

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
Національний авіаційний університет

**ІНЖЕНЕРІЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ**

Лабораторний практикум  
для студентів напрямку підготовки  
6.050102 «Комп'ютерна інженерія»

Київ 2015

УДК 004.415/.416(075.5)

ББК 3973.20-018.2я7

I 621

Укладачі: *Д. П. Кучеров, О.Є. Ніколаєнко, К.А. Мацуєва*

Рецензент: *М.М. Гузій*

*Затверджено методично-редакційною радою Національного авіаційного університету (протокол № \_\_\_/15 від \_\_. \_\_.2015 р.).*

I621      **Інженерія програмного забезпечення** : Лабораторний практикум / укладачі Д. П. Кучеров, О.Є. Ніколаєнко, К.А. Мацуєва. – К. : НАУ, 2015. – 148 с.

Лабораторний практикум призначений для закріплення теоретичних відомостей та набуття практичних навичок при створенні програмного забезпечення для проектів інформаційних систем, містить короткі теоретичні відомості, завдання для виконання лабораторних робіт, запитання та завдання для самоперевірки та розрахований на 2-х семестровий курс навчання.

Рекомендується студентам 2-4 курсів напряму підготовки 6.050102 «Комп'ютерна інженерія».

## ЗМІСТ

ВСТУП .....	
Частина 1. Проектування інформаційних систем	
Модуль I. Моделі життєвого циклу для розробки програмних систем	
Лабораторна робота 1.1. Опис і аналіз інформаційної системи ...	
Лабораторна робота 1.2. Розробка вимог до інформаційної системи .....	
Лабораторна робота 1.3. Графік робіт проекту і перегляд критичного шляху .....	
Лабораторна робота 1.4. Створення графіку робіт та ресурсів за допомогою діаграми Перт.....	
Лабораторна робота 1.5. Рішення проблеми перевантаження ресурсів .....	
Модуль II. Проектування програмного забезпечення .....	
Лабораторна робота 2.1. Структурний підхід опису процесів в нотатії IDEF0 .....	
Лабораторна робота 2.2. Побудова моделей опису процесів при структурному підході. Нотація IDEF3 .....	
Лабораторна робота 2.3. Аналіз вимог і побудова моделей опису процесів при структурному підході. Нотація DFD .....	
Частина 2. Розробка програмного забезпечення	
Модуль III. Формальні специфікації й верифікація програм. Методи перевірки та тестування програм та систем .....	
Лабораторна робота 3.1. Побудова діаграми варіантів використання в середовищі IBM Rational Rose .....	
Лабораторна робота 3.2. Розробка діаграми класів в середовищі IBM Rational Rose .....	
Лабораторна робота 3.3. Діаграми кооперації та послідовностей в середовищі IBM Rational Rose .....	
Лабораторна робота 3.4. Діаграма станів в середовищі IBM Rational Rose .....	
Лабораторна робота 3.5. Діаграма діяльності в середовищі IBM Rational Rose .....	
Модуль IV. Реалізація об'єктно-орієнтованих програмних систем	

Лабораторна робота 4.1. Діаграма компонентів в середовищі IBM Rational Rose .....	
Лабораторна робота 4.2. Діаграма розгортання в середовищі IBM Rational Rose .....	
Лабораторна робота 4.3. Підготовка моделі для генерації програмного коду в середовищі IBM Rational Rose .....	
Лабораторна робота 4.4. Оцінка розміру та вартості проекту ....	
РЕКОМЕНДОВАНА ЛІТЕРАТУРА .....	
Додаток 1 .....	
Додаток 2 .....	

## ВСТУП

Лабораторний практикум призначений для засвоєння теоретичного матеріалу дисципліни «Інженерія програмного забезпечення», отримання й закріплення знань та навичок з розробки проектів зі створення програмного забезпечення для проектів інформаційних систем. Завдання лабораторного практикуму поширюються на програмні системи для управління базами даних, робототехнічними системами та комплексами, інтелектуальними системами промислового та побутового призначення, експертними системами. Основний акцент зроблено на вміння студента працювати з CASE засобами планування проектів та засобами розробки програмного забезпечення.

Головним завданням практикуму є оволодіння універсальними інструментами проектування програмного забезпечення інформаційних систем Microsoft Project, BPwin та IBM Rational Rose. Типові завдання лабораторних робіт передбачають наявність у студентів початкових знань про інформаційні системи, а саме їх призначення, склад і функціонування та самостійну роботу з індивідуальними комп'ютерними засобами, із застосуванням навчальної літератури та методичних рекомендацій. Усі лабораторні роботи супроводжуються достатньою кількістю прикладів із коментарями, що дозволяє самостійно вирішувати поставлені завдання за аналогією.

Рекомендований порядок підготовки та безпосереднє виконання завдань лабораторного практикуму такий: опрацювання лекційного матеріалу, ознайомлення з теоретичними відомостями, робота з пропонуєним програмним забезпеченням відповідно до індивідуального завдання, побудова програмного рішення, складання звіту з лабораторної роботи та захист роботи шляхом відповіді на контрольні запитання.

Лабораторний практикум складається з робіт однакової складності та умовно поділених на дві частини, які відповідають лабораторним роботам 3-го та 4-го семестрів і пропонуються до виконання студентами освітньо-кваліфікаційного рівня «Бакалавр».

При виконанні лабораторних робіт рекомендується суворо дотримуватися вимог правил техніки безпеки при роботі з електроприладами.

## **Частина 1. Проектування інформаційних систем**

### **Модуль I. Моделі життєвого циклу для розробки програмних систем**

#### **Лабораторна робота 1.1 ОПИС І АНАЛІЗ ІНФОРМАЦІЙНОЇ СИСТЕМИ**

**Мета:** вибрати тип інформаційної системи та спроектувати із застосуванням структурного моделювання.

**Завдання:** обрати тип інформаційної системи відповідно до індивідуального завдання, провести збір інформації про компанії, що розробляють та експлуатують аналогічні системи; з'ясувати проблемні питання, що виникають в ході експлуатації, сформулювати цілі розробки; зробити опис інформаційної системи.

#### **Основні теоретичні відомості**

Інформаційна система (ІС) — система обробки інформації із відповідними організаційними ресурсами (людськими, технічними, фінансовими і т.і.), які забезпечують і поширюють інформацію. В більш вузькому розумінні ІС складає з даних, програм й апаратного забезпечення, а також персоналу й організаційних заходів.

Інформаційна система призначена для своєчасного забезпечення певних людей конкретних інформаційними потребами у рамках певної предметної області. Результатом функціонування ІС є інформаційна продукція — документи, інформаційні масиви, бази даних й інформаційні послуги.

За ступенем розподіленості ІС поділяють на настільні або локальні ІС, у яких усі компоненти (БД, СУБД, клієнтські додатки) перебувають на одному комп'ютері та розподілені ІС, які ділять на файл-серверні ІС та клієнт-серверні ІС.

За ступенем автоматизації ІС бувають автоматизовані, які потребують участі персоналу; автоматичні, які не потребують участі персоналу; неавтоматизовані.

За характером обробки даних ІС поділяють на інформаційно-довідкові або інформаційно-пошукові та обробки даних або вирішуючі, підтримки прийняття рішень.

За сферою застосування: економічні, медичні, географічні.

По охопту завдань: особисті, групові, корпоративні.

Проектування ІС охоплює три основні області: проектування об'єктів даних, програм, екраних форм, звітів; врахування конкретного середовища або технології.

Проектування ІС завжди починається з визначення мети проекту, яка визначається рядом взаємопов'язаних завдань, що направлені на задоволення інформаційних потреб. До узагальнених показників ІС відносять необхідні функціональність та рівень її адаптивності до змінних умов функціонування; пропускну здатність; час реакції системи на запит; показники надійності, безпеки, зручності експлуатації та підтримки системи.

### **Порядок виконання роботи**

1. Перед началом роботи пройти інструктаж по техніці безпеки при роботі з електроприборами. Вивчити пропонований теоретичний матеріал згідно з наявним методичним матеріалом, ознайомитися з літературою [1-4].

2. Обрати тип організації для складання докладного опису ІС.

3. Визначити тип ІС, сформулювати її призначення, цілі, показники.

4. На підставі опису системи провести аналіз її реалізуємості. У ході аналізу відповісти на питання:

- які поточні проблеми існують і як нова система допоможе їх вирішити?

- які ризики понесе організація, якщо система не буде введена в експлуатацію?

- які CASE – засоби сприяють розробці ІС?

5. Оцінити доцільність та можливості реалізації проекту.

### **Оформлення результатів роботи**

Результати лабораторної роботи оформлюються у вигляді звіту обов'язковими елементами, якого є: титульний аркуш; формулювання мети та завдання на роботу; основна частина; висновки.

Основна частина роботи повинна містити етапи вирішення індивідуального завдання шляхом аналізу досягнутих результатів в

обраній предметній області. Зразок оформлення опису інформаційної системи наведений в Додатку 1.

Висновки повинні містити результати виконаної роботи.

### **Завдання на роботу**

Провести проектування ІС відповідно до пропонованого списку або узгодити вибір ІС з викладачем.

1. Інформаційно-довідкова система аеропорту.
2. Інформаційно-пошукова система для виконання перельотів.
3. Інформаційна система аеропорту.
4. Система керування польотами.
5. Касово-розрахункова система продажу авіаквітків.
6. Система розрахунку заробітної плати.
7. Система керування діями у назвичайних ситуаціях.
8. Система контролю здійснення перельотів.
9. Система керування діями пасажирів.
10. Система керування освітленням аеропорту.
11. Навчально-тренажерний комплекс авіадиспетчерів.

### **Запитання для самоперевірки**

1. Знайти різницю між поняттями система та комплекс.
2. Охарактеризуйте вплив організаційних факторів на систему.
3. Назвати функціональні компоненти систем та поясніть їх роль в системі.
4. Поясніть процес створення інформаційних систем.
5. Назвати суб'єктів розробки системи.
6. Які особливості ІС впливають на час розробки?
7. Обґрунтувати показники якісної ІС.

## **Лабораторна робота 1.2**

### **РОЗРОБКА ВИМОГ ДО ПРОГРАМНОЇ СИСТЕМИ**

**Мета:** вивчити підходи до складання проекту вимог до ІС, оформити технічне завдання на розробку програмного забезпечення.



**Завдання:** скласти проект вимог до інформаційної системи, оформити технічне завдання на розробку програмного забезпечення, оформити звіт з лабораторної роботи.

### **Основні теоретичні відомості**

Опис функціональних можливостей і обмежень, що накладаються на програмну систему, називають вимогами.

Вимоги до програмних систем виражають у вигляді текстових тверджень і графічних моделей. У класичному технічному підході сукупність вимог використовується на стадії проектування, а також в процесі перевірки ПЗ.

Процес розробки вимог включає чотири основних етапи:

- аналіз технічної реалізуємості системи;
- формування і аналіз вимог;
- специфікування вимог і створення відповідної документації;
- атестація цих вимог.

Етапу розробки вимог передують техніко-економічне обґрунтування, або концептуальна фаза аналізу проекту. Фаза розробки вимог може бути розбита на виявлення вимог (збір, розуміння, розгляд і з'ясування потреб зацікавлених осіб), аналіз (перевірка цілісності й закінченості), специфікація (документування вимог) і перевірка правильності.

Існує три основних підходи до формування вимог: метод, заснований на множині опорних точок зору; сценарії; етнографічний підхід.

Вимоги поділяються на користувальницькі та системні.

### **Порядок виконання роботи**

1. Вивчити пропонований теоретичний матеріал відповідно до літератури [1, 2, 8, 9].

2. Побудувати опорні точки зору на підставі методу VORD для формування і аналізу вимог. Результатом повинні з'явитися дві діаграми: *діаграма ідентифікації* точок зору і *діаграма ієрархії* точок зору.

3. Скласти інформаційну модель майбутньої системи, що включає в себе опис основних об'єктів системи і взаємодії між

ними. На підставі отриманої інформаційної моделі та діаграм ідентифікації точок зору, діаграма ієрархії точок зору сформувані вимоги користувача і системні вимоги.

4. Провести атестацію вимог, указати вибрані типи перевірок.

5. На підставі опису системи (результат виконання лабораторної роботи 1), інформаційної моделі, користувальницьких і системних вимог скласти технічне завдання (ТЗ) на створення програмного забезпечення. ТЗ повинне містити основні розділи, описані в ДЕРЖСТАНДАРТ 34.602-89.

6. Скласти звіт, що включає всі отримані рівні моделі, опис функціональних блоків, потоків даних, пам'яті і зовнішніх об'єктів.

### **Оформлення результатів роботи**

Результати лабораторної роботи оформлюються у вигляді звіту. Основна частина роботи повинна містити етапи вирішення індивідуального завдання шляхом аналізу досягнутих результатів в обраній предметній області. Зразок оформлення вимог до інформаційної системи наведений в Додатку 2.

Висновки повинні містити результати виконаної роботи.

### **Завдання на роботу**

Відповідно до обраної системи у відповідному підрозділі лабораторної роботи 1.1 розробити користувальницькі та системні функціональні й нефункціональні вимоги.

### **Запитання для самоперевірки**

1. Назвати відмінності між користувальницькими та системними вимогами.
2. Пояснити зміст функціональних вимог.
3. За якими ознаками класифікують нефункціональні вимоги?
4. Привести кількісні показники для нефункціональних вимог.
5. Які правила написання користувальницьких вимог існують?
6. В чому сутність методу формування вимог на основі сценаріїв?

7. Провести аналіз вимог за допомогою матриці оперативного контролю.

## **Лабораторна робота 1.3**

### **ГРАФІК РОБІТ ПРОЕКТУ І ПЕРЕГЛЯД КРИТИЧНОГО ШЛЯХУ**

**Мета:** побудова графіку робіт засобами Microsoft Project.

**Завдання:** побудувати графік виконання робіт із проектування ІС та визначити критичний шлях.

#### **Основні теоретичні відомості**

**Microsoft Project** (*MS Project*) – програма керування проектами, яка входить до пакету *Microsoft Office*.

Програма допомагає менеджеру проекту в розробці планів, розподілі ресурсів за завданням, відстеженні прогресу та аналізі обсягів робіт. Microsoft Project утворює розклад критичного шляху. Розклади можуть бути складені з урахуванням використовуваних ресурсів. Ланцюжок візуалізується в діаграмі Ганта.

MS Project оперує трьома сутностями – завдання, ресурси, календар і зв'язки між ними. За суттю – це база даних, яка має користувацький інтерфейс для створення та редагування сутностей з мінімальною автоматизацією.

На основі даних про завдання і ресурси Project дозволяє зробити різні подання з використанням фільтрів, угруповань, сортувань і т.п. Крім цього, обчислювати терміни початку і закінчення завдань із урахуванням доступності призначених ресурсів і зв'язків між завданнями за певним алгоритмом.

**Діаграма Ганта** – це тип стовпчастих діаграм (гістограм), який використовується для ілюстрації плану, графіка робіт з якого-небудь проекту. Є одним з методів планування проектів. Використовується в додатках по керуванню проектами.

За суттю, діаграма Ганта має вигляд смуг, орієнтованих вздовж осі часу. Кожна смуга на діаграмі відповідає за окреме завдання в складі проекту (вид роботи), її кінці – моменти початку і завершення роботи, її довжина – тривалість роботи. Вертикальною віссю діаграми слугує перелік завдань. Крім того, на діаграмі можуть бути відзначені сукупні завдання, відсотки завершення, показники послідовності і залежності робіт, мітки ключових моментів (віхи), мітка теперішнього моменту часу «Сьогодні» та ін.

## Порядок виконання роботи

1. Ознайомитися з літературою [1, 2, 5, 7, 8].

2. **Створити проект.** Здійснити запуск програми через меню «Пуск» панелі завдань, де обрати шлях Програми/Microsoft Office/Microsoft Project 2003.

3. **Настроїти базовий календар.** Встановити дати – початку, кінця та поточну через меню «Проект/Відомості про проект» та календаря поточного місяця. В головному меню програми вибрати «Сервіс/Параметри», де вибрати вкладку «Календар». На цій вкладці встановити параметри фінансового року та робочого тижня. Закрити вікно Параметри натисканням кнопки «ОК».

Індивідуальні календарі кожному працівнику (ресурсу) або групі ресурсів створюються на основі базового. За необхідності вказати програмі неробочі, святкові і скорочені передсвяткові дні в меню «Сервіс/Змінити робочий час». Створити новий базовий календар шляхом натискання на кнопку «Створити».

У новому календарі відзначимо святкові дні (наприклад, 8 березня, 1, 2, 9, 10 травня) як неробочі. Ознакою встановлення є зафарбовування сірим кольором дати неробочого дня. Зберегти проект в меню Файл/Зберегти утворивши файл \*.mpp.

4. **Створити графік робіт.** Ввести назви робіт в рядки поля «Назва завдання», «Розробка технічного завдання», «Тривалість». За необхідністю змінити дату початку завдання переміщаючи смужку-роботу на панелі діаграми. (рис. 1.1).

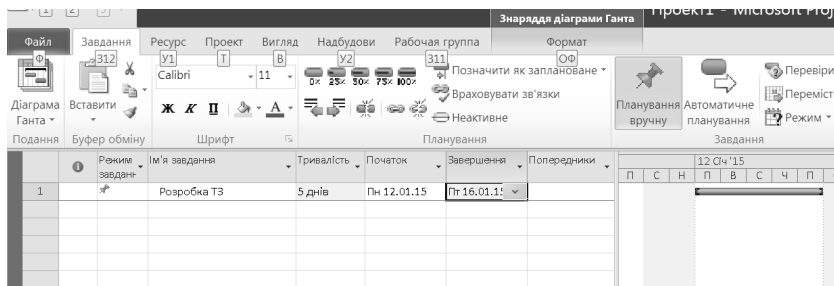


Рис. 1.1. Вікно зміни дати початку роботи

Створити контрольні точки «Початок робіт», для цього виділити відповідну комірку клацанням миши на першому рядку поля «Назва завдання», вибрати команду меню «Вставка/Нове

завдання». У вставлений порожній перший рядок таблиці вводимо «Початок робіт». Далі у полі «Тривалість» встановлюється тривалість роботи «0 днів» і натиснути клавішу «Enter». На панелі діаграми в першому рядку з'явиться чорний ромб з датою початку робіт, який відповідає створеній контрольній точці (рис. 1.2).

- Існує чотири типи можливих зв'язків між окремими роботами:
- закінчення-початок - наступна робота починається після закінчення попередньої;
  - початок-початок - роботи починаються одночасно;
  - закінчення-закінчення - роботи закінчуються одночасно;
  - початок - закінчення - одна робота не може закінчитися доти, поки інша не почнеться.



Рис. 1.2. Таблиця робіт з встановленими контрольними точками

Встановити зв'язок між роботами типу «закінчення-початок». Для цього виділити мишею назву завдання, натиснути і втримувати клавішу «Ctrl», далі виділити мишею назву другої роботи, що залежить від першої: обидві роботи виділені. На панелі інструментів «Стандартна» натиснути кнопку «скрепіти», зв'язок на діаграмі відображається стрілкою (рис. 1.3).

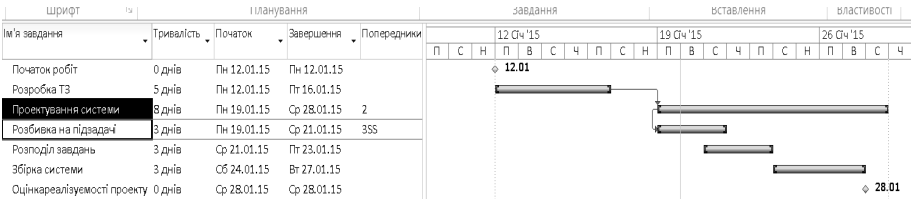


Рис. 1.3. Стрілка показує зв'язок типу закінчення-початок

Помилкові зв'язки видаляються шляхом натискання на кнопку розкріпити панелі інструментів «Стандартна». Інші типи зв'язків встановлюються шляхом подвійного клацання мишею по значку стрілки на діаграмі через вікно «Залежність завдань».

**5. Визначення критичного шляху.** Відформатувати текстові поля, наприклад, контрольні точки шляхом встановлення їх типу шрифту та кольору. Вибрати команду меню «Формат/Стилі тексту». На екрані з'явиться діалог «Стилі тексту». У списку «змінюваний елемент» вибрати «Критичні завдання», для них встановити стиль шрифту «курсив», розмір «10», колір вибрати «Червоний».

Відобразити на діаграмі критичний шлях. В меню «Формат» вибрати команду «Стилі відрізків» (можна також визвати команду подвійним клацанням мишею на вільному полі діаграми). На екрані з'явиться вікно «Стилі відрізків».

Встановити в таблиці новий рядок для критичних робіт. Виділити комірку першого рядка таблиці клацанням мишею в полі «Назва». Натиснути кнопку «Додати рядок» верхньої частини діалогу. В цьому полі ввести із клавіатури «Критичний шлях», у полі «Вид» встановити для нього колір, яким буде відображатися елемент діаграми, наприклад, червоний. У списку «Колір» групи елементів керування «Середина» також вибираємо червоний колір. Зафіксувати обраний колір.

Клацаємо мишею на комірці першого рядка поля «Відображати для наст.завдань», комірку буде виділено. З правого її краю з'явиться кнопка списку, що відкривається. У цьому списку вибираємо критичне завдання і натискаємо клавішу «Enter». Прямокутник виділення переміститься в комірку наступного, другого рядка, у якому буде визначатися зовнішній вигляд смужок звичайних завдань. У списку цієї комірки вибрати «Некритичні завдання» і натиснути клавішу «Enter».

Тепер на діаграмі критичний шлях буде відображатися червоним кольором, а некритичні роботи – синім. Закрити вікно «Стилі відрізків». Установлені елементи форматування відобразяться на діаграмі: червоним кольором виділяється критичний шлях, синім - некритичні роботи, бірюзовим - контрольні точки. Критичний шлях на діаграмі може бути відображений у процесі автоматичного форматування, виконуваного майстром діаграми Ганта, який запускається натисканням кнопки «новий проект» панелі інструментів «Стандартна».

**6. Групування робіт і створення структури графіка робіт.** Провести за етапами: планування; розподіл; реалізація.

Введемо назви цих етапів у полі «Назва завдання» таблиці. Нехай етап планування поєднує підетапи розробки технічного завдання і проектування системи; етап розподілу - включає види робіт: розбивка

на підзадачі та розподіл завдань; останній етап реалізація - поєднує види робіт: написання, впровадження.

Тепер потрібно виділити етапи роботи. Виділити роботи першого етапу: клацанням миші виділити комірку з назвою «розбивка на підзадачі»; натиснути і утримувати клавішу «Shift»; клацнути мишею на комірці з назвою роботи «розподіл завдань»; відпустити клавішу «Shift». Два види робіт будуть виділені.

Не скасовуючи виділення комірок, установити покажчик миші на першому символі назви роботи «розбивка на підзадачі» так, щоб покажчик прийняв форму тонкої горизонтальної лінії зі стрілками на кінцях. Натиснемо і утримуємо ліву кнопку миші, перемістити мишу вправо. При цьому вертикальна лінія, що з'явиться, вкаже положення відступу. Відпустити ліву кнопку миші. Виділені в таблиці назви робіт будуть згруповані. При цьому їх назви змістяться вправо, а назва етапу – «Планування» відобразиться напівжирним накресленням і чорним кольором. Ліворуч від назви етапу з'явиться значок, який означає, що в цей момент усі види робіт цього етапу відображаються на екрані. У поле «Тривалість» з'явиться інформація про тривалість даного етапу, яку *Microsoft Project* визначає на підставі тривалості окремих видів робіт, включених у цей етап.

На діаграмі з'явиться новий елемент у вигляді чорної смуги із трикутними зубцями на кінцях, який позначає етап проекту. Аналогічно включимо наступні роботи в другий і третій етапи. Створена структура буде мати приблизно такий вид, як на рис. 1.4. У структурі можна приховати або показати види робіт, що входять у той або інший етап. Для цього потрібно клацнути мишею на значку ліворуч від назви етапу. Назви робіт, включені у цей етап, будуть приховані. Зникнуть також їхні смужки на діаграмі. Ліворуч від назви етапу з'явиться значок —, який означає, що роботи цього етапу приховані.

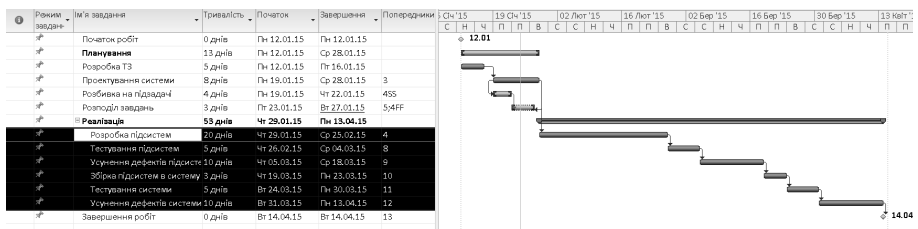


Рис. 1.4. Структура складного проекту



Інший спосіб. Клацанням миші виділити комірку з назвою етапу і натиснути кнопку — на панелі інструментів (рис. 1.5).

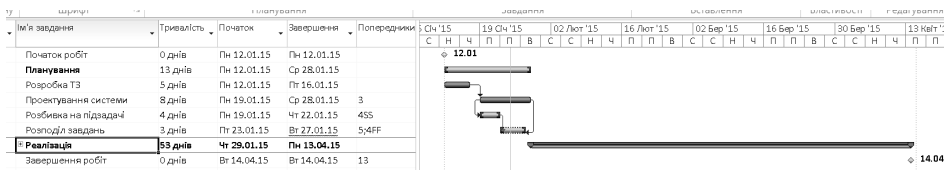


Рис. 1.5. Діаграма із прихованими роботами 2 етапу

Показати всі приховані роботи можна, якщо натиснути кнопку + на панелі інструментів «Форматування».

