


ТОВ «ТЕХНІЧНИЙ УНІВЕРСИТЕТ
«МЕТІНВЕСТ ПОЛІТЕХНІКА»

СТАНДАРТИ ТА МЕТОДОЛОГІЇ БІЗНЕС-АНАЛІЗУ:

методичні рекомендації
до виконання контрольних робіт

Запоріжжя 2025



УДК 005.311:005.52 (072)
С77

Рекомендовано Науково-методичною
радою ТОВ «ТЕХНІЧНИЙ
УНІВЕРСИТЕТ «МЕТІНВЕСТ
ПОЛІТЕХНІКА»
(протокол № 3 від 26.12.2025 р.)

Укладач:
Москаленко В.В., д.т.н.

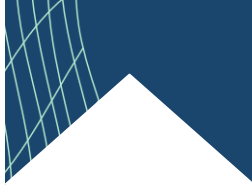
С77 Стандарти та методології бізнес-аналізу : методичні
рекомендації до виконання контрольних робіт / уклад.
В. В. Москаленко. Запоріжжя : ТОВ «ТЕХНІЧНИЙ
УНІВЕРСИТЕТ «МЕТІНВЕСТ ПОЛІТЕХНІКА», 2025. 54 с.

У методичних рекомендаціях наведено умови, поради і методичні підходи до виконання контрольних робіт з дисципліни «Стандарти та методології бізнес-аналізу», вимоги до оформлення, подання та оцінювання результатів виконання контрольних робіт.

Рекомендовано для студентів спеціальності 122 - Комп'ютерні науки першого (бакалаврського) рівня освіти.

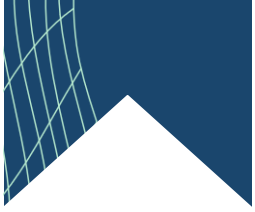
УДК 005.311:005.52 (072)

©ТОВ «ТЕХНІЧНИЙ УНІВЕРСИТЕТ МЕТІНВЕСТ ПОЛІТЕХНІКА», 2025



Зміст

ВСТУП.....	4
1. ПОРЯДОК ВИКОНАННЯ, ПОДАННЯ НА ОЦІНЮВАННЯ ТА ЗАХИСТ КОНТРОЛЬНИХ РОБІТ.....	7
1.1 Перелік контрольних робіт з дисципліни «Стандарти та методології бізнес-аналізу».....	7
1.2 Методичні рекомендації до виконання КР №1	7
1.2.1 Завдання на виконання КР №1.....	7
1.2.2 Відомості про специфікацію варіантів використання.....	8
1.2.3 Опис розділів Use Cases специфікації	15
1.2.4 Зміст звіту з виконання КР №1	19
1.3 Методичні рекомендації до виконання КР № 2	20
1.3.1 Завдання до КР № 2 та етапи виконання.....	20
1.3.2 Відомості про Business Process Model and Notation	20
1.3.3 Рекомендації щодо моделювання бізнес-процесу	24
1.3.4 Приклади створення бізнес-процесів	31
1.3.5 Зміст звіту з виконання КР №2	35
1.4 Оформлення, подання і захист контрольної роботи	36
1.5 Критерії оцінювання результатів виконання контрольної роботи	36
СПИСОК РЕКОМЕНДОВАНИХ ДЖЕРЕЛ	38
Додаток А. Приклад оформлення титульного аркушу.....	40
Додаток Б. Шаблон Use Case Specification	41
Додаток В. Приклади формування Use Case специфікації	50



ВСТУП

Контрольна робота – це форма організації навчання, яка має на меті перевірка закріплених знань, які студенти отримують у процесі вивчення певного модуля дисципліни «Стандарти та методології бізнес-аналізу», а також застосування цих знань для розв'язання практичних задач бізнес-аналізу у сфері ІТ.

Методичні рекомендації до виконання контрольних робіт укладено на підставі Стандарту вищої освіти за спеціальністю 122 «Комп'ютерні науки» у галузі знань 12 «Інформаційні технології» для першого (бакалаврського) рівня вищої освіти (затверджено Наказом Міністерства освіти і науки України від 28.04.2022 р. № 393).

Контрольна робота (КР) – це змістовно завершена робота теоретичного-практичного характеру, що виконується на основі знань, умінь і навичок, набутих у процесі лекційних і практичних занять з дисципліни «Стандарти та методології бізнес-аналізу» у продовж певного модуля. Вона охоплює теми одного змістового модулю відповідно до робочої програми дисципліни.

Виконання контрольних робіт передбачає позааудиторну самостійну роботу студента. Метою такої роботи є перевірка теоретичних знань й практичних навичок, отриманих у процесі засвоєння тем відповідного модуля та які відповідають програмним результатам навчання з освітньої компоненти «Стандарти та методології бізнес-аналізу» освітньої програми «Комп'ютерні науки» бакалаврського рівня освіти.

Дисципліна «Стандарти та методології бізнес-аналізу» орієнтована на підготовку студентів (першого рівня вищої освіти) з бізнес-аналізу у сфері ІТ. Метою дисципліни є забезпечення студентів ґрунтовними знаннями щодо стандартів та методологій бізнес-аналізу, а також сприяння отримання практичних навичок у студентів щодо використання технологій бізнес-аналізу для виявлення та управління вимогами до програмного забезпечення (ПЗ) для систем цифрового інтелекту (СЦІ), які використовуються у сферах професійної діяльності різних фахівців підприємств [1].

Важливим аспектом дисципліни є розгляд проведення бізнес-аналізу за гнучкими методологіями (Agile), використання сучасних знань з моделювання та проектування ПЗ, у тому числі Unified Modeling Language (UML) та моделювання бізнес-процесів з використанням стандартизованої графічної мови BPMN. Студенти отримують знань щодо управління вимогами у процесі проектування та розробки ПЗ для

СЦІ в умовах постійної адаптації ПЗ до змінних вимог замовника. Основні завдання дисципліни включають формування знань у студентів щодо процесів обстеження і співробітництва з зацікавленими сторонами проекту з розробки СЦІ, знань щодо управління життєвим циклом вимог, визначення дизайну майбутнього програмного рішення, оцінювання його бізнес-цінності. Дисципліна інтегрує теоретичні знання та практичний досвід для розв'язання конкретних задач бізнес-аналітика. Знання стандартів, методів бізнес-аналізу й отримання практичних навичок їх використання для розробки ПЗ відповідно тем дипломних робіт дозволить студентам використовувати це надалі у своїй професійній діяльності у сфері ІТ [2, 3, 4]. Викладання дисципліни побудовано на основі новітньої інформації щодо стандартів та методологій бізнес-аналізу у проєктах з диджиталізації кращих українських і світових ІТ-компаній.

Освітня компонента (ОК) «Стандарти та методології бізнес-аналізу» спрямована на отримання здобувачами наступних загальних та спеціальних (фахових) компетентностей (згідно ОПП «Комп'ютерні науки» першого (бакалаврського) рівня вищої освіти) [1]:

- ЗК1 – здатність до абстрактного мислення, аналізу та синтезу;
- ЗК2 – здатність застосовувати знання у практичних ситуаціях;
- ЗК3 – знання та розуміння предметної області та розуміння професійної діяльності;
- ЗК6 – здатність вчитися й оволодівати сучасними знаннями;
- ЗК7 – здатність до пошуку, оброблення та аналізу інформації з різних джерел;
- ЗК8 – здатність генерувати нові ідеї (креативність);
- ЗК10 – здатність бути критичним і самокритичним;
- ЗК11 – здатність приймати обґрунтовані рішення;
- ЗК12 – здатність оцінювати та забезпечувати якість виконуваних робіт;
- ЗК16 – здатність ухвалювати рішення та діяти, дотримуючись принципу неприпустимості корупції та будь-яких інших проявів недобросовісності;
- СК6 – здатність до системного мислення, застосування методології системного аналізу для дослідження складних проблем різної природи, методів формалізації та розв'язування системних задач, що мають суперечливі цілі, невизначеності та ризику;

– СК7 – здатність застосовувати теоретичні та практичні основи методології та технології моделювання для дослідження характеристик і поведінки складних об'єктів і систем, проводити обчислювальні експерименти з обробкою й аналізом результатів;

– СК11 – здатність до інтелектуального аналізу даних на основі методів обчислювального інтелекту включно з великими та погано структурованими даними, їхньої оперативної обробки та візуалізації результатів аналізу в процесі розв'язування прикладних задач;

– СК15 – здатність до аналізу та функціонального моделювання бізнес-процесів, побудови та практичного застосування функціональних моделей організаційно-економічних і виробничо-технічних систем, методів оцінювання ризиків їх проектування.

Програмні результати ОК за ОПП «Комп'ютерні науки» [1]:

– ПР11 – володіти навичками управління життєвим циклом програмного забезпечення, продуктів і сервісів інформаційних технологій відповідно до вимог і обмежень замовника, вміти розробляти проєктну документацію (техніко-економічне обґрунтування, технічне завдання, бізнес-план, угоду, договір, контракт);

– ПР14 – вміти застосовувати знання методології та CASE-засобів проєктування складних систем, методів структурного аналізу систем, об'єктно-орієнтованої методології проєктування при розробці і дослідженні функціональних моделей організаційно-економічних і виробничо-технічних систем.

Інші програмні результати:

– вміти використовувати техніки бізнес-аналізу для виявлення вимог щодо проєктування та розробки програмних систем у певних галузях економіки;

– застосовувати сучасні знання стандартів бізнес-аналізу для управління вимогами до програмного забезпечення згідно Agile;

– мати знання з методологій бізнес-аналізу для управління вимогами до систем цифрового інтелекту;

– мати спеціалізовані уміння/навички щодо документування результатів бізнес-аналізу, враховуючи методології розробки програмного забезпечення.

Виконання контрольних робіт передбачає інтеграцію навчальної, практичної, комунікативної та інших видів діяльності здобувачів освіти з використанням матеріалів реальних проєктів у сфері ІТ підприємств, зокрема активів Групи МЕТІНВЕСТ [2, 3, 4].

1. ПОРЯДОК ВИКОНАННЯ, ПОДАННЯ НА ОЦІНЮВАННЯ ТА ЗАХИСТ КОНТРОЛЬНИХ РОБІТ

1.1 Перелік контрольних робіт з дисципліни «Стандарти та методології бізнес-аналізу»

Згідно робочої програми дисципліни передбачено виконання двох контрольних робіт для студентів. Опис контрольних робіт надано у табл.1.

Таблиця 1 – Опис контрольних робіт

№ з/п	Опис індивідуального завдання
1	Розробка Use Case специфікації для формування вимог до ПЗ
2	Моделювання бізнес-процесів для проектування програмного забезпечення

1.2 Методичні рекомендації до виконання КР №1

1.2.1 Завдання на виконання КР №1

Мета контрольного завдання – це представлення специфікації варіантів використання як результат засвоєних знань та отриманих практичних навичок збору та формування функціональних вимог до програмного забезпечення, яке розроблюється в межах дипломної роботи.

Завдання для КР №1:

1) сформулювати мету розробки ПЗ та надати перелік основних бізнес-вимог до ПЗ (використовуйте результати ІДЗ №1, де було надано опис функціональних вимог до ПЗ, опис діаграми варіантів використання, активності тощо у нотації UML);

2) розробити use cases специфікацію згідно шаблону, який надано у додатку Б, приклади заповнення шаблону надано у додатку В.;

3) оформити звіт за результатами виконання контрольної роботи;

4) надати до перевірки результати контрольної роботи викладачу на практичному занятті, після схвалення результатів додати звіт до системи Moodle.

1.2.2 Відомості про специфікацію варіантів використання

Специфікація варіантів використання – це документ, який фіксує вимоги до програмної системи(ПС) або будь-якого програмного забезпечення (ПЗ), які представлені у вигляді варіантів використання (Use Case). Цей документ містить опис кожної вимоги у логічній послідовності, щоб розробники та тестувальники могли зрозуміти, що потрібно системі для створення функціональних можливостей ПЗ, деталізованих у варіантах використання, та для їх тестування [5, 6].

Практичні знання про випадки використання та їх структуру забезпечує хорошу основу для розуміння функціональних можливостей ПЗ та переходу до формування та використання історій користувачів за гнучкими методологіями розробки ПЗ.

Варіант використання можна розділити на декілька історій користувача [7].

Специфікація варіанту використання зазвичай створюється на етапі аналізу вимог та проектування ПЗ ітеративним способом [8].

На початковій фазі розробки вимог до ПЗ кожен Use Case описується на загальному рівні: назва випадку використання та його опис за допомогою діаграми активності, схеми потоків даних, діаграми станів тощо [9, 10].

Отже, спочатку наводиться лише короткий опис кроків, необхідних для виконання звичайного сценарію використання (яка функціональність надається варіантом використання).

Продовжуючи аналіз вимоги кроки конкретизуються, щоб додати більше деталей – додаються потоки дій до сценарію використання.

Однак, не всі випадки використання ПЗ обов'язково потребують однакового рівня деталізації.

Кожен проект може мати свій шаблон специфікації варіанту використання.

Варіант використання може бути візуалізований як Use Case діаграма або/або у форматі структурованої текстової специфікації (рис. 1) [9].

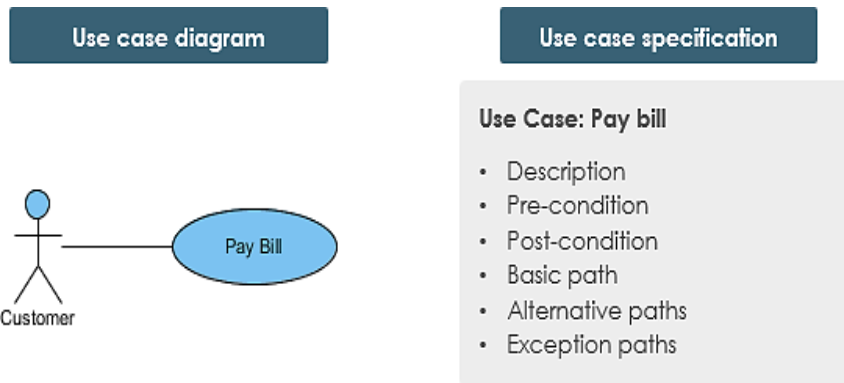


Рисунок 1 – Представлення варіанту використання [6]

Варіант використання має (рис. 2):

- тільки одну мету;
- одну початкову точку;
- одну кінцеву точку;
- декілька шляхів для переходу від початку до кінця (мати різні сценарії дій для різних можливих умов, або кожна умова може вимагати певних дій).

