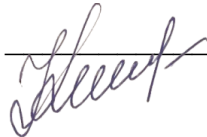



МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ОДЕСЬКИЙ ДЕРЖАВНИЙ ЕКОЛОГІЧНИЙ УНІВЕРСИТЕТ

МЕТОДИЧНІ ВКАЗІВКИ (Частина 2)
до лабораторних занять з навчальної дисципліни
«Системний аналіз і проектування інформаційних систем»
для студентів денної та заочної форми навчання
спеціальність – «комп'ютерні науки»

Затверджено
на засіданні групи забезпечення спеціальності
Протокол № 13 від « 30 » березня 2023р.
Голова групи  Кузніченко С.Д.

Затверджено
на засіданні кафедри _____ ІТ _____
Протокол № 6 від «16» березня 2023р.
Завідувач кафедри  Казакова Н. Ф.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ОДЕСЬКИЙ ДЕРЖАВНИЙ ЕКОЛОГІЧНИЙ УНІВЕРСИТЕТ

МЕТОДИЧНІ ВКАЗІВКИ (Частина 2)
до лабораторних занять з навчальної дисципліни
«Системний аналіз і проектування інформаційних систем»
для студентів денної та дистанційної форми навчання
спеціальність – «комп'ютерні науки»

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ОДЕСЬКИЙ ДЕРЖАВНИЙ ЕКОЛОГІЧНИЙ УНІВЕРСИТЕТ

МЕТОДИЧНІ ВКАЗІВКИ (Частина 2)
до лабораторних занять з навчальної дисципліни
«Системний аналіз і проектування інформаційних систем»
для студентів денної та дистанційної форми навчання
спеціальність – «комп'ютерні науки»

Затверджено
на засіданні групи забезпечення спеціальності
Протокол № _ _
від «_ _» __2023р.

МЕТОДИЧНІ ВКАЗІВКИ до виконання лабораторних робіт (частина 2) з навчальної дисципліни «Системний аналіз і проектування інформаційних систем» для студентів денної та заочної форми навчання напрямку – «комп'ютерні науки».

Укладачі: Казакова Н. Ф., к.т.н., проф., завідувач кафедри інформаційних технологій, Штефан Н. З., ст. викл. кафедри інформаційних технологій. ОДЕкУ, 2023, 133 с., укр. мова.

ЗМІСТ

ВСТУП.....	7
ПРАВИЛА ТЕХНІКИ БЕЗПЕКИ ТА ОХОРОНИ ПРАЦІ	8
ВАРІАНТИ ЗАВДАНЬ	9
Варіант №1.....	9
Варіант №2.....	10
Варіант №3.....	11
Варіант №4.....	13
Варіант №5.....	14
Варіант №6.....	16
Варіант №7.....	17
Варіант №8.....	18
Варіант №9.....	20
Варіант №10.....	21
Варіант №11.....	22
Варіант №12.....	24
Варіант №13.....	25
Варіант №14.....	27
Варіант №15.....	28
Варіант №16.....	29
Варіант №17.....	31
Варіант №18.....	32
Варіант №19.....	33
Варіант №20.....	35
Варіант №21.....	36
Варіант №22.....	38
Варіант №23.....	39
Варіант №24.....	40

ЛАБОРАТОРНА РОБОТА №5 «СТВОРЕННЯ МОДЕЛІ ДАНИХ В СЕРЕДОВИЩІ ERwin».....	42
Практична підготовка до виконання лабораторної роботи:	44
ЛАБОРАТОРНА РОБОТА №6.....	75
«ФІЗИЧНА МОДЕЛЬ ДАНИХ ТА ГЕНЕРАЦІЯ КОДУ КЛІЄНТСЬКОЇ ЧАСТИНИ ЗА ДОПОМОГОЮ ERWIN»	75
Практична підготовка до виконання лабораторної роботи	76
ЛАБОРАТОРНА РОБОТА №7.....	91
«ПОБУДОВА ДІАГРАМИ ВАРІАНТІВ ВИКОРИСТАННЯ».....	91
Практична підготовка до виконання лабораторної роботи	94
ЛАБОРАТОРНА РОБОТА № 8.....	117
«ВИЗНАЧЕННЯ СКЛАДОВИХ ЕЛЕМЕНТІВ ПРОЕКТУ».....	117
Практична підготовка до виконання лабораторної роботи	123
ЛАБОРАТОРНА РОБОТА №9.....	129
«СКЛАДАННЯ ТЕХНІЧНОГО ЗАВДАННЯ ДО ПРОЕКТУ».....	129
ЛІТЕРАТУРА ОСНОВНА.....	134
ДОДАТКОВА:.....	134

ВСТУП

Дане видання призначене для вивчення та практичного засвоєння студентами всіх форм навчання з основ системного проєтування та управління інформаційними системами. Ця галузь ІТ-технологій охоплює принципи і методології проєтування а також підготовку технічної документації проєкту.

В даних методичних вказівках розміщені основні, базові теоретичні відомості, необхідні для виконання лабораторних робіт. Таким чином для успішного виконання лабораторної роботи та при підготовці до її захисту відповідно до графіка студенти повинні ознайомитися з конспектом лекцій та рекомендованою літературою.

Для одержання заліку з кожної роботи студент здає викладачу цілком оформлений звіт (студент завантажує в електронному вигляді у систему Moodle), а також демонструє на екрані комп'ютера результати виконання лабораторної роботи (в аудиторії/за допомогою Zoom).

Звіт має містити:

- титульний аркуш;
- тему та мету роботи;
- завдання до роботи;
- знімки екрану, що відображають результати роботи;
- відповіді на контрольні питання до роботи;
- змістовний аналіз отриманих результатів та висновки.

Звіт виконують у електронному форматі А4(210×297 мм). Поля сторінки з усіх боків – 20 мм.

Під час співбесіди при захисті лабораторної роботи студент повинен виявити знання про мету роботи, по теоретичному матеріалу, про методи виконання кожного етапу роботи, по змісту основних розділів оформленого звіту з демонстрацією результатів на конкретних прикладах.

Студент повинен вміти правильно аналізувати отримані результати. Для самоперевірки при підготовці до виконання і захисту роботи студент повинен відповісти на контрольні запитання, наведені наприкінці опису відповідної роботи.

ПРАВИЛА ТЕХНІКИ БЕЗПЕКИ ТА ОХОРОНИ ПРАЦІ

Кожен студент перед початком учбової практики мусить вивчити правила роботи з комп'ютерною технікою в лабораторіях кафедри інформаційних технологій, пройти співбесіду з інструктором по техніці безпеки та розписатися в журналі по техніці безпеки.

ВАРІАНТИ ЗАВДАНЬ

Варіант №1

«ІНФОРМАЦІЙНА СИСТЕМА БІБЛІОТЕКИ ВНЗ»

Опис предметної галузі.

Читачами бібліотеки ВНЗ мають право бути: студенти усіх форм навчання, професорсько-викладацький склад університету, аспіранти, асистенти та інші співробітники підрозділів ВНЗ, слухачі підготовчого відділення, факультети підвищення кваліфікації, стажери, абітурієнти. Різні категорії читачів серед інших мають характеристики, специфічні для своєї категорії: для студентів це назва факультету, номер групи, для викладача – посада, ступінь, звання. Слухачі ФПК, абітурієнти, стажери – разові читачі мають право користуватися лише читальними залами.

Читачі бібліотеки мають право отримувати книги та інші джерела інформації на всіх пунктах видачі бібліотеки: абонементи та читальні зали. Читачі, які приходять на пункт видачі, зобов'язані мати при собі читацький квиток з відмітками про запис та перереєстрацію цього року на даному пункті видачі.

За порушення правил користування бібліотекою читачі позбавляються права користуватись усіма пунктами обслуговування бібліотеки на встановлені адміністрацією терміни. У разі втрати або псування книг читач зобов'язаний замінити їх такими ж або іншими виданнями, визнаними бібліотекою рівноцінними, або відшкодувати їх 10-кратну вартість. При вибуття з ВНЗ (відрахування, закінчення навчання, звільнення) читачі зобов'язані повернути видання, що числяться за ними, і здати читацькі квитки.

При надходженні нових видань до бібліотеки вони мають бути внесені до картотеки із зазначенням їх кількості для кожного абонементу та читального залу.

Передбачити можливість одержання наступної інформації:

1. Перелік читачів для даного читального залу або абонементу або по всій бібліотеці.

2. Список усіх читачів-боргів, боргів з терміном більше 10 днів на цьому абонементі або по всій бібліотеці.
3. Перелік двадцяти книг, що найбільш часто замовляються в даному читальному залі для даного факультету, для всього вузу.
4. Перелік книг, що надійшли за останній рік або загублених за останній рік, для цього читального залу, абонементу або по всій бібліотеці за вказаним автором.
5. Визначити, чи є ця книга в наявності на абонементів та в якій кількості.
6. Повну інформацію про читача на його прізвище: групу, курс, факультет або кафедру, правопорушення, їх кількість, штрафи, втрачені книги.

Варіант №2

«ІНФОРМАЦІЙНА СИСТЕМА ТУРИСТИЧНОГО КЛУБУ»

Опис предметної галузі.

Туристи, що приходять до туристичного клубу, займаються в різних секціях протягом усього року. Для цього вони записуються до груп по секціях. Туристів можна розділити на любителів, спортсменів та тренерів зі своїм набором характеристик-атрибутів. Секції клубу очолюють керівники. У роботі секції беруть участь тренери, які належать до однієї із секцій і можуть тренувати декілька груп.

Щороку складається розклад роботи секцій. У ньому вказується, які проводитимуться тренування, та у яких секціях: їх кількість, місце, час тощо. Відповідно до цього керівники секцій здійснюють розподіл навантаження для тренерів. Відомості про проведені тренування та відвідуваність тренувань збираються керівниками.

Протягом року клуб організовує різноманітні походи. Кожен похід має свій маршрут, на який приділяється певна кількість днів. За маршрутом та

кількістю днів визначається категорія складності даного походу. Похід очолює інструктор, яким може бути тренер або спортсмен. Він набирає групу в кількості 5-15 осіб для свого походу, виходячи з типу походу (піший, кінний, водний, гірський) та фізичних даних туристів за їхніми заняттями у секціях.

Походи можуть бути плановими та неплановими. Для кожного планового походу існує точний план, у якому вказується маршрут, розклад привалів та стоянок щодня. Під час планового походу триває щоденник. Непланові походи мають лише маршрут та повний час його проходження. Кожному туристові надається категорія максимально складного з пройдених ним планових походів.

Передбачити можливість одержання наступної інформації.

1. Список туристів, які займаються у клубі, у зазначеній секції, групі.
1. Список тренерів зазначеної секції по всіх секціях.
2. Список тренерів, які проводили тренування у зазначеній групі, за вказаний період.
3. Список туристів із певної секції, групи, які ходили у задану кількість походів, ходили у зазначений похід.
4. Навантаження тренерів, її обсяг за певними видами занять та загальне навантаження за вказаний період часу.
5. Перелік маршрутів, якими ходили туристи із зазначеної секції, у зазначений період.
6. Перелік маршрутів, якими водив свої групи даний інструктор.
7. Перелік маршрутів, якими пройшла зазначена кількість груп.

Варіант №3

«ІНФОРМАЦІЙНА СИСТЕМА ТОРГІВЕЛЬНОЇ ОРГАНІЗАЦІЇ»

Опис предметної області.

Торгова організація торгує в торгових точках різних типів: універмаги, магазини, кіоски, лотки, у штаті яких працюють продавці. Універмаги та

магазини розділені на окремі секції, керовані керуючими секціями. У кожній секції працює кілька продавців. Універмаги, магазини, кіоски можуть мати такі характеристики, як розмір торгової точки, платежі за оренду, комунальні послуги, кількість прилавків тощо.

Замовлення постачальнику на постачання товарів складаються на основі заявок, що надходять із торгових точок. На основі заявок менеджери торгової організації обирають постачальника, формують замовлення, в яких перераховуються найменування товарів і кількість, що замовляється, яка може відрізнитися від запиту з торгової точки. На основі маркетингових робіт постійно вивчається ринок постачальників, у результаті можуть з'являтися нові постачальники та зникати старі. У цьому одні й самі товари торгова організація може отримувати від різних постачальників і, звісно, за цінами.

Товари, що надійшли, розподіляються по торгових точках у відповідність до заявок. Продавці торгових точок ведуть продаж товарів, враховуючи всі продажі, фіксуючи найменування та кількість проданого товару, а продавці універмагів та магазинів додатково фіксують імена та характеристики покупців, що дозволяє вести облік покупців та зроблених ними покупок.

Передбачити можливість одержання наступної інформації.

1. Перелік постачальників, які постачають вказаний вид товару за вказаний період.
2. Перелік покупців, які купили зазначений вид товару за певний період в обсязі, не менший за заданий.
3. Найменування та обсяг товарів у зазначеній торговій точці.
4. Дані про вироблення кожного продавця за вказаний період за всіма торговими точками.
5. Дані щодо вироблення окремо взятого продавця окремо взятої торгової точки за вказаний період.
6. Відомості про постачання певного товару вказаним постачальником за певний період.

7. Відомості про найактивніших покупців з цієї торгової точки.
8. Дані про товарообіг торгової точки за вказаний період.

Варіант №4

«ІНФОРМАЦІЙНА СИСТЕМА СПОРТИВНИХ ЗАХОДІВ»

Опис предметної галузі.

Спортивну базу міста представлено спортивними спорудами різних типів: спортивні зали, манежі, стадіони, корти, басейни. Кожна з категорій спортивних споруд має атрибути, специфічні тільки для неї: стадіон характеризується місткістю, kort – типом покриття, басейн – довжиною доріжки.

Спортсмени під керівництвом тренерів займаються окремими видами спорту, при цьому той самий спортсмен може займатися кількома видами спорту, а в рамках одного виду спорту може тренуватися тільки в одного тренера. Усі спортсмени об'єднуються у спортивні клуби, при цьому кожен із них може виступати лише за один клуб. Кожен спортсмен у певному виді спорту може мати відповідний розряд (перший, другий, третій, кандидат у майстри спорту, майстер спорту). Кожен тренер має відповідну кваліфікацію.

Спортсмени можуть брати участь у змаганнях. При цьому вони можуть змагатися як в особистій першості, так і командній. Організатори змагань проводять змагання з окремих видів спорту на спортивних спорудах міста. За результатами участі спортсменів у змаганнях проводиться визначення зайнятих ними місць, нагородження переможців та присвоєння відповідних розрядів.

Передбачити можливість одержання наступної інформації.

1. Перелік спортивних споруд зазначеного типу чи задовольняють заданим характеристикам.
2. Список спортсменів, які займаються зазначеним видом спорту або не нижче за певний розряд.

3. Список спортсменів, які тренуються у заданого тренера.
4. Список спортсменів, які займаються більш ніж одним видом спорту із зазначенням цих видів спорту.
5. Список тренерів вказаного клубу.
6. Список тренерів з певного виду спорту.
7. Перелік змагань, проведених протягом заданого часу.
8. Список призерів вказаного змагання.
9. Перелік змагань, проведених у зазначеній спортивній споруді.
10. Перелік спортивних клубів та спортсменів цих клубів, які брали участь у спортивних змаганнях протягом заданого часу.
11. Список організаторів змагань та кількість проведених ними змагань протягом певного періоду часу.
12. Перелік спортивних споруд та дати проведення на них змагань протягом певного періоду часу.

Варіант №5

«ІНФОРМАЦІЙНА СИСТЕМА МЕДИЧНИХ УСТАНОВ»

Опис предметної області.

Кожна міська лікарня складається з одного або кількох корпусів, у кожному з яких розміщується одне або кілька відділень, що спеціалізуються на лікуванні певної групи хвороб. Кожне відділення має кілька палат на кілька ліжок. До складу лікарні може входити поліклініка. Лабораторії, які виконують медичні аналізи, можуть обслуговувати різні лікарні та поліклініки. У цьому кожна лабораторія може проводити різноманітні дослідження: біохімічні, фізіологічні, хімічні.

Як лікарні, так і поліклініки обслуговуються лікарським персоналом (хірурги, терапевти, невропатологи, окулісти, стоматологи, рентгенологи, ендокринологи) та обслуговуючим персоналом (мед. сестри, санітари, прибиральниці). Кожна категорія має характеристики, властиві тільки

фахівцям цього профілю і по-різному бере участь у зв'язках. Дозволено сумісництво, тому кожен лікар може працювати в лікарні та в поліклініці, або у двох лікарнях, або у двох поліклініках. Лікарі будь-якого профілю можуть мати ступінь кандидата чи доктора медичних наук. Лікарі зі ступенями можуть консультувати у кількох лікарнях чи поліклініках.

Пацієнти амбулаторно лікуються в одній із поліклінік, і при необхідності направляються на стаціонарне лікування до своєї чи якоїсь іншої лікарні залежно від необхідного лікування. Як у лікарні, так і в поліклініці ведеться облік пацієнтів, повна історія їхніх хвороб, призначення, операції тощо. Пацієнт у кожний момент часу має одного лікаря в лікарні, і кілька – у поліклініці.

Передбачити можливість одержання наступної інформації.

1. Перелік лікарів зазначеного профілю для заданого медичного закладу. Перелік лікарів зі ступенем кандидата чи доктора медичних наук для заданого медичного закладу.
2. Перелік обслуговуючого персоналу зазначеної спеціальності для заданого медичного закладу.
3. Перелік пацієнтів зазначеної лікарні, відділення або конкретної палати зазначеного відділення із зазначенням дати надходження, стану, температури, лікаря.
4. Перелік пацієнтів, які спостерігаються у лікаря зазначеного профілю у конкретній поліклініці.
5. Дані про пацієнтів, у яких зазначений лікар на даний час є лікарем.
6. Дані про кількість проведених обстежень у лабораторії за вказаний період для цього медичного закладу.

Варіант №6**«ІНФОРМАЦІЙНА СИСТЕМА АВТОПІДПРИЄМСТВА МІСТА»**Опис предметної галузі.

Автопідприємство міста займається організацією пасажирських та вантажних перевезень усередині міста. Підприємство відповідно має в своєму розпорядженні автотранспорт різного призначення: легковий транспорт, вантажний і транспорт допоміжного характеру. Кожна категорія транспорту має властиві лише їй характеристики, вантажний транспорт характеризується вантажопідйомністю, пасажирський місткістю.

Підприємство має штат водіїв, закріплених за автомобілями, одним автомобілем може бути закріплено кілька водіїв. Обслуговуючий персонал займається технічним обслуговуванням автомобільної техніки. Обслуговуючий персонал та водії об'єднуються у бригади, якими керують бригадири, начальники дільниць та цехів.

Пасажирський автотранспорт перевозить пасажирів за певними маршрутами, за кожним маршрутом закріплено окремі одиниці автотранспорту. Ведеться облік числа пасажирів, що перевозяться, на підставі чого проводиться перерозподіл транспорту з одного маршруту на інший. Для вантажного транспорту обліковується обсяг вантажоперевезень, для транспорту допоміжного призначення – інтенсивність використання. По всьому автотранспорту враховується пробіг, кількість ремонтів та витрати на ремонт. Ведеться облік інтенсивності роботи бригад з ремонту (кількість ремонтів, обсяг виконаних робіт) по кожній автомашині, дільниці, цеху, підприємству.

Передбачити можливість одержання наступної інформації.

1. Перелік водіїв по підприємству, за вказаною машиною.
2. Дані про розподіл пасажирського автотранспорту маршрутами.
3. Відомості про кількість перевезених пасажирів конкретною автомашинною за вказаний день, місяць, рік.

4. Відомості про пробіг автотранспорту певної категорії чи конкретної автомашини за вказаний день, місяць, рік.

5. Дані про ремонти та їхню вартість для автотранспорту певної категорії, зазначеної автомашини за зазначений період.

6. Відомості про вантажоперевезення, виконані зазначеною автомашиною за зазначений період.

7. Отримати склад підлеглих вказаного бригадира, начальника дільниці, начальника цеху.

8. Отримати дані про роботи, виконані вказаним фахівцем за зазначений період загалом та за конкретною автомашиною.

Варіант №7

«ІНФОРМАЦІЙНА СИСТЕМА ПРОЕКТНОЇ ОРГАНІЗАЦІЇ»

Опис предметної галузі.

Проектна організація представлена такими категоріями співробітників: конструктори, інженери, техніки, лаборанти, обслуговуючий персонал. Кожна категорія може мати властиві лише їй атрибути. Конструктор характеризується числом авторських свідоцтв, техніки – обладнанням, яке можуть обслуговувати, інженер чи конструктор може керувати договором чи проектом. Співробітники розділені на відділи, керовані начальником, кожен співробітник працює лише у одному відділі.

В рамках укладених проектною організацією договорів із замовниками виконуються різноманітні проекти, причому за одним договором може виконуватися більше одного проекту, і один проект може виконуватися для декількох договорів. Сумарна вартість договору визначається вартістю всіх проектних робіт, які виконуються для цього договору. Кожен проект має керівника та групу співробітників, які виконують цей проект, причому це можуть бути співробітники не лише одного відділу.

Проекти виконуються з допомогою різного устаткування, частина якого приписано окремим відділам, а частина є колективною власністю проектною організацією. Для виконання проекту обладнання передається групі, яка працює над проектом, якщо це обладнання не використовується в іншому проекті.

Передбачити можливість одержання наступної інформації.

1. Дані про склад зазначеного відділу, за вказаною категорією співробітників, за віковим складом.
2. Перелік керівників відділів.
3. Перелік договорів або проектів, що виконуються на даний момент або у період зазначеного інтервалу часу.
4. Перелік проектів, що виконуються у межах зазначеного договору.
5. Дані про розподіл обладнання на даний час по відділах.
6. Відомості про використання обладнання зазначеними проектами.
7. Відомості про участь зазначеного працівника у проектах за певний період часу.
8. Дані про склад працівників за окремими категоріями, що беруть участь у зазначеному проекті.

Варіант №8

«ІНФОРМАЦІЙНА СИСТЕМА АВІАБУДІВЕЛЬНОГО ПІДПРИЄМСТВА»

Опис предметної галузі.

Підприємство випускає вироби: літаки (цивільні, транспортні, військові), планери, гелікоптери, дельтаплани, ракети (артилерійські, авіаційні, військово-морські). Кожна категорія виробів має специфічні, властиві лише їй атрибути. Для літаків це число двигунів, для ракети – потужність заряду тощо. За кожною категорією виробів може збиратися кілька видів виробів.

Структурно підприємство розбите на цехи, які у свою чергу поділяються на ділянки. Робітники об'єднуються у бригади, якими керують бригадири. Бригадири вибираються з-поміж робітників. Начальники дільниць та цехів призначаються з числа інженерно-технічного персоналу.

Кожен виріб збирається у своєму цеху (у цеху може збиратися кілька видів виробів) і в процесі виготовлення проходить певний цикл робіт, переміщаючись з однієї ділянки на іншу. Усі роботи зі збирання конкретного виробу певному ділянці виконує одна бригада робочих, у своїй дільниці може працювати кілька бригад.

Зібраний виріб проходить серію випробувань у випробувальних лабораторіях, які можуть обслуговувати декілька цехів, при цьому цехи можуть використовувати кілька випробувальних лабораторій.

Передбачити можливість одержання наступної інформації.

1. Перелік видів виробів окремої категорії, які збираються зазначеним цехом, усім підприємством.
2. Перелік виробів, зібраних зазначеним цехом, ділянкою, підприємством загалом за певний відрізок часу.
3. Отримати дані про кадровий склад цеху, підприємства.
4. Перелік ділянок зазначеного цеху та їх начальників.
5. Перелік робіт, що проходять вказаний виріб.
6. Склад бригад вказаної ділянки, цеху.
7. Склад бригад, які беруть участь у складанні зазначеного виробу.
8. Перелік випробувальних лабораторій, що у випробуваннях деякого конкретного виробу.
9. Перелік виробів окремої категорії, які проходили випробування у вказаній лабораторії за певний період.

Варіант №9**«ІНФОРМАЦІЙНА СИСТЕМА БУДІВЕЛЬНОЇ ОРГАНІЗАЦІЇ»**Опис предметної галузі.

Будівельна організація займається будівництвом різноманітних об'єктів (житлових будинків, лікарень, шкіл, мостів, доріг тощо) за договорами із замовниками. Кожна категорія об'єктів має характеристики, властиві тільки цій категорії: до характеристик житлових будинків відноситься поверховість, тип будівельного матеріалу, кількість квартир, для мостів унікальними характеристиками є тип прогонової будови, ширина, кількість смуг для руху.

Будівельна організація складається з будівельних управлінь, які ведуть роботи на одній або кількох ділянках будівництва. Інженерно-технічного персоналу (інженери, технологи, техніки) та робітникам (муляри, бетонники, оздоблювальні машини, зварювальники, електрики, слюсарі) властиві характерні лише цій групі атрибути. Робітники об'єднуються у бригади, якими керують бригадири. Бригадири вибираються з-поміж робітників. Майстри, виконроби, начальники дільниць та управлінь призначаються з числа інженерно-технічного персоналу.

На кожній ділянці зводиться один чи кілька об'єктів, кожному об'єкті роботу ведуть чи кілька бригад. Кожна бригада виконує певний вид робіт на об'єкті. Для організації робіт на об'єкті складаються графіки робіт, які вказують, у які терміни виконуються ті чи інші роботи. Закінчивши роботу, бригада переходить до іншого об'єкта на цій чи іншій ділянці.

Передбачити можливість одержання наступної інформації.

1. Перелік будівельних управлінь, дільниць та їх керівників.
2. Список фахівців інженерно-технічного складу зазначеної ділянки чи будівельного управління із зазначенням їх посад.
3. Перелік об'єктів, що зводяться зазначеним будівельним управлінням, ділянкою, та графіки їх зведення.
4. Склад бригад, які працюють на будівництві зазначеного об'єкта.

5. Перелік об'єктів, споруджуваних у певному будівельному управлінні, у яких у зазначений період виконувався зазначений вид будівельних робіт.

6. Перелік видів будівельних робіт, за якими мало місце перевищення строків виконання на зазначеній ділянці, будівельному управлінні або загалом з будівельної організації.

7. Перелік видів будівельних робіт, виконаних зазначеною бригадою протягом зазначеного періоду часу із зазначенням об'єктів, на яких ці роботи виконувались.

Варіант №10

«ІНФОРМАЦІЙНА СИСТЕМА БІБЛІОТЕЧНОГО ФОНДУ МІСТА»

Опис предметної галузі.

Бібліотечний фонд знаходиться у бібліотеках, що розташовані на території міста. Кожна бібліотека включає абонементи та читальні зали. Користувачами бібліотек є різні категорії читачів: студенти, науковці, викладачі, школярі, робітники, пенсіонери та інші мешканці міста. Кожна категорія читачів може мати особливі характеристики: для студентів це назва навчального закладу, факультет, курс, для наукового працівника – назва організації, наукова тема. Кожен читач, зареєстрований в одній із бібліотек, має доступ до всього бібліотечного фонду міста.

Бібліотечний фонд (книги, журнали, газети, збірки статей, збірки віршів, дисертації, реферати, збірники доповідей) розміщений у залах-сховищах різних бібліотек на певних місцях зберігання (номер залу, стелажу, полиці) та ідентифікується інвентарними номерами. При цьому існують різні правила щодо тих чи інших видань: якісь підлягають лише читанню у читальних залах бібліотек, для тих, що видаються, може бути встановлений різний термін видачі. З одного боку, бібліотечний фонд може поповнюватися, з іншого – з часом відбувається його списання.

Співробітники бібліотеки, які працюють у різних залах бібліотек, ведуть облік читачів, а також облік розміщення та видачі літератури.

Передбачити можливість одержання наступної інформації.

1. Список читачів із заданими характеристиками: список студентів зазначеного навчального закладу та факультету; список науковців, які цікавляться певною тематикою.
2. Перелік читачів, яким видано вказаний твір.
3. Список видань, які протягом деякого часу отримували зазначений читач із фонду бібліотеки, де його зареєстровано.
4. Список читачів із простроченим терміном повернення літератури.
5. Список літератури, яка надійшла або була списана протягом деякого періоду.
6. Список читачів, які не відвідували бібліотеку протягом зазначеного часу.
7. Список інвентарних номерів та назви видань, у яких міститься зазначений твір.
8. Список інвентарних номерів та назви видань, які містять твори зазначеного автора.
9. Список найпопулярніших творів.

Варіант №11

«ІНФОРМАЦІЙНА СИСТЕМА АВТОМОБІЛЕБУДІВЕЛЬНОГО ПІДПРИЄМСТВА»

Опис предметної галузі.

Структурно підприємство складається з цехів, які у свою чергу поділяються на ділянки. Вироби підприємства: вантажні, легкові автомобілі, автобуси, сільськогосподарські, дорожньобудівельні машини, мотоцикли та інші вироби. Кожна категорія виробів має специфічні, властиві лише їй

атрибути. Для автобусів це місткість, для сільськогосподарських та дорожньо-будівельних машин – продуктивність.

Кожній категорії інженерно-технічного персоналу (інженери, технологи, техніки) та робітників (збирачі, токарі, слюсарі, зварювальники та ін.) також характерні атрибути, властиві тільки для цієї групи. Робітники об'єднуються у бригади, якими керують бригадири. Бригадири вибираються з числа робітників, начальники дільниць та цехів призначаються з числа інженерно-технічного персоналу.

У цеху може збиратися кілька видів виробів. У процесі збирання проходить певний цикл робіт, переміщаючись з однієї ділянки на іншу. Усі роботи зі збирання конкретного виробу певному ділянці виконує одна бригада робочих, у своїй дільниці може працювати кілька бригад. Різні вироби можуть проходити одні й самі цикли робіт на тих самих ділянках цеху.

Зібраний виріб проходить серію випробувань у випробувальних лабораторіях. Випробувальні лабораторії можуть обслуговувати кілька цехів, своєю чергою цехи можуть користуватися кількома лабораторіями. Випробування проводяться на обладнанні випробувальної лабораторії, при цьому під час випробування конкретного виробу в лабораторії можуть бути задіяні різні види обладнання.

Передбачити можливість одержання наступної інформації.

1. Перелік виробів окремої категорії, зібраних зазначеним цехом, ділянкою, підприємством за певний час.
2. Дані про кадровий склад цеху, підприємства за вказаними категоріями інженерно-технічного персоналу та робітників.
3. Перелік ділянок зазначеного цеху, підприємства та їх начальників.
4. Перелік робіт, що проходять вказаний виріб.
5. Склад бригад вказаної ділянки, цеху.
6. Склад бригад, що беруть участь у складанні зазначеного виробу.
7. Перелік випробувальних лабораторій, що у випробуваннях деякого конкретного виробу.

8. Склад обладнання, що використовувалося під час випробування зазначеного виробу, виробів окремої категорії у зазначеній лабораторії.

Варіант №12

«ІНФОРМАЦІЙНА СИСТЕМА ГОТЕЛЬНОГО КОМПЛЕКСУ»

Опис предметної галузі.

Готельний комплекс складається з кількох будівель-готелів (корпусів). Кожен корпус має ряд характеристик, таких, як клас готелю, кількість поверхів у будівлі, загальна кількість кімнат, кімнат на поверсі, кількість місць у номерах, наявність служб побуту: пральня, хімчистка, харчування (ресторани, бари) та розваги (басейн, сауна), більярд та ін.). Від типу корпусу та кількості місць у номері залежить сума сплати за нього. Послуги служб побуту провадяться за окрему плату.

