

КОНТРОЛЬНА РОБОТА ПОБУДОВА ДІАГРАМ ВАРІАНТІВ ВИКОРИСТАННЯ (USECASE DIAGRAMS)

Мета роботи: відповідно до варіанта завдання, узгодженого з викладачем,:

- знайти приклади 3-4 аналогічних програмних систем та проаналізувати принципи їх функціонування. На основі отриманих даних створити проект технічного завдання для розробки власної системи;
- на основі проведеного аналізу прикладної галузі розробити usecases для своєї системи;
- побудувати діаграми варіантів використання (діаграми прецедентів) для обраного варіанту комп'ютерної системи.

Обладнання:

- ПК IBM PC x86 CPU з встановленою операційною системою;
- доступ до мережі інтернет.

1 Теоретичні відомості

Usecase – це текстовий опис сукупності сценаріїв, що виконуються користувачем при роботі з системою для досягнення певної мети.

Сценарій – послідовність дій при взаємодії користувача із системою для виконання певної операції.

Наприклад, купівля товарів у супермаркеті з використанням кредитної картки і невдала спроба купівлі через перевищення кредитного ліміту – це сценарій, а купівля товарів у супермаркеті – usecase.

Приклад 1. Зняття готівки в банкоматі.

Користувач підходить до банкомату та вставляє свою картку. Система перевіряє картку та просить ввести пін-код. Після перевірки пін-коду система виводить головне меню, де користувач обирає опцію видачі готівки. Він визначає суму грошей та підтверджує виконання операції.

Банкомат видає готівку та чек. Система видає запит про здійснення наступної операції. Користувач обирає опцію «не виконувати», забирає картку та йде.

Приклад 2. Замовлення літератури в бібліотеці.

Головний сценарій (успішний):

Користувач підходить до бібліотекаря і надає йому замовлення (перелік видань). Бібліотекар заносить номер читацького квитка до системи і перевіряє облікову картку користувача. Бібліотекар шукає кожне видання в базі і визначає кількість екземплярів вільних у даний момент часу. Система видає шифри (коди), за якими бібліотекар знаходить видання у сховищі, заносить номери книг до картки користувача і видає йому літературу. При цьому читацький квиток лишається у бібліотекаря.

Альтернативні сценарії:

1. Користувач бібліотеки є боржником, бібліотекар не може видати йому літературу.
2. Бібліотекар не може видати користувачу літературу, оскільки в даний момент часу немає вільних екземплярів у сховищі.
3. Читацький квиток користувача є недійсним. Бібліотекар вилучає його.
4. Технічний збій роботи системи. Видається повідомлення «немає зв'язку з сервером баз даних». Бібліотекар викликає адміністратора системи.

Написання usecases дозволяє чітко визначити хто є користувачем системи, які її сценарії роботи, що є метою використання системи.

Usecases – це функціональні та поведінкові вимоги до системи, які показують, що саме вона має робити.

Існують **три форми** написання usecases:

1. Коротка – короткий опис в один абзац одного зі сценаріїв (зазвичай успішного) роботи системи (приклад 1). Виконуються під час початкового аналізу вимог до системи.
2. Поверхнева – поверхневий опис у вільній формі усіх сценаріїв (головного і альтернативних) одного з usecases (приклад 2). Виконуються під час початкового аналізу вимог до системи.
3. Повна – всі кроки і дії детально описані, включаючи перед- та постумови виконання usecase. Виконуються на стадії відбору з повного переліку usecases у короткій та поверхневій формах невеликої частини важливих (критичних для роботи системи) usecases.

Повна форма опису usecase має перелік розділів, коротко описаних у таблиці 1. Нижче наведено приклад складання usecase у повній формі для системи продажу товарів у супермаркеті.