## Оформлення результатів роботи

Результати лабораторної роботи оформлюються у вигляді звіту обов'язковими елементами, якого є: титульний аркуш; формулювання мети та завдання на роботу; основна частина; висновки.

Основна частина роботи повинна містити етапи вирішення індивідуального завдання шляхом аналізу досягнутих результатів в обраній предметній області.

Висновки повинні містити результати виконаної роботи.

## Індивідуальні завдання

Скласти графік робіт відповідно до проектуемого бізнес-процесу, список яких пропонувалося в лабораторній роботі 3 або узгодити вибір бізнес-процесів з викладачем.

## Запитання для самоперевірки

1. З якими сутностями оперує MS Project ?
2. Що відкладається на вертикальній осі діаграми Ганта?
3. Як в MS Project знайти вкладку календар?
4. Як підключити календар до створюваного проекту?
5. Що являє собою критичний шлях та як його визначити засобами MS Project?
6. Як в MS Project розподіляться вартість ресурсів кожного виду робіт ?
7. Як показати приховані роботи етапу?

## **Лабораторна робота 1.4.**

### **СТВОРЕННЯ ГРАФІКУ РОБІТ ТА РЕСУРСІВ ЗА ДОПОМОГОЮ ДІАГРАМИ PERT**

**Мета:** вивчити можливості перегляду інформації в Microsoft Project.

**Завдання:** створити графік робіт за допомогою діаграми PERT, побудувати графік ресурсів, порівняти діаграми Ганта й PERT.

#### **Основні теоретичні відомості**

Техніка PERT (Program Evaluation and Review Technique) – це спосіб аналізу завдань, необхідних для виконання проекту, особливо, аналізу часу, який потрібен для виконання кожного окремого завдання, а також визначення мінімального необхідного часу для виконання всього проекту.

**Діаграма PERT** являє собою множину точок-вершин (події) та орієнтовані дуги (роботи), що їх з'єднують. Кожній дузі, розглянутій в якості деякої роботи із числа потрібних для здійснення проекту, приписуються певні кількісні характеристики. У такий спосіб відображається той факт, що ні до однієї з робіт не можна приступити раніш, ніж будуть виконані всі роботи, що передують їй згідно з технологією реалізації проекту. Початок цього процесу – вершина без вхідних, а закінчення – вершина без вихідних дуг. Інші вершини повинні мати як вхідні, так і вихідні дуги. Послідовність дуг, у якій кінець кожної попередньої збігається з початком наступної, трактується як шлях від відправної вершини до завершальної, а сума довжин таких дуг – його тривалість. Найбільша визначає тривалість усього цього проекту. Цей шлях – критичний, тобто саме від тривалості складових його робіт залежить загальна тривалість проекту.

#### **Порядок виконання роботи**

1. Ознайомитися з літературою [1, 2, 9, 10].
2. **Створення графіку робіт за допомогою діаграми PERT.** В меню «Вид» вибрати вкладку «Мережевий графік». Кожний вид роботи тут представлений прямокутником, а стрілки, що з'єднують ці прямокутники, символізують зв'язки між роботами. Усередині

кожного прямокутника зазначена назва роботи, її порядковий номер у таблиці, тривалість, дати початку і завершення.

**3. Діаграма в режимі використання завдань.** В меню «Вид» вибрати вкладку «Використання завдань» (рис. 1.6). У робочому вікні програми знайти дві панелі. У лівій панелі знайти таблицю зі списком робіт, під кожним з яких вказуються призначені йому ресурси. Для кожного виду робіт і ресурсів у сусідніх колонках виводиться обсяг робіт у годинах, тривалість, дати початку і завершення робіт. На правій панелі розглянути таблицю-календар, з обсягами робіт для кожного ресурсу, видом робіт на кожний робочий день проекту в годинах.

Название задачи	Трудозатраты	Подробности	24 Июн '08											
начало работ	0 ч	Трудозатр.	С	Ч	П	С	В	П	В	С	Ч	П		
разработка технического э	240 ч	Трудозатр.	30ч	48ч	48ч	48ч	48ч	18ч						
Аналитик	120 ч	Трудозатр.	15ч	24ч	24ч	24ч	24ч	9ч						
Менеджер	120 ч	Трудозатр.	15ч	24ч	24ч	24ч	24ч	9ч						
проектирование системы	104 ч	Трудозатр.						24ч	24ч	24ч	24ч	8ч		
Аналитик	34,67 ч	Трудозатр.						8ч	8ч	8ч	8ч	2,67ч		
Руководитель прое.	34,67 ч	Трудозатр.						8ч	8ч	8ч	8ч	2,67ч		
Разработчик	34,67 ч	Трудозатр.						8ч	8ч	8ч	8ч	2,67ч		
разбиение на подзадачи	32 ч	Трудозатр.											5,33ч	
Руководитель прое.	32 ч	Трудозатр.											5,33ч	

Рис. 1.6. Режим використання робіт

У цьому режимі розподіляють людські та інші ресурси, можна вводити і редагувати таку інформацію про види робіт і ресурси: обсяг робіт і ресурсів, вартість, початкову та кінцеву дати і т.ін.; кількість годин, запланованих для кожного ресурсу на конкретний вид робіт; змінити кількість робочих годин, запланованих для того або іншого ресурсу; розділити роботу на частини так, щоб наступна її частина починалася з більш пізньої дати.

**4. Графік ресурсів.** Часова шкала графіку ресурсів відображає інформацію про розподіл, обсяги робіт і вартості ресурсів. Переглянути дві панелі. На лівій відображається назва ресурсу, а на правій - діаграма використання цього ресурсу в обраному масштабі часу. Діаграму кожного ресурсу можна відобразити, використовуючи горизонтальну смугу прокручування лівої панелі.

В меню «Вид» вибираємо вкладку «Графік».

**5. Використання ресурсів.** В цьому режимі ресурси згрупувати за роботами. Ліва панель робочого вікна містить таблицю зі структурованим списком ресурсів і робіт, яким ці ресурси призначені. У колонці «Працевзатрати» вставити загальну кількість

робочих годин для кожного ресурсу і кожного виду робіт. На правій панелі, у таблиці-календарі переглянути щоденні обсяги робіт у годинах для кожного ресурсу і виду робіт. Для перегляду діаграми в інших режимах вибрати в меню «Вид» вкладки «Панель подань» та «Інші подання». У вікні «Інші подання» вибрати доступний режим перегляду і натиснути кнопку «Застосувати».

### **Оформлення результатів роботи**

Результати лабораторної роботи оформлюються у вигляді звіту, обов'язковими елементами, якого є: титульний аркуш; формулювання мети та завдання на роботу; основна частина; висновки.

Основна частина роботи повинна містити етапи вирішення індивідуального завдання шляхом аналізу досягнутих результатів в обраній предметній області.

Висновки повинні містити результати виконаної роботи.

### **Індивідуальні завдання**

Скласти діаграму PERT відповідно до проектуемого бізнес-процесу, список яких пропонувалося в лабораторній роботі 1.1 або узгодити вибір бізнес-процесів з викладачем.

### **Оформлення результатів роботи**

Результати лабораторної роботи оформлюються у вигляді звіту обов'язковими елементами, якого є: титульний аркуш; формулювання мети та завдання на роботу; основна частина; висновки. Зразок титульного листа наведений у Додатку 1.

Основна частина роботи повинна містити етапи вирішення індивідуального завдання шляхом аналізу досягнутих результатів в обраній предметній області.

Висновки повинні містити результати виконаної роботи.

### **Індивідуальні завдання**

Скласти діаграму PERT відповідно до проектуемого бізнес-процеса, список яких пропонувалося в лабораторній роботі 1.3 або узгодити вибір бізнес-процесів з викладачем.

### **Запитання для самоперевірки**

1. Назвати основне призначення подання графіку робіт діаграмою PERT.
2. В чому відмінність PERT-події та PERT-активності?
3. Назвати основні складові розрахунку очікуваного часу.
4. Що розуміють під критичним шляхом?
5. Що відображається на горизонтальній та вертикальних вісях графіку ресурсів?
6. Що позначається на дугах діаграми PERT?
7. Про що говорить вершина графа?

## **Лабораторна робота 1.5**

### **РІШЕННЯ ПРОБЛЕМИ ПЕРЕВАНТАЖЕННЯ РЕСУРСІВ**

**Мета:** засвоїти підходи до вирішення проблеми перевантаження ресурсів засобами Microsoft Project.

**Завдання:** визначити за допомогою діаграми Ганта перевантажені ресурси і оптимізувати їх; оцінити фактичні витрати та побудувати оптимізований графік робіт; оформити звіт про виконану роботу.

### **Основні теоретичні відомості**

Перевищення доступності ресурсу полягає в тому, що для виконання призначеної роботи ресурсу потрібно більше часу, ніж є. Існує декілька причин, здатних привести до цього. Найпоширенішою серед них є призначення ресурсу на завдання, виконання яких повністю або частково здійснюється одночасно. Іншим варіантом може бути збільшення обсягу робіт завдання, що приводить до перевищення припустимого рівня завантаження ресурсу. Нарешті, призначення ресурсу через зміни в плані може припадати на дні, коли ресурс недоступний. Вирівняти завантаження ресурсів можна декількома способами, наприклад:

- змінити обсяг роботи перевантажених ресурсів, скоротити деякі завдання або призначити додаткових співробітників на їх виконання;

- позбутись перетинання завдань, вставивши в розклад перерви в завданнях або призначеннях чи змінивши дати їх початку і завершення;
- врахувавши роботу, що виконується ресурсом поверх норми.

Вирівняти завантаження ресурсів в Microsoft Project можна автоматизовано, а можна й ручним засобом. Як правило, використовуються обидва способи, оскільки команда автоматизованого вирівнювання не охоплює всіх ресурсів.

Ручне вирівнювання ресурсів здійснюється у два етапи. Спочатку потрібно знайти ті завдання, які перевантажують ресурси. Потім потрібно визначити, як позбутися перевантаження, оскільки варіантів досить багато. Можна перенести завдання, перервати або змінити їх тривалість. Можна зменшити обсяг роботи ресурса або вилучити призначення, шляхом як виділення завдання для іншого співробітника замість перевантаженого, так і не робити цього. Нарешті, можна зберегти перевантаження, якщо перевести надлишкові працезатрати ресурсу в понаднормові.

### **Порядок виконання роботи**

**1.** Ознайомитися з літературою [1, 2, 9, 10].

**2. Визначення перевантаження ресурсів.** Ресурс вважається перевантаженим, якщо йому призначений обсяг роботи більший, ніж він може виконати у робочий час. В інформаційному полі цей ресурс відзначається спеціальним значком «!». Якщо мишею клацнути по цьому значку, то на екрані з'явиться повідомлення про те, що даний ресурс перевантажений. Якщо ж робота запланована в меншому обсязі, то ресурс буде недовантаженим.

Проектом для кожної роботи повинно бути призначено достатню кількість ресурсів на планований час робіт. І кожному ресурсу повинен бути визначений оптимальний обсяг робіт, що виключає перевантаження і недовантаження.

Щоб розв'язати проблему перевантаження, потрібно визначити спочатку, які ресурси, і в який час перевантажені, і які роботи при цьому повинні виконуватися. Після цього можна вибрати один із двох варіантів:

- змінити кількість робочого часу, необхідного ресурсам для виконання даної роботи, тобто призначити понаднормовий час або роботу у вихідні дні;

- змінити графік робіт так, щоб роботи виконувалися в той час, коли ресурси не будуть перевантажені.

Який із цих варіантів треба вибрати, залежить від різних чинників проекту: бюджету, доступності ресурсів, характеру виконуваних робіт і т.ін.

Розглянемо перевантаження менеджера, яке обумовлене тим, що він одночасно виконує роботи з розробки технічного завдання і проектування системи, які вимагають 14-16 годин робочого часу.

### **3. Розв'язання проблеми перевантаження ресурсів.**

Виконується послідовність робіт: переключитися в режим діаграми Ганта; виділити в таблиці роботу «розробка технічного завдання»; натиснути кнопку вікна «Відомості про завдання» на панелі інструментів «Стандартна»; зайти по вкладці «Ресурси»; призначити ресурсу «Менеджер» значення 50%; ввійти по вкладці «Додатково»; у списку «Тип роботи» вибрати «Фіксована тривалість», інакше тривалість роботи буде скорочена на 50%. Закриваємо вікно кнопкою «ОК». Аналогічно виконати дії для роботи «Проектування системи». В режимі «використання ресурсів» переконатися, що перевантаження менеджера ліквідоване.

Інший варіант розв'язання проблеми: перемістити перевантажений ресурс для виконання роботи на інший час або змінює графік даної роботи так, щоб вона виконувалася, коли ресурс буде вільний; зменшити обсяг робіт, які повинен виконувати перевантажений ресурс; змінити робочий календар перевантаженого ресурсу так, щоб він мав більшу кількість робочих годин; затримати початок виконання роботи доти, поки перевантажений ресурс не зможе приступити до неї; призначити даному виду робіт додаткові ресурси; розділити роботу, яку повинен виконувати перевантажений ресурс, на частини так, щоб він міг виконувати в інший час.

**4. Способи оптимізації графіка робіт.** На цьому етапі потрібно з'ясувати:

- чи відповідає проект очікуваним результатам;
- чи досягаються цілі проекту;
- чи не перевищується проектом встановлена вартість;
- чи ефективно використовуються ресурси;
- чи не занадто розтягнуті строки його реалізації.

Найбільш очевидним шляхом скорочення тривалості проекту є скорочення критичного шляху за допомогою зменшення тривалості окремих критичних робіт. Починати оптимізацію завжди необхідно із самої тривалої роботи на критичному шляху.

Зменшити тривалість роботи на критичному шляху можна також скоротивши обсяг роботи, передбачений для даного виду робіт. За замовчуванням MS Project обчислює тривалість роботи на підставі загального обсягу роботи, кількості одиниць ресурсів, призначених даному виду робіт, робочого часу і обсягу робіт, визначеного для кожного ресурсу. Змінити обсяг робіт можна в режимі «використання робіт», зменшивши в полі «Працевитрати» загальний обсяг роботи, запланований для даного виду робіт.

Ще один спосіб зменшення довжини критичного шляху полягає у видаленні або комбінуванні робіт. Для зменшення довжини критичного шляху можна використовувати також призначення додаткових ресурсів критичним роботам.

У деяких випадках зменшення тривалості робіт на критичному шляху можна добитися, призначивши ресурсам, які їх виконують, понаднормові роботи.

Зменшити довжину критичного шляху, призначивши ресурсу «Керівник проекту» роботи у вихідні дні 25 і 26 червня. Робота в ці дні згодом буде компенсована наданням додаткових днів відпочинку. Для цього відзначимо в базовому календарі «Календар 1» ресурсу «Керівник проекту» зазначені дні як робочі. Далі виконуємо операції:

- в меню «Сервіс» вибираємо команду «Зміна робочого часу»;
- у списку «Для» вибираємо ресурс «Керівник проекту»;
- за допомогою смуги прокручування праворуч від календаря встановимо місяць «червень»;
- виділити в календарі комірки з датами 25 і 26 червня;
- установити перемикач «Нестандартний робочий час»;
- у полях «З» і «До» вводимо робочий час у ці дні: 9:00 -13:00, 14:00 - 18:00;
- зняти виділення, при цьому у календарі ці дні будуть відзначені як робочі, закрити вікно.

Переконалися по діаграмі Ганта, що тривалість роботи ресурсу «Керівник проекту» тепер складає 4 календарних, а не робочих дня, а дата завершення проекту пересунулася з 04.07 на 30.06.



Створення проекту завершено, тому його треба зберегти разом з базовим планом:

- у меню «Сервіс» виділяємо пункт «Відстеження», а потім вибираємо команду «Зберегти базовий план»;
- установимо перемикач у положення «Зберегти базовий план», після чого вибираємо в списку план, який необхідно зберегти;
- установимо перемикач у положення «всього проекту».

**5. Виконання ходу робіт і фактичних витрат.** Ввести дату початку робіт і зробити оцінку проходження контрольної точки «Початок робіт» так:

- у режимі діаграми Ганта виділити контрольну точку «Початок робіт» у полі «Назва завдання»;
- в меню «Сервіс» вибрати команду «Відстеження/ Обновити завдання».

У полі «Назва» вказана назва виділеної контрольної точки – «Початок робіт», а в полі «Тривалість» її планова тривалість. У полях з лічильниками «% Завершення», «Фактична тривалість», і «Тривалість, що залишилася» відзначаються відповідно: відсоток виконання - 0%, фактична тривалість - 0 днів, тривалість, що залишилася - 0 днів.

Група полів «Поточні» містить планові строки початку і завершення роботи. У списку «Початок» групи полів «Фактичні дати» на календарі вкажемо дату «2 травня». У полі з лічильником «% Завершення» вказати «100%». Замість зазначення відсотка виконання можна також вибрати дату завершення - у списку «Завершення». Закрити вікно «Відновлення завдань». В інформаційному полі таблиці ліворуч від назви контрольної крапки «Початок робіт» з'явиться оцінка про її проходження. Оцінку про виконання робіт першого етапу «Планування» зробити так:

- виділити в таблиці назви робіт «Розробка технічного завдання» і «Проектування системи» утримуючи натиснутою клавішу «Ctrl»;
- в меню «Сервіс» вибрати команду «Відстеження/Обновити проект».

Установити перемикач «оновити витрати як завершені», в списку, що ліворуч від нього, виберемо дату завершення робіт 31.07, установимо перемикач «Для обраних завдань». В інформаційному полі таблиці проконтролювати оцінки виконання

зазначених видів робіт, а на діаграмі, усередині горизонтальних смужок відповідних робіт - чорні смуги, що відображають відсоток виконаних робіт - 100%, рис. 1.7.

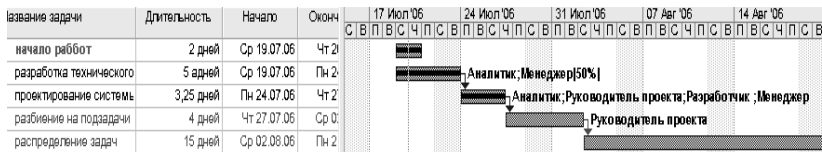


Рис. 1.7. Оцінка виконання робіт

Якщо виконана робота перестала бути критичною, її колір змінюється. Так, в проєкті критична робота «Розробка змісту» змінила червоний колір на синій.

Ввести відсотки виконаної роботи у вікні «Відомості про завдання». Для цього виконати таку послідовність:

- виділити в таблиці роботу «Розбивка на підзадачі»;
- відкрити вікно «Відомості про завдання» панелі інструментів «Стандартна»;
- переходимо на вкладку «Загальні».

У полі «Відсоток завершення» встановлюємо 50%. Чорна смуга на прямокутнику-роботі «Розбивка на підзадачі» показує відсоток виконання цієї роботи.

**6. Користування панеллю інструментів «Відстеження».** В меню «Вид» виберемо команду «Панелі інструментів / Відстеження», виділити роботу «Розбивка на підзадачі»; на панелі інструментів «Відстеження» натиснути кнопку 100%. Контролювати, що робота відзначена як виконана.

**7. Ввод фактичної дати завершення роботи.** На панелі інструментів «Відстеження» викликаємо вікно «Відновлення завдань», де встановити дату 4.06 у списку «Завершення» групи «Фактичні дати»; проконтролювати фактичну дату завершення роботи у таблиці і на діаграмі.

**8. Ввод оцінки виконання роботи.** Виконати таку послідовність операцій:

- установити покажчик миші на лівому краю смужки-роботи «Розподіл завдань», за допомогою лівої кнопки миші відкрити інформаційне вікно роботи, де зазначені дата її початку і завершення;

- не відпускаючи ліву кнопку миші, перемістити мишу по смужці-роботі, при цьому в інформаційному вікні проконтролювати зміну дати завершення цієї роботи, встановлюємо дату завершення роботи відповідно плановій - 9.06;

- закрити інформаційне вікно, проконтролювати в таблиці, що робота «Розподіл завдань» відзначена як виконана.

При необхідності перенести частину роботи, що залишилася, на більш пізній термін.

Якщо робота «Написання» розпочата у встановлений термін і протягом двох днів виконана на 50%, після чого виникла необхідність перервати її виконання. Вставляємо відповідну оцінку так:

- виділити в таблиці роботу «Впровадження»;
- на панелі інструментів «Відстеження» натиснути кнопку 50%, при цьому на діаграмі з'являється оцінка часткового виконання роботи;

- установити покажчик миші на смужці-роботі «Написання» праворуч смуги виконання;

- натиснути і утримувати ліву кнопку миші, перемістити мишу вправо так, щоб в інформаційному вікні відобразилася нова дата початку частини роботи, що залишилася;

- відпустити ліву кнопку миші та проконтролювати на діаграмі між виконаною і невиконаною роботами появу розриву в один день. Відзначити 100% виконання частини роботи, що залишилася, «Написання» будь-яким відомим способом.

**9. Контроль виконання проекту.** В меню «Вид» вибрати «Діаграма Ганта з відстеженням», рис. 1.8. На цій діаграмі сірими смугами показаний базовий план, а синіми - фактичне виконання. Праворуч від кожної виконаної роботи зазначається відсоток виконання 100%. Невиконані критичні роботи відображаються червоним кольором, і праворуч від них зазначається відсоток виконання – 0%.

Виконані етапи також відзначені спеціальними умовними позначками. Переглянути числові показники можна в таблиці відхилень. В меню «Вид» вибрати команду «Таблиця-Введення/Відхилення». Перемістити праву межу лівої панелі до правого краю робочого вікна, щоб побачити таблицю повністю (рис. 1.9).

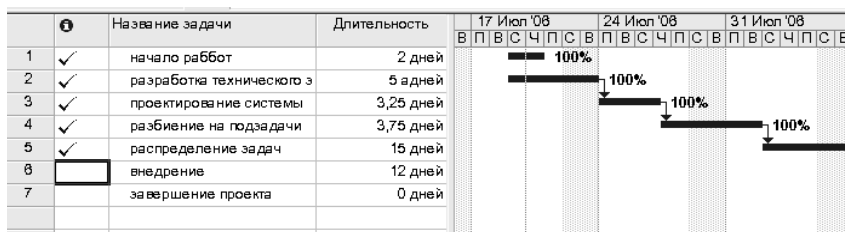


Рис. 1.8. Діаграма Ганта з відстеженням

Название задачи	Начало	Окончание	Базовое начало	Базовое окончание	Отклон. начала	Отклон. окончания
1 начало работ	Ср 19.07.06	Чт 20.07.06	НД	НД	0 дней	0 дней
2 разработка технического	Ср 19.07.06	Пн 24.07.06	НД	НД	0 дней	0 дней
3 проектирование системы	Пн 24.07.06	Чт 27.07.06	НД	НД	0 дней	0 дней
4 разбиение на подзадачи	Чт 27.07.06	Вт 01.08.06	НД	НД	0 дней	0 дней
5 распределение задач	Ср 02.08.06	Пт 18.08.06	НД	НД	0 дней	0 дней
6 внедрение	Пн 21.08.06	Вт 05.09.06	НД	НД	0 дней	0 дней
7 завершение проекта	Вт 05.09.06	Вт 05.09.06	НД	НД	0 дней	0 дней

Рис. 1.9. Таблица відхилень

У цій таблиці в колонках «Відхилення початку» і «Відхилення завершення» переглянути різницю між фактичними і плановими датами. Відхилення може бути позитивним і негативним. Позитивне відхилення означає, що робота почата або завершена пізніше планового строку, а негативне – раніше. По таблиці чітко видно, що початок і завершення робіт першого етапу йде у встановлений термін, а другого і третього раніше запланованого.

У випадку затримки необхідно прийняти екстрені заходи для завершення проекту в строк. Одним зі способів розв'язання цього завдання може бути скорочення тривалості деякої роботи шляхом призначення понаднормових робіт ресурсу, призначеному цьому виду робіт. Додавання понаднормових робіт в обсязі 8 годин скоротить тривалість цього виду робіт на 1 день.

**10. Додавання понаднормових робіт.** В меню «Вікно» вибрати команду «Розділити». Робоче вікно програми буде розділене на два вікна по горизонталі: у верхньому вікні буде відображатися діаграма Ганта, що відстежує, а в нижньому - форма завдань. Виконати такі операції: зробити завдання поточним; в меню «Формат» вибрати команду «Подробиці /Працевитрати на ресурси»; виділити роботу на діаграмі Ганта з відстеженням, у верхньому вікні; на формі завдань у полі «Назва ресурсу» таблиці виділити комірку першого рядка поля «Понаднормові

працездатності» форми завдань; установити необхідну кількість годин понаднормової роботи і нажати клавішу «Enter»; в меню «Вікно» вибрати команду «Зняти поділ» і проконтролювати введені зміни по діаграмі Ганта з відстеженням.

Зберегти новий проміжний план. Для цього в меню «Сервіс» вибрати команду «Відстеження/Зберегти базовий план». Установити перемикач «Зберегти проміжний план». Закрити вікно.

**11. Відобразити хід виконання робіт для будь-якої дати часової шкали.** На панелі інструментів «Відстеження» натиснути кнопку «Σ», при цьому покажчик миші прийме зигзагоподібну форму; в інформаційному вікні «Індикатор виконання», де при русі миші встановлюємо дату, що цікавить, і клацаємо лівою кнопкою миші по діаграмі, де відобразиться графік виконання проекту.

Для одержання інформації про планову і фактичну вартість робіт, витрачених засобах, і засобах, необхідно виконати:

- в меню «Вид» вибрати команду «Таблиця/Витрати»;
- встановлюємо повний розмір таблиці (рис. 1.10):

	Название задачи	Фиксированные затраты	Начисление фикс. затрат	Общие затраты	Базовые	Отклонение	Фактические	Оставшиеся
1	начало работ	0,00р.	Пропорциональное	0,00р.	0,00р.	0,00р.	0,00р.	0,00р.
2	разработка технического	0,00р.	Пропорциональное	48 800,00р.	0,00р.	48 800,00р.	48 800,00р.	0,00р.
3	проектирование системы	0,00р.	Пропорциональное	34 140,00р.	0,00р.	34 140,00р.	34 140,00р.	0,00р.
4	разбиение на подзадачи	0,00р.	Пропорциональное	8 340,00р.	0,00р.	8 340,00р.	8 340,00р.	0,00р.
5	распределение задач	0,00р.	Пропорциональное	26 740,00р.	0,00р.	26 740,00р.	26 740,00р.	0,00р.
6	внедрение	0,00р.	Пропорциональное	16 130,00р.	0,00р.	16 130,00р.	9 410,00р.	6 720,00р.
7	завершение проекта	0,00р.	Пропорциональное	0,00р.	0,00р.	0,00р.	0,00р.	0,00р.