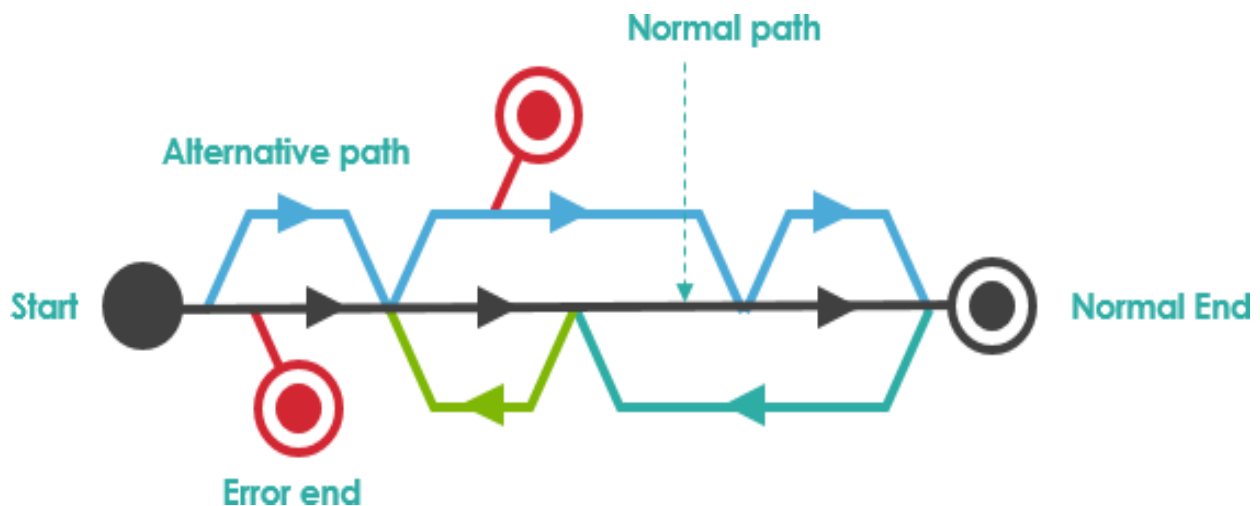


Рисунок 2 – Характеристики Use Case [6]

Наприклад, для Use Case «клієнт оплачує рахунок» (рис.1) існує кілька шляхів досягнення мети: оплата по телефону; поштою; особисто; за чеком; готівкою тощо. Шлях, що не веде до мети – це кредитну картку відхилено.

Use Case модель та окремі варіанти використання «розвиваються» з часом рівень за рівнем в залежності від отриманої інформації про вимогу, якій відповідає Use Case.

Варіанти використання як опис вимоги можуть бути написані на різних рівнях даних і їх обсягу, кожен з яких служить певній меті (рис.3).

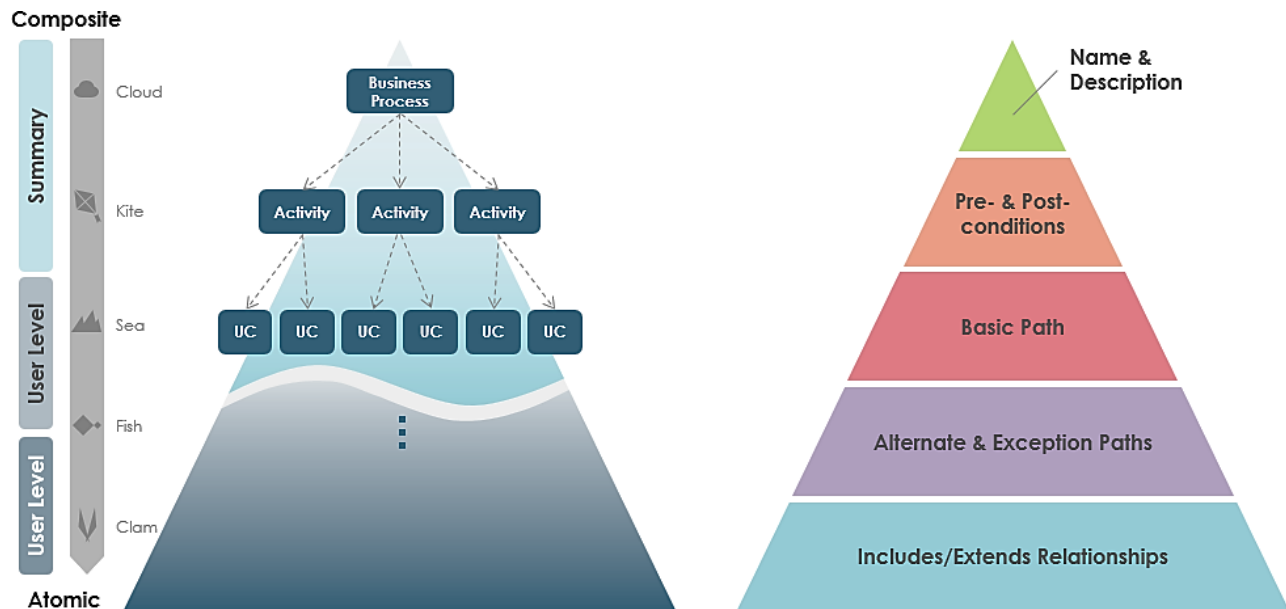


Рисунок 3 – Рівні деталізації вимоги (дій) у Use Case [6]

Резюме (Summary): загальні описи Use Case та широкі огляди функціональності системи або бізнес-процесів.

Рівень користувача (User Level): пов'язані із завданнями описи користувачів і способів їх взаємодії з ПЗ; описи конкретного бізнес-процесу. Use Case на рівні користувача представляються у вигляді завдання, яке виконується користувачем. Наприклад, отримання готівки у банкоматі є завданням і буде представлено як Use Case на базовому рівні, але введення PIN-коду користувача не буде розглядатися на цьому рівні, оскільки воно підтримує основну роботу ПЗ.

Підфункція (Sub-function): опис дій нижчого рівня, які виконуються для завершення частин основного сценарію використання.

Детальний варіант використання – це текстове представлення, яке ілюструє послідовність подій разом з іншою пов'язаною інформацією про варіант використання у певному шаблоні.

Заповнений шаблон документу (специфікації) варіанту використання надає повну інформацію щодо реалізації певної вимоги до ПЗ (рис.4).

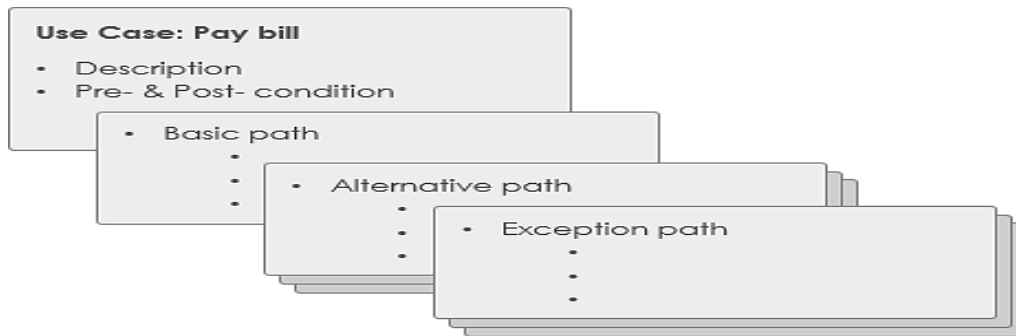


Рисунок 4 – Детальна специфікація випадку використання [6]

Існує декілька стилів нотації для випадків використання (наприклад, Use Case діаграма, текстовий формат, таблиця). Будь-яка нотація має бути легкою для розуміння.

Приклад опису Use Case у вигляді тексту надано у документації інструменту для проектування ПЗ Figma (<https://www.figma.com/resource-library/what-is-a-use-case/>).

Менеджер продукту розробляє програму мобільного банкінгу для своєї компанії. ПЗ має оптимізувати реєстрацію користувачів і налаштування облікового запису. Опис ПЗ та її варіантів використання виглядає так:

Довідкова інформація:

- система: мобільний банкінг;
- основний користувач (актор): клієнти, які хочуть відкрити рахунок;
- другорядний користувач (актор): андеррайтери та автоматизовані інструменти для розрахунку відсоткових ставок і максимального залишку основної суми;
- цілі: заощадити час на реєстрації облікового запису та адаптації;
- зацікавлені сторони: генеральний директор і віце-президент з продукту ІТ-компанії;
- передумови: користувачі завантажують програму, вони відповідають вимогам облікового запису;
- тригери: користувач вибирає створення нового облікового запису у програмі.

Сценарії:

- **Основний процес:** користувачі завантажують програму та вирішують створити новий обліковий запис. Додаток збирає інформацію про інші облікові записи та кредитні рейтинги користувача.

Звідти він автоматично ділиться обліковими записами, на які вони мають право, і їхніми процентними ставками. Користувач знаходить обліковий запис, який відповідає його потребам, і реєструється.

- **Альтернативний потік 1:** користувачі вводять свою фінансову інформацію, і програма швидко генерує параметри облікового запису. Проте за замовчуванням для кожного облікового запису встановлено найвищу відсоткову ставку, яку дозволяє їхнє фінансове становище. Отже, користувачі залишають програму, щоб знайти нижчу ставку.

- **Альтернативний потік 2:** процес адаптації працює належним чином, але програма стикається з проблемами відповідності, як-от вимоги «Знай свого клієнта». Хоча програма може надавати параметри облікового запису, додаткові кроки відповідності сповільнюють процес.

- **Альтернативний потік 3:** оскільки програма розглядає лише існуючі облікові записи та кредитні рейтинги, то вона не може запропонувати повний спектр параметрів облікового запису. Наприклад, вона може пропонувати лише кредитні картки та кредитні лінії. Отже, клієнтам, які шукають іпотеку, доводиться йти до інших програм. Для упорядкування текстової інформації використовують табличне представлення опису варіанту використання.

Можна використовувати зразок опису Use Case, наведений у табл.1, для створення специфікації варіантів використання [10].

Наприклад, можна використовувати шаблон у вигляді таблиці від [Алістера Кокберна](#), який наведено у табл.2 для варіанту використання “Зняття готівки” [6].

Отже, можна використовувати різні варіанти опису вимог у вигляді варіантів використання - Use Case, в залежності від проєкту та прийнятих шаблонів у ІТ-компанії.

Ось посилання на джерела, де пропонуються різні шаблони та приклади для документування варіантів використання:

- <https://templatelab.com/use-case-templates/>
- <https://www.projectmanagementdocs.com/template/project-documents/use-case-document/>
- <https://guides.visual-paradigm.com/demystifying-use-cases-scenarios-flow-of-events-and-templates/>
- <https://docs.google.com/document/d/1GD83XXdIBxI6iNBAOVLtG-lIcMHqS11QtaMKroxrJFs/edit?tab=t.0>
- <https://www.overleaf.com/latex/templates/use-case-specification/zdwhztshzgmxd>

Таблиця 1. Зразок схеми специфікації варіанту використання

Розділ	Опис
Назва випадку використання	Вказує назву випадку використання. Як правило, назва виражає об'єктивний або видимий результат використання, наприклад « Зняти готівку » у випадку банкомата.
Короткий опис	Описує роль і призначення варіанту використання.
Перебіг подій:	Представляє основний потік і альтернативні потоки. Потік подій описує поведінку системи; він не описує, як працює система, деталі презентації або деталі інтерфейсу користувача. Якщо відбувається обмін інформацією, варіант використання має бути конкретним щодо того, що передається перед його виконанням та після нього. Наприклад, замість того, щоб описувати дію як «актор вводить інформацію про клієнта», вкажіть, що «актор вводить ім'я та адресу».
- основний потік	Описує ідеальну, первинну поведінку системи.
- альтернативні потоки	Описує винятки або відхилення від основного потоку, наприклад, як поводить система, коли актор вводить неправильний ідентифікатор користувача, а автентифікація користувача не вдається.
Особливі вимоги	Нефункціональні вимоги, які є специфічними для варіанту використання, але не вказані у тексті потоку подій Use Case. Приклади спеціальних вимог: законодавчі та нормативні вимоги; стандарти застосування; атрибути якості системи, включаючи зручність використання, надійність, продуктивність і можливість підтримки; операційні системи та середовища; вимоги сумісності; і проектні обмеження.
Передумови	Стан системи, який має бути присутнім перед початком реалізації варіанту використання.
Постумови	Список можливих станів для системи одразу після завершення сценарію використання
Точки розширення	Точка в потоці подій варіанта використання, у якій посилаються інший варіант використання.

