмного забезпечення.

3.1.5 Захист від несанкціонованого доступу

Захист ІС від несанкціонованого доступу реалізовано на системному рівні. При запуску системи відбувається авторизація користувача. Доступ до даних і функцій дозволяється тільки при правильному введенні користувальницького імені і пароля.

3.2 Рішення по організаційному забезпеченню

3.2.1 Організаційна структура підприємства

У ході впровадження і використання ІС, його функціями охоплюються всі суб'єкти підприємства. Схема організаційної структури підприємства при взаємодії із ІС (у виді діаграми варіантів використання) приведена на рисунку 3.2.

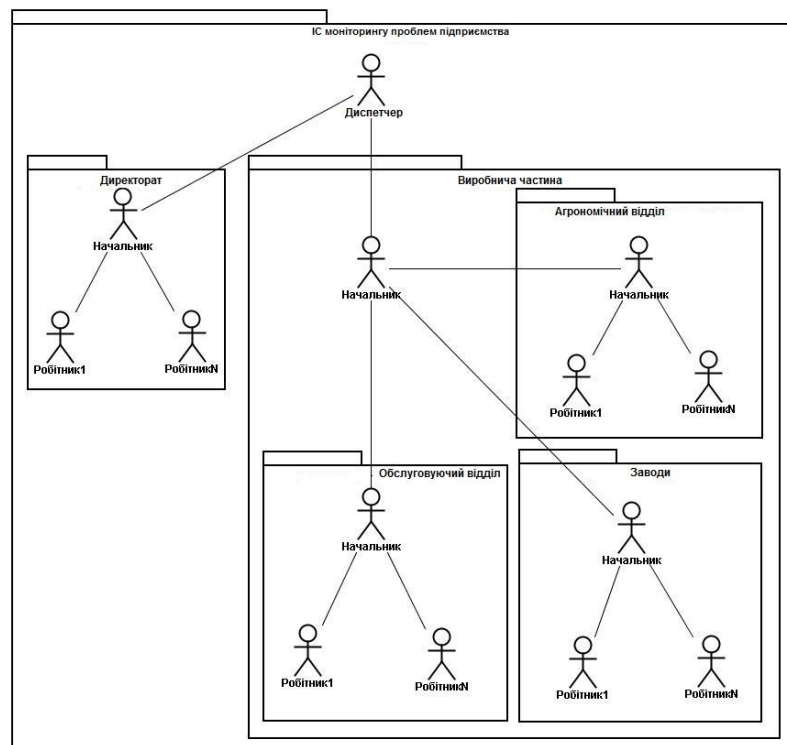


Рисунок 3.2.- Суб'єкти підприємства, які взаємодіють із системою

3.2.2 Інструкція користувача

Для нормального функціонування ІС, всі операції, проведені в ньому користувачами, повинні відповідати Інструкції користувача, приведеної в Додатку Г.

3.2.3 Заходи щодо організаційного забезпечення

Ефективна робота ІС моніторингу стану агротехнічного підприємства вимагає організаційних заходів, цілями яких є підтримка даних про процеси, що відбуваються, в актуальному стані і грамотній своєчасній реакції користувачів на інформацію, що їй надається системою:

- навчання користувачів роботі в системі згідно виконуваних функцій (посадових обов'язків);
- досвідчена експлуатація ІС з імітацією керування будівництвом об'єкта, що був побудований раніше з максимальним наближенням до реальності (у виді ділової гри);
- мінімізація часу, що проходить між зміною версій документів і зміни їхнього стану (іншими словами, негайна реакція керівництва на зміни у виробничому процесі);
- обов'язкова участь у формуванні даних усіх суб'єктів підприємства; регулярний оперативний контроль зв'язаних процесів керівниками підрозділів і контроль виробничих циклів.

3.2.4 Організаційно-кадрові заходи

Для підтримки апаратних і програмних засобів у робочому стані, а також рішення поточних задач налаштування, установки і відновлення загальносистемного програмного й апаратного забезпечення на підприємстві повинна бути введена штатна одиниця системного адміністратора, або для цих цілей застосовуватися аутсорсінг (outsourcing – надання ІТ-функцій сторонніми організаціями).

Для здійснення управління розроблювальним ІС повинна бути введена штатна одиниця диспетчера системи моніторингу стану підприємства.

3.3 Рішення по інформаційному забезпеченню

3.3.1 Опис інформаційного забезпечення ІС

Одну з основ інформаційного забезпечення ІС складає визначення її внутрішнього стану, що буде відбито в моделі класів. На даному етапі нас цікавлять довгоживущі і постійні сутності бізнес-процесів. Цей вид класів будемо називати класами-сутностями (чи класами предметної області).

Для створення моделі класів складемо словник даних, ґрунтуючись на сутностях, виявлених на етапі бізнесу-аналізу.

Користувач – людина яка взаємодіє з системою

Диспетчер – користувач, на якого є хоча б одне посилання у іншого користувача

Звіт - звіт формується користувачем, та відсилається безпосередньому начальнику

Завдання – завдання сформоване одним користувачем до іншого

Дія – зміна статусу завдання (створене, відхилене, то що) проходить через створення користувачем "дії".

На даному етапі проектування сутності словника даних стають кандидатами для класів проектованої системи. Модель бізнес-сутностей ІС показана на рисунку 3.3.

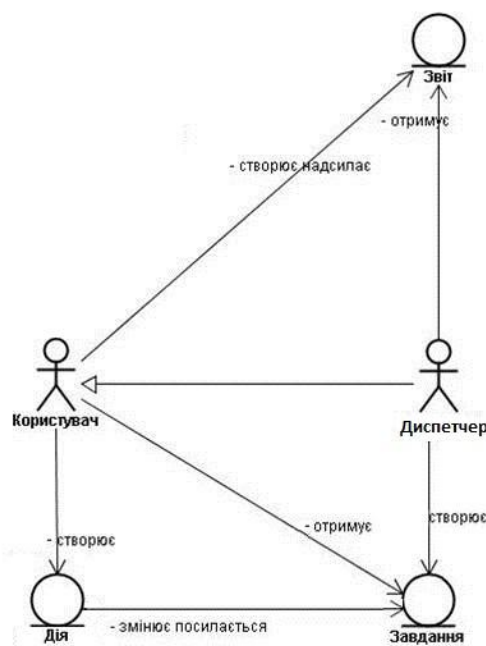


Рисунок 3.3. - Діаграма бізнес-сутностей системи

Для подальшої деталізації сутностей і зв'язків між ними використовуємо діаграму "сутність-зв'язок" (ERD), що дозволить найбільше повно представити концептуальну модель даних, а також перейти в наслідку до їх логічного представлення. Побудована на підставі аналізу бізнес-сутностей системи модель у термінах "сутність-зв'язок" у нотації UML приведена на рисунку 3.4.

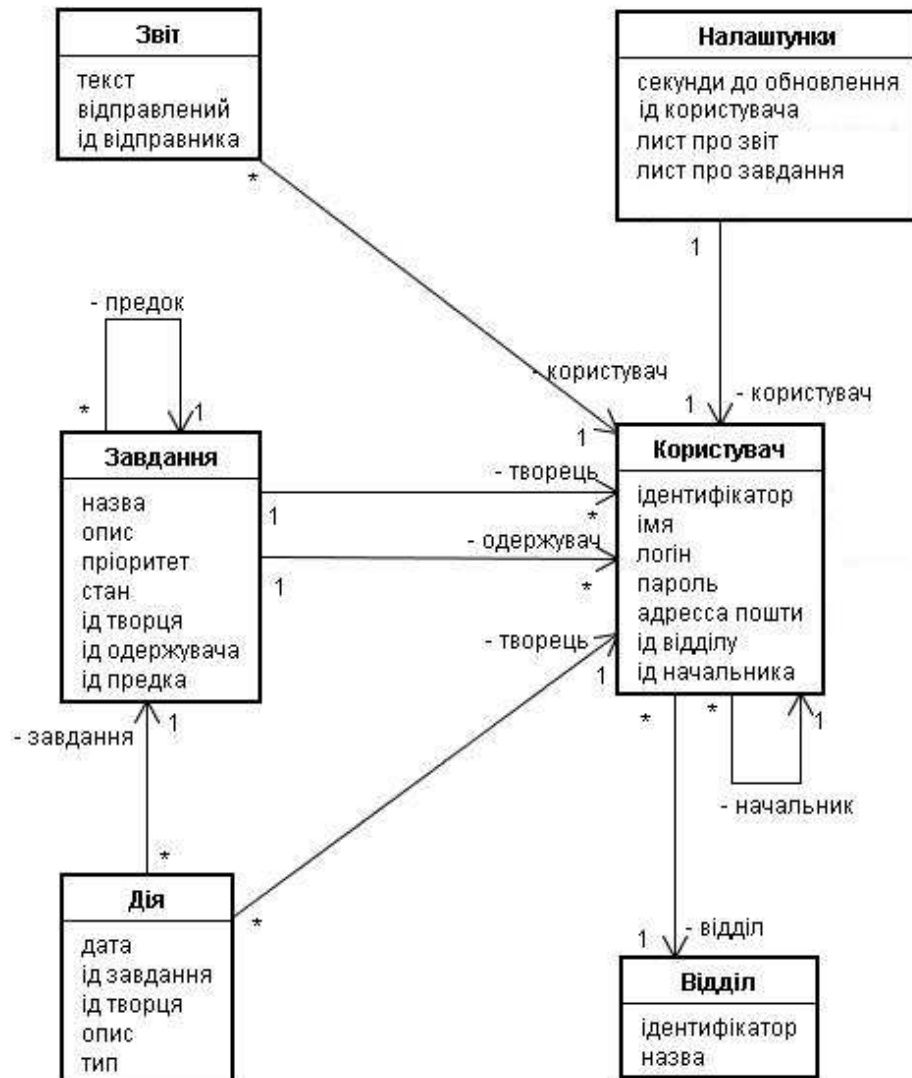


Рисунок 3.4. - Інформаційна модель об'єкту управління

3.4 Рішення по технічному забезпеченню

3.4.1 Схема автоматизації

Рациональний вибір технічних засобів, необхідних для впровадження ІС буде значною мірою визначати загальний чи успіх невдачу проекту. На підставі прийнятих загальносистемних рішень були визначені склад і організація комплексу технічних засобів.

Вимоги до продуктивності:

Обчислювальний комплекс повинний забезпечувати достатню продуктивність для рішення передбачуваних задач при бажаній мінімізації фінансових витрат на його придбання. Крім того, потрібно забезпечити сприятний час реакції системи при роботі в інтерактивних режимах, що складають значну частину від всіх експлуатаційних режимів. Цим буде забезпечена комфортна робота користувачів. Проектована система не буде пред'являти підвищених вимог до обчислювальної потужності. Створювану систему можна віднести до класу облікових задач, у яких загальна продуктивність системи визначається швидкістю операцій уведення-висновку при неповному залученні потужності центрального процесора. У такий спосіб будь-яке сучасне апаратне рішення з пропонованих на ринку буде відповідати вимогам до продуктивності. Розподіл програмних компонентів майбутньої системи дозволить не збільшити навантаження на існуючі клієнтські робочі місця і не висувати вимог до їх модернізації.

Вимоги до надійності:

Розв'язувана задача висуває стандартні вимоги до надійності апаратних засобів без вимоги забезпечення повної відмовостійкості. Основною вимогою в даній групі буде вимога до схоронності даних. Передбачаються наступні заходи для його виконання:

- включення сервера в електромережу через пристрій безперебійного живлення (UPS);
- використання на сервері БД апаратного IDE RAID контролера з можливістю віддзеркалення дисків (дане рішення має досить прийнятну вартість);
- використання спеціального пристрою резервного копіювання; як такий пристрій можливе застосування CD-RW привода.

Вимоги до експлуатаційних характеристик:

З перерахованих вимог можна зробити висновок про застосовність для сервера БД і клієнтських робочих місць стандартного апаратного забезпечення

на платформі x86. До сервера БД пред'являється традиційний набір вимог для такого рішення.

3.5 Опис алгоритмів функціонування системи

Для уточнення алгоритмів функціонування розроблювальної системи використовуємо засобу об'єктно-орієнтованого аналізу й опису поведінки об'єктів. При цьому будемо розглядати як поведінку системи в цілому (для уточнення виду, характеру і порядку взаємодії між класами), так і реалізацію функціональності окремих класів системи. Діаграма класів системи, побудована на підставі аналізу сутностей і операцій над ними представлена на рисунку 3.5.

Укажемо відповідальність кожного з приведених на діаграмі класів:

User (Користувач) – людина яка взаємодіє з системою.

Settings (Начальник) – користувач, на якого є хоча б одне посилання у іншого користувача.

Report (Звіт) – звіт формується користувачем, та відсилається безпосередньому начальнику.

Note (Повідомлення) – користувачі можуть обмінюватися повідомленнями.

Task (Завдання) – завдання на виконання роботи зформоване одним користувачем до іншого.

Action (Дія) – зміна статусу завдання (створення, відхилення, то що) проходить через створення користувачем "дії".

Mailer – клас який займається надсиланням листів.

MailerException – виключення екземпляр якого може виникнути при використанні класу Mailer.

CharEncodingFilter – фільтер який преобразує дані який приходять від Http запиту та відповіді.

LookupUtil – клас реалізуючий технологію JNDI для отримання екземплярів SessionBean класів у веб модулі.

AdminSession – інтерфейс описуючий дії адміністратора системи.

AdminSessionBean – клас який його реалізує.