Рис. 1.10. Таблица витрат

У колонці «Загальні витрати» вказується загальна вартість кожного виду робіт, а в колонці «Базові» - планова. Різниця між ними відображається в полі «Відхилення». Якщо ця різниця негативна, то засоби зекономлені; а якщо ні, то - перевитрачені. У колонці «Фактичні» зазначається вартість уже виконаних робіт, а в колонці «Ті, що залишилися» - ще не виконаних.

Аналіз витрат засобів, призначених кожному ресурсу, проводиться так:

- таблицю використання завдань в меню «Вид» викликаємо командой «Використання завдань»;
- таблиця витрат в меню «Вид» вибираємо команду

«Таблиця/ Витрати» рис. 1.11.

	❶	Название задачи	Трудозатраты	Длительность	Начало	Окончание
1	✓	начало работ	0 ч	2 дней	Ср 19.07.06	Чт 20.07.06
2	✓	▢ разработка технического з	180 ч	5 адней	Ср 19.07.06	Пн 24.07.06
		Аналитик	120 ч		Ср 19.07.06	Пн 24.07.06
		Менеджер	60 ч		Ср 19.07.06	Пн 24.07.06
3	✓	▢ проектирование системы	104 ч	3,25 дней	Пн 24.07.06	Чт 27.07.06
		Аналитик	26 ч		Пн 24.07.06	Чт 27.07.06
		Менеджер	26 ч		Пн 24.07.06	Чт 27.07.06
		Руководитель проек	26 ч		Пн 24.07.06	Чт 27.07.06
		Разработчик	26 ч		Пн 24.07.06	Чт 27.07.06
4	✓	▢ разбиение на подзадачи	30 ч	3,75 дней	Чт 27.07.06	Вт 01.08.06
		Руководитель проек	30 ч		Чт 27.07.06	Вт 01.08.06
5	✓	▢ распределение задач	120 ч	15 дней	Ср 02.08.06	Пт 18.08.06
		Руководитель проек	120 ч		Ср 02.08.06	Пт 18.08.06
6		▢ внедрение	96 ч	12 дней	Пн 21.08.06	Вт 05.09.06
		Внедренец	96 ч		Пн 21.08.06	Вт 05.09.06
7		завершение проекта	0 ч	0 дней	Вт 05.09.06	Вт 05.09.06

Рис.1.11. Таблица витрат по каждому ресурсу

На рис. 1.11 показано, що вартість тут розподілена не тільки по видах робіт, але і по ресурсах. Це дозволяє оперативно одержати інформацію про витрату засобів кожним ресурсом окремо.

### Оформлення результатів роботи

Результати лабораторної роботи оформлюються у вигляді звіту обов'язковими елементами, якого є: титульний аркуш; формулювання мети та завдання на роботу; основна частина; висновки.

Основна частина роботи повинна містити етапи вирішення індивідуального завдання шляхом аналізу досягнутих результатів в обраній предметній області.

Висновки повинні містити результати виконаної роботи.

### Індивідуальні завдання

Відповідно до діаграми Ганта розвантажити ресурси відповідно до проектуємого бізнес-процеса, список яких пропонувалося в лабораторній роботі 1.3 або узгодити вибір бізнес-процесів з викладачем.

### Запитання для самоперевірки

1. Дати визначення поняттю перевантаження ресурсів.
2. В чому полягає перевищення доступності ресурсу?

3. Які Ви знаєте способи розв'язання завантаження ресурсів?
4. Назвати порядок дій для розв'язання проблеми перевантаження ресурсів.
5. Назвати способи оптимізації графіка робіт.
6. Назвати порядок дій при визначенні виконання ходу робіт і фактичних витрат.
7. Які існують способи відстеження ходу реалізації проекту?

## **Лабораторна робота 2.1**

### **ФУНКЦІОНАЛЬНЕ МОДЕЛЮВАННЯ ІНФОРМАЦІЙНИХ СИСТЕМ**

**Мета:** вивчити основи моделюванні бізнес-процесів засобами Vpwin в нотації IDEF0.

**Завдання:** побудувати функціональну модель інформаційної системи в середовищі Vpwin.

#### **Основні теоретичні відомості**

*Бізнес-процес* - це сукупність взаємопов'язаних заходів або задач, спрямованих на створення певного продукту або послуги для споживачів. *Структурним аналізом* прийнято називати метод дослідження системи, який починається з її загального огляду, а потім деталізується, набуваючи ієрархічну структуру з дедалі більшим числом рівнів. *Моделювання* ділових процесів, як правило, виконується за допомогою Case-засобів. До таких засобів відноситься Vpwin, що розроблена компанією Logic Works. Назва Vpwin склалося зі скорочення BP (англ. *business process*) і суфікса *win*, що відбиває орієнтацію на графічні операційні системи.

Vpwin підтримує три методології моделювання: *функціональне моделювання* (IDEF0), *опис бізнес-процесів* (IDEF3); *діаграми потоків даних* (DFD).

Модель в Vpwin розглядається як сукупність робіт, кожна з яких оперує з деяким набором даних. Робота зображується у вигляді прямокутників, дані – у вигляді стрілок. Якщо клацнути по будь-якому об'єкту моделі лівою кнопкою миші, з'являється контекстне меню, кожен пункт якого відповідає редактору певної властивості об'єкта.

Модель може містити чотири типи діаграм:

- контекстну діаграму (у кожній моделі може бути тільки одна контекстна діаграма);
- діаграми декомпозиції;
- діаграми дерева вузлів;
- діаграми тільки для експозиції (FEO).

*Контекстна діаграма* є вершиною деревовидної структури діаграм і являє собою саме загальний опис системи та її взаємодії із зовнішнім середовищем. Після опису системи в цілому проводиться розбиття її на великі фрагменти. Цей процес називається *функціональною декомпозицією*, а діаграми, які описують кожен фрагмент і взаємодію фрагментів, називаються *діаграмами декомпозиції*. Після декомпозиції контекстної діаграми проводиться декомпозиція кожного великого фрагмента системи на більш дрібні і так далі, до досягнення потрібного рівня подробности опису. Після кожного сеансу декомпозиції проводяться сеанси експертизи – експерти предметної області вказують на відповідність реальних бізнес-процесів створеним діаграммам. Знайдені невідповідності виправляються, і тільки після проходження експертизи без зауважень приступають до наступного сеансу декомпозиції. Синтаксис опису системи в цілому і кожного її фрагмента однаковий у всій моделі.

*Діаграма дерева вузлів* показує ієрархічну залежність робіт, але не взаємозв'язки між роботами. Діаграм дерев вузлів може бути в моделі як завгодно багато, оскільки дерево може бути побудоване на довільну глибину і не обов'язково з кореня.

*Діаграми для експозиції* (FEO) будуються для ілюстрації окремих фрагментів моделі, для ілюстрації альтернативної точки зору, або для спеціальних цілей.

### **Порядок виконання роботи**

1. Ознайомитися з літературою [1, 2, 11].
2. Запустити BPWin і створити нову модель в меню File → New.
3. Ввести ім'я моделі «Поставка продукції» (без лапок) і натисніть кнопку «ОК».
4. У вікні «Properties for New Model» ввести відомості про автора.
5. Отримати контекстну діаграму.



6. Установити основні властивості моделі, натиснувши правою кнопкою миші на будь-якому вільному місці моделі і в меню вибрати пункт «Model properties ...»

7. У вікні «Model properties» на вкладці «General» в полі «Project» ввести ім'я проекту «Моделювання процесу поставки».

8. Відкрити вкладку «Purpose» і в полі «Purpose» ввести «Моделювати існуючий процес поставки продукції», а в полі «Viewpoint» - «Відділ постачання», закрити вікно «Model Properties».

9. Перейти на контекстну діаграму і правою кнопкою миші клацнути по роботі. В меню вибрати пункт «Ім'я ...» і ввести ім'я «Поставка продукції», зберегти створену модель і ввести ім'я файлу "delivery".

Створити стрілки на контекстній діаграмі. Для цього виконати стрілки дії.

1. Натиснути кнопку з символом стрілки «Precedence Arrow Tool» на панелі інструментів.

2. Перенести курсор до однієї зі сторін екрану (наприклад, до лівої, якщо вноситься стрілка входу), поки не з'явиться темна смужка

3. Ввести початок та кінець стрілки.

4. В панелі інструментів вибрати опцію редагування, для чого клацнути по кнопці «Pointer Tool».

5. Ввести ім'я стрілки (рис. 2.4), для цього в меню вибрати пункт «Name ...» і натиснути кнопку «ОК». Зробити стрілки згідно переліку, наведеному в таблиці 1.1.

**Таблиця 1.1**

**Стрілки контекстної діаграми**

Ім'я стрілки	Визначення стрілки	Тип стрілки
Замовлення клієнтів	Відомості про продукцію, що цікавить клієнтів	Input
Дані про якість продукції на складі	Дані про існуючу на складі продукцію для поставки	Input
Дані про наявність продукції на складі	Дані про існуючу на складі продукцію після поставки	Output

Система автоматизированого обліку	Облік і обробка замовлень клієнтів, формування даних про складські приміщення і т.д.	Mechanism
Відділ забезпечення	Вибір постачальника, визначення терміну поставки і т.д.	Mechanism
Відділ маркетингу	Формування даних про потреби в продукції	Mechanism
Правила і процедури	Правила обробки замовлень клієнтів, формування замовлення на поставку і т.д.	Control

10. Зробити декомпозицію діаграми в послідовності.

10.1. В панелі інструментів увійти до діаграми «Go to Child Diagram».

10.2. У вікні «Activity Box Count» встановити кількість робіт в діаграмі декомпозиції рівне 4.

10.3. В меню вибрати пункт «Name...», ввести ім'я роботи. Імена робіт - «Формування замовлення на поставку», «Узгодження з постачальником термінів і форми оплати», «Доставка продукції», «Прийом поставленої продукції».

10.4. Зв'язати граничні стрілки і вставити на діаграму внутрішні стрілки. Для внутрішніх стрілок ввести імена.

10.5. Зберегти створену модель

11. Провести розщеплення моделі

11.1. Клацнути правою кнопкою миші по роботі «Формування замовлення на поставку», в меню вибрати пункт «Split Model ...» і у вікні «Split Options» ввести ім'я моделі - «Формування замовлення на поставку».

11.2. Переключитися на нову модель (використовуючи вікно «Model Explorer»).

11.3. Зберегти створену модель, для чого вибрати пункт меню Save (File → Save) і ввести ім'я файлу «dlvr\_dfd».

11.4. Клацнути правою кнопкою миші по роботі «Узгодження з постачальником термінів і форми оплати», в меню вибрати пункт «Split Model ...» і у вікні «Split Options» ввести ім'я моделі - «Узгодження з постачальником термінів і форми оплати».

11.5. Переключитися на нову модель через використовуючи вікно «Model Explorer».

11.6. Зберегти створену модель, для чого вибрати пункт меню Save (File → Save) і ввести ім'я файлу «dlvr\_idef3».

11.7. В результаті у вікні «Model Explorer» можна буде побачити вихідну модель і моделі, що з'явилися в результаті розщиплення, рис. 2.1.

11.8. Вихідні з робіт «Формування замовлення на поставку» і «Узгодження з постачальником термінів і форми оплати» граничні стрілки, що з'явилися в результаті декомпозиції, можна видалити.

11.9. Закрити створені в результаті розщеплення моделі.

12. Декомпонувати роботу «Формування замовлення на поставку».

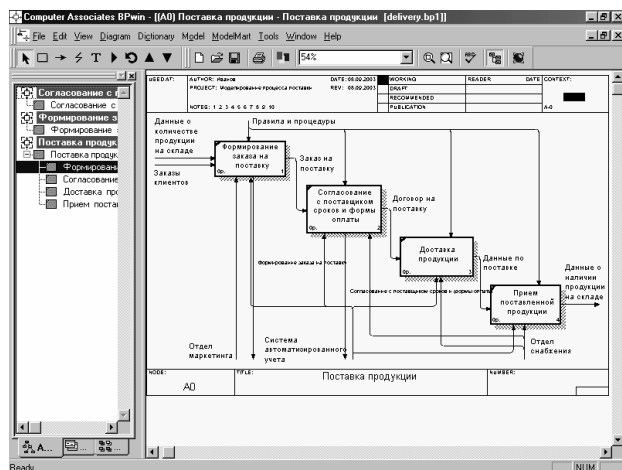


Рис. 2.1. Результат розщеплення моделі.

12.1. Вибрати роботу «Формування замовлення на поставку» і клацнути мишею по кнопці «Go to Child Diagram» в панелі інструментів

12.2. У вікні «Activity Box Count» встановити кількість робіт в діаграмі декомпозиції рівне 7.

12.3. Для робіт на діаграмі ввести такі назви: «Визначення номенклатури продукції, що замовляється», «Визначення списку постачальників», «Аналіз прайс-листів», «Аналіз термінів поставки», «Вибір постачальника», «Відправлення даних замовлення», «Отримання підтвердження замовлення».

12.4. Граничну стрілку «Дані про наявність продукції на складі» перейменувати в «Замовлення на поставку» і пов'язати з роботою «Отримання підтвердження замовлення». Пов'язати з роботами інші граничні стрілки.

12.5. Вставити на діаграму внутрішні стрілки.

13. Декомпозувати роботу «Узгодження з постачальником термінів і форми оплати».

13.1. В роботі «Узгодження з постачальником термінів і форми оплати» клацнути мишею по кнопці «Go to Child Diagram» в панелі інструментів.

13.2. У вікні «Activity Box Count» встановити кількість робіт в діаграмі декомпозиції рівне 5.

13.3. Для робіт на діаграмі ввести такі назви: «Узгодження строків поставки», «Узгодження форми оплати», «Узгодження термінів оплати», «Укладення договору на поставку», «Оплата поставки або видача гарантійного документа».

13.4. Додати на діаграму стрілки.

14. Сформувати діаграму дерева вузлів.

14.1. Для створення діаграми дерева вузлів потрібно вибрати в меню пункт «Diagram» і в вертикальному меню вибрати пункт «Add Node Tree ...», отримати вікно «Node Tree Wizard».

14.2. Натиснути кнопку «Готово», в результаті чого на екран буде виведена діаграма, зовнішній вигляд якої наведено на рис.2.2, збергти створену модель.

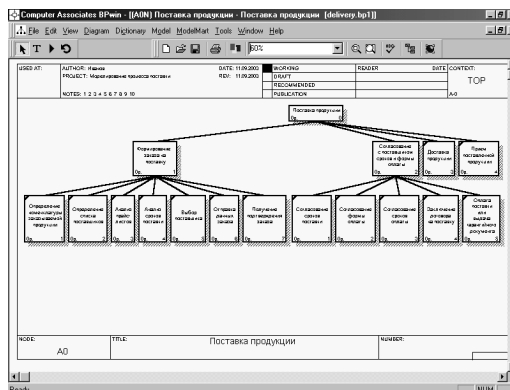


Рис. 2.2. Кінцева діаграма

## **Оформлення результатів роботи**

Результати лабораторної роботи оформлюються у вигляді звіту обов'язковими елементами, якого є: титульний аркуш; формулювання мети та завдання на роботу; основна частина; висновки.

Основна частина роботи повинна містити етапи вирішення індивідуального завдання шляхом аналізу досягнутих результатів в обраній предметній області.

Висновки повинні містити результати виконаної роботи.

## **Індивідуальні завдання**

Провести проектування бізнес-процесів відповідно до пропонованого списку або узгодити вибір бізнес-процесів з викладачем.

1. Продаж авіаквітків.
2. Покупка обладнання.
3. Закупка медичних препаратів.
4. Функціонування платіжної системи.
5. Закупка науково-технічної літератури.
6. Оказання авіаційних послуг.
7. Здійснення транспортних перевезень.
8. Надання ремонтних послуг.
9. Підготовка фахівців.
10. Діяльність складу.
11. Організація аудиторської діяльності.

## **Запитання для самоперевірки**

1. Що таке Case-засоби? Які Case-засоби Вам відомі?
2. Що таке бізнес-процес?
3. Якими Case-засобами відбувається моделювання бізнес-процесів?
4. Охарактеризувати можливості BPwin по моделюванню бізнес-процесів.
5. Як на діаграмі IDEF0 позначаються роботи, зв'язки?
6. Назвати основні типи зв'язків на контексній діаграмі.
7. Як позначається на контексній діаграмі тунелювання?

## **Лабораторна робота 2.2**

### **ПОБУДОВА МОДЕЛЕЙ ОПИСУ ПРОЦЕСІВ ПРИ СТРУКТУРНОМУ ПІДХОДІ МОДЕЛЮВАННЯ**

**Мета:** ознайомитись з технологією побудови опису процесів IDEF3.

**Завдання:** створити діаграму в нотації IDEF3 обраного бізнес-процесу.

#### **Основні теоретичні відомості**

**IDEF3** (*Integrated DEfinition for Process Description Capture Method*) – методологія моделювання та стандарт документування процесу, який показує причинно-наслідкові зв'язки між ситуаціями і подіями в зрозумілій експерту формі, використовуючи структурний метод вираження знань про те, як функціонує система, процес або підприємство. Система описується як упорядкована послідовність подій з одночасним описом об'єктів, що мають відношення до процесу, що моделюється.

IDEF3 складається з двох методів опису: технологічних процесів Process Flow Description (PFD) і переходів станів об'єктів Object State Transition Description (OSTD).

Основу методології IDEF3 становить графічна мова опису процесів. Модель в нотації IDEF3 може містити два типи діаграм:

- діаграму «Описи послідовності етапів процесу» (Process flow description diagrams, PFDD);
- діаграму «Мережі трансформації стану об'єктів» (Object state transition network, OSTN).

Діаграма IDEF3 типом Process Flow Description включає 7 основних описових блоків: роботи (boxes, activities); стрілки або зв'язки (arrows, links); перехрестя (junctions); об'єкти посилаць; unit of behavior; decomposition; elaboration.

#### **Порядок виконання роботи**

1. Ознайомитися з літературою [1, 2, 12].
2. Відкрити модель, створену в результаті розщеплення вихідної моделі і збережену в файлі «dlvr\_idef3.bp1».

3. Виконати декомпозицію моделі, вказавши при цьому тип моделі - IDEF3 і кількість робіт в діаграмі - 5.

#### 4. Формування діаграми

4.1. Для робіт на діаграмі ввести такі назви: «Узгодження строків поставки», «Узгодження форми оплати», «Узгодження термінів оплати», «Укладення договору на поставку», «Повна оплата поставки».

4.2. Додати дві нові роботи - «Часткова оплата поставки» та «Видача гарантійного документа про наступної оплати». Для додавання нової роботи клацнути мишею по кнопці «Activity Box Tool», а потім клацнути по тому місцю діаграми, де потрібно розмістити нову роботу.

4.3. Створити на діаграмі перехрестя клацнувши мишею по кнопці «Junction Tool», а потім клацнути по тому місцю діаграми, де потрібна розмістити перехрестя. Типи перехресть та їх розміщення на діаграмі показані на рис.2.3. Вибрати тип перехрестя з таких міркувань:

- узгодженості форми оплати і термінів оплати;
- проведення оплати поставки тільки після завершення договору.

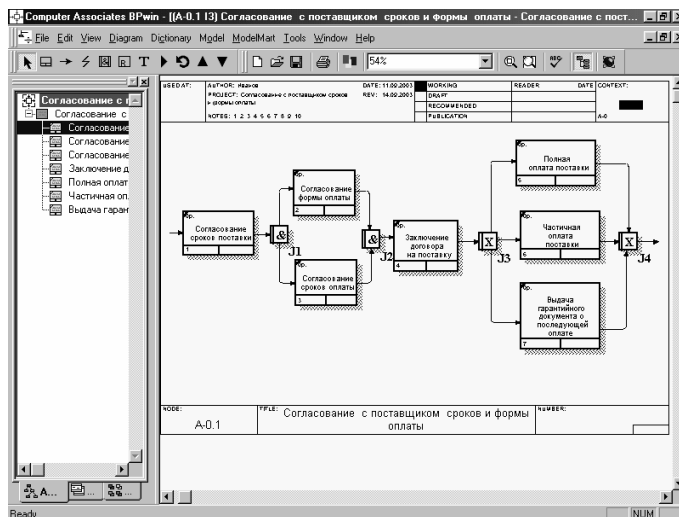


Рис. 2.3. Розміщення робіт на діаграмі.

4.4. Розмістити на діаграмі внутрішні і граничні стрілки.

### **Оформлення результатів роботи**

Результати лабораторної роботи оформлюються у вигляді звіту обов'язковими елементами, якого є: титульний аркуш; формулювання мети та завдання на роботу; основна частина; висновки.

Основна частина роботи повинна містити етапи вирішення індивідуального завдання шляхом аналізу досягнутих результатів в обраній предметній області.

Висновки повинні містити результати виконаної роботи.

### **Індивідуальні завдання**

Провести проектування бізнес-процесів відповідно до списку пропонованого в лабораторній роботі 3 або узгодити вибір бізнес-процесів з викладачем.

### **Запитання для самоперевірки**

1. Що таке діаграма та які види діаграм використовуються в нотації IDEF3?

2. В чому заключається основна ідея функціонального аналізу бізнес процесів?

3. З яких будівельних блоків складається діаграма в нотації IDEF3?

4. Які генерації звітів моделі в нотації IDEF3 існують?

5. Який зміст счеплення та розчеплення моделей?

6. В чому полягає різниця між нотаціями IDEF0 та IDEF3?

7. В чому полягає різниця між функціональною і об'єктною методиками?

## **Лабораторна робота 2.3 ПОБУДОВА МОДЕЛЕЙ ПОТОКІВ ДАНИХ**

**Мета:** ознайомитись з технологією DFD.

**Завдання:** побудувати діаграму потоків даних для обраної інформаційної системи.

### **Основні теоретичні відомості**



*Діаграми потоків даних (Data Flow Diagramming) є основним засобом моделювання функціональних вимог до проєктованої системи. Вимоги представляються у вигляді ієрархії процесів, пов'язаних потоками даних. Діаграми потоків даних показують, як кожен процес перетворить свої вхідні дані у вихідні, і виявляють відносини між цими процесами. DFD-діаграми успішно використовуються як доповнення до моделі IDEF0 для опису документообігу та обробки інформації. Подібно IDEF0, DFD представляє модельовану систему як мережу зв'язаних робіт. Основні компоненти DFD – процеси або роботи, зовнішні сутності, потоки даних, накопичувачі даних (сховища).*

У VPwin для побудови діаграм потоків даних використовується *нотація Гейна-Сарсона*. Діаграма DFD розглядає систему як *сукупність предметів*. Стрілки на DFD показують, як об'єкти (включаючи дані) рухаються від однієї роботи до іншої. Потоки робіт зображуються стрілками і описують рух об'єктів з однієї частини системи в іншу. Оскільки в DFD кожна сторона роботи не має чіткого призначення, як в IDEF0, стрілки можуть підходити і виходити з будь-якої грані прямокутника роботи. У DFD також застосовуються *двонаправлені стрілки* для опису діалогів типу "команда-відповідь" між роботами, між роботою і зовнішньою сутністю і між зовнішніми сутностями.

У матеріальних системах сховища даних зображуються там, де об'єкти очікують обробки, наприклад в черзі. У системах обробки інформації сховища даних є механізмом, який дозволяє зберегти дані для подальших процесів.

У DFD стрілки можуть зливатися і розгалужуватися, що дозволяє описати *декомпозицію стрілок*. Кожен новий сегмент такої стрілки може мати власне ім'я.

У DFD номер кожної роботи може включати префікс А, номер батьківської роботи і номер об'єкта. *Номер об'єкта* - це унікальний номер роботи на діаграмі. Наприклад, робота може мати номер А.12.4. Унікальний номер мають сховища даних і зовнішні сутності незалежно від їх розташування на діаграмі. Кожне сховище даних має префікс D і унікальний номер, наприклад D5. Кожна зовнішня сутність має префікс E і унікальний номер, наприклад E5.

## **Порядок виконання роботи**

1. Ознайомитися з літературою [1, 2, 12].
2. Відкрити модель, створену в результаті розщеплення вихідної моделі і збережену в файлі «dlvr\_dfd3.bp1».
3. Виконати декомпозицію моделі, вказавши при цьому тип моделі – DFD і кількість робіт в декомпозированої діаграмі - 7.
4. Зформувати діаграму (рис. 2.4).

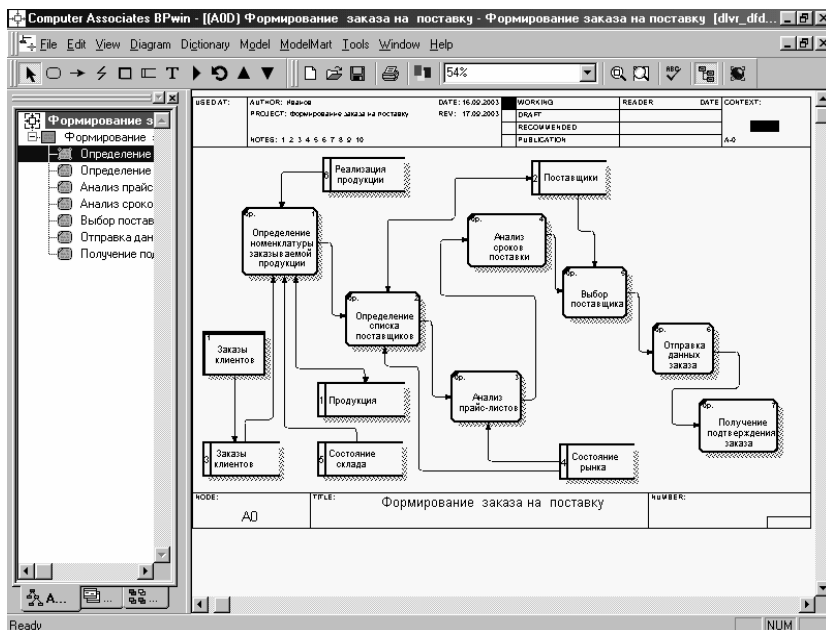


Рис. 2.4. Поеднання робіт на діаграмі графічними стрілками.