З великими організаціями (туристичні фірми, організації, що займаються проведенням міжнародних симпозіумів, конгресів, семінарів, карнавалів) укладаються договори, що дозволяють організаціям заздалегідь бронювати номери груп людей з великими знижками. Кожна організація має характеристики, властиві тільки їй. У броні вказується клас готелю, поверх, кількість кімнат та загальна кількість людей. Броня може бути скасована за тиждень до заселення.

Нові мешканці поповнюють перелік клієнтів готелю. Ведеться облік вільних номерів, витрат постояльців готелю та облік витрат та доходів готельного комплексу. Досліджується думка мешканців про ціни та сервіс. Скарги фіксуються та досліджуються. Вивчається статистика популярності номерів.

Передбачити можливість одержання наступної інформації.

1. Перелік фірм, які забронювали місця за період співробітництва, або за певний зазначений період.

2. Перелік постояльців, які заселялися в номери із зазначеними характеристиками за певний період.
3. Відомості про вільні номери із зазначеними характеристиками.
4. Відомості про конкретний номер: протягом якого часу він буде вільний і про його характеристики.
5. Список зайнятих номерів, що звільняються до зазначеного терміну.
6. Дані бронювання номерів цією фірмою за вказаний період.
7. Список незадоволених клієнтів та їх скарги.
8. Відомості про постояльца із заданого номера: його рахунок за додаткові послуги, види наданих додаткових послуг.
9. Відомості про фірми, з якими укладено договори про броню на зазначений період.
10. Відомості про конкретну людину, скільки разів він відвідував готель, у яких номерах та в який період зупинявся, які рахунки оплачував.
11. Відомості про конкретний номер: ким він був зайнятий у певний період.

Варіант №13

«ІНФОРМАЦІЙНА СИСТЕМА МАГАЗИНА АВТОЗАПЧАСТИН»

Опис предметної області.

Магазин роздрібною торгівлі здійснює замовлення запчастин у різних постачальників. Постачальники бувають різних категорій: великі фірми, що безпосередньо виробляють деталі, невеликі виробництва, дилери, дрібні постачальники та магазини. Кожна категорія має набір атрибутів, що відрізняється. Фірми та дилери – найнадійніші партнери, у них закупається великий обсяг продукції, вони можуть запропонувати повний пакет документів, знижки та гарантію якості. Невелике виробництво – це низькі ціни, але гарантії якості. У дрібних лавках можна вигідно купити невелику кількість простих деталей, на яких відразу видно шлюб. Фірми та дилери

поставляють деталі на основі договорів, чого не робиться для невеликого виробництва та дрібного магазину.

Ведеться статистика продажів, що відбиває попит ті чи інші деталі, і основі цього складаються замовлення на необхідні товари. Вибір постачальника на кожне конкретне замовлення здійснюють менеджери магазину. У замовленнях перераховується найменування товару, кількість.

Після отримання товару він доставляється на склад магазину. Деталі зберігаються на складі у певних осередках. Усі осередки пронумеровані. Каса займається прийомом грошей від покупців за товар, а також здійснює повернення грошей за шлюб. Шлюб, якщо можливо, повертається постачальнику, який здійснює заміну бракованої деталі. Інформація про шлюб (постачальник, фірма-виробник, деталь) фіксується.

Передбачити можливість одержання наступної інформації.

1. Перелік постачальників певної категорії, які постачають зазначений вид товару у заданому обсязі за певний період.
2. Відомості про конкретний вид деталей: якими постачальниками поставляється за якою ціною час поставки.
3. Перелік покупців, які купили зазначений вид товару за певний період, або зробили купівлю товару обсягом, щонайменше зазначеного.
4. Перелік, обсяг та номер осередку для всіх деталей, що зберігаються на складі.
5. Середня кількість продажів за місяць за будь-яким видом деталей.
6. Перелік та загальна кількість непроданого товару на складі за певний період.
7. Перелік бракованого товару, що прийшов за певний період, та перелік постачальників, які поставили бракований товар.

Варіант №14**«ІНФОРМАЦІЙНА СИСТЕМА ПРЕДСТАВНИЦТВА ТУРИСТИЧНОЇ ФІРМИ»**Опис предметної сфери.

Туристична фірма в Україні формує групу туристів для відвідин зарубіжної країни. При цьому на кожного туриста заповнюється пакет документів для отримання візи у відділі еміграції (ПІБ, паспортні дані, стать, вік, діти, в якому готелі хочуть жити). Туристична група ділиться кілька категорій. Туристи, які їдуть відпочити (вони більше цікавляться екскурсіями та не цікавляться товаром) та туристи, які їдуть за вантажем (вони цікавляться товаром та не цікавляться екскурсіями). Окрему категорію складають діти. Діти не можуть отримати візу, самі переселитися, ходити кудись без супроводу батьків.

Представництво туристичної фірми відкриває візи для туристів, займається прийомом туристів в аеропорту, вирішує проблеми, пов'язані з митницею, бронює місця та розселяє групу в готелях, організовує харчування туристів. Представництво пропонує розклад екскурсій та здійснює запис бажаючих на певні екскурсії. Складений список передається до агенції організації екскурсій.

Представництво готує повний фінансовий звіт по кожному туристу та по всій групі та подає його до туристичної фірми. Усі статті витрати: готель, перевезення, екскурсії, непередбачені витрати, розрахунки в аеропорту, заносяться до фінансового звіту.

Передбачити можливість одержання наступної інформації.

1. Список туристів за зазначеною категорією для митного контролю.
2. Списки на розселення готелями зазначеної категорії туристів.
3. Список туристів, які побували в країні за певний період загалом та за певною категорією туристів.

4. Відомості про конкретного туриста: скільки разів був у країні, дати прильоту/відльоту, в яких готелях зупинявся, які екскурсії та в яких агентствах замовляв.

5. Список готелів, в яких провадиться розселення туристів, із зазначенням кількості займаних номерів та осіб, що в них проживали, за певний період.

6. Список туристів, які замовили екскурсії за певний період.

7. Повний фінансовий звіт щодо зазначеної групи та для певної категорії туристів.

Варіант №15

«ІНФОРМАЦІЙНА СИСТЕМА АПТЕКИ»

Опис предметної галузі.

Аптека продає готові медикаменти і спеціалізується на приготуванні ліків за рецептами. Це можуть бути: мікстури, мазі, розчини, настойки, порошки, які відрізняються один від одного способом застосування (для зовнішнього, внутрішнього застосування та змішування з іншими ліками). Ліки відрізняються також способом та часом приготування. Існує довідник технологій приготування різноманітних ліків. У ньому вказуються: ідентифікаційний номер технології, назва ліків та спосіб приготування.

Готові ліки можуть відпускатися без рецепта, а деякі з них лише за рецептом лікаря. Для виготовлення аптекою ліки, покупець повинен обов'язково принести рецепт від лікаря. Реєстратура аптеки приймає замовлення. Якщо всі компоненти, що входять до ліків, є, то вони резервуються для приготування зазначених ліків. Покупець оплачує замовлення і йому повідомляють дату його отримання. Відомості про замовлення поповнюють довідник замовлень виробництва ліків. Відомості про отримання готових ліків поповнюють список виконаних замовлень.

Усі готові ліки та складові для приготування ліків зберігаються на складі. Через певний проміжок часу провадиться інвентаризація складу. Це робиться для того, щоб визначити, чи є ліки з критичною нормою, чи ліки, у яких вийшов термін зберігання, чи ліки, що бракують. За результатами інвентаризації складаються заявки на доставку необхідних медикаментів із оптових складів.

Передбачити можливість одержання наступної інформації.

1. Відомості про покупців, які не прийшли забрати своє замовлення у призначений їм час.
2. Перелік найчастіше використовуваних медикаментів.
3. Перелік покупців, які замовили виготовлення певних ліків або певних типів ліків за цей період.
4. Перелік ліків, що досягли своєї критичної норми або закінчилися.
5. Перелік замовлень, що у виробництві.
6. Технології приготування ліків зазначених типів, конкретних ліків, ліків, що у довіднику замовлень на виробництві.
7. Відомості про ціни на зазначені ліки в готовому вигляді, про обсяг та ціни на всі компоненти, потрібні для цього ліки.
8. Отримати відомості про конкретні ліки.

Варіант №16

«ІНФОРМАЦІЙНА СИСТЕМА МІСЬКОЇ ТЕЛЕФОННОЇ МЕРЕЖІ»

Опис предметної області.

Міська телефонна мережа є розгалуженою мережею локальних АТС. Кожна АТС має свої абоненти. В абонента може стояти телефон одного із трьох типів: основний, паралельний чи блокторний. За кожним абонентом (у нього є прізвище, ім'я, по батькові, стать, вік) закріплений свій номер телефону, причому у кількох абонентів може бути один і той же номер (при паралельному або блокторному підключенні). Кожному номеру телефону

відповідає адреса (індекс, район, вулиця, будинок, квартира), причому паралельні або блоаторні телефони обов'язково мають бути в одному будинку.

Всі телефони міської АТС мають вихід на міжмісто, але для конкретного абонента він може бути або відкритий, або закритий з будь-якої причини (відключений за бажанням абонента, за несплату тощо). Відомості про міжміські переговори збираються та аналізуються на ГТС.

Абоненти зобов'язані сплачувати абонентську плату. Плата повинна вноситись щомісяця до 20 числа. При несплаті після письмового повідомлення протягом п'яти діб вимикається телефон. При заборгованості за міжміські розмови та несплаті після письмового повідомлення здійснюється відключення лише можливості виходу на міжмісто. Включення того чи іншого провадиться при оплаті вартості абонентської плати та включення.

Абонентів будь-якої АТС можна поділити на простих та пільгових. До категорії пільговиків належать пенсіонери, інваліди тощо. Пільговики сплачують лише 50% абонентської плати. Відповідно до всього цього (тип телефону, пільговик чи ні, чи є вихід на міжмісто) розраховується розмір абонентської плати.

Передбачити можливість одержання наступної інформації.

1. Перелік абонентів зазначеної АТС повністю, лише пільговиків, за віковою ознакою, за групою прізвищ.
2. Перелік боржників на зазначеній АТС, абонентів, які мають заборгованість уже більше тижня (місяця), за ознакою заборгованості за міжмісто та (або) за абонентською платою, за розміром боргу.
3. Перелік абонентів зазначеної АТС, які мають паралельні телефони, лише пільговиків, які мають паралельні телефони.
4. Повну інформацію про абонентів із заданим телефонним номером.
5. Перелік та загальна кількість боржників на зазначеній АТС, яким слід надіслати письмове повідомлення, відключити телефон або вихід на міжмісто.

Варіант №17**«ІНФОРМАЦІЙНА СИСТЕМА ТЕАТРУ»**Опис предметної галузі.

Працівників театру можна поділити на акторів, музикантів, постановників та службовців. Кожна з наведених категорій має унікальні характеристики. Актори театру можуть мати звання заслужених та народних артистів, можуть бути лауреатами конкурсів. Акторами театру можуть бути студенти театральних училищ. Кожен актор має свої вокальні та зовнішні дані (стаття, вік, голос, зростання), які можуть підходити для якихось ролей. Театр очолює директор, який контролює постановку та показ вистав, затверджує репертуар, приймає на роботу нових службовців, запрошує акторів, постановників.

Для постановки будь-якої вистави необхідно підібрати акторів на ролі та дублерів на кожну головну роль. Один і той же актор не може грати більше однієї ролі у виставі, але може грати у різних спектаклях. У вистави також є режисер-постановник, художник-постановник, диригент-постановник, автор. Вистави можна розділити за жанрами: музична комедія, трагедія, оперета. З іншого боку, вистави можна поділити на дитячі, молодіжні та ін.

У репертуарі театру вказується, які спектаклі, у які дні і коли відбуватимуться, а також дати прем'єр. У касах театру можна наперед придбати квитки на будь-які спектаклі. Ціна квитків залежить від місця, та спектаклю. На прем'єри квитки дорожчі. Адміністрацією театру фіксується кількість проданих квитків на кожну виставу.

Передбачити можливість одержання наступної інформації.

1. Список усіх працівників театру, акторів, музикантів, за стажем роботи в театрі, за роком народження.
2. Перелік вистав, зазначених у репертуарі на цей сезон, вже зіграних вистав, вистав зазначеного жанру.
3. Перелік всіх поставлених вистав, вистав зазначеного жанру, будь-коли поставлених у театрі, поставлених за вказаний період.

4. Список авторів поставлених вистав, авторів, що жили у вказаному столітті, авторів вистав зазначеного жанру, поставлених у театрі.

5. Список акторів, які підходять за своїми даними на зазначену роль.

6. Список акторів театру, які мають звання, які отримали їх за певний період, на вказаних конкурсах за статевою ознакою за віком.

7. Для зазначеної вистави список акторів, дублерів, імена режисера-постановника, художника-постановника, диригента-постановника, авторів, дату прем'єри.

8. Перелік ролей, зіграних зазначеним актором всього, за певний період часу, у спектаклях певного жанру, у спектаклях зазначеного режисера-постановника, у дитячих спектаклях.

Варіант №18

«ІНФОРМАЦІЙНА СИСТЕМА АЕРОПОРТУ»

Опис предметної галузі.

Працівників аеропорту можна поділити на пілотів, диспетчерів, техніків, касирів, працівників служби безпеки, довідкової служби та інших, які адміністративно ставляться кожен до свого відділу. Кожна категорія працівників має унікальні атрибути-характеристики, які визначаються професійною спрямованістю. У відділах існує розбиття працівників на бригади. Відділи очолюються начальниками, які є адміністрацією аеропорту.

У функції адміністрації входить формування кадрового складу аеропорту, складання розкладу польотів, підготовка та проведення рейсів. За кожним літаком закріплюється бригада пілотів, техніків та обслуговуючого персоналу. Літак повинен своєчасно оглядатись техніками та при необхідності ремонтуватися. Підготовка до рейсу складається з технічної частини (техогляд, заправка необхідної кількості палива) та обслуговуючої частини (прибирання салону, запас продуктів харчування).

У розкладі вказується тип літака, номер рейсу, дні вильоту, час вильоту та прильоту, маршрут (початковий та кінцевий пункти призначення, пункт

пересадки), вартість квитка. Квитки на авіарейси можна придбати в день вильоту, заздалегідь або забронювати в авіакасах. Ціна квитка залежить не тільки від маршруту та типу літака, а й від часу вильоту (вночі або рано вранці – ціна квитка нижче). За потреби квиток можна повернути до відправлення рейсу. Авіарейси можуть бути затримані через погодні умови, технічні неполадки.

Пасажир при посадці в літак повинен пред'явити квиток, паспорт, а для міжнародного рейсу також зобов'язаний пред'явити закордонний паспорт і пройти митний огляд. Пасажири можуть здавати свої речі у багажне відділення.

Передбачити можливість одержання наступної інформації.

1. Список працівників аеропорту, начальників відділів, працівників зазначеного відділу, за стажем роботи, за розміром заробітної плати.
2. Перелік працівників бригад, які обслуговують конкретний рейс.
3. Перелік літаків, приписаних до аеропорту, що знаходяться в ньому у зазначений час, за кількістю скоєних рейсів.
4. Перелік літаків, що пройшли техогляд, відправлені в ремонт, за кількістю скоєних рейсів без ремонту.
5. Перелік рейсів за вказаним маршрутом, за тривалістю перельоту. Перелік затриманих рейсів повністю за вказаним маршрутом та кількість зданих квитків за час затримки.

Перелік вільних та заброньованих місць на вказаному рейсі, на визначений день за вказаним маршрутом за часом вильоту.

Варіант №19

«ІНФОРМАЦІЙНА СИСТЕМА ЗООПАРКУ»

Опис предметної галузі.

Службовців зоопарку можна поділити на кілька категорій: ветеринари, прибиральники, дресирувальники, будівельники-ремонтники, працівники адміністрації. Кожна з перерахованих категорій працівників має унікальні

атрибути-характеристики, які визначаються професійною спрямованістю. Кожна тварина доглядає певне коло службовців, причому тільки ветеринарам, прибиральникам і дресирувальникам дозволено доступ до клітин до тварин.

При розселенні тварин у зоопарку по клітинах необхідно враховувати як потреби цього виду залежно від зони проживання, а й їх сумісність із тваринами у сусідніх клітинах.

Ветеринари повинні проводити медогляди, стежити за вагою, зростанням, розвитком тварини, ставити своєчасно щеплення та заносити всі ці дані до картки, яка заводиться на кожну особину за її появи в зоопарку. Хворим тваринам призначається лікування і за необхідності їх можна ізолювати у стаціонарі.

Для годування тварин необхідні різні типи кормів: рослинний, живий, м'ясо та різні комбікорми. Рослинний корм – це фрукти та овочі, зерно та сіно. Живий корм – миші, птахи, корми для риб. Для кожного виду тварин розраховується свій раціон, який у свою чергу варіюється в залежності від віку, фізичного стану тварини та сезону. Таким чином, у кожної тварини в зоопарку є меню на кожен день, в якому вказується кількість та час годівлі на день, кількість та вид їжі. Зоопарк має постачальників кормів для тварин. Кожен постачальник спеціалізується на якихось конкретних видах кормів. Частина кормів зоопарк може виробляти сама: запасати сіно, розводити мишей тощо.

Передбачити можливість одержання наступної інформації.

1. Список службовців зоопарку, або службовців цієї категорії, за стажем роботи у зоопарку, віком, розміром заробітної плати.
2. Список службовців зоопарку, відповідальних за зазначений вид тварин або за конкретну особину.
3. Список всіх тварин у зоопарку чи тварин зазначеного виду.
4. Перелік тварин, що потребують теплого приміщення на зиму.
5. Перелік тварин, яким зроблено зазначене щеплення, або перехворіли на деяку хворобу.

Перелік постачальників кормів повністю, або поставляють лише певний корм, за кількістю корму, що поставляється, ціною.

Перелік кормів, що виробляються зоопарком повністю, або лише тих кормів, постачання яких зоопарк не потребує.

Варіант №20

«ІНФОРМАЦІЙНА СИСТЕМА ГІБДД»

Опис предметної галузі.

Однією з функцій ДІБДР є реєстрація автотранспортних засобів. До автотранспортних засобів належать: легкові, вантажні автомобілі, причепи, мотоцикли, автобуси, мікроавтобуси. ДІБДР займається виділенням та обліком номерних знаків на автотранспорт. На різні види транспорту видаються різні види номерів і до бази даних заносяться різні показники. Номери можуть виділятися як приватним власникам, так і організаціям. У довіднику номерів, виданих приватним власникам, фіксується: номер, ПІБ власника, його адреса, марка автомобіля, дата випуску, об'єм двигуна, номери двигуна, шасі та кузова, колір. У довіднику номерів, виданих організації, додатково фіксується назва організації, район, адреса, керівник. Існує довідник вільних номерів (серія, діапазон номерів).

ДІБДР періодично проводить технічний огляд машин. Періодичність проходження залежить від року випуску та виду транспортного засобу. Технічні характеристики, що перевіряються, залежать від виду транспортного засобу. Для проходження техогляду потрібна квитанція про сплату податків, сума оплати залежить від обсягу двигуна.

ДІБДР займається обліком та аналізом ДТП (дорожньо-транспортна пригода). Під час реєстрації ДТП фіксується: дата, тип події, місце події, марки потерпілих автомобілів, державний номер, тип машини, короткий зміст, кількість потерпілих. Викрадення або зникнення винуватця ДТП з місця події потребує оперативного втручання всіх постів ДІБДР та патрульних машин.

Інформація про машину, що розшукується, витягується з бази зареєстрованих номерів і передається по рації всім постам.

Передбачити можливість одержання наступної інформації.

1. Відомості про власника автотранспортного засобу за державним номером автомашини.
2. Відомості про автомобіль за державним номером, чи брав участь у ДТП, чи пройшов техогляд.
3. Перелік власників машин, які не пройшли техогляд.
4. Результати аналізу ДТП: найнебезпечніші місця у місті, найчастіша причина ДТП.
5. Список машин, оголошених у розшуку, чи то ті, що зникли з місця ДТП, чи викрадені.

Дані щодо ефективності розшукової роботи: кількість знайдених машин у відсотковому відношенні.

Статистику по угонах: марки машин, що найчастіше викрадаються, найнадійніші сигналізації.

Варіант №21

«ІНФОРМАЦІЙНА СИСТЕМА ФОТОЦЕНТРУ»

Опис предметної галузі.

Фотоцентр має головний офіс та мережу філій та кіосків прийому замовлень, розташованих за певними адресами. У кіосках здійснюється лише прийом замовлень, тому кожен кіоск прикріплений до певної філії, в якій ці замовлення виконуються. У філіях є необхідне обладнання для проявлення плівок та друку фотографій. Філії та кіоски приймають замовлення на проявлення плівок, друк фотографій. У замовленні на друк вказується кількість фотографій кожного кадру, загальна кількість фотографій, формат, тип паперу та термін виконання замовлення. Під час замовлення великої кількості фотографій надаються знижки. Термінові замовлення приймаються лише у філіях, і вони мають ціну вдвічі більшу, ніж звичайне замовлення.

Плівка, придбана в тій самій філії, куди вона принесена на прояв, проявляється безкоштовно.

Фотомагазини та кіоски пропонують до продажу різні фототовари: фотоплівки, фотоапарати, альбоми та інше приладдя. Фотомагазини також пропонують додаткові послуги: фотографії на документи, реставрація фотографій, прокат фотоапаратів, художнє фото, надання послуг професійного фотографа.

Відомості про виконані замовлення та продаж різних фототоварів збираються та обробляються, і на їх основі робиться загальне замовлення на постачання витратних матеріалів (фотопапір, фотоплівка, хімічні реактиви), фототоварів та обладнання. Отримані товари та матеріали розподіляються відповідно до запитів по кіосках та магазинах. Фотоцентр може мати кілька постачальників, які спеціалізуються на різних поставках, або на поставках фототоварів різних фірм.

Передбачити можливість одержання наступної інформації.

1. Перелік та загальна кількість пунктів прийому замовлень на фотороботи по філіях, кіоскам прийому замовлень, фотоцентру.

2. Кількість надрукованих фотографій у рамках простих та термінових замовлень за вказаною філією, кіоском прийому замовлень, фотоцентром за певний період часу.

3. Кількість виявлених фотоплівок у рамках простих та термінових замовлень за вказаною філією, кіоском прийому замовлень, фотоцентром за певний період часу.

4. Перелік постачальників фотоцентром, постачальників окремих видів фототоварів, які зробили поставки в певний період.

5. Перелік реалізованих фототоварів та обсяги їх реалізації за фотоцентром, за вказаною філією, проданими протягом деякого періоду часу.

Варіант №22**«ІНФОРМАЦІЙНА СИСТЕМА ЗАЛІЗНИЧНОЇ ПАСАЖИРСЬКОЇ СТАНЦІЇ»**Опис предметної галузі.

Працівників залізничної станції можна поділити на водіїв рухомого складу, диспетчерів, ремонтників рухомого складу, ремонтників колій, касирів, довідкової служби та інших, які адміністративно ставляться кожен до свого відділу. Кожна з перерахованих категорій працівників має унікальні атрибути характеристики, що визначаються професійною спрямованістю.

У відділах існує розбиття працівників на бригади. Відділи очолюються начальниками, які є адміністрацією залізничної станції. До функцій адміністрації входить планування маршрутів, складання розкладів, формування кадрового складу залізничної станції.

За кожним локомотивом закріплюється локомотивна бригада. За кількома локомотивами закріплюється бригада техніків-ремонтників, що виконує рейсовий та плановий техогляд (за певним графіком), ремонт, технічне обслуговування. Підготовка до рейсу включає технічну частину (рейсовий техогляд, дрібний ремонт) і обслуговувальну частину (прибирання вагонів, запас продуктів харчування тощо).

У розкладі вказується тип поїзда (швидкий, пасажирський тощо), номер поїзда, дні та час відправлення та прибуття, маршрут (початковий та кінцевий пункти призначення, основні вузлові станції), вартість квитка. Квитки на поїзд можна придбати заздалегідь або забронювати у залізничних касах. До відправлення поїзда, якщо є потреба, квиток можна повернути.

Передбачити можливість одержання наступної інформації.

1. Перелік та загальна кількість усіх працівників залізничної станції, начальників відділів, працівників зазначеного відділу, за стажем роботи на станції, статевою ознакою, віком.

2. Перелік та загальна кількість локомотивів, приписаних до залізничної станції, що знаходяться на ній у вказаний час, за часом прибуття на станції, за кількістю скоєних маршрутів.

3. Перелік та загальна кількість локомотивів, що пройшли плановий техогляд за певний період часу, відремонтованих вказану кількість разів, за кількістю скоєних рейсів до ремонту, за віком локомотива.

4. Перелік та кількість проданих квитків за вказаний час на певні маршрути за тривалістю маршруту за ціною квитка.

5. Отримати перелік та загальну кількість маршрутів зазначеної категорії, що прямують у певному напрямку.

Варіант №23

«ІНФОРМАЦІЙНА СИСТЕМА МІСЬКОЇ ФІЛАРМОНІЇ»

Опис предметної галузі.

Інфраструктура міської філармонії представлена культурними спорудами різного типу: театри, концертні майданчики, естради, палаци культури тощо. Кожна з категорій культурних споруд має атрибути, специфічні тільки для неї: театр характеризується місткістю, кінотеатр – розміром екрану.

Артисти під керівництвом імпресаріо виступають у різних жанрах, при цьому той самий артист може виступати в кількох жанрах, і може працювати з кількома імпресаріо.

Організатори концертних заходів проводять виступи, концерти, конкурси у культурних спорудах міста, організовуючи участь у ньому артистів. За результатами участі артистів у конкурсах проводиться нагородження.

Передбачити можливість одержання наступної інформації.

1. Перелік культурних споруд зазначеного типу в цілому або задовольняє заданим характеристикам (наприклад, зали, що вміщують не менше зазначеної кількості глядачів).
2. Список артистів, які у деякому жанрі.
3. Список артистів, які працюють з деяким імпресарію.
4. Список артистів, які виступають більш ніж в одному жанрі за їх вказівкою.
5. Список імпресарію вказаного артиста.
6. Перелік концертних заходів, проведених протягом заданого періоду часу загалом чи зазначеним організатором.
7. Список призерів вказаного конкурсу.
8. Перелік концертних заходів, проведених у зазначеній культурній споруді.
9. Список імпресарію певного жанру.
10. Список артистів, які не брали участь у жодних конкурсах протягом певного періоду часу.
11. Список організаторів культурних заходів та кількість проведених ними концертів протягом певного періоду часу.
12. Перелік культурних споруд, а також дати проведення на них культурних заходів протягом певного періоду часу.

Варіант №24

«ІНФОРМАЦІЙНА СИСТЕМА ПУНКТУ ПРОКАТУ ВІДЕОКАСЕТ»

Опис предметної області.

Пункт прокату містить каталог касет, які є в даний момент часу. Відеофільми, записані на касетах, можуть належати до художніх фільмів, анімаційних, науково-популярних, документальних. Кожна з категорій відеофільмів може мати свій набір характеристик.

Інформаційна система повинна підтримувати роботу каталогу, дозволяючи службовцям прокату додавати нові найменування касет, видаляти старі та редагувати дані про касети.

Клієнт, який звернувся до пункту, обирає касету за каталогом, вносить заставу та забирає її на певний термін. Термін прокату, що вимірюється на добу, обумовлюється при видачі касети. Вартість прокату обчислюється виходячи з тарифу за добу та строку прокату. Клієнт повертає касету та оплачує прокат. Якщо касета не пошкоджена, клієнт повертає заставу. Службовець пункту прокату реєструє здачу касети клієнту та її повернення. Якщо клієнт пошкодив касету, касета видаляється з каталогу, а заставка залишається у касі прокату.

Постійним клієнтам надаються знижки, а також приймаються заявки на поповнення асортименту касет. Заявки реєструються у системі. За ними готується підсумковий звіт, керуючись яким службовці пункту прокату оновлюють асортимент касет.

Передбачити можливість одержання наступної інформації.

1. Відомості про наявність відеокасети з назвою.
2. Список відеокасет певного жанру, певного режисера, знятих певною кіностудією, що є в даний час у наявності або взятих у прокат.
3. Відомості про те, коли буде повернена будь-яка касета з тих, що взяті у прокат.
4. Відомості про постійних клієнтів пункту прокату (чи користувався прокатом 5 або більше разів) та розміри знижок, якими користується кожен клієнт.

ЛАБОРАТОРНА РОБОТА №5 «СТВОРЕННЯ МОДЕЛІ ДАНИХ В СЕРЕДОВИЩІ ERwin»

Мета роботи:

- знайомство з інструментальним середовищем програми ERWin.
- створення логічної моделі даних, визначення сутностей, завдання атрибутів, вибір ключових атрибутів, встановлення зв'язків між сутностями.

Завдання для лабораторної роботи: Виконати проектування інформаційної системи за допомогою логічної моделі даних "Сутність-Зв'язок" засобами програми ERWin.

Контрольні питання:

1. Для чого застосовується програма ERWin?
2. Як додати сутність на діаграму?
3. Як зв'язати між собою сутності?

Звіт про виконання лабораторної роботи:

1. Тема лабораторної роботи.
2. Відповіді на запитання.
3. Результати виконання завдання в електронному вигляді.

Теоретична підготовка до виконання лабораторної роботи:

ERwin – засіб для проектування ІС та розробки структури бази даних (БД). ERwin поєднує графічний інтерфейс Windows, інструменти для побудови ER-діаграм, редактори для створення логічного та фізичного опису моделі даних і прозору підтримку провідних реляційних СУБД і настільних баз даних. За допомогою ERwin можна створювати або проводити зворотне проектування (реінжиніринг) баз даних. Цей програмний продукт – лідер

серед засобів моделювання баз даних і сховищ даних. Дозволяє проектувати, документувати і супроводжувати бази даних різних типів. Підтримуючи пряме і зворотне проектування для 20 типів СУБД, ERwin підвищує якість розроблюваної БД, продуктивність праці і швидкість розробки.

Процес моделювання в ERwin базується на методології проектування реляційних баз даних IDEF1X. Методологія IDEF1X визначає стандарти термінології, використовуваної при інформаційному моделюванні і графічного зображення типових елементів на ER-діаграмах.

ERwin має два рівня уявлення моделі – логічний і фізичний.

Логічний рівень – це абстрактний погляд на дані, на ньому дані представляються так, як виглядають в реальному світі. Об'єкти моделі, представлені на логічному рівні, називаються сутностями і атрибутами. Логічна модель даних може бути побудована на основі іншої логічної моделі, наприклад на основі моделі процесів (BPwin).

Фізичний рівень являє фізичну модель даних, яка залежить від конкретної СУБД, фактично будучи відображенням системного каталогу.

Отже, одній і тій ж логічній моделі можуть відповідати кілька різних фізичних моделей. Якщо в логічній моделі не має значення, який саме тип даних має атрибут, то у фізичній моделі важливо описати всю інформацію про конкретних фізичних об'єктах – таблицях, колонках, індексах, процедурах тощо.