Таблиця 1 - Параметри опису usecase

Usecase section	Comment
Use Case Name	Назва usecase (починається з дієслова)
Scope	System or Business
Level	User-goal or Sub-function
Primary Actor	Головний актор
Stakeholders and interests	Перелік осіб, які зацікавлені в виконанні даного usecase та мета, яку вони при цьому переслідують
Preconditions	Список передумов, які повинні виконуватись для початку виконання usecase
Success guarantee	Список умов, при виконанні яких можна говорити про успішне закінчення виконання usecase
Main Success Scenario	Головний успішний сценарій usecase. Найчастіше є безумовним
Extensions	Альтернативні сценарії успішного чи неуспішного закінчення usecase
Special Requirements	Спеціальні (нефункціональні) вимоги
Technology and Data Variations List	Поради для реалізації певних кроків usecase
Frequency of Occurrence	Частота виконання usecase при користуванні системою (у відсотках)
Miscellaneous	Додаткові вимоги чи відкриті питання

Приклад 3. Купівля товарів у супермаркеті.

Scope: система продажу товарів в супермаркеті (System)

Level: User-goal

Primary Actor: касир

Stakeholders and interests:

1 Касир: зацікавлений у точному швидкому вводі інформації про товари та відсутності помилок вводу, які призведуть до штрафу (зменшення зарплатні касира)

2 Покупець: зацікавлений у швидкому придбанні товарів (отриманні послуг) та зручному відображенні розрахунків суми покупки.

3 Компанія (продавець): зацікавлена в точній обробці транзакцій при покупці товарів та задоволенні інтересів користувача.

4 Менеджер: зацікавлений у швидкому розв'язанні проблем під час повернення товарів та легкій перевірці операцій, що здійснюють касири.

5 Державна податкова адміністрація: зацікавлена в отриманні податків від кожного продажу товарів.

Preconditions: касир виконав вхід до системи (авторизація)

Main Success Scenario:

1 Покупець підходить до каси з сукупністю товарів (послуг), які він хоче придбати.

2 Касир стартує нову продаж.

3 Касир вводить послідовно всі товари.

4 Система опрацьовує код кожного товару та виводить його назву, кількість та суму покупки для кожного товару.

5 Система розраховує загальну суму покупки та суму нарахованих податків.

6 Касир озвучує суму покупцеві та просить розрахуватися.

7 Покупець розраховується і система опрацьовує оплату.

8 Система запам'ятовує суму покупки до бази даних.

9 Система друкує чек.

10 Користувач відходить від каси з чеком і товарами.

Extensions:

a) у будь-який час менеджер виконує специфічну операцію:

1) менеджер вводить свій код авторизації;

2) менеджер виконує специфічну операцію (наприклад, перевіряє баланс даної каси);

3) менеджер виходить з системи, і вона повертається до режиму роботи з касиром.

b) у будь-який час виникає фатальна помилка в системі:

1) касир перевантажує систему, входить до неї та вводить запит про повернення до попереднього стану;

2) система відновлює попередній стан.

2a) система не може відновити попередній стан:

- система видає помилку на екран та зберігає її у лог;

- касир починає новий продаж.

1a) покупець чи менеджер хоче повернутися до відкладеного продажу:

1) касир виконує операцію повернення та вводить код продажу;

2) система відображає відтворений продаж.

2a) продаж не було знайдено:

- система виводить помилку на екран;

- касир розпочинає новий продаж і вводить дані про всі товари з початку.

2-4a) покупець говорить касиру, що у нього є картка для отримання знижки:

1) касир вводить номер картки до системи;

2) система запам'ятовує код картки та вид знижки для розрахунку загальної суми продажу.

3a) код товару не знайдено в базі даних:

1) система виводить помилку на екран;

2) касир опрацьовує помилку:

2a) додатковий код товару знаходиться на ньому і касир може його прочитати:

- касир вводить код вручну;

- система опрацьовує код та виводить назву товару та ціну.

2b) касир виконує usecase «Знайти код невідомого товару» для ідентифікації товару.

3b) у покупця є декілька однакових товарів:

1) касир може ввести код товару та його кількість вручну.

3-6a) користувач просить видалити один з товарів із поточного продажу:

1) касир вводить номер товару для видалення;

2) система видаляє строку товару з продажу і перераховує загальну суму.