- 4.1. Видалити з діаграми всі граничні стрілки.
- 4.2. На діаграмі ввести такі назви робіт: «Визначення номенклатури продукції, що замовляється», «Визначення списку постачальників», «Аналіз прайс-листів», «Аналіз строків поставки», «Вибір постачальника», «Відправлення даних замовлення», «Отримання підтвердження замовлення».
- 4.3. Внести на діаграму зовнішнє посилання «Замовлення клієнтів» шляхом виклику «Зовнішнє завдання Інструмент» з панелі інструментів, а потім клацнути по тому місцю діаграми, де потрібна розмістити зовнішнє посилання.

4.4. Розмістити на діаграмі сховища даних. Для цього клацнути мишею по кнопці «магазин даних інструментів» на панелі інструментів та місці діаграми, де буде розміщено сховище даних. Створити сховища даних з назвами «Продукція», «Постачальники», «Замовлення клієнтів», «Стан ринку», «Стан складу», «Реалізація продукції».

4.5. Створити внутрішні стрілки і зв'язати їх на діаграмі. За необхідністю створити двонаправлені стрілки. Для цього в меню вибрати пункт «Стиль» і у вікні «Стрілка Властивості», у розділі «Тип» встановити перемикач в положення «двобічна» і зберегти діаграму.

### **Оформлення результатів роботи**

Результати лабораторної роботи оформлюються у вигляді звіту обов'язковими елементами, якого є: титульний аркуш; формулювання мети та завдання на роботу; основна частина; висновки.

Основна частина роботи повинна містити етапи вирішення індивідуального завдання шляхом аналізу досягнутих результатів в обраній предметній області.

Висновки повинні містити результати виконаної роботи.

### **Індивідуальні завдання**

Побудувати діаграму потоків даних для бізнес-процесів відповідно до списку пропонованого в лабораторній роботі 3 або узгодити їх вибір з викладачем.

### **Запитання для самоперевірки**

1. В чому заключається відмінність діаграм *IDEF0* і *DFD*?
2. Назвати основні компоненти діаграми *DFD*.
3. В яких випадках доцільно будувати діаграму *DFD*.
4. В чому відмінність побудови стрілок на *DFD* від *IDEF0*?
5. Пояснити позначення компонентів діаграми *DFD*.
6. Як згенерувати звіт по діаграмі *DFD*?
7. Пояснити призначення сховищ та зовнішніх суттностей.

## **Модль 3. Реалізація об'єктно-орієнтованих програмних систем**

### **Лабораторна робота 3.1. ПОБУДОВА ДІАГРАМИ ВАРІАНТІВ ВИКОРИСТАННЯ В СЕРЕДОВИЩІ IBM RATIONAL ROSE**

**Мета:** побудова use-case діаграм в середовищі IBM Rational Rose.

**Завдання:** побудувати діаграму використання інформаційної системи.

#### **Основні теоретичні відомості**

Розробка специфікації, проектування та документування інформаційних систем графічними засобами (діаграмами) відбувається за допомогою уніфікованої мови (Unified Modeling Language, UML).

Базовий елемент мови – діаграма, що являє собою графічне подання елементів системи графом з вершинами (сутностями) і ребрами (відносинами).

Діаграма прецедентів вважається основним видом діаграми візуального моделювання. Основні елементи діаграми прецедентів є ектор, прецедент і відносини між ними.

*Ектор* – це сутність, з якою взаємодіє система в процесі своєї роботи. В якості ектора можуть виступати люди, зовнішня система, підсистема, клас і т. ін.

*Прецедент* (use case) – подання послідовних подій, виконуваних системою, які ведуть до спостережуваного ектором результату. Прецеденти на діаграмі позначаються еліпсами. Вони розміщуються всередині прямокутника, який зветься рамками системи, контекстом або просто системою. У верхній частині прямокутника вказується назва модельованої системи.

Ектор і прецеденти зв'язані між собою відносинами. На діаграмі прецедентів використовуються відносини асоціації, включення, розширення, успадкування та коментарі. Асоціації між ектором і прецедентом завжди бінарні, представляють відносини типу "один до одного". Подання кратності відносин між прецедентами здійснюється відношенням включення.

*Включення* означає, що деяка точка базового прецеденту містить поведінку іншого прецедента. Зображується включення залежністю (пунктирна лінія зі стрілкою) зі стереотипом «include». При цьому стрілка спрямована у бік включаємого прецедента.

*Розширення* доповнює прецедент іншими прецедентами, тобто додає у вихідний прецедент послідовність дій, що міститься в іншому прецеденті. Зображується розширення залежністю (пунктирна лінія зі стрілкою) зі стереотипом «extend».

*Узагальнення* - це відношення між предком і нащадком, і стрілка завжди вказує на предка. Зображується узагальнення лінією з "незафарбованою" трикутною стрілкою на кінці.

У відношенні спадкування, можуть перебувати між собою пакети, ектор, прецеденти. Єдине допустиме відношення між екторами – генералізація (успадкування). Коментарі з'єднуються пунктирною лінією без стрілки.

### **Порядок виконання роботи**

Пропонується ознайомитися з літературою [1, 2, 14-16] та підготуватися до розробки інформаційної системи «банкомат». Виконати послідовність дій 1-13, за їх результатами отримати діаграму, представлену на рис. 3.1.

1. *Відкрити діалогове вікно* налаштування спеціальних панелей інструментів для діаграм в середовищі IBM Rational Rose 2006 одним із способів:

- з головного меню: Tools→ Options (Інструменти→Параметри);
- через вкладку Toolbars (Панелі інструментів) і натиснути відповідну кнопку Use Case diagram в групі опцій Customize Toolbars (Налаштування панелей інструментів);
- за допомогою операції контекстного меню Customize (Налаштування) при позиціонуванні курсора на спеціальній панелі інструментів. Проект назвати «ATMmodel».

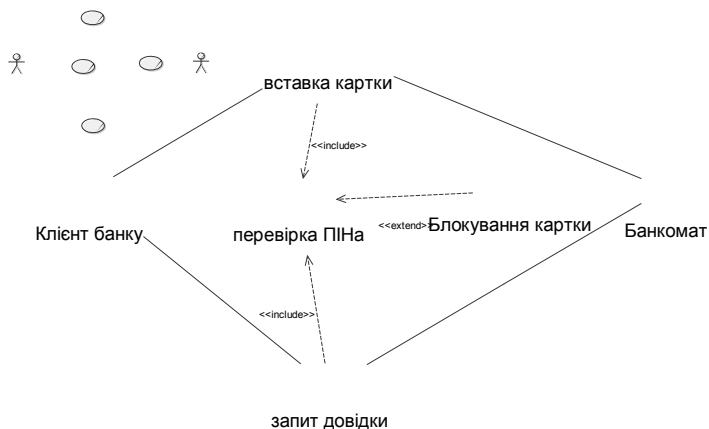


Рис. 3.1. Діаграма використання проекту «Банкомат»

2. *Додати Ектора* на діаграму варіантів використання і відредагувати його властивості шляхом зміни імені, заданого програмою за умовчанням «NewClass» на «Клієнт Банкомату». Змінити його стереотип і додати текст документації. Для зміни стереотипу у вкладеному списку Stereotype потрібно вибрати рядок «Business Actor» (бізнес-актор). Для додавання тексту документації в секцію «Documentation» слід ввести текст: «Будь-яка фізична особа, яка користується послугами банкомата» і натиснути кнопку «Apply» або «OK».

3. *Додати і відредагувати варіант використання.* Замінити пропонуване програмою ім'я прецеденту за замовчуванням «NewUseCase» на «Зняття готівки по кредитній картці». Також змінити стереотип у вкладеному списку «Stereotype», вибравши рядок «Business Use Case». У секцію «Documentation» ввести текст: «Основний варіант використання для розроблюваної моделі банкомату» і закрити діалогове вікно.

4. *Додати асоціацію між актором і варіантом використання* на діаграму. Зробити спрямовану асоціацію ненаправленою, для чого слід скористатися діалоговим вікном властивостей асоціації. Подвійним клацанням відкрити це вікно на зображенні лінії асоціації на діаграмі, після чого прибрати позначку рядка вибору «Navigable» на вкладці «Role A Detail».

5. Додати варіант використання з ім'ям «Перевірка ПІН-коду». Після цього за допомогою лівої кнопки миші натиснути кнопку із зображенням піктограми залежності на спеціальній панелі інструментів, відпустити ліву кнопку миші, клацнути лівою кнопкою миші на зображенні варіанта використання «Зняття готівки по кредитній картці» і відпустити її на зображенні варіанта використання «Перевірка ПІН-коду». Для доданої відносини залежності слід додатково вказати текстовий стереотип «include».
6. Додати актора з ім'ям «Банк», для якого вибрати стереотип «Service», що означає, що банкомат використовує деякі послуги Банку в якості сервісу.
7. Додати варіант використання «Отримання довідки про стан рахунку», для якого вибрати стереотип «Business Use Case».
8. Додати варіант використання «Блокування кредитної картки».
9. Додати спрямовану асоціацію від бізнес-актора «Клієнт Банкомату» до варіанту використання «Отримання довідки про стан рахунку».
10. Додати спрямовану асоціацію від варіанту використання «Зняття готівки по кредитній картці» до сервісу «Банк».
11. Додати спрямовану асоціацію від варіанту використання «Отримання довідки про стан рахунку» до сервісу «Банк».
12. Додати відношення залежності зі стереотипом «include», спрямоване від варіанту використання «Отримання довідки про стан рахунку» до варіанту використання «Перевірка ПІН-коду».
13. Додати відношення залежності зі стереотипом «extend», спрямоване від варіанту використання «Блокування кредитної картки» до варіанту використання «Перевірка ПІН-коду».

Після закінчення роботи над проектом виконану роботу необхідно зберегти через меню File→ Save або File→ Save As у файлі проекту з розширенням «.MDL».

### **Оформлення результатів роботи**

Результати лабораторної роботи оформлюються у вигляді звіту обов'язковими елементами, якого є: титульний аркуш; формулювання мети та завдання на роботу; основна частина; висновки. Зразок титульного листа наведений у Додатку 1.

Основна частина роботи повинна містити етапи вирішення індивідуального завдання шляхом аналізу досягнутих результатів в обраній предметній області.

Висновки повинні містити результати виконаної роботи.

### **Індивідуальні завдання**

Розробити use-case діаграми для систем, список яких дається або узгодити вибір бізнес-процесів з викладачем.

1. Електронна бібліотека.
2. Система керування доставкою вантажів.
3. Система обслуговування хворих.
4. Система покупки авіаквітків.
5. Інтернет-магазин.
6. Система обслуговування автівок.
7. Система доставки продуктів.
8. Система оплати платежів.
9. Система електронних переказів.
10. Система ремонту побутової техніки.
11. Система надання послуг мобільного зв'язку.

### **Запитання для самоперевірки**

1. Як нефункціональні вимоги відображаються на діаграмах прецедентів?
2. Які способи зображення ектора вам відомі?
3. В які відносини можуть вступати ектори між собою?
4. У чому полягає сенс відносин включення та розширення?
5. Що розуміють під точкою розширення?
6. Перелічити причини використання прецедентів.
7. Як прецеденти застосовують в прямому і зворотному проектуванні?

## **Лабораторна робота 3.2**

### **РОЗРОБКА ДІАГРАМИ КЛАСІВ В СЕРЕДОВИЩІ IBM RATIONAL ROSE**

**Мета:** навчитися будувати діаграми класів в середовищі IBM Rational Rose.



**Завдання:** побудувати діаграму класів для заданої інформаційної системи.

### Основні теоретичні відомості

*Діаграма класів* – це набір статичних, декларативних елементів моделі. Елементами діаграми класів можуть бути інтерфейси, пакети, відносини, об'єкти та зв'язки.

*Клас* на діаграмі зображується у вигляді прямокутника, розділеного горизонтальними лініями на три секції. Перша секція містить назву класу. Як правило, ім'я класу складається з одного, максимум двох слів. Друга містить перелік атрибутів класу, які характеризують той чи інший об'єкт цього класу в моделі предметної області. Третя містить перелік операцій, що відображають його поведінку в моделі предметної області.

*Атрибут* – змістовна характеристика класу, що описує множину значень, які можуть приймати окремі об'єкти цього класу.

*Операція* – це сервіс, що надається кожним екземпляром або об'єктом класу на вимогу своїх клієнтів, в якості яких можуть виступати інші об'єкти, в тому числі і екземпляри даного класу.

*Інтерфейс* – іменована множина операцій, які характеризують поведінку окремого елемента моделі. Інтерфейс в контексті мови UML є спеціальним випадком класу, у якого є операції, але відсутні атрибути. Для позначення інтерфейсу використовується спеціальний графічний символ коло або стандартний спосіб – прямокутник класу зі стереотипом «interface».

**Таблиця 3.1**

*Символи видимості атрибуту класу*

Символ	Значення
+	public - відкритий доступ
-	private - тільки для операцій того ж класу
#	protected - тільки для операцій того ж класу і класів, що створюються на його основі

Відношення *залежність* виникає тоді, коли реалізація класу одного об'єкта залежить від специфікації операцій класу іншого об'єкта.

*Асоціація* – це зв'язок між об'єктами. Асоціація може мати ім'я, яке показує природу відносин між об'єктами, при цьому в імені може вказуватися напрямок читання зв'язку за допомогою трикутного маркера. Однонапрямлена асоціація може зображуватися стрілкою. Крім напрямку, на асоціації в діаграмі вказуються ролі, які кожен клас відіграє в даному відношенні, і кратність, тобто кількість об'єктів, пов'язаних відношенням. Якщо асоціація об'єднує три і більше класів, то її називають *n*-арною і зображуються ромбом на перетині ліній. Асоціація з агрегуванням є складним відношенням між класами типу "частина-ціле".

Розрізняють асоціації агрегації і композиції. Проста агрегація припускає, що частини, які відділені від цілого, можуть продовжувати своє існування незалежно від нього. Під композитним агрегуванням розуміється ситуація, коли ціле володіє своїми частинами і їх час життя відповідає часу життя цілого, тобто незалежно від цілого частини існувати не можуть.

Графічно відношення агрегації зображується суцільною лінією, один з кінців якої являє собою незафарбований усередині ромб, а відношення композиції – суцільною лінією з зафарбованим всередині ромбом. Цей ромб вказує на той клас, який являє собою або клас-контейнер або клас-комполит.

Відношення узагальнення позначається суцільною лінією з трикутною стрілкою на одному з кінців. Стрілка вказує на більш загальний клас (клас-предок або суперклас), а її початок – на більш спеціальний клас (клас-нащадок або підклас).

### **Порядок виконання роботи**

Ознайомитися з літературою [1, 2, 14-16], виконати послідовність дій 1-32, за їх результатами отримати діаграму у вигляді, який показано на рис. 3.2.

1. Викликати діаграму класів та змінити запропоноване за замовчуванням ім'я діаграми «Main» на «Діаграма класів АТМ», а ім'я доданого класу на діаграму – на «Транзакція Банкомату». Імена класів, їх атрибутів і операцій для більшої наочності і розуміння подаються українською мовою з пробілами.



Рис. 3.2. Діаграма класів для варіанта використання «Зняття грошей»

2. Додати клас «Транзакція Банкомату» на діаграму класів. Для класу Транзакція Банкомату уточнити його призначення в моделі за допомогою вказання стереотипу і пояснювального тексту в документації. Для цього відкрити діалогове вікно специфікації властивостей цього класу і на вкладці «General» вибрати з вкладеного списку «Stereotype» стереотип «entity». У секцію документації даного класу можна ввести пояснювальний текст: «Використовується для збереження інформації про виконані банкоматом транзакції» і натиснути кнопку «Apply» або «OK» для збереження результатів редагування.

3. Додати на діаграму другий клас «Контролер Банкомату», для якого у вікні специфікації властивостей виберемо стереотип «control», а для документації вводим текст: «Реалізує логіку функціонування банкомату». При цьому атрибути та операції у даного класу будуть відсутні.

4. Додати на діаграму третій клас з ім'ям «Пристрій читання картки», для якого у вікні специфікації властивостей виберемо стереотип «boundary», який означає, що даний клас знаходиться на межі модельованої системи, в якості якої розглядається модель банкомату. Після цього в секцію документації даного класу можна ввести пояснювальний текст: «Встановлюється на банкоматі».

5. Додати клас з ім'ям «ІКонтроллер Банку», для якого вибрати стереотип «Interface», що означає, що банкомат користується послугами «Банку» при обробці своїх транзакцій. Першою літерою в імені цього класу є англійське «I», яке служить в мові UML для вказання інтерфейсу.

6. Задаємо операції класів. Для класу «Транзакція Банкомату» в якості імені першої операції задаємо: створити нову транзакцію. Для операції «створити нову транзакцію ()» вибрати з вкладеного списку квантор видимості «public». У секцію документації даної операції класу вводиться пояснювальний текст: «Викликається після того, як кредитна картка вставлена в Пристрій читання картки» і натиснути кнопку «Apply» або «OK».

Додати в специфікацію класу «Транзакція Банкомату» наступні атрибути і операції.

- для значення ПІН-коду картки встановити квантор видимості «public». Атрибутом значення обрати тип «Integer», а в секцію документації атрибута ввести пояснювальний текст: «Пристрій читання картки зчитує значення цього атрибута з кредитної картки клієнта».

- для атрибута «введений ПІН-код» встановити квантор видимості «public». Для цього атрибута вибрати тип Integer, а в секцію документації атрибута ввести пояснювальний текст: «Значення цього атрибута вводиться клієнтом з клавіатури банкомату».

- для атрибута «сума готівки» встановити квантор видимості «public». Для цього атрибута обрати тип «Currency», а в секцію документації атрибута ввести пояснювальний текст: «Значення цього атрибута вводиться клієнтом з клавіатури банкомату».

- для операції «перевірити правильність ПІН-коду ()» встановити квантор видимості «public». В якості типу що повертається результату для цієї операції слід вибрати тип Boolean (логічний), а в секцію її документації вводиться пояснювальний текст: "Викликається після того, як клієнт ввів значення ПІН-коду з клавіатури банкомату".

- для операції «завершити транзакцію ()» встановити квантор видимості public. У секцію її документації ввести пояснювальний текст: «Викликається після завершення всіх дій банкомату з обслуговування клієнта».

7. Створюємо напрямлену асоціацію між класом «Контролер Банкомату» і класом «Транзакція Банкомату». Викликаємо діалогове вікно специфікації властивостей асоціації «Association Specification» шляхом виділення лінії асоціації на діаграмі класів або в браузері проекту і подвійного клацання на ній лівою кнопкою миші. Задамо її кратність рівну 1 для кінця асоціації у класі «Контролер Банкомату». Для цього у вікні специфікації властивостей асоціації переходимо на вкладку «Role B Detail» і вибираємо значення 1 з вкладеного списку «Multiplicity». Аналогічним чином задаємо кратність рівну 1..n для кінця асоціації у класі «Транзакція Банкомату», для чого на вкладці «Role A Detail» і вибираємо значення 1..n з вкладеного списку «Multiplicity». Це означає, що кожен об'єкт класу «Контролер Банкомату» може бути пов'язаний з одним або декількома об'єктами класу «Транзакція Банкомату».

Додаємо напрямлену асоціацію між класом «Контролер Банкомату» і додатково створеним абстрактним класом «Контролер». Останній клас може бути призначений для специфікації системних атрибутів і операцій, необхідних при виконанні відповідної програми. На абстрактний характер класу вказує написання курсивом його імені, а для специфікації даної властивості класу необхідно на вкладці «Detail» вікна специфікації властивостей класу «Контролер» виставити позначку в рядку вибора «Abstract».

8. Для класу «Інтерфейс Банку» додати операцію «перевірити ідентифікатор картки» (ідентифікатор картки – Integer) з квантором видимості «public». Для результату цієї операції вибрати тип «Boolean», а для цілочисленого аргументу задати ідентифікатор картки. Для завдання аргументу необхідно перейти на вкладку «Detail» вікна специфікації властивостей даної операції і після додавання аргументу за допомогою операції контекстного меню «Insert» ввести ім'я аргументу і його тип «Integer» у відповідні поля введення.

9. Для класу «Інтерфейс Банку» додати операцію «відкрити рахунок клієнта» (ідентифікатор картки – Integer) з квантором видимості «public». В якості цілочисленого аргументу цієї операції слід задати «ідентифікатор картки».

10. Для класу «Інтерфейс Банку» додати операцію «перевірити баланс клієнта» (ідентифікатор картки – Integer, тип введеної суми готівки – Currency) з квантором видимості «public». В якості типу що повертається результату для цієї операції слід вибрати тип «Boolean». В якості першого цілочисленного аргументу цієї операції слід задати «ідентифікатор картки», а в якості другого аргументу – «введена сума готівки» встановити тип «Currency».

11. Для класу «Інтерфейс Банку» додати операцію «зменшити рахунок клієнта» (ідентифікатор картки – Integer, введена сума готівки – Currency) з квантором видимості «public». В якості типу результату, що повертається, для цієї операції слід вибрати тип «Boolean». В якості першого цілочисленного аргументу цієї операції задається «ідентифікатор картки», а другому аргументу – «введена сума готівки» вказати тип «Currency».

12. Класу «Пристрій читання картки» додати операцію «прочитати ідентифікатор картки ()» з квантором видимості «public». В якості типу результату, що повертається, для цієї операції слід вибрати тип «Integer», а в секцію документації даної операції вводиться пояснювальний текст: «Викликається після того, як кредитна картка вставлена в Пристрій читання картки».

13. Для класу «Пристрій читання картки» додати операцію «прочитати ПІН-код ()» з квантором видимості «public». В якості типу що повертається результату для цієї операції вибрати тип «Integer», а в секцію документації даної операції ввести пояснювальний текст: «Викликається після того, як кредитна картка вставлена в Пристрій читання картки».

14. Для класу «Пристрій читання картки» додати операцію «повернути кредитну картку ()» з квантором видимості «public». У секцію документації даної операції ввести пояснювальний текст: «Викликається після завершення транзакції».

15. Для класу «Пристрій читання картки» додати операцію «блокувати кредитну картку ()» з квантором видимості «public». У секцію документації даної операції вводиться пояснювальний текст: «Викликається після того, як встановлений факт втрати кредитної картки власником».

16. Додати клас з ім'ям «Екран Банкомату», для якого вибрати стереотип «boundary». Даний клас також знаходиться на межі модельованої системи, на що і вказує цей стереотип. У секцію

документації даного класу слід ввести пояснювальний текст: «Встановлюється на банкоматі».

17. Для класу «Екран Банкомату» додати операцію «показати меню опцій ()» з квантором видимості «public».

18. Для класу «Екран Банкомату» додати операцію «показати меню зняття суми ()» з квантором видимості «public».

19. Додати клас з ім'ям «клавіатура банкомату», для якого вибрати стереотип «boundary». У секцію документації даного класу ввести пояснювальний текст: «Встановлюється на банкоматі».

20. Для класу «клавіатура банкомату» додати операцію «ввести ПІН-код ()» з квантором видимості «public». В якості типу результату, що повертається, для цієї операції вибрати тип «Integer», а в секцію документації даної операції ввести пояснювальний текст: «Викликається після того, як клієнт ввів значення ПІН-коду з клавіатури».

21. Для класу «клавіатура банкомату» додати операцію «ввести тип транзакції ()» з квантором видимості «public». В якості типу результату, що повертається для цієї операції вибрати тип «Boolean», а в секцію документації даної операції слід ввести пояснювальний текст: «Повертає значення *Істина*, якщо клієнт вибирає зняття готівки, і значення *Неправда*, якщо клієнт вибирає отримання довідки про стан рахунку».

22. Для класу «клавіатура банкомату» додати операцію «ввести суму зняття готівки ()» з квантором видимості «public». В якості типу результату, що повертається, для цієї операції вибрати тип «Currency», а в секцію документації даної операції ввести пояснювальний текст: «Викликається після того, як клієнт ввів значення суми, що знімається з клавіатури».

23. Додати клас з ім'ям «Пристрій видачі готівки», для якого вибрати стереотип «boundary». У секцію документації даного класу ввести пояснювальний текст: «Встановлюється на банкоматі».

24. Для класу «Пристрій видачі готівки» додати операцію «видати готівку ()» з квантором видимості «public». У секцію документації даної операції ввести пояснювальний текст: «Викликається після того, як виконано зняття запитаної клієнтом суми з рахунку».

25. Додати клас з ім'ям «Пристрій видачі готівки», для якого вибрати стереотип «boundary». У секцію документації даного класу вводиться пояснювальний текст: «Встановлюється на банкоматі».

26. Для класу «Принтер Банкомату» додати операцію «роздрукувати чек ()» з квантором видимості «public». У секцію документації даної операції вводиться пояснювальний текст: «Викликається по додатковому запиту клієнта».

27. Додати напрямлену асоціацію від класу «Контролер Банкомату» до класу «Пристрій читання картки». Значення кратності кінцівок цієї асоціації встановити рівним 1.

28. Додати напрямлену асоціацію від класу «Контролер Банкомату» до класу «Принтер Банкомати». Встановити значення кратності кінцівок цієї асоціації рівним 1.

29. Додати напрямлену асоціацію від класу «Контролер Банкомату» до класу «клавіатура банкомату». В якості кратності кінцівок цієї асоціації встановити значення 1.