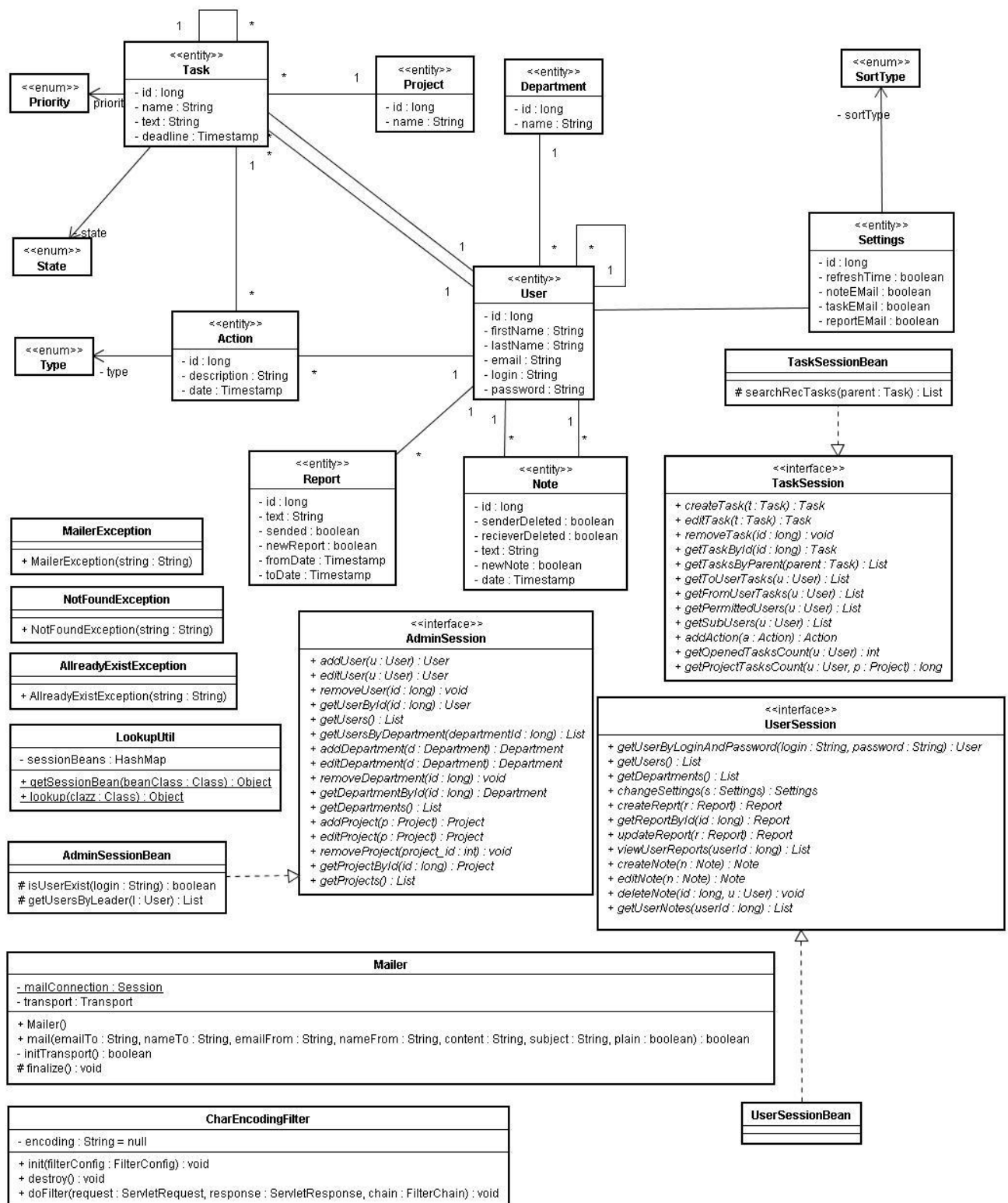


Рисунок 3.5. - Діаграма класів системи

UserSession – інтерфейс описуючий дії користувача, як то створення звіту, відправлення повідомлення тощо.

UserSessionBean – клас який його реалізує.

TaskSession – інтерфейс описуючий методи для роботи з завданнями.

TaskSessionBean – клас який його реалізує.

NotFoundException – виключення екземпляр якого створюється якщо не знайдений екземпляр у базі даних.

AlreadyExistException – виключення екземпляр якого створюється якщо не може бути створений об'єкт (користувач з тим логіном який вже використовується системою).

Розглянута вище діаграма класів являє собою логічну модель статичного представлення системи, що моделюється. Мова йде про те, що на даній діаграмі зображуються тільки взаємозв'язки структурного характеру, що не залежать від часу реакції системи на зовнішні події. На відміну від цього представлення, загальний алгоритм поведінки системи може бути описаний за допомогою діаграми станів (statechart diagram). Загальний алгоритм поведінки важливий для визначення взаємодій між компонентами системи й уточнення причин, що викликають та чи інша відповідна дія. Алгоритм поведінки системи в нотації діаграми станів приведений на рисунку 3.6.

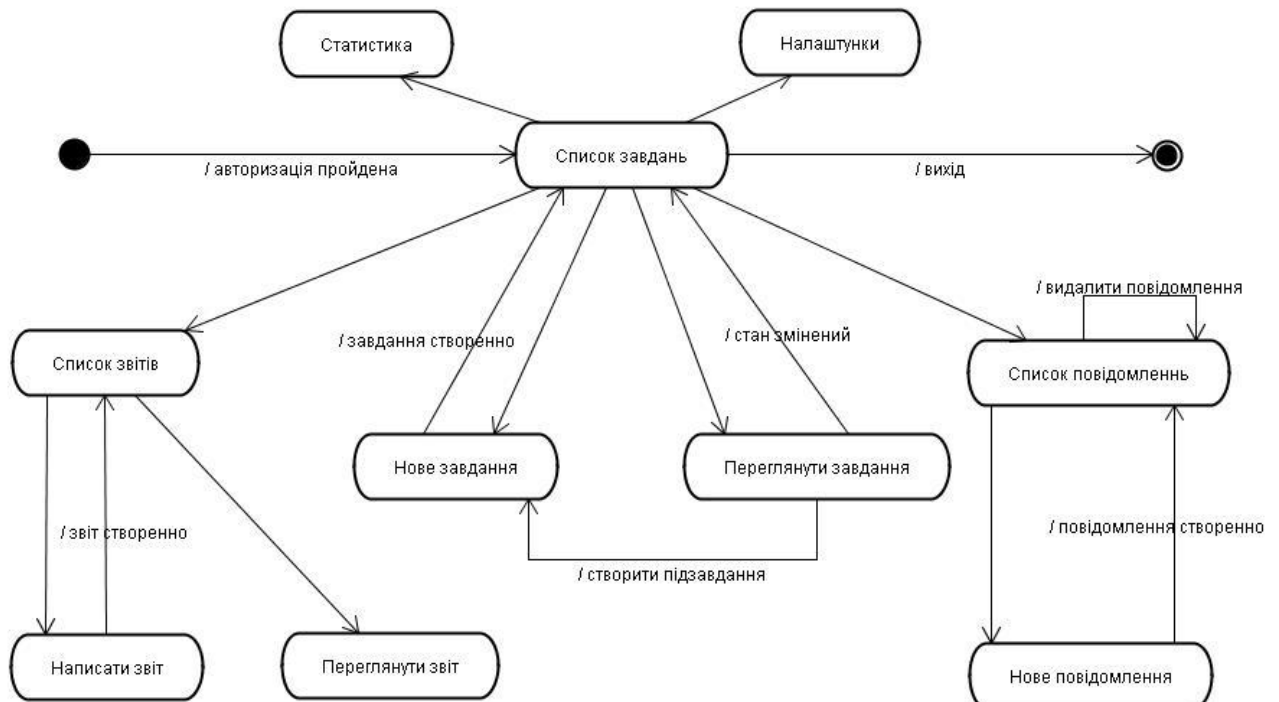


Рисунок 3.6. - Загальний алгоритм поведінки системи

При переході по посиланню на перегляд отриманого завдання, користувач може відмовитися від нього, завершити або створити нове підзавдання. Якщо ж відбувається перегляд створеного користувачем завдання тоді він може його видалити (якщо стан завдання не відмінний від начального та завдання не має "потомків" створити підзавдання або якщо завдання було відхилене або завершене він може його схвалити чи навпаки (з указівкою причин). Усі перелічені дії повертають на попередній екран.

З цього ж екрану користувач може перейти:

- на екран де може змінити свої налаштування (кількість секунд для оновлення екрану завдань, отримувати листа при отриманні завдання, звіту або повідомлення);
- на екран статистики де відображується статистика за обраний період, кількість виконаних та отриманих завдань, а також їх перелік, якщо у користувача є підлеглі то він може переглянути їхню статистику теж;
- на екран звітів, де бачить свої звіти, або якщо у користувача є підлеглі то відіслані ними звіти теж, створити новий звіт або відіслати існуючий начальству;
- на екран повідомлень, де бачить повідомлення адресовані йому, або надіслані ним, він може їх видалити, створити нове;

Користувач може вийти з системи, користувач може завершити дію веб оглядача, але у цілях безпеки рекомендується виходити з системи.

3.6 Рішення по програмному забезпеченню

3.6.1 Структура і функції частин програмного забезпечення

Функціональність і структуру проектованої ІС поряд з моделлю поводження визначить модель її станів. Будемо будувати модель станів шляхом уточнення моделі бізнесі-сутностей, побудованої нами на підставі вимог користувачів до функцій системи й оброблюваних даних. Вхідними даними для даного етапу проектування будуть моделі сутностей, а результатом стане модель специфікації проектних конструкцій розроблювального програмного забезпечення. У відповідності з прийнятим у для даної роботи підходом

об'єктно-орієнтованого аналізу і проектування виконаємо подальшу специфікацію проектних рішень у виді діаграм класів. Класи визначають стан пропонованої програмної системи (через значення своїх атрибутів) і, значною мірою, її поведінку (через операції чи методи класів).

Сформульовані користувальницькі вимоги тепер приймемо за обмеження. Поряд з моделлю сутностей для проектування класів будемо користатися побудованими раніше діаграмами прецедентів (рисунок 3.1), у відповідності до рекомендацій методики Rational Unified Process. При цьому модель класів буде відповідати словнику даних, що був отриманий на підставі аналізу прецедентів, і який є текстовим аналогом моделі сутностей.

3.6.2. Логічна модель бази даних

Для забезпечення можливості зберігати дані в реляційній базі даних, з моделі класів була отримана логічна модель даних. Для цього серед усіх класів були визначені постійні. Це такі класи, об'єкти яких повинні зберігати свої значення не тільки під час роботи програми. Далі було проведено відображення постійних класів на таблиці моделі даних. При цьому атрибути класів відобразилися в поля реляційних таблиць. Асоціації між класами змінилися в зв'язки між реляційними таблицями з використанням механізму зовнішніх ключів. Для забезпечення можливості підтримки цілісності даних по суттєвостям були використані первинні ключі. Обрана СКБД не підтримує зберігання ієрархії спадкування класів, тому при будівництві логічної моделі даних ієрархії відображалися на таблиці за принципом одна таблиця на клас. Метод згортання типів в супертип при якому ієрархія змінюється в одну таблицю був відхилений із-за великої надмірності даних. Розроблена логічна модель наведена на рисунку 3.7.

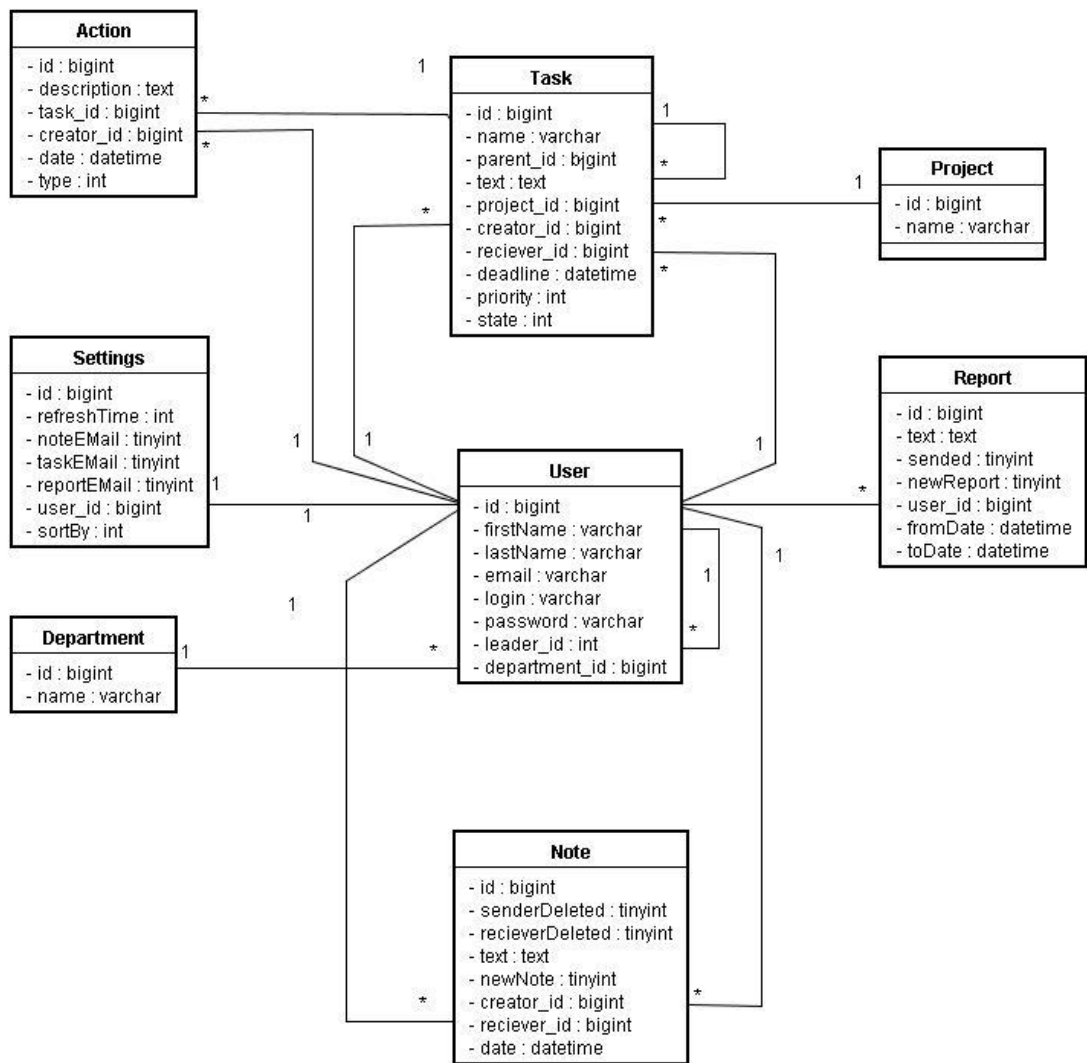


Рисунок 3.7. Логічна модель бази даних

3.6.3. Методи і засоби розробки програмного забезпечення.

При аналізі і проектуванні ІС застосовувався об'єктно-орієнтований підхід (ООП), виходячи з чого програмне забезпечення системи повинне бути розроблене з його використанням. Основними принципами ООП є спадкування, інкапсуляція і поліморфізм.

Використання ООП у розробці дозволяє:

- використовувати отримані в результаті аналізу і проектування логічні моделі для первинної кодогенерації;
- скоротити час розробки за рахунок повторного використання коду програмних компонентів;
- проводити налагодження і тестування ґрунтуючись на поводженні об'єктів, без знання їхньої внутрішньої структури;

- цілком використовувати потенціал обраних засобів розробки ПО.