3-6b) покупець просить касира відмінити продаж у цілому:

1) касир відмінює продаж.

3-6с) касир відкладає продаж:

1) система запам'ятовує продаж, яка може бути відновлена в будь-який час;

2) система видає спеціальний чек із переліком товарів та кодом продажу, який дозволить продовжити продаж.

4b) Покупець вважає, що товар є зіпсованим і пропонує купити його за нижчою ціною:

1) касир робить запит у менеджера;

2) менеджер дозволяє виконати продаж за нижчою ціною;

3) касир вводить вручну ціну, що є нижчою за попередню;

4) система відображає нову ціну для даного товару в поточному продажі.

5a) користувач говорить, що він повинен отримати знижку:

1) касир робить запит щодо знижки;

2) касир вводить ідентифікаційний код покупця;

3) система розраховує сумарну знижку на продаж.

6a) покупець говорить касиру, що він хотів розплатитися готівкою, але не має достатньо грошей:

1) касир просить обрати інший метод сплати за товари:

1a) покупець просить відмінити продаж.

7a) оплата готівкою:

1) касир вводить суму грошей, яку дав покупець;

2) система розраховує суму решти і відкриває касу;

3) касир видає решту покупцю;

4) система вносить випадок оплати до бази даних.

7b) оплата кредитною карткою:

1) покупець вводить інформацію про кредитну картку;

2) система показує суму оплати;

3) касир підтверджує суму оплати;

4) система відсилає авторизаційний запит до зовнішньої Системи Оплати Товарів та послуг та робить запит на підтвердження оплати;

5) система отримує підтвердження оплати, виводить інформацію для касира та відкриває касу;

6) система зберігає даний вид оплати за товари у базі даних;

7) касир просить покупця зробити підпис на чеку для підтвердження оплати. Покупець ставить підпис;

8) касир складає чек до каси та замикає її.

7с) касир робить відміну оплати:

1) система повертається до режиму вводу товарів.

9а) покупець просить видати подарунковий чек (без суми продажу):

1) касир робить запит про подарунковий чек. Система друкує подарунковий чек.

9b) в принтері закінчився папір:

1) система дає сигнал про закінчення паперу;

2) касир замінює папір;

3) касир робить запит про повторення друку чеку.

Special Requirements:

1) великий плоский монітор, що має функції сенсорного екрану. Текст повинно бути гарно видно з відстані в 1 м;

2) система авторизації запитів про оплату кредитною картою повинна видавати результат за 30 секунд у 90 відсотків випадків;

3) система перекладу тексту на декілька мов;

4) додаткові бізнес-правила можуть бути додані до пунктів 3-7.

Technology and Data Variations List:

1) менеджер виконує авторизацію під час скасування певної операції за допомогою зчитування номера картки менеджера кард-рідером чи введенням коду менеджера з клавіатури;

2) код товару вводиться сканером штрих кодів товарів чи з клавіатури;

3) система може використовувати декілька схем кодування товарів (UTC, EAN, JAN, SKU);

4) інформація про кредитну картку вводиться за допомогою кардрідера чи з клавіатури;

5) підпис покупця для підтвердження кредитної операції робиться на товарному чеці.

Frequency of Occurrence: 95 %.

Miscellaneous (Open Issues):

- 1) провести аналіз різних варіантів сплати податків;
- 2) вивчити можливість відновлення роботи системи після збою;
- 3) яка додаткова функціональність потрібна для різних прикладних галузей?
- 4) чи повинен касир забирати гроші з каси, коли він виходить із системи (закінчує зміну)?

5) чи може покупець використовувати сканер для ідентифікації товарів під час покупки, чи це повинен робити касир?

Даний приклад є більш ніж детальним прикладом опису usecase в повній формі, але є цілком реальним (розроблений під час аналізу вимог для системи продажу товарів NextGen POS, яка реалізована мовою Java).

Діаграми варіантів використання (usecase diagrams) використовуються для відображення сценаріїв використання системи (usecases) та користувачів системи (actors), які використовують її функції.