30. Додати напрямлену асоціацію від класу «Контролер Банкомату» до класу «Пристрій видачі готівки». Позначити кратність кінцівок цієї асоціації значенням 1.

31. Додати напрямлену асоціацію від класу «Контролер Банкомату» до класу «Екран Банкомату». В якості кратності кінцівок цієї асоціації встановити значення 1.

32. Додати напрямлену асоціацію від класу «Контролер Банкомату» до класу «Контроллер Банку». В якості кратності кінця цієї асоціації для першого класу встановити значення «0..n», а кратності кінця асоціації для другого класу встановити значення 1. В якості стереотипу даної асоціації вибрати з вкладеного списку значення «communicate». Застосування даного стереотипу означає, що між цими класами повинен існувати фізичний взаємозв'язок.

### **Оформлення результатів роботи**

Результати лабораторної роботи оформлюються у вигляді звіту обов'язковими елементами, якого є: титульний аркуш; формулювання мети та завдання на роботу; основна частина; висновки. Зразок титульного листа наведений у Додатку 1.

Основна частина роботи повинна містити етапи вирішення індивідуального завдання шляхом аналізу досягнутих результатів в обраній предметній області.



Висновки повинні містити результати виконаної роботи.

### **Індивідуальні завдання**

Розробити діаграму класів для систем, список яких надано в лабораторній роботі 3.1 або узгодити вибір бізнес-процесів з викладачем.

### **Запитання для самоперевірки**

1. Які принципи знаходяться в основі ООП?
2. Що розуміють під *n*-арною асоціацією?
3. В чому різниця між агрегацією та композицією?
4. Що таке асоціація класу?
5. Як на діаграмі класів відображаються відношення агрегації та композиції?
6. З яких основних елементів складається діаграма класів?
7. Як записується атрибут на діаграмі класів?

### **Лабораторна робота 3.3**

#### **ДІАГРАММИ КООПЕРАЦІЇ ТА ПОСЛІДОВНОСТЕЙ В СЕРЕДОВИЩІ IBM RATIONAL ROSE**

**Мета:** вивчити порядок побудови діаграм кооперації та послідовностей в середовищі IBM Rational Rose.

**Завдання:** побудувати діаграми *кооперації та послідовностей* стосовно розроблюваного проекту інформаційної системи.

### **Основні теоретичні відомості**

*Діаграма взаємодії* показує потік повідомлень між об'єктами системи та основні асоціації між ними.

Різновидами діаграми взаємодії є діаграми кооперації та послідовностей, які в контексті мови UML описують динамічний аспект взаємодії об'єктів при реалізації окремих варіантів використання.

*Діаграма кооперації* – діаграма взаємодій, в якій основний акцент зроблено на структурній організації об'єктів, що посиляють і отримують повідомлення.

*Діаграма послідовностей* – відображає часові особливості передачі і прийому повідомлень об'єктами.

На діаграмі кооперації розміщуються об'єкти, що представляють собою екземпляри класів, зв'язки між ними, які в свою чергу є екземплярами асоціацій та повідомлення. Зв'язки доповнюються стрілками повідомлень, при цьому показуються тільки ті об'єкти, які беруть участь у реалізації модельованої кооперації. Далі, як і на діаграмі класів, показуються структурні відносини між об'єктами у вигляді різних з'єднувальних ліній. Зв'язки можуть доповнюватися іменами ролей, які грають об'єкти в даному взаємозв'язку. Зображуються також динамічні взаємозв'язки – потоки повідомлень у формі стрілок з вказівкою напрямку поряд зі сполучними лініями між об'єктами, при цьому задаються імена повідомлень та їх порядкові номери в загальній послідовності повідомлень.

Можливі такі варіанти записів повного імені об'єкта:

- о: С – об'єкт із власним ім'ям о, екземпляр класу С;
- : С – анонімний об'єкт, екземпляр класу С;
- о: (або просто о) – об'єкт-сирота з власним ім'ям о;
- о / R: С – об'єкт із власним ім'ям о, екземпляр класу С, який грає роль R;
- / R: С - анонімний об'єкт, екземпляр класу С, який грає роль R;
- о / R – об'єкт - сирота з власним ім'ям о, який грає роль R;
- / R – анонімний об'єкт і одночасно об'єкт сирота, який грає роль R.

У контексті мови UML всі об'єкти діляться на дві категорії: пасивні та активні. *Пасивний* об'єкт оперує тільки даними і не може ініціювати діяльність з управління іншими об'єктами. Однак пасивні об'єкти можуть посылати сигнали в процесі виконання запитів, які вони обробляють. На діаграмі кооперації пасивні об'єкти зображуються звичайним чином без додаткових стереотипів.

*Активний* об'єкт має власний процес управління і може ініціювати діяльність з управління іншими об'єктами.

*Мультиоб'єкт* являє собою множину анонімних об'єктів, які можуть бути утворені на основі одного класу.

На діаграмі кооперації мультиоб'єкт використовується для того, щоб показати операції і сигнали, які адресовані всій множині анонімних об'єктів.

Повідомлення на діаграмі кооперації зображаються додатковими стрілками поруч з відповідним зв'язком або роллю асоціації. Напрямок стрілки вказує на одержувача повідомлення. Зовнішній вигляд стрілки повідомлення має певний сенс. На діаграмах кооперації може використовуватися один з трьох типів стрілок для позначення повідомлень

- Суцільна лінія з трикутною стрілкою позначає виклик процедури (операції) або передачу потоку управління.
- Суцільна лінія з V-подібною стрілкою позначає асинхронне повідомлення у простому потоці управління.
- Пунктирна лінія з V-подібною стрілкою позначає повернення з виклику процедури.

В UML визначені наступні стереотипи повідомлень:

- «call» – повідомлення, що вимагає виклику операції або процедури об'єкта-одержувача;
- «return» – повідомлення, яке повертає значення виконаної операції або процедури об'єкту, що її викликав;
- «create» – повідомлення, що вимагає створення іншого об'єкта для виконання певних дій;
- «destroy» – повідомлення, що вимагає знищити відповідний об'єкт;
- «send» – позначає посилення об'єкту сигналу, який асинхронно ініціюється одним об'єктом і приймається (перехоплюється) іншим.

### **Порядок виконання роботи**

Ознайомитися з літературою [1, 2, 14-16]. Діаграма кооперації, що описує реалізацію типового перебігу подій варіанта використання «Зняття готівки по кредитній картці» для проекту системи управління банкоматом, показана на рис. 3.3. Для того, щоб її отримати виконати послідовність дій 1-22.

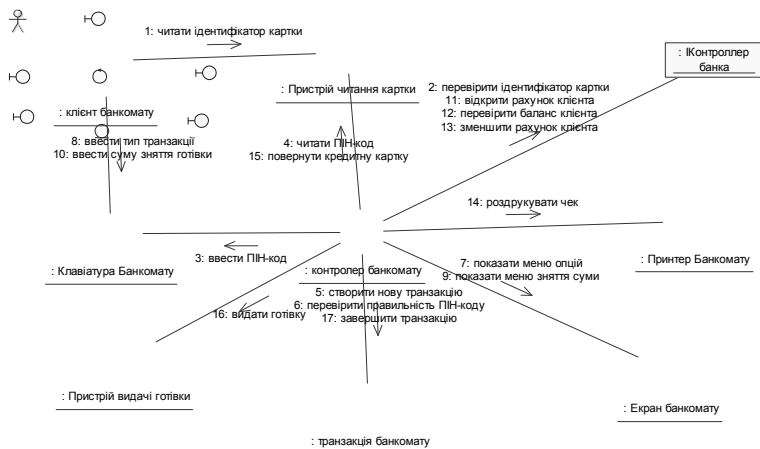


Рис. 3.3. Діаграма кооперації для типового перебігу подій варіанта використання «Зняття готівки по кредитній картці»

1. Активізувати робоче вікно діаграми кооперації в програмі IBM Rational Rose можна декількома способами:

- клацнути на кнопці із зображенням діаграми взаємодії на стандартній панелі інструментів і вибрати для побудови нову діаграму кооперації;
- виконати операцію головного меню: Browse Interaction Diagram (Браузер Діаграма взаємодії) і вибрати для побудови нову діаграму кооперації;
- виконати операцію контекстного меню: New Collaboration Diagram (Нова Діаграма кооперації) для логічного представлення або подання варіантів використання в браузері проекту.

2. Ознайомитися з кнопками панелі інструментів.

3. Задаємо ім'я діаграмі «Зняття готівки по кредитній картці».

4. У браузері проекту виділяємо необхідний клас і, утримуючи ліву кнопку миші, перетягнути зображення піктограми класу з браузера на вільне місце робочого листа діаграми кооперації. У вікні властивостей для об'єкта обраного класу можна задавати: власне ім'я об'єкта, особливості його реалізації і множинність екземплярів.

5. Встановити зв'язок між об'єктом класу «Клієнт Банкомату» і об'єктом класу «Пристрій читання картки». Для цього потрібно

натиснути кнопку із зображенням зв'язку на спеціальній панелі інструментів, клацнути лівою кнопкою миші на зображенні одного об'єкта на діаграмі і відпустити її на зображенні іншого об'єкта. При необхідності можна задати ім'я зв'язку за допомогою діалогового вікна специфікації властивостей даної зв'язку.

Крім імені повідомлення можна також задати стереотип синхронізації і частоту передачі. Для цієї мети слід скористатися діалоговим вікном специфікації властивостей повідомлень, які можна відкрити подвійним клацанням на імені повідомлення в списку розглянутої вкладки «Messages» вікна специфікації властивостей зв'язку.

6. Додати повідомлення за допомогою діалогового вікна властивостей зв'язків. Для цього подвійним клацанням на лінії зв'язку викликається вікно її властивостей і розкривається вкладка «Messages». Після цього слід виконати операцію контекстного меню «Insert To», де вибрати одну з операцій цільового класу для специфікації імені повідомлення.

Для першого повідомлення слід вибрати операцію «прочитати ідентифікатор картки ()». Після вибору операції для даного повідомлення воно додається в список повідомлень даного зв'язку, а поруч з лінією зв'язку на діаграмі кооперації з'явиться стрілка з номером і ім'ям цього повідомлення.

7. Додати об'єкти класів з іменами «Контролер Банкомату», «Транзакція Банкомату», «клавіатура банкомату», «Екран Банкомату», «Принтер Банкомату», «Пристрій видачі готівки та Інтерфейс Банку».

8. Додати зв'язки, що з'єднують об'єкти класів з іменами «Контролер Банкомату» з «Пристроєм читання картки», «Контролер Банкомату» з «Транзакція Банкомату», «Контролер Банкомату» з «Клавіатура Банкомату», «Контролер Банкомату» з «Екран банкомату», «Контролер Банкомату» з «Принтером Банкомату», «Контролер Банкомату» з «Пристрій видачі готівки» та «Контролер Банкомату» з «Інтерфейс банку».

9. Додати повідомлення «перевірити ідентифікатор картки (Integer)», спрямоване від об'єкта класу «Контролер Банкомату» до об'єкта класу «Інтерфейс Банку».

10. Додати повідомлення «ввести ПІН-код ()», спрямоване від об'єкта класу-актора «Клієнт Банкомату» до об'єкта класу «клавіатура банкомату».

11. Додати повідомлення «читати ПІН-код ()», спрямоване від об'єкта класу «Контролер Банкомату» до об'єкта класу «Пристрій читання картки».

12. Додати повідомлення «створити нову транзакцію ()», спрямоване від об'єкта класу «Контролер Банкомату» до об'єкта класу «Транзакція Банкомати».

13. Додати повідомлення «перевірити правильність ПІН-коду ()», спрямоване від об'єкта класу «Контролер Банкомату» до об'єкта класу «Транзакція Банкомати».

14. Додати повідомлення «показати меню опцій ()», спрямоване від об'єкта класу «Контролер Банкомату» до об'єкта класу «Екран Банкомати».

15. Додати повідомлення «ввести тип транзакції ()», спрямоване від об'єкта класу-актора «Клієнт Банкомату» до об'єкта класу «клавіатура банкомату».

16. Додати повідомлення «показати меню зняття суми ()», спрямоване від об'єкта класу «Контролер Банкомату» до об'єкта класу «Екран Банкомати».

17. Додати повідомлення «ввести суму зняття готівки ()», спрямоване від об'єкта класу-актора «Клієнт Банкомату» до об'єкта класу «клавіатура банкомату».

18. Послідовно додати 3 повідомлення «відкрити рахунок клієнта (Integer)», «перевірити баланс клієнта (Integer, Currency)» і «зменшити рахунок клієнта (Integer, Currency)», спрямовані від об'єкта класу «Контролер Банкомату» до об'єкта класу «Інтерфейс Банку».

19. Додати повідомлення «роздрукувати чек ()», спрямоване від об'єкта класу «Контролер Банкомату» до об'єкта класу «Принтер Банкомати».

20. Додати повідомлення «повернути кредитну картку ()», спрямоване від об'єкта класу «Контролер Банкомату» до об'єкта класу «Пристрій читання картки».

21. Додати повідомлення «видати готівку ()», спрямоване від об'єкта класу «Контролер Банкомату» до об'єкта класу «Пристрій видачі готівки».

22. Додати повідомлення «завершити транзакцію ()», спрямоване від об'єкта класу «Контролер Банкомату» до об'єкта класу «Транзакція Банкомату».

23. Активізувати діаграму послідовності натисненням на клавишу F5.

### **Оформлення результатів роботи**

Результати лабораторної роботи оформлюються у вигляді звіту обов'язковими елементами, якого є: титульний аркуш; формулювання мети та завдання на роботу; основна частина; висновки.

Основна частина роботи повинна містити етапи вирішення індивідуального завдання шляхом аналізу досягнутих результатів в обраній предметній області.

Висновки повинні містити результати виконаної роботи.

### **Індивідуальні завдання**

Розробити діаграму кооперації та послідовностей для систем, список яких надано в лабораторній роботі 3.1 або узгодити вибір бізнес-процесів з викладачем.

#### **Запитання для самоперевірки**

1. В чому призначення діаграми кооперації?
2. У чому різниця між активними і пасивними об'єктами?
3. Чим асинхронне повідомлення відрізняється від синхронного?
4. Що таке мультиоб'єкт?
5. Що таке композитний об'єкт і як він пов'язаний з поняттям кооперації?
6. Як можна уникнути ускладнення діаграми взаємодії з розгалуженим потоком управління?
7. Як показуються повідомлення на діаграмі кооперації?

### Лабораторна робота 3.4

## РОЗРОБКА ДІАГРАМИ СТАНІВ В СЕРЕДОВИЩІ IBM RATIONAL ROSE

**Мета:** засвоїти призначення та способи складання діаграми станів в середовищі IBM Rational Rose

**Завдання:** розробити діаграму станів для обраної інформаційної системи.

### Основні теоретичні відомості

*Діаграми станів* використовуються для опису поведінки окремих систем і підсистем та можуть бути застосовані для моделювання всіх можливих змін *станів* конкретних об'єктів. *Діаграма станів* є графом спеціального виду, який призначений для подання *кінцевого автомата*.

*Стан* на діаграмі зображується прямокутником зі скругленими вершинами. Цей *прямокутник* може бути розділений на дві секції горизонтальною лінією. Якщо зазначається лише одна секція, то в ній записується тільки ім'я *стану*. А якщо ні, то в першій з них записується ім'я *стану*, а в другий - *список* деяких внутрішніх дій або *переходів* у даному *стані*.

Дія приводить до зміни *стану* системи, і може бути реалізована шляхом передачі повідомлення об'єкту, модифікацією зв'язку або значення атрибута.

Кожна дія записується окремим рядком у форматі «мітка дії '/' вираз дії». *Мітка дії* вказує на обставини або умови, при яких буде виконуватися *діяльність*, визначена виразом дії. При цьому *вираз дії* може використовувати будь-які атрибути і зв'язки, що належать області імен або контексту моделюємого об'єкту.

Псевдостан (pseudo-state) – вершина кінцевого автомату, яка має форму стану, але не має поведінки. Прикладами псевдостанів, які визначені в мові *UML*, є початковий і кінцевий *стани*.

Перехід – відношення між двома станами, яке вказує на те, що об'єкт у першому стані повинен виконати певні дії і перейти в інший стан.

Спрацьовування переходу може залежати не тільки від настання події, але і від виконання певної умови, яка називається *сторожовою*. Об'єкт перейде з одного стану до іншого в тому



випадку, якщо відбулася зазначена подія і сторожова умова прийняла значення "істина". На діаграмі станів перехід зображується суцільною лінією зі стрілкою, яка виходить із вихідного стану і спрямована в цільовий стан.

Залежно від виду подій, що відбуваються, у мові UML розрізняють два типи переходів: тригерні та нетригерні.

Перехід називається *тригерним*, якщо його специфікує подія-тригер, пов'язана із зовнішніми умовами стосовно розглянутого стану. У цьому випадку поруч зі стрілкою тригерного переходу обов'язково вказується ім'я події у формі рядка тексту, що починається з рядкової букви.

Перехід називається *нетригерним*, якщо він відбувається по завершенню виконання діяльності в даному стані. Для них поруч зі стрілкою переходу не вказується ніякого імені події, а у вихідному стані повинна бути описана внутрішня діяльність, по завершенню якої відбудеться той або інший нетригерний перехід.

*Сторожова умова* – логічна умова, записується в прямих дужках, що і представляє собою булевий вираз, який приймає значення, що взаємно виключають "істина" або "неправда".

### **Порядок виконання роботи**

Ознайомитися з літературою [1, 2, 14-16], побудувати діаграму станів за прикладом розглянутої моделі банкомата, що показана на рис. 3.4. Для цього виконати послідовність дій 1-22.

1. Активізувати діаграму *станів* для обраного елемента моделі або моделюємої системи в цілому одним з наступних способів:

- на панелі інструментів «стандартна» вибрати подання і тип розроблювальної діаграми – нова діаграма станів;
- у браузері проекту виділити логічне подання або подання варіантів використання і виконати операцію контекстного меню: New → Statechart Diagram.
- Виконати операцію головного меню: Browse → State Machine Diagram, після чого слід вибрати подання і тип розроблювальної діаграми.

2. Вивчити призначення кнопок спеціальної панелі інструментів діаграми станів.



Рис. 3.4. Діаграма станів для модельованої поведінки банкомата

3. Задати ім'я діаграми станів моделі банкомата «Діаграма станів АТМ», а в секцію її документації ввести текст «Діаграма станів описує кінцевий автомат банкомата».

Ввести перший стан, змінити ім'я першого доданого стану «Newstate» на «Очікування картки» у вікні редактора або у вікні специфікації властивостей нового стану. Задати текстовий стереотип стану, визначити внутрішні дії на вході і виході, а також внутрішню діяльність. Ці властивості доступні для редагування на вкладці General і Actions. На вкладці Transitions визначати або редагувати переходи, які входять і виходять із розглянутого стану.

4. Додати на діаграму станів початковий стан і з'єднати його переходом зі станом «Очікування картки». Додати стани з іменами «Очікування введення ПІН-коду», «Перевірка ПІН-коду», «Очікування вибору клієнта», «Обробка запиту на зняття готівки», «Обробка запиту на одержання довідки», «Видача готівки», «Друк», «Повернення картки», «Завершення транзакції» і «фінальний стан».

5. Додати перехід «картка вставлена», спрямований від стану «Очікування картки» до стану «Очікування введення ПІН-коду».

6. Додати перехід «ПІН-код введений», спрямований від стану «Очікування введення ПІН-коду» до стану «Перевірка ПІН-коду».

7. Додати перехід «скасування транзакції», спрямований від стану «Очікування введення ПІН-коду» до стану «Повернення картки».

8. Додати перехід зі сторожовою умовою «[ПІН-код вірний]», спрямований від стану «Перевірка ПІН-коду» до стану «Очікування вибору клієнта».

9. Додати перехід зі сторожовою умовою «[ПІН-код невірний]», спрямований від стану «Перевірка ПІН-коду» до стану «Очікування введення ПІН-коду».

10. Додати перехід три невдачі з дією на переході «конфіскація картки», спрямований від стану «Перевірка ПІН-коду» до стану «Завершення транзакції». Для завдання дії на даному переході слід увести текст «конфіскація картки» в поле введення «Action» на вкладці «Detail» вікна специфікації властивостей даного переходу.

11. Додати перехід вибір суми зі сторожовою умовою «[сума введена]», спрямований від стану «Очікування вибору клієнта» до стану «Обробка запиту на зняття готівки».

12. Додати перехід «вибір довідки», спрямований від стану «Очікування вибору клієнта» до стану «Обробка запиту на одержання довідки».

13. Додати перехід «скасування транзакції», спрямований від стану «Очікування вибору клієнта» до стану «Повернення картки».

14. Додати перехід зі сторожовою умовою «[кредит не перевищений]», спрямований від стану «Обробка запиту на зняття готівки» до стану «Видача готівки».

15. Додати перехід зі сторожовою умовою «[кредит перевищений]» з дією на переході повідомлення, спрямований від стану «Обробка запиту на зняття готівки» до стану «Повернення картки».

16. Додати перехід «готівка видана» зі сторожовою умовою «[обрано друк чека]», спрямований від стану «Видача готівки» до стану «Друк».

17. Додати перехід «готівка видана» зі сторожовою умовою «[друк чека не обраний]», спрямований від стану «Видача готівки» до стану «Повернення картки».

18. Додати перехід «довідка сформована», спрямований від стану «Обробка запиту на одержання довідки» до стану «Друк».

19. Додати перехід «друк завершений», спрямований від стану «Друк» до стану «Повернення картки».

20. Додати перехід «картка повернута», спрямоване від стану «Повернення картки» до стану «Завершення транзакції».

21. Додати перехід «транзакція завершена», спрямований від стану «Завершення транзакції» до стану «Очікування картки».

22. Додати перехід, спрямований від стану «Очікування картки» до фінального «стану».

### **Оформлення результатів роботи**

Результати лабораторної роботи оформлюються у вигляді звіту, обов'язковими елементами, якого: титульний аркуш; формулювання мети та завдання на роботу; основна частина; висновки.

Основна частина роботи повинна містити етапи вирішення індивідуального завдання шляхом аналізу досягнутих результатів в обраній предметній області.

Висновки повинні містити результати виконаної роботи.

### **Індивідуальні завдання**

Розробити діаграму станів для систем, список яких надано в лабораторній роботі 3.1 або узгодити вибір процесу з викладачем.

### **Запитання для самоперевірки**

1. Пояснити доцільність створення діаграми станів.
2. Назвати основні елементи діаграми станів.
3. Пояснити взаємовідношення «стан-дія».
4. Назвати відомі псевдостани.
5. Що таке сторожова умова?
6. Як виникають нетригерні переходи?
7. Пояснити відмінності між тригерними та нетригерними переходами.

### Лабораторна робота 3.5.

## РОЗРОБКА ДІАГРАМИ ДІЯЛЬНОСТІ В СЕРЕДОВИЩІ IBM RATIONAL ROSE

**Мета:** вивчити призначення та порядок розробки діаграми діяльності в середовищі IBM Rational Rose.

**Завдання:** розробити діаграму діяльності для обраної системи, оформити звіт із виконаної роботи, представити його для захисту.

### Основні теоретичні відомості

*Діаграма* діяльності представляється у формі графа, вершинами якого є стани дії або діяльності, а дугами – переходи від одного стану дії до іншого.

Основним напрямком використання діаграм діяльності є візуалізація особливостей реалізації операцій класів, коли необхідно представити алгоритми їх виконання.

Стан діяльності служить для подання процедурної послідовності дій, що вимагають певного часу.

Графічно стани діяльності і дії зображуються однаковою фігурою, що нагадує прямокутник з опуклими дугами по боковим сторонам. Усередині цієї фігури записується ім'я діяльності або дії у формі виразу. Воно повинне бути унікальним у межах однієї діаграми діяльності.

При побудові діаграми діяльності використовуються тільки нетриггерні переходи. На діаграмі такий перехід зображується суцільною лінією зі стрілкою.

Графічно *розгалуження* на діаграмі діяльності позначається символом *розв'язку*, зображуваного у формі маленького ромба, без тексту усередині. У ромб може входити тільки одна стрілка від того стану дії, після виконання якого потік керування повинен бути продовжений по одній з гілок, що йдуть далі. Прийнято вхідну стрілку приєднувати до верхньої або лівої вершини символу розв'язку. Вихідних стрілок може бути дві або більше, але кожній з них вказується відповідна сторожова умова у формі булевого виразу.

*Доріжка* – графічна область діаграми діяльності, що містить елементи моделі, відповідальність за виконання яких належить окремим підсистемам.

### Порядок виконання роботи

Ознайомитися з літературою [1, 2, 14-16], побудувати діаграму діяльності станів за прикладом розглянутої моделі банкомата, що показана на рис. 3.5. Для цього виконати послідовність дій 1-22.



Рис. 3.5. Діаграма діяльності для моделі банкомата

1. Активізувати діаграму діяльності для обраного елемента моделі або моделюємої системи в цілому одним з наступних способів:

- на панелі інструментів «стандартна» клацнути по кнопці із зображенням діаграми станів та вибрати подання і тип розроблювальної діаграми;
- у браузері проекту виділити логічне подання або подання варіантів використання і виконати операцію контекстного меню: New → Activity Diagram;

- у головному меню Browse → State Machine Diagram вибрати подання і тип розроблюваної діаграми.

2. Вивчити призначення окремих кнопок панелі інструментів.

3. Задати ім'я «Діаграма діяльності АТМ» новій діаграмі діяльності, а в секцію її документації ввести текст «Діаграма діяльності описує послідовність дій клієнта при використанні банкомата».

4. Додати нову діяльність на діаграму та ввести її ім'я «Вставити картку».

5. Додати діяльності з іменами «Ввести ПІН-код», «Вибрати тип транзакції», «Ввести суму», «Одержати довідку про стан рахунку», «Одержати готівку», «Одержати чек», «Одержати картку» і «фінальний стан».