Для проектування й аналізу ІС використовувалася CASE-система Rational Rose виробництва компанії Rational Software. На сьогоднішній день це один із самих могутніх CASE-засобів на ринку і де-факто стандарт галузі. Rational Rose використовується на всіх етапах проектування від аналізу до автоматичної генерації коду програмної системи і має засобу для роботи з всіма інструментами, наданими мовою UML.

Основним засобом розробки програмного забезпечення системи є мова програмування високого рівня Java та мова ІС, використовуються такі технології: Servlets, EJB3, JSPx. У якості середовища був використаний продукт Netbeans IDE v8.0.

На Web-сервері повинен бути встановлений HTTP-сервер. Найбільш розповсюдженими HTTP-сервера, що задовольняють зазначеним вимогам і мають високу продуктивність, є Apache Server. На Web-сервері повинна бути встановлена ОС яка підтримує цей сервер.

Так як використовуються технології організовані на Java то операційна система на сервері доступу до БД може бути будь яка з перелічених:

- Macintosh
- MS Windows
- Linux

Також потрібний Java WEB сервер встановлений на сервері будь який з:

- JBoss
- GlassFish
- Tomcat

На сервері баз даних повинна бути встановлена серверна система управління базами даних (СУБД).

Необхідна сучасна серверна реляційна СУБД, що має високу продуктивність, здатна працювати з великими обсягами інформації і підтримує стандарт SQL-92. Крім того, СУБД повинна працювати під управлінням Windows. Серверними СУБД, що задовольняють зазначеним вимогам, на сьогодні є:

- MS SQL Server.
- MySQL.
- PostgreSQL

На клієнтських машинах може використовуватися будь-яка операційна система з можливістю встановлення віддаленого мережного з'єднання і графічним інтерфейсом. Операційні системи, що рекомендуються – MS Windows, Linux, тощо.

Програмне забезпечення, з яким буде працювати користувач, являє собою Web-оглядач з підтримкою HTML 4.0, JavaScript 1.0, CSS 2 (наприклад, MS Internet Explorer, FireFox, Opera та інші).

4. РОЗРОБКА РОБОЧОЇ ДОКУМЕНТАЦІЇ ІС МОНІТОРИНГУ СТАНУ АГРОТЕХНІЧНОГО ПІДПРИЄМСТВА

ІС моніторингу стану аграрного підприємства призначена для автоматизації, централізації та систематизації роботи зі збору та обробки інформації про наявні проблеми підприємства на будь-якому рівні, а також автоматизації процесу їх вирішення. Функції, що автоматизуються системою, пов'язані з організацією робочого часу працівників, отримання завдань та іншого документообігу, як то звіти та статистичні показники.

Основними функціями, що автоматизуються є:

- Облік заявок про проблеми, що надходять до системи.
- Облік завдань та інформації про їхній статус.
- Формування звітів.
- Усі інші вимоги не є головними і можуть змінюватися під час розробка або експлуатації.

При впровадженні програмного забезпечення основними цілями є:

- Простий і зручний інтерфейс користувача.
- Доступ до системи через Internet з будь-якого комп'ютера глобальної мережі.
- Висока масштабованість системи.
- Висока продуктивність.
- Надійність.
- Конфіденційність інформації.

Для ефективного використання програмного забезпечення важливо впровадити на підприємстві організаційні і технічні заходи, що передбачено в п. 3.2 та п. 3.4.

Спроектована система відповідає вимогам які викладені в технічному завданні (Додаток А).

Розроблена система є розподіленою, тому треба правильно виконувати її розгортання та налаштування компонентів. Оскільки робота присвячена розробці ІС моніторингу стану підприємства, то особливої уваги заслуговує "Керівництво користувача" (Додаток В). При роботі з програмним забезпеченням системи також варто керуватися таким документом, як: "Загальний опис роботи ІС моніторингу стану агротехнічного підприємства" (Додаток Б).

5. РОЗРАХУНОК ЕКОНОМІЧНОЇ ЕФЕКТИВНОСТІ ВПРОВАДЖЕННЯ ІНФОРМАЦІЙНОЇ УПРАВЛЯЮЧОЇ СИСТЕМИ АГРОТЕХНІЧНОГО ПІДПРИЄМСТВА

Проблема пошуку дефектів цифрових пристроїв є актуальною протягом декількох останніх десятиріч. Це визначається швидким розвитком і ускладненням структури радіоелектронних приладів. З ускладненням цифрових пристроїв ускладнюються і методи та засоби їх діагностики, що неминуче веде до значного погіршення ефективності діагностування дефектів існуючими системами.

Дана робота присвячена аналізу та оптимізації ресурсів, які необхідні для забезпечення вигідних умов роботи інформаційної системи моніторингу стану агротехнічного підприємства з метою підвищення ефективності роботи, і, як наслідок, підвищення прибутку підприємства.

Обґрунтувати доцільність розробки з економічної точки зору можливо розраховавши собівартість оптимізації програмного забезпечення і додатковий прибуток, що буде отриманий в результаті впровадження. Основним показником, що визначає економічну доцільність витрат на створення програмного забезпечення є річний економічний ефект. Характерним показником економічної ефективності створення і функціонування програмного продукту, що розраховується на стадії створення і впровадження і використання, є ефективність витрат і розрахунковий коефіцієнт економічної ефективності капітальних вкладень (строк окупності програмного забезпечення).

На першому етапі розраховується собівартість розробки програмного забезпечення. Потім капітальні витрати, необхідні для придбання устаткування. На підставі цього можна одержати річний економічний ефект і строки окупності даного проекту.

5.1 Розрахунок середньої вартості впровадження інформаційної системи моніторингу стану агротехнічного підприємства

Системи планування ресурсів підприємства – ERP (ERP, Enterprise Resource Planning) служать для інтеграції усіх даних і процесів організації в єдину систему. Для цього типова ERP – система використовує безліч різних програмних і апаратних компонентів. Ключовим компонентом більшості ERP – систем є єдина база даних, що зберігає в собі дані різних системних модулів.

Завдяки широкій сфері застосування ERP – системи зазвичай складні, і їх впровадження вимагає значної зміни навичок персоналу (у іншому випадку необхідність впровадження є спірною). При впровадженні ERP-системи зазвичай неможливо обійтися своїми фахівцями і доводиться притягати консультантів з боку, завдяки чому навіть невеликі проекти стають дорогими.

Тривалість впровадження ERP-системи залежить від розмірів підприємства, об'ємів доопрацювання і того, чи побажає замовник стати власником проекту.

Велике підприємство налічує близько 500-1000 чоловік і має широкий діапазон господарської діяльності. Організація такого підприємства передбачає призначення кваліфікованих управлінських кадрів на усі особливо важливі ділянки діяльності. Децентралізація управління потрібна при відмінностях в місцезнаходженні відділень фірми, а також при великій різноманітності асортименту товарів і категорій споживачів. Правління директорів представляє власників компанії. Воно призначає президента і головного керівника справами для управління корпорацією. Інші члени правління займають відповідальні пости у фірмі. Розрахуємо вартість впровадження КІС (таблиця 5.1).

Таблиця 5.1. Вартість впровадження КІС на великому підприємстві

Етап впровадження	Вартість	Кількість робочих місць	Загальна ціна на етапі
Програмне забезпечення	200 тис. грн. на 100 робочих місць	500-1000	1-2 млн. грн.
Ліцензія на ПЗ	6-7 тис. грн. на 10 робочих місць	500-1000	325-650 тис. грн.
Впровадження	500 грн. на 1 людину в день	10-15 людей протягом 40-50 робочих днів	225-338 тис. грн.
Навчання персоналу	2500 грн. за курс для 1 робітника	500-1000	1,25-2,5 млн. грн.

Проведення локальної мережі	Кабель – 4 грн./метр; Комутатор – 300 грн. на 10 машин; Налагодження мережі – 50 грн./машина	Кабель – 2000-3000 метрів; Комутатори – 50-100 штук; Машин – 500-1000	8-12 тис. грн. 15-30 тис. грн. 25-50 тис. грн.
Забезпечення комп'ютерами	3500 грн. за машину	500-1000	1,75-3,5 млн. грн.
Загальна сума			4,6-9,1 млн. грн.

Середнє підприємство

На ринку інформаційних систем є досить великий сегмент, що складається з середніх підприємств. Ці підприємства також мають потребу в комплексній інформаційній системі, але меншого масштабу і, відповідно, меншою вартістю. Їх реальна готовність придбати і впровадити таку систему знаходиться на рівні десятків тисяч доларів.

Під сучасним середнім підприємством видається комерційна організація переважно торгова або торгово-виробнича, штат якої налічує від 50 до 500 чоловік. Як правило, це вчорашні розвинуті малі фірми, що наслідують як організацію ведення бізнесу, необхідне устаткування, оргтехніку, комп'ютери, так і програмні продукти, що автоматизували раніше працю нечисленних співробітників.

Розрахуємо вартість впровадження КІС на середньому підприємстві (таблиця 5.2).

Таблиця 5.2. Вартість впровадження КІС на середньому підприємстві

Етап впровадження	Вартість	Кількість робочих місць	Загальна ціна на етапі
Програмне забезпечення	200 тис. грн. на 100 робочих місць	50-500	100 тис. – 1 млн. грн.
Ліцензія на ПЗ	6-7 тис. грн. на 10 робочих місць	50-500	32,5-325 тис. грн.
Впровадження	500 грн. на 1 людину в день	5-10 людей протягом 40-50 робочих днів	113-225 тис. грн.
Навчання персоналу	2500 грн. за курс для 1 робітника	50-500	125 тис.-1,25 млн. грн.
Проведення локальної мережі	Кабель – 4 грн./метр; Комутатор-300 грн. на 10 машин; Налагодження мережі – 50 грн./машина	Кабель 200-1000 метрів; Комутатори – 5-50 штук; Машин – 50-500	0,8-4 тис. грн. 1,5-15 тис. грн. 2,5-25 тис. грн.

Забезпечення комп'ютерами	3500 грн. за машину	50-500	175 тис.-1,75 млн. грн.
Загальна сума			550 тис. – 4,6 млн. грн.

Мале підприємство

Малим підприємством вважається таке підприємство, яке має у своєму складі до 25 працівників. Організація такого підприємства може пройти декілька стадій доки, нарешті, власник не вирішить питання про розподіл прав і відповідальності працівників. Основні функції працівників полягають в здійсненні основної діяльності і включають: продаж, закупівлю, зберігання товарів і обслуговування ними споживачів.

Розрахуємо вартість впровадження КІС на малому підприємстві (таблиця 5.3).

Таблиця 5.3. Вартість впровадження КІС на малому підприємстві

Етап впровадження	Вартість	Кількість робочих місць	Загальна ціна на етапі
Програмне забезпечення	20 тис. грн. на 10 робочих місць	10-25	20-50 тис. грн.
Ліцензія на ПЗ	6-7 тис. грн. на 10 робочих місць	10-25	6,5-16 тис. грн.
Впровадження	500 грн. на 1 людину в день	1-2 людей протягом 40-50 робочих днів	22,5-45 тис. грн.
Навчання персоналу	2500 грн. за курс для 1 робітника	10-25	25-62,5 тис. грн.
Проведення локальної мережі	Кабель – 4 грн./метр; Комутатор – 300 грн. на 10 машин; Налагодження мережі – 50 грн./машина	Кабель – 50-200 метрів; Комутатори – 1-3 штук; Машин – 10-25	200-800 грн. 300-900 грн. 0,5-1,25 тис. грн.
Забезпечення комп'ютерами	3500 грн. за машину	10-25	35-87,5 тис. грн.
Загальна сума			110 – 388 тис. грн.

Отже, якщо враховувати вище розраховані ціни, то отримаємо, що **загальна вартість впровадження інформаційної системи** складатиме:

- для малого підприємства: 110 – 388 тис. грн.
- для середнього підприємства: 550 тис. – 4,6 млн. грн.
- для великого підприємства: 4,6 – 9,1 млн. грн.

5.2 Розрахунок витрат на оптимізацію й експлуатацію інформаційної системи

Розглянемо зведені витрати B , необхідні на розробку та впровадження автоматизованої системи:

$$B = C + K, \quad (5.1)$$

де C – поточні витрати;

K – капітальні вкладення.

Вільні витрати B розраховують у сфері виробництв (у розроблювача) або в сфері експлуатації (у користувача). У нашому випадку доцільно розраховувати витрати в сфері виробництва. Поточні витрати при розробці C – це собівартість створення системи C_C :

$$C_C = C_P + C_B + C_M + C_H, \quad (5.2)$$

де C_P – заробітна плата розробників;

C_B – відрахування на соціальні заходи (47,5% від фонду оплати праці).

C_M – вартість машинного часу, необхідного для розробки-настроювання системи;

C_H – накладні витрати в розмірі 50-100% від витрат на оплату праці (містять оплату праці управлінському персоналу з нарахуваннями оплат службових відряджень; оплату консультативно-інформаційних послуг оплати ремонту і техобслуговування основних фондів, крім ПК; оплату оренди приміщення).

$$C_M = t_m \cdot C_{M-H}, \quad (5.3)$$

де t_m – кількість машинного часу, необхідного для розробки (год.);

C_{M-H} – собівартість 1 машино-години роботи ПК (грн.):

$$C_{M-H} = \frac{\sum_i U_i}{T_{ЕФ}}, \quad (5.4)$$

де U_i – річні експлуатаційні витрати при обслуговуванні ПК, що включають:

– зарплату фахівця, що обслуговує машину (з виправленням на коефіцієнт

зайнятості працівника обслуговуванням ПК);

- нарахування на зарплату (37,5%);
- витрати на електроенергію, що визначаються як добуток вартості 1 кВт/год. електроенергії на потужність, споживаної ПК, і на ефективний фонд часу ПК (за рік);

- накладні витрати – 50-150% від витрат на оплату праці;

$T_{\text{ЕФ}}$ – ефективний фонд часу роботи ПК за рік, що визначається виходячи з календарного річного фонду часу, крім вихідних і святкових днів.

$$C_{M-\Gamma} = \frac{500 + 500 \cdot 0.375 + 500 \cdot 0.5 + 252}{1512} = 0.787 \text{ грн./год.};$$

Собівартість створення системи при тривалості розробки в 2 місяці та навантаженні роботи 6 годин з 8ми на день дорівнює:

$$C_P = 3600 \cdot 2 \cdot 2 \cdot \frac{5}{8} = 9000 \text{ грн.}; C_B = 9000 \cdot 0.475 = 4275 \text{ грн.};$$

$$C_M = 211.1 \cdot 0.787 = 166.1 \text{ грн.}; C_H = 9000 \cdot 0.5 = 4500 \text{ грн.};$$

$$C_C = 9000 + 4275 + 166.1 + 4500 = 17941.1 \text{ грн.}$$

Капітальні вкладення K – це витрати, що носять одноразовий характер. У даному випадку внесемо сюди вартість покупного ліцензійного програмного забезпечення, технічного й організаційного забезпечення, необхідного для створення системи.