Таблиця 2 – Приклад специфікації випадку використання

Use Case:	Зняти готівку
Актор(и):	Клієнт (основний), банківська система (вторинний)
Короткий опис:	Дозволяє будь-якому клієнту банку знімати готівку зі свого банківського рахунку.
Пріоритет:	Must Have (Необхідно мати у реалізації)
Статус:	Середній рівень деталізації
Попередня умова:	У клієнта банку є картка, яку потрібно вставити до банкомату. Банкомат працює нормально
Постумова (постумови):	Клієнт банку отримав готівку (і, за бажанням, квитанцію) Банк списав кошти з рахунку клієнта та записав деталі операції
Основний потік подій:	<ol style="list-style-type: none"> 1. Клієнт вводить свою картку в банкомат. 2. Банкомат перевіряє, чи є картка дійсно банківською. 3. Банкомат запитує PIN-код. 4. Клієнт вводить свій PIN-код. 5. Банкомат перевіряє банківську картку за пін-кодом. 6. Банкомат надає такі послуги, як "Зняти". 7. Клієнт вибирає «Вивести». 8. Банкомат пропонує варіанти сум. 9. Клієнт вибирає суму або вводить її. 10. Банкомат перевіряє, чи достатньо готівки в бункері. 11. Банкомат перевіряє, чи клієнт перевищив ліміт зняття. 12. Банкомат перевіряє наявність коштів на рахунку клієнта. 13. Банкомат списує кошти з банківського рахунку клієнта. 14. Банкомат повертає клієнту банківську картку. 15. Клієнт забирає свою банківську картку. 16. Банкомат видає клієнту готівку. 17. Клієнт забирає готівку.
Альтернативні потоки подій:	<ol style="list-style-type: none"> 2а. Недійсна картка. 2б. Картка догори дном. 5а. Вкрадена картка. 5б. PIN-код недійсний. 10а. Недостатньо готівки в бункері. 10б. Неправильний номінал готівки в бункері. 11а. Зняття понад лімітів. 12а. Недостатньо коштів на банківському рахунку клієнта. 14а. Банківська картка застрягла в автоматі. 15а. Клієнт не бере банківську картку. 16а. Готівка застрягла в автоматі. 17а. Клієнт не бере готівку: <ol style="list-style-type: none"> 1) банкомат не може зв'язатися з банківською системою; 2) клієнт не відповідає на запит банкомату.

Продовження табл. 2

Use Case:	Зняти готівку
Бізнес-правила:	B1: Формат PIN-коду. B2: Кількість повторів PIN-коду. B3: Варіанти обслуговування. B4: Варіанти суми. B5: Ліміт зняття. B6: Картку необхідно забрати перед видачею готівки.
Нефункціональні вимоги:	NF1: Час для завершення транзакції. NF2: Безпека для введення PIN-коду. NF3: Час дозволити отримання картки та готівки. NF4: Підтримка мови. NF5: Сліпа та частково сліпа підтримка.

У даній контрольній роботі будемо використовувати шаблон опису Use Case (<https://www.softwarereqs.com/>), запропонований К. Вігерсом у роботі [11]. Далі розглянемо опис основних розділів цієї специфікації.

Для системного представлення усіх вимог, які описуються у Use Cases специфікації рекомендовано попередньо представити Use Case діаграму у нотації UML [7, 9], яку можна побудувати за допомогою різних он-лайн сервісів, наприклад:

Draw.io [<https://drawio-app.com/blog/uml-use-case-diagrams-with-draw-io/>]

Visual Paradigm [<https://online.visual-paradigm.com/diagrams/features/uml-tool/>]

Lucidchart [<https://www.lucidchart.com/pages/uml-use-case-diagram>]

Edraw Max [<https://www.edrawmax.com/article/use-case-diagram-uml.html>]

Visio [<https://support.microsoft.com/en-us/office/uml-diagrams-in-visio-ca4e3ae9-d413-4c94-8a7a-38dac30cbcd6>] та ін.

1.2.3 Опис розділів Use Cases специфікації

Ідентифікатор та ім'я варіанту використання (Use Case ID and Name). Надайте кожному випадку використання унікальний числовий ідентифікатор. Укажіть стисло назву випадку використання, яка вказує на цінність, яку цей варіант використання надасть певному користувачеві. Починайте з дієслова дії, за яким слідує предмет.

Автор і дата створення (Author and Date Created). Введіть ім'я особи, яка спочатку написала цей приклад використання, і дату його написання.

Це потрібно для подальшого управління вимогами.

Первинні та вторинні актори (Primary and Secondary Actors).

Суб'єкт – це особа або інший суб'єкт, зовнішній по відношенню до вказаної програмної системи, який взаємодіє з системою і виконує випадки використання (use cases) для досягнення чогось цінного. Різні актори часто відповідають різним класам користувачів або ролям. Назвіть основного актора, який ініціюватиме цей випадок використання, та будь-яких інших вторинних акторів (людей або системи), які братимуть участь у виконанні варіанту використання.

Опис (Description). Надайте короткий опис причини та результату цього випадку використання, або високорівневий опис послідовності дій та результату виконання варіанту використання.

У цьому описі має бути пояснено функціональні можливості, які надає варіант використання, і додаткову цінність для користувача, який його активує. Діаграма варіантів використання описує функціональні можливості з точки зору користувача, тому це також застосовуватиметься до специфікації UC. Зразок опису може бути таким: «Цей варіант використання дозволяє надрукувати вибраний документ» [12].

Тригер (Trigger). Визначте подію або дію користувача, яка ініціює варіант використання. Цей тригер попереджає систему про необхідність протестувати попередні умови, щоб вона могла вирішити, чи продовжувати виконання.

Передумови (Preconditions). Кожен варіант використання може мати певні умови, які повинні бути виконані для його виконання. Перелічіть будь-які дії, які повинні відбутися, або будь-які умови, які повинні бути виконані, перш ніж можна буде почати використання Use Case. Система повинна мати можливість тестувати кожну передумову.

Правильно встановлений принтер або підключення до Інтернету можуть бути прикладами попередніх умов. Без цих вимог Use Case не буде мати сенсу й не виконуватиметься.

Пронумеруйте кожну передумову.

Приклад: PRE-1. Особистість користувача була автентифікована.

Постумови (Postconditions). Опишіть стан системи при успішному завершенні виконання варіанту використання.

Наприклад, постумовами можуть бути, відсутність помилок або те, що всі помилки зареєстровано в журналі помилок.

Source: <https://www.ictdemy.com/software-design/uml/uml-use-case-specification>

Позначте кожну умову поста у вигляді POST-X, де X – порядковий номер.

Приклад: Post-1. Ціна товару в базі даних оновилася новим значенням.

Нормальний потік (Normal Flow). Надайте опис дій користувача та відповідні системні реакції, які відбудуться під час виконання варіанту використання за нормальних, очікуваних умов. Ця послідовність діалогу приведе до досягнення мети, зазначеної в назві та описі випадку використання.

Базовий потік не передбачає можливих помилок і описує плавний хід, коли варіант використання досягне своєї мети на останньому кроці.

Покажіть пронумерований список дій, виконаних актором, що чергуються з відповідями, наданими системою.

Нормальний потік має номер «X.0», де «X» — це ідентифікатор випадку використання.

Альтернативні потоки (Alternative Flows). Задokumentуйте інші успішні варіанти використання, які можуть мати місце у цьому випадку використання. Сформулюйте альтернативний потік і опишіть будь-які відмінності у послідовності кроків, які відбуваються.

Це збої чи помилки, допущені користувачем (неправильно введений пароль) або системою (не вдалося надрукувати документ). Альтернативний потік завжди посилається на певний крок основного потоку та вирішує його нестандартну поведінку. Це відноситься до кроку основного потоку, де основний потік продовжується.

Отже, пронумеруйте кожен альтернативний потік у формі «X.Y», де «X» — це ідентифікатор випадку використання, а Y — порядковий номер альтернативного потоку. *Наприклад, «5.3» укаже на третій альтернативний потік для випадку використання під номером 5.* Вкажіть, де кожен альтернативний потік буде відгалужуватися від нормального потоку, і якщо це доречно, де він знову приєднується до нормального потоку.

Винятки (Exceptions). Опишіть будь-які очікувані умови помилки, які можуть виникнути під час виконання варіанту використання, і як система повинна реагувати на ці умови. Пронумеруйте кожен альтернативний потік у формі «X.Y.EZ», де «X» — це ідентифікатор випадку використання, Y — звичайний (0) або альтернативний (>0) потік, під час якого може відбутися цей виняток, «E» — виняток, а «Z» — порядковий номер для винятків. *Наприклад, «5.0.E2» укаже на другий виняток для нормального потоку для випадку використання*

номер 5. Вкажіть, де в нормальному (або альтернативному) потоці може виникнути кожен виняток.

Пріоритет (Priority). Вкажіть відносний пріоритет реалізації функціональних можливостей, необхідних для виконання цього випадку використання. Використовуйте ту саму схему пріоритету, що використовується для інших вимог.

Вимоги можуть мати пріоритет згідно моделі MoSCoW (табл.3.)

Таблиця 3 – Пріоритети вимог за моделлю MoSCoW

Клас	Опис пріоритету
П – Повинно бути (M – Must Have)	Опис вимог, що повинні бути реалізовані обов'язково у рішенні. Це критичні вимоги, без яких проект не може бути завершений. Якщо вони не виконуються, проект вважається невдалим.
В – Варто було б мати (S – Should Have)	Це високо-пріоритетні вимоги, які повинні бути виконані у рішення, якщо це можливо. Дуже часто це важливі вимоги, проте кожна з них може бути реалізована у наступній версії продукту. Отже, це не настільки критичні вимоги до ПЗ, вони не є чутливими до часу.
М – Можливо мати (C – Could Have)	Це вимоги, які вважаються бажаними, але не обов'язковими. Вони будуть включені, якщо дозволять час і ресурси.
Х – Хотілося б мати, але не зараз (W – Won't Have)	Ці функції мають найнижчий пріоритет або не потрібні для поточного циклу доставки. Це вимоги, які були погоджені з зацікавленими сторонами, вони не будуть додаватися до розробки у проект, проте можуть бути розглянуті у подальшому.

Бізнес-правила (Business Rules). Перелічіть будь-які бізнес-правила, які впливають на цей варіант використання. Можна не включати сюди текст бізнес-правила, а лише його ідентифікатор, щоб можна було знайти його в іншому документі. Існує два основних типи бізнес-правил: орієнтована на бази даних та орієнтована на додатки. Обидва типи накладають форму обмеження та допомагають забезпечити та підтримувати загальну цілісність даних. Однак вони різняться щодо того, де і як вони включені в систему.

Приклади бізнес-правил та вимоги, з яких вони виникають:

*Вимога: Працівник складу повинен мати можливість відправляти замовлення, щоб забезпечити покупців покупками.
Бізнес-правило: дата доставки не може передувати даті замовлення.*

Вимога: Співробітник оренди повинен мати можливість призначити автомобіль орендареві, щоб той міг їздити. Правило

бізнесу: Орендар не може орендувати більше, ніж один автомобіль одночасно.

Вимога: Покупець повинен мати можливість розміщувати замовлення без втручання інших. Бізнес-правило: Він може вести бізнес лише з постачальниками з регіону. Він також може самостійно робити покупки до певної суми.

Вимога: Продавець може надавати знижку покупцям для сприяння задоволенню споживачів. Бізнес-правило: Знижку можуть отримати лише довгострокові клієнти. Знижка також становить максимум 15% на всі покупки.

Припущення (Assumptions). Перелічіть будь-які припущення, які були зроблені щодо цього випадку використання або як цей Use Case може працювати.

Додаткова інформація (Other Information). Визначте будь-які додаткові відповідні вимоги або обмеження для випадку використання, такі як **атрибути якості**.

Опишіть, що має статися, якщо виконання варіанту використання не вдається з якоїсь непередбаченої або системної причини (наприклад, втрата доступу до мережі, тайм-аут). Якщо випадок використання призводить до тривалої зміни стану в базі даних або у зовнішньому середовищі, повідомте, чи зміна відкочується назад, завершена правильно, частково завершена з відомим станом або залишена у невизначеному стані у результаті винятку.

1.2.4 Зміст звіту з виконання КР №1

Після виконання КР №1 студентом складається звіт за результатами.

Звіт повинен містити:

- 1) титульний аркуш (приклад у додатку А);
- 2) основну частину, а саме:
 - мета роботи;
 - короткий виклад виконання роботи: призначення ПЗ, що розробляється, use case специфікація з обов'язковим представленням use case діаграми (або use case діаграм для кожного користувача ПЗ);
 - висновки (що було вивчено, які завдання вирішені і які отримані навички практичної роботи)

1.3 Методичні рекомендації до виконання КР № 2

1.3.1 Завдання до КР № 2 та етапи виконання

Мета – засвоєння техніки розробки бізнес-процесу для програмного рішення, що проектується в межах дипломної роботи

Етапи виконання контрольної роботи №2:

3) надати опис основного бізнес-процесу, який досліджується в межах дипломної роботи (або автоматизація якого здійснюється за допомогою ПЗ, або частка такого бізнес-процесу);

4) розробити бізнес-процес предметної області, яка розглядається в межах дипломної роботи та які будуть реалізовані у програмному забезпеченні, використовуючи нотацію BPMN;

5) оформите звіт за результатами виконання контрольної роботи.

6) надати до перевірки результати контрольної роботи викладачу на практичному занятті, після схвалення результатів додати звіт до системи Moodle.

1.3.2 Відомості про Business Process Model and Notation

BPM (Business Process Management, управління бізнес-процесами) — концепція процесного управління організацією, яка розглядає бізнес-процеси як особливі ресурси підприємства, що безперервно адаптуються до постійних змін. Основні принципи даної концепції — зрозумілість і прозорість бізнес-процесів. Досягається це за рахунок їх моделювання з використанням формальних нотацій, використання програмного забезпечення для симуляції, моніторингу, моделювання та аналізу бізнес-процесів, динамічного перестроювання моделей бізнес-процесів силами персоналу і засобами програмних систем.

Нотація моделі бізнес-процесів (BPMN) – це стандартний спосіб графічного представлення бізнес-процесів. Програма Visio містить шаблон, який містить графічні елементи, описані в специфікації BPMN 2.0 після класу відповідності аналітики [13].

BPMN ([англ. Business Process Model and Notation](#), *модель та нотація бізнес-процесів*) — система умовних позначень ([нотація](#)) для моделювання [бізнес-процесів](#). Розроблена Business Process Management Initiative (BPMI) та підтримується [Object Management Group](#) після їх злиття в 2005 році. Остання версія BPMN — 2.0, що була прийнята у 2014 року.

Моделювання з використанням BPMN виконується у вигляді діаграм, що складаються з різних елементів. Розрізняють чотири категорії елементів [<https://www.wikiwand.com/uk/BPMN>]:

- 1) об'єкти потоку керування: дії, події та логічні оператори;
 - 2) поєднуючі елементи: потік керування, потік повідомлень та асоціації;
 - 3) ролі: пули та доріжки;
 - 4) артефакти: дані, групи та текстові анотації.
- Позначення дій та задач наведено на рис. 5.