Література:[1], [3], [5]

Практична підготовка до виконання лабораторної роботи:

5.1 Вивчення інструментального середовища ERWin

Запуск середовища СА ERWin Data Modeler

При запуску ERwin відкривається головне вікно програми СА ERwin Data Modeler (рис.5.1)

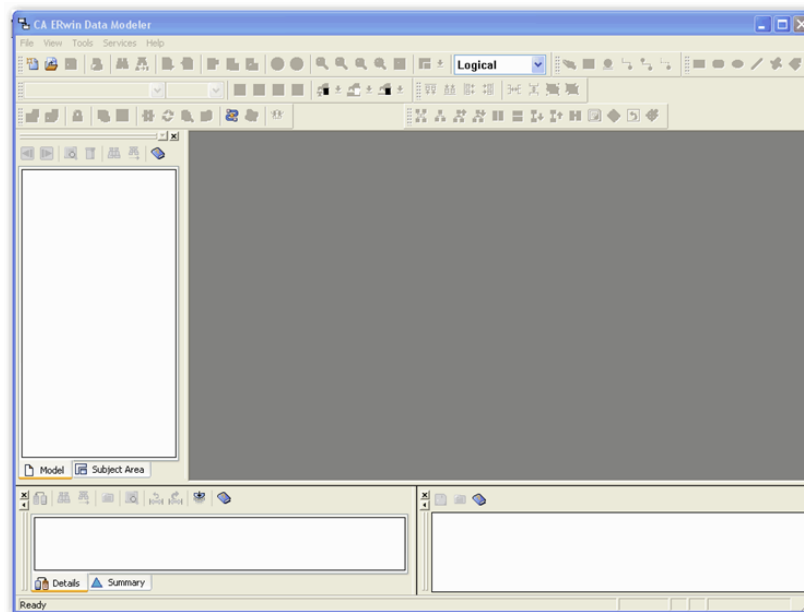


Рисунок 5.1 – Головне вікно програми СА ERwin Data Modeler

Порядок побудови моделі даних в середовищі ERwin розглянемо на прикладі інформаційної системи «Навчання студента у ВНЗ», призначеної для автоматизації аналізу та обліку даних про студентів, що навчаються на факультеті, і їх успішності в поточній сесії.

ERwin має два рівня уявлення моделі – логічний і фізичний.

Створення нової моделі даних починається з вибору типу моделі: *логічна, фізична, логіко-фізична*. При проектуванні бази даних доцільніше вибирати логіко-фізичну модель.

Почнемо моделювання даних зі створення логічної моделі. Після опису логічної моделі, проектувальник може вибрати необхідну СУБД і ERwin автоматично створить відповідну фізичну модель.

Вибір типу нової моделі здійснюється в діалозі *Create Model* (рис. 5.2), що викликається через *File > New* або кнопку створення нового файлу.

У вікні вибору моделей вибрати тип нової моделі *New model*: логічна, фізична або модель, яка допускає перетворення однієї в іншу. Оскільки потрібно спроектувати БД, слід вибрати логіко-фізичну модель (рис. 5.2).

При виборі фізичної або логіко-фізичної моделі в діалозі *Create Model* необхідно також вказати необхідну СУБД і номер її версії (в полях *Database i Version*).

Даний діалог дозволяє також відкрити існуючу модель даних зазначеного типу за допомогою активізації кнопки.

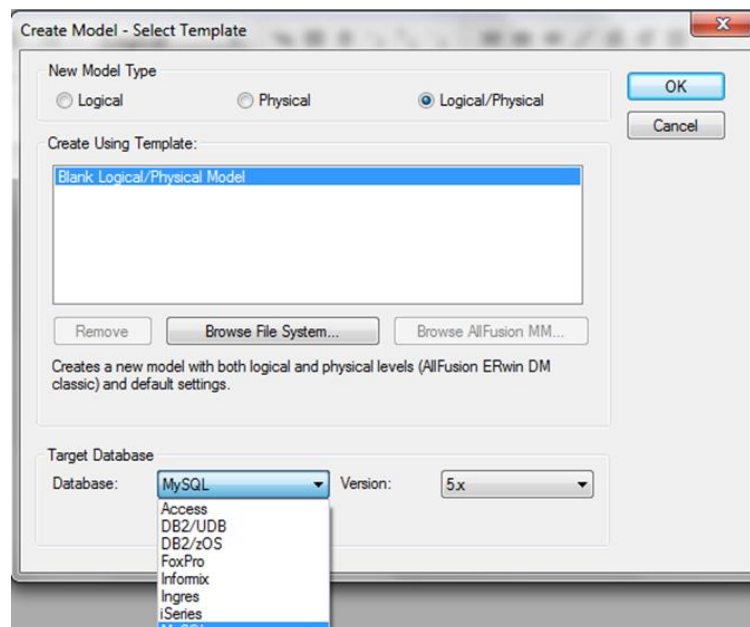


Рисунок 5.2 – Вікно вибору моделі

Відповідно до вибраної дії при активізації кнопки ОК відкривається вікно для побудови нової моделі даних (рис. 6.3) або вікно редагування моделі, створеної раніше.

При запуску програми ERWin і виборі відповідної моделі ми отримуємо доступ до інтегрованої середовищі розробки (Рис.6.4), яка включає:

- основну панель інструментів;
- палітру інструментів, склад якої залежить від обраного стандарту представлення моделей і типу моделі: логічного або фізичного;
- браузер (провідник) моделі
- робочу область для побудови моделі.

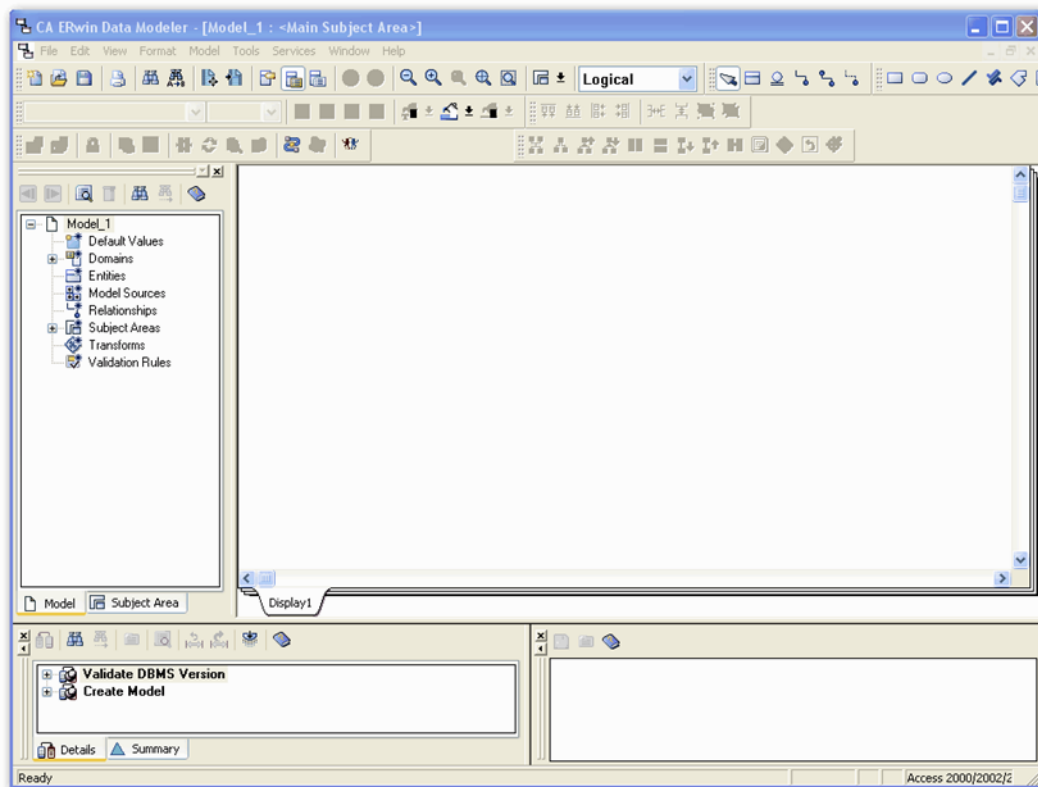









Рисунок 5.3 – Вікно побудови нової моделі даних

Для побудови ER-моделей використовується інструментальна панель, склад якої залежить від обраного стандарту представлення моделей і типу моделі: логічна або фізична.

У даній роботі рекомендується використовувати стандарт IDEF1X (Integration DEFinition for information modeling), в якому використовуються представлені позначення (табл. 5.1).

Таблиця 5.1 – Палітра інструментів логічного рівня

Умовне позначення	Призначення кнопок в логічній моделі
	Створення нової сутності – для створення сутності необхідно клацнути лівою кнопкою миші по кнопці і один раз по вільному простору на моделі
	Створення категорії – для встановлення категоріальної зв'язку(спеціальний тип зв'язку між сутностями) необхідно клацнути лівою кнопкою миші по кнопці категорії, потім один раз клацнути по батьківській суті і потім один раз по суті-нащадку
	Створення ідентифікуючого зв'язку
	Створення не ідентифікуючого зв'язку
	Створення зв'язку «багато до багатьох»
	Кнопка покажчика (режим миші) – в цьому режимі можна встановлювати фокус на якомусь об'єкті моделі.
	Кнопка внесення текстового блоку

Залежно від глибини подання інформації про дані розрізняють 3 підрівні логічного рівня моделі даних:

- діаграма сутність-зв'язок (Entity Relationship Diagram, ERD);
- модель даних, заснована на ключах (Key Based model, KB);
- повна атрибутивна модель даних (Fully Attributed model, FA).

Діаграма сутність-зв'язок включає сутності і взаємозв'язки, що відображають основні бізнес-правила предметної області.

Модель даних, заснована на ключах – більш докладне уявлення даних. Дана модель включає опис всіх сутностей і первинних ключів, необхідних для докладного опису предметної області.

Повна атрибутивна модель даних – найбільш детальне представлення структури даних предметної області. Дана модель являє дані в третій нормальній формі і включає всі сутності, атрибути та зв'язку.

Для перемикання між логічною і фізичною моделлю даних служить список вибору в центральній частині панелі інструментів Erwin (Рис.5.5). Якщо при перемиканні фізичної моделі ще не існує, вона буде створена автоматично.

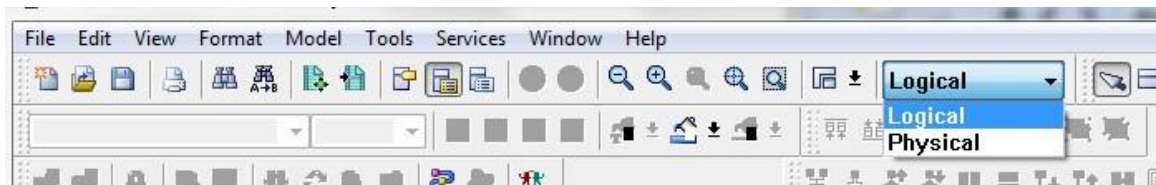


Рисунок 5.5 – Переключення між логічною та фізичною моделями

На логічному рівні ERwin підтримує дві нотації (IE і IDEF1X), на фізичному рівні – три нотації (IE, IDEF1X і DM). За замовчуванням використовується нотація IDEF1X (Integration DEFinition for information modeling).

Зміну нотації можна зробити на вкладці Notation діалогу Model Properties, що викликається через меню Model> Model Properties (рис. 5.6).

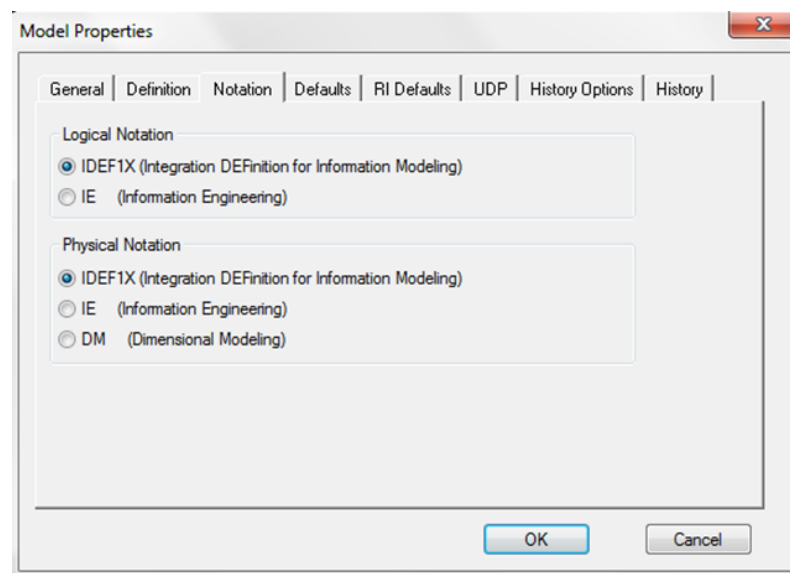


Рисунок 5.6 –Заміна нотації на вкладці Notation діалогу Model Properties

Рівні відображення моделі

ERwin має кілька рівнів відображення діаграми: рівень сутностей, рівень атрибутів, рівень визначень, рівень первинних ключів і рівень іконок.

Переключитися між першими трьома рівнями можна з використанням кнопок панелі інструментів.

Переключитися на інші рівні відображення можна за допомогою головного меню або контекстного меню, яке з'являється, якщо клацнути правою кнопкою миші по будь-якому місці діаграми, що не зайнятого об'єктами моделі. У контекстному меню слід вибрати пункт Format / Display Level і потім необхідний рівень відображення (Рис.5.7).

ERwin дозволяє зв'язати з сутністю велику і малу іконки. При перемиканні на рівень іконок показується велика іконка. Для відображення малої іконки слід вибрати в головному меню Format / Entity Display або в контекстному меню пункт Entity Display і в каскадному меню включити опцію Entity Icon (Рис.6.8). Мала іконка буде показана зліва від імені суті на всіх рівнях відображення моделі.

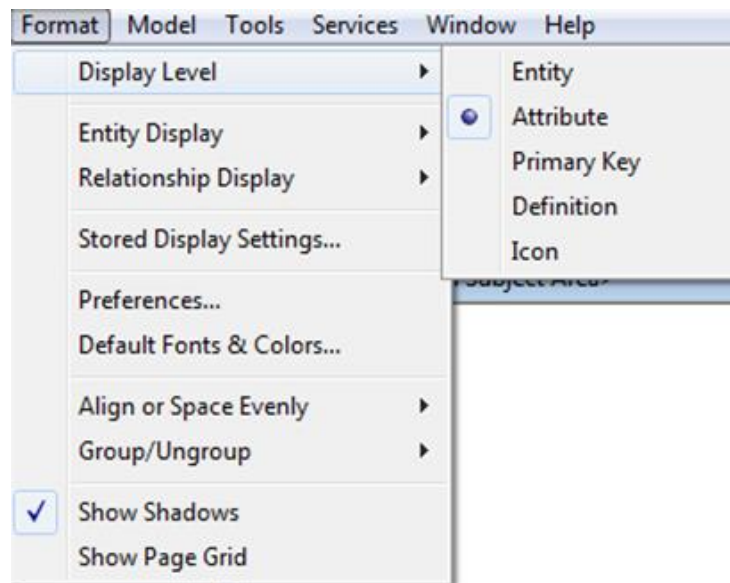


Рисунок 5.7 – Вибір рівнів відображення діаграми

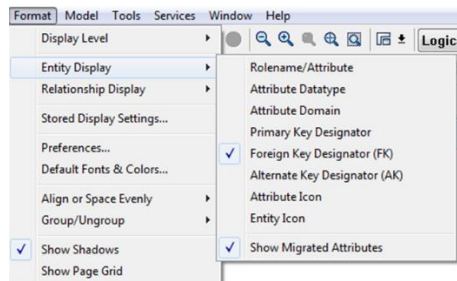


Рисунок 5.8 – Вибір відображення малої іконки

Установка кольору та шрифту

Для редагування шрифту і кольору всіх об'єктів моделі або для будь-якої окремої категорії об'єктів слід вибрати пункт меню *Format / Default Fonts & Colors* (Рис.5.9). Кожна закладка на діалозі дозволяє редагувати шрифт і колір для певної категорії об'єктів.

Для редагування шрифту і кольору конкретного об'єкта слід, натиснути правою кнопкою миші по сутності або зв'язку і вибрати з контекстного меню пункт *Object Font & Color*. Діалог *Entity Fonts & Colors* має поля, в яких можна вибрати шрифт і встановити його розмір, стиль і колір, встановити колір заливки (зкладка *Fill Color*, тільки для сутностей) і колір ліній (зкладка *Outline Color*, тільки для сутностей) (Рис. 5.10).

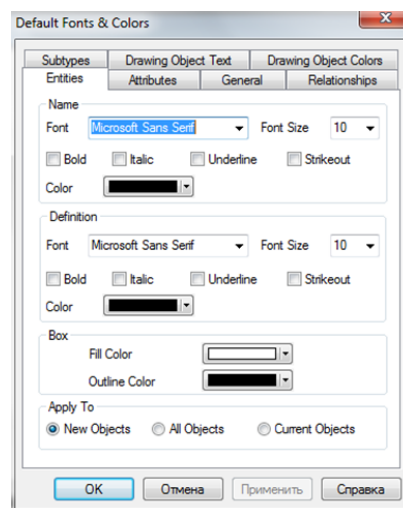


Рисунок 5.9 – Вікно вибору шрифтів і кольорів всіх об'єктів моделі

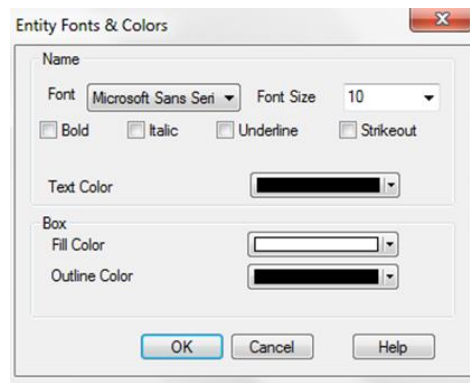


Рисунок 5.10 – Вікно вибору шрифту і кольору конкретної моделі

5.2 Створення логічної моделі даних

Створення моделі даних починається з розробки логічної моделі, яка повинна представляти склад сутностей предметної області з переліком атрибутів і відносин між ними.

Сутності

Для внесення суті в модель необхідно (переконавшись попередньо, що ви перебуваєте на рівні логічної моделі) клацнути лівою кнопкою миші по кнопці суті і розмістити в поле проектування моделі (перемістити курсор і клацнути лівою кнопкою миші); в результаті в поле проектування буде розміщений блок суті з ім'ям за умовчанням «E/1» (Рис.5.11).

За допомогою пункту головного меню Model / Entities або пункту спливаючого меню Entity Properties (по кліку правої кнопки миші в поле блоку сутності) відкривається вікно для визначення ім'я сутності (рис.5.12).

Діалог Entities (рис.5.12) містить вкладки, які дозволяють призначити і редагувати властивості сутності: ім'я Name, визначення Definition, примітки Note, Note2, Note3, визначені користувачем властивості UDP – User definition properties, іконки Icon, історію History.

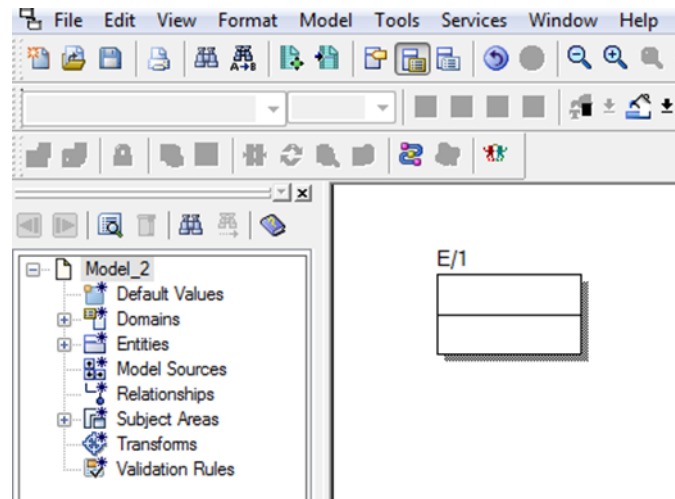


Рисунок 5.11 –Зображення сутності E/1 в полі проектування за замовчуванням

У вкладці Name задаємо ім'я сутності. Вкладка Definition використовується для введення визначення сутності у вигляді її текстового опису. На логічному рівні визначення дозволяють чіткіше зрозуміти об'єкт, а на фізичному рівні – їх можна експортувати як частина схеми і використовувати в реальній базі даних.

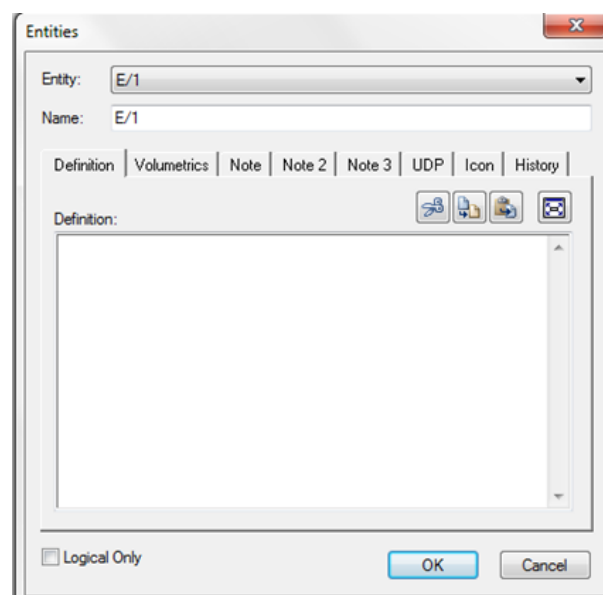


Рисунок 5.12 – Вікно для іменування сутності

Вкладка *Volumetrics* дозволяє вказати передбачуване початкове і максимальну кількість екземплярів сутності і можливе їх збільшення.

Вкладка *Note* дозволяє додавати додаткові зауваження про сутність, які не були відображені у визначенні, введеному в визначенні *Definition*. Тут можна ввести корисне зауваження, яке описує якесь бізнес-правило або угоду з організації діаграми.

Вкладка *Note 2* можна задокументувати деякі можливі запити, які, як очікується, будуть використовуватися по відношенню до сутності в БД. При переході до фізичного проектування, записані запити допоможуть приймати такі рішення щодо проектування, які зроблять БД ефективнішою.

Вкладка *Note 3* дозволяє вводити приклади даних для сутності (в довільній формі).

Вкладка *UDP (User – Defined Properties)* дозволяє користувачеві визначити властивості сутності і обсяг збережених даних. При натисканні на кнопку цієї вкладці викликається діалог *User Defined Property Editor* (також викликається з меню *Edit / UDPs*).

У ньому необхідно вказати вид об'єкта, для якого зводиться *UDP* (діаграма в цілому, сутність, атрибут і т.д.) і тип даних. Для внесення нової властивості слід клацнути в таблиці по кнопці і внести ім'я, тип даних, значення за замовчуванням і визначення.

Вкладка *Icon* дозволяє кожній сутності поставити у відповідність зображення, яке буде відображатися в режимі перегляду моделі на рівні іконок. У цій вкладці можна задати як велику іконку, яка буде відображатися на рівні *Icon*, так і малу іконку, яка буде відображатися на всіх рівнях перегляду моделі.

Вкладка *History* дозволяє переглянути історію всіх змін, пов'язаних із сутністю та додати коментар до зміни в вікні *comment*.

Атрибути

Для опису атрибутів і їх характеристик слід, натиснути правою кнопкою миші по сутності, вибрати в меню пункт Attribute. З'являється діалогове вікно Attributes (Рис.5.13).

При активізації кнопки New діалогового вікна Attributes відкривається діалозі New Attribute (рис.5.14), що дозволяє додати новий атрибут для виділеної сутності.

У діалоговому вікні Attribute Name вказуємо ім'я нового атрибута (поле Attribute Name), ім'я відповідної йому у фізичній моделі колонки (поле Column Name) і домен. Домен атрибута буде використовуватися при визначенні типу колонки на рівні фізичної моделі.

Атрибути повинні іменуватися в однині і мати чітке смислове значення, що дозволяє частково вирішити проблему нормалізації даних на етапі визначення атрибутів. Наприклад, створення для сутності Студент атрибута Телефони студента суперечить вимогам нормалізації, т. к. Атрибут повинен бути атомарним, тобто не містити множинних значень.

Згідно синтаксису нотації IDEF1X ім'я атрибута повинно бути унікальною в рамках моделі, тому новий атрибут, в разі збігу його імені з вже існуючим в рамках моделі, повинен бути перейменований.

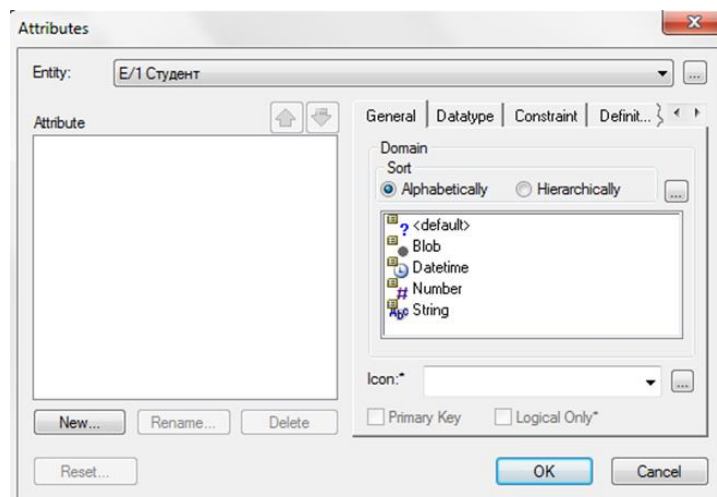


Рисунок 5.13 – Визначення атрибуту

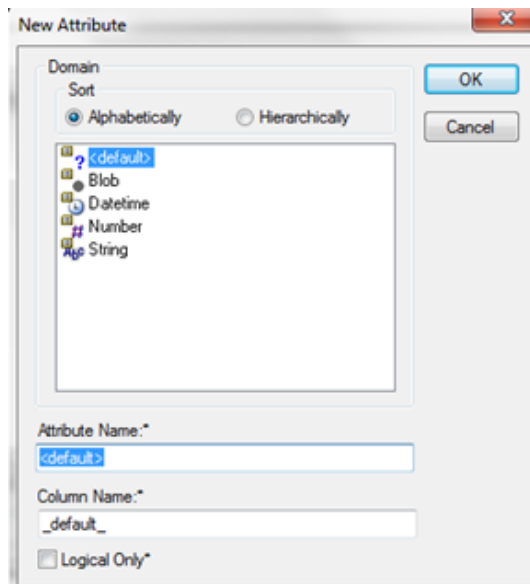


Рисунок 5.14 – Визначення нового атрибуту для сутності Студент

При активізації кнопки ОК новий атрибут додається в список атрибутів виділеної сутності, що відображається в полі Attribute діалогового вікна Attributes (рис. 5.15).

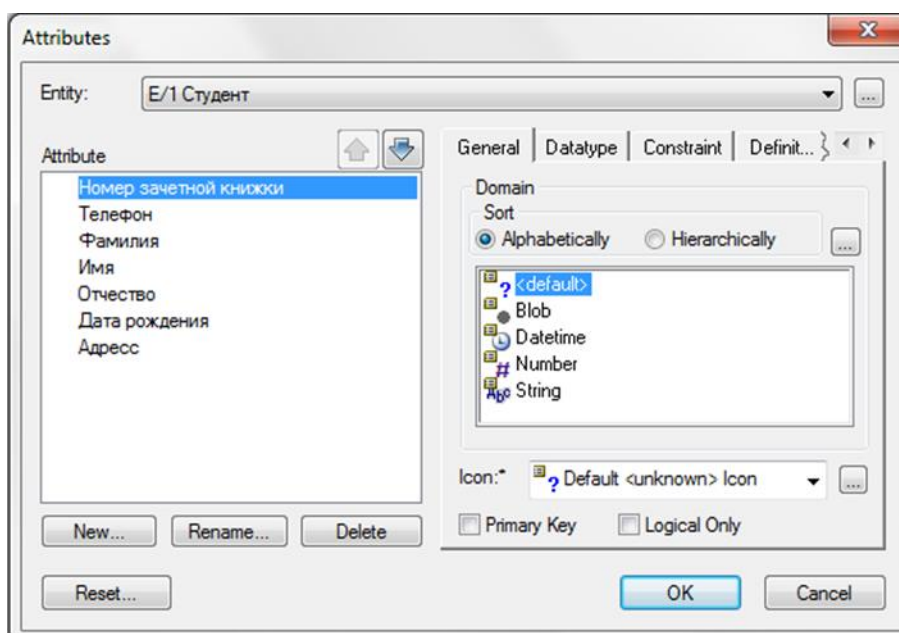


Рисунок 5.15 – Атрибути сутності Студент

Для атрибутів первинного ключа в закладці *General* діалогу *Attribute* необхідно зробити помітку в вікні вибору Primary Key

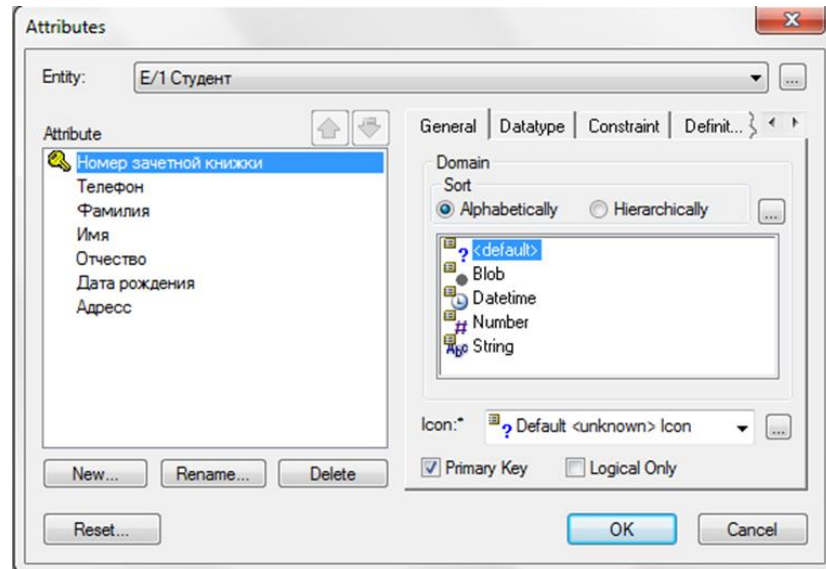


Рисунок 5.16 – Визначення первинного ключа в вікні вибору Primary Key

Для більшої наочності діаграми кожен атрибут можна пов'язати з іконкою. За допомогою списку вибору Icon в закладці *General* можна пов'язати іконку з атрибутом.