Вартість необхідного програмного забезпечення на сьогоднішній день приведена в таблиці 5.4. Вартість необхідного технічного забезпечення приведена в таблиці 5.5.

Таблиця 5.4. Вартість програмного забезпечення

<i>Найменування ПЗ</i>	<i>Вартість, грн.</i>	<i>Кількість</i>	<i>Разом, грн.</i>
Komodo IDE 5.3	2360	2	4720

Таблиця 5.5. Вартість технічного забезпечення

<i>Найменування технічного засобу</i>	<i>Вартість, грн.</i>	<i>Кількість</i>	<i>Разом, грн.</i>
ПК Intel Core i7-920	9242	1	9242

Витрати на впровадження системи включають установку ПО, навчання персоналу. Витрати на впровадження складають 36000 грн. Капітальні вкладення приведені в таблиці 5.6.

Таблиця 5.6. Капітальні вкладення по варіантах

<i>ПЗ, грн.</i>	<i>Технічні засоби, грн.</i>	<i>Впровадження, грн.</i>	<i>Разом, грн.</i>
4720	9242	36000	49962

Загальні витрати зведені до таблиці 5.7.

Таблиця 5.7. Зведені витрати

<i>Собівартість, грн.</i>	<i>Капітальні вкладення, грн.</i>	<i>Витрати, грн.</i>
17941,1	49962	67903,1

5.3 Розрахунок економічної ефективності

Згідно прийнятої схеми роботи підприємства, реорганізація підприємства проводиться раз на квартал. Прийняття рішення стосовно реорганізації займає один робочий день, тобто 8 годин. Підприємство має 19 представництв, на кожному з яких працює по менеджеру, відповідальному за реорганізаційні заходи. Заробітна платня менеджера – 4000 грн. Також є 7 головних менеджерів, що також приймають участь у прийнятті рішення. Їхня заробітна платня складає 8000 грн. При впровадженні даної системи весь цей час фактично економиться.

Отже, вивільняється наступна кількість годин, витрачених на реорганізацію роботи підприємства:

Менеджери: $19 \cdot 4 \cdot 8 = 608$ год./рік;

Головні менеджери: $7 \cdot 4 \cdot 8 = 224$ год./рік.

Також менеджери щотижня готують звіт щодо змін у попиту на продукцію, що займає в них 3 години (щоп'ятниці). При впровадженні даної системи, звіт немає необхідності готувати, потрібно лише передати системі дані, що займає лише півгодини. Отже додатково економиться:

Менеджери: $19 \cdot 50.66 \cdot (3 - 0.5) = 2406$ год./рік,

де 50,66 – середня кількість п'ятниць у році.

Визначимо вивільнені кошти після впровадження продукту, ґрунтуючись на тому, що середня кількість робочих годин на місяць складає 168 год.:

$$\frac{4000}{168} \cdot (608 + 2406) + \frac{8000}{168} \cdot 224 = 82428.57 \text{ грн./рік};$$

Визначимо пряму економічну ефективність:

$$E_{ef} = \Delta C - E_K \cdot \Delta K, \quad (5.5)$$

де ΔC – це вивільнені кошти після впровадження продукту мінус експлуатаційні витрати;

E_K – коефіцієнт амортизації – 0,6;

ΔK – одноразові витрати (тобто, вартість програми);

$$\Delta C = 82428.57 - 9242 \cdot 0.6 = 76883.37 \text{ грн.};$$

$$E_{ef} = 76883.37 - 0.6 \cdot 67903.1 = 36141.51 \text{ грн.}$$

5.4 Розрахунок строку окупності програми

До впровадження програмного продукту на підприємстві витрачалося 14476,19 грн. на заробітну плату менеджерам представництв. При оптимізації інформаційної системи ця сума вивільняється вивільняються. Затрати на розробку та впровадження складають 9458 грн. Строк окупності продукту вираховуємо так:

$$T_{OK} = \frac{\Delta K}{\Delta C} = \frac{67903.1}{76883.37} = 0.88 \text{ року, тобто } 10,5 \text{ місяців.}$$

Таким чином витрати на створення програмного продукту окупляться протягом перших 10 з половиною місяців.

6. ОХОРОНА ПРАЦІ

6.1 Загальні визначення та актуальність питання охорони праці

Охорона праці є системою правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних та лікувально-профілактичних заходів та засобів, спрямованих на збереження здоров'я і працездатності людей в процесі праці.

Як відомо, цілком безпечних і нешкідливих виробництв не існує, тому основною задачею охорони праці є зведення до мінімуму імовірність поразки чи захворювання працюючого з одночасним забезпеченням комфорту при максимальній продуктивності праці. Основний принцип у галузі охорони праці – це пріоритет життя і здоров'я працівника по відношенню до результатів виробничої діяльності. При цьому роботодавці зобов'язані забезпечувати здорові і безпечні умови праці для своїх працівників, вживати всіх заходів щодо запобігання ризиків на робочих місцях, навчати працівників правилам і нормам охорони праці і виконувати інші вимоги законодавства.

Турбота про охорону праці є справою державної важливості. 14 жовтня 1992 року був прийнятий Закон України "Про охорону праці", у якому закріплені найважливіші положення в області охорони праці, і який поширюється на всіх суб'єктів підприємницької діяльності незалежно від форм власності.

Ст.4 Закону "Про охорону праці" формулює принципи державної політики у сфері охорони праці. Ці принципи в силу вказаного їх визначення, слід кваліфікувати і як принципи інституту охорони праці, які підлягають застосуванню при відсутності відповідних конкретних правових норм, і як такі, що визначають напрямки подальшого розвитку інституту охорони праці, і обов'язки держави, які вона взяла на себе і має додержувати при здійсненні своїй політики у майбутньому. З урахуванням цього принципи, зазначені ст. 4 Закону "Про охорону праці" мають певне регулятивне значення. В статті закріплені такі принципи:

- пріоритет життя і здоров'я працівників відносно результатів виробничої

діяльності підприємства, повна відповідальність власника за створення безпечних і нешкідливих умов праці;

- комплексне вирішення завдань охорони праці на засадах національних програм з цих питань, а також з урахуванням інших напрямків економічної і соціальної політики, досягнень в галузі науки, техніки охорони здоров'я;

- соціальний захист працівників, повне відшкодування шкоди особам потерпілим від нещасних випадків на виробництві та професійних захворювань;

- установлення єдиних нормативів з охорони праці для всіх підприємств незалежно від форм власності і видів їх діяльності;

- використання економічних методів управління охороною праці. проведення політики пільгового оподаткування, то сприяє створенню безпечних та нешкідливих умов праці, участь держави в фінансуванні заходів з охорони праці;

- здійснення навчання населення, професійної підготовки та підвищення кваліфікації працівників з питань охорони праці;

- забезпечення координації діяльності державних органів, установ, організацій та об'єднань громадян, що вирішують різні проблеми охорони здоров'я, гігієни та безпеки праці, а також співробітництво.

Державне керування органами охорони праці здійснюють: Кабінет Міністрів України, державний комітет України по нагляду за охороною праці, міністерство й інші централізовані органи, місцеві ради народних депутатів і їхні виконавчі комітети. Державний нагляд за дотриманням законодавства й інших нормативних актів по охороні праці здійснюють: державний комітет України по нагляду за охороною праці, державний комітет України по атомній і радіаційній безпеці, органи державного пожежного нагляду, органи і служби санітарно епідемічного нагляду. А трудові колективи в особі своїх уповноважених і профспілки здійснюють суспільний контроль за охороною праці.

Вищий нагляд за точним виконанням законів про працю міністерствами і відомствами, підприємствами, установами, громадянами і посадовими особами

покладений на Генерального Прокурора України.

Велике значення в створенні здорових і безпечних умов праці має стандартизація. Вона дозволяє вжити діючих заходів по підвищенню технічного рівня й упорядкуванню розробки нормативно-технічної документації по безпеці праці. З 1972 року створюється Система стандартів по безпеці праці (ССБП), що являє собою комплекс великого числа взаємозалежних стандартів, спрямованих на забезпечення безпеки праці. Ця система встановлює загальні вимоги і норми по видах небезпечних і шкідливих виробничих факторів, загальні вимоги безпеки до виробничого устаткування і процесів, вимоги до засобів захисту працюючих, методи оцінки безпеки праці.

Охорона праці – один з центральних інститутів трудового права. Він має виключно практичне значення. Недодержання вимог охорони праці створює небезпеку для здоров'я і життя працівників. У свою чергу і ті, кого законодавець називає власником або уповноваженим ним органом, несуть сувору, у тому числі й кримінальну, відповідальність за порушення правил охорони праці.

Однією із характерних особливостей сучасного розвитку суспільства є зростання сфер діяльності людини, в яких використовуються інформаційні технології. Широке розповсюдження отримали персональні комп'ютери. Однак їх використання загостило проблеми збереження власного та суспільного здоров'я, вимагає удосконалення існуючих та розробки нових підходів до організації робочих місць, проведення профілактичних заходів для запобігання розвитку негативних наслідків впливу ПК на здоров'я користувачів. Тому питання охорони праці при роботі з ПК є досить актуальним.

6.2 Аналіз небезпечних і шкідливих факторів при роботі з ПК

З точки зору впливу небезпечних і шкідливих факторів при роботі з ПК можна виділити наступні:

- вплив електромагнітного та електростатичного випромінювань;
- вплив недостатньої освітленості робочого місця;
- вплив електричного струму;
- вплив шуму;

- вплив несприятливих умов мікроклімату;
- вплив нераціонального розташування устаткування і неправильної організації робочого місця.

Тривале перебування людини в зоні комбінованого впливу різних несприятливих факторів може призвести до професійного захворювання. Розглянемо більш детально вплив означених факторів на користувача ПК.

1) Вплив електромагнітного та електростатичного випромінювань.

При роботі ПК виникають два типи випромінювань: електростатичне та електромагнітне. Навколо електростатично-зарядженого монітора підвищується концентрація пилу. Електромагнітне випромінювання створюється магнітними котушками. Результати медичних досліджень показують, що електризований пил викликає запалення шкіри, а електромагнітне випромінювання приводить до порушення біологічних процесів (порушення сну, зниження кров'яного тиску, уповільнення серцевої діяльності, зміни складу крові).

2) Вплив недостатньої освітленості робочого місця.

Напружена зорова робота внаслідок нераціонального освітлення може бути причиною функціональних порушень у зоровому аналізаторі і привести до розладу зору, а у важких випадках – і до повної втрати. Втома органів зору залежить від ступеня напруженості процесів, що супроводжують зорове сприйняття. При роботі оператора ПК одним з основних навантажень на організм є навантаження на зір, тому дуже важливо приділити особливу увагу освітленості робочого місця.

У залежності від джерела світла виробниче висвітлення може бути трьох видів: природним, штучним і сполученим. При роботі з ПК вдень у ясну погоду використовується природне освітлення, яке здійснюється через світлові віконні прорізи. При нестачі природного світла, а також при роботі у вечірній час використовується штучне освітлення, джерелами якого є газорозрядні лампи. При такому освітленні величезне значення має правильне розташування джерел світла, їхній тип і кількість.

Правильна освітленість робочого місця сприяє високій працездатності користувача ПК і зводить до мінімуму шкідливий вплив на органи зору.

Освітленість робочого місця, що рекомендується для користувача ПК складає 400–750 лк.

3) Вплив електричного струму.

Робота комп'ютерів і периферійних пристроїв можлива завдяки електричному струмові, до джерел якого вони підключаються, тому існує небезпека ураження користувача струмом. Щоб уникнути нещасних випадків варто строго дотримуватися правил електробезпеки.

Небезпека електричного струму на відміну від інших небезпек збільшується за рахунок того, що людина не в змозі без спеціальних приладів виявити напругу дистанційно, а також швидкоплинності поразки – небезпека виявляється, коли людина вже уражена. Ураження струмом приводить до різних порушень в організмі, викликаючи як місцеву поразку тканин і органів, так і загальне ураження організму. Людина відчуває дратівну дію перемінного струму промислової частоти силою 0,6–1,5 мА і постійного струму 5–7 мА. Ці струми не представляють серйозної небезпеки для організму людини, тому що при такій силі струму можливе самостійне звільнення людини від контакту зі струмоведучими частинами. Для перемінного струму промислової частоти сила невідпускаючого струму знаходиться в межах 6–20 мА. Постійний струм не викликає невідпускаючого ефекту, але приводить до сильних болючих відчуттів, сила такого струму 15–80 мА і більше.

4) Вплив шуму.

Шумом прийнято називати всякий небажаний для людини звук, що заважає сприйняттю корисних сигналів. Шум являє собою безладне сполучення звуків різної інтенсивності і частоти. Він викликає труднощі з розпізнанням колірних сигналів, знижує швидкість сприйняття кольору, гостроту зору, зорову адаптацію, порушує сприйняття візуальної інформації, зменшує на 5–12% продуктивність праці. Тривалий вплив шуму з рівнем звукового тиску 90 дБ знижує продуктивність праці на 30–60%.

Робота ПК може супроводжуватися шумами, викликаними деякими рухливими пристроями усередині комп'ютера, а також підключеними до нього. До таких пристроїв можна віднести вентилятори, пристрої читання дисків,

принтери. Виникаючий шум, у залежності від інтенсивності, може привести до підвищення нервової напруги і зниження працездатності.

5) Вплив несприятливих умов мікроклімату.

Мікрокліматичні параметри (температура, відносна вологість і швидкість руху повітря) впливають на самопочуття і здоров'я людини, а також на надійність роботи засобів обчислювальної техніки.

При підвищеній температурі і вологості в приміщенні відбувається перегрівання тіла, що погіршує самопочуття і працездатність. Варто враховувати, що ПК, які знаходяться в приміщенні, сприяють збільшенню температури.

Для комфортної роботи приймається температура $23 (\pm 2)^{\circ}\text{C}$ й $18 (\pm 2)^{\circ}\text{C}$ в теплий і холодний час року відповідно, при відносній вологості $55 (\pm 5)\%$.

б) Вплив нераціонального розташування устаткування і неправильної організації робочого місця.

Робоче місце – це частина простору, у якій користувач ПК здійснює трудову діяльність і проводить велику частину робочого часу. Важливу роль грає планування робочого місця, яке повинне задовольняти вимогам зручності виконання робіт і економії енергії і часу користувача, зручності обслуговувань ЕОМ, дотримання правил охорони праці.