Рисунок 5 – Позначення дій та задач

Дії.

Задача — одиниця роботи. Якщо задача є підпроцесом, то вона може бути деталізована.

Транзакція — набір логічно пов'язаних дій. Для транзакції може бути визначений протокол виконання.

Подієвий підпроцес розміщується всередині іншого процесу. Він починає виконуватися, якщо ініціюється його початкова подія. Подієвий підпроцес може переривати батьківський підпроцес або виконуватися паралельно з ним.

Викликаюча дія є точкою входу для глобально визначеного підпроцесу, що повторно використовується в даному процесі.

Типи подій.

Проста: нетипізована подія, зазвичай вказує на початок, зміну стану, чи завершення процесу.

Повідомлення: Отримання і відправлення повідомлень.

Таймер: циклічні події, моменти часу, часові періоди і тайм-аути.

Ескалація: перенесення розгляду задачі на більш високий рівень організаційної ієрархії.

Умовна: реакція на зміну бізнес-умов або інтеграція бізнес-правил.

Посилання: пара відповідних посилань, що еквівалентна неперервному потоку керування.

Помилка: генерація й обробка помилок заданого типу.

Скасування: обробка скасування транзакції або ініціювання скасування.

Компенсація: обробка або ініціювання компенсації.

Сигнал: передається між процесами та може оброблятися декількома одержувачами одночасно.

Складена: обробка однієї події із багатьох або генерація всіх визначених подій.

Паралельна складена: обробка всієї множини паралельних подій.

Зупинник: викликає негайне припинення виконання процесу.

Логічні оператори наведено на рис. 6.



Рисунок 6– Логічні оператори

Логічні оператори.

Оператор виключаючого АБО, що керується даними. При розгалуженні оператор активує один із вихідних потоків. При об'єднанні — очікує завершення одного вхідного потоку і активує вихідний потік.

Оператор виключаючого АБО, що керується подіями. Передує тільки обробляючим подіям або завданням отримання повідомлення. Виконується тільки той потік, де подія сталася раніше.

Оператор I. При розгалуженні оператор активує всі вихідні потоки. При об'єднанні — очікує завершення всіх вхідних потоків і активує вихідний потік.

Оператор АБО. При розгалуженні активує один або більше вихідних потоків. При об'єднанні всі запущені вхідні потоки повинні бути завершені.

Складний оператор. Моделює складні умови розгалуження та злиття.

Оператор виключного АБО, що керується подіями (створює новий екземпляр). Настання кожної подій, яким передує оператор, створює екземпляр процесу.

Оператор *I*, що керується подіями (створює новий екземпляр). Настання всіх подій, яким передує оператор, створює екземпляр процесу.

Потоки керування.

Потік керування визначає порядок виконання дій.

Потік за умовчанням визначає потік, що буде виконаний, якщо умови всіх інших потоків розгалуження невірні.

Умовний потік визначає потік, що буде виконаний, якщо пов'язана з цим потоком умова вірна.

Ролі.

Пули (учасники) і доріжки відображають розподіл обов'язків. Пул або доріжка позначає організацію, роль або систему. Доріжки дають змогу ієрархічно поділяти пули та інші доріжки.

Потік повідомлень описує інформаційний потік між учасниками процесу. Потік повідомлень може приєднуватися до пулів, дій чи подій-повідомлень.

Порядок обміну повідомленнями може бути заданий за допомогою потоку повідомлень і потоку керування.

Дані.

Вхідні дані — це вхідний параметр процесу. *Вихідні дані* — результат виконання процесу (вихідний параметр). Під час виконання дії використовують вхідні дані та продукують вихідні дані.

Об'єкт даних представляє інформацію, що оброблюється в ході процесу, наприклад документ або лист.

Колекція об'єктів даних представляє групу об'єктів, що несуть інформацію, наприклад перелік замовлених товарів.

Дані наведено на рис. 7.



Рисунок 7 – Дані

Сховище даних — це об'єкт, який процес може використовувати для запису та вибірки даних, наприклад база даних. Сховище даних дає змогу зберігати дані після закінчення життєвого циклу екземпляра процесу.

Повідомлення дає змогу явно продемонструвати передачу інформації в ході спілкування двох учасників. Біле повідомлення надсилається ініціатором спілкування, сіре — іншим учасникам.

1.3.3 Рекомендації щодо моделювання бізнес-процесу

Розуміння доріжок і басейнів. Полотно для карти, намальованої за допомогою BPMN, складається з смуг (доріжок) і басейнів (lanes and pools) (рис.8- <https://www.process.st/bpmn-tutorial/>)

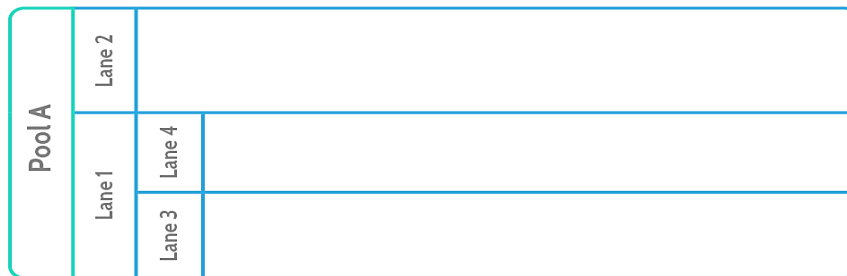


Рисунок 8 – Доріжки та басейн

Pools. Пули представляють різні організації або цілком окремі процеси. Доріжки представляють різні команди або окремих осіб в одній організації.

Пули є найбільшою одиницею на карті: це області, які містять доріжки, події, завдання тощо. Вони представляють, хто виконує завдання. Наприклад, для процесу підтримки клієнтів, був би один пул для Process Street і один для клієнта. Робота може проходити між пулами, означаючи щось на зразок надсилання клієнту повідомлення.

Точне використання басейнів(пулів) і доріжок може бути справою смаку. Незалежно від того, чи використовуєте ви окремі пули для представлення різних відділів компанії, чи просто обираєте використання смуг, є кілька жорстких правил:

- кожен пул може містити максимум один процес;
- пули представляють учасників: компанії, клієнти або відділи.

Lanes. Часто процеси потребують двох або більше відділів для взаємодії один з одним. Наприклад, простий процес виконання замовлення може виглядати так:

- клієнт робить замовлення по телефону за допомогою кредитної картки;
- замовник стягує плату з кредитної картки;
- клерк записує sku та дані клієнта;
- клерк надсилає на склад електронною поштою sku та деталі;
- товар запаковано з етикеткою;
- відправка поштою клієнту.

На рис.9 цей процес наведено у BPMN.

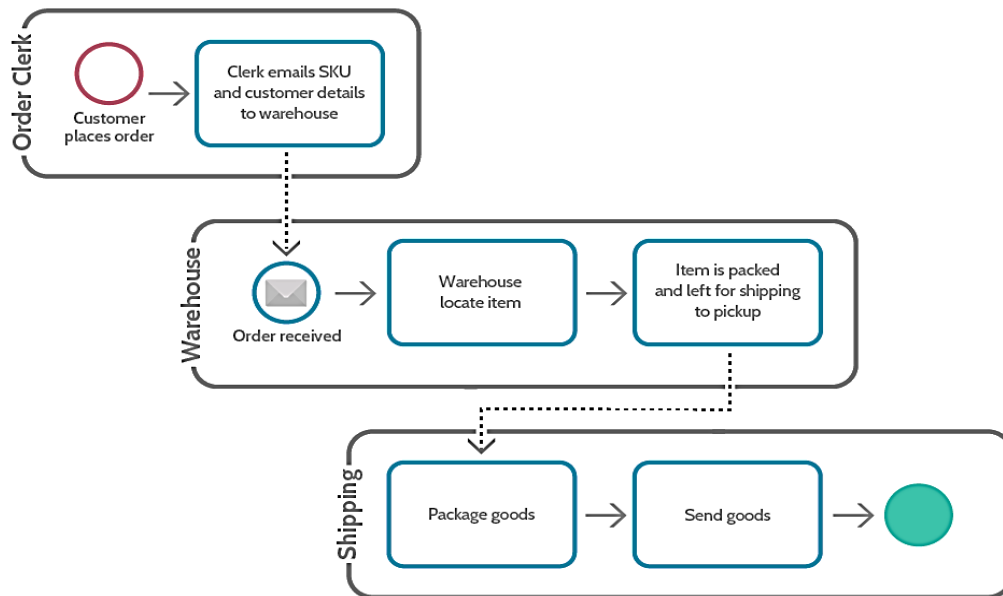


Рисунок 9 – Використання доріжок (lanes)

Основне полотно (або пул) карти процесу можна розділити на ряди (або доріжки). Кожна доріжка представляє одну паралель процесу та показує як точні завдання та обов'язки кожного відділу, так і взаємодію між ними.

Доріжки не завжди означають відділи чи людей. Відповідно до посилання Camunda (<https://camunda.com/bpmn/reference/>), вони можуть бути:

- посади в первинній організації, наприклад, бухгалтер;

- ролі у вторинній організації, наприклад, спеціаліст із захисту даних;
- загальні ролі, наприклад, клієнт;
- відділи, наприклад, продажів;
- іт-додатки, наприклад, система CRM;
- розрізнення ручних і автоматичних кроків.

На рис.10 представлена форма повідомлення, яка означає інформацію, що передається між двома діями, а заштриховане коло означає кінець процесу.

Основні форми BPMN дуже схожі на блок-схеми (рис.4.12), але замість того, щоб представляти чисті логічні етапи, вони можуть представляти специфічні для бізнесу дії, як-от обмін повідомленнями з іншими відділами та ескалація проблем.

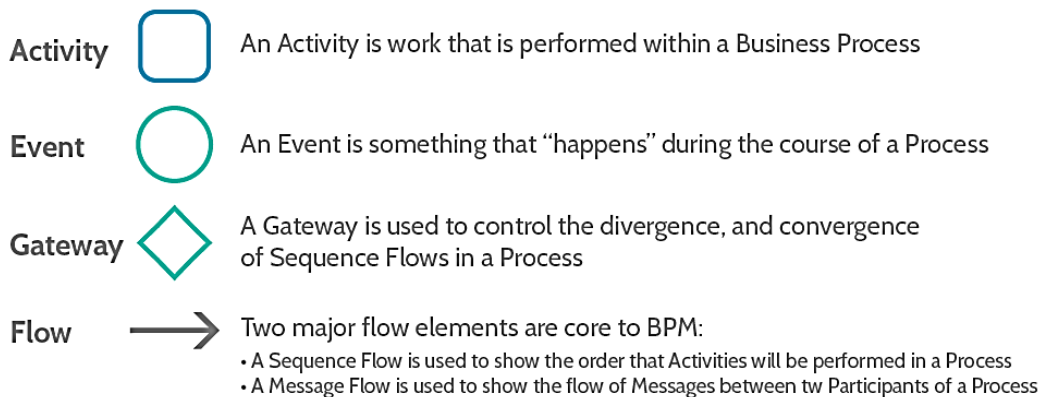


Рисунок 10 – Основні форми BPMN

Дії. Просто прямокутник із закругленими кутами. Вони представляють роботу, яку потрібно виконати, як прапорець у контрольному списку або пункт у списку справ.

З'єднувальні об'єкти. Оскільки BPMN має на меті показати потік дій, він використовує три різні типи з'єднувальних об'єктів, які показують, як дії пов'язані одна з одною (рис.11)

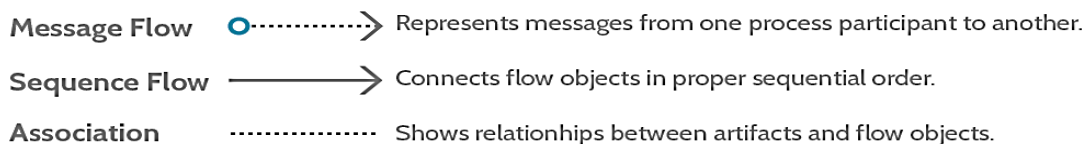


Рисунок 11 – З'єднувальні об'єкти

Sequence flow – основна лінія, яка з'єднує елементи вашої карти. Він показує хід роботи та необхідний для з'єднання видів діяльності. Без потоку послідовності ваша карта недійсна.

Потік повідомлень (Message flow) використовується, коли різні відділи чи організації передають інформацію один одному. Оскільки ви не встаєте й не йдете контролювати роботу в різних відділах, коли хочете виконати завдання, потік повідомлень представляє запит або обмін інформацією, а не сувору дію. Це єдиний тип потоку, який може відбуватися між пулами або доріжками (організаціями чи відділами).

Асоціація (Association) використовується для зв'язування документів, баз даних та інших артефактів із діяльністю. Наприклад, якщо у вас була операція з підписання замовлення на купівлю, ви б використали символ документа та лінію асоціації, щоб зв'язати їх разом.

Приклад надано на рис.12.

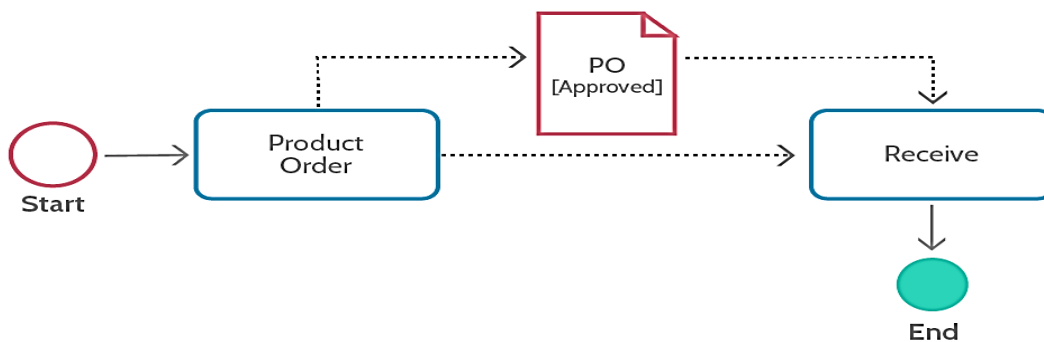


Рисунок 12 – Дії та з'єднувальні об'єкти

Шлюзи мають справу з логікою/рішеннями. Подібно до ромбів у звичайних блок-схемах, шлюзи представляють розбиття або конвергенцію в діаграмі процесу (рис.13).