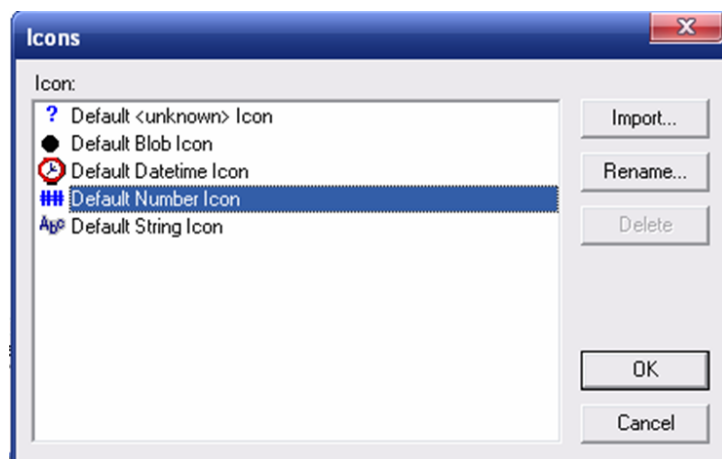


Рисунок 5.17 – Діалог Icons

Кожному домену відповідає стандартна іконка, однак можна імпортувати і додаткові зображення. Кнопка "..." праворуч від списку вибору Icon викликає діалог Icons, натиснувши кнопку Import можна додати в список необхідну іконку.

Для відображення іконки атрибута слід вибрати в контекстному меню пункт Format / Entities Display і в каскадному меню включити опцію Attribute Icon. Мала іконка буде показана зліва від імені атрибута на рівні атрибутів відображення моделі. Як видно з рис. 5.18, ім'я сутності показується над прямокутником, що зображує сутність, список атрибутів сутності – усередині прямокутника. Список розділений горизонтальною лінією, вище якої розташовані атрибути первинного ключа, нижче – не ключових атрибутів.



Рисунок 5.18 – Відображення сутності на рівні атрибутів з включено опцією Attribute Icon

Вкладка Datatype дозволяє за допомогою списку на сторінці уточнити тип даних для виділеного атрибуту (рис. 5.19).

Вкладка Constraint дозволяє задати обмеження для виділеного атрибуту. Вкладка Definition дозволяє записувати визначення окремих атрибутів.

Вкладка Note дозволяє додавати зауваження про одне або декілька атрибутах сутності, які не ввійшли в визначення.

Вкладка UDP служить для завдання значень властивостей, які визначаються користувачем. Попередньо ці властивості повинні бути внесені в діалог User-Defined Property Editor як властивості атрибутів.

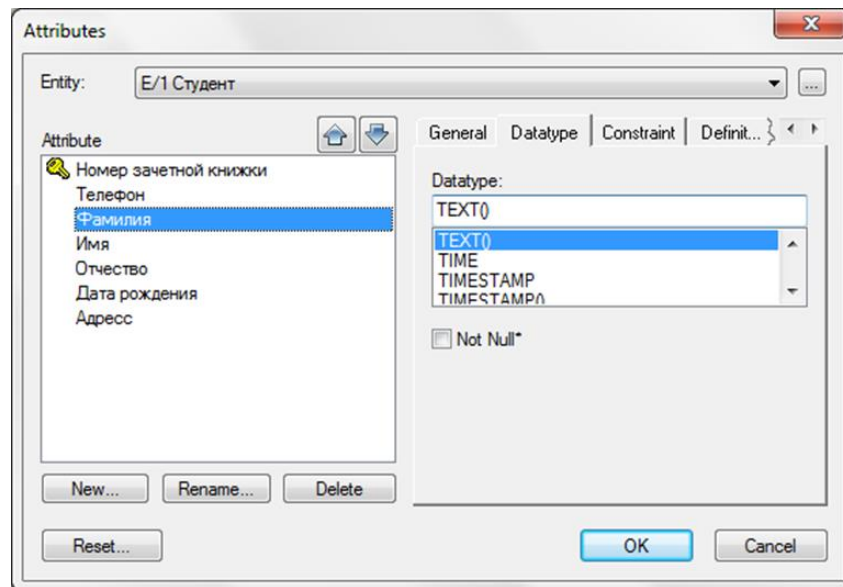


Рисунок 5.19 – Уточнення типу даних атрибуту Адрес

Вкладка Key Group (рис. 5.20) призначена для відображення атрибутів, що входять в групу ключових атрибутів сутності. Зокрема, на вкладці показано, що первинним ключем сутності Студент є атрибут Номер залікової книжки.

Сутність може мати кілька можливих (потенційних) ключів. Один потенційний ключ оголошується первинним, а решта можуть бути оголошені альтернативними ключами (Alternate Key).

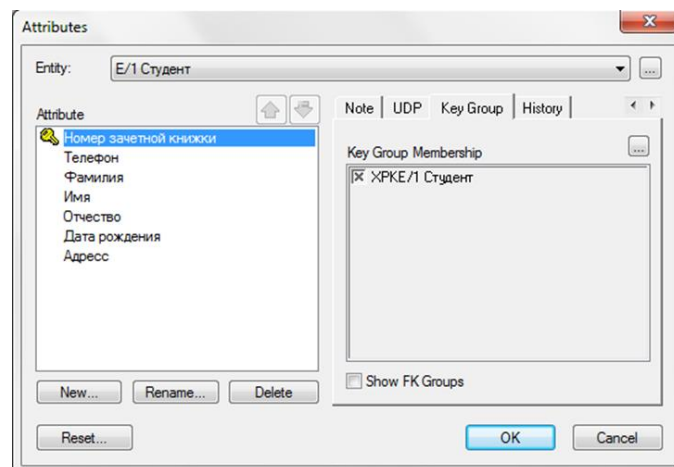


Рисунок 5.20 – Вкладка Key Group діалогу Attributes

ERwin за замовчуванням на фізичному рівні генерує унікальні індекси з первинного та альтернативним джерелам. ERwin дозволяє також будувати не унікальні індекси по атрибутам, званим інверсними входами (Inversion Entries). Не унікальний індекс забезпечує швидкий доступ не лише до рядку таблиці, а до кількох, які мають однакове значення інверсного входу. Це прискорює доступ до даних.

Альтернативні ключі і інверсні входи створюються в діалозі Key Groups, який з'являється після вибору відповідного пункту контекстного меню (рис. 5.21).

З рис.5.21 видно, що є первинний ключ (PK), який включає атрибут Номер залікової книжки. Створимо інверсний вхід по атрибуту Прізвище Студента. Для цього виділяємо атрибут Прізвище студента і натискаємо на кнопку New. З'являється діалог New Key Group (рис. 5.22).

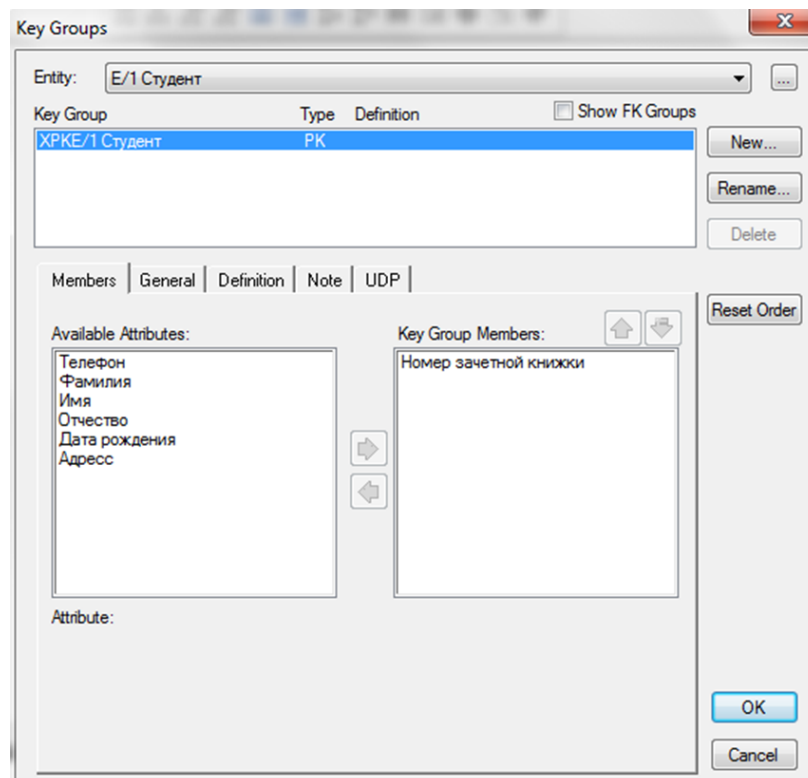


Рисунок 5.21 – Діалог Key Groups

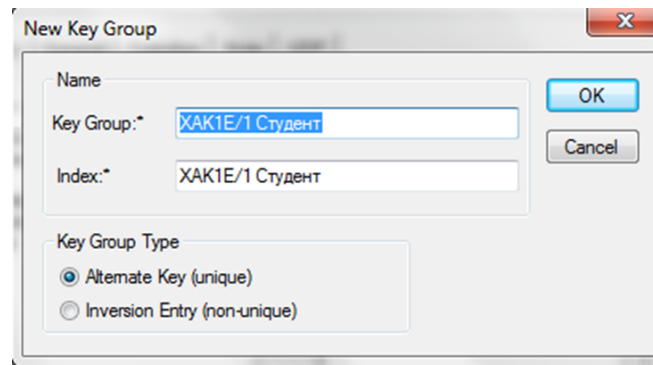


Рисунок 5.22 – Діалог New Key Group

У цьому діалозі вибираємо Inversion Entry. Ім'я ключової групи (Key Group) і індексу (Index) присвоюються автоматично. Клацаємо по кнопці ОК і повертаємося в діалог Key Groups (рис. 5.23).

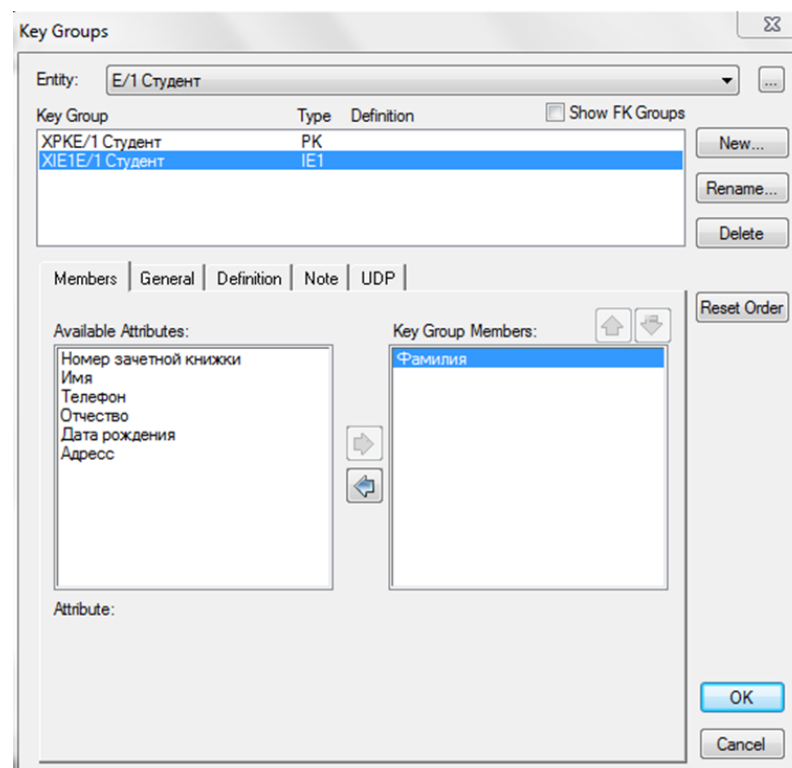


Рисунок 5.23 – Діалог Key Groups при побудові інверсного входу

У верхньому вікні Key Group вкладки вибираємо ключову групу ХІЕ1 Студент, що є інверсним входом (ІЕ). У вікні Available Attributes вибираємо

атрибут Прізвище студента і за допомогою кнопки включаємо його до складу ключової групи.

Альтернативні ключі створюються аналогічно.

Вкладка History діалогу Attributes відображає історію створення і зміни властивостей атрибута і дозволяє додати коментарі до зміни в вікні Comment (рис. 5.24).

При натисканні на кнопку ОК в полі проектування моделі відобразиться сутність з атрибутами, визначеними користувачем (рис. 5.25).

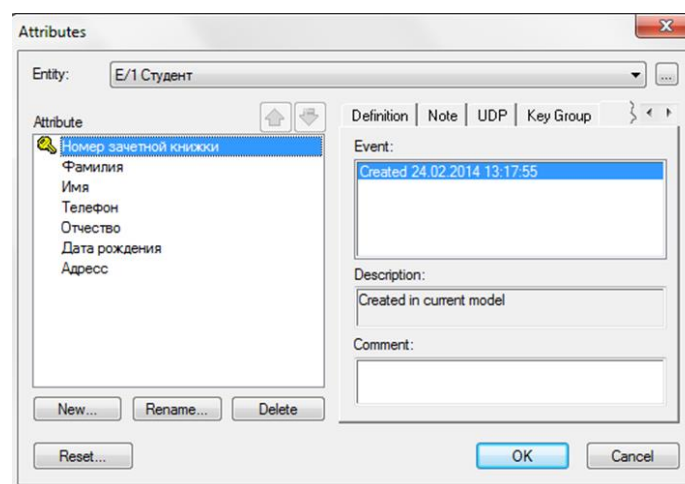


Рисунок 5.24 – Історія створення змін властивостей атрибута Фамілія

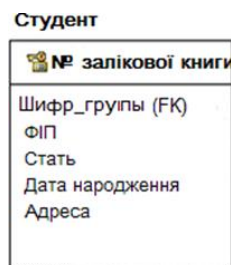


Рисунок 5.25 – Сутність студент з заданими параметрами

Діалогове вікно Attributes дозволяє користувачеві редагувати імена атрибутів виділеної сутності і коригувати список атрибутів шляхом виконання операції видалення помилково певного атрибута.

Редагування імені атрибута здійснюється в діалозі Rename Attribute, який відкривається при активізації кнопки діалогового вікна Attributes (рис. 5.26).

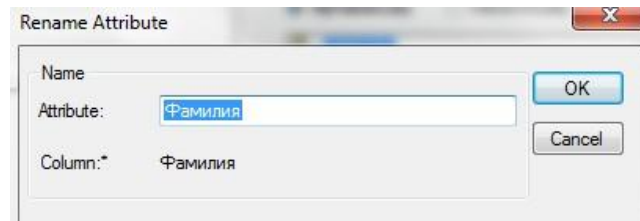





Рисунок 5.26 – Редагування імені атрибута Фамілія студента

Створення зв'язків

Зв'язок є логічним співвідношенням між сутностями. Кожному виду зв'язку відповідає певна кнопка, розташована на палітрі інструментів.

Для створення нового зв'язку необхідно встановити курсор на кнопці, розташованій на палітрі інструментів *ERwin Toolbox* і відповідної необхідному виду зв'язку. Клацнути лівою кнопкою миші спочатку по батьківській, а потім по дочірній сутності. У палітрі інструментів кнопки:

-  відповідає ідентифікує зв'язку (один-до-багатьох),
-  зв'язку багато-до-багатьох;
-  відповідає не ідентифікуючому зв'язку (один-до-багатьох).

Змінити форму лінії зв'язку і її розташування між пов'язаними сутностями можна шляхом захоплення мишею виділеної лінії зв'язку і перенесення її з місця на місце, поки лінія не прийме необхідне розташування та форму.

В нотації IDEF1X розрізняють залежні і незалежні сутності. Тип суті визначається її зв'язком з іншими сутностями. Ідентифікуючий зв'язок встановлюється між незалежною (батьківський кінець зв'язку) і залежною (дочірній кінець зв'язку) сутностями та показується на діаграмі суцільною лінією з жирною крапкою на дочірньому кінці зв'язку.

При встановленні ідентифікуючого зв'язку ERwin автоматично

перетворює дочірню сутність в залежну. Залежна сутність зображується прямокутником із закругленими кутами (сутність *Дисципліна* на рис. 5.27). При встановленні ідентифікуючого зв'язку атрибути первинного ключа батьківської сутності автоматично переносяться до складу первинного ключа дочірньої сутності.

Ця операція доповнення атрибутів дочірньої сутності при створенні зв'язку називається міграцією атрибутів. У дочірньої сутності нові атрибути позначаються як зовнішній ключ (FK).

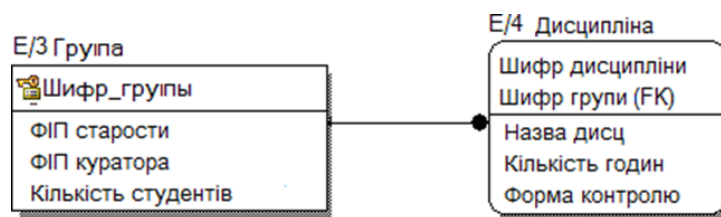


Рисунок 5.27 – Приклад залежної сутності Дисципліна

Неідентифікуючий зв'язок показується на діаграмі пунктирною лінією з жирною крапкою і служить для встановлення зв'язку між незалежними сутностями. При встановленні неідентифікуючого зв'язку дочірня сутність залишається незалежною, а атрибути первинного ключа мігрують до складу не ключових атрибутів батьківської сутності (рис. 5.28).

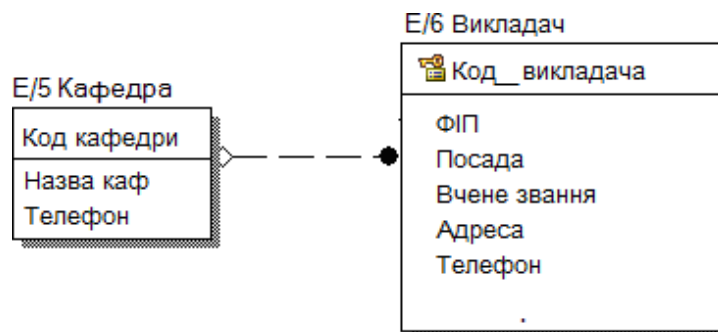


Рисунок 5.28 – Приклад не ідентифікуючого зв'язку між незалежними сутностями

Зв'язок "багато-до-багатьох" може бути створений тільки на рівні логічної моделі. Такий зв'язок позначається суцільною лінією з двома крапками на кінцях. Встановлення зв'язку "багато-до-багатьох" здійснюється при активізації відповідної кнопки на палітрі інструментів за допомогою клацання лівої кнопки миші спочатку по одній, а потім по іншій сутності.

З метою полегшення читання діаграми зв'язок "багато-до-багатьох" повинен мати двох направлених імя: від батьківської до дочірньої сутності (*Parent-to-Child*) та від дочірньої до батьківської сутності (*Child-to-Parent*).

Прикладом зв'язку "багато-до-багатьох" може служити зв'язок між сутностями *Група* та *Викладач* (рис. 5.29), кожна Група вчить одну або декілька дисциплін а кожен Викладач викладає у однієї або декількох груп.

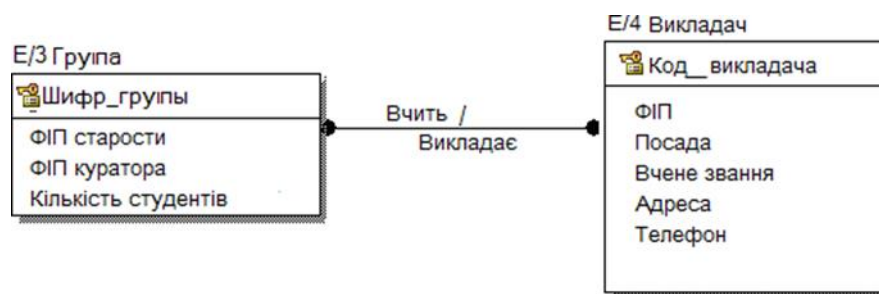


Рисунок 5.29 – Приклад зв'язку багато-до-багатьох

Нотація IDEF1X вимагає, щоб на фізичному рівні зв'язок "багато-до-багатьох" був перетворений, так як СУБД не підтримують зв'язок "багато-до-багатьох". За замовчуванням при переході до фізичного рівня ERwin автоматично НЕ перетворює зв'язок "багато-до-багатьох". У цьому випадку на фізичному рівні діаграма виглядає так само, як і на логічному, однак при генерації схеми бази даних системи такий зв'язок ігнорується. Тому вже на логічному рівні переважно таких зв'язків уникати. Крім того, цього вимагає і синтаксис нормалізації до третьої нормальної форми.

ERwin дозволяє реалізувати примусове перетворення зв'язку "багато-до-багатьох", яка включає створення нової (зв'язує) таблиці і двох нових зв'язків "один-до-багатьох" від старих таблиць до нової таблиці. При цьому імя нової

таблиці може присвоюватися як автоматично, так і визначатися користувачем. У разі автоматичного присвоєння імені зв'язуючи таблиці призначається ім'я, що включає імена обох сутностей.

Для примусового перетворення відношення багато-до-багатьох в два відношення один-до-багатьох необхідно виділити зв'язок, вибрати пункти меню Edit / Many to Many transform (рис.5.30)

Діалог Many-To-Many Transform Wizard пропонує виконати чотири кроки для перетворення зв'язку. Для переходу до наступного кроку необхідно клацнути по кнопці Далі. На другому і третьому кроці слід задати ім'я новостворюваної таблиці і ім'я перетворення. Після закінчення виконання дій діалогу Many-To-Many Transform Wizard на діаграмі буде додана нова таблиця з заданим користувачем ім'ям, а зв'язок "багато-до-багатьох" заміниться ідентифікуючими зв'язками "один-до-багатьох " від старих таблиць до нової таблиці. При цьому нові зв'язки автоматично отримують відповідні імена від зв'язку "багато-до-багатьох".

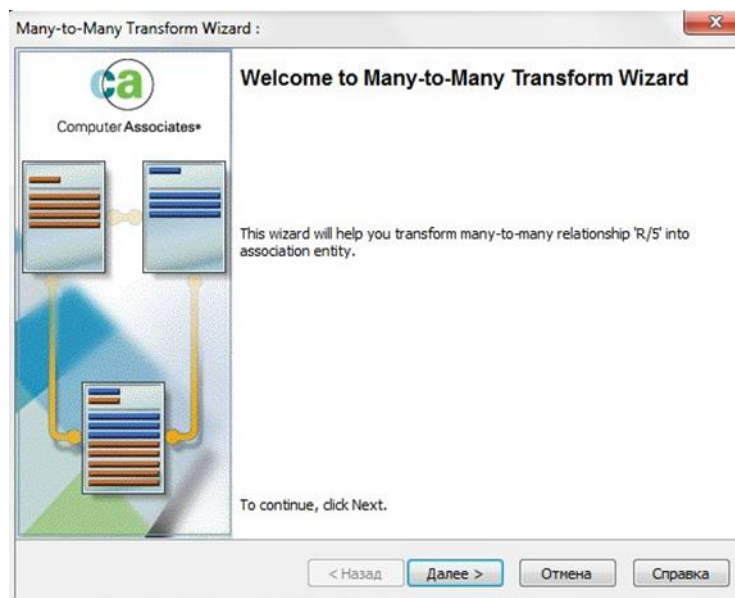


Рисунок 5.30 – Вікно діалогу Many-To-Many Transform Wizard

Приклад результату примусового перетворення зв'язку "багато-до-

багатьох" між сутностями *Група* та *Викладач*, реалізованого за допомогою діалогу *Many-To-Many Transform Wizard*, наведено на рис.5.31.

Особливим типом об'єднання сутностей є ієрархія наслідування (ієрархія категорій або категоріальна зв'язок), згідно з якою поділяються їх загальні характеристики. Зазвичай ієрархію спадкування створюють, коли кілька сутностей мають загальні за змістом атрибути, або коли сутності мають загальні за змістом зв'язку, або коли це диктується бізнес – правилами. Щоб представити інформацію, загальну для всіх типів екземплярів сутності, з їхніми спільними властивостями можна сформувану узагальнену сутність (родовий предок).



Рисунок 5.31 – Приклад результату примусового перетворення зв'язку багато-до-багатьох

При цьому специфічна для кожного типу екземпляру сутності інформація буде розташовуватися в категоріальних сутності (нащадках).

Наприклад, на факультеті працюють Декан, Викладач і Інженер, які мають подібний за змістом зв'язок "працює в" з сутністю Факультет. Щоб уявити інформацію, загальну для всіх типів працівників, з їхніх спільних властивостей можна сформувану узагальнену сутність Співробітник факультету. При цьому категоріальним сутностями будуть Декан і Викладач (рис. 5.32).

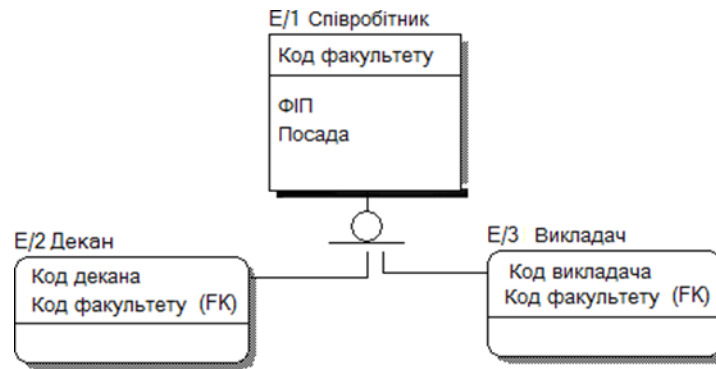


Рисунок 5.32 – Приклад ієрархії наслідування (категоріального зв'язку)

Для створення категоріального зв'язку слід:

- встановити курсор на кнопці, розташованій на палітрі інструментів і натиснути ліву кнопку миші;
- клацнути спочатку по родовому предку, а потім – по нащадку;
- встановити другу зв'язок в ієрархії категорії, вибравши кнопку і клацнувши спочатку по символу категорії на діаграмі, потім – по другому нащадку.

Редагування категорій здійснюється в діалоговому вікні *Subtype Properties*, яке відкривається при виборі пункту *Subtype Properties ...* контекстного меню, яке відображається після клацання правою кнопкою миші по символу категорії (рис. 5.33).

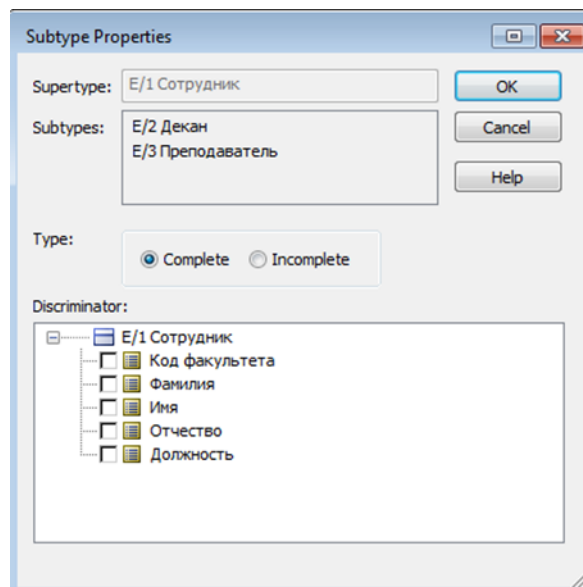


Рисунок 5.33 – Діалогове вікно Subtype Properties

Поля діалогового вікна дозволяють вказати дискримінатор – атрибут категорії (*список Discriminator*) і тип категорії – повна / неповна (кнопки *Complete / Incomplete*).

Дискримінатор категорії – це атрибут родового предка, який показує, як відрізнити одну категоріальну сутність від іншої.

Ієрархії категорій діляться на два типи – повні і неповні. Повною категорії одному примірнику родового предка обов'язково відповідає екземпляр в будь-якому нащадку. Наприклад, співробітник обов'язково є або деканом, або викладачем.

Якщо категорія ще не вибудована повністю і в родового предка можуть існувати екземпляри, які не мають відповідних примірників в нащадках, то така категорія буде неповною. Наприклад, співробітник може бути не тільки Деканом або Викладачем, але і Інженером. У разі неповної категорії сутність Інженер ще не внесена в ієрархію наслідування (рис. 5.32). На рис.5.34 представлений приклад повної категорії. Можлива також комбінація повної і неповної категорій.



Рисунок 5.34 – Приклад повної категорії

Редагування зв'язків

Для редагування властивостей зв'язку слід натиснути правою кнопкою миші по зв'язку і вибрати в пункті контекстного меню пункт *Relationship Property* (рис.5.35)

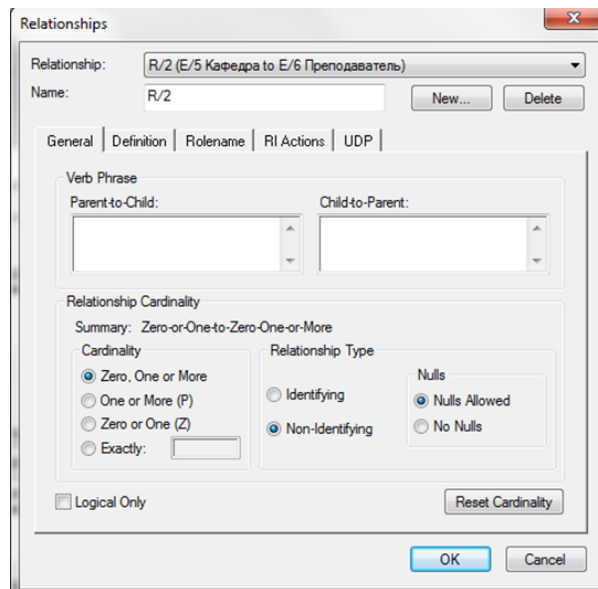


Рисунок 5.35 – Вікно редагування зв'язку

Вкладка General діалогового вікна може задати потужність, ім'я та тип зв'язку.

Потужність зв'язку (Cardinality) служить для позначення відношення числа примірників батьківської сутності до числа екземплярів дочірньої сутності. Розрізняють 4 типи потужності зв'язку (рис. 5.36).

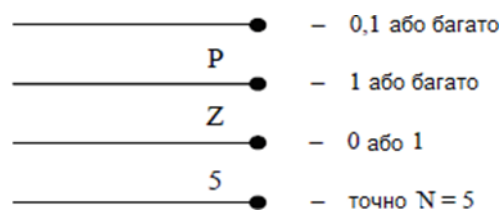


Рисунок 5.36 – Позначення потужності зв'язків

За замовчуванням символ, що позначає потужність зв'язку, не з'являється на діаграмі. Для відображення потужності з цим слід включити опцію Cardinality пункту Relationship Display контекстного меню, яке з'являється після клацання правою кнопкою миші по будь-якому місцю діаграми, яке не зайняте об'єктами моделі.

Ім'я зв'язку (Verb Phrase) – фраза, що характеризує відношення між

батьківської та дочірньої сутностями. Для зв'язку "один-до-багатьох" (ідентифікуюча або неідентифікуюча) досить вказати ім'я, що характеризує відношення від батьківської до дочірньої сутності (Parent-to-Child), а для зв'язку "багато-до-багатьох" необхідно вказувати ім'я зв'язку як від батьківської до дочірньої сутності (Parent-to-Child), так і від дочірньої до батьківської сутності (Child-to-Parent).

Тип зв'язку (Relationships Type) дозволяє позначити ідентифікуючий (Identifying) або не ідентифікуючий (Non-Identifying) зв'язок. Для не ідентифікуючого зв'язку необхідно вказати обов'язковість зв'язку (Nulls Allowed або No Nulls).

У разі обов'язкової зв'язку (No Nulls), незважаючи на те, що зовнішній ключ не увійде до складу первинного ключа дочірньої сутності, при генерації схеми бази даних атрибут зовнішнього ключа отримає ознака NOT NULL. У разі необов'язковою зв'язку (Nulls Allowed) зовнішній ключ може приймати значення NULL.