6. Додати символи розгалуження (рішення), розташувавши їх між діяльностями з іменами «Ввести ПІН-код» і «Вибрати тип транзакції», «Вибрати тип транзакції» і «Ввести суму», «Ввести суму» і «Одержати довідку про стан рахунку», «Одержати готівку» і «Одержати чек», і «Одержати картку». При цьому останній символ рішення буде використовуватися в якості символу з'єднання.

7. Додати перехід, спрямований від діяльності «Ввести ПІН-код» до символу «рішення».

8. Додати перехід зі сторожовою умовою «[ПІН-код вірний]», спрямований від символу рішення до діяльності «Вибрати тип транзакції». Для завдання сторожової умови даного переходу слід ввести текст «ПІН-код вірний» у поле введення «Guard Condition» на вкладці «Detail» вікна специфікації властивостей даного переходу. При цьому текст сторожової умови вводиться без дужок.

9. Додати перехід зі сторожовою умовою «[ПІН-код невірний]», спрямований від символу рішення до символу з'єднання.

10. Додати перехід, спрямований від діяльності «Вибрати тип транзакції» до символу рішення.

11. Додати перехід зі сторожовою умовою «[вибір зняття суми]», спрямований від символу рішення до діяльності «Ввести суму».

12. Додати перехід зі сторожовою умовою «[вибір способу одержання довідки]», спрямований від символу рішення до діяльності «Одержати довідку про стан рахунку».

13. Додати перехід, спрямований від діяльності «Ввести суму» до символу рішення.

14. Додати перехід зі сторожовою умовою «[сума не перевищує кредит]», спрямований від символу рішення до діяльності «Одержати готівку».

15. Додати перехід зі сторожовою умовою «[сума перевищує кредит]», спрямований від символу рішення до символу з'єднання.

16. Додати перехід, спрямований від діяльності «Одержати готівку» до символу рішення.

17. Додати перехід зі сторожовою умовою «[обраний друк чека]», спрямований від символу рішення до діяльності «Одержати чек».

18. Додати перехід зі сторожовою умовою «[друк чека не обраний]», спрямований від символу рішення до символу з'єднання.

19. Додати перехід, спрямований від діяльності «Одержати чек» до символу з'єднання.

20. Додати перехід, спрямований від діяльності «Одержати довідку про стан рахунку» до символу з'єднання.

21. Додати перехід, спрямований від символу з'єднання до діяльності «Одержати картку».

22. Додати перехід, спрямований від діяльності «Одержати картку» до фінального стану.

### **Оформлення результатів роботи**

Результати лабораторної роботи оформлюються у вигляді звіту, обов'язковими елементами, якого: титульний аркуш; формулювання мети та завдання роботи; основна частина; висновки.

Основна частина роботи повинна містити етапи вирішення індивідуального завдання шляхом аналізу досягнутих результатів обраної предметної області.

Висновки повинні містити результати виконаної роботи.

### **Індивідуальні завдання**

Розробити діаграму станів для систем, список яких надано в лабораторній роботі 3.1 або узгодити вибір бізнес-процесів з викладачем.



### **Запитання для самоперевірки**

1. Назвати основне призначення діаграм діяльності.
2. Як позначаються діяльності на діаграмі?
3. Назвати основні елементи діаграми діяльності.
4. Які переходи використовуються на діаграмі діяльності?
5. Що подається ромбом на діаграмі діяльності?
6. Пояснити призначення сторожової умови на діаграмі діяльності.
7. Як зветься графічна область діаграми діяльності, що містить елементи моделі?

## **Модуль IV. Реалізація об'єктно-орієнтованих програмних систем**

### **Лабораторна робота 4.1 ДІАГРАМА КОМПОНЕНТІВ В СЕРЕДОВИЩІ IBM RATIONAL ROSE**

**Мета:** вивчити призначення та порядок побудови діаграми компонентів в середовищі IBM Rational Rose.

**Завдання:** розробити діаграму компонентів для обраної інформаційної системи, оформити звіт та представити його для захисту.

### **Основні теоретичні відомості**

*Діаграма компонентів* описує особливості фізичного подання системи, що дозволяє визначити архітектуру розроблювальної системи, установивши залежності між програмними компонентами, у ролі яких може виступати вихідний, бінарний код. Пунктирні стрілки, що з'єднують модулі, показують відносини взаємозалежності, аналогічні тим, які мають місце при компіляції вихідних текстів програм. Основними графічними елементами діаграми компонентів є *компоненти*, *інтерфейси* і залежності між ними.

Для графічного подання компонента використовується спеціальний символ – прямокутник із вставленими ліворуч двома більш дрібними прямокутниками. Усередині прямокутника

записується ім'я компонента і, можливо, додаткова інформація. Цей символ є базовим позначенням *компонента* в мові UML.

У мові UML для компонентів визначені наступні стереотипи:

- «file» – визначає найбільш загальний різновид *компонента*, який представляється у вигляді довільного фізичного файлу;
- «executable» – визначає різновид компонента-файлу, який є здійсненним файлом і може виконуватися на комп'ютерній платформі;
- «document» – визначає різновид компонента-файлу, який представляється у формі документа довільного змісту, що не є здійсненним файлом або файлом з вихідним текстом програми;
- «library» – визначає різновид компонента-файлу, який представляється у формі динамічної або статичної бібліотеки;
- «source» – визначає різновид компонента-файлу, що представляє собою файл із вихідним текстом програми, який після компіляції може бути перетворений у виконуємий файл;
- «table» – визначає різновид компонента, який представляється у формі таблиці бази даних.

Відношення *залежності* на діаграмі компонентів зображується пунктирною лінією зі стрілкою, спрямованою від клієнта або залежного елемента до джерела або незалежного елемента моделі.

Відношення *реалізації* інтерфейсу позначається на діаграмі компонентів звичайною лінією без стрілки.

### **Порядок виконання роботи**

Ознайомитися з літературою [1, 2, 14-16], побудувати діаграму компонентів за прикладом розглянутої моделі банкомата, що показана на рис. 4.1. Для цього виконати послідовність дій 1-12.

1. Активізувати діаграму *компонентів* одним з наступних способів:

- на панелі інструментів «стандартна» клацнути на кнопку із зображенням діаграми компонентів;
- розкрити подання компонентів Component View у браузері і двічі клацнути на піктограмі Main;
- через пункт меню Browse → Component Diagram.

2. Вивчити призначення панелі інструментів необхідних для розробки діаграми компонентів.

3. Додати компонент на діаграму. Змінити ім'я діаграми, запропоноване за замовчуванням «Main», на «Діаграма компонентів АТМ», а першому доданому компоненту задати ім'я «Mainatm.exe».

Компоненту «Mainatm.exe» вибрати стереотип «EXE» із вкладеного списку, оскільки стосовно до розроблюваної моделі передбачається реалізація цього компонента у формі виконуємого файлу.

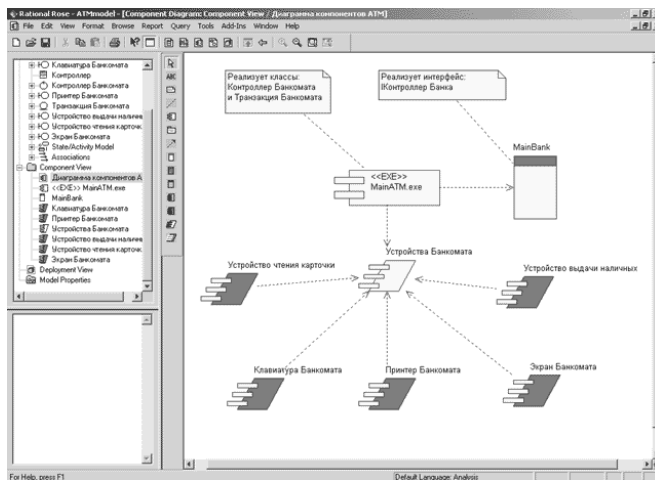


Рис. 4.1. Діаграма компонентів моделі банкомату

4. Додати компонент з ім'ям «Mainbank», для якого вибрати стереотип «Main Program» і додати залежність між двома компонентами. За результатом цих дій на діаграмі з'явиться зображення відносини залежності у формі пунктирної лінії зі стрілкою, що з'єднає два обрані компоненти.

5. Додати компонент з ім'ям «Пристрої Банкомата», для якого задати стереотип «Task Specification».

6. Додати компоненти з іменами «Пристрій читання картки», «Клавіатура Банкомата», «Принтер Банкомата», «Екран Банкомата», «Пристрій видачі готівки», яким задати стереотип «Task Body».

7. Додати залежність від компонента «Mainatm.exe» до компонента «Пристрої Банкомата».

8. Додати залежність від компонента «Пристрій читання картки» до компонента «Пристрою Банкомата».

9. Додати залежність від компонента «Клавіатура Банкомата» до компонента «Пристрої Банкомата».

10. Додати залежність від компонента з «Принтер Банкомата» до компонента «Пристрої Банкомата».

11. Додати залежність від компонента «Екран Банкомата» до компонента «Пристрої Банкомата».

12. Додати залежність від компонента «Пристрій видачі готівки» до компонента «Пристрої Банкомата».

Проконтролювати відповідність виду діаграми компонентів можна по рис. 4.1.

### **Оформлення результатів роботи**

Результати лабораторної роботи оформлюються у вигляді звіту, обов'язковими елементами, якого: титульний аркуш; формулювання мети та завдання на роботу; основна частина; висновки.

Основна частина роботи повинна містити етапи вирішення індивідуального завдання шляхом аналізу досягнутих результатів обраної предметної області.

Висновки повинні містити результати виконаної роботи.

### **Індивідуальні завдання**

Розробити діаграму станів для систем, список яких надано в лабораторній роботі 3.1 або узгодити вибір теми з викладачем.

### **Запитання для самоперевірки**

1. До якого класу діаграм відноситься діаграма компонентів?
2. Назвати основні елементи діаграми компонентів.
3. Що таке компонент та яких видів він буває?
4. Як графічно відображається компонент на діаграмі компонентів?
5. Які стереотипи визначені в UML для компонентів?
6. Які властивості компонента відображаються на відповідній діаграмі?
7. Які відношення існують в діаграмі компонентів та як вони відображаються?

## **Лабораторна робота 4.2**

### **ДІАГРАМА РОЗГОРТАННЯ В СЕРЕДОВИЩІ IBM RATIONAL ROSE**

**Мета:** вивчити призначення і порядок розробки діаграми розгортання в середовищі IBM Rational Rose.

**Завдання:** розробити діаграму розгортання для обраної інформаційної системи, підготувати звіт з виконаної роботи та представити його для захисту.

#### **Основні теоретичні відомості**

На *діаграмі розгортання* представлені вузли виконання програмних компонентів реального часу, процесів і об'єктів.

Діаграма розгортання застосовується для представлення загальної конфігурації і топології розподіленої програмної системи і містить зображення розміщення компонентів по окремих вузлах системи. Крім того, діаграма розгортання показує наявність фізичних сполук – маршрутів передачі інформації між апаратними пристроями, задіяними в реалізації системи.

Діаграма розгортання містить графічні зображення процесорів, пристроїв, процесів і зв'язків між ними. На відміну від діаграм логічного подання, діаграма розгортання є єдиною для системи в цілому, оскільки повинна відображати всі особливості її реалізації.

Вузол являє собою фізично існуючий елемент системи, який володіє обчислювальним ресурсом або є технічним пристроєм. Графічно вузол на діаграмі розгортання зображується у формі тривимірного куба. Вузол має ім'я, яке вказується всередині цього графічного символу. Самі вузли можуть представлятися як на рівні типу, так і на рівні екземпляра.

Зображення вузлів можуть розширюватися, щоб включити додаткову інформацію про специфікації вузла. Якщо додаткова інформація відноситься до імені вузла, то вона записується під цим ім'ям у формі поміченого значення

Як доповнення до імені вузла можуть використовуватися різні текстові стереотипи, які явно специфікують призначення цього вузла. Для цього використовуються наступні текстові стереотипи: "processor", "sensor", "modem", "net", "printer" та інші, зміст яких зрозумілий з контексту.

В якості відносин виступають фізичні з'єднання між вузлами, а також залежності між вузлами і компонентами, які допускається зображати на діаграмах розгортання.

Сполучення є різновидом асоціації і зображуються відрізками ліній без стрілок.

### Порядок виконання роботи

Ознайомитися з літературою [1, 2, 14-16], побудувати діаграму розгортання за прикладом розглянутої моделі банкомата, що показана на рис. 4.2. Для цього виконати послідовність дій 1-8.

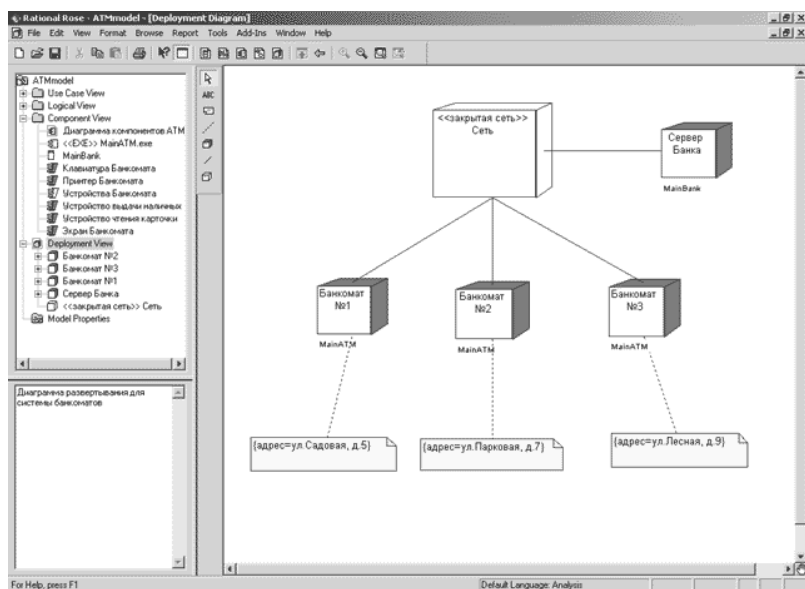


Рис. 4.2. Діаграма розгортання розробляємої моделі управління банкоматом

1. Активізація діаграми розгортання виконується одним із таких способів:

- на панелі інструментів «стандартна» вибрати мишею зображення діаграми розгортання;
- в браузері проекту подвійним клацанням по піктограмі уявлення розгортання (Deployment View);

- виконати операцію головного меню: Browse → Deployment Diagram.

2. По спеціальній панелі інструментів вивчити кнопки із зображенням графічних примітивів, що необхідні для розробки діаграми розгортання.

3. На діаграму розгортання додати вузол типу процесор і задаємо йому ім'я «Банкомат №1», для якого у формі примітки вкажемо позначене значення: «{адреса = вул. Садова, буд.5}».

4. Додати з'єднання вузлів з іменами «Банкомат №1» і «Мережа».

5. Додати процесор з ім'ям «Банкомат №2», якому задати значення у формі примітки: «{адреса = вул. Паркова, б.7}», а на вкладці властивостей «Detail» визначити новий процес і вибрати для нього ім'я «MainATM» з вкладеного списку.

6. Додати процесор з ім'ям «Банкомат №3», якому задати примітку: «{адреса = вул. Лісова, д.9}», а на вкладці властивостей «Detail» визначити новий процес і вибрати для нього ім'я «MainATM» з вкладеного списку.

7. Додати процесор з ім'ям «Сервер Банку», якому на вкладці властивостей «Detail» визначити новий процес з ім'ям «MainBank».

8. Додати з'єднання між вузлом «Мережа» і вузлами з іменами «Банкомат №2», «Банкомат №3» і «Сервер Банку».

Построенная таким образом *диаграмма развертывания* будет иметь следующий вид рис. .

### **Оформлення результатів роботи**

Результати лабораторної роботи оформлюються у вигляді звіту обов'язковими елементами, якого є: титульний аркуш; формулювання мети та завдання на роботу; основна частина; висновки. Зразок титульного листа наведений у Додатку 1.

Основна частина роботи повинна містити етапи вирішення індивідуального завдання шляхом аналізу досягнутих результатів в обраній предметній області.

Висновки повинні містити результати виконаної роботи.

### **Індивідуальні завдання**

Розробити діаграму станів для систем, список яких надано в лабораторній роботі 3.1 або узгодити вибір бізнес-процесів з викладачем.

### **Запитання для самоперевірки**

1. Чи можна діаграму розгортання виключити з процесу розробки інформаційної системи?
2. Що входить до основних елементів діаграми розгортання?
3. Які відносини застосовують в діаграмах розгортання?
4. Як на діаграмах розгортання позначають вузол?
5. Які властивості може мати вузол?
6. Які стереотипи використовують в діаграмах розгортання?
7. Чи припустиме використання фізичних з'єднань на діаграмі розгортання?

## **Лабораторна робота 4.3**

### **ПІДГОТОВКА МОДЕЛІ ДЛЯ ГЕНЕРАЦІЇ ПРОГРАМНОГО КОДУ В СЕРЕДОВИЩІ IBM RATIONAL ROSE**

**Мета:** вивчити призначення і порядок підготовки моделі до генерації програмного коду в середовищі IBM Rational Rose

**Завдання:** обрану інформаційну модель підготувати для генерації програмного коду, оформити звіт з виконаної роботи та представити його для захисту.

### **Основні теоретичні відомості**

Однією з найбільш важливих властивостей програми IBM Rational Rose є можливість генерації програмного коду на декількох мовах програмування, яка може бути використана розробником після побудови моделі. Для цієї мети в середовищі IBM Rational Rose присутній досить великий вибір мов програмування і схем баз даних.

Загальна послідовність дій, які необхідно виконати для генерації програмного коду в середовищі IBM Rational Rose, складається з наступних етапів:

1. Перевірити модель на відсутність помилок.
2. Створити компоненти для реалізації класів.



3. Відобразити класи на компоненти.
4. Вибрати мову програмування для генерації тексту програмного коду.
5. Встановити властивості генерації програмного коду.
6. Вибрати класи, компоненти або пакети.
7. Генерувати програмний код.

Особливості виконання кожного з етапів можуть змінюватися залежно від вибору мови програмування або схеми бази даних.

В якості мови реалізації проекту рекомендується вибирати мову програмування ANSI C ++, яка не вимагає інсталяції додаткових програм і поставляється практично у всіх конфігураціях IBM Rational Rose.

### Порядок виконання роботи

Ознайомитися з літературою [1, 2, 14-16].

1. Поміняти імена класів, атрибутів і операцій, при цьому діаграма класів моделі банкомату буде мати вид (рис. 4.3).

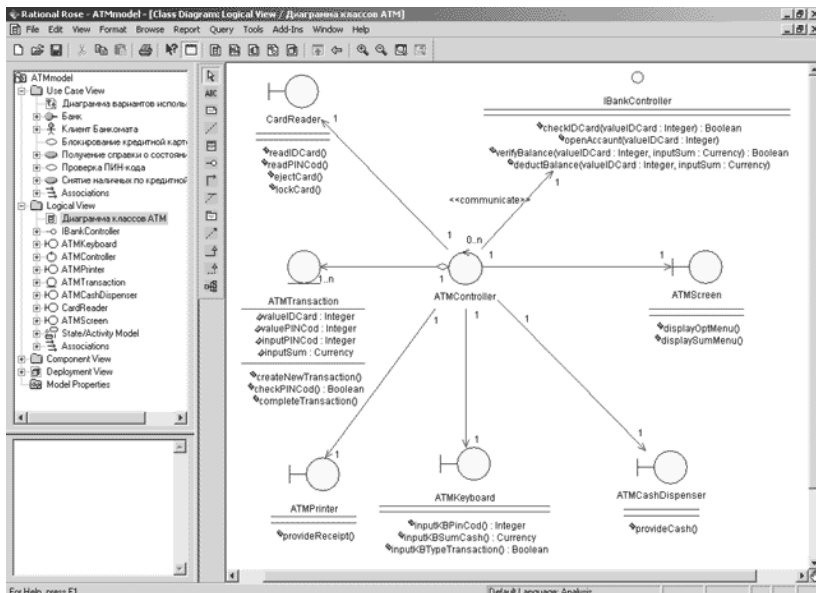


Рис. 4.3. Діаграма класів моделі банкомату

2. Перевірити модель шляхом виконання операції головного меню: Tools Check Model. Проконтролювати результати перевірки моделі на наявність помилок у вікні журналу. Усунути помилки і попередження, про що має свідчити чисте вікно журналу (рис. 4.4).



Рис. 4.4. Вид журналу за відсутності помилок за результатами перевірки

3. Перейменувати компоненти, задавши їм англійські імена. Відповідна діаграма компонентів моделі банкомата матиме вигляд, представлений на рис. 4.5.

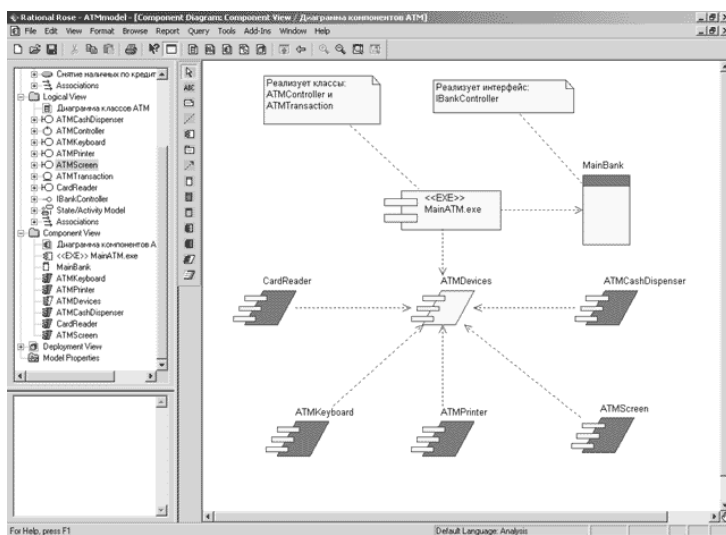


Рис. 4.5. Діаграма компонентів моделі банкомата після перетворення імен компонентів

4. Компоненту «MainATM.exe» для генерації програмного коду вибрати класи «ATMTransaction» і «ATMController» (рис. 4.6).

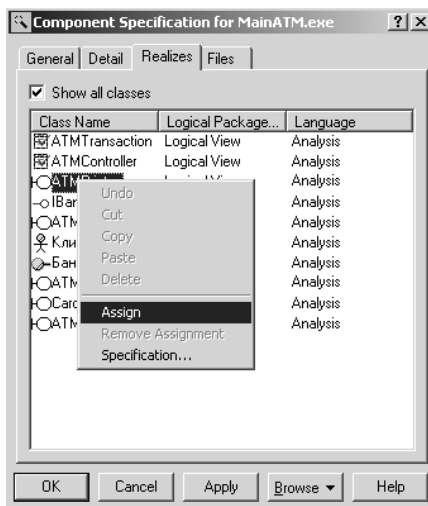


Рис. 4.6. Діалогове вікно налаштування властивостей реалізації класів у компоненті «MainATM.exe»

5. Для вибору мови ANSI C ++ в якості мови реалізації моделі виконати операцію головного меню «Tools Options», в результаті чого викликається діалогове вікно налаштування параметрів моделі. На вкладці «Notation» у рядку «Default Language» з вкладеного списку слід вибрати мову ANSI C ++.

6. Змінити мову в рядку «Language» на вкладці «General» вікна специфікації властивостей компонента, для чого з вкладеного списку слід вибрати мову ANSI C ++ (рис. 4.7).

Після вибору мови програмування привести у відповідність типи атрибутів, типи аргументів і значень операцій. З цією метою потрібно переглянути всі класи діаграми класів і змінити ті типи даних, які не є синтаксично допустимими в обраній мові програмування. Для мови ANSI C ++ слід замінити тип Integer на int, Boolean на bool, Currency на float. Інакше відповідні виправлення доведеться виконувати вручну після генерації програмного коду.

7. З моделі системи управління банкоматом для генерації програмного коду мовою ANSI C ++ виберемо компонент з ім'ям «MainATM.exe». Для генерації коду потрібного елемента виділити в браузері проекту і виконати операцію контекстного меню: ANSI C ++ Generate Code\_. В результаті цього буде відкрито діалогове

вікно, що запрошує вибрати класи для генерації програмного коду обраною мовою програмування (рис. 4.8).



Рис. 4.7. Вікно специфікації властивостей компонента «MainATM.exe» при виборі мови його реалізації

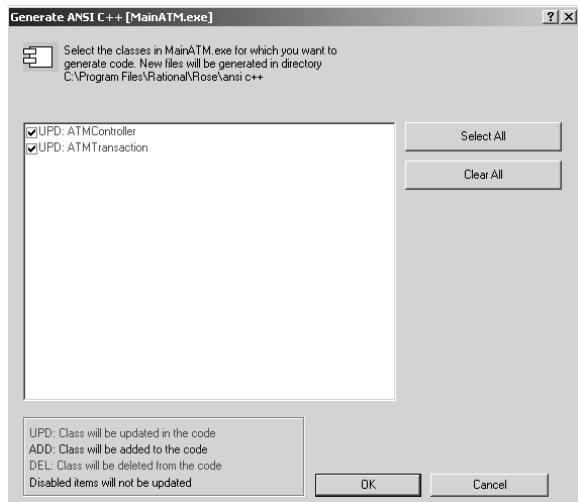
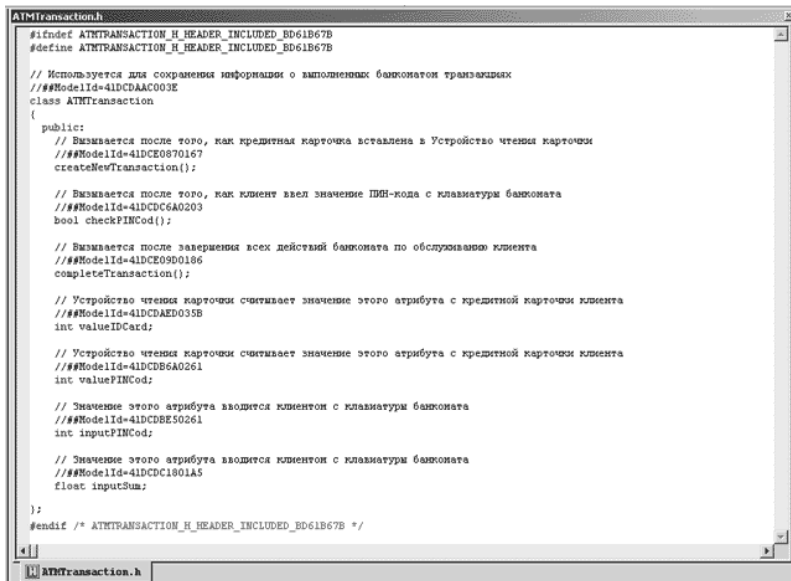


Рис. 4.8. Вікно вибору класів для генерації програмного коду

8. Для перегляду і редагування створених файлів з текстом програмного коду мовою ANSI C ++ призначений вбудований текстовий редактор, який можна відкрити за допомогою операції контекстного меню ANSI C ++ → Browse Header\_ або ANSI C ++ → Browse Body\_ для обраного класу в браузері проекту.