При плануванні робочого місця необхідно враховувати зони досяжності при розташуванні моніторів, клавіатури, маніпулятора "миша". Ці зони встановлюються на підставі антропологічних даних людського тіла. Під час роботи користувач ПК не повинен піддаватися впливові сторони подразників, якими можуть бути неадекватне фарбування пристроїв ЕОМ або приміщення.

6.3 Розрахунок системи загального освітлення приміщення з ПК

Рациональне освітлення приміщень – один з найбільш важливих факторів, від яких залежить ефективність трудової діяльності людини. Гарне освітлення необхідне для виконання більшості задач.

До систем освітлення висувають наступні вимоги:

– відповідність рівня освітленості робочих місць характеру виконуваної

зорової роботи;

- досить рівномірний розподіл яскравості на робочих поверхнях і в навколишньому просторі;
- сталість освітленості в часі;
- оптимальна спрямованість випромінюваного освітлювальними приладами світлового потоку;
- довговічність, економічність, електро- та пожежабезпечність, естетичність, зручність і простота експлуатації.

В приміщенні з ПК необхідно організувати змішане освітлення (комбінування природного і штучного освітлення). Природне освітлення створюється бічним освітленням через вікно. Штучне освітлення використовується при недостатньому природному освітленні. У даному приміщенні використовується загальне штучне освітлення. Розрахунок виконаємо за методом світлового потоку з урахування потоку, віддзеркаленого від стін та стелі.

Необхідно зробити розрахунок загального освітлення виробничої приміщення. Довжина $A = 10$ м, ширина $B = 7$ м, висота $H = 3$ м. Висота робочої поверхні $h_p = 1$ м. Для освітлення використовується світильні УПМ-15. Мінімальна освітленість лампи наколювання по нормах $E_{min} = 100$ лк. Коефіцієнт відображення стелі $S_n = 50\%$, стін $S_c = 30\%$, робочої поверхні $S_p = 10\%$. Напруга мережі 220 В.

Відстань від стелі до робочої поверхні:

$$H_0 = H - h_p = 3 - 1 = 2 \text{ м.}$$

Відстань від стелі до світильника:

$$h_c = 0,2 \cdot H_0 = 0,2 \cdot 2 = 0,4 \text{ м.}$$

Висота підвісу світильника над освітленою поверхнею:

$$h = H_0 - h_c = 2 - 0,4 = 1,6 \text{ м.}$$

Висота підвісу світильника над підлогою:

$$H_n = h + h_p = 1,6 + 1 = 2,6 \text{ м.}$$

Для досягнення найбільшої рівномірності освітлення приймаємо

відношення $L/h = 1,5$.

Відстань між центрами світильників:

$$L = 1,5 \cdot h = 1,5 \cdot 1,6 = 2,4 \text{ м.}$$

Необхідна кількість світильників:

$$N = S/L^2 = (10 \cdot 7) / (2,4)^2 \approx 12.$$

Приймаємо кількість світильників: 12 (3 ряди по 4 світильники).

Індекс приміщення:

$$i = (A \cdot B) / (h \cdot (A + B)) = (10 \cdot 7) / (1,6 \cdot (10 + 7)) = 2,57$$

(найближче значення в таблиці 2,5).

По таблиці коефіцієнтів використання світлового потоку при $i = 2,5$, $S_n = 50\%$, $S_c = 30\%$, $S_p = 10\%$ для світильника типу "УПМ-15" коефіцієнт використання світлового потоку $\eta = 0,59$.

Світловий потік однієї лампи:

$$\Phi = (E_{min} \cdot S \cdot K_z \cdot Z) / (N \cdot \eta) = (100 \cdot 70 \cdot 1,5 \cdot 1,1) / (12 \cdot 0,59) = 1631 \text{ лм,}$$

де K_z – коефіцієнт запасу, який враховує запилення світильників і знос джерел світла в процесі експлуатації; для приміщень, освітлюваних люмінесцентними лампами $K_z = 1,5$ за умови чищення світильників не рідше двох разів на рік;

Z – коефіцієнт нерівномірності освітлення ($Z = 1,1$);

За знайденим світовим потоком із таблиці "Параметри ламп накаливання загального призначення з розрахунковими напругами 130 і 220 В" обираємо лампу Г 215-225-150 накаливання потужністю 150 Вт, яка має світовий потік 2090 лк, що найбільше близький до розрахункового.

Фактична освітленість:

$$E_{\phi} = E_{min} \cdot \Phi_{\lambda} / \Phi_p = 100 \cdot 2090 / 1631 = 128,14 \text{ лк.}$$

Загальна потужність:

$$P_{\text{заг}} = P_{\text{л}} \cdot N = 150 \cdot 12 = 1800 \text{ Вт} = 1,8 \text{ кВт.}$$

Таким чином, для оптимальної роботи в приміщенні необхідно встановити 3 ряди по 4 світильники типу Г 215-225-150. Загальна потужність усіх світильників складе 1,8 кВт.

6.4 Розробка заходів щодо забезпечення сприятливих умов праці при роботі з персональним комп'ютером

При виконанні робіт на комп'ютерах необхідно дотримуватись вимог загальної та даної інструкції з охорони праці.

До самостійної роботи на комп'ютерах допускаються особи, які пройшли медичний огляд, навчання по професії, вступний інструктаж з охорони праці та первинний інструктаж з охорони праці на робочому місці. В подальшому вони проходять повторні інструктажі з охорони праці на робочому місці один раз на півріччя, періодичні медичні огляди один раз на два роки.

Під час роботи на комп'ютерах можуть діяти такі небезпечні та шкідливі фактори, як:

- фізичні;
- психофізіологічні.

Основним обладнанням робочого місця користувача комп'ютера є монітор, системний блок та клавіатура.

Робочі місця мають бути розташовані на відстані не менше 1,5 м від стіни з вікнами, від інших стін на відстані 1м, між собою на відстані не менше 1,5 м. Відносно вікон робоче місце доцільно розташовувати таким чином, щоб природне світло падало на нього збоку, переважно зліва.

Робочі місця слід розташовувати так, щоб уникнути попадання в очі прямого світла. Джерела освітлення рекомендується розташовувати з обох боків екрану паралельно напрямку погляду. Для уникнення світлових відблисків екрану, клавіатури в напрямку очей користувача, від світильників загального освітлення або сонячних променів, необхідно використовувати антиполюскові сітки, спеціальні фільтри для екранів, захисні козирки, на вікнах – жалюзі.

Фільтри з металевої або нейлонової сітки використовувати не рекомендується, тому що сітка спотворює зображення через інтерференцію світла. Найкращу якість зображення забезпечують скляні поляризаційні фільтри. Вони усувають практично всі відблиски, роблять зображення чітким і контрастним.

При роботі з текстовою інформацією (в режимі введення даних та редагування тексту, читання з екрану) найбільш фізіологічним правильним є зображення чорних знаків на світлому (чорному) фоні.

Монітор повинен бути розташований на робочому місці так, щоб поверхня екрана знаходилася в центрі поля зору на відстані 400-700 мм від очей користувача. Рекомендується розміщувати елементи робочого місця так, щоб витримувалася однакова відстань очей від екрана, клавіатури, тексту.

Зручна робоча поза при роботі з комп'ютером забезпечується регулюванням висоти робочого столу, крісла та підставки для ніг. Раціональною робочою позою може вважатися таке положення, при якому ступні працівника розташовані горизонтально на підлозі або підставці для ніг, стегна зорієнтовані у горизонтальній площині, верхні частини рук – вертикальні. Кут ліктьового суглоба коливається в межах 70-90°, зап'ястя зігнуті під кутом не більше ніж 20°, нахил голови 15-20°.

Для нейтралізації зарядів статичної електрики в приміщенні, де виконується робота на комп'ютерах, в тому числі на лазерних та світлодіодних принтерах, рекомендується збільшувати вологість повітря за допомогою кімнатних зволожувачів. Не рекомендується носити одяг з синтетичних матеріалів.

Вимоги безпеки перед початком роботи:

- увімкнути систему кондиціонування в приміщенні;
- перевірити надійність встановлення апаратури на робочому столі.

Повернути монітор так, щоб було зручно дивитися на екран – під прямим кутом (а не збоку) і трохи зверху вниз, при цьому екран має бути трохи нахиленим, нижній його край ближче до оператора;

- перевірити загальний стан апаратури, перевірити справність електропроводки, з'єднувальних шнурів, штепсельних вилок, розеток, заземлення захисного екрана;

- відрегулювати освітленість робочого місця;
- відрегулювати та зафіксувати висоту крісла, зручний для користувача нахил його спинки;

- приєднати до системного блоку необхідну апаратуру. Усі кабелі, що з'єднують системний блок з іншими пристроями, слід вставляти та виймати при вимкненому комп'ютері;

- ввімкнути апаратуру комп'ютера вимикачами на корпусах в послідовності: монітор, системний блок, принтер (якщо передбачається друкування);

- відрегулювати яскравість свічення монітора, мінімальний розмір світної точки, фокусування, контрастність. Не слід робити зображення надто яскравим, щоб не втомлювати очей.

Рекомендується:

- яскравість свічення екрана – не менше 100 kg/m^2 ;
- відношення яскравості монітора до яскравості оточуючих його поверхонь в робочій зоні – не більше 3:1;

- мінімальний розмір точки свічення не більше 0,4 мм для монохромного монітора і не менше 0,6 мм для кольорового, контрастність зображення знаку – не менше 0,8.

При виявленні будь-яких несправностей роботу не розпочинати, повідомити про це керівника.

Вимоги безпеки під час виконання роботи:

- необхідно стійко розташовувати клавіатуру на робочому столі, не опускати її хитання. Під час роботи на клавіатурі сидіти прямо, не напружуватися;

- для забезпечення несприятливого впливу на користувача пристроїв типу "миша" належить забезпечувати вільну велику поверхню столу для переміщення "миші" і зручного упору ліктьового суглоба;

- не дозволяються по сторонні розмови, подразнюючі шуми;

- періодично при вимкненому комп'ютері прибирати ледь змоченою мильним розчином бавовняною ганчіркою порох з поверхонь апаратури. Екран ВДТ та захисний екран протирають ганчіркою, змоченою у спирті. Не дозволяється використовувати рідинні або аерозольні засоби чищення поверхонь комп'ютера.

Забороняється:

- самостійно ремонтувати апаратуру. Ремонт апаратури здійснюється спеціалістами з технічного обслуговування комп'ютера;
- класти будь-яку предмети на апаратуру комп'ютера;
- закривати будь-чим вентиляційні отвори апаратури, що може призвести до її перегрівання і виходу з ладу.

Для зняття статичної електрики рекомендується час від часу доторкатися до металевих поверхонь.

Розташувати принтер необхідно поруч з системним блоком таким чином, щоб з'єднувальний шнур не був натягнутий. Забороняється ставити принтери на системний блок.

Для досягнення найбільш чистих, з високою розподільністю зображень і щоб не зіпсувати апарат, має використовуватися папір, вказаний в інструкції до принтера. При застряганні паперу потрібно відкрити кришку і обережно витягнути лоток з папером.

Згідно з інструкцією фірми-виробника потрібно дотримуватися правил зберігання картриджа.

Забороняється:

- зберігати картриджі без упаковки;
- ставити картриджі вертикально;
- перевертати картридж етикеткою донизу;
- відкривати кришку валика і доторкатися до нього;
- самому заповнювати використаний картридж.

Вимоги безпеки після закінчення роботи:

- закінчити та записати у пам'ять комп'ютера файл, що знаходиться в роботі;
- вимкнути принтер та інші периферійні пристрої. Штепсельні вилки витягнути з розеток. Накрити клавіатуру кришкою запобігання попаданню в неї пилу;
- прибрати робоче місце;
- ретельно вмити руки теплою водою з милом;

- вимкнути кондиціонер, освітлення і загальне електроживлення;

пройти в спеціально обладнаному приміщенні сеанс психофізіологічного розвантаження і зняття втоми з виконанням спеціальних вправ аутогенного тренування.

7. ОХОРОНА НАВКОЛИШНЬОГО СЕРЕДОВИЩА

7.1 Вступна частина

У наш час захист навколишнього середовища висувається на перший план. Наслідки недостатньої уваги до проблеми можуть бути катастрофічними. Мова йде не тільки про добробут людства, а про його виживання. Особливо тривожно те, що деградація природного середовища і може виявитися необоротною.

Під охороною навколишнього середовища розуміють сукупність міжнародних, державних і регіональних правових актів, інструкцій і стандартів, які доводять загальні юридичні вимоги до кожного конкретного забруднювача і забезпечують його зацікавленість у виконанні цих вимог, конкретних природоохоронних заходів по втіленню в життя цих вимог.

Тільки якщо всі ці складові частини відповідають один одному за змістом і темпами розвитку, тобто складаються в єдину систему охорони навколишнього природного середовища, можна розраховувати на успіх.

Оскільки не була вирішена вчасно завдання охорони природи від негативного впливу людини, тепер все частіше постає завдання захисту людини від впливу змінилася природного середовища. Обидва ці поняття інтегруються в терміні "охорона навколишнього (людини) природного середовища".

Охорона навколишнього природного середовища складається з:

- правової охорони, що формулює наукові екологічні принципи у вигляді юридичних законів, обов'язкових для виконання;
- матеріального стимулювання природоохоронної діяльності з ціллю зробити її економічно вигідною для підприємств;
- інженерної охорони, що розробляє природоохоронну та ресурсозберігаючу технологію і техніку.

Основними принципами охорони навколишнього природного середовища є:

- пріоритет забезпечення сприятливих екологічних умов для життя, праці та відпочинку населення;
- науково обґрунтоване поєднання екологічних і економічних інтересів

суспільства;

- облік законів природи і можливостей самовідновлення і самоочищення її ресурсів;
- недопущення незворотних наслідків для охорони природного середовища і здоров'я людини;
- право населення та громадських організацій на своєчасну і достовірну інформацію про стан навколишнього середовища та негативному впливі на неї і на здоров'я людей різних виробничих об'єктів;
- невідворотність відповідальності за порушення вимог природоохоронного законодавства.

7.2 Забруднення навколишнього середовища підприємствами

Забруднення навколишнього середовища – надходження в природне середовище будь-яких твердих, газоподібних речовин, мікроорганізмів або енергій (у вигляді звуків, шумів, електромагнітного й радіоактивного випромінювання) у кількостях, що викликають зміну складу й властивостей компонентів природи й (або) чинять шкідливий вплив на людину, флору й фауну.

Види забруднень навколишнього середовища різні й різноманітні: викиди в атмосферу хімічних сполук і сумішей; надходження у водне середовище всіляких виробничих і комунально-побутових відходів, влучення в неї нафтових продуктів; засмічування ландшафтів сміттям і пакувальними матеріалами; забруднення полів, луків, лісів і водних об'єктів пестицидами; підвищення рівня іонізуючої радіації, виробничих і побутових шумів, вібрацій, а також нагромадження тепла в атмосфері.