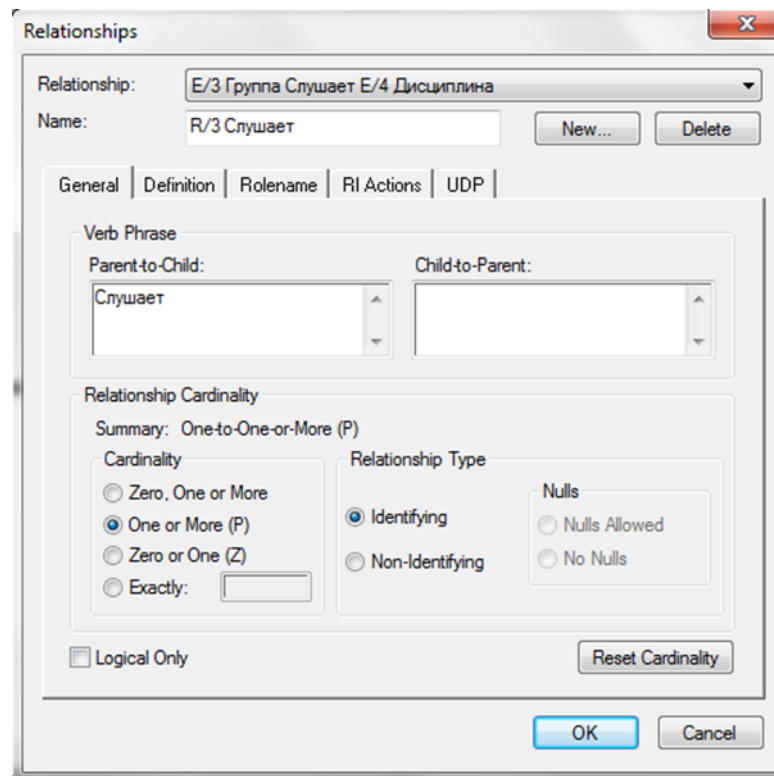


Рисунок 5.37 – Приклад визначення потужності зв'язку між сутностями
Група та Дисципліна

Це означає, що екземпляр дочірньої сутності не буде пов'язаний ні з одним екземпляром батьківської сутності. Необов'язковий неідентифікуючий зв'язок позначається прозорим ромбом з боку батьківської сутності (рис. 5.38).

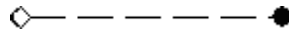


Рисунок 5.38 – Графічне представлення необов'язкового не ідентифікуючого зв'язку

Вкладка Definition дозволяє дати більш повне визначення зв'язку для того, щоб в подальшому мати можливість на нього посилатися.

Вкладка Rolename може задати ім'я ролі атрибута зовнішнього ключа і правила посилальної цілісності. Ім'я ролі (функціональне ім'я) – це синонім атрибута зовнішнього ключа, який показує, яку роль відіграє атрибут в дочірньої сутності (рис.5.39).

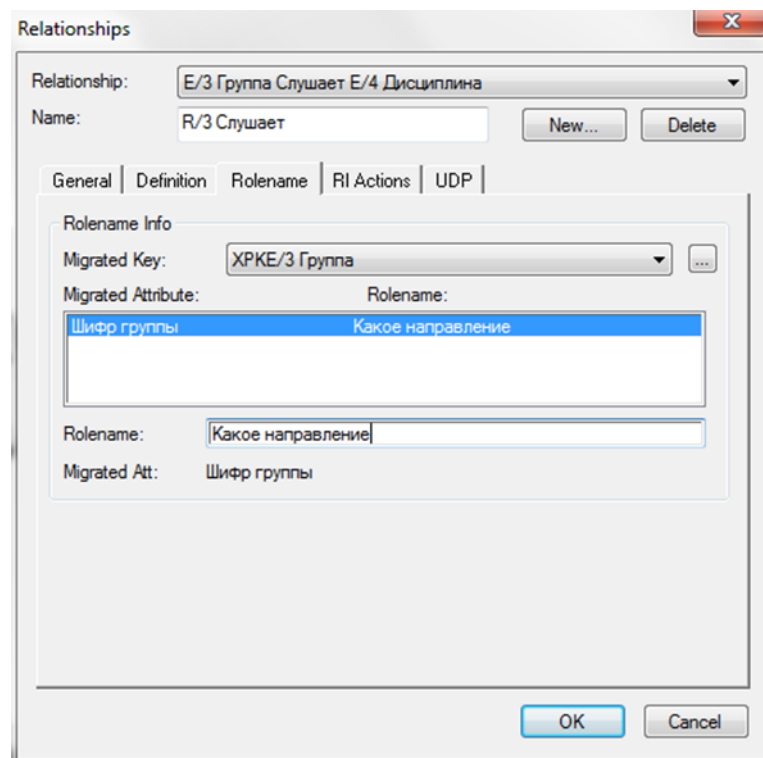


Рисунок 5.39 – Вікно діалогу Relationships вкладки Rolename

Ключі

Первинний ключ (*primary key*) це атрибут або група атрибутів, однозначно ідентифікує екземпляр сутності. Атрибути первинного ключа на діаграмі не вимагають спеціального позначення - це ті атрибути, які знаходяться в списку атрибутів вище горизонтальної лінії.

При внесенні нового атрибута в діалозі Attribute Editor для того, щоб зробити його атрибутом первинного ключа, потрібно включити прапорець Primary Key в нижній частині закладки General Рис. 5.40. На діаграмі не ключовий атрибут можна внести до складу первинного ключа, скориставшись режимом перенесення атрибутів (кнопка на панелі інструментів).

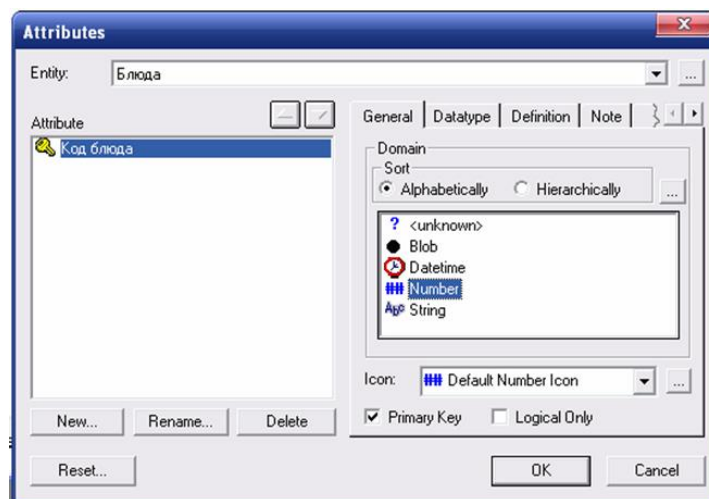


Рисунок 5.40 – Задання ключа

Домени

У ERwin домен може бути визначений тільки один раз і використовуватися як в логічній, так і в фізичній моделі.

Для створення домену в логічній моделі служить діалог Domain Dictionary Editor, (Рис. 5.41). Його можна викликати з меню Edit / Domain Dictionary по кнопці, розташованій у верхній лівій частині закладки General

діалогу Attribute Editor. Для створення нового домену в діалозі Domain Dictionary Editor слід:

- клацнути по кнопці New. З'являється діалог New Domain (рис. 5.42);
- вибрати батьківський домен зі списку Domain Parent.

Новий домен можна створити на основі вже створеного користувачем домену або на основі спочатку існуючого. За замовчуванням ERwin має чотири домени (*String*, *Number*, *Blob*, *Datetime*). Новий домен успадковує всі властивості батьківського домену. Ці властивості в подальшому можна перевизначити;

- набрати ім'я домену в полі Logical Name. Можна також вказати ім'я домену на фізичному рівні в поле Physical Name. Якщо фізична ім'я не вказано, за замовчуванням воно приймає значення логічного імені;
- клацнути по кнопці ОК.

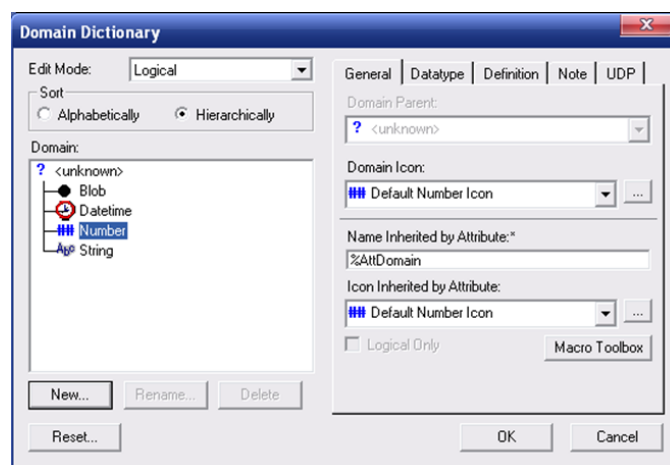


Рисунок 5.41 – Діалог Domain Dictionary Editor

У діалозі *Domain Dictionary Editor* можна зв'язати домен з іконкою, з якої він буде відображатися в списку доменів *Domain Icon*, і іконкою, з якої атрибут, визначений на домені, буде відображатися в моделі *Icon Inherited by Attribute*.



Рисунок 5.42 – Діалог New Domain

Кожен домен може бути описаний в закладці Definition, забезпечений коментарем в закладці Note або властивістю, певним користувачем в закладці UDP.

ЛАБОРАТОРНА РОБОТА №6
«ФІЗИЧНА МОДЕЛЬ ДАНИХ ТА ГЕНЕРАЦІЯ КОДУ КЛІЄНТСЬКОЇ
ЧАСТИНИ ЗА ДОПОМОГОЮ ERWIN»

Мета роботи:

- знайомство з інструментальним середовищем програми erwin.
- побудова фізичної моделі даних.
- генерація моделі даних в вибрану субд

Завдання для лабораторної роботи

Побудова фізичної моделі даних:

- вибір цільової СУБД, виділення уявлення, правила валідації для колонок, значення за замовчуванням, визначення індексованих полів.
- побудова на базі логічної моделі фізичної моделі.
- генерування БД на мові цільової СУБД.

Контрольні питання:

1. Що таке зв'язок багато до багатьох в моделі БД?
2. Які правила зображення зв'язку багато до багатьох в логічній моделі БД?
3. Які правила розв'язання зв'язку багато до багатьох в фізичній моделі БД?
4. Поясніть в чому різниця між логічною та фізичною моделлю в AllFusion ERwin Data Modeler?
5. Як задати правила валідації на фізичній моделі?
6. Як згенерувати структуру БД з фізичної моделі в AllFusion ERwin Data Modeler?
7. Яка команда слугує для модифікації структури таблиці?
8. Чи можна створити об'єднання декількох таблиць?
9. Як відредагувати структуру таблиці?

10. Що таке представлення?
11. Для чого використовують представлення?
12. Якими методами можна створити представлення?

Звіт про виконання лабораторної роботи:

1. Тема лабораторної роботи.
2. Відповіді на запитання.
3. Результати виконання завдання в електронному вигляді.

Теоретична підготовка до виконання лабораторної роботи:

Для виконання лабораторної роботи слід повторити теоретичний матеріал попередніх лабораторних робіт.

Література:[2], [3], [5]

Практична підготовка до виконання лабораторної роботи

Побудова фізичної моделі даних. Запуск середовища ERWin

Запустити середовище розробки ERWin. При створенні моделі виникає діалог, в якому слід вибрати режим відкрити існуючу модель *Open existing file*.

Після опису логічної моделі, вибираємо необхідну *СУБД* і ERwin автоматично перетворює логічну модель в відповідну фізичну модель.

Для перемикання між логічною і фізичною моделлю даних служить *список вибору* в центральній частині панелі інструментів Erwin.

При переході на фізичну модель зміниться палітра інструментів, з'являються нові кнопки, зміниться вид сутностей в графічній моделі додадуться позначення типів даних (Рис. 6.1).

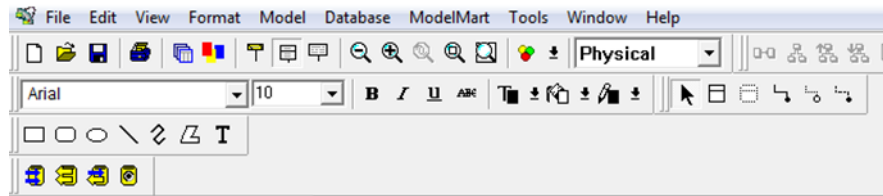



Рисунок 6.1 – Перемикач між логічною та фізичною моделями

Вибір сервера

Фізичний рівень представлення моделі залежить від обраного сервера. Для вибору СУБД служить редактор *Target Server* (Рис. 6.2). Його можна викликати з меню *Database / Choose Database* (доступно тільки на фізичному рівні) або за допомогою кнопки .

Діалог *Target Server* дозволяє задати тип даних і опцію *NULL* для нових колонок, а також правила зсилочної цілісності, прийняті за замовчуванням.

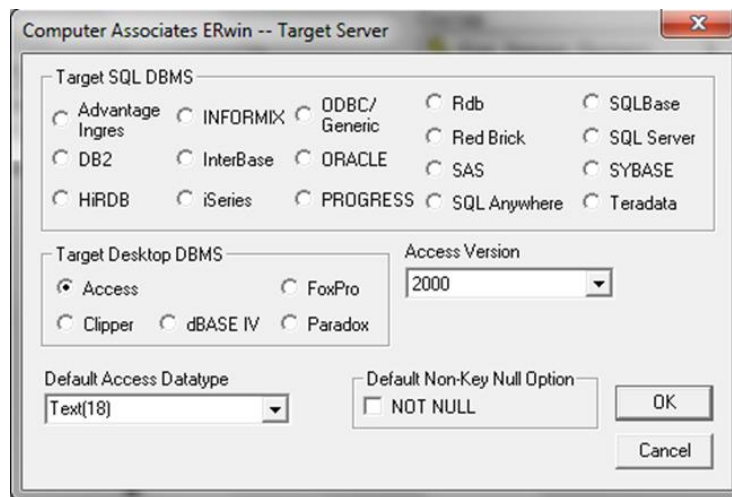



Рисунок 7.2 – Редактор Target Server

Група кнопок *Default Non Key Null Option* дозволяє вирішити або заборонити значення *NULL* для не ключових колонок.

За замовчуванням ERwin генерує імена таблиць і індексів за шаблоном на основі імен відповідних сутностей і ключів логічної моделі. Імена таблиць і колонок за замовчуванням будуть згенеровані на основі імен сутностей і атрибутів логічної моделі. Якщо в імені суті або атрибута зустрічається

пробіл, він буде замінений на символ "_".

Таблиці

Для внесення нової таблиці в модель на фізичному рівні служить кнопка на палітрі інструментів . Зв'язки між таблицями створюються так само, як на логічному рівні. Клацнувши правою клав'яшею миші по таблиці і вибравши у спливаючому меню пункти *Table Properties / Comment*, можна викликати редактори для задання властивостей таблиць і колонок (Рис. 6.3).

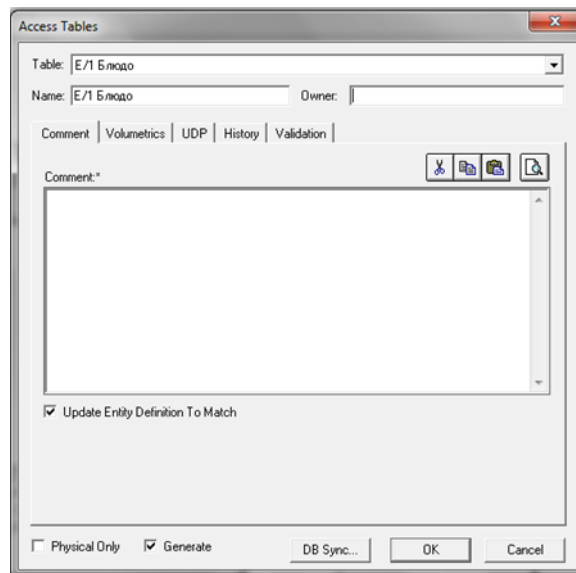


Рисунок 6.3 – Редактор таблиць Access Tables

Редактор *Access Tables* дозволяє задати властивості будь-якої таблиці моделі, відмінні від значення за замовчуванням, в тому числі ім'я таблиці, синоніми, правила валідації, процедури і т.д.

Переключитися на іншу таблицю можна за допомогою списку вибору у верхній частині діалогу.

Поле *Name* служить для задання імені поточної таблиці.

Поле *Owner* дозволяє внести ім'я власника таблиці, відмінне від імені користувача, що проводить генерацію схеми БД.

Вікно вибору *Physical Only* служить для створення об'єктів тільки на фізичному рівні.

Якщо Ви вибрали *Generate*, при генерації схеми БД буде виконуватися команда *CREATE TABLE*.

Кнопка *DB Sync* служить для негайної синхронізації моделі з системним каталогом БД.

Діалог *Access Tables* містить наступні закладки:

У закладці *Comment* вносяться коментарі до таблиці. Закладка *Volumetrics* служить для оцінки розміру БД.

У закладці *UDP* задаються властивості, визначені користувачем. У закладці *Validation* задаються правила валідації.

Колонки

Для задання властивостей колонок, відмінних від значення за замовчуванням, служить редактор *Columns* (Рис. 6.4). Щоб викликати його, потрібно натиснути правою клавішею миші по таблиці і вибрати у спливаючому меню пункт *Columns* або *Model / Columns*.

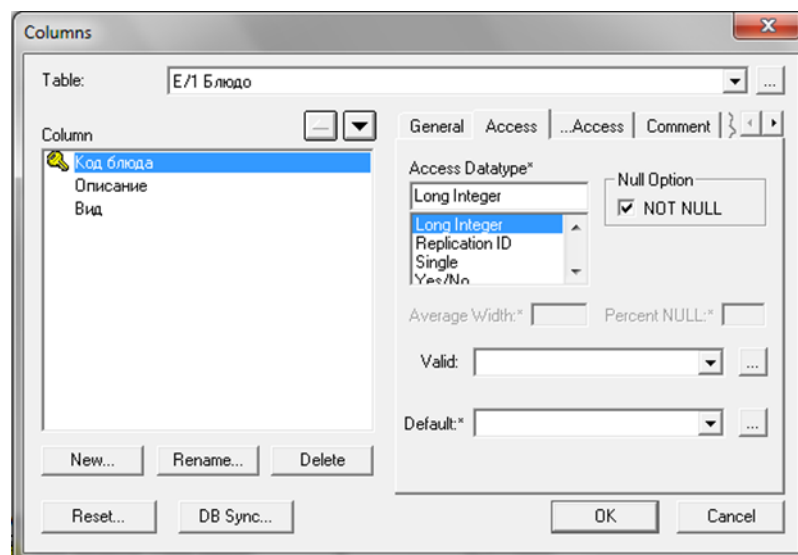


Рисунок 6.4 – Діалог Columns

За замовчуванням ERwin привласнює режими нульових значень всім неключових колонкам, виходячи з значень за замовчуванням, що

встановлюються в редакторі *Target Server*. Для колонок первинного ключа і альтернативних ключів в поле *Null Option* встановлюється режим *NOT NULL*.

Режим *NOT NULL* присвоюють автоматично інверсійним входів (*Inversion Entry*).

Зовні діалог *Columns* нагадує діалог *Attribute Editor*. У правій частині діалогу знаходяться закладки:

Закладка *General* дозволяє привласнити колонку певного домену, створити колонку тільки на фізичному рівні і включити її до складу первинного ключа.

Закладка, відповідна обраної СУБД (*Access*). Ім'я закладки встановлюється автоматично відповідає обраній СУБД. Дозволяє задати тип даних, опцію *NULL*, правила валідації і значення за замовчуванням.

Закладка *Comment* служить для внесення коментаря до кожної колонці. У закладці *UDP* задаються властивості, визначені користувачем.

Закладка *Data Source* доступна тільки при моделюванні сховищ даних. Закладка *Index* служить для включення колонки до складу індексів.

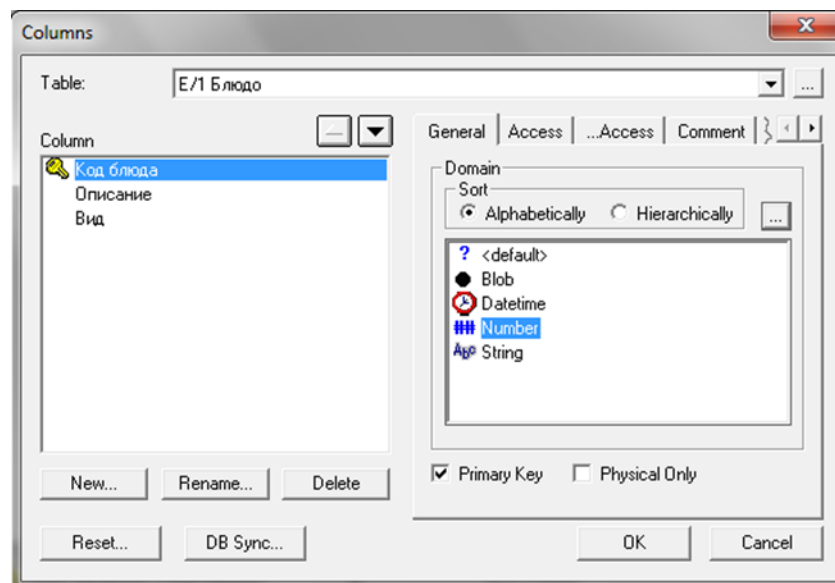




Рисунок 6.5 – Вікно редагування колонок

У лівій частині діалогу міститься впорядкований список колонок таблиці. Кнопки  призначені для переміщення колонки в списку на позицію вгору і вниз. Кнопки *New*, *Rename* і *Delete* служать відповідно для створення, перейменування і видалення колонки. За допомогою кнопки *Reset* можна перевстановити властивості колонки, задані вручну, на значення за замовчуванням. Кнопка *DB Sync* дозволяє запустити процес синхронізації моделі з системним каталогом БД.

Представлення

Представлення *View*, або, як їх іноді називають, тимчасові або похідні таблиці, є об'єктами БД, дані в яких не зберігаються постійно, як в таблиці, а формуються динамічно при зверненні до подання. Подання не може існувати саме по собі, а визначається тільки в термінах однієї або декількох таблиць. Застосування представлень дозволяє розробнику БД забезпечити кожному користувачу або групі користувачів свій погляд на дані, що вирішує проблеми простоти використання і безпеки даних.

ERwin має спеціальні інструменти для створення і редагування представлень. Палітра інструментів на фізичному рівні містить кнопки внесення подань та встановлення зв'язків між таблицями і уявленнями. Для внесення подання потрібно клацнути по кнопці  в палітрі інструментів, потім по вільному місцю діаграми. За замовчуванням уявлення отримує номер V_n , де n – унікальний порядковий номер подання (рис.6.6).

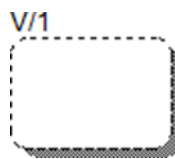



Рисунок 6.6 – Представлення

Для встановлення зв'язку потрібно клацнути по кнопці , потім по батьківській таблиці і, нарешті, по поданню. Зв'язки з представленнями і прямокутники представлень показуються на діаграмі пунктирними лініями.

Для редагування уявлення служить діалог *Views*. Для його виклику слід натиснути правою кнопкою миші по наданню й вибрати в меню пункт *Views* (рис.6.7).

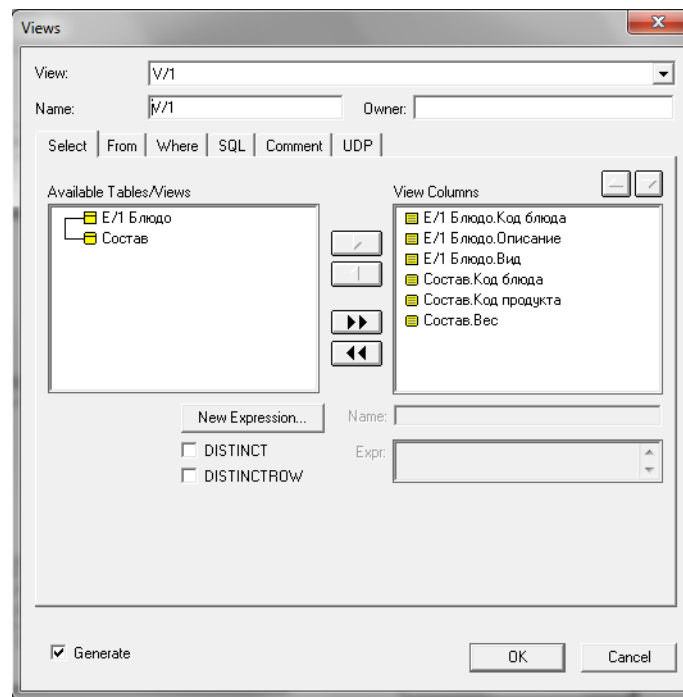


Рисунок 6.7 – Діалог View Editor

Список, що розкривається *Views* дозволяє вибрати для редагування будь-яке представлення моделі.

Вікно *Name* служить для редагування імені.

Вікно *Owner* – власника представлення.

Кнопка *New Expression* дозволяє задати вираз в якості вихідного стовпця. За замовчуванням при створенні зв'язку ERwin включає в представлення всіх колонок батьківських таблиць.

Діалог *Views* має наступні закладки:

Закладка *Select* має два списки: в правому відображаються колонки представлення, в лівому – колонки доступні для включення в представлення.

Закладка *From* дозволяє вибрати батьківські таблиці представлення. За замовчуванням включаються таблиці, з якими пов'язане представлення. Кожній таблиці можна задати синонім (поле *Alias*), який буде використовуватися при створенні SQL-команди створення представлення.

Закладка *Where* містить три поля – *Where*, *Group By* і *Having*. На основі цієї інформації Erwin генерує SQL-команду створення представлення, причому на основі змісту цих полів генеруються пропозиції SQL-запиту.

Закладка *SQL* містить поле, в якому відображається SQL-запит створення уявлення і вікно вибору *User-Defined SQL*. За замовчуванням опція *User-Defined SQL* вимкнена, і SQL-запит генерується автоматично на основі інформації, занесеної в закладки *Select*, *From* і *Where*.

У закладці *Comment* можна внести коментар для представлення.

Закладка *UDP* дозволяє пов'язати з представленням властивості, визначені користувачем.

Для редагування властивостей колонок представлення служить редактор *View Columns* (Рис. 6.8). Для його виклику слід натиснути правою кнопкою миші по наданню й вибрати в меню пункт *View Columns*.

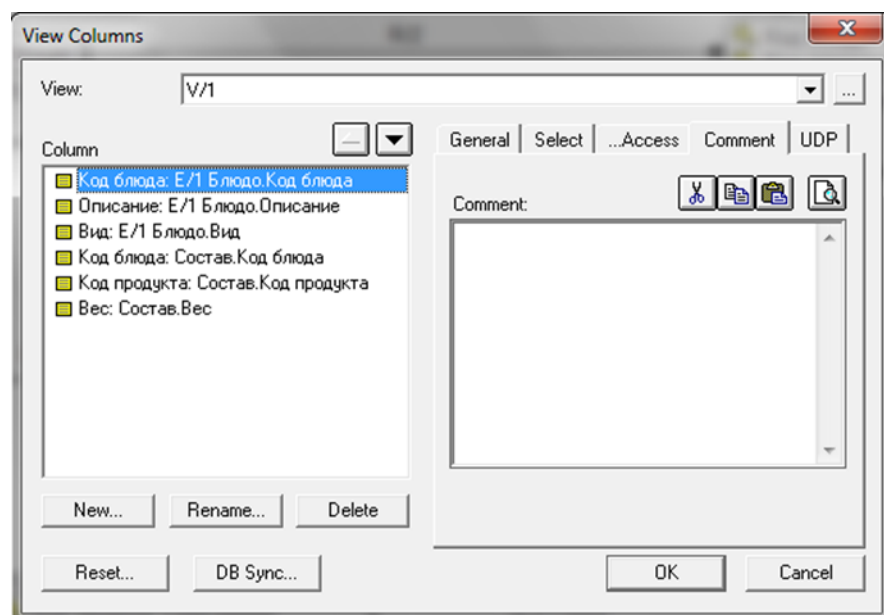


Рисунок 6.8 – Діалог View Columns

Редактор *View Columns* містить наступні закладки:

Закладка *General* дозволяє асоціювати колонку з доменом. За замовчуванням колонка уявлення належить тому ж домену, що і колонка батьківської таблиці.

Закладка *Select*, так само як в діалозі *View Editor* (закладка *Select*, кнопка *New Expression*), тут можна створити вираз (в тому числі включає агрегативної функції) для колонки.


Закладка *Access* використовуються для завдання специфічних властивостей колонок уявлень в *AS / 400* або *Access*.

Закладка *Comment* містить коментар для кожної колонки.

Закладка *UDP* дозволяє зв'язати з колонкою властивості, визначені користувачем.

Правила валідації і значення за замовчуванням

ERwin підтримує правила валідації для колонок, а також значення, що привласнюється колонкам за замовчуванням. Правило валідації задає список допустимих значень для конкретної колонки і/або правила перевірки допустимих значень. Значення за замовчуванням – значення, яке потрібно ввести в колонку, якщо ніяке інше значення не задано явно під час введення даних. З кожною колонкою або доменом можна зв'язати значення за замовчуванням (якщо обрана СУБД підтримує домени).

Якщо клацнути по кнопці , розташованій праворуч від списку *Valid*, з'являється діалог *Model\Validation Rule*, який служить для задання правил валідації (Рис. 6.9).

У ньому можна задати максимальне і мінімальне значення і тип валідації (де перевіряти – на сервері або в клієнтському додатку).

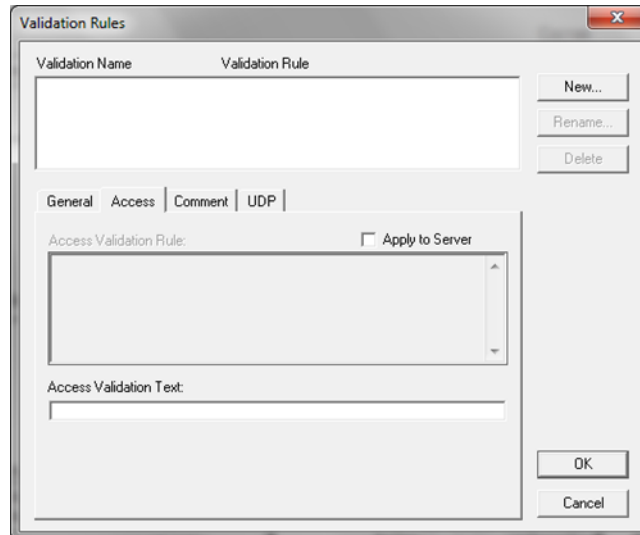


Рисунок 6.9 – Діалог Validation Rule

У верхній частині редактора *Validation Rule* міститься список всіх існуючих правил валідації. Для створення нового правила валідації слід клацнути по кнопці *New*, ввести ім'я правила в поле *Name* діалогу *New Validation* і клацнути по кнопці *OK*. Після цього можна ввести вираз для правила валідації. В закладці *General / Min / Max* поля *Min* і *Max* служать для задання нижньої і верхньої межі діапазону значення.