Після генерації програмного коду для компонента «MainATM.exe» кожному класу, реалізованому в даному компоненті, відповідатиме 2 файли з текстом коду на мові ANSI C ++. Так, наприклад, для класу «ATMTransaction» буде згенеровано заголовний файл з розширенням «h» (рис. 4.9) і файл реалізації з розширенням «cpp» (рис. 4.10).



```
ATMTransaction.h
#ifndef ATMTRANSACTION_H_HEADER_INCLUDED_BD61B67B
#define ATMTRANSACTION_H_HEADER_INCLUDED_BD61B67B

// Используется для сохранения информации о выполнении банкоматом транзакций
///##ModelId=41DCDAAC003E
class ATMTransaction
{
public:
    // Вызывается после того, как кредитная карточка вставлена в Устройство чтения карточек
    ///##ModelId=41DCE0870167
    createNewTransaction();

    // Вызывается после того, как клиент ввел значение ПИН-кода с клавиатуры банкомата
    ///##ModelId=41DCDC6A0203
    bool checkPINCode();

    // Вызывается после завершения всех действий банкомата по обслуживанию клиента
    ///##ModelId=41DCE09D0186
    completeTransaction();

    // Устройство чтения карточек считывает значение этого атрибута с кредитной карточки клиента
    ///##ModelId=41DCDAED035B
    int valueIDCard;

    // Устройство чтения карточек считывает значение этого атрибута с кредитной карточки клиента
    ///##ModelId=41DCDB6A0261
    int valuePINCode;

    // Значение этого атрибута вводится клиентом с клавиатуры банкомата
    ///##ModelId=41DCDBE50261
    int inputPINCode;

    // Значение этого атрибута вводится клиентом с клавиатуры банкомата
    ///##ModelId=41DCDC1801A5
    float inputSum;
};

#endif /* ATMTRANSACTION_H_HEADER_INCLUDED_BD61B67B */
```

Рис. 4.9. Вид вбудованого текстового редактора з завантаженням в нього заголовного файлу ATMTransaction.h

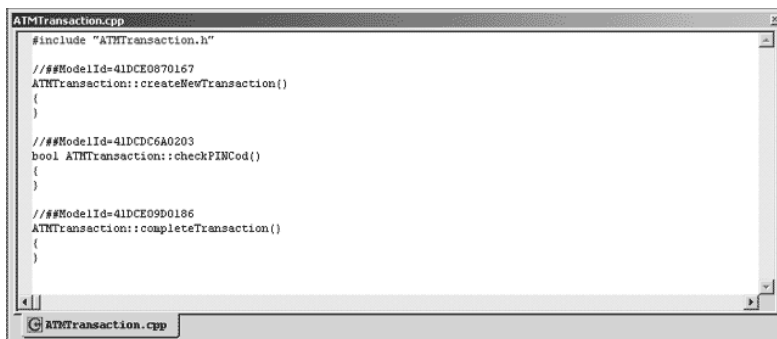


Рис. 4.10. Вид вбудованого текстового редактора з завантаженням в нього заголовними файлами «ATMTransaction.cpp»

8. Далі робота виконується в обраному інтегрованому середовищі програмування, наприклад, MS Visual C ++ або Borland C ++.

### **Оформлення результатів роботи**

Результати лабораторної роботи оформлюються у вигляді звіту обов'язковими елементами, якого є: титульний аркуш; формулювання мети та завдання на роботу; основна частина; висновки. Зразок титульного листа наведений у Додатку 1.

Основна частина роботи повинна містити етапи вирішення індивідуального завдання шляхом аналізу досягнутих результатів в обраній предметній області.

Висновки повинні містити результати виконаної роботи.

### **Індивідуальні завдання**

Провести генерацію коду для систем, список яких надано в лабораторній роботі 3.1 або узгодити вибір бізнес-процесів з викладачем.

### **Запитання для самоперевірки**

1. Який порядок генерації коду в середовищі *IBM Rational Rose*?
2. Які мови програмування підтримує середовище *IBM Rational Rose*?
3. Які діаграми не є обов'язковими для генерації програмного коду?

4. Який вигляд приймає програмний код після проведення операції генерації коду?

5. Назвати послідовність операцій для вибору мови програмування.

6. Показати порядок вибору компонентів для генерації коду.

7. Назвати порядок дій для перевірки моделі на наявність помилок.

### **Лабораторна робота 4.4. ОЦІНКА РОЗМІРУ ТА ВАРТОСТІ ПРОЕКТУ**

**Мета:** набуття навиків у прогнозуванні числових характеристик проектів ПЗ з використанням конструктивної моделі вартості (COnstructive COst MOdel).

**Завдання:** оцінити витрати на програмування за моделлю COSOMO.

#### **Основні теоретичні відомості**

COSOMO – це множина моделей, яка дозволяє обчислити вартість проекту ПЗ. В якості одиниці виміру вибрати кількість рядків коду (LOC). COSOMO дає усереднене значення оцінок. COSOMO включає наступні моделі:

- базову COSOMO (застосовується у фазі специфікування вимог);
- проміжну COSOMO (застосовується у фазах розробки множин вхідних умов проекту, наприклад - досвід персоналу, апаратні обмеження, обмеження у інструментах розробки);
- удосконалену COSOMO (застосовується після розробки ПЗ).

Основними виразами базового COSOMO є:

$$E = ab \times (KLOC)^{bb}, \quad (4.1)$$

$$D = cb^{db}, \quad (4.2)$$

де  $E$  – витрати на створення проекту, людино-місяці;  $KLOC$  - кількість тисяч рядків коду;  $ab$ ,  $bb$ ,  $cb$  та  $db$  коефіцієнти, які дані у табл. 4.2,  $D$  - час розробки у календарних місяцях.

Таблиця 4.2

## Типи проектів

Тип проекту	ab	bb	cb	db
Organic	2.4	1.05	2.5	0.38
Semi-detached	3.0	1.12	2.5	0.35
Embedded	3.6	1.20	2.5	0.32

Тип „Organic” представляє відносно невеликий та простий проект, який виношується невеликою командою з добрим досвідом.

Тип „Semi-detached” передбачає середній за розміром та складністю проект, в якому команда має змішаний рівень досвіду і відносно жорсткі вимоги.

Тип „Embedded” представляє проект, який виконується в умовах жорстких технічних, програмних та експлуатаційних обмежень.

## Порядок виконання роботи

Ознайомитися з літературою [1, 2, 10, 11]. За формулами (4.1), (4.2) за допомогою Excel провести розрахунки людино-місяців проекту і кількість у календарних місяцях, що будуть витрачені на розробку проекту інформаційної системи для різних типів проекту. В якості значення *KLOC* вибрати значення:

1. 1000;
2. 2000;
3. 2500;
4. 3000;
5. 3500;
6. 4000;
7. 4500;
8. 5000;
9. 5000;
10. 5500;
11. 6000.

Побудувати графік залежності  $E(KLOC)$  зробити аналіз.

## Оформлення результатів роботи



Результати лабораторної роботи оформлюються у вигляді звіту обов'язковими елементами, якого є: титульний аркуш; формулювання мети та завдання на роботу; основна частина; висновки. Зразок титульного листа наведений у Додатку 1.

Основна частина роботи повинна містити етапи вирішення індивідуального завдання, вирази для розрахунку формул (4.1), (4.2), графік залежності  $E(KLOC)$ .

Висновки повинні містити результати виконаної роботи.

### **Індивідуальні завдання**

Зробити оцінку витрат на розробку програмного забезпечення для виконуємої розробки інформаційної системи за списком лабораторної роботи 3.1.

### **Запитання для самоперевірки**

1. В яких одиницях відбувається вимірювання трудомісткості програмної продукції?
2. До якого класу відносяться проекти з невеликою кількістю розробників з добрим досвідом та не жорсткими вимогами?
3. Назвати базові рівняння моделі СОСОМО.
4. Які обмеження має модель СОСОМО?
5. Що включає в себе модель базового рівня?
6. Які критерії застосовуються для використання СОСОМО?
7. Що включає в себе модель середнього рівня?

## СПИСОК ДЖЕРЕЛ

1. Сидоров М.О. Вступ до інженерії програмного забезпечення / М.О.Сидоров. – К.: Вид-во НАУ «НАУ-друк», 2010. – 112 с.
2. Sommerwyl I. Ingenieria programnoho obespechenia, 6-e izdanie: Per. s angl. - M.: Vilyams, 2002. - 624 s.
3. Бабенко Л.П. Основи програмної інженерії / Бабенко Л.П., Лавріщева К.М. – К.: Т-во «Знання», КОО, 2001. – 269 с.
4. Буч Г. Унифицированный процес разработки программного обеспечения / Буч Г., Якобсон А., Рамбо Дж.. – СПб.: Питер, 2002. - 496 с.
5. Кобер А. Современные методы описания функциональных требований к системам, М.: ИД "Лори", 2002. – 670с.
6. Константайн Л., Локвуд Л. Разработка программного обеспечения. - СПб.: Питер, 2004. - 592 с.
7. Иванова Г.С. Технология программирования: Учебник для вузов. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2002. - 320 с.
8. ДЕРЖСТАНДАРТ 34.602-89 Технічне завдання на створення автоматизованої системи.
9. ДЕРЖСТАНДАРТ 19.201-78 Технічне завдання. Вимоги до змісту і оформлення.
10. Липаев В.В. Проектирование программных средств / В.В. Липаев. - М.:Высш. шк., 1990.- 452с.
11. Опалева Э.А. Технология разработки программного обеспечения / Опалева Э.А., Самойленко В.П. - Л.: ЛЭТИ, 1988.- 358с.
12. Брауде Э. Дж. Технологии разработки программного обеспечения - СПб.: Питер, 2004.-389с.
13. Буч Г. Объектно-ориентированный анализ и проектирование с примерами приложений / Буч Г. та інш. - М.: ООО "И.Д. Вильямс", 2008. - 720 с.
14. Орлов С. А. Технологии разработки программного обеспечения / Орлов С. А. – СПб.: Питер, 2002. — 464 с.
15. Иванова Г.С. Технология программирования / Иванова Г.С. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2002. – 320 с.
16. Трофимов С.А. Case-технологии. Практическая работа в Rational Rose / Трофимов С.А. – М.: Бином-Пресс, 2002. – 288 с.

## Опис інформаційної системи

### Коротка інформація про компанію "МЕД"

Компанія - дистриб'ютор "МЕД" закуповує медичні препарати вітчизняних та зарубіжних виробників і реалізує їх через власну дистриб'юторську мережу і мережу аптек. Компанія здійснює доставку товарів як власним транспортом, так і за допомогою послуг сторонніх організацій.

Основні *бізнес-процеси* компанії – закупівля, складування запасів, продаж, взаєморозрахунки з постачальниками і клієнтами.

Конкуренція компанії на ринку останнім часом збільшилась, завдяки виходу на ринок двох нових конкурентів, до яких перейшла частина клієнтів і ряд найбільш кваліфікованих співробітників ЗАТ "МЕД". ЗАТ "МЕД" має дві філії – у Києві і Харкові. Кожна філія функціонує як самостійна юридична особа, будучи дочірними компаніями ЗАТ "МЕД".

За попередніми планами, компанія має намір відкрити дочірнє підприємство для організації виробництва в безпосередній близькості до своїх замовників.

### Адреси та телефони

Київ, К-123, вул. Київська, д. 20, стр. 7, офіс 709.

Телефон: (044) 345-6789, факс: (095) 345-9876.

### Контактні особи

Борис Петренко - Генеральний директор.

Дмитро Кузьменко - Виконавчий директор

Артур Маркитенко - Директор по маркетингу.

### Співробітники

На момент проведення аналізу штат компанії складає 110 співробітників.

Основними цілями проекту автоматизації компанії "МЕД" є:

- розробка та впровадження комплексної автоматизованої системи підтримки логістичних процесів компанії;
- підвищення ефективності роботи всіх підрозділів компанії і забезпечення ведення обліку у єдиній інформаційній системі.

### Бачення виконання проекту і границі проекту

В рамках проекту розгортання нової системи передбачається здійснити лише у підрозділах ЗАТ "МЕД":

- відділах закупівель, приймання, продажу, маркетингу;
- групі планування і маркетингу;
- групі логістики;
- обліково-операційному відділу;

- обліковому відділі;
- відділу сертифікації (в частині обліку сертифікатів на медикаменти);
- бухгалтерії (в частині обліку закупівель, продажів, надходжень і платежів).

В межах проекту не розглядається автоматизація обліку основних засобів, розрахунку та нарахування заробітної плати, управління кадрами та автоматизація процесів взаємовідносин з клієнтами.

Кількість робочих місць користувача - 50.

### **Звіт з**

Список програмного забезпечення, використововуваного компанією на момент обстеження

1. "1С: Підприємство 7.7" ("Бухгалтерія", "Торгівля", "Зарплата", "Кадри", "Каса", "Банк") для роботи бухгалтерії.
2. Дві власні розробки на базі *конфігуратора* "1С" - "Закупівлі" і "Продажі".
3. Власна розробка на базі FOXPRO для фінансового відділу.
4. Excel для планування продажів.

### **Існуючий рівень автоматизації**

Кількість робочих станцій, всього:	90
Кількість працівників відділу ІТ	2
Кількість ПК, одночасно працюючих в мережі	50
Наявність і форма зв'язку з віддаленими об'єктами	Термінальний зв'язок зі складом
Кількість робочих станцій на віддаленому об'єкті	8
Характеристики комп'ютерів	Від Celeron 600 і вище
Операційна система	Windows 7, 8.1, 10
Системи, які представляється можливим залишити без зміни	"1С: Підприємство 7.7" в модульному складі "Бухгалтерія", "Зарплата", "Кадри", для роботи бухгалтерії

### **Бачення обліку інформаційної системи**

Одна з основних вимог компанії "МЕД" до майбутнього рішення полягає в тому, щоб воно було побудовано на фундаменті єдиної

інтегрованої системи, а робота всіх співробітників велася в одному інформаційному просторі.

Ключові функціональні вимоги до програмної системи:

1. Потужні засоби захисту даних від несанкціонованого доступу. Розмежування доступу до даних у відповідності з посадовими обов'язками.
2. Можливість віддаленого доступу.
3. Управління запасами. Оперативне отримання інформації про залишки на складі.
4. Управління закупівлями. Планування закупівель в розрізі постачальників.
5. Управління продажами. Контроль ліміту заборгованості з можливістю блокування формування відвантажувальних документів.
6. Повний контроль взаєморозрахунків з постачальниками і клієнтами.
7. Отримання управлінських звітів у необхідних аналітичних зрізах - як детальних для менеджерів, так і агрегованих для керівників підрозділів та директорів фірми.

Приклади форм звітних документів

Інвентарний код	Назва матеріалу товару	Од. вимірювання	Потрібно закупити	Попередня дата придбання		
				Назва постачальника	Дата останнього придбання	Вартість придбання

### Опис системи обліку

ЗАТ "МЕД" використовує типовий план рахунків, три аналітики (контрагенти, договори, регіони).

Фрагмент плану рахунків компанії

Номер бухг. Рахунку	Найменування рахунку
01.000	Основні засоби
02.000	Амортизація основних засобів
03.000	Прибуткові вкладення в матеріальні цінності
04.000	Нематеріальні активи
05.000	Амортизація нематеріальних активів

08.000	Вкладення у необоротні активи
10.000	Матеріали
10.100	Сировина і матеріали
10.200	Інші матеріали
10.300	Інвентар та господарські речі
14.000	Резерви під зниження вартості МЦ
16.000	<i>Відхилення вартості МЦ</i>
19.000	ПДВ з придбань
...	...

### Фрагмент облікової політики

**Виручка і прибуток.** Виручка від реалізації продукції і надання послуг визначається в міру відвантаження реалізованої продукції, надання послуг і відображається у фінансовій звітності за методом нарахування.

**Запаси.** Компанія з метою визначення фактичної собівартості товарів, реалізованих у звітному періоді, використовує варіант їх оцінки за собівартістю перших за часом придбання матеріалів (ФІФО).

### Опис довідників

Фрагмент опису довідників, використовуваних для автоматизації компанії "МЕД", наведено в таблиці.

№	Найменування довідника	Код	Найменування
	Клієнти		
		AC_Ap_00001	Покупатель_АПТЕКІ
		AC_Ds_00001	Покупатель_Дістribьютери
		OTHER_00001	Інші
	Постачальники / підрядники		
		B_00001	Банки
		L_00001	Приватні особи
		I_0001	Страхові організації
		OTHER_00001	Інші
	Договору		
	1- наші послуги	1_COM_D / М / Е	Договір комісії_Д / М / Г

		1_SERV_D_ / М / Е	Договір на надання наших услуг_Д / М / Г
	2 - послуги нам	2_COM_D / М / Е	Договір комісії_Д / М / Г по послугах нам
		2_SERV_D / М / Е	Договір на вказівку Г послуг нам _Д / М
		2_COM_D / М / Е	Договір комісії_Д / М / Г по послугах нам

Код довідника відображає рівні ієрархії. Довідники клієнтів і договорів мають трирівневу структуру. Довідник постачальників - дворівневу структуру. У коді довідника для відображення рівня застосований символ підкреслення. Наприклад, в коді довідника клієнтів перший рівень позначений символами "АС" – покупець; другий рівень - "Ар" – аптекі, "Ds" – дистриб'ютори; для позначення третього рівня передбачені п'ятизначні порядкові номери 00001, 00002 і т.д.

## Додаток 2

### Приклад оформлення технічного завдання

#### Зміст

1.	Загальні відомості	5
1.1.	Найменування системи	5
1.2.	Номер договору	5
1.3.	Найменування Розробника та Замовника робіт та їх реквізити	5
1.4.	Підстава для проведення робіт	5
1.5.	Терміни початку і закінчення робіт	5
1.6.	Джерела і порядок фінансування робіт	5
1.7.	Порядок оформлення і пред'явлення Замовнику результатів робіт	6
2.	Призначення і цілі створення системи	6
2.1.	Призначення системи	6
2.2.	Мета створення системи	6
3.	Характеристика об'єкта автоматизації	6
3.1.	Робота зі звітами	6
4.	Вимоги до системи	7

4.1.	Вимоги до системи в цілому	7
4.1.1.	Вимоги до структури системи	7
4.1.2.	Вимоги до режимів функціонування системи	7
4.1.3.	Вимоги до способів і засобів зв'язку для інформаційного обміну між компонентами системи	7
4.1.4.	Вимоги до сумісності з суміжними системами	7
4.1.5.	Перспективи розвитку системи	7
4.1.6.	Вимоги до чисельності та кваліфікації персоналу та режиму його роботи	8
4.1.7.	Показники призначення	9
4.1.8.	Вимоги до надійності	9
4.1.9.	Вимоги по ергономіці і технічній естетиці	10
4.1.10.	Вимоги щодо безпеки	10
4.1.11.	Вимоги до експлуатації, технічного обслуговування, ремонту та зберігання	10
4.1.12.	Вимоги щодо збереження інформації	11
4.2.	Вимоги до видів забезпечення	12
4.2.1.	Загальні відомості	12
4.2.2.	Вимоги до лінгвістичного забезпечення	12
4.2.3.	Вимоги до технічному забезпеченню	13
4.2.4.	Вимоги до програмного забезпечення	13
4.2.5.	Вимоги до технічному забезпеченню	14
4.2.6.	Вимоги до організаційного забезпечення	14
5.	Склад і зміст робіт зі створення системи	15
6.	Порядок контролю та приймання системи	17
7.	Вимоги до документування	18
7.1.	Загальні вимоги до документування	18
7.2.	Перелік підлягають розробці документів	18

## Продовження Додатку 2

8.	Вимоги до складу та змісту робіт з підготовки об'єкта автоматизації до введення системи в дію	18
9.	Порядок внесення змін	19



## **1. Загальні відомості**

### **1.1. Найменування системи**

#### **Повне найменування системи:**

*Система "Платежі та взаєморозрахунки з кредиторами ".*

#### **Умовне позначення системи:**

*АІС "Платежі та взаєморозрахунки з кредиторами "*

### **1.2. Номер договору**

Договір №135426 від 14 травня 2015 на поставку, впровадження та супровід прикладного програмного забезпечення для автоматизації обробки безготівкових, готівки, рублевих і валютних платежів через кілька банків, що здійснюються бухгалтерією і фінансовою службою.

### **1.3. Найменування Розробника та Замовника робіт та їх реквізити** **Розробник:**

Закрите акціонерне товариство "*Програмні системи* "

Адреса: 03237, Київ, проспект Вернадського, буд.3

Тел. : (044) 922-33-55, факс: (044) 922-33-44.

Банківські реквізити: ЗАТ "*Програмні системи* ", ПІН 7501004321, р / рах № 40603410800020007021 в АКБ Ощадбанк, МФО 044579857, кор.

Рахунок № 30101820400000000335

#### **Замовник:**

Закрите акціонерне товариство "Оргсинтез"

Адреса: 03000, м. Харків, вул. Московське шосе, буд.12

Тел. : (056) 44-10-18, факс: (056) 44-10-10

Банківські реквізити: ЗАТ "Оргсинтез", ПІН 7501004321, р / рах № 40603410800020004521 в СКБ Банк "Гарантія", БИК 044573421, кор.

рахунок № 30101820400000001234

### **1.4. Підстава для проведення робіт**

Підставою для проведення робіт зі створення системи АІС "Платежі та взаєморозрахунки з *кредиторами* " є такі документи:

Договір № 135426 від 14.05.2015

Наказ №56 від 10.05.2015

Розпорядження №35 від 11.05.2015.

### **1.5. Терміни початку і закінчення робіт**

Дата початку робіт: 01.12.2015

Дата закінчення робіт: 01.05.2016

### **1.6. Джерела і порядок фінансування робіт**

Фінансування робіт здійснюється з коштів ЗАТ "Оргсинтез". Порядок фінансування робіт визначається умовами Договору № 135426 від 14.05.2015 р.

### **1.7. Порядок оформлення і пред'явлення Замовнику результатів робіт**

Роботи зі створення Системи виробляються і приймаються поетапно.

Після закінчення кожного з етапів робіт Розробник представляє Замовникові відповідну документацію і підписаний з боку Розробника Акт здачі-приймання робіт, а по закінченні етапів "Пусконаладжувальні роботи" і "Дослідна експлуатація" додатково повідомляє Замовника про готовність Системи та її частин до випробувань.

## **2. Призначення і цілі створення системи**

### **2.1. Призначення системи**

**АІС "Платежі та взаєморозрахунки з кредиторами"** - Прикладне програмне забезпечення, призначене для:

- підготовці / узгодженні / затвердження документів;
- планування робіт;
- ведення обліку та контролю виконання робіт;
- призначення виконавців за кожним завданням, відстеження процесу виконання завдань та вирішення проблем;
- оперативне планування робіт відділу;
- облік робочого часу на виконання завдань;
- збір статистичної інформації по роботах і виконавцям.

### **2.2. Мета створення системи**

Основними цілями впровадження системи є:

- створення єдиного механізму планування і здійснення робіт по взаєморозрахунках з *кредиторами*;
- створення функціонально повного механізму підготовки, погодження та зберігання різних документів (при інтеграції з сховищем Documentum);
- забезпечення повноти, достовірності і оперативності інформаційної підтримки прийняття рішень для здійснення

готівкових, безготівкових і валютних взаєморозрахунків з постачальниками.

### **3. Характеристика об'єкта**

Об'єктом є набір процесів, зазначених у "Методології моделювання предметної області", які мають *місце* в рамках здійснення взаєморозрахунків з *кредиторами*, а також ряду додаткових учасників, виконують функції інформаційної підтримки, контролю, а також нормативного регулювання об'єкта автоматизації.

#### **3.1. Робота зі звітами**

У додатку ПС "Платежі та взаєморозрахунки з *кредиторами*" передбачена можливість побудови різних звітів. Сформовані звіти виводяться в додаток *MS Excel*. Користувач має можливість вивести звіт на друк або зберегти звіт на диску.

Основні типи звітів:

- План поставок;
- План платежів;
- Зведена таблиця платежів;
- Звіт про залишки коштів на рахунках у банках;
- Звіт з затвердженими заявками про перерахування грошових коштів;
- Зведена таблиця платежів з урахуванням залишків коштів на розрахункових рахунках на 1 день (на тиждень, на місяць);
- Зведена таблиця платежів з урахуванням здійснених платежів;
- Зведена таблиця платежів з урахуванням здійснених платежів і виписок з *розрахункового рахунку*;
- Звіт з виведенням сальдо по взаєморозрахунках з постачальниками.

### **4. Вимоги до системи**

#### **4.1. Вимоги до системи в цілому**

##### **4.1.1. Вимоги до структури системи**

ПС "Платежі та взаєморозрахунки з *кредиторами*" призначена для обміну інформацією між об'єктами та процесу обробки заявок. Автоматизації підлягають операції підготовки, реєстрації, відстеження статусу заявок, розсилки заявок на отримання інформації та документообіг проходження заявок по робочих місцях користувачів додатку відповідно до логіки обробки заявок, побудова звітів.