Актори на діаграмі варіантів використання позначаються символом людини, а варіанти використання – еліпсом. Актори та варіанти використання поєднуються напрямленою асоціацією (unidirectional association) – стрілкою, що спрямована від актора до варіанта використання. Також актори можуть поєднуватися з використанням зв'язків узагальнення. На рис. 1 показано фрагмент діаграми варіантів використання для інтернет-магазину. Актор «Покупець» при цьому може виконати сценарій «Замовити товар».

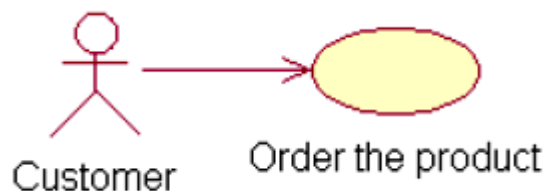


Рисунок 1 - Фрагмент діаграми варіантів використання

Варіанти використання можуть бути пов'язані між собою трьома видами зв'язків: узагальненням (generalization),

розширенням (extend relationship) та включенням (include relationship).

Відношення узагальнення (generalization) показують відношення між загальним і частковим. Наприклад на рис. 2 варіанти використання «Search by category» та «Search by producer» є частковими випадками загального варіанта «Search product», тому вони поєднані даним відношенням. Також дане відношення може використовуватися для поєднання акторів. Актор «Administrator» може виконувати всі функції актора «Customer», тобто виступає частковим випадком покупця, але може виконувати і специфічні операції (варіант використання «Check db info»).

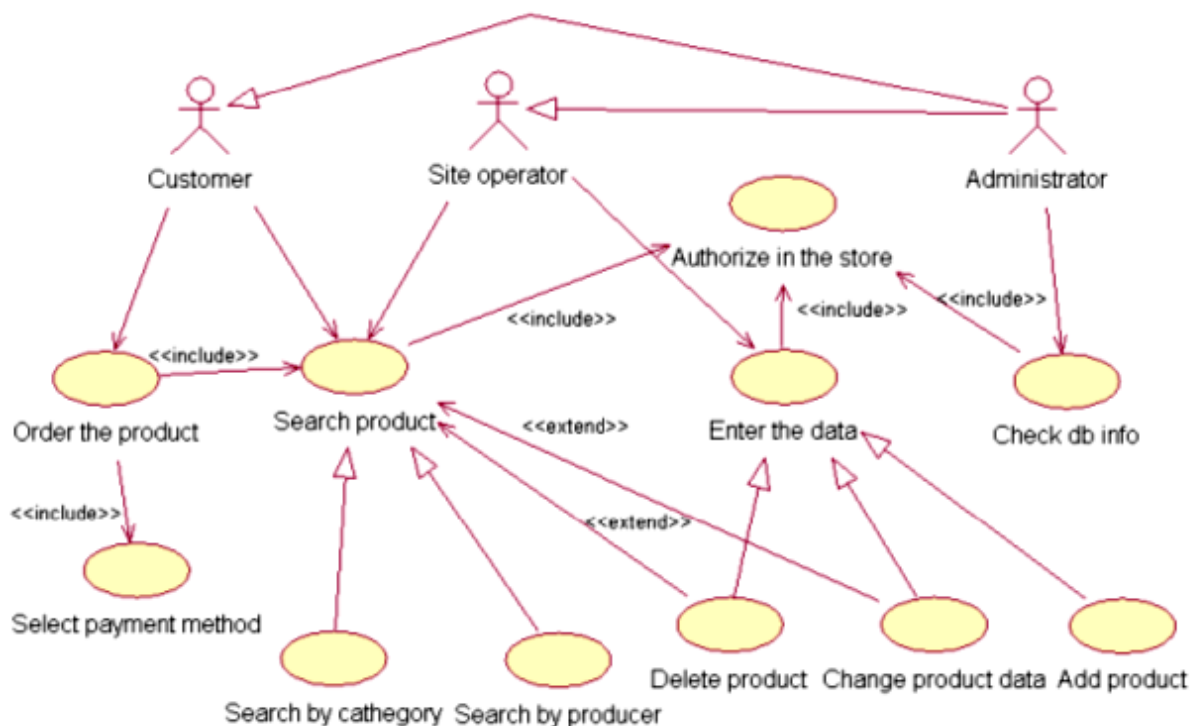


Рисунок 2 - Повна діаграма варіантів використання для інтернет-магазину

Відношення включення (include) відображає зв'язок «ціле – частина», тобто один варіант завжди в певний момент виконання повністю включає інший. Для прикладу частиною варіанта використання «Order the product» є сценарії «Search product» та «Select payment method», оскільки для того, щоб замовити товар