Кнопка *Valid Value List* викликає редактор *Valid Value List* (Рис.6.10)

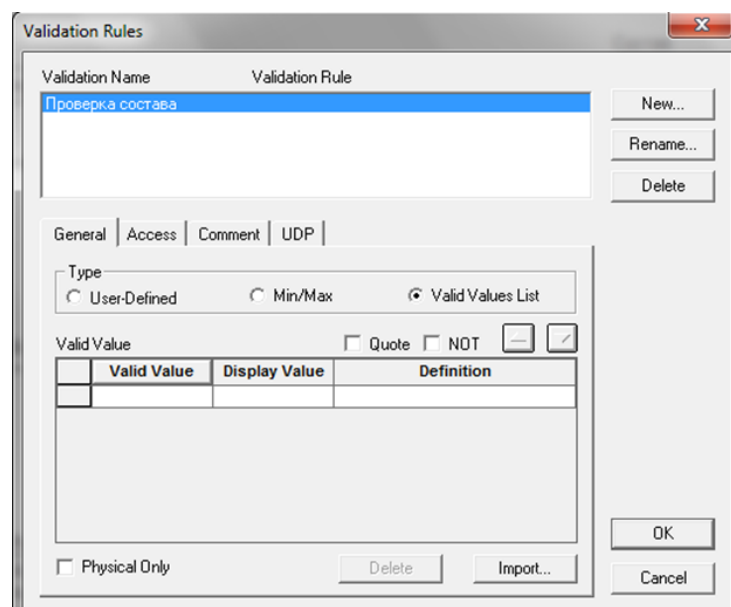


Рисунок 6.10 – Вікно редактора Validation Rule

Редактор *Valid Value List* дозволяє створювати список всіх допустимих значень, які можна зберігати в колонці, і пов'язати його з правилом валідації.

Список, що розкривається у верхній частині редактора *Valid Value* містить всі правила, валідації. Щоб ввести нове значення в список допустимих значень, потрібно клацнути полю і ввести значення. За замовчуванням ERwin створює вираз – команду мови СУБД, використовуючи значення, пов'язані з правилом валідації, і розділяючи значення комами (наприклад, С, D, М). У деяких випадках правила синтаксису СУБД вимагають, щоб кожне значення в команді полягало в одинарні лапки ('С', 'D', 'М'). Щоб автоматично укласти кожне значення в одинарні лапки, потрібно включити опцію *Quote*.

Редактор *Default/Initial Value* дозволяє створити значення, яке автоматично, за умовчанням, присвоюється колонці. Для виклику редактора в пункті меню *Model/Default Value* відкривається діалог *Default / Initial Value* (Рис. 6.11).

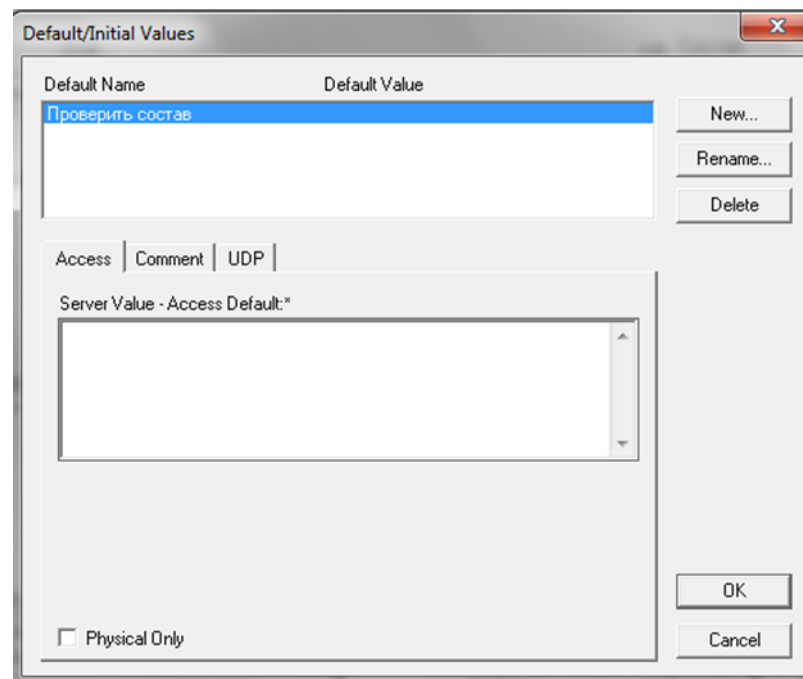


Рисунок 6.11 – Діалог Default/Initial Values

Для створення нового значення за замовчуванням слід клацнути по кнопці *New*, ввести ім'я правила в поле *Name* діалогу *New Default* і клацнути по кнопці *OK*.

У вікні *Default Name* показується список всіх імен значень за замовчуванням. Для видалення і перейменування значення служать відповідно кнопки *Delete* і *Rename*.

Поле *Server Value - Access Default* служить для внесення значення для сервера.

Після створення правила валідації і значення за замовчуванням їх можна присвоїти одній або декільком колонкам або доменам.

Індекси

Щоб вирішити проблему пошуку даних, СУБД використовує особливий об'єкт, званий *індексом*. Він подібний до змісту книги, яка вказує на всі номери сторінок, присвячених конкретній темі. Індекс містить відсортовану по колонці або декільком колонкам інформацію і вказує на рядки, в яких зберігається конкретне значення колонки.

Генерація файлів в БД

Для генерації файлів необхідно перейти з логічної моделі на фізичну. У фізичній моделі натискаємо на кнопку *Select target Server* (Рис. 6.12).

У вікні в закладці *Target Desktop DBMS* виділяємо поле *Access*, в закладці *Access Version* вибираємо версію.

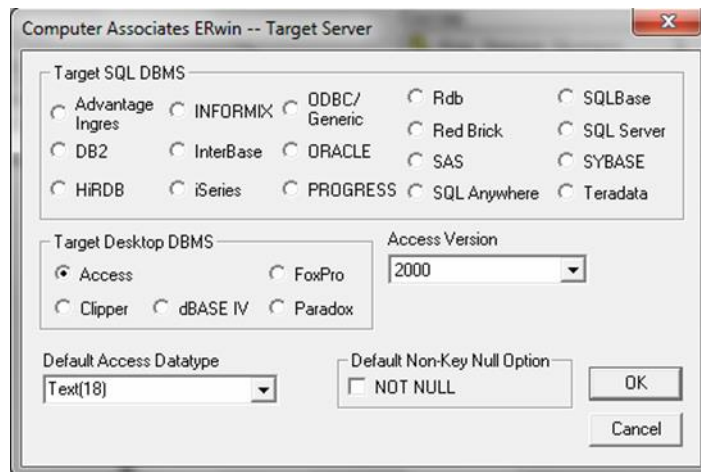



Рисунок 6.12 – Вікно вибору БД для генерації файлу в БД

Для наступної генерації файлу БД потрібно створити за допомогою заданої СУБД (в даному випадку Access) *порожній файл БД*.

За допомогою меню *Database/Database Connection* відкрити вікно діалогу *Access Connection* для введення параметрів зв'язку ER-моделі з БД (Рис. 6.13). В поле *User name* слід ввести ім'я користувача *admin*, потім вибрати за допомогою клавіші *Browse* і діалогового вікна шлях і ім'я файлу БД (вибираємо, заздалегідь створений *ПУСТИЙ*, аксесний файл), далі натиснути клавішу *Connect* буде виконана зв'язок ER-моделі з файлом БД.



Рисунок 6.13 – Вікно діалогу Access Connection

За допомогою меню *Tools / Forward Engineer / Schema Generation* або кнопки *Forward Engineer*  на панелі інструментів відкрити вікно для перевірки і зміни параметрів генерації файлів БД (Рис. 6.14).

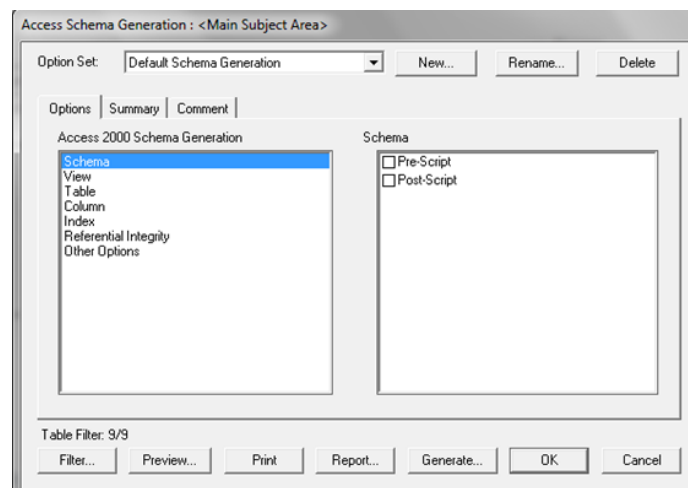



Рисунок 6.14 – Вікно перевірки параметрів генерації файлів в БД

Натиснути кнопку *Generate*; у вікні (Рис. 7.14) будуть виведені оператори, виконані при генерації файлів БД. Далі піде генерація, якщо в моделі будуть помилки, то буде значок . Але натиснувши на *Continue*, продовжиться генерація, поки не буде (Рис. 6.15).

У разі успішного закінчення процесу генерації файлів БД (*Schema Generation Complete* – Генерація схеми виконана) в раніше створеної БД будуть знаходитися відповідні таблиці і схема БД (Рис.6.16-6.17). Якщо в БД раніше була створена схема, то з створеної моделі до БД будуть додані тільки таблиці, і схему даних треба буде модифікувати вручну.

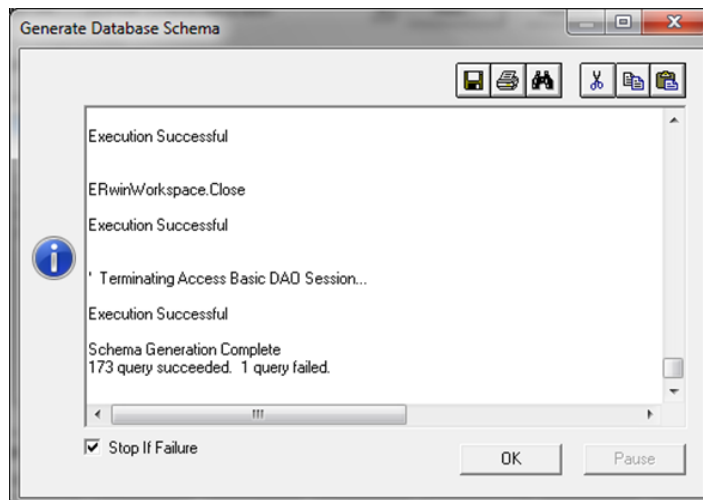


Рисунок 6.15 – Діалог Generation Database Schema

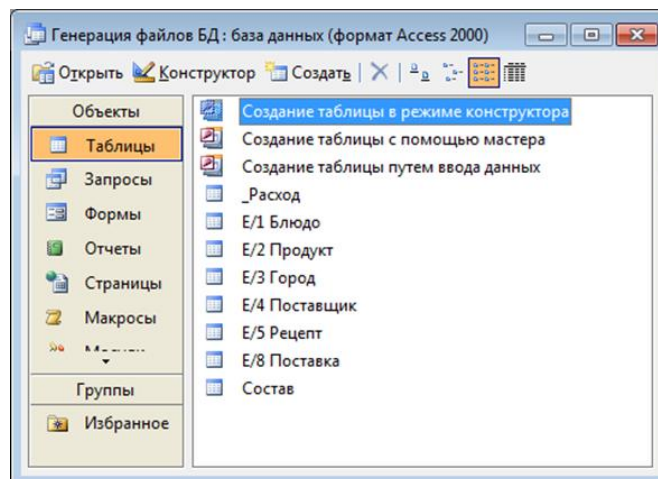


Рисунок 6.16 – Вікно генерації файлів в БД

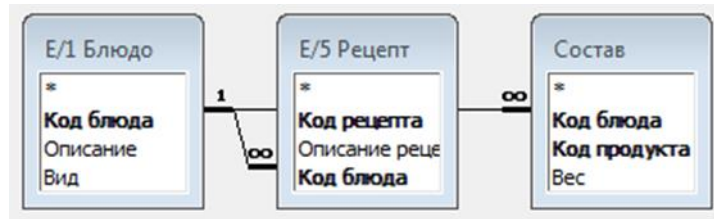


Рисунок 6.17 – Схема БД

ЛАБОРАТОРНА РОБОТА №7

«ПОБУДОВА ДІАГРАМИ ВАРІАНТІВ ВИКОРИСТАННЯ»

Мета роботи

- ознайомитися з методикою побудови UML діаграм
- освоїти прийоми побудови інформаційної моделі предметної області.
- ознайомитися з інтерфейсом програми Rational Rose
- реалізувати за допомогою програми Rational Rose UML діаграму варіантів використання.

Завдання для лабораторної роботи

На основі аналізу предметної області необхідно:

- ознайомитися з інтерфейсом програми Rational Rose;
- побудувати UML діаграму варіантів використання
- визначити дійові особи системи і сценарії поведінки.

Звіт про виконання лабораторної роботи:

1. Тема лабораторної роботи.
2. Відповіді на запитання.
3. Результати виконання завдання в електронному вигляді.

Контрольні питання:

1. З чого складається UML
2. З яких елементів складається модель?
3. У чому сенс варіантів використання?
4. Яке призначення діаграм варіантів використання?
5. Назвіть основні властивості варіантів використання.
6. Назвіть основні компоненти діаграм варіантів використання.
7. Що таке "дійова особа"?

8. Яку роль можуть грати дійові особи по відношенню до варіанту використання?

Література:[2], [6]

Теоретична підготовка до виконання лабораторної роботи

Для діаграм мови UML існують три типи візуальних позначень, які важливі з точки зору укладеної в них інформації:

- зв'язки, які надаються різними лініями на площині;
- текст, що міститься всередині кордонів окремих геометричних фігур;
- графічні символи, зображувані поблизу візуальних елементів діаграм.

При графічному зображенні діаграм рекомендується дотримуватися наступних правил:

- кожна діаграма повинна бути закінченим поданням деякого фрагмента модельованої предметної області;
- представлені на діаграмі сутності моделі повинні бути одного концептуального рівня;
- вся інформація про сутності повинна бути явно представлена на діаграмі;
- діаграми не повинні містити суперечливої інформації;
- діаграми не слід перевантажувати текстовою інформацією;
- кожна діаграма повинна бути самодостатньою для правильної інтерпретації всіх її елементів;
- кількість типів діаграм, необхідних для опису конкретної системи, не є строго фіксованим і визначається розробником;
- моделі системи повинні містити тільки ті елементи, які визначено

Етапи проектування ІС із застосуванням UML

UML забезпечує підтримку всіх етапів життєвого циклу ІС та надає для цих цілей ряд графічних засобів - діаграм.

Етап створення концептуальної моделі

При розробці концептуальної моделі застосовують діаграми варіантів використання і діаграми діяльності, моделі бізнес-об'єктів, діаграми послідовностей.

При цьому, діаграма варіантів використання представляє концептуальну модель системи, яка є вихідною для побудови всіх інших діаграм.

Етап створення логічної моделі ІС

На етапі роботи над логічною моделлю ІС описати вимоги до системи дозволяють діаграми варіантів використання, а при попередньому проектуванні використовують діаграми класів, діаграми станів, діаграми послідовностей.

Діаграма класів є логічною моделлю, що відбиває статичні аспекти структурної побудови системи, а діаграми поведінки, також є різновидами логічної моделі, відображають динамічні аспекти її функціонування.

Етап створення фізичної моделі

Детальне проектування при розробці фізичної моделі виконують із застосуванням діаграм класів, діаграм розгортання, діаграм компонентів.

Діаграми реалізації служать для подання компонентів системи і відносяться до її фізичної моделі.

На рис. 7.1 показані відносини між різними видами діаграм UML. Показники стрілок можна інтерпретувати як відношення «є джерелом вхідних даних для ...» (наприклад, діаграма прецедентів є джерелом даних для діаграм видів діяльності і послідовності).

Наведена схема є наочною ілюстрацією ітеративного характеру розробки моделей з використанням UML.

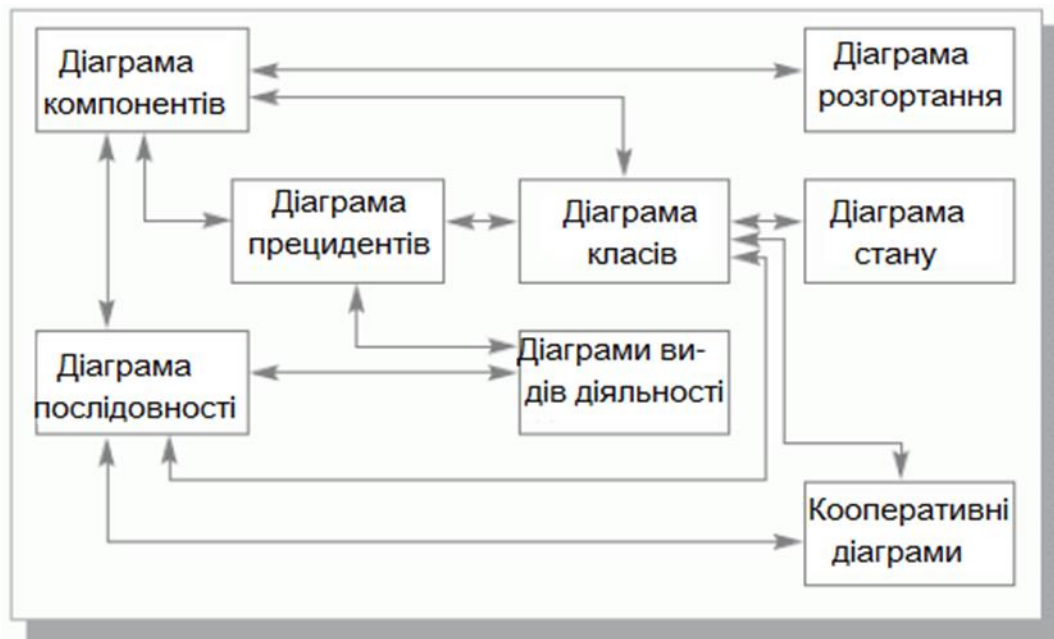


Рисунок 7.1 – Взаємозв'язки між діаграмами UML

Практична підготовка до виконання лабораторної роботи

Опис предметної області «Торгової мережі»

Торгова мережа взаємодіє із зовнішніми об'єктами: постачальниками, банками, покупцями.

Система управління торговою мережею повинна бути орієнтована на роботу наступних категорій користувачів: продавців, менеджерів з продажу, менеджерів по закупівлях, керівників відділів продажів, керівників торгової діяльності, співробітників складу і бухгалтерів.

В системі повинен бути реалізована робота в оперативному режимі, тобто реєстрація операцій продажів в реальному масштабі часу. вихідна інформація - список товарів, які хоче придбати покупець. Вихідна інформація - список дійсно проданих товарів і його вартість. Результат - чек.

Роботи з прайс-листом. Користувач - менеджер з продажу, повинен має можливість формувати і коригувати набори товарів для обговорення їх з клієнтом. Вхідна інформація - список товарів, які хоче придбати покупець. Вихідна інформація - відкоригований список товарів і його вартість. Результат - рахунок клієнта.

Роботи з рахунками клієнтів. Користувач - керівник відділу продажів (або відповідальна особа) повинен мати можливість представити рахунки до виконання або внести корективи. Повинні бути передбачені аналітичні функції для отримання ясною деталізованої оперативної інформації про торговельну діяльність підприємства в розрізі рахунків. Користувач - керівник торговельною мережею. Вхідна інформація - список рахунків, вихідна інформація - виконані рахунки, інформація про боржників і поточних заборгованості.

Формування плану закупівель. Користувач - менеджер із закупівель повинен мати можливість доповнювати або коригувати його для формування остаточного плану закупівель. Вхідна інформація - список рахунків. Вихідна інформація - план закупівель.

Для складського обліку. Користувач - працівник складу. Повинна бути реалізована можливість реєстрації надходження товарів на склад за системою замовлень і поза нею, реєстрація догляду товарів зі складу. Повинні бути аналітичні функції для обліку торгової діяльності в розрізі товарів. Вхідна інформація - список поставок і рахунків на виконання. Вихідна інформація - поточний стан складу і динаміка руху товарів.

Побудова діаграми варіантів використання

Діаграма варіантів використання (Use Case Diagram) являє собою графічне зображення взаємодії деякої сутності (дійової особи) і моделюється. Кожен варіант використання охоплює деяку функцію системи і вирішує деяку дискретну задачу, поставлену сутністю перед системою. Список всіх варіантів використання фактично визначає функціональні вимоги до системи.

Таким чином, діаграма варіантів використання є вихідним концептуальним поданням або концептуальною моделлю системи в процесі її проектування і розробки.

При розробці діаграми варіантів використання повинні бути вирішені наступні завдання:

- повинні бути визначені спільні кордони і контекст предметної області, що моделюється;
- повинні бути сформульовані загальні вимоги до функціонального поведінки проектованої системи;
- повинна бути розроблена вихідна концептуальна модель системи.

Основними елементами діаграми варіантів використання є: *дійова особа, варіант використання, інтерфейс, відношення, примітки.*

Дійова особа

Дійова особа або *актор* (Actor) являє собою деяку зовнішню по відношенню до моделюється системі сутність, яка взаємодіє з системою і використовує її функціональні можливості для досягнення певних цілей. У загальному випадку актор знаходиться поза системою, його внутрішня структура ніяк не визначається.

Дійовій особі ставиться у відповідність безліч логічно пов'язаних ролей, виконуваних при взаємодії з системою. Під *ролю* (Role) розуміється поведінка сутності, яка бере участь в конкретному контексті.

Стандартним графічним позначенням актора на діаграмах є фігурка «чоловічка», під якою записується його ім'я (рис. 7.2).



Рисунок 7.2 – Графічне зображення актора на мові UML

Акторами є сутності, що мають відношення до концептуальної моделі відповідної предметної області: *клієнт, службовець, стільниковий телефон, станція стільникового зв'язку, абонент, автомобіль тощо.* В якості акторів

можуть виступати також інші системи, підсистеми проектованої системи або окремі класи.

Дійові особи взаємодіють з системою за допомогою передачі та прийому повідомлень від варіантів використання. *Повідомлення* являє собою запит актором сервісу від системи і отримання цього сервісу. Крім того, з акторами можуть бути пов'язані інтерфейси, які визначають, яким чином інші елементи моделі взаємодіють з цими акторами.

Варіант використання

Варіант використання (Use Case) являє собою зовнішню специфікацію послідовності дій, які система або інша сутність можуть виконувати в процесі взаємодії з діючими особами.

Варіанти використання призначені для визначення функціональних вимог до системи. Вони служать для опису сервісів, які система надає актору. Іншими словами, кожен варіант використання визначає кінцевий набір дій, який чинять системою при діалозі з актором, а також описує реакції суті системи на отримання окремих повідомлень від дійових осіб і сприйняття цих повідомлень за межами суті. При цьому реалізація самої взаємодії не визначається.

Варіант використання відображається на діаграмі еліпсом (рис. 8.3), всередині якого або під ним розташовується його ім'я (у формі іменника чи дієслова).

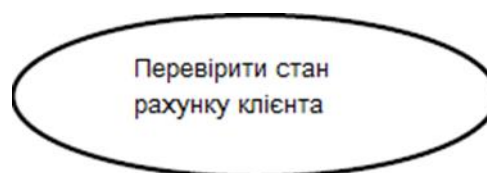


Рисунок 7.3 – Графічне зображення варіанту використання

Інтерфейс

Інтерфейс (Interface) представляє собою безліч іменованих операцій, які характеризують поведінку окремого елемента моделі. Інтерфейси можуть містити тільки операції без вказівки особливостей їх реалізації.

Інтерфейс зображується у вигляді маленького кола, поруч з яким записується його ім'я (рис. 7.4).



Рисунок 7.4 – Графічне зображення інтерфейсів

Формально інтерфейс не тільки відокремлює специфікацію операцій системи від їх реалізації, а й визначає загальні межі проектованої системи. Інтерфейси визначають стикувальні вузли в проектованій системі і дозволяють безболісно модифікувати вже існуючу систему при переході на нові технологічні рішення.

Відношення

Відношення (Relationship) являє собою семантичну зв'язок між окремими елементами моделі. Різні види відносин в тій чи іншій мірі використовуються при побудові всіх діаграм.

У мові UML визначені наступні види відносин між дійовими особами та варіантами використання: *асоціації*, *розширення*, *включення*, *узагальнення*.

Відношення асоціації (Association Relationship) являє собою структурний ставлення, що показує, що об'єкти одного типу деяким чином пов'язані з об'єктами іншого типу. В асоціації можуть зв'язуватися два або більше об'єктів. Тоді асоціація називається відповідно *бінарною* або *n-арною*. Графічно відношення асоціації позначається суцільною лінією між взаємодіючими об'єктами системи (рис. 7.5).

Асоціації може бути присвоєно ім'я (Name), що характеризує природу зв'язку. Асоціація також може *мати кратність* (Multiplicity), що характеризує

загальна кількість конкретних екземплярів даного компонента, які можуть виступати в якості елементів даної асоціації. Кратність вказується поряд з позначенням компонента діаграми, яка є учасником даної асоціації.

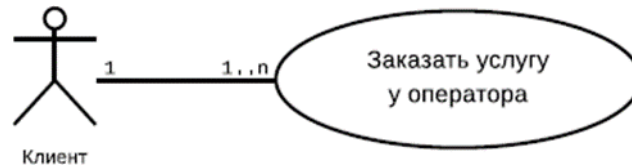


Рисунок 7.5 – Графічне зображення асоціації між актором і варіантом використання з кратністю

Відношення розширення (Extend Relationship) визначає взаємозв'язок базового варіанту використання з іншим варіантом використання, функціональне поведінка якого задіюється базовим не завжди, а тільки при виконанні додаткових умов.

Відношення розширення є відношенням залежності (Dependency Relationship), тобто зв'язком між об'єктами системи, при якій зміна в специфікації одного об'єкта може вплинути на поведінку іншого об'єкта, що використовує перший об'єкт. Залежність зображується переривчастою лінією зі стрілкою, спрямованою до об'єкта, від якого є залежність.

Відношення розширення між варіантами використання позначається, як і залежність, *пунктирною лінією зі стрілкою*, спрямованою від того варіанту використання, який є розширенням для початкового варіанту використання. Дана лінія зі стрілкою позначається ключовим словом «*extend*» (рис. 8.6).

У представленому на рис. 8.6 прикладі при оформленні замовлення послуг у оператора тільки в деяких випадках може знадобитися надання клієнту списку всіх можливих послуг. При цьому умовою розширення є запит від клієнта на отримання списку можливих послуг.

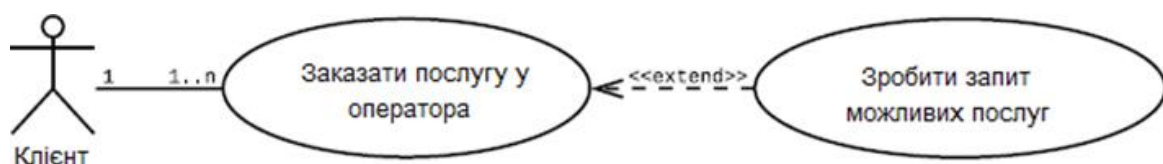


Рисунок 7.6 – Графічне зображення відношення розширення

Відношення включення (Include Relationship) також є різновидом відносини залежності, але встановлюється тільки між двома варіантами використання і вказує на те, що заданий поведінка для одного варіанта використання включається як складовий фрагмента в послідовність поведінки іншого варіанту використання. Графічно дане відношення позначається як відношення залежності у формі пунктирною лінії зі стрілкою, спрямованої від базового варіанту використання до такого, що включається варіанту використання, із зазначенням ключового слова «Include» (рис. 7.7).

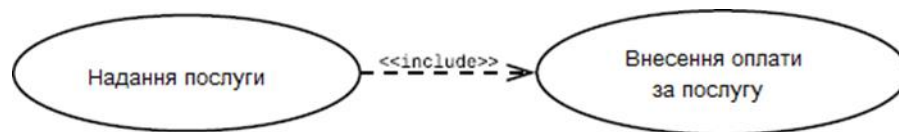


Рисунок 7.7 - Графічне зображення відношення включення

Відношення узагальнення (Generalization Relationship) являє собою зв'язок між загальною сутністю, званої батьком, і більш спеціалізованої різновидом цієї сутності, званої нащадком. Нащадок успадковує всі властивості і поведінку свого батька, але нащадок може мати власну поведінку і властивості.

В UML допускається і множинне спадкування, коли один об'єкт-нащадок визначається на основі декількох батьківських об'єктів. Графічно дане відношення узагальнення позначається суцільною лінією зі стрілкою в формі незафарбованого трикутника, що вказує на батьківський об'єкт (рис. 7.8).

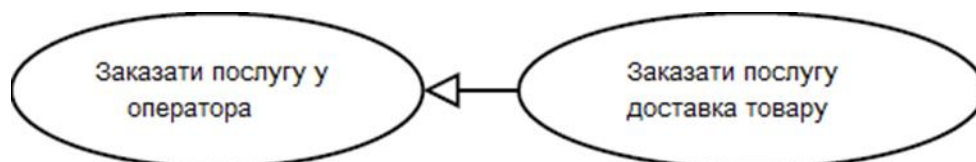


Рисунок 7.8 - Графічне зображення відношення узагальнення

Розробка діаграм варіантів використання в середовищі Rational Rose

У CASE-засобі IBM Rational Rose реалізовані загальноприйняті стандарти на робочий інтерфейс програми, аналогічно відомим середовищах візуального програмування.

Робочий інтерфейс програми IBM Rational Rose складається з різних елементів, основними з яких є (рис. 7.9):

- головне меню;
- стандартна панель інструментів;
- спеціальна панель інструментів;
- вікно браузера проекту;
- робоча область зображення діаграми або вікно діаграми;
- вікно документації;
- вікно журналу.

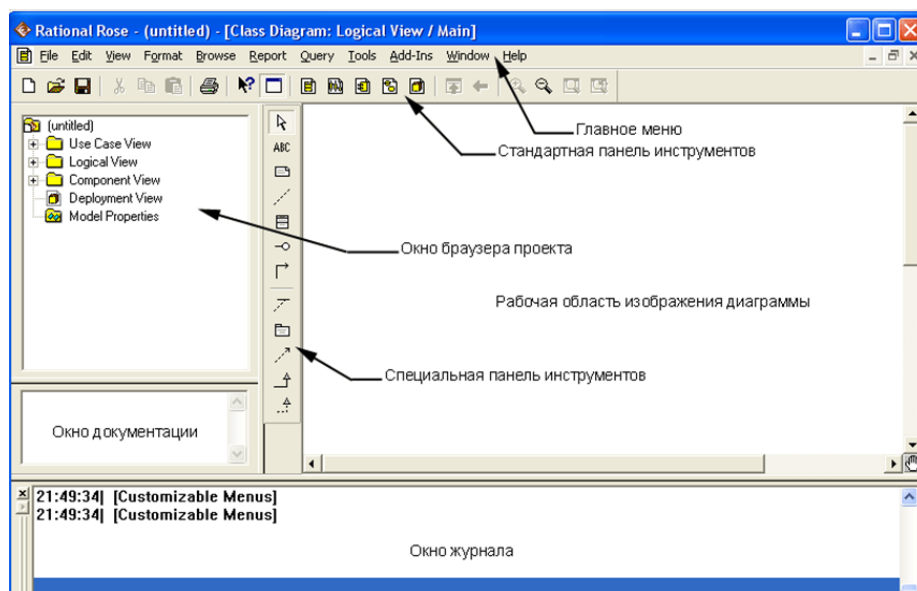


Рисунок 7.9 – Загальний вид робочого інтерфейсу CASE-засобу IBM Rational Rose

Головне меню програми IBM Rational Rose 2003 виконано в загальноприйнятому стандарті і має наступний вигляд (рис. 7.10).