*Функціональна структура* Системи повинна включати основні прикладні підсистеми, що виконують завдання обміну інформацією та обробки заявок на безготівкові, готівкові, рублеві і валютні платежі, що здійснюються бухгалтерією і фінансовою службою, а також забезпечують підсистеми, що виконують завдання підтримки спільної роботи всіх складових Системи.

#### **4.1.2. Вимоги до режимів функціонування системи**

Повинна забезпечуватися робота в двох режимах:

- мережевий режим взаємодії;
- автономний.

#### **4.1.3. Вимоги до способів і засобів зв'язку для обміну між компонентами системи**

- Обмін даними між підсистемами повинен здійснюватися через єдиний інформаційний простір і за допомогою використання стандартизованих протоколів і форматів обміну даними.
- Всі програмні компоненти підсистем повинні функціонувати в межах єдиного логічного простору, забезпеченого інтегрованими засобами серверів даних і серверів додатків.

#### **4.1.4. Вимоги до сумісності з суміжними системами**

- Програмне забезпечення системи має забезпечувати інтеграцію і сумісність на інформаційному рівні з іншими системами. Інформаційна сумісність повинна забезпечуватися, на рівні експорту-імпорту XML-документів.
- Вимоги до складу даних і режимам інформаційного обміну між підсистемами і системами визначаються в загальному регламенті взаємодії.
- Необхідними умовами, що накладаються на архітектуру взаємодії, є:
  - узгодженість з розробленими регламентами використання системи;
  - використання відкритих форматів обміну при організації взаємодії між підсистемами і системами, експлуатуються на об'єкті.

#### **4.1.5. Перспективи розвитку системи**

*ПС повинна мати тривалий життєвий цикл.*

*ПС повинна бути побудована з використанням стандартизованих і ефективно супроводжуваних рішень.*

*ПС повинна бути реалізована як відкрита система, і повинна допускати нарощування функціональних можливостей.*

*ПС повинна забезпечувати можливість модернізації як шляхом заміни технічного та загального програмного забезпечення (ПЗ), так і шляхом вдосконалення інформаційного забезпечення.*

#### **4.1.6. Вимоги до чисельності та кваліфікації персоналу та режиму його роботи**

**Вимоги до чисельності та кваліфікації персоналу та режиму його роботи**

Кількість користувачів *ПС* визначається поточними потребами ВАТ "Оргсинтез".

Кількість адміністраторів *ПС* може бути визначене за такою методикою: 1 адміністратор на 20-30 користувачів плюс 1 провідний спеціаліст або 1 начальник відділу автоматизації.

*Поточний контроль* технічного стану обладнання *АСУ* слід покласти на відділ автоматизації.

Перелік заходів поточного контролю технічного стану обладнання *АСУ* повинен бути узгоджений на стадії передпроектного обстеження.

## **Вимоги до кваліфікації персоналу**

Користувачі *ПС* повинні мати базові навички роботи з операційними системами Microsoft (будь-яка з версій: Microsoft Windows 7, 8.1, офісним програмним забезпеченням Microsoft Office).

Технічне обслуговування та адміністрування обладнання *ПС* повинно виконуватися фахівцями, які мають відповідну кваліфікацію та навички виконання робіт.

Всі адміністратори *ПС* повинні мати кваліфікацію "інженер" і обов'язкові навички *адміністрування мережі* на основі операційної системи Microsoft Windows 8.1.

### **4.1.7. Показники призначення**

Цільове призначення системи повинне зберігатися протягом усього терміну експлуатації *ПС* ЗАТ "Оргсинтез". Термін експлуатації *ПС* ЗАТ "Оргсинтез" визначається терміном стійкої роботи апаратних засобів обчислювальних комплексів, своєчасним проведенням робіт по заміні (оновленню) апаратних засобів, з супроводу програмного забезпечення системи і його модернізації.

Час виконання запитів інформації в *ПС* визначається на стадії проектування системи.

Спеціальні вимоги до ймовірно-часових характеристик, при яких зберігається цільове призначення *ПС* ЗАТ "Оргсинтез", визначаються відповідними вимогами до прикладних систем.

Інші показники призначення *ПС* розробляються після проведення передпроектного обстеження.

### **4.1.8. Вимоги до надійності**

## **Показники надійності**

Час відновлення працездатності прикладного ПО при будь-яких збоях і відмовах не повинно перевищувати одного робочого дня, крім випадків несправності серверного обладнання.

Інші значення показників надійності повинні бути визначені після проведення передпроектного обстеження.

### **Вимоги до надійності**

В *ПС* повинна бути забезпечена коректна обробка збоїв електронно-механічних пристроїв (наприклад, принтерів) при виконанні функцій, пов'язаних з формуванням твердих копій документів.

В *ПС* повинна бути забезпечена можливість "гарячої" заміни збійного або вийшов з ладу активного накопичувача на жорсткому магнітному диску (серверного устаткування *ПС*) без зупинки функціонування і втрат інформації.

В *ПС* повинна бути забезпечена можливість відновлення даних з зовнішнього накопичувача після відновлення активного накопичувача. Конкретний склад вимог з відновлення даних доповнюється відповідними вимогами на підсистеми.

Має здійснюватися розмежування прав доступу до системи.

Повинен вестися журнал подій системи.

Імпульсні перешкоди, збої або припинення електроживлення не повинні призводити до виходу з ладу технічних засобів *ПС*, що знаходяться в спеціально обладнаному приміщенні і підключених до системи безперебійного електропостачання, в т.ч. автономного. Конкретний склад вимог щодо захисту обладнання від імпульсних перешкод, збоїв і припинення електроживлення доповнюється відповідними вимогами на підсистеми.

В *ПС* всіх рівнів повинні бути реалізовані функції коректної автоматичної зупинки роботи технічних засобів, підключених до системи безперебійного електропостачання, в т.ч. автономного, при тривалій відсутності електроживлення.

#### ***4.1.9. Вимоги по ергономіці і технічній естетиці***

### **Вимоги до зовнішнього оформлення**

Реалізація графічного багатовіконного режиму.

Настроюваність графічних елементів інтерфейсу, у тому числі кольорового оформлення, в межах можливостей операційної системи.

## Вимоги до діалогу з користувачем

Інтерфейс повинен забезпечувати зручну навігацію в діалозі з користувачем, який добре знає свою предметну область і не є фахівцем в області автоматизації.

Наявність контекстно-залежної допомоги.

### **4.1.10. Вимоги щодо безпеки**

При монтажі, наладці, експлуатації, обслуговуванні та ремонті технічних засобів Системи повинні виконуватися заходи електробезпеки відповідно до "Правил улаштування електроустановок" і "Правилами техніки безпеки при експлуатації електроустановок споживачів".

Апаратне забезпечення Системи повинно відповідати вимогам пожежної безпеки у виробничих приміщеннях по ГОСТ 12.1.004-91. "ССБТ. Пожежна безпека. Загальні вимоги".

Повинно бути забезпечено дотримання загальних *вимог безпеки* відповідно до ГОСТ 12.2.003-91. "ССБТ. Обладнання виробниче. Загальні *вимоги безпеки* "при обслуговування Системи в процесі експлуатації".

Апаратна частина Системи повинна бути заземлена відповідно до вимог ГОСТ Р 50571.22-2000. "Електроустановки будівель. Частина 7. Вимоги до спеціальних електроустановок. Розділ 707. Заземлення устаткування обробки інформації".

Значення еквівалентного рівня акустичного шуму, створюваного апаратурою Системи, повинно відповідати ГОСТ 21552-84 "Засоби обчислювальної техніки. Загальні технічні вимоги, приймання, методи випробувань, маркування, упаковка, транспортування і зберігання", але не перевищувати таких величин:

- 50 дБ - при роботі технологічного обладнання та засобів обчислювальної техніки без *друкувального пристрою*;
- 60 дБ - при їх же роботі з *друкуючим пристроєм*.

### **4.1.11. Вимоги до експлуатації, технічного обслуговування, ремонту та зберігання**

Система повинна забезпечувати безперервний цілодобовий режим експлуатації з урахуванням часу на технічне обслуговування.

У приміщеннях, призначених для експлуатації Системи, повинні бути відсутніми агресивні середовища, масова концентрація пилу в повітрі має бути не більше  $0,75 \text{ мг / м}^3$ , електрична складова електромагнітного поля перешкод не повинна перевищувати  $0,3 \text{ Н/м}$  в діапазоні частот від 0,15 до 300,00 МГц.

Напруга живлення мережі має бути 220 В, 50 Гц.

Вимоги щодо забезпечення пожежної безпеки та електробезпеки (заземлення) у приміщеннях повинні бути виконані відповідно до ГОСТ

12.1.004-91 "ССБТ. Пожежна безпека. Загальні вимоги", ГОСТ Р 50571.22-2000. "Електроустановки будівель. Частина 7. Вимоги до спеціальних електроустановок. Розділ 707. Заземлення устаткування обробки інформації", "Правилами улаштування електроустановок", "Правилами техніки безпеки при експлуатації електроустановок споживачів".

Кліматичні чинники приміщення для експлуатації виробів повинні бути по ГОСТ 15150-69 (з ізм. 2004) "Машины, прилади та інші технічні вироби. Виконання для різних кліматичних районів. Категорії, умови експлуатації, зберігання і транспортування в частині впливу кліматичних факторів зовнішнього середовища" для виду кліматичного виконання УХЛ категорії 4.2.

Нормальними кліматичними умовами експлуатації системи є:

- температура навколишнього повітря +15 - +25°C;
- відносна вологість навколишнього повітря 75% при атмосфері повітря +18° С;
- атмосферний тиск 740 – 770 мм рт.ст.

Система повинна зберігати працездатність при впливі наступних кліматичних факторів:

- температура навколишнього повітря від 10 до 45 ;
- відносна вологість повітря від 40 до 80% при температурі +10°C .

#### ***4.1.12. Вимоги щодо збереження інформації***

Захист даних від руйнувань при аваріях і збоях

**Повинна забезпечуватися збереження інформації при настанні таких подій:**

- відмова обладнання робочої станції, у разі зберігання даних на сервері;
- відключення живлення на сервері баз даних;
- відмова ліній зв'язку;
- відмова апаратури сервера (процесор, накопичувачі на жорстких дисках).

**Засобами забезпечення збереження інформації при аваріях і збоях в процесі експлуатації є:**

- носії інформації (змінні: оптичні - дискові або магнітні - стрічкові, накопичувачі на змінних жорстких дисках);



- створення резервної копії бази даних;
- створення резервної копії програмного забезпечення.

Для відновлення даних та програмного забезпечення з резервної копії повинні використовуватися кошти резервного копіювання та архівування. *ПС* повинна забезпечувати можливість резервування всіх даних, що зберігаються на сервері, а також можливість їх відновлення.

Резервне копіювання даних повинне здійснюватися експлуатаційним персоналом ЗАТ "Оргсинтез" щодня, автоматично за розкладом. Для скорочення обсягу копіюваних даних процедура копіювання може бути інкрементальною (копіювання тільки змін з попереднього копіювання), але при цьому не рідше разу на тиждень повинно проводитися і повне копіювання.

Повинна бути передбачена можливість відновлення даних за день збою за допомогою їх повторного введення або імпорту (для даних із зовнішніх систем, одержуваних автоматично).

## **4.2. Вимоги до видів забезпечення**

### **4.2.1. Загальні відомості**

Підсистема створюється як об'єктова комплексна програмна система, яка повинна бути організованою в єдине ціле сукупністю частин, тобто являти собою комплекс різних видів забезпечення. Основними з видів забезпечення Системи є організаційне, інформаційне, програмне та технічне забезпечення системи.

Серверне та клієнтське програмне забезпечення *ПС* має базуватися на погодженій з Замовником та Виконавцем поширеною промисловою *мережевий операційній системі*.

Вимоги до організаційного, інформаційного та програмного забезпечення наведені в Приватних технічних завданнях на створення підсистем "Програмно-апаратний комплекс **"Платежі та взаєморозрахунки з кредиторамі"**".

У цьому ТЗ наведені специфічні вимоги до лінгвістичного і *технічного забезпечення*.

### **4.2.2. Вимоги до лінгвістичного забезпечення**

Загальні вимоги до *лінгвістичного забезпечення* наведені в Приватних технічних завданнях на створення підсистем "Програмно-апаратний комплекс **ПС "Платежі та взаєморозрахунки з кредиторамі"**" на об'єктах автоматизації.

## *Мови програмування*

Розробка прикладного програмного забезпечення повинна вестися з використанням мов високого рівня.

## *Мови взаємодії користувачів і системи*

Основною мовою взаємодії користувачів і системи є українська мова:

- взаємодія користувача з ПК повинно здійснюватися українською мовою (виняток можуть становити лише системні повідомлення, що видаються програмними продуктами третіх компаній);
- всі документи і звіти Підсистеми готуються і виводяться користувачеві українською мовою;
- графічний інтерфейс користувача Підсистеми повинен бути створений українською мовою.

## *Мови взаємодії адміністраторів та системи*

Комплектування Підсистеми програмним забезпеченням і документацією англійською мовою допускається тільки в тому випадку, якщо це програмне забезпечення і документація використовуються тільки адміністратором системи.

### **4.2.3. Вимоги до технічного забезпечення**

Нижче представлений перелік мінімальних вимог, що пред'являються до компонентів апаратного та програмного забезпечення Підсистеми на об'єктах автоматизації.

Технічне забезпечення із зазначеними характеристиками має бути достатньо для введення Підсистеми в *дослідну експлуатацію* на об'єкті автоматизації. У ході *дослідної експлуатації* вимоги до характеристик повинні бути уточнені, при перекладі Підсистеми в промислову експлуатацію та її введенні в дію на нових об'єктах автоматизації може знадобитися модернізація або заміна технічних засобів на обладнання з іншими характеристиками.

### *Загальні вимоги*

Всі серверне обладнання повинно монтуватися в стандартні стійки 42У. Кожна стійка повинна обладнуватися системою охолодження і стабілізаторами електроживлення. Устаткування має бути підключений з використанням ІБП і управлятися через консоль *KVM* (з консольним ЖК монітором і суміщеною з клавіатурою мишею).

Для забезпечення регулярного резервного копіювання необхідно передбачити можливість встановлення обладнання для виконання резервного копіювання інформації на локальні стрічкові накопичувачі формату *DLT* з відповідною розміром банків даних ємністю набору

змінних носіїв, або можливість використання технічних засобів, що дозволяють проводити резервне копіювання на виділене мережеве пристрій.

#### **4.2.4. Вимоги до програмного забезпечення**

Програмне забезпечення повинно поставлятися Замовнику на оптичних (CD-ROM) носіях в наступному складі:

- комплект файлів, необхідних для установки системи і роботи користувача;
- комплект файлів, необхідних для супроводу та модернізації прикладної системи.

#### **4.2.5. Вимоги до технічного забезпечення**

Детальні вимоги до *технічного забезпечення* повинні бути сформульовані після проведення передпроектного обстеження.

#### **4.2.6. Вимоги до організаційного забезпечення**

Для забезпечення впровадження та ефективної роботи ЗАТ "Оргсинтез" з використанням прикладної системи рекомендується на договірному рівні призвести регламентацію взаємовідносин між ЗАТ "Оргсинтез" і ЗАТ "Програмні системи" по наступних позиціях:

##### **1. Права Виконавця:**

- Одержувати доступ до інформації, що надається прикладними системами ПУ.
- Посилати пропозиції для формування інформації, що розміщується в прикладних системах.

##### **2. Обов'язки Виконавця:**

- Організувати робочі місця і обладнати їх засобами обчислювальної техніки, периферійним обладнанням, програмним забезпеченням і засобами зв'язку, що забезпечують своєчасне і достовірне надання інформації відповідно до *вимогами Замовника*.
- Забезпечити ведення журналу обліку одержуваних приписів, рекомендації з проведення робіт, донесень та іншої інформації, одержуваної від Замовника.
- Організувати профілактичні заходи та роботи урахуванням інформації, одержуваної від прикладних систем Замовника.
- Надавати Замовнику інформацію про проведені заходи і виконуваних роботах відповідно до регламенту.
- Своєчасно інформувати Замовника про ліквідацію наслідків позаштатних ситуацій.
- Оперативно усувати недоліки за приписом Замовника з відображенням факту виконання робіт у журналі обліку.
- Надавати плани заходів та робіт за запитом Замовника.

##### **3. Права Замовника:**

- Видавати приписи на виконання робіт у випадках порушення технології утримання та невиконання нормативних вимог.
  - Вимагати надання планів заходів та робіт на підставі даних прикладних систем.
  - Контролювати несення чергувань і ведення журналу обліку.
  - При щомісячній прийманні виконаних робіт і послуг, зіставляти представлені обсяги і види робіт з даними, що отримуються від прикладних систем; при істотному розходженні цих даних вимагати надання обґрунтувань.
4. **Обов'язки Замовника:**
- Формувати і передавати інформацію, що сприяє ефективній роботі Виконавця з використанням прикладних систем.
  - Надавати дані про здійснення взаєморозрахунків з *кредиторами*.
  - Надати доступ до необхідної інформації.
  - Забезпечити регулярне оновлення інформації, що розміщується на сайті.

5. **Відповідальність сторін:**

- Виконавець несе майнову відповідальність (штрафні санкції) за несвоєчасне виконання приписаних обов'язків, у разі якщо інформація від Замовника була отримана своєчасно.
- Виконавець зобов'язаний надавати обґрунтовують матеріали за фактом істотної розбіжності обсягів окремих видів робіт, обсягу та видів виконаних робіт в цілому, представлених при прийманні робіт, порівняно даними, одержуваними від ПС.

**5. Склад і зміст робіт зі створення системи**

Здійснення всього комплексу *робіт* зі створення повинно здійснюватися в кілька черг. Специфікація *робіт* зі створення першої черги ПС "Платежі та взаєморозрахунки з *кредиторами*" в обсязі вимог цього ТЗ наведена в табл. 1.

Таблиця 1.

Стадія робіт	Виконувані роботи	Терміни	Підсумки виконання роботи
Формування вимог	Обстеження об'єктів автоматизації	виконано	Звіт про результати обстеження
	Розробка Приватного технічного завдання		Затвердження замовником ЧТЗ на

	на створення Підсистеми		створення Підсистеми
Проектування	Розробка <i>технічного проекту</i> на Підсистему Розробка прототипу Підсистеми	2 місяці з початку виконання робіт	<i>Технічний проект</i> на Підсистему Специфікації програмно-апаратних засобів Підсистеми
	Розробка проектів організаційно-розпорядчої, програмної та експлуатаційної документації на Підсистему		
Поставка програмно-технічних засобів для <i>дослідної експлуатації</i>	Поставка програмно-технічних засобів (ліцензійне ПЗ) для <i>дослідної експлуатації</i> на об'єктах автоматизації, що входять до складу дослідної зони	3 місяці з початку виконання робіт	Акти
Розробка програмних засобів	Розробка, налагодження і тестування програмних засобів Підсистеми	4 місяці з початку виконання робіт	Програмні засоби на машинозчитуваних носіях Комплект проектів організаційно-розпорядчої, програмної та експлуатаційної документації на Підсистему
Приймання робіт	Проведення <i>попередніх випробувань</i> на стенді Виконавця	4 місяці з початку виконання робіт	Протоколи випробувань Акт готовності підсистеми до розгортання в дослідній зоні

Проведення розгортання в дослідній зоні, впровадження та *дослідна експлуатація* виконуються за окремими ЧТЗ на *розгортання ПС* "Платежі та взаєморозрахунки з *кредиторами* "на об'єктах дослідної зони за окремими договорами.

Типовий склад *робіт* з розгортання, впровадження та *дослідної експлуатації* , який повинен бути передбачений у ЧТЗ на *розгортання* , наведено в табл. 2.

Таблиця 2

Вид робіт	Склад робіт
Підготовка регламентів застосування (посадових інструкцій з експлуатації Підсистеми)	Регламент застосування користувачів підсистеми
	Регламент застосування системного адміністратора
Навчання	Навчання користувачів
	Навчання адміністраторів
Розгортання підсистеми	Монтаж та пусконаладження серверів
	Установка серверного ПЗ
	Установка ПЗ на робочі станції користувачів
	Установка ПЗ на робочі станції адміністраторів
	Налаштування процедур резервного копіювання
Впровадження підсистеми	Введення структури довідників і <i>класифікаторів</i>
	Імпорт і введення довідників і <i>класифікаторів</i>
	Створення БД заявок
	Налаштування процесів документообігу по обробці заявок
	Налагодження та тестування взаємодії між об'єктами дослідної зони
	Налаштування процедур аналітичної обробки статистики роботи системи
	Участь у комплексі робіт щодо забезпечення інформаційної безпеки

	Проведення випробувань на об'єкті, передача в <i>дослідну експлуатацію</i>
<i>Дослідна експлуатація</i>	Технічна підтримка на протязі <i>дослідної експлуатації</i>
	Усунення помилок в розробленому ПЗ

## 6. Порядок контролю та приймання системи

Випробування Підсистеми повинні проводитися відповідно до вимог ГОСТ 34.603-92 "Інформаційна технологія. Види випробувань автоматизованих систем". При реалізації Підсистеми в рамках цієї ТЗ встановлюються *попередні випробування* на стенді Виконавця по створенню Підсистеми.

Випробування Підсистеми повинні здійснюватися відповідно до документа "Програма та методика випробувань", який повинен встановлювати необхідний і достатній обсяг випробувань, що забезпечує необхідний рівень достовірності отриманих результатів. *Програма* та методика випробувань затверджується Замовником.

Приймання *робіт* повинна здійснювати приймальна комісія, до складу якої включаються:

- представники Замовника;
- представники Виконавця.

При проведенні випробувань приймальної комісії пред'являються розроблені Виконавцем матеріали (конструкторська, програмна та експлуатаційна документація та *програмне забезпечення* у вихідних і виконуваних кодах). Комплектність наданої документації визначається вимоги цього ТЗ.

*Попередні випробування* закінчуються підписанням приймальною комісією протоколу випробування із зазначенням у ньому переліку необхідних доробок програмного забезпечення, конструкторської, програмної та експлуатаційної документації та термінів їх виконання. Після усунення зауважень, здійснюються повторні *попередні випробування* Підсистеми. На повторні *попередні випробування* Виконавцем пред'являються допрацьовані за результатами раніше виконаних випробувань матеріали. Випробування завершуються оформленням Акта готовності Підсистеми до розгортання в дослідній зоні.

Окремі пункти ТЗ можуть змінюватися і уточнюватися за погодженням сторін.

У тижневий термін після початку *робіт виконавець* надає на погодження ВАТ "Оргсинтез" план-графік *робіт* по даному етапу.

## 7. Вимоги до документування

### **7.1. Загальні вимоги до документування**

Документи повинні бути представлені на паперовому вигляді (оригінал) та на магнітному носії (копія). Вихідні тексти програм - тільки на магнітному носії (оригінал). Можливе надання комплекту документації і текстів програм на компакт-дисках.

Всі документи повинні бути оформлені російською мовою. Склад документів на загальне програмне забезпечення, що постачається в складі ПС "Платежі та взаєморозрахунки з *кредиторами*", повинен відповідати комплекту поставки компанії - виробника.

### **7.2. Перелік документів, що підлягають розробці**

У ході створення Підсистеми повинен бути підготовлений і переданий Замовнику комплект документації у складі:

- проектна документація та матеріали техно-робочого проекту на розробку Підсистеми;
- конструкторська, програмна та експлуатаційна документація на Підсистему;
- супровідна документація на поставляються програмно-апаратні засоби в комплектності поставки заводом-виробником;
- пропозиції щодо організації системно-технічної підтримки функціонування Підсистеми.

Склад і зміст комплекту документації на Підсистему може бути уточнений на стадії проектування.

Підготовлені документи повинні задовольняти вимогам державних стандартів і рекомендацій щодо оформлення, змісту, форматування, використання термінів, визначень і написів, позначень програм і програмних документів.

### **8. Вимоги до складу та змісту робіт з підготовки об'єкта автоматизації до введення системи в дію**

У процесі Створення Підсистеми повинен бути підготовлений і переданий Замовнику комплект документації у складі:

- проектна документація та матеріали техно-робочого проекту на розробку Підсистеми;
- конструкторська, програмна та експлуатаційна документація на Підсистему;
- супровідна документація на поставляються програмно-апаратні засоби в комплектності поставки заводом-виробником;
- пропозиції щодо організації системно-технічної підтримки функціонування Підсистеми.

Склад і зміст комплекту документації на Підсистему може бути уточнений на стадії проектування.



Підготовлені документи повинні задовольняти вимогам державних стандартів і рекомендацій щодо оформлення, змісту, форматування, використання термінів, визначень і написів, позначень програм і програмних документів.

**9. Порядок внесення змін**

Справжнє ТЗ може доповнюватися і змінюватися в процесі розробки та *приймальних випробувань* в установленому порядку за взаємною згодою Замовника і Розробника.

		Розробники		Підписи	
Організація, підприємство	Посада	Прізвище, ім'я, по батькові		Підпис	Дата
Організація, підприємства	Посада	Прізвище, ім'я, по батькові		Підпис	Дата

*Навчальне видання*

КУЧЕРОВ Дмитро Павлович  
Ніколаєнко Олександр Євгенович  
Мацуєва Каріна Андріївна

Інженерія програмного забезпечення  
Лабораторний практикум

Редактор  
Технічний редактор

Підп. до друку 00.00.14. Формат 60х84/16. Папір офс.  
Офс. друк. Ум. друк. арк. 10,0. Обл.-вид. арк. 10,75.  
Тираж 100 прим. Замовлення № 000-0.

Видавець і виготовлювач  
Національний авіаційний університет  
03680. Київ-58, проспект Космонавта Комарова, 1.  
Свідоцтво про внесення до Державного реєстру ДК № 977 від 05.07.2002