У результаті господарської діяльності людини в атмосфері з'являється велика кількість забруднюючих речовин. Взаємодія атмосферного повітря з водою і ґрунтом призводить до якісних і кількісних змін усієї біосфери в цілому, підсилюючи і прискорюючи небажані зміни складу і структури атмосферного повітря, клімату Землі. Найбільш сильні зміни клімату і якості атмосферного повітря спостерігаються у великих містах. Якщо кисень в атмосферному повітрі

буде дуже сильно забруднений усілякими речовинами, то поступово у всього живого на Землі буде скорочуватися термін життя, доки не зменшиться до винищення всіх і всього.

Різноманітне втручання людини в природні процеси в біосфері можна згрупувати за наступними видами забруднень, розуміючи під ними будь-які небажані для екосистем антропогенні зміни:

- інгредієнтне (інгредієнт – складова частина складного з'єднання або суміші) забруднення як сукупність речовин, кількісно або якісно далеких природнім біогеоценозам;
- параметричне забруднення (параметр навколишнього середовища – одне з його властивостей, наприклад рівень шуму, освітленості, радіації і т.д.), пов'язане зі зміною якісних параметрів навколишнього середовища;
- біоценотичне забруднення, що полягає у впливі на склад і структуру популяції живих організмів;
- стаціонально-деструкційне забруднення (стація – місце проживання популяції, деструкція – руйнування), яке являє собою зміну ландшафтів екологічних систем у процесі природокористування.

Відомо більше 200 забруднюючих атмосферу речовин, причому в міру освоєння нових технологічних процесів і створення нових продуктів виробництва число забруднюючих речовин зростає. Цьому сприяють також реакції в атмосфері й інших середовищах між забруднюючими речовинами під впливом сонячного світла, окислювально-відновних процесів і інших реакцій. Стрімкий розвиток усіх галузей промисловості, енергетики, транспорту, збільшення чисельності населення й урбанізація, хімізація всіх сфер діяльності людини привели до визначених змін навколишнього природного середовища, у тому числі несприятливих, що полягають головним чином у забрудненні біосфери. Вплив шкідливих речовин антропогенного походження на природне середовище, а так само відгук середовища на ці впливи стають глобальними, всеосяжними.

У великих містах до числа основних джерел забруднення атмосферного повітря відноситься автотранспорт. Гази двигунів, що відходять, містять

складну суміш, їх більш ніж двісті компонентів, серед яких чимало канцерогенів. Найбільш значимі фактори негативного впливу автомобільного транспорту на людину і навколишнє середовище наступні:

- забруднення повітря;
- забруднення навколишнього середовища;
- шум, вібрація;
- виділення тепла (розсіювання енергії).

Основна причина забруднення повітря полягає у неповному і нерівномірному згорянні палива. Всього 15% його витрачається на рух автомобіля, а 85% "летить на вітер". До того ж камери згоряння автомобільного двигуна – це своєрідний хімічний реактор, що синтезує отруйні речовини і викидає їх в атмосферу.

Рухаючись зі швидкістю 80-90 км/год в середньому автомобіль перетворює на вуглекислоту стільки ж кисню, скільки 300-350 чоловік. Але справа не тільки в вуглекислоті. Річний вихлоп одного автомобіля – це 800 кг окису вуглецю, 40 кг окислів азоту і більше 200 кг різних вуглеводнів. У цьому наборі дуже підступна окис вуглецю. Через високу токсичність її допустима концентрація в атмосферному повітрі не повинна перевищувати 1 мг/м³.

Рівень загазованості магістралей залежить від інтенсивності руху автомобілів, ширини і рельєфу вулиці, швидкості вітру, частки вантажного транспорту та автобусів в загальному потоці і інших факторів. При інтенсивності руху 500 транспортних одиниць на годину концентрація окису вуглецю на відкритій території на відстані 30-40 м від автомагістралі знижується в 3 рази і досягає норми.

Ускладнене розсіювання викидів автомобілів на тісних вулицях. У результаті практично всі жителі міста відчувають на собі шкідливий вплив забрудненого повітря.

Чадний газ і окиси азоту, настільки інтенсивно виділяються на перший погляд безневинним голубуватим димком глушника автомобіля – ось одна з основних причин головних болів, втоми, невмотивованого роздратування, низькою працездатності. Сірчистий газ здатний впливати на генетичний апарат,

сприяючи безпліддя і природженої потворності, а всі разом ці чинники ведуть до стресів, нервовим проявам, прагненню до усамітнення, байдужості до найближчих людей. У великих містах також більш широко поширені захворювання органів кровообігу і дихання, інфаркти, гіпертонія і новоутворення. За розрахунками фахівців, "внесок" автомобільного транспорту в атмосферу становить до 90% по окису вуглецю і 70% по окису азоту. Автомобіль також додає в ґрунт і повітря важкі метали та інші шкідливі речовини.

В умовах сильного міського шуму відбувається постійна напруга слухового аналізатора. Це викликає збільшення порогу чутності (10 дБ для більшості людей з нормальним слухом) на 10-25 дБ.

Високі рівні шуму в міському середовищі є одним з агресивних подразників центральної нервової системи. Міський шум справляє негативний вплив і на серцево-судинну систему. Ішемічна хвороба серця, гіпертонічна хвороба, підвищений вміст холестерину в крові зустрічаються частіше в осіб, що проживають у гучних районах.

Найбільші рівні шуму 90-95 дБ відзначаються на магістральних вулицях міст з середньою інтенсивністю руху 2-3 тис. і більше транспортних одиниць на годину.

Рівень вуличних шумів обумовлюється інтенсивністю, швидкістю і характером (складом) транспортного потоку. Крім того, він залежить від планувальних рішень (поздовжній і поперечний профіль вулиць, висота і щільність забудови) і таких елементів благоустрою, як покриття проїзної частини та наявність зелених насаджень. Кожен з цих факторів здатний змінити рівень транспортного шуму в межах до 10 дБ.

Шум, що виникає на проїжджій частині магістралі, поширюється не тільки на при магістральну територію, але і вглиб житлової забудови. Так, в зоні найбільш сильного впливу шуму знаходяться частини кварталів і мікрорайонів, розташованих уздовж магістралей загальноміського значення (еквівалентні рівні шуму від 67,4 до 76,8 дБ). Рівні шуму, заміряні в житлових кімнатах при відкритих вікнах, орієнтованих на зазначені магістралі, всього на 10-15 дБ

нижче.

7.3 Основні заходи щодо зниження забруднення атмосферного повітря викидами автомобільного транспорту

Увага вчених, інженерів, агрономів, інших фахівців і громадськості зосереджено на усуненні всіх видів забруднюючих речовин, що представляють найбільшу небезпеку для здоров'я людей. Способи й шляхи боротьби із забрудненнями навколишнього середовища різноманітні: пристрої очисних споруджень, установка фільтрів, створення й впровадження безвідхідних технологій, замкнених циклів водокористування, заміна хімічних способів боротьби зі шкідниками й хворобами сільськогосподарських і лісових рослин біологічними, звукопоглинаючі пристрої, використання відходів для одержання корисної продукції, пошуки нових видів палива, використання для технічних і побутових потреб тепла геотермальних джерел, сили вітру, морських припливів, сонячного тепла, установлення режиму користування автомобільним і авіаційним транспортом

Проблема забруднення навколишнього середовища автомобільним транспортом цілком очевидно є глобальною. По всьому світу кількість автомобілів з кожним днем збільшується в геометричній прогресії. Все більше і більше людей мають свою власну машину. Але багато хто зовсім не замислюється про те, до чого все це врешті-решт приведе. Для того, щоб зберегти людству автомобіль, необхідно якщо не виключити, то звести до мінімуму шкідливі викиди. Робота в цьому напрямі ведеться в усьому світі і дають певні результати. Автомобілі випускаються в даний час в промислово розвинених країнах, викидають шкідливих речовин у 10-15 разів менше, ніж 10-15 років тому. У всіх розвинених країнах відбувається посилення нормативів на шкідливі викиди при роботі двигуна. У 2000 р. введені більш суворі норми. Відбувається не тільки кількісне посилення норм, але і їх якісна зміна. Постійно розширюється перелік речовин, зміст яких має знаходитися під контролем.

В даний час йде боротьба з автомобільною небезпекою. Конструюються фільтри, розробляються нові види пального, що містять менше свинцю.

Загальні шляхи зниження екологічних збитків від транспорту полягають в наступному:

- оптимізація руху міського транспорту
- розробка альтернативних енергетичних джерел;
- опалювання і очищення органічного палива;
- створення (модифікація) двигунів, що використовують альтернативні палива;
- захист від шуму;
- економічні ініціативи з управління автомобільним парком і рухом.

7.4 Заходи запобігання забруднення навколишнього середовища застарілою комп'ютерною технікою підприємства

Природоохоронною є будь-яка діяльність, спрямована на збереження якості навколишнього середовища на рівні, що забезпечує стійкість біосфери. До неї відноситься як великомасштабна, здійснювана на загальнодержавному рівні діяльність, так і діяльність окремих підприємств.

Існують два основних напрямки природоохоронної діяльності підприємств. Перший – очищення шкідливих викидів. Цей шлях "у чистому вигляді" малоефективний, тому що з його допомогою далеко не завжди вдається цілком припинити надходження шкідливих речовин у біосферу. До того ж скорочення рівня забруднення одного компонента навколишнього середовища веде до посилення рівня забруднення іншого.

Другий напрямок – усунення самих причин забруднення, що вимагає розробки маловідходних, а в перспективі й безвідходних технологій виробництва, які дозволяють комплексно використовувати вихідну сировину й утилізувати максимум шкідливих для біосфери речовин.

Однак далеко не для всіх галузей знайдені прийнятні техніко-економічні рішення по різкому скороченню кількості відходів, що утворюються, і їхньої утилізації, тому в цей час доводиться працювати по обох зазначених напрямках.

Людство знаходиться в постійній залежності від процесу модернізації і

заміни електронної техніки. Виробництво офісного устаткування, комп'ютерів і мобільних пристроїв з кожним роком нестримно збільшується. Проте є в прогресу і неприваблива сторона – застарілу апаратуру, що вийшла з використання, необхідно кудись дівати – виникає поняття "електронне сміття", і про його об'єми можна лише здогадуватися.

На відміну від органічних відходів, техногенне сміття неможливо утилізувати, використовуючи природні процеси регенерації. Більш того, враховуючи присутність в багато компонентах важких металів і отруйних речовин, спалювати його подібно до побутових відходів теж не можна. Переробка застарілої електроніки вимагає значних зусиль і є чималою проблемою. Ігнорувати її неможливо, тому що вона несе серйозну загрозу для екології.

Навіть по приблизним оцінках з 1991 р. до України було завезено більше 10 млн. одиниць персональної і оргтехніки загальною масою понад 400 тис. т і ще порядку 20 млн мобільних телефонів (близько 2400 т). Причому в сучасної електроніки є своя специфіка – вона в набагато більшій мірі складається з неліквідних матеріалів. Так, один комп'ютер класу 286 або 386 в комплекті з ЕПТ-монітором орієнтування містить від 10 до 20 кг скла, 3 кг АВС-пластика, приблизно стільки ж чорних металів, по 200 г міді і алюмінію, всього 1 г срібла і не більше 10 міліграма золота. Тобто в основному йдеться про матеріали, вторинне використання яких неможливе, і тому вартість переробки значно перевищує вартість отримуваних металів.

Окрему проблему утилізації представляють кінескопи моніторів – із-за вмісту отруйних речовин їх не можна навіть розібрати – як і ядерні відходи, їх слід лише ховати в спеціальних могильниках. Розвинені країни задихаються від електронного сміття, оскільки одних лише неробочих кінескопів і моніторів в них скупчилося більше 100 млн. Те ж саме відноситься і до батарей живлення і акумуляторів, що містять такі метали, як свинець, кадмій, цинк і нікель.

У Японії, США і країнах ЄС останніми роками активно розвивається напрям утилізації старих комп'ютерів. У США компанія Dell почала переробку застарілої техніки восени 2002 р. У Німеччині щорік викидається 2,2 млн

комп'ютерів, принтерів і моніторів. Середньостатистичний німець повністю або частково оновлює свій ПК кожні 2–3 роки і вимушений платити за утилізацію старого (залежно від моделі) від 15 до 30 євро.

В наш час німецький концерн Siemens і японська компанія Fujitsu мають намір створити спільне підприємство по утилізації персональних комп'ютерів – вони планують побудувати заводи в п'яти містах Німеччини. Є в намірах також створення аналогічних виробництв у Великобританії, Франції і Скандинавських країнах. На думку ініціаторів проекту, централізована переробка ПК поліпшить екологічну обстановку і сприятиме збільшенню попиту на нові моделі.

У Японії Sony організувала службу по утилізації застарілих персональних комп'ютерів своєї марки. Компанія створила спеціальну мережу по їх збору і доставці, а вартість переробки одного ПК складає 4–5 тис. ієн (\$40).

У нашій державі, на відміну від індустріально розвинених країн, проблема переробки електронного сміття практично пущена на самоплив – утилізація застарілої техніки є турботою виключно її власників, тому найчастіше устаткування закінчує свої дні на звалищі.

Згідно із законом для списання і утилізації ПК і офісного устаткування юридична особа повинна звернутися до одного із спеціалізованих підприємств, які займаються виробництвом, ремонтом і обслуговуванням персональної техніки, з проханням про проведення експертизи, і отримати висновок про те, що дана модель морально застаріла, знята з виробництва і ремонту не підлягає. Лише після цього можна укласти договір з відповідною організацією про утилізацію відходів.

Ліцензію на утилізацію відходів, що містять дорогоцінні метали, видає Міністерство фінансів, кольорові і чорні – Міністерство промислової політики, а полімери і отруйні матеріали – Міністерство екології і природних ресурсів. Але, як ми вже з'ясували, персональні комп'ютери містять всі ці речовини. Тому в їх відношенні діє такий порядок: устаткування необхідно здати підприємству, що має дозвіл на роботу з дорогоцінними металами. Там здійснюються розбирання техніки і сортування матеріалів, які потім передаються компаніям, на законних підставах що займається переробкою певного вигляду відходів.

Утилізація застарілої офісної техніки вимагає грошей, проте такі витрати неминучі. У силах же законодавчих органів зробити цей процес по можливості простим і ефективним. При розумній організації переробка "електронного сміття" може стати економічно вигідною діяльністю і сприяти створенню додаткових робочих місць.