Рисунок 13 – Шлюзи

Наприклад, коли ви вибираєте, що їсти, процес розділиться залежно від остаточного вибору їжі, але зійдеться в точці, де їжа з'їдена,

тому що незалежно від того, яку їжу вибрано, вона врешті-решт буде з'їдена.

Шлюзи подій. Однак не всі шлюзи можна вибрати. Можливо, продовження процесу залежить від отримання додаткової кількості даних, очікування певного часу чи виконання інших умов. У цьому випадку він представлений як шлюз подій і розгалужується на будь-яку наступну подію. Наприклад, якщо ви чекаєте на доставку більше тижня, ви можете дійти до моменту, коли ви вирішите, чи чекати терпляче, чи зателефонувати транспортній компанії. На цьому етапі у вас буде шлюз на основі подій, який розгалужуватиметься на очікування більше часу або на виклик. Приклад на рис. 14.

На рис.14 зверніть увагу на форми часу та повідомлення. Час може бути днями/годинами/хвилинами або конкретними датами/роками/кварталами.

Повідомлення – це будь-яка інформація, яка надсилається певному одержувачу (особі, відділу, організації тощо). Це можуть бути дані, файли, електронні листи, телефонні дзвінки, листи тощо.

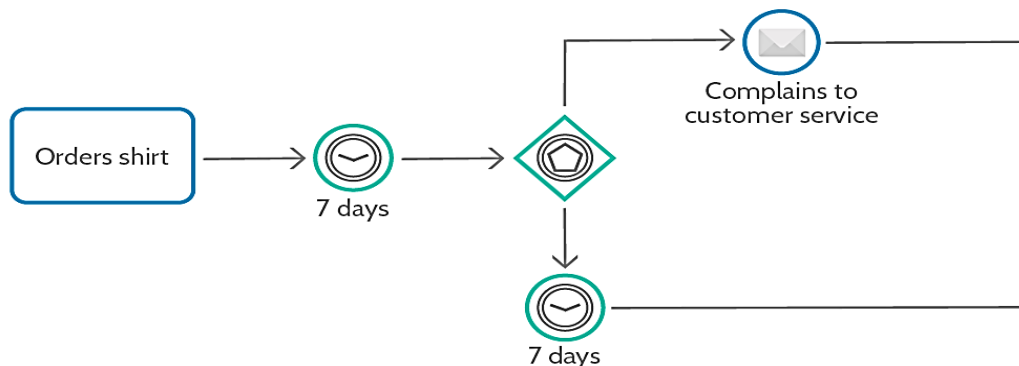


Рисунок 14 – Шлюз на основі подій

Паралельні шлюзи. Якщо завдання не залежить від іншого завдання, немає причин не виконувати їх паралельно на початку процесу. Щоб відобразити це на карті BPMN, використовуйте паралельний шлюз (рис.15). Це означає, що тепер є шанс для запуску обох наступних завдань: процес розбивається на дві паралельні гілки.

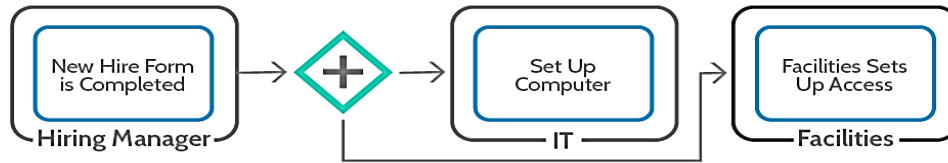


Рисунок 15 – Використання паралельного шлюзу

Ексклюзивні шлюзи. Іноді процес може йти лише **одним** із двох (або будь-якою кількістю) шляхів. Наприклад, запис подкасту приймається та переходить на етап редагування, або відхиляється та повертається до редактора для повторного проходу.

На рис.16 приклад того, коли використовується ексклюзивний шлюз, оскільки третього варіанту немає.

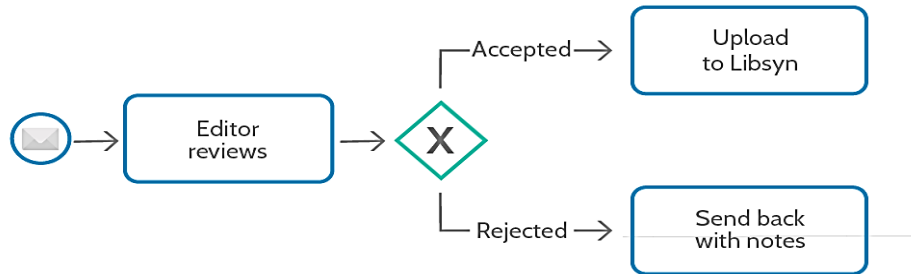


Рисунок 16 – Використання ексклюзивного шлюзу

Інклюзивні шлюзи. Інклюзивний шлюз розбиває процес на кілька гілок із поясненням того, яка умова повинна бути виконана, щоб потік продовжувався вниз по цій гілці. Приклад, наведений на рис.17 для пояснення інклюзивного шлюзу, — це процес, який розгалужується на основі відповіді конкретного клієнта на опитування. Пояснення: «Один процес запускається, якщо споживач задоволений продуктом А. Інший потік запускається, коли споживач вказує, що він задоволений продуктом В. Третій процес запускається, якщо він не задоволений продуктом А. Буде один мінімальний потік та максимум два».

Інклюзивні шлюзи дозволяють вибрати кілька можливих результатів. Наприклад, клієнт може бути як «не задоволений А», так і «задоволений Б», і в такому випадку йому буде надіслано ваучер, а також буде додано до списку Б.

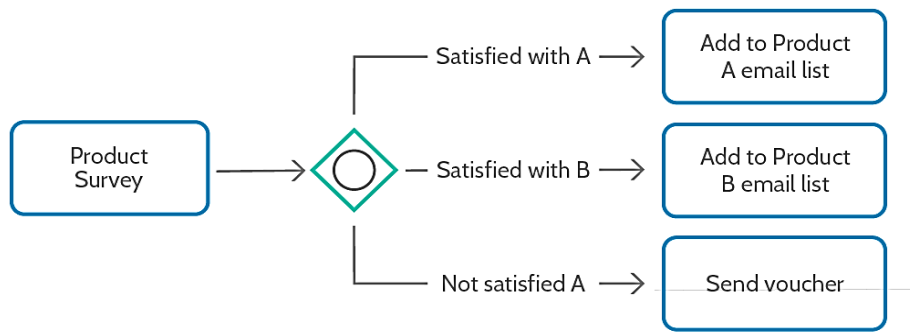


Рисунок 17 – Приклад інклюзивного шлюзу

Приклад бізнес-процесу, створеного за допомогою Draw.io з використанням основних форм, наведено на рис.18. У цьому прикладі є два басейни з однією доріжкою кожен. Процес запускає клієнт, який надсилає заявку в службу підтримки компанії. Підтримка відповідає, а потім чекає два дні. Якщо клієнт не відповів, заявка закрита. Якщо клієнт задоволений, тикет закривається. В іншому випадку спілкування продовжується, доки не буде виконано одну з цих умов.

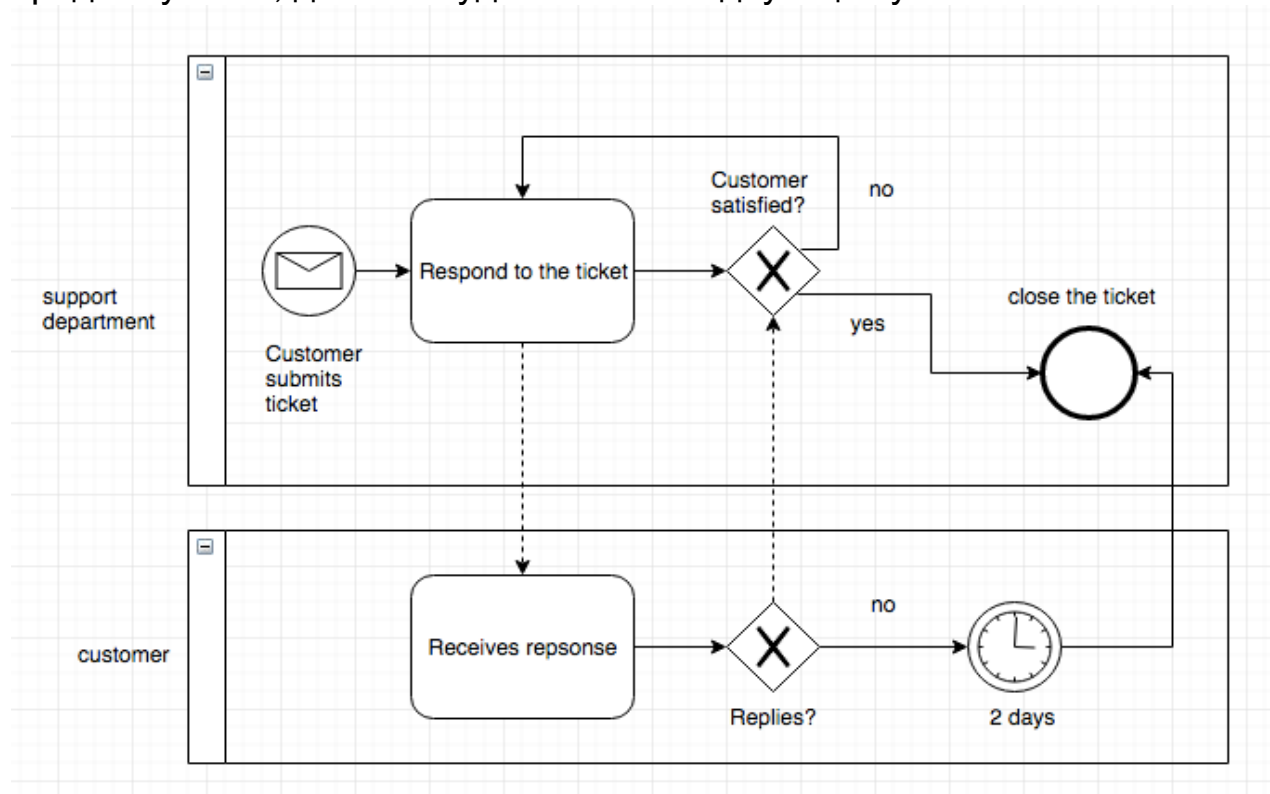


Рисунок 18 – Приклад бізнес-процесу

1.3.4 Приклади створення бізнес-процесів

Правила розробки бізнес-процесу у Lucidchart дивись за [ПОСИЛАННЯМ https://www.visual-paradigm.com/support/documents/vpuserguide/2821/286/7114_drawingbusin.html](https://www.visual-paradigm.com/support/documents/vpuserguide/2821/286/7114_drawingbusin.html) та [та https://www.visual-paradigm.com/tutorials/as-is-to-be-business-process.jsp](https://www.visual-paradigm.com/tutorials/as-is-to-be-business-process.jsp). Приклад створення бізнес-процесу у Lucidchart наведено на рис. 19.

Представлення BPMN 2.0 у Visio можна подивитися за [ПОСИЛАННЯМ https://www.microsoft.com/en-us/microsoft-365/blog/2012/11/19/introducing-bpmn-2-0-in-visio/](https://www.microsoft.com/en-us/microsoft-365/blog/2012/11/19/introducing-bpmn-2-0-in-visio/) та [та https://www.modernanalyst.com/Resources/Articles/tabid/115/ID/2533/Democratizing-BPM-With-Visio-and-Office-365.aspx](https://www.modernanalyst.com/Resources/Articles/tabid/115/ID/2533/Democratizing-BPM-With-Visio-and-Office-365.aspx)

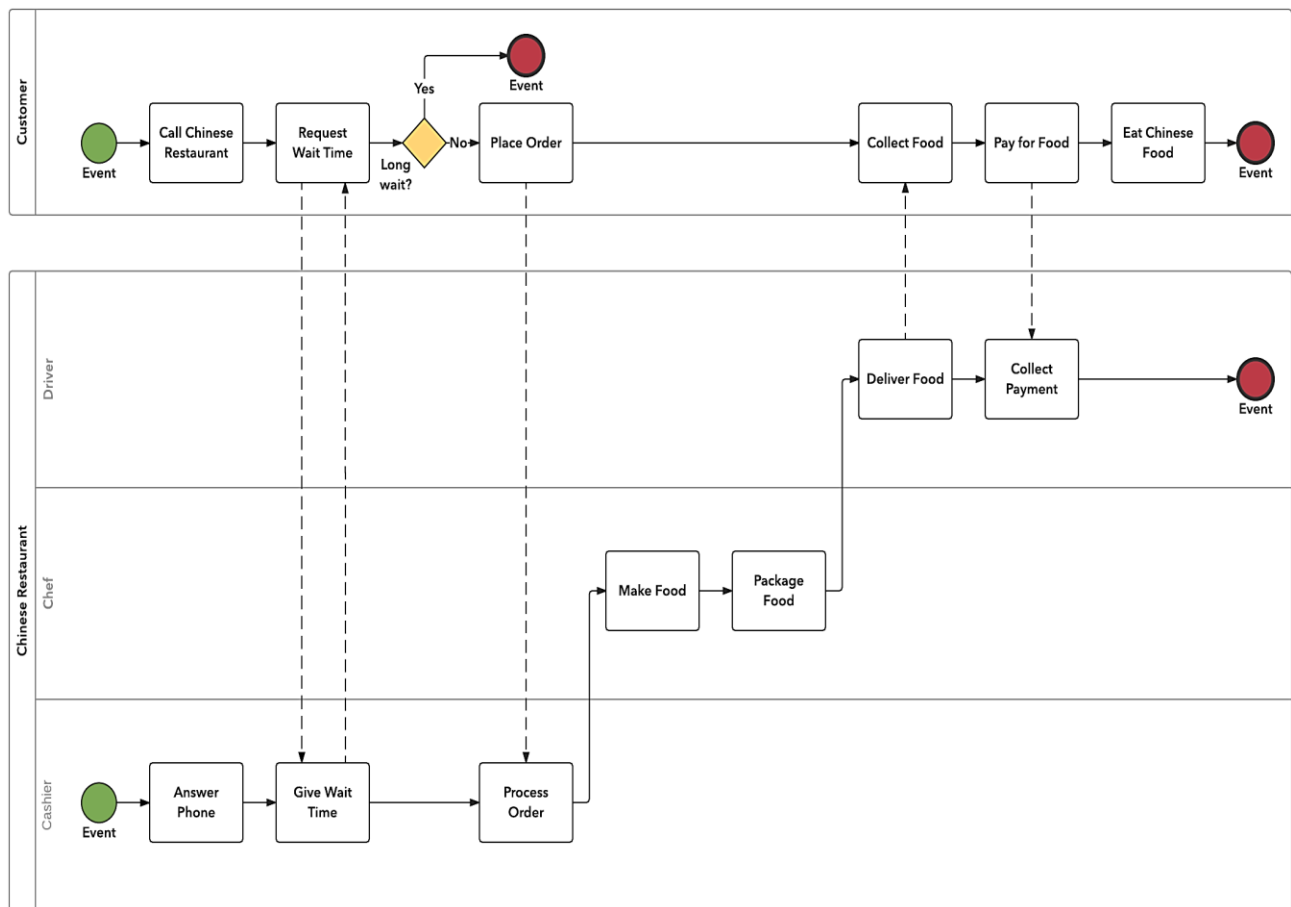


Рисунок 19 – Приклад створення бізнес-процесу у Lucidchart

Приклад BPMN- бізнес-процес веб-магазину.