File Edit View Format Browse Report Query Tools Add-Ins Window Help

Рисунок 7.10 - Зовнішній вигляд головного меню програми

Стандартна панель інструментів розташовується нижче рядка головного меню і має такий вигляд (рис. 7.11).



Рисунок 7.11 - Зовнішній вигляд стандартної панелі інструментів

Користувач може налаштувати зовнішній вигляд цієї панелі на свій розсуд. Для цього необхідно виконати операцію головного меню: *Tools Options* (Інструменти Параметри), відкрити вкладку *Toolbars* (Панелі інструментів) діалогового вікна і натиснути кнопку *Standard* (Стандартна). У додатково відкритому вікні можна переносити необхідні кнопки з лівого списку в правий список, а непотрібні кнопки – з правого списку в лівий. Даним способом можна показати або приховати різні кнопки інструментів, а також змінити їх розмір. Призначення окремих кнопок стандартної панелі інструментів наведена далі при розгляді операцій головного меню.

Ієрархічне представлення структури кожного проекту, що розробляється організовано у формі наступних представлень:

Use Case View – представлення варіантів використання, в якому містяться діаграми варіантів використання і їх реалізації у вигляді варіантів взаємодії;

Logical View – логічне уявлення, в якому містяться діаграми класів, діаграми станів і діаграми діяльності;

Component View – представлення компонентів, в якому містяться діаграми компонентів моделі, що розробляється;

Deployment View – представлення розгортання, в якому міститься єдина діаграма розгортання розробляється моделі.

Призначення операцій головного меню

Робочий інтерфейс засобу IBM Rational Rose має головне меню, яке дозволяє користувачеві завантажувати і зберігати інформацію в зовнішніх файлах, змінювати зовнішній вигляд елементів графічного інтерфейсу, викликати довідкову інформацію, викликати інші діалогові вікна для роботи з програмою IBM Rational Rose.

Операції головного меню *File (Файл)* дозволяють створювати нові моделі в нотації мови UML, завантажувати і зберігати розробляється модель в зовнішньому файлі, роздруковувати на принтері розроблені діаграми.

Операції головного меню *Edit (Редагування)* дозволяють виконувати дії по редагуванню елементів моделі і їх властивостей, а також виконувати пошук елементів в рамках проекту, що розробляється.

Операції головного меню *View (Вид)* дозволяють відображати на екрані різні елементи робочого інтерфейсу і змінювати графічне представлення діаграм.

Операції головного меню *Format (Формат)* дозволяють виконувати дії зі зміни зовнішнього вигляду елементів моделі на різних діаграмах.

Операції головного меню *Browse (Огляд)* дозволяють відображати робочі вікна з різними канонічними діаграмами розробляємої моделі і викликати діалогові вікна редагування властивостей окремих елементів моделі.

Діаграми варіантів використання і редагування властивостей її елементів

Робота над моделлю в середовищі IBM Rational Rose починається з загального аналізу проблеми і побудови діаграми варіантів використання. Для новостворюваного проекту можна скористатися майстром типових проектів, якщо він встановлений в даній конфігурації. Майстер типових проектів доступний з меню *File New (Файл Новий)*.

Для розробки діаграми варіантів використання моделі в середовищі IBM Rational Rose необхідно активізувати відповідну діаграму у вікні діаграми. Це можна зробити наступними способами:

- розкрити уявлення варіантів використання Use Case View в браузері проекту і двічі клацнути на піктограмі Main (Головна);
- за допомогою операції головного меню *Browse Use Case Diagram* (Браузер Діаграма варіантів використання).

При цьому з'являється нове вікно з чистим робочим листом діаграми варіантів використання і спеціальна панель інструментів, що містить кнопки із зображенням графічних елементів, необхідних для розробки діаграми варіантів використання. Призначення окремих кнопок даної панелі можна дізнатися також з спливаючих підказок, які з'являються, якщо підвести і затримати на деякий час покажчик миші над тією чи іншою кнопкою.

На спеціальній панелі інструментів за замовчуванням присутній тільки частина кнопок з піктограмами елементів, які можуть бути використані для побудови діаграми. Додати кнопки з піктограмами інших графічних елементів, наприклад, таких як бізнес-варіант використання (*business use case*), бізнес-актор (*business actor*), співробітник (*business worker*), або видалити непотрібні кнопки можна за допомогою настройки спеціальної панелі інструментів.

Відкрити діалогове вікно налаштування спеціальних панелей інструментів для діаграм в середовищі IBM Rational Rose можна за допомогою операції головного меню: *Tools Options* (Інструменти Параметри), розкривши вкладку *Toolbars* (Панелі інструментів) і натиснувши відповідну кнопку (наприклад, *Use Case diagram*) в групі опцій *Customize Toolbars* (Налаштування панелей інструментів). Це вікно налаштування також можна відкрити за допомогою операції контекстного меню *Customize* (Налаштування) при позиціонуванні курсору на спеціальній панелі інструментів (рис.7.12).

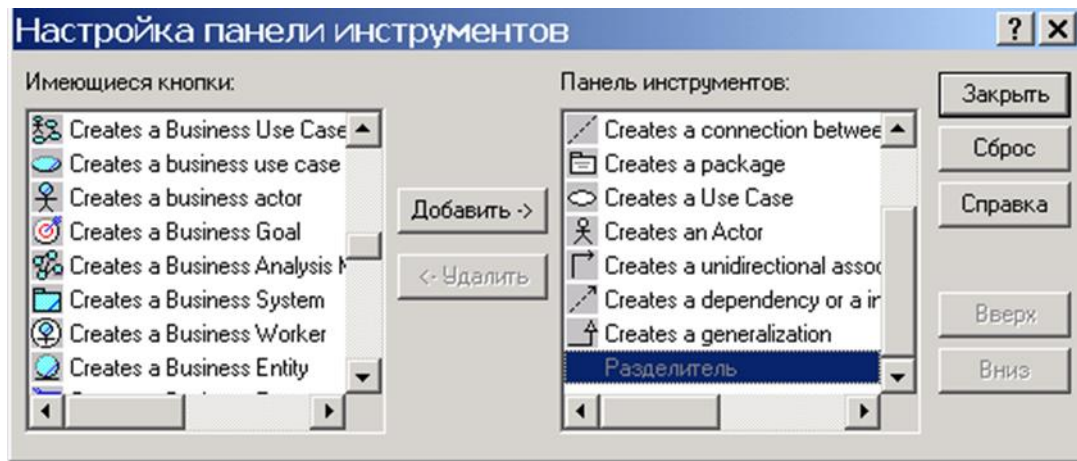


Рисунок 7.12 - Диалогове вікно настройки спеціальної панелі інструментів для діаграми варіантів використання

Для додавання необхідних кнопок на панель слід виділити їх в лівому вікні зі списком піктограм графічних елементів, після чого натиснути кнопку *Додати* в центрі діалогового вікна. Для видалення непотрібних кнопок з панелі інструментів слід виділити їх в правому вікні зі списком піктограм графічних елементів, після чого натиснути кнопку *Видалити* в центрі діалогового вікна. Для відновлення набору піктограм за замовчуванням можна натиснути кнопку *Скидання*. Після настройки спеціальної панелі інструментів відповідне вікно слід закрити натисканням на кнопку *Закрити*.

Додавання актора на діаграму варіантів використання і редагування його властивостей

Для додавання актора на діаграму варіанту використання потрібно за допомогою лівої кнопки миші натиснути кнопку із зображенням піктограми актора на спеціальній панелі інструментів, відпустити ліву кнопку миші і клацнути лівою кнопкою миші на вільному місці робочого листа діаграми. На діаграмі з'явиться зображення актора з маркерами зміни його геометричних розмірів і запропонованим програмою ім'ям за умовчанням *NewClass*. Для розроблюваної моделі Торгової мережі запропоноване програмою ім'я актора слід змінити на Менеджер з продажу(рис. 7.13).

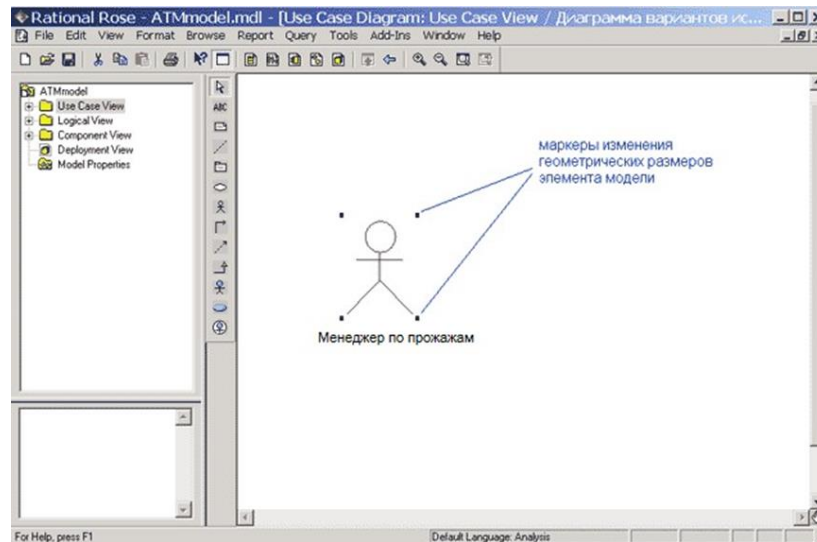


Рисунок 7.13 - Діаграма варіантів використання після додавання на неї актора

Щоб змінити розташування зображення графічного елемента моделі, слід клацанням лівої кнопки миші виділити його в робочій області діаграми, і, не відпускаючи лівої кнопки, перемістити в потрібне місце діаграми.

Ім'я розміщеного на діаграму елемента розробник може змінити або відразу після додавання елемента на діаграму, або в ході подальшої роботи над проектом. Для будь-якого графічного елемента моделі по кліку правої кнопкою миші на обраному елементі викликається контекстне меню даного елемента, серед операцій якого є пункт *Open Specification* (Відкрити специфікацію). У цьому випадку з'являється додаткове діалогове вікно зі спеціальними вкладками, в поля введення яких можна занести всю інформацію по даному елементу. Для доданого актора Менеджер по продажу вікно специфікації властивостей виглядає наступним чином (рис. 7.14).

Слід зазначити, що відкрити діалогове вікно специфікації властивостей будь-якого елемента моделі можна також подвійним клацанням лівою кнопкою миші на графічному зображенні цього елемента на діаграмі.

Для актора Менеджер по продажу можна уточнити його призначення в моделі. З цією метою слід 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: «Фізична особа, яка обслуговує торгову мережу» і натиснути кнопку *Apply* (Застосувати) або *OK*.

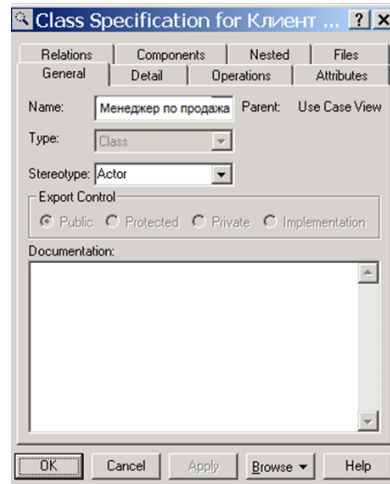


Рисунок 7.14 - Діалогове вікно специфікації властивостей актора Менеджера по продажу

Після зміни даних властивостей актора Менеджера по продажу вікно специфікації властивостей буде виглядати наступним чином (рис. 7.15).

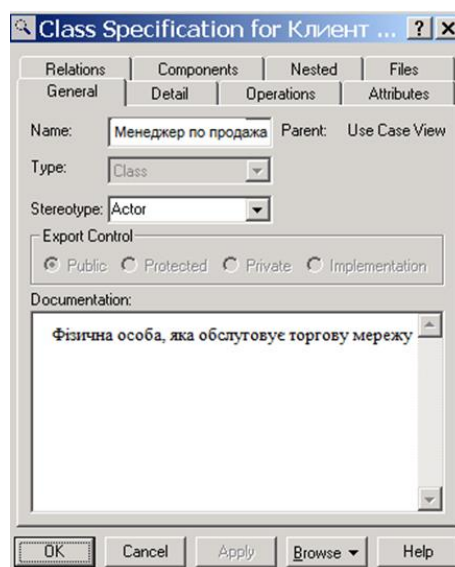


Рисунок 7.15 - Діалогове вікно специфікації властивостей після зміни стереотипу і додавання тексту документації для актора Менеджер по продажу

Додавання та редагування варіанту використання

Для додавання варіанту використання на діаграму потрібно за допомогою лівої кнопки миші натиснути кнопку із зображенням варіанти використання на спеціальній панелі інструментів, відпустити ліву кнопку миші і клацнути лівою кнопкою миші на вільному місці діаграми.

На діаграмі з'явиться зображення варіанти використання з маркерами зміни його геометричних розмірів і запропонованим програмою ім'ям за умовчанням *NewUseCase*.

Для розроблюваної моделі Торгової мережі запропоноване програмою ім'я варіанти використання слід змінити на Видача заказу (рис.7.16).

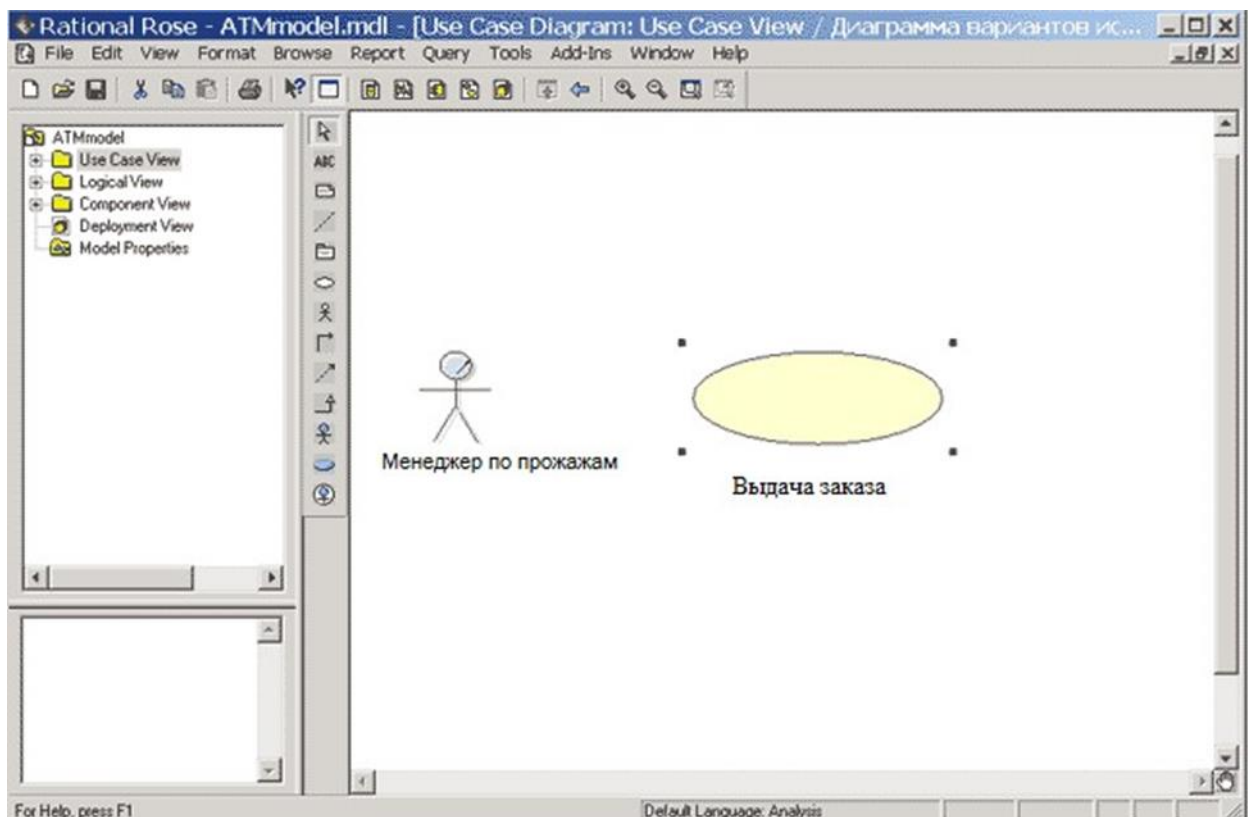


Рисунок 7.16 - Діаграма варіантів використання після додавання на неї варіанту використання

Додавання асоціації

Для додавання асоціацію між актором і варіантом використання на діаграму потрібно за допомогою лівої кнопки миші натиснути на спеціальній панелі інструментів кнопку з зображенням піктограми спрямованої асоціації, відпустити ліву кнопку миші, клацнути лівою кнопкою миші на зображенні актора на діаграмі і відпустити її на зображенні варіанти використання. В результаті цих дій на діаграмі з'явиться зображення асоціації, що з'єднує актора з варіантом використання (рис. 7.17).

При необхідності можна зробити спрямовану асоціацію ненаправленою, для чого слід скористатися діалоговим вікном властивостей асоціації. Відкрити це вікно можна, наприклад, двічі клацнувши на зображенні лінії асоціації на діаграмі, після чого прибрати позначку рядка вибору Navigable (Навігація) на вкладці Role A Detail (Детальні властивості кінцевий точки асоціації A).

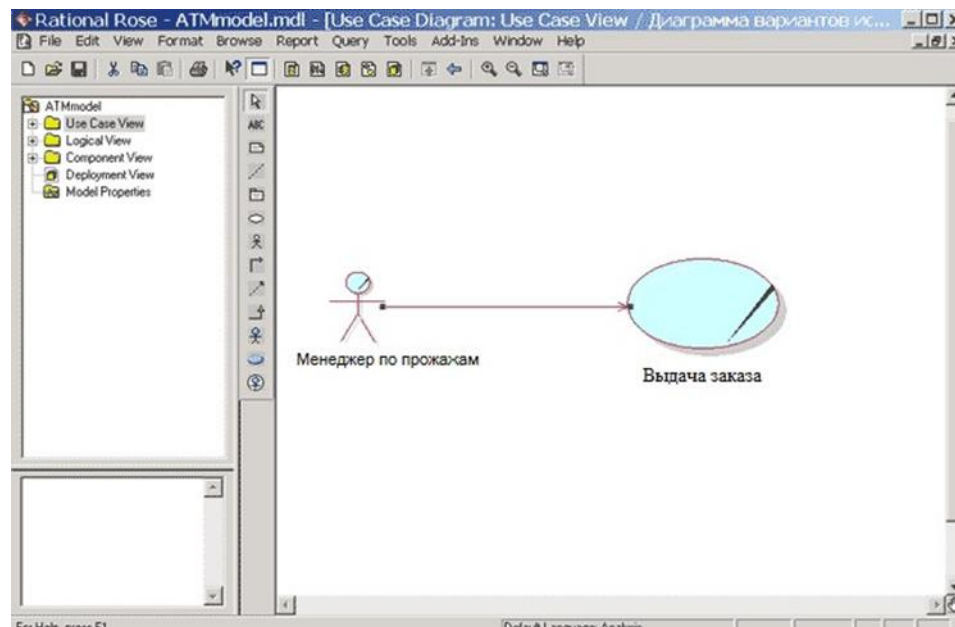


Рисунок 7.17 - Діаграма варіантів використання після додавання на неї спрямованої асоціації

Додавання відносини залежності і редагування його властивостей

Для додавання відносини залежності між двома варіантами використання на діаграму необхідно попередньо розглянути вище способом

додати другий варіант використання з ім'ям *Видача заказу*. Після цього за допомогою лівої кнопки миші натиснути кнопку із зображенням піктограми залежно на спеціальній панелі інструментів, відпустити ліву кнопку миші, клацнути лівою кнопкою миші на зображенні варіанти використання *Оформлення заказу на товар* і відпустити її на зображенні варіанти використання формування заказу. В результаті цих дій на діаграмі з'явиться зображення відносини залежності, яке з'єднує два обраних варіанти використання.

Оскільки варіант використання *Оформлення заказу* виконується завжди, для доданого відносини залежності додатково слід вказати текстовий стереотип `<< include >>`. Виконати це можна вже відомим способом за допомогою діалогового вікна специфікації властивостей цього відносини і вибору потрібного стереотипу із запропонованого списку.

Після задання для даного відносини залежності стереотипу `<< include >>` текст цього стереотипу в кутових дужках з'явиться поруч із зображенням пунктирною лінією залежності, що зв'язує відповідні варіанти використання (рис. 7.18). З метою кращої візуалізації діаграми текстову область стереотипу можна перемістити в потрібне місце діаграми. Виконати це можна за допомогою загального способу переміщення графічних елементів моделі.

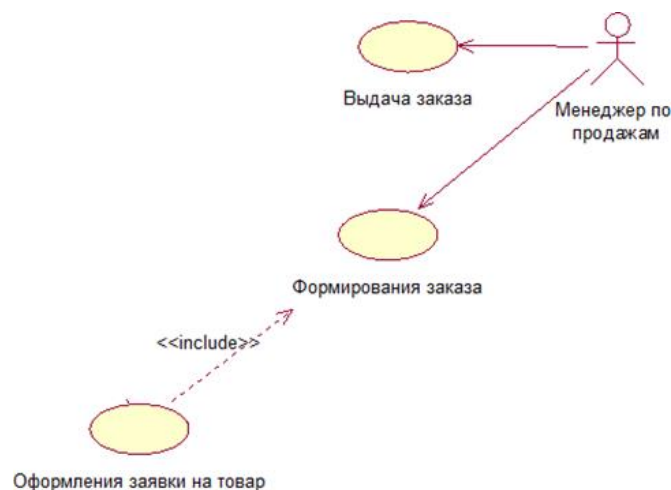


Рисунок 7.18 - Діаграма варіантів використання після додавання на неї відносини залежності

Аналогічним чином можуть бути додані на діаграму варіантів використання відносини залежності зі стереотипом << *extend* >>, які застосовуються для моделювання винятків при виконанні окремих варіантів використання.

Остаточна побудова діаграми варіантів використання

До окремого варіанту використання можна додати текстовий файл з описом сценарію його виконання. Для цього необхідно виділити цей варіант використання в браузері проекту і виконати операцію контекстне меню: *New File* (Новий Файл). В результаті цього буде викликано стандартне вікно відкриття файлу, в якому необхідно задати ім'я попередньо створеного за допомогою офісної програми *MS Word* додається файлу. Після натискання кнопки *Відкрити піктограма доданого файлу* з'явиться в браузері проекту нижче відповідного варіанту використання.

В подальшому можна повернутися до редагування цього файлу сценарію, виконавши подвійне клацання на цій піктограмі. При цьому файл сценарію буде відкритий у відповідній програмі □ в текстовому процесорі *MS Word*.

Діаграма варіантів використання є високорівневим концептуальним поданням моделі, тому вона не повинна містити надто багато варіантів використання і акторів. В подальшому побудована діаграма може бути змінена за допомогою додавання нових елементів, таких як варіанти використання і актори, або їх видалення.

Для видалення будь-якого графічного елементу з діаграми його слід виділити на діаграмі і натиснути клавішу *Delete* на клавіатурі. При цьому виділений елемент буде видалений з активною діаграмою, але не з моделі. Для видалення елемента не тільки з діаграмою, а й з моделі проекту необхідно виділити видаляється елемент на діаграмі і скористатися операцією головного меню *Edit Delete from Model* (Редагування Видалити з моделі). Для цієї ж мети служить комбінація клавіш швидкого доступу: *Ctrl + D*.

Після закінчення сеансу роботи над проектом виконану роботу необхідно зберегти в файлі проекту з розширенням «.MDL». Це можна зробити через меню *File Save* (Файл Зберегти) або *File Save As* (Файл Зберегти як). При цьому вся інформація про проект, включаючи діаграми і специфікації елементів, буде збережена в одному файлі.

Побудова діаграми варіантів Торгової мережі.

Побудуємо діаграму варіантів використання Торгової мережі. Подібний вид діяльності виконаємо в такій послідовності:

- 1) Визначення дійових осіб.
- 2) Визначення прецедентів.
- 3) Складання опису кожного прецеденту.
- 4) Опис моделі прецедентів в цілому.

Оформимо вимоги в табл.7.1. У ліву колонку поміщаємо варіанти використання, а в праву - дійових осіб, що беруть участь в варіанті використання. Отже, можна виділити наступних акторів. Покупець буде здійснювати запит до системи з метою визначення потрібного йому товару. Керівник контролює роботу співробітників, контролює і встановлює ціни, формує каталог товарів продукції.

Менеджер з продажу буде перевіряти за запитом покупця, який товар його цікавить, чи є наявний товар у вільному продажу або на складі, виписує документи на оплату і заносить інформацію про оплату товару. Оформляє заявку на товар і отримане замовлення покупцеві.

Менеджер з постачання співпрацює з постачальниками, оформляє заявку на поставку товару постачальникам, поставку товару.

На підставі потреб виділяємо варіанти використання.

Варіант використання «Формування замовлення» дозволяє менеджеру з продажу скласти заявку на замовлений товар, включаючи при необхідності їх доставку.

Варіант використання «Видача замовлення» дозволяє менеджеру з продажу видати товар торгової мережі.

Таблиця 7.1 – Вимоги до системи

Варіант використання	Діючі особа
1	2
Замовлення товар	Торгова мережа
Видача чеків	
Прийом на роботу	
Формування цін на товар	
Формування замовлення	Менеджер з продажу
Видача замовлення	
Введення даних про клієнтів	
Формування звітів	
Оформлення поставок товару	Менеджер з постачання
Оформлення заявки на товар	
Оформлення доставки	
Поставка товару	Постачальник
Формування списку товарів	
Довідник по товарах	Каталог
Формування каталогу товарів	
Вибір способу оплати	Покупець
Вибрати товар	
Замовити товар	
Облік приходу поставок	Склад
Реалізація товарів зі складу	
Оформлення звітів	

Варіант використання «Оформлення заявки на товар» дозволяє менеджеру з продажу на підставі сформованого замовлення від торгової мережі і цін на товари сформувавши заявку на поставку товару постачальникам.

Варіант використання «Оформлення поставок товару» дозволяє менеджеру з продажу отримати товар і оформити необхідну документацію.

Варіант використання «Оформлення доставки» дозволяє менеджеру з продажу оформити доставку товару від постачальників до торгової мережі.

Варіант використання «Формування списку замовлення» дозволяє постачальнику оформити заявку на товар відповідно до наявності продукції.

Варіант використання «Поставка товару» здійснюється поставка товару торгової мережі.

Варіант використання «Замовлення товару» дозволяє покупцеві відповідно до наданого каталогом товарів замовити вибраний товар.

Варіант використання «Вибрати спосіб оплати» дозволяє покупцеві вибрати спосіб оплати із запропонованих варіантів.

Варіант використання «Вибрати товар» дозволяє покупцеві вибрати товар.

Варіант використання «Каталог товарів» надає перелік послуг і товарів торгової мережі.

Останнім елементом діаграми прецедентів є асоціативні відносини між акторами і варіантами використання.

Виходячи з умови задачі, обраних акторів і варіантів використання, можна визначити відносини:

Актор покупець пов'язаний з наступними варіантами використання: вибір способу оплати, вибрати товар, при цьому він пов'язаний з каталогом товару і замовити товар, при цьому він пов'язаний з торговою мережею;

Актор менеджер з продажів пов'язаний з варіантами використання: видачею замовлення і формуванням замовлення;

Актор менеджер по поставкам пов'язаний з варіантами використання: оформленням поставок на товар, оформленням доставки через варіант використання формування замовлення, оформленням заявки на товар через варіант використання формування цін на товари;

Актор постачальник пов'язаний з варіантом використання формування списку замовлення через варіант використання оформлення доставки і формування замовлення і варіантом використання поставка товару;

Актор керівник пов'язаний з варіантом використання формування цін на продукти, прийом співробітників на роботу і каталог товарів;

Актор покупець пов'язаний з варіантом використання вибрати товар через варіант використання каталог товарів, вибрати спосіб оплати і замовити товар в торговій мережі.

Варіант використання каталог товарів включає в себе через ставлення включення формування цін на товари.

Варіант використання вибрати товар включає в себе через ставлення включення каталог товарів.

Варіант використання оформлення заявки на товар включає в себе через ставлення включення формування цін на товари і формування замовлення.

Варіант використання формування списку замовлення включає в себе через ставлення включення оформлення доставки.

Варіант використання оформлення доставки включає в себе ставлення формування замовлення.

Далі створюємо діаграму варіантів використання для інших користувачів системи (рис.7.19 – 7.21).