ВИСНОВКИ

У результаті виконання даної дипломної роботи було досліджено існуючі методи автоматизації роботи підприємств, спроектовано та розроблено інформаційну систему моніторингу стану агротехнічного підприємства.

Для досягнення цієї мети були вирішені наступні задачі:

- досліджено існуючі програмні продукти;
- сформульовані вимоги до ІС;
- розроблені концепції системи, та проведено вибір оптимального варіанту;
- розроблені проектні рішення по системі з різних видів забезпечення, передбачених етапами ескізного та технічного проектів, зокрема загальносистемні рішення, рішення з організаційного забезпечення, рішення з інформаційного забезпечення, рішення з технічного забезпечення, рішення з програмного забезпечення.
- розроблена робоча документація по ІС;

Отримані результати дозволяють зробити висновок, що у разі успішної реалізації та впровадження ІС моніторингу стану, підвищиться ефективність керування виробництвом, швидкість та якість реагування та розв'язання існуючих проблем, що приведе до загального підвищення ефективності роботи підприємства. Також це дозволить підвищити рівень організації і досвіду робітників в використанні інформаційних технологій, що підготує умови для наступного етапу інформатизації підприємства.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Буренок, І. П. (2012). Методи експертних оцінок у економічних дослідженнях. Київ: КНЕУ.
2. Тихомиров, Ю. В. (2009). Математичні методи експертних оцінок та прийняття рішень. Москва: МГТУ.
3. Шендерович, А. І. (2011). Методи оцінки та прогнозування в економіці: теорія і практика. Київ: Лібра.
4. Гришко, В. І. (2014). Методи прийняття рішень в умовах невизначеності: теорія і практика. Львів: Світ.
5. Карпенко, В. М. (2007). Моделювання та експертні оцінки в системах прийняття рішень. Харків: ХНУ.
6. Keeney, R. L., & Raiffa, H. (1993). *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*. Cambridge University Press.
7. von Neumann, J., & Morgenstern, O. (1944). *Theory of Games and Economic Behavior*. Princeton University Press.
8. Saaty, T. L. (1980). *The Analytic Hierarchy Process: Planning, Priority Setting, Resource Allocation*. McGraw-Hill.
9. Zadeh, L. A. (1975). The Concept of a Linguistic Variable and Its Application to Approximate Reasoning – I. *Information Sciences*.
10. Dubois, D., & Prade, H. (1980). *Fuzzy Sets and Systems: Theory and Applications*. Academic Press.
11. Ruspini, E. H. (1991). *Fuzzy Sets: Theories and Applications*. Kluwer Academic Publishers.
12. Mendel, J. M. (1995). *Fuzzy Logic Systems for Engineering: A Tutorial*. Prentice Hall.
13. Yager, R. R., & Filev, D. P. (1994). *Essentials of Fuzzy Modeling and Control*. Wiley-Interscience.
14. Mendel, J. M., & John, R. L. (2001). *Fuzzy Sets, Fuzzy Logic, and Fuzzy Systems: Selected Papers by Lotfi A. Zadeh*. World Scientific.
15. Russell, S., & Norvig, P. (2021). *Artificial Intelligence: A Modern Approach*. Prentice Hall.

16. Shneiderman, B., Plaisant, C., Cohen, M., & Jacobs, S. (2016). Designing the User Interface: Strategies for Effective Human-Computer Interaction. Pearson.

ТЕХНІЧНЕ ЗАВДАННЯ

Вступ

Назва програмного забезпечення – інформаційна система моніторингу агротехнічного підприємства.

Область застосування: Агротехнічне підприємство.

1. Підстави для розробки

Завдання на дипломне проектування видане кафедрою ІУСТ згідно заявці агротехнічного підприємства – створення централізованої системи та автоматизованого пункту збору та обробки інформації про стан підприємства.

2. Призначення розробки

2.1 Функціональне призначення

Оптимізація операцій, що пов'язані зі збором інформацій про стан та проблеми підприємства, обробки цієї інформації, створення на підставі обробки відповідних звітів та завдань, контроль усього процесу їх виконання, централізація відомостей про подібні події, можливість зберігати інформацію.

2.2 Експлуатаційне призначення

Збір, аналіз даних про стан підприємства, дані про проблеми, що виникають під час його функціонування, їх систематизація та автоматизація роботи з усунення визначених проблем.

Оперативне оновлення, зберігання інформації, забезпечення зручного отримання її користувачем.

Розроблене програмне забезпечення дозволить працівникам підприємства подавати заявки про наявні проблеми у електронному форматі, переглядати наявні завдання, спостерігати за процесом усунення тих чи інших проблем. Допоможе керівництву контролювати стан підприємства.

Підвищить ефективність роботи підприємства, подовжить терміни експлуатації ОС, пришвидшить процес обробки службових записів, запобігатиме ушкодженню чи втраті інформації про наявні проблеми.

3. Вимоги до програмного продукту

3.1 Вимоги до функціональних характеристик

Програмне забезпечення є інформаційною системою для контролю та моніторингу стану підприємства. Програмний продукт повинен працювати в операційній системі Windows, Linux, Mac.

3.1.1 Вимоги до складу виконуваних функцій

Розглянута система будується навколо бази даних.

Програма повинна надати користувачам можливість виконання перерахованих нижче функцій:

1) Функції адміністрування БД:

- Створення облікового запису диспетчера;
- Авторизація диспетчера.
- Створення облікового запису працівника;
- Авторизація працівника.

З облікового запису працівника будуть доступні наступні дії:

2) Функція додавання нового завдання, проблеми;

3) Функція швидкого пошуку за такими критеріями:

- Назвою проблеми;
- Категорією проблеми.

4) Функція поширеного пошуку за всіма властивостями проблеми перерахованими вище.

5) Функція сортування проблем за назвою, категорією, датою додавання;

6) Функція перегляду повної інформації про проблему;

З облікового запису диспетчера будуть доступні також наступні дії:

7) Функція видалення завдання;

8) Функція редагування інформації про проблему;

9) Функція створення звітів;

10) Функція порівняння проблем, які відносяться до однієї категорії.

3.1.2 Вимоги до організації вхідних та вихідних даних

Вхідні дані повинні бути представлені у вигляді таблиць в базі даних. Вхідні дані вводить працівник підприємства, що подає заявку через спеціальну

електронну форму, після чого програмний продукт автоматично структурує заявку та заносить в базу даних.

Вхідні дані для заявки повинні бути наступними:

- a) Дата подання заявки;
- b) Назва проблеми;
- c) Зміст проблеми;
- d) Категорія(тип проблеми);
- e) Id працівника.

Вихідні дані представляють собою динамічно сформовані таблиці з інформацією про проблеми.

На підставі запису поданої таблиці диспетчер може створювати завдання, що зберігаються у таблиці завдань, подана таблиця має наступні поля:

- Назва;
- Ідентифікаційний номер;
- Ступінь пріоритетності;
- Стан;
- Відповідальний;
- Виконавець.

3.1.3 Вимоги до часових характеристик

Вимоги до часових характеристик програми не висуваються

3.2 Вимоги до надійності

3.2.1 Вимоги для забезпечення надійного функціонування програми

Надійне функціонування програми має бути забезпечено виконанням сукупності організаційно-технічних заходів, перелік яких подано нижче:

- 1) використанням ліцензійного програмного забезпечення;
- 2) перевіркою програмних засобів на наявність комп'ютерних вірусів;
- 3) резервне копіювання даних.

3.2.2 Відмови через некоректні дії оператора

Відмови програми можливі згодом некоректних дій оператора (користувача) при взаємодії з ПЗ. Програма повинна виконувати перевірку введених даних. У випадки, коли введені дані не відповідають вимогам,

програма повинна вивести повідомлення про помилку і запропонувати відредагувати вхідні дані.

3.4 Умови експлуатації

Програмний продукт повинен бути розрахований на роботу в діалоговому режимі з користувачем, який не є професійним програмістом. Користувач повинен вивчити інструкцію по роботі з програмою.

3.5 Вимоги до складу і параметрів технічних засобів

Даний програмний продукт вимагає технічні засоби із параметрами не менше наступних:

- Процесор: не менше Pentium III 630 MHz;
- Оперативна пам'ять: не менше 512 Mb;
- Відео пам'ять: не менше 64 Mb;
- Місце на жорсткому диску: не менше 100 Mb;
- Маніпулятори: миша, клавіатура.

3.5 Вимоги до інформаційної та програмної сумісності

Програма розрахована для роботи в операційних системах Windows, Linux, MacOS з встановленою Java 7.

3.6 Вимоги до маркування та упаковки

Маркування та упаковка проектованого програмного продукту повинні відповідати всім стандартним вимогам, що застосовуються до маркування, варіантам і способам упаковки.

3.7 Вимоги до транспортування і зберігання

Для даного програмного продукту повинні бути дотримані всі умови транспортування, місця зберігання, умови складування, терміни зберігання в різних умовах, які відповідають всім стандартам і вимогам програмного забезпечення по даному пункту.

3.8 Вимоги до програмної документації

3.8.1 Попередній склад програмної документації

Склад програмної документації повинен включати в себе:

- Технічне завдання;
- Текст програми;

- Програму і методики випробувань;
- Керівництво користувача;
- Опис програми.

Програма та методики випробувань потрібні, щоб показати замовнику, що розроблена виконавцем програма відповідає вимогам погодженого і затвердженого технічного завдання. Після проведення спільних (приймально-здавальних) випробувань замовник і виконавець підпишуть акт завершення роботи. Тим самим, робота буде закрита, умови договору виконані.

4. Техніко-економічні показники

Даний проект спрямований підвищити ефективність роботи підприємства за рахунок оптимізації системи обробки заявок про присутні проблеми.

5. Стадії та етапи розробки

Стадії та етапи розробки відповідають календарному плану, затвердженому кафедрою ІУСТ.

6. Порядок контролю і приймання

Для контролю та приймання повинен бути надано опис програми, а також програма і методика випробувань.

Порядок контролю і приймання даної розробки виконується представником замовника за присутності представника розробника згідно з програмою та методикою випробувань.

Якщо програма не пройшла випробування, виконавець зобов'язаний виправити помилки і недоліки в строк, що не більше ніж за 1 місяць з дня випробувань.

За результатами прийому складається акт, який підписується представником замовника і представником розробника і затверджується керівниками організації-замовника та організації-розробника.

У разі знаходження помилок під час прийому програмного виробу складається акт про знайдені помилки, який підписується представниками

замовника і розробника і затверджується керівниками організації - замовника та організації - розробника. Розробник повинен в перебігу не більше як 1-го місяця виправити зазначені помилки, і оповістити замовника про повторне проведення перевірки, не пізніше ніж за 2 тижні до початку прийому програмного продукту.

ЗАГАЛЬНИЙ ОПИС ІНФОРМАЦІЙНОЇ СИСТЕМИ МОНІТОРИНГУ СТАНУ ПІДПРИЄМСТВА

1. Загальні відомості

Поточна розробка є "інформаційна системи моніторингу стану агротехнічного підприємства", надалі ІС.

Для функціонування ІС потрібно наступне програмне забезпечення:

Windows XP/7/8;

Glassfish;

MySQL 5.

Програмне забезпечення ІС представляє собою дистрибутив з .ear файлу, бібліотек, шаблонів і т.п., а також схеми таблиць бази даних.

Програмне забезпечення ІС реалізоване мовою програмування Java з використанням середовища розробки Netbeans IDE v 8.0, програмне забезпечення серверної частини реалізовано у виді об'єктів бази даних (збережених процедур, тригерів і т.п.) з використанням SQL-діалекту MySQL.

2. Функціональне призначення

ІС являє собою інструмент керування документообігу на підприємстві. Основними функціями системи є:

- Ведення переліку існуючих проблем;
- Можливість створення завдання на підставі заявки про проблему;
- Облік завдань та інформації про їхній статус;
- Формування звітів;
- Можливість перепризначення виконавців, якщо у працівника є підлеглі.

3. Опис логічної структури

При розробці програмного забезпечення ІС використано методи об'єктно-орієнтованого проектування та розробки. Основною структурною одиницею ПЗ є клас, діаграма класів приведена в проектній документації. Для побудови ІС

використана архітектура «клієнт-сервер», основними частинами програмного забезпечення є клієнтська та серверна частина, як правило, розташовані на різних комп'ютерах у мережі підприємства. Клієнтською частиною є веб оглядач який має бути встановлений на комп'ютерах користувачів системи. Система виконує функції збереження, підтримки цілісності даних їхньої основної обробки шляхом використання об'єктів БД при доступі до даних.

4. Використовувані технічні засоби

Для функціонування програмного забезпечення ІС використовуються:

- робочі місця користувачів, що представляють собою персональні комп'ютери(ПК) з мінімальною конфігурацією класу Intel Pentium D820/1Гб/120 із встановленим веб оглядачем; мінімальний обсяг вільного місця на твердому диску 10Гб;
- сервер SmartServer SE100R-2U;
- локальна мережа для надання з робочих місць користувачів доступу до даних, що знаходиться в БД сервера, із пропускною здатністю 100Мбіт/с з використанням комутатора Allied Telesyn AT-8024 (24 порта).

5. Виклик і завантаження

Виклик і завантаження ІС виробляється з клієнтського робочого місця запуском веб оглядача та введенням адреси системи в мережі, після чого відбувається безпосередньо вхід у головне вікно системи з запитом імені користувача і пароля.

Завантаження ІС проходить як копіювання jar-архіву до Java серверу.

ІНСТРУКЦІЯ КОРИСТУВАЧА ІНФОРМАЦІЙНОЇ СИСТЕМИ

1. Призначення програми

Програмне забезпечення інформаційної системи моніторингу стану агротехнічного підприємства призначено для автоматизації роботи з обробки інформації про ефективність роботи підприємства, збору заявок про проблеми, що виникають на підприємстві та всього процесу їх вирішення. Функції, що автоматизуються системою, пов'язані з організацією робочого часу працівників, усуненням проблем на підприємстві, отриманням завдань та допоміжним документообігом, як то звіт та статистичні показники.

Задачею ІС, є автоматизація наступних функцій користувача:

- Ведення переліку існуючих проблем;
- Можливість створення завдання на підставі заявки про проблему;
- Облік завдань та інформації про їхній статус;
- Формування звітів;
- Можливість перепризначення виконавців, якщо у працівника є підлегли.

Для створення інформаційної системи необхідно розробити єдину базу даних для збереження інформації про проблеми, завдання та їхні статуси. Також повинно бути передбачене програмне забезпечення для адміністрування системи. Оскільки ПЗ функціонує в мережі підприємства, обмін даними будується через єдину інформаційну базу.