Одним із основних видів діяльності інтернет-магазину є виконання замовлень. На високому рівні ми можемо виділити наступні кроки в процесі замовлення:

1. Отримати замовлення,
2. перевірити кредит,
3. Виконати замовлення і
4. відправити рахунок-фактуру.

Почнемо з візуалізації цих кроків у простій моделі BPMN. Модель має виглядати як на рис.20.

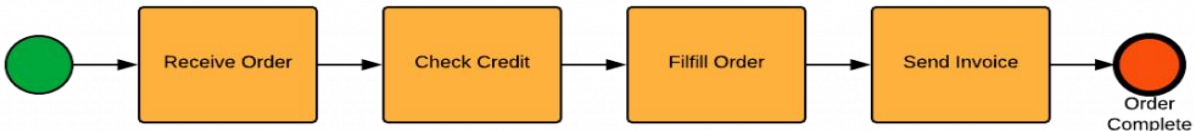


Рисунок 20 – Загальний опис процесу

Наведена діаграма BPMN ще не є реальною моделлю процесу. Реальна модель процесу показує всі шляхи потоку діяльності від початку до кінця. Отже, якщо є дії, які можна виконати успішно чи неуспішно, це означатиме, що модель повинна включати шляхи для обох умов.

У моделі, яку створили, є два екземпляри, які можуть і повинні бути перевірені шлюзом: дія «перевірити кредит» і «виконати замовлення» (рис. 21). Якщо будь-яка з цих двох дій не вдається, замовлення не виконується. Тому ми повинні додати до діаграми ще один можливий кінцевий стан: замовлення не вдалося. Два кінцевих стану означають дві кінцеві події, позначені назвою кінцевого стану. Це не є частиною специфікації BPMN, але є частиною найкращої практики. Це допомагає зробити речі простими.

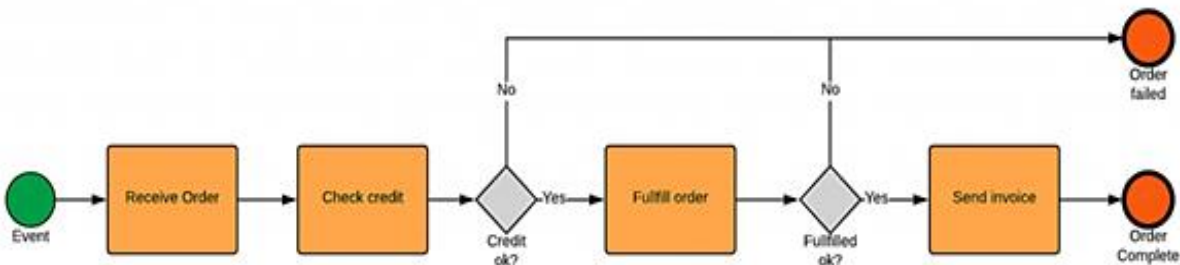


Рисунок 21 – Додавання шлюзів

Наскільки треба розрізняти кінцеві стани та кроки процесу, вирішує аналітик. У реальному сценарії веб-магазину продавець, ймовірно,

продовжить виконання замовлення, якщо його немає в наявності, що призведе до іншого процесу та можливих інших кінцевих станів. Крім того, активність перевірки кредиту в цій моделі є атомарною, ми не можемо побачити, з яких підпроцесів скомпільована ця діяльність. Якби діяльність була комплексною, можливо було б більше результатів.

Виходячи з початкового опису цього конкретного процесу, треба припустити, що цікавий процес, який призводить до завершення замовлення та невдачі. Таким чином, цієї моделі достатньо.

У моделюванні процесів реального світу дослідницьке питання або процес, який треба вдосконалити, визначають рівень деталізації, який додається до моделі. Залежно від зацікавлених сторін і обсягу процесу можна додати скільки завгодно деталей. Наприклад, розглянемо **діаграму BPMN з пулом**.

Пул слід розглядати як контейнер для всього процесу. Таким чином, у цій моделі можна просто помістити весь наскрізний процес у пул і дати пулу назву процесу (рис.22).

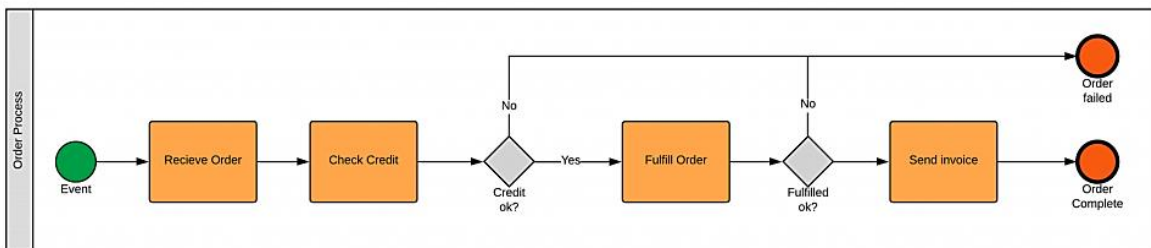


Рисунок 22 – Діаграма BPMN з пулом

Розглянемо приклад BPMN з пулами та доріжками. Тепер додамо більше деталей до діаграми процесу замовлення, додавши смуги. Доріжка – це підрозділ у пулі, який використовується для організації та категоризації дій процесу. Найчастіше смуга представляє організаційну роль, як-от розробник, продаж або менеджер. Таким чином, пул, який містить цілий процес, може бути скомпільований із кількох смуг усередині нього. Тепер додамо смуги до пулу процесів замовлення, який було створено. Додаємо до пулу три смуги: одну для продажів, одну для фінансів і одну для складу, і розміщуємо діяльність у правих смугах (рис.23).

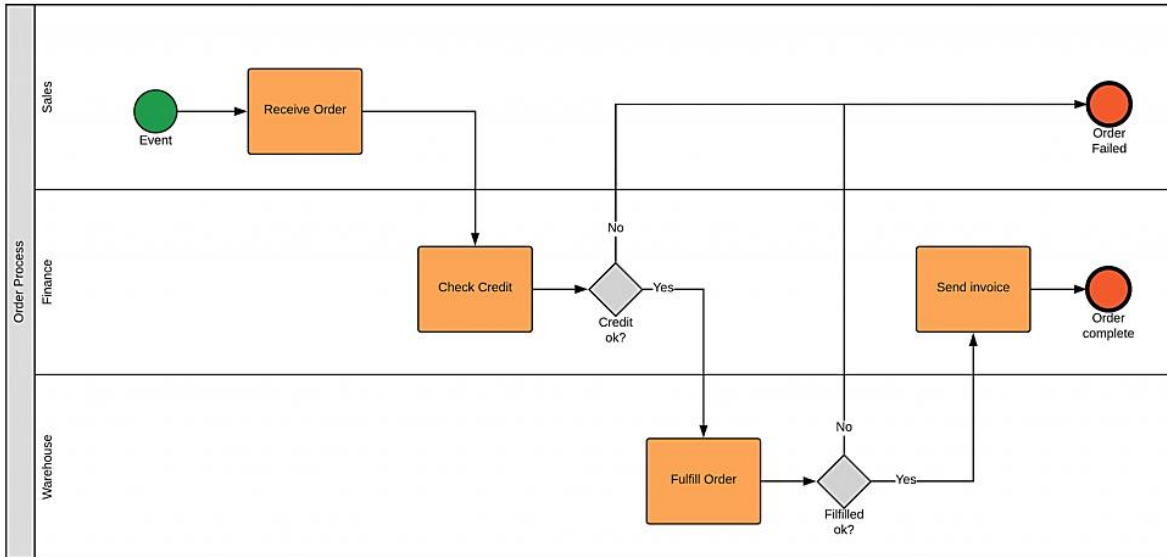


Рисунок 23 – Діаграма BPMN з пулом та доріжками

Далі створемо **діаграму BPMN з різними типами завдань**. Дії є єдиними об'єктами потоку в BPMN, які представляють роботу. Діяльність є або підпроцесом, або завданням. У палеті першого рівня BPMN завдання додатково визначаються трьома типами завдань. Різні типи завдань візуалізуються різними значками всередині форми завдання.

1. Перший тип завдання визначається як «користувач». Цей тип завдань представляє будь-яку атомарну діяльність, яку виконує людина. Завдання відрізняється символом людини.

2. Другий тип завдань визначається як «сервіс». Тип службового завдання представляє будь-яку автоматизовану атомарну діяльність. Автоматизовано означає, що користувач не бере участі в цьому етапі процесу. Це візуалізується символом шестиріччя всередині завдання.

3. Третій тип події позначається абстрактним типом завдання або без завдання, який ми використовували досі і не має жодного символу всередині.

Розширимо приклад моделі BPMN різними типами завдань. Отримати замовлення - завдання користувача. Позначаємо завдання символом людини. Перевірка кредиту – це, як правило, автоматизований етап процесу, у якому жоден користувач не задіяний. Тому позначимо цей крок як службове завдання. Виконати замовлення — це не завдання, а підпроцес. Змінимо цю дію на підпроцес, додавши символ +. Зараз надсилання рахунка-фактури, на щастя, є

автоматизованим процесом, але припустімо, що рахунок-фактура надсилається вручну. Змінити надсилання рахунку на завдання користувача. Приклад діаграми BPMN на рис.24.

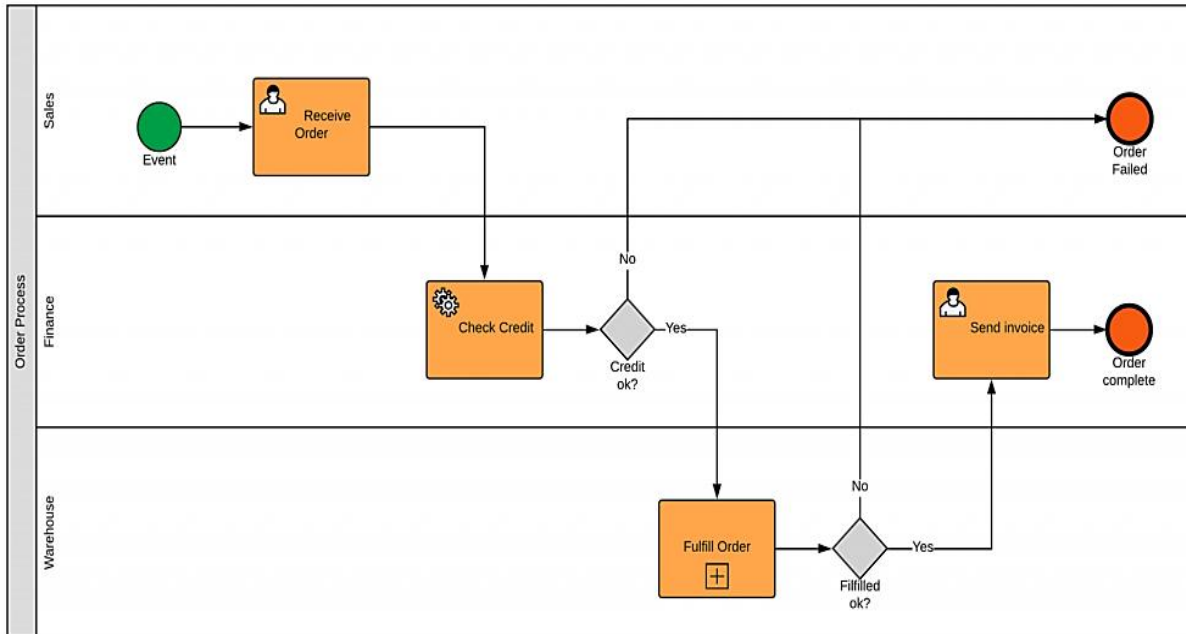


Рисунок 24 – Діаграма BPMN з різними типами завдань

Приклади:

<https://camunda.com/bpmn/>

<https://www.conceptdraw.com/examples/booking-process>

<https://www.compellingadverts.com/2022/09/business-modelling-ultimate-guide-to.html>

<https://www.process.st/bpmn-tutorial/>

1.3.5 Зміст звіту з виконання КР №2

Звіт повинен містити:

- 1) титульний аркуш
- 2) основну частину, а саме:
 - мета роботи;
 - виклад виконання роботи, мета бізнес-процесу, рисунок бізнес-процесу у нотації BPMN, короткий опис бізнес-процесу;
 - висновки (що було вивчено, які завдання вирішені і які отримані навички практичної роботи).