покупець завжди має відшукати його в каталозі та обрати метод оплати.

Відношення розширення (extend) визначає такий тип відношення, коли один варіант за певних умов повністю використовує інший (розширює його). Так, наприклад, оператор Інтернет магазину може видалити товар («Delete product»), знаючи його ідентифікатор, або провівши попередньо пошук товару («Search product»).

2 Порядок виконання роботи:

Контрольна робота повинна складатися з частин:

Перша частина містить аналіз аналогічних систем. Для кожного аналогу ПЗ визначити наступні характеристики:

- назва;
- розробник (дистриб'ютор);
- архітектура (desktop application, client-server, 3tier web application);
- мова реалізації;
- перелік функцій, характеристик (не менше 5);
- аналіз переваг та недоліків даного ПЗ;
- джерело інформації (веб-сайт);

Друга частина визначає перелік характеристик системи, що проектується та розробляється студентом. Для програмної системи, що проектується визначити:

- призначення ПЗ;
- основні функції (не менше 5);
- користувачі системи (2-3 користувачі);
- архітектура системи;
- опис структури (схема);
- сценарії роботи (не менше 3-х сценаріїв);
- основні логічні сутності (не менше 5);
- основні таблиці БД, що використовуються (не менше 5 таблиць);
- засоби апаратної та програмної реалізації (платформа, ОС та мова програмування);
- output (вихідні дані): таблиці, звіти, графіки (не менше 3).

Третя частина. Відповідно до обраної теми (варіанта) на основі проведеного аналізу прикладної галузі розробити три різні

usecases (по одному в короткій, поверхневій та повній формах відповідно) для своєї системи. Повна форма опису має містити всі пункти наведені в таблиці 1. Головний успішний сценарій повинен мати не менше 10 кроків.

Передбачити не менше 5 альтернативних сценаріїв.

Четверта частина. З використанням обраного веб-сервісу (наприклад, creately.com) створити діаграму варіантів використання для обраного варіанту комп'ютерної системи. Діаграма повинна містити усіх акторів (користувачів системи) та по три варіанти використання для кожного актора. Пов'язати варіанти використання та акторів, при цьому використати усі види зв'язків (unidirectional association, generalization, extend relationship, include relationship).

3 Варіанти завдання:

1. GPS навігатор
2. Електронний словник (перекладач).
3. Банкомат.
4. Система обліку товарів на складі.
5. Автоматизована бібліотечна система.
6. Інтернет-блог.
7. Інтернет-магазин.
8. Інтернет-форум.
9. Система підтримки складання розкладу занять.
10. Система продажу товарів у супермаркеті.
11. Редактор зображень.
12. Система документообігу на підприємстві.
13. Робоче місце бухгалтера з нарахування заробітної платні.
14. Електронна система продажу нерухомості.
15. Система оплати комунальних послуг.
16. Електронна система продажу ліків в аптеці.
17. Система керування контентом мобільного оператора.
18. Електронна система для туристичного оператора.
19. Електронна система продажу квитків до кінотеатру.
20. Електронна система продажу залізничних квитків.
21. Торгівельний автомат.
22. Музичний автомат.
23. Система електронних переказів готівки.

24.Міні-АТС.

25.Електронна система обліку робочого часу працівника на підприємстві.

4 Зміст звіту:

Звіт повинен містити:

- титульний аркуш;
- мету роботи і завдання;
- покроковий опис роботи, копії екранів з результатами виконаної роботи;
- висновки.