2. Умови виконання програми

Для виконання програми необхідна наявність на підприємстві наступних апаратних і програмних засобів:

- робочі місця користувачів, що представляють собою персональні комп'ютери(ПК) з мінімальною конфігурацією класу Intel Pentium D820/1Гб/120 із встановленим веб оглядачем; мінімальний обсяг вільного місця на твердому диску 10Гб;

- сервера, що може бути реалізований на базі ПК із мінімальною конфігурацією Intel Core 2 Duo E6750 /2Гб/200, із встановленої ОС SUSE Linux 10.1 і СКБД MySQL;

- локальної мережі для надання з робочих місць користувачів доступу до даних, що знаходиться в БД сервера, із пропускнуою здатністю 100 Мбіт/с.

3. Виконання програми

Виконання програми на кожному з робочих місць починається запуском веб-оглядача та вводом адреси системи у мережі. Безпосередньо після цього Користувач системи повинний пройти авторизацію – ввести в діалоговому вікні, що з'явилося, користувальницьке ім'я і пароль. Пароль є унікальною комбінацією буквено-цифрових символів, відомих тільки користувачу. Якщо ви забули пароль – звернетесь до системного адміністратора.

Після успішного входу в систему користувач бачить екран з двома списками: перший – список усіх завдань які існують у системі, другий – список завдань на підставі створених користувачем заявок.

При переході за посиланням на перегляд проблеми, диспетчер може створити нове завдання на підставі поточної заяви. Якщо завдання вже існує – користувач може продивитися статус його виконання.

З цього ж екрану користувач може перейти:

- на екран де може змінити свої налаштування (кількість секунд для оновлення екрану, отримувати листа при отриманні завдання, звіту або повідомлення), після заповнення форми користувач повинен натиснути на кнопку "Save" щоб дані були збереженні у базі;

- на екран статистики, де відображується статистика за обраний період.

Після вибору початкового періоду та кінцевого користувач бачить кількість виконаних та отриманих завдань, а також їх перелік. Якщо у користувача є підлеглі, то він може переглянути їхню статистику теж, вибором працівника у випадаючому списку;

- на екран звітів, де бачить свої звіти у вигляді посилань на сторінку з текстом звіту, або якщо у користувача є підлеглі то відіслані ними звіти теж.

Користувач може створити новий звіт натиском на кнопку "New Report";

Користувач може вийти з системи натиском на відповідне посилання з екрану списку звітів, він може завершити дію веб оглядача, але у цілях безпеки рекомендується виходити з системи.

4. Повідомлення користувачу

При введенні некоректних даних над полями уведення виводиться діалогове вікно з описом типу і місця виникнення помилки.

При неправильному введенні чи пароля імені користувача виводиться повідомлення "Wrong login or password".

ТЕКСТИ ПРОГРАМНИХ МОДУЛІВ СИСТЕМИ

Веб дескриптор (web.xml)

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app version="2.4"
  xmlns="http://java.sun.com/xml/ns/j2ee"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee
http://java.sun.com/xml/ns/j2ee/web-app_2_4.xsd" >
  <context-param>
    <param-name>javax.servlet.jsp.jstl.fmt.localizationContext</param-name>
    <param-value>MessageResources</param-value>
  </context-param>
  <servlet>
    <servlet-name>action</servlet-name>
    <servlet-class>org.apache.struts.action.ActionServlet</servlet-class>
    <init-param>
      <param-name>config</param-name>
      <param-value>/WEB-INF/struts-config.xml</param-value>
    </init-param>
  </servlet>
  <servlet-mapping>
    <servlet-name>action</servlet-name>
    <url-pattern>*.html</url-pattern>
  </servlet-mapping>

  <filter>
    <filter-name>UrlRewriteFilter</filter-name>
    <filter-class>org.tuckey.web.filters.urlrewrite.UrlRewriteFilter</filter-
class>
    <init-param>
      <param-name>logLevel</param-name>
      <param-value>WARNING</param-value>
    </init-param>
  </filter>
  <filter-mapping>
    <filter-name>UrlRewriteFilter</filter-name>
    <url-pattern>/*</url-pattern>
  </filter-mapping>

  <filter>
    <filter-name>CharEncodingFilter</filter-name>
    <filter-class>org.dionis.todo.web.utils.CharEncodingFilter</filter-class>
  </filter>
  <filter-mapping>
    <filter-name>CharEncodingFilter</filter-name>
    <url-pattern>/*</url-pattern>
  </filter-mapping>

  <jsp-config>
    <jsp-property-group>
      <url-pattern>*.jspx</url-pattern>
      <page-encoding>utf-8</page-encoding>
      <is-xml>true</is-xml>
    </jsp-property-group>
  </jsp-config>

  <session-config>
    <session-timeout>30</session-timeout>
  </session-config>
```

```

<welcome-file-list>
    <welcome-file>index.jsp</welcome-file>
    <welcome-file>admin/index.jsp</welcome-file>
</welcome-file-list>

<error-page>
    <error-code>404</error-code>
    <location>/nopage.html</location>
</error-page>

<security-role>
    <description>Authorized to access to admin zone</description>
    <role-name>todo</role-name>
</security-role>
<security-constraint>
    <display-name>Security</display-name>
    <web-resource-collection>
        <web-resource-name>Admin Zone</web-resource-name>
        <url-pattern>/admin/*</url-pattern>
    </web-resource-collection>
    <auth-constraint>
        <role-name>todo</role-name>
    </auth-constraint>
</security-constraint>
<login-config>
    <auth-method>BASIC</auth-method>
</login-config>
</web-app>

```

Клас для роботи з електронною поштою (Mailer.java)

```
package org.agro.todo.ejb.utils;
```

```

import javax.mail.Transport;
import javax.mail.Session;
import javax.mail.Message;
import javax.mail.Address;
import javax.mail.internet.MimeMessage;
import javax.mail.internet.InternetAddress;
import java.util.Properties;
import java.util.Date;

```

```

public class Mailer {
    private Transport transport = null;
    private static Session mailConnection;

    public Mailer() {
        try {
            Properties props = new Properties();
            props.put("mail.smtp.auth", true);
            mailConnection = Session.getInstance(props, null);
            transport.connect("localhost", "", "");
        } catch (Exception e) {
            transport = null;
            e.printStackTrace();
        }
    }

    public boolean mail(String emailTo, String nameTo, String emailFrom, String
nameFrom, String content, String subject, boolean plain) {
        if (null == transport && !initTransport()) return false;
        try {
            Message msg = new MimeMessage(mailConnection);
            Address addressTo = new InternetAddress(emailTo, nameTo);
            Address addressFrom = new InternetAddress(emailFrom, nameFrom);

```

```

        msg.setContent(content, plain ? "text/plain" : "text/html");
        msg.setFrom(addressFrom);
        msg.setRecipient(Message.RecipientType.TO, addressTo);
        msg.setSubject(subject);
        msg.setSentDate(new Date());
        msg.saveChanges();
        transport.sendMessage(msg, msg.getAllRecipients());
        return true;
    } catch (Exception e) {
        e.printStackTrace();
        return false;
    }
}

private boolean initTransport() {
    try {
        Properties props = new Properties();
        props.put("mail.smtp.auth", "true");
        mailConnection = Session.getInstance(props, null);
        transport = mailConnection.getTransport("smtp");
        transport.connect("localhost", "", "");
        return true;
    } catch (Exception e) {
        e.printStackTrace();
        transport = null;
        return false;
    }
}

@Override
protected void finalize() throws Throwable {
    transport.close();
    super.finalize();
}
}

```

SessionBean клас реалізує методи для роботи з заявками (TaskSessionBean.java)

```

package org.agro.todo.ejb.session;

import org.agro.todo.ejb.entity.Task;
import org.agro.todo.ejb.entity.User;
import org.agro.todo.ejb.entity.Action;
import org.agro.todo.ejb.exception.NotFoundException;
import org.agro.todo.ejb.exception.TODOException;
import javax.ejb.Stateless;
import javax.persistence.PersistenceContext;
import javax.persistence.EntityManager;
import java.util.List;

@Stateless
public class TaskSessionBean implements TaskSession {
    @PersistenceContext(unitName = "todo")
    EntityManager em;

    public Task createTask(Task t) {
        em.persist(t);
        return t;
    }

    public Task editTask(Task t) throws NotFoundException, TODOException {
        Task tt = em.find(Task.class, t.getId());
        if (tt == null) throw new NotFoundException("Task with this id not found");
        if (searchTasks(tt).isEmpty()) throw new TODOException("Task have childrens");
    }
}

```

```

        tt = t;
        return tt;
    }

    public void removeTask(long id) throws NotFoundException, TodoException {
        Task tt = em.find(Task.class, id);
        if (tt == null) throw new NotFoundException("Task with this id not
found");
        if (searchTasks(tt).isEmpty()) throw new TodoException("Task have
childrens");
        em.remove(tt);
    }

    public Task getTaskById(long id) {
        return em.find(Task.class, id);
    }

    protected List<Task> searchRecTasks(Task parent) {
        List<Task> l = em.createQuery("select t from Task t where t.parent =
:parent").setParameter("parent", parent).getResultList();
        if (!l.isEmpty())
            for (Task task : l) {
                l.addAll(searchRecTasks(task));
            }
        return l;
    }

    public List<Task> searchTasks(Task parent) {
        return searchRecTasks(parent);
    }

    public List<Task> getToUserTasks(User u) {
        return em.createQuery("select t from Task t where t.reciever = :u and
t.state <> :state").setParameter("state",
Task.State.DONE_APPROVED).setParameter("u", u).getResultList();
    }

    public List<Task> getFromUserTasks(User u) {
        if (u.getSettings().isShowDoneNoteApproved()) {
            return em.createQuery("select t from Task t where t.creator = :u and
t.state <> :state").setParameter("state",
Task.State.DONE_APPROVED).setParameter("u", u).getResultList();
        } else {
            return em.createQuery("select t from Task t where t.creator = :u and
t.state <> :state1 and t.state <> :state2").setParameter("state1",
Task.State.DONE_APPROVED).setParameter("state2",
Task.State.DONE).setParameter("u", u).getResultList();
        }
    }

    public List<User> getPermittedUsers(User u) {
        return em.createQuery("select u from User u where u.leader = :l or
u.department = :d").setParameter("l", u.getLeader()).setParameter("d",
u.getDepartment()).getResultList();
    }

    public List<User> getSubUsers(User u) {
        return em.createQuery("select u from User u where u.leader =
:l").setParameter("l", u).getResultList();
    }

    public Action addAction(Action a) {
        em.persist(a);
        return a;
    }

```

```

    }
}

```

jspx сторінка з формою авторизації користувача (login.jsx)

```

<jsp:root
    xmlns:jsp="http://java.sun.com/JSP/Page"
    xmlns:html="http://struts.apache.org/tags-html-el"
    xmlns:fmt="http://java.sun.com/jsp/jstl/fmt"
    xmlns:c="http://java.sun.com/jstl/core_rt"
    xmlns="http://www.w3.org/1999/xhtml" version="2.0">
<jsp:output omit-xml-declaration="true"
    doctype-public="-//W3C//DTD XHTML 1.0 Transitional//EN"
    doctype-system="http://www.w3c.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"
    doctype-root-element="html"/>
<jsp:directive.page contentType="text/html"/>
<html:html xhtml="true">
<head>
    <title>Login</title>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
    <link rel="stylesheet" type="text/css" href="/css/main.css"/>
</head>
<body>
    <div class="spacer"><jsp:text/></div>
    <h1 class="errors" align="center"><c:out value="{errors}"/></h1>
    <html:form action="/login" method="post">
        <html:hidden property="act" value="login"/>
        <table cellpadding="5" cellspacing="5">
            <tr>
                <td><fmt:message key="loginForm.login"/> </td>
                <td><html:text property="login"/></td>
            </tr>
            <tr>
                <td> </td>
                <td class="errors"><html:errors property="login"
name="validatorErrors"/></td>
            </tr>
            <tr>
                <td><fmt:message key="loginForm.password"/> </td>
                <td><html:password property="password"/></td>
            </tr>
            <tr>
                <td> </td>
                <td class="errors"><html:errors property="password"
name="validatorErrors"/></td>
            </tr>
            <tr>
                <td> </td>
                <td><html:submit/></td>
            </tr>
        </table>
    </html:form>
</body>
</html:html>
</jsp:root>

```

Action клас реалізуючий авторизацію користувача (LoginAction.java)

```

package org. agro.todo.web.action;

import org.apache.struts.action.*;
import org.apache.struts.validator.DynaValidatorForm;
import org. agro.todo.ejb.session.UserSession;
import org. agro.todo.ejb.session.UserSessionBean;
import org. agro.todo.ejb.entity.User;

```

```

import org. jgsoft.todo.web.utils.LookupUtil;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class LoginAction extends Action {
    public ActionForward execute(ActionMapping mapping, ActionForm aForm,
    HttpServletRequest request, HttpServletResponse response) throws Exception {
        ActionMessages validatorErrors = new ActionMessages();
        request.setAttribute("validatorErrors", validatorErrors);
        DynaValidatorForm form = (DynaValidatorForm) aForm;
        String errors = "";
        try {
            if ("login".equals((String) form.get("act"))) {
                validatorErrors.add(form.validate(mapping, request));
                if (validatorErrors.isEmpty()) {
                    UserSession uSession =
LookupUtil.getSessionBean(UserSessionBean.class);
                    User user =
uSession.getUserByLoginAndPassword((String) form.get("login"),
                        (String) form.get("password"));
                    request.getSession().setAttribute("loggedUser", user);
                    return mapping.findForward("tasksListAction");
                }
            }
        } catch(Exception te) {
            errors = te.getMessage();
        }
        request.setAttribute("errors", errors);
        return mapping.findForward("login");
    }
}

```

Фільтер переводу символів у UTF-8 (CharEncodingFilter.java)

```

package org. agro.todo.web.utils;

import javax.servlet.*;
import java.io.IOException;

public class CharEncodingFilter implements Filter {
    private FilterConfig filterConfig = null;
    private String encoding = null;

    public void init(FilterConfig filterConfig) throws ServletException {
        this.filterConfig = filterConfig;
        this.encoding = this.filterConfig.getInitParameter("encoding");
    }

    public void destroy() {
        this.filterConfig = null;
    }

    public void doFilter(ServletRequest request, ServletResponse response,
        FilterChain chain) throws IOException, ServletException
    {
        if (encoding != null) {
            request.setCharacterEncoding(encoding);
            response.setCharacterEncoding(encoding);
        } else {
            request.setCharacterEncoding("UTF-8");
            response.setCharacterEncoding("UTF-8");
        }
        chain.doFilter(request, response);
    }
}

```