1.4 Оформлення, подання і захист контрольної роботи

Звіт щодо виконання КР може бути подано на розгляд викладачу у вигляді текстового файлу у форматі *.docx* і прикріплено до відповідної активності у системі Moodle у форматі *.pdf*.

Звіт виконують державною мовою, науковим стилем.

Вимоги до тексту звіту: формату А4 (210x297мм), шрифт Arial, 14 кегль, інтервал – 1,5; береги: верхній, нижній – 2 см, правий – 1,5 см; лівий – 3 см, абзац – 1,27.

Посилання у тексті на джерела зазначають порядковим номером за переліком посилань у квадратних дужках, наприклад, «... у роботі [1] ...». Оформлення списку використаних джерел здійснюється за Національним стандартом України «Інформація та документація. Бібліографічна посилання. Загальні положення та правила складання. ДСТУ 8302:2015» .

Захист КР відбувається на практичному занятті згідно з графіком контрольних точок, передбаченим робочою програмою дисципліни. Оцінка за виконання КР виставляється викладачем у відповідній активності у системі Moodle, враховуючи поточну успішність студента.

1.5 Критерії оцінювання результатів виконання контрольної роботи

Кожна модульна контрольна робота включає блок задач з матеріалу модуля (max 30 балів).

Задачі передбачають обґрунтування порядку розв'язання проблем щодо проєктування ПЗ згідно предметної області дипломної роботи, оформлення Use Cases специфікації та моделювання бізнес-процесів.

При розробці специфікації оцінюється логіка та обґрунтованість представленого матеріалу – знання предметної області, правильність формування системи вимог до майбутнього ПЗ та моделювання бізнес-процесу, в якому буде використано ПЗ, яке проєктується в межах дипломної роботи.

Критерії оцінювання:

– студент підготував роботу відповідно до поставленого завдання, в якій: правильно визначив проблеми, комплекс факторів, які могли вплинути на їх виникнення, обґрунтував своє бачення теоретичними концепціями або моделями, виконав необхідне проєктування, представив висновок або власне бачення вирішення проблеми і окреслив можливі перспективи, обмеженість такого рішення, навів за необхідності методику і результати; робота

структурована, викладена діловим, науковим або публіцистичним стилем (15 балів);

– робота містить комплексну, логічну і оригінальну пропозицію проблематики контрольної роботи аж до міждисциплінарного підходу; використання штучного інтелекту (ШІ) не забороняється, оскільки пропозиції відомих застосунків ШІ суттєво залежать від обміркованої постановки питання і уточнюючих питань; однак в разі, якщо відповідь, отримана з використанням ШІ, не є комплексною або не відповідає за стилем і викладеними позиціями іншим частинам роботи або завдання, містить очевидно неправдиву інформацію, то оцінка за цим критерієм знижується (15 балів);

Додаткові зауваження:

- студент може оскаржити отримані оцінки в порядку, передбаченому Положенням про організацію освітнього процесу (Нормативні документи : Polytechnic (metinvest.university)) та Положенням про політику та процедури врегулювання конфліктних ситуацій (Академічні політики : Polytechnic (metinvest.university));

- викладач не має права знижувати оцінку за контрольну роботу, якщо вона не була складена вчасно, однак у разі, якщо така робота була оцінена пізніше, ніж момент завершення теоретичного навчання у семестрі, то відповідна оцінка не враховується у рейтингу здобувачів освіти.

Оскарження оцінки може бути здійснене на останньому практичному занятті модуля.

СПИСОК РЕКОМЕНДОВАНИХ ДЖЕРЕЛ

1. Робоча програма навчальної дисципліни «Стандарти та методології бізнес-аналізу» / уклад. В. В. Москаленко. Запоріжжя : ТОВ «ТЕХНІЧНИЙ УНІВЕРСИТЕТ «МЕТІНВЕСТ ПОЛІТЕХНІКА», 2024. 13 с.
2. Положення про організацію освітнього процесу у ТОВ «ТЕХНІЧНИЙ УНІВЕРСИТЕТ «МЕТІНВЕСТ ПОЛІТЕХНІКА». URL: <https://metinvest.university/data/file/14/86/148623526b844f999bad7c51257baa51.pdf>.
3. Положення про визнання у ТОВ «ТЕХНІЧНИЙ УНІВЕРСИТЕТ «МЕТІНВЕСТ ПОЛІТЕХНІКА» результатів навчання, набутих у неформальній/інформальній освіті. URL: <https://metinvest.university/data/file/eb/ef/ebef5aa7e673444797bf73d46e33615b.pdf>.
4. Положення про академічну доброчесність здобувачів вищої освіти та науково-педагогічних працівників ТОВ «ТЕХНІЧНИЙ УНІВЕРСИТЕТ «МЕТІНВЕСТ ПОЛІТЕХНІКА». URL: <https://metinvest.university/data/file/c1/c9/c1c998364cec4bdbb42478109c72e17c.pdf>.
5. Use Case Specification Guideline – Best Tips & Guidance for 2025 : веб-сайт. URL: <https://businessanalystmentor.com/use-case-specification-guidelines/> (дата звернення: 01.12.2025).
6. What is Use Case Specification? : веб-сайт. URL: <https://www.visual-paradigm.com/guide/use-case/what-is-use-case-specification/> (дата звернення: 01.12.2025).
7. Use Case Diagrams: An Introduction : веб-сайт. URL: <https://www.altkomsoftware.com/blog/use-case-diagrams-an-introduction/> (дата звернення: 01.12.2025).
8. Lesson 2 - UML - Use Case Diagram Source : веб-сайт. URL: <https://www.ictdemy.com/software-design/uml/uml-use-case-diagram> (дата звернення: 01.12.2025).
9. Use Case Description : веб-сайт. URL: <https://www.visual-paradigm.com/features/use-case-description/> (дата звернення: 01.12.2025).
10. Use case specification outline : веб-сайт. URL: <https://www.ibm.com/docs/en/engineering-lifecycle-management-suite/doors-next/7.1?topic=cases-use-case-specification-outline> (дата звернення: 01.12.2025).
11. Wiegiers K., Hokanson C. Software Requirements Essentials: Core Practices for Successful Business Analysis. Addison-Wesley

Professional. 2023. 208 p. URL: <https://www.softwarereqs.com/> (дата звернення: 01.12.2025).

12. Lesson 3 - UML - Use Case Specification : веб-сайт. URL: <https://www.ictdemy.com/software-design/uml/uml-use-case-specification> (дата звернення: 01.12.2025).

13. Система для моделювання процесів у нотації BPMN : веб-сайт. URL: <https://www.terrasoft.ua/page/bpmn> (дата звернення: 01.12.2025).

14. 9 програм для моделювання бізнес-процесів : веб-сайт. URL: <https://uk.crmoz.com/blogs/post/business-process-modeling> (дата звернення: 01.12.2025).

15. Introducing BPMN 2.0 in Visio : веб-сайт. URL: <https://www.microsoft.com/en-us/microsoft-365/blog/2012/11/19/introducing-bpmn-2-0-in-visio> (дата звернення: 01.12.2025).

16. Business Process Model and Notation (BPMN) : веб-сайт. URL: https://sparxsystems.com/enterprise_architect_user_guide/16.0/modeling_languages/bpmn_1_4.html (дата звернення: 01.12.2025).

17. Типові помилки при моделюванні бізнес-процесів в нотації BPMN : веб-сайт. URL: <https://www.ba.in.ua/2022/11/25/typovi-pomylyky-pry-modelyuvanni-biznes-proczesiv-v-notacziyi-bpmn-ch-1/> (дата звернення: 01.12.2025).

Додаток А. Приклад оформлення титульного аркушу



ТОВ «ТЕХНІЧНИЙ УНІВЕРСИТЕТ «МЕТІНВЕСТ ПОЛІТЕХНІКА»
Факультет автоматизації виробництва та цифрових технологій
Кафедра цифрових технологій та проектно-аналітичних рішень

**ЗВІТ
З КОНТРОЛЬНОЇ РОБОТИ №1**

з дисципліни: «СТАНДАРТИ ТА МЕТОДОЛОГІЇ
БІЗНЕС-АНАЛІЗУ»

на тему: «Розробка Use Case специфікації для формування
вимог до ПЗ»

Роботу виконав	Студент групи 122-22-1п Микола КОВАЛЬ
Роботу перевірів	Валентина МОСКАЛЕНКО

ЗАПОРІЖЖЯ 2025



Специфікація Use Cases

для

<Програмної системи
автоматизації.....>

Затверджено: версія 1.0

Підготував: <авт>

<організація>

Дата: 5 січня 2026

Затвердження

Стейкхолдер	Назва компанії	Схвалено/Не затверджено	Підпис	Дата
Джон Нортон	ТОВ «Капітал»	схвалено		01.12.2024

Контроль версій

Версія	Ім'я та Прізвище	Назва	Контактні дані	Дата	Короткий зміст змін
[Версія]	[Ім'я та прізвище автора]	[Назва]	[Електронна адреса]	[Дата]	[Вставте опис змін, внесених до документа, та причини змін]
V1_1					

Пов'язані документи

Назва документа	Розташування документа
Специфікація бізнес-вимог	Google disk , One Drive
Звіт з лабораторної роботи №1	Moodle

Коментар: за текстом сірим надано приклади або пояснення, треба вставити текст відповідно ПЗ, що розроблюється, та зробити текст кольором «чорний» або Авто.

Зміст

1.	Назва функції 1	3
1.1	Модель варіантів використання/діаграма активності.....	3
1.2	Список випадків використання для функції FR1.....	3
1.3	Варіант використання UC-1.....	3
1.4	Варіант використання UC-2.....	6
2.	Назва функції 2.....	7
2.1	Модель варіантів використання/діаграма активності... ..	7
2.2	Список випадків використання для функції FR2.....	7
2.3	Варіант використання UC-3.....	7
2.4	Варіант використання UC-4.....	8

Назва функції 1

[Вставте назву функції, наприклад, FR1 «Створення екзаменаційної сесії»]

1.1 Діаграма варіантів використання

[Додайте діаграму варіантів використання у нотації UML або для окремих функцій, або для усієї системи (можна їх розмістити у додатку та надати тут посилання на неї)]

Надайте модель варіанту використання, що показує приклад використання та акторів, можна надати огляд кроків варіанту використання на діаграмі активності – див. ІДЗ №1. За потребою можна надати інші моделі у нотації UML для пояснення реалізації відповідної функції або варіанту використання].

1.2 Список варіантів використання для функції FR1

Головна дійова особа (Primary Actor)	Випадки використання (Use Cases)
Користувач 1 [Адміністратор]	UC-1 – назва Use Case 1 (Add Exam Sessions) UC-2 – назва Use Case 2 (Change Exam Sessions)
.....

[На діаграмі варіантів використання у відповідних Use Case повинні також стояти ідентифікатори UC-1, UC-2...]

1.3 Варіант використання UC-1

Ідентифікатор та ім'я:	Введіть коротку назву випадку використання, використовуючи активну дієслівну фразу, наприклад UC-1 Add Exam Sessions		
Автор:		Дата створення:	
Головний актор:	<i>Ім'я актора (тобто роль), який буде ініціювати цей варіант використання. Суб'єктом може бути особа або система, або зовнішня організація, наприклад, Адміністратор веб-сайту</i>		
Другорядний актор:	<i>Де це застосовано ще, назвіть інших учасників, які братимуть участь у завершенні сценарію використання, наприклад, керівник закладу зможе змінювати період сесії</i>		
Опис:	<i>Високорівневий опис послідовності дій та результату виконання варіанту використання. Наприклад,</i>		

	Адміністратор формує на сайті розклад екзаменаційної сесії
Тригер:	<i>Тригер просто визначає точну дію, яка призводить до Випадку використання. Наприклад, Адміністратор ініціює початок події щодо створення заходу</i>
Передумови:	<i>Перелічіть будь-які дії, які повинні відбутися, або будь-які умови, які повинні бути виконані, перш ніж можна буде почати використання випадку. Пронумеруйте кожен передумову. наприклад PRE-1. Актор увійшов як адміністратор веб-сайту PRE-2. Адміністратор веб-сайту переглядає графік технічного обслуговування головного розподільного щита. Якщо передумови визначені погано, розроблена дія може взагалі не працювати або матиме серйозні помилки.</i>
Постумови:	<i>Опис стану системи на момент виконання сценарію використання. Повинна включати в себе як мінімальні гарантії (що повинно статися, навіть якщо мета актора не досягнута), так і гарантії успіху (що відбувається при досягненні мети актора). Пронумеруйте кожен постумову. наприклад POST-1. Додається екзаменаційна сесія. POST-2. Повідомлення розсилається викладачам.</i>
Нормальний потік:	<i>Надайте детальний опис дій користувача та реакцій системи, які відбудуться під час виконання сценарію використання за нормальних, очікуваних умов. Ця діалогова послідовність в кінцевому підсумку призведе до досягнення мети, зазначеної в назві випадку використання. Пронумеруйте кожен крок. Наприклад: 1. Адміністратор веб-сайту вибирає «Розклад іспитів» 2. Адміністратор веб-сайту вибирає «Збір та налаштування даних» 3. Адміністратор веб-сайту вибирає «Деталі сеансу» 4. Система відображає екран «frmExam_sessions» 5. Адміністратор веб-сайту вибирає «створити сеанс», щоб створити новий сеанс 6. Адміністратор веб-сайту вводить дані сеансу. 7. Таблиця оновлень системи в базі даних 'tblXtableExam_sessions(dbo)'</i>
Альтернативні потоки:	<i>Документуйте гілки від основного потоку для обробки спеціальних умов (розширення). Для кожного альтернативного потоку посилайтеся на номер кроку розгалуження нормального потоку та</i>