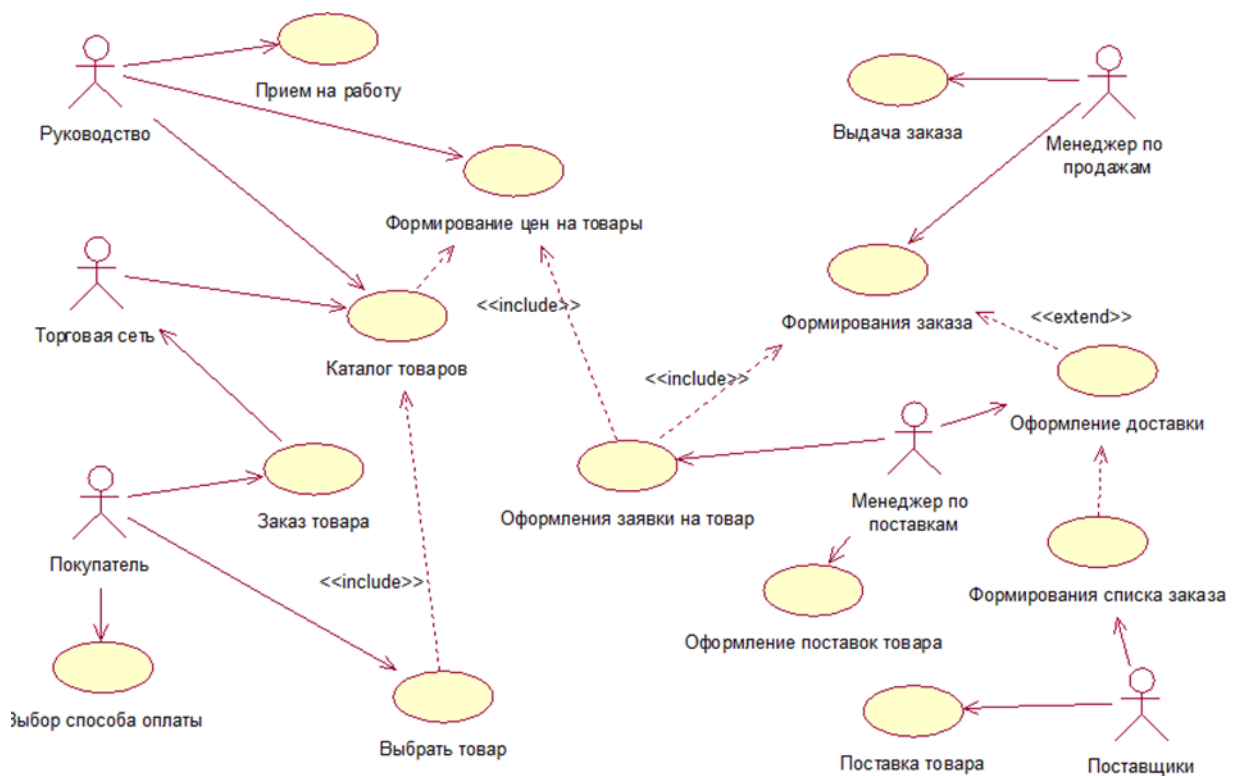


Рисунок 7.19 – Діаграма варіантів використання «Торгової мережі»

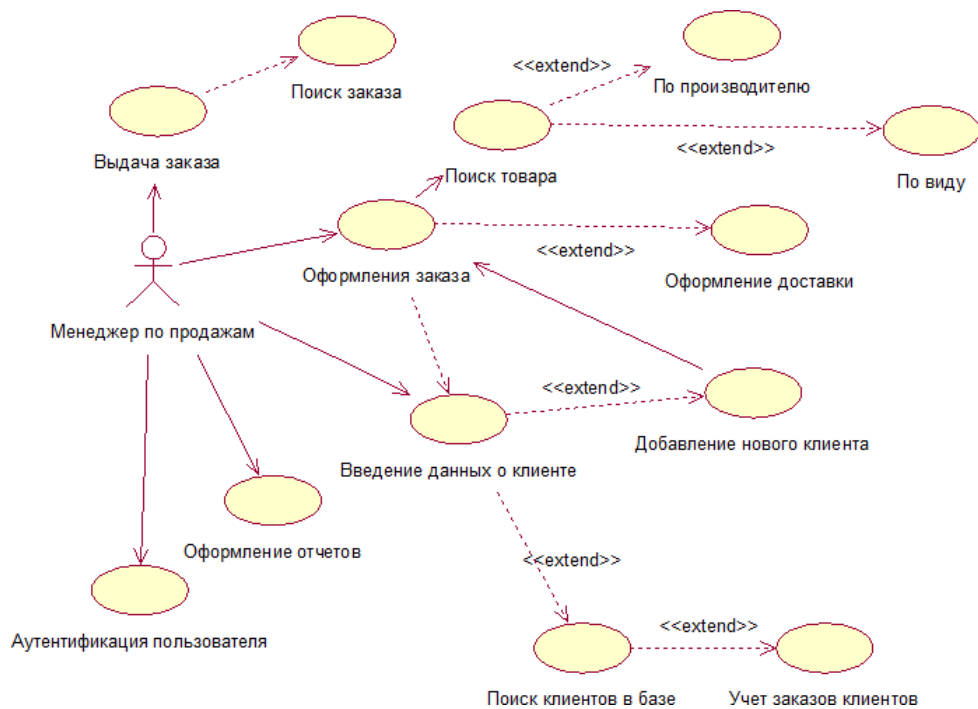


Рисунок 7.20 – Диаграмма вариантов использования «Менеджера з продажу»

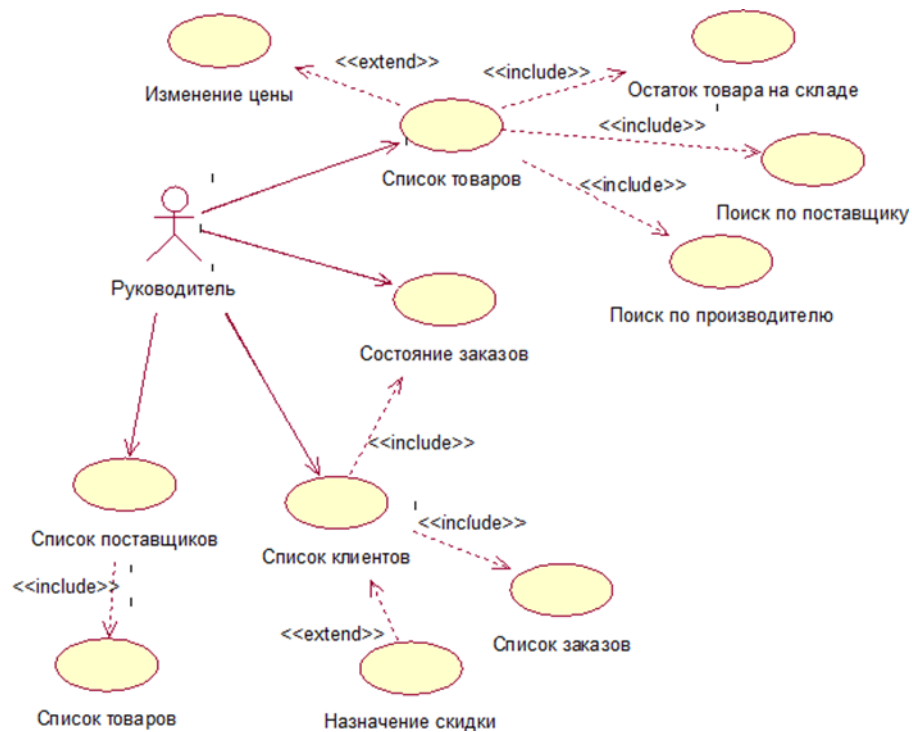


Рисунок 7.21 – Диаграмма вариантов использования «Керівник торгової мережі»

ЛАБОРАТОРНА РОБОТА № 8

«ВИЗНАЧЕННЯ СКЛАДОВИХ ЕЛЕМЕНТІВ ПРОЕКТУ»

Мета роботи: Виконати дослідження характеристик системи, її складових робіт для побудови проекту системи. визначити елементи та характеристики проекту, необхідних для управління ним.

Завдання для лабораторної роботи

- відповісти на запитання;
- визначити тип проекту та особливості управління ним;
- сформулювати мету проекту;
- виділити фази етапів життєвого циклу проекту;
- сформулювати список задач проекту;
- визначити учасників проекту.

Звіт про виконання лабораторної роботи:

1. Тема лабораторної роботи.
2. Відповіді на запитання.
3. Результати виконання завдання в електронному вигляді.

Контрольні питання:

1. Які складові проекту Ви знаєте?
2. Дайте визначення життєвому циклу проекту. Опишіть структуру життєвого циклу проекту.
3. Опишіть фази ЖЦ проекту. Якими параметрами характеризується проект?
4. Що таке задача проекту? Якими параметрами характеризується задача проекту?
5. Що таке ресурс? Типи ресурсів для виконання проекту.
6. Якими процесами описується фази життєвого циклу проекту?

Література:[2], [4]

Теоретична підготовка до виконання лабораторної роботи

Основи планування при управлінні проектом

Проектом називається сукупність розподілених у часі заходів або робіт, спрямованих на досягнення поставленої мети.

Проект завжди має чітко визначену мету, яка виражається в отриманні деякого результату. Досягнення цього результату означає успішне завершення і закінчення проекту

Проект має чітко окреслений початок, який збігається з початком першої роботи, спрямованої на досягнення поставленої мети. Початок може здаватися директивним, або розраховуватися в результаті складання плану робіт по проекту.

Проект має чітко окреслений кінець, який збігається з кінцем останньої роботи, спрямованої на отримання заданого результату. Як і початок, кінець проекту може здаватися директивним, або розраховуватися при складанні плану робіт.

Проект виконується командою, до складу якої входить керівник проекту, менеджери, виконавці. Крім основної команди в ньому можуть брати участь сторонні виконавці, команди і організації, які залучаються на тимчасовій основі для виконання окремих робіт.

При реалізації проекту використовуються матеріальні ресурси. Їх номенклатура і кількість визначаються характером проекту та що до нього робіт.

Проект має бюджет. Вартість проекту складається з вартості витрачених матеріальних ресурсів, витрат з оплати праці реалізує його команди і інших витрат, пов'язаних з особливостями конкретних видів робіт.

Життєвий цикл проекту

Поняття життєвого циклу проекту є одним з найважливіших для менеджера, оскільки визначення саме поточної стадії впровадження проекту

формує задачі і види діяльності керівника проекту, методику й інструментальні засоби, що ним мають бути використані.

Життєвий цикл проекту - це проміжок часу між моментами його початку і завершення.

Структура ЖЦП відображена на рис.8.1

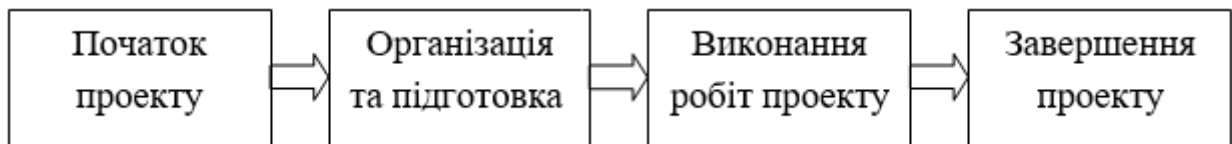


Рисунок 8.1 – Структура ЖЦП

Початок проекту. Включає формулювання цілей, аналіз інвестиційних можливостей, обґрунтування здійсненності (техніко-економічне обґрунтування) і планування проекту.

Організація та підготовка. Включає визначення структури робіт і виконавців, побудова календарних графіків робіт, бюджету проекту, розробку проектно-кошторисної документації, переговори та укладення контрактів з підрядниками і постачальниками.

Виконання робіт проекту. Включає роботи з реалізації проекту, у тому числі будівництво, маркетинг, навчання персоналу тощо

Завершення проекту. Включає в загальному випадку приймальні випробування, дослідну експлуатацію і здачу проекту в експлуатацію.

Результат проекту - це деяка продукція або корисний ефект, створювані в ході реалізації проекту. В якості результату, залежно від мети проекту, можуть виступати: наукова розробка, новий технологічний процес, програмний засіб, будівельний об'єкт, реалізована навчальна програма, реструктурована компанія, сертифікована система якості і т.д.

Про успішність проекту судять по тому, наскільки його результат відповідає за своїми витратним, дохідним, інноваційним, якісним,

тимчасовим, соціальним, екологічним та іншим характеристикам запланованому рівню.

Основою планування проектів є складання плану і відстеження виконання робіт по ньому.

Проект робиться для досягнення певного результату у визначені строки та за певні гроші. План проекту складається для того, щоб визначити, за допомогою яких робіт буде досягтися результат проекту, які люди і обладнання потрібні для виконання цих робіт і в який час ці люди та обладнання будуть зайняті роботою по проекту. Тому проектний план містить три основні елементи: задачі, ресурси та призначення.

Задачами називається робота, здійснювана в рамках проекту для досягнення певного результату. Зазвичай проект містить багато задач, то для зручності відстеження плану їх об'єднують в групи, або фази. Сукупність фаз проекту називається його життєвим циклом.

Фаза проекту складається з однієї або кількох задач, в результаті виконання яких досягається один або кілька основних результатів проекту. Таким чином, результати, досягнуті завдяки виконанню кожної з задач, що входять у фазу, формують її результат.

Якщо для досягнення результатів задачі потрібно виконати тільки її, то для досягнення результату фази потрібно виконати групу інших задач. І в цьому полягає відмінність фази від задачі: її результат підсумовує результати інших задач. Фази можуть складатися як із задач, так і з інших фаз.

Розбиття проекту на фази дозволяє представити його у вигляді списку основних результатів і дат, до яких вони повинні бути отримані. Керівник проекту здійснює безпосередній контроль виконання кожного завдання всередині проекту, повідомляючи вищестоящому менеджеру тільки про досягнення фазових результатів. Цьому менеджеру, в свою чергу, для контролю виконання проекту цілком достатньо таких даних.

Кожен проект робиться для досягнення певної мети, і зазвичай досягти її не можна, не досягнувши декількох проміжних цілей.

Задачі, в результаті виконання яких досягаються проміжні цілі, називаються завершальними завданнями (*віхами*). Зазвичай результатом фази є досягнення проміжної мети, тому віхою в плані проекту прийнято позначати останню задачу фази, в результаті якої досягається її результат.

Іноді, якщо такої задачі немає, а фазовий результат досягається, наприклад одночасним завершенням кількох задач, то створюється фіктивна завершальна задача. Тривалість такої задачі встановлюється в 0 днів, і на неї не виділяються виконавці. Вона присутня в плані виключно для позначення моменту завершення фази, що полегшує відстеження плану проекту.

Тривалість задачі – це період робочого часу, який необхідний для того, щоб виконати її.

Задачі в плані проекту взаємопов'язані, наприклад, часто одна задача не може початися, поки не закінчена інша.

На плані проекту залежно позначаються за допомогою зв'язків, і обидва ці терміни – залежність і зв'язок – використовуються з одним і тим же змістом, позначаючи логіку, визначальну послідовність робіт в плані проекту.

Під *ресурсами* розуміються співробітники та обладнання, необхідні для виконання проектних завдань. Кожен співробітник, який бере участь у проекті, отримує певну роль відповідно до своєї кваліфікації, вимог проекту і регламентами, що діють у організації.

Важлива *властивість ресурсів* – вартість (витрати) їх використання у проекті. Існує два типи вартості ресурсів: погодинна ставка і вартість за використання. Погодинна ставка виражається у вартості використання ресурсу в одиницю часу, величина витрати на використання позначає вартість використання обладнання або співробітника в задачі, яка не залежить від того, скільки часу задіяний в задачі співробітник або матеріальний ресурс. Загальні витрати на використання ресурсу визначаються шляхом множення вартості використання на число завдань, в яких він задіяний.

Призначення – це зв'язок певної задачі і ресурсів, необхідних для її виконання. При цьому на одну задачу можуть бути призначені кілька ресурсів, як матеріальних, так і нематеріальних.

Призначення об'єднують в плані ресурси і завдання, роблячи план цілісним. Завдяки призначень вирішується цілий ряд завдань планування. По-перше, визначаються відповідальні за виконання завдань. По-друге, коли визначені завдання, за які відповідає ресурс, можна розрахувати загальний обсяг часу, що витрачаються їм на проект, а значить, його вартість для проекту. По-третє, визначивши вартість участі всіх ресурсів у проекті, можна підрахувати його загальну вартість. Нарешті, призначаючи ресурси на завдання, можна скорочувати термін виконання робіт, виділяючи на них більше ресурсів і тим самим скорочуючи загальну тривалість проекту.

Кожній фазі життєвого циклу проекту відповідають групи процесів управління проектами. Основні групи процесів відображені на рис.2.

Група процесів ініціації – визначає й авторизує проект або фазу проекту.
Група процесів планування – визначає й уточнює цілі й планує дії,
необхідні для досягнення цілей і змісту, заради яких був розпочатий проект.

Група процесів виконання – поєднує людські й інші ресурси для виконання плану керування проектом.



Рисунок 8.2 – Групи процесів життєвого циклу проекту

Група процесів моніторингу й керування – регулярно оцінює прогрес проекту й здійснює моніторинг, щоб виявити відхилення від плану керування проектом, і, якщо буде потреба провести корегувальні дії для досягнення цілей проекту.

Група завершальних процесів – формалізує приймання продукту, послуги або результату й підводить проект або фазу проекту до правильного завершення.

Практична підготовка до виконання лабораторної роботи

Приклад

Постановка задачі

Створити проект забезпечення підприємства інформаційними технологіями для підвищення ефективності роботи підприємства.

Опис предметної області: Потрібно скласти проект повного циклу розробки програмного комплексу, що включає проектування, визначення структури збережених даних, реалізацію відповідного інтерфейсу, написання програмного коду, тестування програмного продукту, оформлення відповідної документації. Результатом проекту повинен бути розроблений програмний комплекс який забезпечить підприємство інформаційними технологіями та призведе до підвищення стабільності технологічного процесу, зменшення людського фактора, поліпшенню прозорості виробництва, що в кінцевому

підсумку позитивно позначиться на якості готової продукції і призведе до зниження її собівартості.

Аналіз предметної області

Першим етапом розробки програмного забезпечення для підприємства буде процес ініціації.

Мета цього етапу - узгодити із замовником концепцію проекту. На цьому етапі в числі учасників проекту виступають тільки керівник проекту та

замовник. Вони формують найважливіші обмеження проекту – терміни, ресурси і предметну область проекту, узгодять цілі проекту.

Другим етапом розробки програмного забезпечення для підприємства буде процес планування.

На цій стадії всі учасники проекту включаються в роботу над створенням плану проекту. Для того щоб система MS Project могла розрахувати і створити робочий графік, потрібно подати відомості про задачі, які членам команди проекту і субпідрядникам потрібно виконувати, про устаткування і матеріалах, використовуваних для виконання задач, а також про плановані витрати. Кожен учасник проекту повинен сконцентруватися на задачах, які перераховані в таблиці, щоб успішно завершити процес планування і перейти до наступного етапу. Результатом цього етапу є детальний план проекту та бюджет.

Третім етапом розробки програмного забезпечення для підприємства буде процес реалізації та контролю.

Мета цього етапу – виконання робіт проекту відповідно до затвердженого плану. На цій стадії всі учасники мають доступ до даних проекту. Члени проектної групи звітують про фактичний хід виконання задач. Керівник проекту повинен вчасно визначити проблеми та вузькі місця, відслідковувати баланс завантаження ресурсів. Завершення цього етапу означає, що команда проекту та інші учасники проекту згодні з тим, що реалізовані роботи відповідають цілям і планом проекту і реалізовані успішно.

Четвертим етапом розробки програмного забезпечення для підприємства буде процес завершення.

На цьому етапі необхідно зробити ретельний аналіз всіх рішень, дій і зусиль, зроблених в ході проекту, зробити якісний аналіз досягнень і недоліків проекту, довести інформацію до замовника, а також провести заключну нараду за проектом і здати документи до архіву. Це дозволить співробітникам організації скористатися результатами в наступних проектах

Цілі проекту: створити програмне забезпечення для підприємства.

Даний проект буде реалізований на підприємстві.

Продуктом проекту буде розроблено програмний комплекс для автоматизації роботи підприємства.

Автоматизація дозволить вести облік товарообігу і грошових потоків будь-якої складності, враховуючи при цьому безліч різних факторів.

Планування робіт виконання проекту

На основі аналізу предметної області визначимо процеси фаз життєвого циклу проекту «Розробка програмного забезпечення для підприємства»:

Сфера – впровадження інформаційних технологій.

1 - Фаза ініціалізації

Віха 1 – Зародження проекту (генеральна ідея). Укладення контракту.

1.1 Діагностика підприємства Замовника проекту

1.2 Аналіз доцільності, своєчасності та фінансової можливості реалізації проекту (ТЕО)

1.3 Розробка технічного завдання (ТЗ) на проект Віха 2 - ТЗ на проект

2 - Фаза планування

Віха 3 - Проведення переговорів, узгодження та підпис ТЗ

2.1 Обробка, аналіз і систематизація отриманої інформації

2.2 Формалізація бізнес-проектів підприємства Замовника

2.3 Побудова моделі «Як є»

2.4 Розробка і затвердження концепції впровадження проекту

2.5 Побудова моделі «Як повинно бути»

Віха 4 Концептуальний проект впровадження ІС

3 - Фаза реалізації

Віха 5 Постановка завдання на впровадження проекту

3.1 Аналіз ринку АСУ відповідно з фінансовими, функціональними і технічними обмеженнями

3.2 Проведення закупівель необхідного обладнання

3.3 Виконання робіт по проекту

3.4 Розробка програмного забезпечення 3Віха 6 - Акт приймання виконаних робіт 4 - Фаза завершення

Віха 7 - Введення в експлуатацію програмного продукту

4.1 Прийняття та введення в експлуатацію

4.2 Налагодження програмного продукту

4.3 Приймання і запуск в робочу експлуатацію

4.4 Закриття проекту, архівація даних і аудит проекту Віха 8 - Розпуск команди проекту

Після визначення процесів життєвого циклу проекту приступимо до планування реалізації проекту «Розробка програмного забезпечення для підприємства».

При розробці проекту необхідно визначити наступні етапи роботи:

- визначити мету проекту, та задачі які необхідні для виконання;
- призначити терміни виконання задач, тривалість, послідовності виконання кожної задачі;
- спланувати логічні зв'язки робіт;
- визначити потреби в ресурсах і скласти ресурсний план проекту;

На першому етапі розробки проекту програмного забезпечення для підприємства визначимо мету створення проекту і задачі, необхідні для виконання.

Мета – розробка програмного забезпечення для підприємства – розробити програмне забезпечення для забезпечення підприємства інформаційними технологіями: отримати точний і повний розклад проекту з урахуванням робіт, їх термінів, необхідних ресурсів, які служать основою для виконання проекту.

Для виконання проекту необхідно розглянути наступні задачі:

- 1) постановка задачі;
- збір вимог;
 - аналіз вимог;

- погодження вимог із замовником;
- внутрішня постановка задачі завершена;
- постановка задачі завершена;
- 2) проектування;
 - проектування логічної моделі;
 - проектування фізичної моделі;
 - створення плану тестування;
 - проектування завершено.
- 3) реалізація проекту;
 - створення структури бази даних;
 - розроблення інтерфейсів;
 - розробка бізнес-сервісів;
 - розробка клієнтської частини;
 - реалізація завершена.
- 4) тестування проекту;
 - складання релізу;
 - тестування-1;
 - виправлення помилок;
 - тестування-2;
 - тестування завершено.
- 5) здавання-приймання проекту;
 - складання програмної документації;
 - погодження часу приймання;
 - демонстрація замовнику;
 - продукт переданий замовнику.
- б) завершення проекту.

На наступному етапі розробки проекту визначимо дату початку проекту, терміни виконання задач, тривалість, послідовності виконання кожної задачі проекту (попередники робіт) (табл.1).

Таблиця 8.1– Дані для виконання проекту

Номер задачі	Назва задачі	Термін виконання задачі	Попередники робіт
1	2	3	4
1	ПОСТАНОВКА ЗАДАЧІ		
2	Збір вимог	7 днів	1
3	Аналіз вимог	5 днів	2
4	Погодження вимог із замовником	1 тиждень	3
5	Внутрішня постановка задачі завершена	1 днів	4
6	Постановка завдання завершена	0 днів	5
7	ПРОЕКТУВАННЯ		
8	Проектування логічної моделі	3 дня	6
9	Проектування фізичної моделі	4,5 днів	8
10	Створення плану тестування	5 годин	9
11	Проектування завершено	0 днів	9,10
12	РЕАЛІЗАЦІЯ		
13	Створення структури бази даних	1 день	9
14	Розроблення інтерфейсів	7 годин	8
15	Розробка бізнес-сервісів	6 годин	13
16	Розробка клієнтської частини	7 годин	14
17	Реалізація завершена	0 днів	15,16
18	ТЕСТУВАННЯ		
19	Складання релізу	4 години	17
20	Тестування-1	2 дні	19
21	Виправлення помилок	3 дні	20
22	Тестування-2	1 день	21
23	Тестування завершено	0 днів	22
24	ЗДАВАННЯ-ПРИЙМАННЯ ПРОЕКТУ		
25	Складання програмної документації	3 дні	23
26	Погодження часу приймання	1 день	25
27	Демонстрація замовнику	5 годин	26
28	Продукт переданий замовнику	1 день	27
29	Проект завершений	0 днів	27

На наступному етапі призначимо ресурси проекту, які необхідні для виконання задач проекту.

Замовник – спілкується з керівником проекту, визначає цілі та предметну область проекту, призначає відповідальних, аналізує проблеми, надає загальні документи. У процесі планування спілкується з керівником проекту, стверджує календарні плани і бюджет. У процесі реалізації відстежує прогрес і відхилення (звіти в режимі реального часу), має можливість аналізувати вартість, відстеження проблеми, затверджувати зміни предметної області. У процесі завершення проекту підтверджує здачі робіт, контролює за остаточним дозволом всіх проблем.

Керівник проекту – створює узагальнений план для надання замовнику, створює календарний план, планує ресурси для проектів. Управляє, аналізує, складає звітність по проекту. Головний обов'язок керівника проекту - зробити все, що обіцяно замовнику в потрібний час і вклавшись в бюджет.

Проектувальник – проектує етапи проекту, проект.

Програміст – створює структуру бази даних, розробляє інтерфейс, розробляє бізнес-сервіс, розробляє клієнтську частину, виправляє помилки, демонструє замовнику.

Тестувальник – тестує проект, описує помилки, демонструє замовнику.

Аналітик – збирає вимоги по проекту, аналізує всі вимоги, погоджує вимоги із замовником.

Завдання для виконання лабораторної роботи:

Провести аналіз предметної області у відповідності із варіантом (варіант видається викладачем), та визначити основні характеристики проекту:

- 1) мету проекту;
- 2) список необхідних задач для виконання проекту;
- 3) призначити терміни, тривалість, послідовності виконання кожної задачі;
- 4) визначити список необхідних для виконання проекту ресурсів.

ЛАБОРАТОРНА РОБОТА №9

«СКЛАДАННЯ ТЕХНІЧНОГО ЗАВДАННЯ ДО ПРОЕКТУ»

Мета: розглянути порядок розробки ТЗ на основі нормативної документації.

Завдання: студентам необхідно відповідно до обраного варіанту розробити ТЗ для інформаційної системи на основі наведеного шаблону

Звіт про виконання лабораторної роботи:

1. Тема лабораторної роботи.
2. Відповіді на запитання.
3. Результати виконання завдання в електронному вигляді.

Контрольні питання:

1. Дайте визначення ТЗ.
2. Які вимоги існують для оформлювання ТЗ?

Література:[2], [3]

Практична підготовка до роботи

Порядок розробки технічного завдання

1. Загальні положення
 - 1.1. Технічне завдання оформляють відповідно до ГОСТ 19.106-78 на аркушах формату А4 та А3 за ГОСТ 2.301-68, як правило, без заповнення полів аркуша. Номери аркушів проставляють у верхній частині аркуша над текстом.
 - 1.2. Технічне завдання має містити такі розділи:
 - 1.2.1. Вступ;
 - 1.2.2. найменування та сфера застосування;
 - 1.2.3. основи розробки;

1.2.4. технічні вимоги до програми чи програмного виробу;

1.2.5. техніко-економічні показники;

1.2.6. стадії та етапи розробки;

1.2.7. порядок контролю та приймання;

1.2.8. програми.

1.3. Залежно від особливостей програми чи програмного виробу допускається уточнювати зміст розділів, вводити нові розділи чи об'єднувати окремі.

2. Зміст розділів

2.1. Введення має містити коротку характеристику області застосування програми або програмного продукту, а також об'єкта, в якому передбачається їх використовувати. Основне призначення вступу – продемонструвати актуальність даної розробки та показати, яке місце ця технологія займає у ряді подібних.

2.2. У розділі «Найменування та область застосування» вказують найменування, коротку характеристику та область застосування

програми

та об'єкта, в якому використовують програму.

2.3. У розділі «Підстава для розробки» слід вказати:

2.3.1. документ, виходячи з яких ведеться розробка. Таким документом може бути план, наказ, договір тощо.

2.3.2. організація, яка затвердила цей документ, та дата його затвердження

2.3.3. найменування та (або) умовне позначення теми розробки.

2.4. У розділі "Призначення розробки" має бути зазначено функціональне

та експлуатаційне призначення програми.

2.5. Розділ «Технічні вимоги до програми» має містити такі підрозділи:

- вимоги до функціональних характеристик;
- вимоги до надійності;

- умови експлуатації;
- вимоги до складу та параметрів технічних засобів;
- вимоги до інформаційної та програмної сумісності;
- вимоги до маркування та упаковки;
- вимоги до транспортування та зберігання;
- особливі вимоги.

2.5.1. У розділі «Вимоги до функціональних характеристик» повинні бути зазначені вимоги до складу виконуваних функцій, організації вхідних та вихідних даних, тимчасова характеристика тощо.

2.5.2. У розділі «Вимоги до надійності» мають бути зазначені вимоги щодо забезпечення надійного функціонування (забезпечення сталого функціонування, контроль вхідної та вихідної інформації, час відновлення після відмови тощо)

2.5.3. У розділі «Умови експлуатації» повинні бути зазначені умови експлуатації (температура навколишнього повітря, відносна вологість тощо для вибраних типів носіїв даних), за яких повинні забезпечуватись задані характеристики, а також вид обслуговування, необхідна кількість та кваліфікація персоналу.

2.5.4. У розділі «Вимоги до складу та параметрів технічних засобів» вказують необхідний склад технічних засобів із зазначенням їх технічних характеристик.

2.5.5. У розділі «Вимоги до інформаційної та програмної сумісності» мають бути зазначені вимоги до інформаційних структур на вході та виході та методів вирішення, вихідних кодів, мов програмування.

2.5.6. У розділі "Вимоги до маркування та упаковки" в загальному випадку вказують вимоги до маркування програми, варіанти та способи упаковки.

2.5.7. У розділі "Вимоги до транспортування та зберігання" повинні бути зазначені для програмного продукту умови транспортування, місця зберігання, умови зберігання, умови складування, термін

зберігання в різних умовах.

2.5.8. У розділі «Техніко-економічні показники» мають бути зазначені: орієнтовна економічна ефективність, передбачувана річна потреба, переваги порівняно з найкращими вітчизняними та зарубіжними зразками та аналогами.

2.6. У розділі «Стадії та етапи розробки» встановлюють необхідні стадії розробки, етапи та зміст робіт, а також, як правило, терміни розробки та визначають виконавців.

2.7. У розділі «Порядок контролю та приймання» мають бути зазначені види випробувань та загальні вимоги до приймання роботи.

2.8. У додатках до технічного завдання при необхідності наводять:

2.8.1. перелік науково-дослідних та інших робіт, що обґрунтовують розробку;

2.8.2. схеми алгоритмів, таблиці, описи, обґрунтування, розрахунки та інші документи, які можуть бути використані для розробки;

2.8.3. інші джерела розробки.

ЛІТЕРАТУРА ОСНОВНА

1. Software Engineering. 10 Ediotioion. Ian Sommerville. ISBN 978-0-13-394303-0, published by Pearson Education . 2016
2. Проектування інформаційних систем: навчальний посібник / В.С. Авраменко, А.С. Авраменко. – Черкаси: Черкаський національний університет ім. Б. Хмельницького, 2017. – 434 с.: іл. ISBN 978-966-920-208-6

ДОДАТКОВА:

3. The Complete Guide To Understand IDEF Diagram,. URL: <https://www.edrawmax.com/article/the-complete-guide-to-understand-idef-diagram.html> (дата звернення 21.06.22)
4. Qing Li, Yu-Liu Chen. Modeling and Analysis of Enterprise and Information Systems. From Requirements to Realization. Springer 2009.
5. Data Governance & Data Modeling Solutions. URL: <https://www.erwin.com/> (дата звернення 20.10.2022)
6. ЩО ТАКЕ USE CASE ТА ДЛІА ЧОГО ВОНИ ПОТРІБНІ. URL: <https://training.qatestlab.com/blog/technical-articles/what-is-a-use-case-and-what-are-they-for/> (дата звернення 20.10.2022)