	<p>умову, яка повинна бути істинною, щоб це розширення було виконано. Наприклад, альтернативні потоки в завданні «Додати екзаменаційну сесію»:</p> <p>6а. На кроці 6 нормального потоку, якщо введено дублікат сеансу</p> <ol style="list-style-type: none"> 1. Система запропонує Адміністратору веб-сайту прийняти дублікат сеансу 2. Адміністратор веб-сайту приймає 3. Сценарій використання відновлюється на кроці 7 нормального потоку <p><i>Примітка: Вставте новий рядок для кожного окремого альтернативного потоку.</i></p>
Винятки:	<p>Опишіть будь-які очікувані умови помилки, які можуть виникнути під час виконання сценарію використання, і визначте, як система повинна реагувати на ці умови.</p> <p>Наприклад, винятки з додаткової екзаменаційної сесії</p> <p>6б. На кроці 6 норм. потоку, якщо вводяться невірні дані</p> <ol style="list-style-type: none"> 1. Сеанс відхилене 2. Повідомлення адміністратору для повторного введення даних 3. Адміністратор вводить допустимі дані 4. Сценарій відновлюється на кроці 7 норм. потоку <p><i>Примітка: вставляйте новий рядок для кожного потоку винятків.</i></p>
Пріоритет:	<p>Визначити відповідний пріоритет обраній моделі пріоритетизації, наприклад, згідно MoSCoW - Must Have</p>
Бізнес-правила:	<p>Перелічіть будь-які бізнес-правила, які впливають на цей варіант використання, або надайте посилання на документ, наприклад, на Специфікацію бізнес-вимог, Специфікацію функціональних вимог тощо.</p>
Припущення:	<p>Перерахуйте будь-які припущення, які були зроблені під час аналізу, що призвело до прийняття цього випадку використання в опис продукту та написання опису випадку використання.</p>
Інша інформація:	<p>Наприклад вимоги до безпеки даних, до несанкціонованого доступу щодо змін даних сесії тощо</p>

1.4 Варіант використання UC-2

Ідентифікатор та ім'я:	UC-2		
Автор:		Дата створення:	
Головний актор:			
Другорядний актор:			
Опис:			
Тригер:			
Передумови:	PRE-1		
Постумови:	POST-1		
Нормальний потік:	<ol style="list-style-type: none"> 1. 2. 		
Альтернативні потоки:			
Винятки:			
Пріоритет:			
Бізнес-правила:			
Припущення:			
Інша інформація:			

Назва функції 2

[Вставте назву функції, наприклад, FR2 «Аналізувати результати екзаменаційної сесії»]

2.1 Модель варіантів використання/діаграма активності

[Додайте діаграму варіантів використання для функції 2 (або посилання на загальну діаграму) та відповідні діаграмі активності та інші моделі для пояснення реалізації функції].

2.2 Список випадків використання для функції FR2

Головна дійова особа (Primary Actor)	Випадки використання (Use Cases)
Користувач 2	UC-3 – назва Use Case
Користувач 3	UC-4 – назва Use Case
.....

2.3 Варіант використання UC-3

Ідентифікатор та ім'я:			
Автор:		Дата створення:	
Головний актор:			
Другорядний актор:			
Опис:			
Тригер:			
Передумови:	PRE-1		
Постумови:	POST-1		
Нормальний потік:	1. 2.		
Альтернативні потоки:			
Винятки:			
Пріоритет:			
Бізнес-правила:			
Припущення:			
Інша інформація:			

2.4 Варіант використання UC-4

Ідентифікатор та ім'я:			
Автор:		Дата створення:	
Головний актор:			
Другорядний актор::			
Опис:			
Тригер:			
Передумови:	PRE-1		
Постумови:	POST-1		
Нормальний потік:	1. 2.		
Альтернативні потоки:			
Винятки:			
Пріоритет:			
Бізнес-правила:			
Припущення:			
Інша інформація:			

Додаток В. Приклади формування Use Case специфікації

Приклад Б.1. Опишімо вимоги до системи керування вмістом ICT.social, опис якої представлено у джерелі [<https://www.ictdemy.com/software-design/uml/uml-use-case-specification>] (ICT - Information and Communication Technology).

1. Діаграма варіантів використання

Розроблено діаграму варіантів використання до системи керування вмістом ICT.social (рис.Б.1).

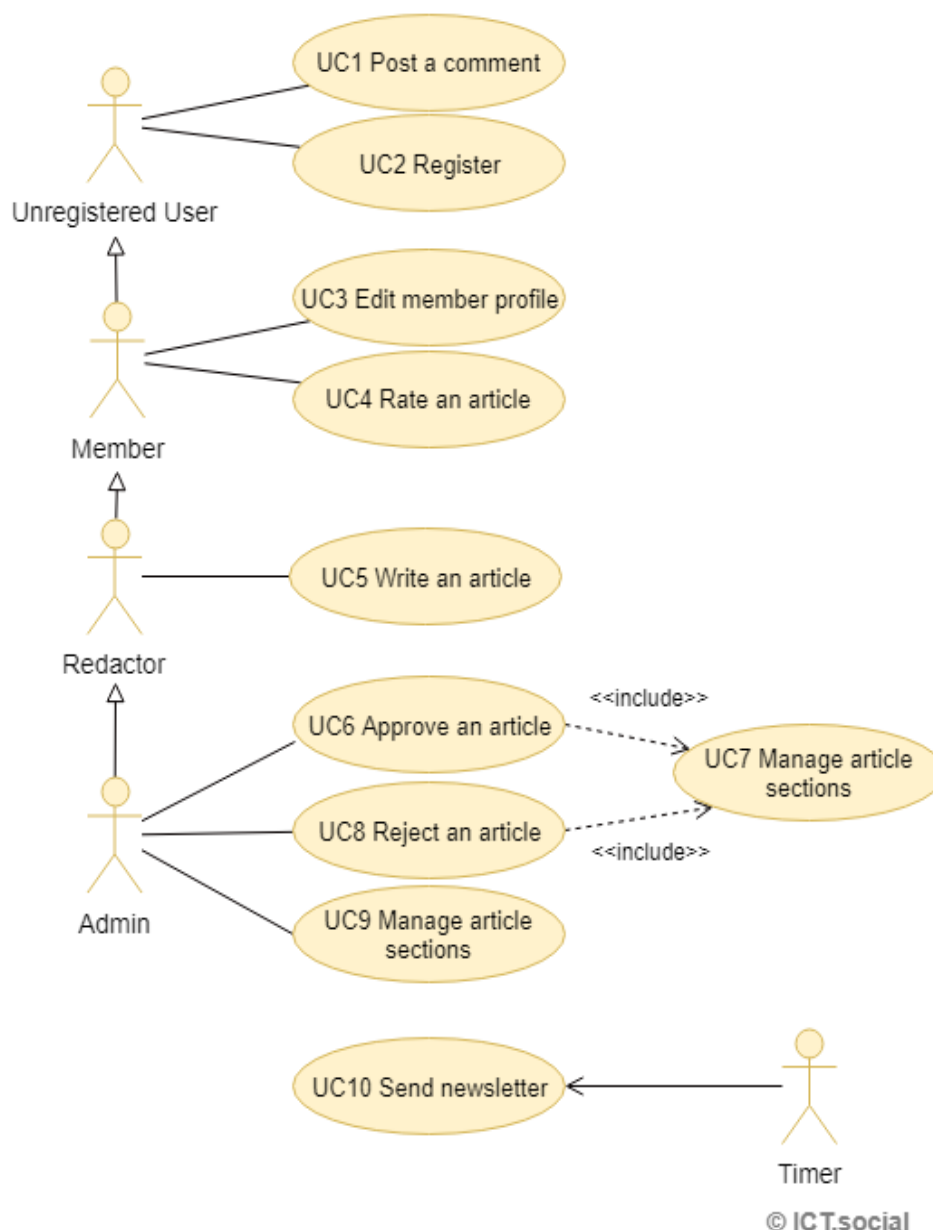


Рисунок В.1 – Діаграма варіантів використання для програми ICT.social [<https://www.ictdemy.com/software-design/uml/uml-use-case-diagram>]

2. Опис діаграми варіантів використання

Незареєстрований користувач може залишати коментарі або реєструватися.

Учасник може виконувати ті ж дії, що й незареєстрований користувач, оскільки він успадковує від нього. Крім того, учасник може керувати своїм профілем і переглядати статті.

Редактори можуть виконувати ті самі дії, що й учасники, але вони також можуть писати статті.

Адміністратори можуть виконувати кілька додаткових функцій над редакторами. Варіант використання, пов'язаний зв'язком <<include>>, завжди виконується, коли виконується варіант використання, з яким він пов'язаний — коли автор статті отримує сповіщення про схвалення та відхилення його статті. Саме сповіщення міститиме більше логіки, напр. він може надсилати електронні листи, SMS тощо. Це все було представлено одним випадком використання.

Актором є також таймер, який виконується через певні проміжки часу та надсилає електронну розсилку.

3. Специфікація Use Case

Опишемо специфікацію випадку використання “Post a Comment (Опублікувати коментар)”. У даному випадку найменування вимоги та варіанту використання однакові.

Варіант використання

Ідентифікатор та ім'я:	UC1. Post a Comment		
Актор:		Дата створення:	
Головна дійова особа:	Незареєстрований користувач (Юзер)		
Другорядні дійові особи:	Редактор, Адмін		
Опис:	Цей варіант використання дозволяє коментувати статтю системи керування вмістом		
Тригер:	Зайти до системи		
Передумови:	PRE-1: У користувача повинна бути відкрита відповідна стаття.		
Постумови:	POST-1: Новий коментар буде коректно збережено в базі даних		
Нормальний потік:	<ol style="list-style-type: none"> 1. Система генерує форму та зображення з 4 довільними поверненими літерами. 2. Користувач заповнює текст свого повідомлення, своє повне ім'я та адресу електронної пошти. Далі вони переписують текст із зображення у відповідне поле та надсилають форму. 3. Система перевіряє дані користувача. 4. Система зберігає повідомлення. 		

Альтернативні потоки:	<p>Альтернативний потік 1 Якщо користувач зареєстрований, йому не потрібно проходити тест, переписуючи текст контрольного зображення (captcha).</p> <p>Альтернативний потік 2 1. Якщо користувач ввів неправильні дані, система сповістить користувача про проблему та не дозволить надіслати повідомлення. 2. Користувач виправить неправильні введення/введення, і процес продовжиться на 2-му кроці основного потоку</p>
Винятки:	
Пріоритет*:	Must
Бізнес-правила:	Коментар повинен містити не більше 1800 знаків
Припущення:	
Інша інформація:	У якості нефункціональних вимог розглядаються вимоги до продуктивності та безпеки даних

Приклад 2. <https://www.projectmanagementdocs.com/template/project-documents/use-case-document/>

Назва випадку використання:	Замовити матеріали		
Створено:	Корпорації ABC	Востаннє оновлено:	Дж. Доу
Дата створення:	15.02.xx	Дата останнього перегляду:	22.02.xx
опис:	Покупець ABC Corp. надсилає замовлення на матеріал одному з попередньо затвердженого списку постачальників матеріалів		
Актори:	Покупець ABC Corp., матеріальний модуль SAP, попередньо затверджений постачальник		
Передумови:	<ol style="list-style-type: none"> 1. Постачальник має попереднє схвалення в системі замовлення ABC Corp. 2. Доступне фінансування для замовлення матеріалів 3. Замовлений матеріал доступний для покупки 		
постумови:	<ol style="list-style-type: none"> 1. Постачальник отримує кошти для придбання матеріалів 2. ABC Corp. отримує матеріали протягом визначеного терміну 3. Матеріальний рахунок ABC Corp. зменшується на вартість замовлення на матеріал 4. Інвентарні номери ABC Corp. успішно оновлюються після отримання матеріалу 		
Потік:	<ol style="list-style-type: none"> 1. Покупець ABC Corp. визначає матеріал, який потрібно замовити 2. Покупець ABC Corp. перевіряє попередньо затверджений список постачальників, щоб визначити постачальника 3. Покупець ABC Corp. підтверджує наявність фінансування 4. Покупець ABC Corp. надсилає замовлення попередньо затвердженому постачальнику 		

	<p>5. Постачальник отримує замовлення та перевіряє наявність матеріалу та приймає переказ коштів</p> <p>6. Постачальник отримує замовлення на матеріал і подає замовлення на доставку, щоб відправити матеріал</p> <p>7. ABC Corp. отримує матеріал</p> <p>8. ABC Corp. вводить перевірку надходження матеріалу в SAP, і рівень запасів оновлюється</p> <p>9. Підтверджуються транзакції фінансування між покупцем і постачальником ABC Corp.</p>
Альтернативні потоки:	<p>5. На кроці 5 звичайного потоку, якщо у постачальника немає доступного матеріалу</p> <p>1. Постачальник розміщує замовлення в статусі утримання та повідомляє покупця ABC Corp.</p> <p>2. Постачальник надає оновлення та приблизний термін отримання матеріалу</p> <p>3. Після надходження матеріалу сценарій використання відновлюється на кроці 6 нормального потоку</p>
Винятки:	<p>2. На кроці 2 звичайного процесу, якщо ABC Corp. визначає потрібний матеріал без попередньо затвердженого постачальника</p> <p>1. Покупець ABC Corp. ініціює внутрішній процес для визначення постачальників для нового матеріалу</p> <p>2. Покупець ABC Corp. координує угоду між ABC Corp. і потенційним постачальником</p> <p>3. Після отримання згоди та схвалення постачальника додається до списку попередньо затверджених постачальників</p> <p>4. Випадок використання відновлюється на крок 3 нормального потоку</p>
Інші вимоги:	<p>Наступні вимоги мають бути виконані перед виконанням варіанту використання</p> <p>1. Наявність фінансування має бути перевірено перед надсиланням будь-яких закупівель матеріалів</p> <p>2. Усі замовлення на матеріали мають відповідати внутрішнім інструкціям і процедурам ABC Corp.</p>

Навчально-методичне видання

Валентина Володимирівна Москаленко

СТАНДАРТИ ТА МЕТОДОЛОГІЇ БІЗНЕС-АНАЛІЗУ

методичні рекомендації
до виконання контрольних робіт

Самостійне електронне мережеве видання

Публікується в авторській редакції