

КЕРУВАННЯ ПРОГРАМНИМИ ПРОЄКТАМИ

УРОК 3

РОБОТА З ВИМОГАМИ, ІНСТРУМЕНТАЛЬНІ ЗАСОБИ, ЯКІ ВИКОРИСТОВУЮТЬСЯ В РАМКАХ ПРОЄКТУ

ЗМІСТ

1. Що таке вимога?	3
2. Класифікація вимог до програмного продукту	8
Якості, які повинні мати гарні вимоги	10
3. Що таке розробка вимог?	12
4. Ризики у розробці вимог	15
5. С-вимоги (вимоги замовника) та D-вимоги (вимоги розробника)	19
6. Підходи для вилучення С-вимог	21
7. Підходи до аналізу С-вимог	24
Підходи до розробки D-вимог	24
Приклад перетворення С-вимоги на D-вимогу	26
8. Документування вимог	35
9. Керування змінами вимог	36
Домашнє завдання	40

1. ЩО ТАКЕ ВИМОГА?

В індустрії розробки програмного забезпечення є декілька визначень терміна «вимога», однак, ми спиратимемося на таке:

***Вимога** – це умови або можливості, необхідні користувачу на вирішення проблем або досягнення цілей.*

© B IEEE Standard Glossary of Software Engineering Terminology (1990)

В інтернеті можна знайти багато ілюстрацій, що демонструють важливість процесу роботи з вимогами. Одна з діаграм показує наступну статистику (рис. 1):

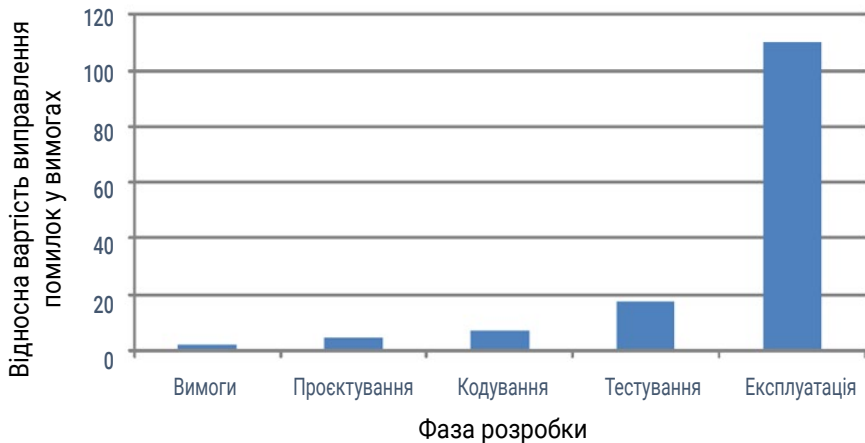


Рисунок 1

На цій діаграмі зображено залежність вартості внесення виправлення помилок у вимогах залежно від етапу проєкту. Вартість виправлення помилок на етапі експлуатації створеного програмного продукту може перевищувати

вартість всього того, що було раніше зроблено в проєкті.

Можна знайти й інші дані, що демонструють важливість процесу керування вимогами до програмного продукту:

- повнота вимог, зібраних шляхом обговорення із замовником, становить 25-50%, досягаючи у піку 70-80%;
- повнота виявлення такого типу вимог, як атрибути якості, в середньому складає 50%-60%, тобто без особливої уваги до цього типу вимог, аналітику вдається виявити близько половини усіх атрибутів якості, які є важливими для замовника проєкту.

В ІТ-проєктах спостерігається дуже цікавий ефект: практично завжди між замовником та командою виконавців проєкту виникає стіна нерозуміння. Замовник говорить бізнес-мовою, а команди – сленговою ІТ-мовою. Усунути проблему нерозуміння покликаний бізнес-аналітик.

Бізнес-аналітик – це окрема професія в ІТ-індустрії, для оволодіння якою потрібно багато зусиль і часу. У бізнес-аналітиків є звід знань, що містить інформацію щодо бізнес-аналізу – **BABOK** (*Business Analysis Body of Knowledge*).

BABOK узагальнює досвід провідних бізнес-аналітиків усього світу та описує 30 основних завдань бізнес-аналізу та 50 загальноновживаних технік, що використовуються для їх вирішення. З використанням BABOK проводиться підготовка спеціалістів до сертифікаційних іспитів Міжнародного інституту бізнес-аналізу (ІІБА).

До завдань бізнес-аналітика при реалізації проєкту може входити:

- **Збір (вилучення) вимог.** На цьому етапі аналітику необхідно зібрати максимально докладні вимоги до продукту проекту (наприклад, інформаційній системі).
- **Аналіз вимог.** Необхідно перевірити зібрані вимоги на повноту та реалізованість. Деякі вимоги можуть бути надто складними в реалізації або взагалі не реалізовані на вибраних технологіях.
- **Виявлення проблем та протиріч.** На цьому етапі аналітик повинен переконатися, що вимоги не суперечать одна одній, що кожна сформульована вимога не має протиріч.

Приклад протиріч у вимогах описаний тут: <https://www.ijedr.org/papers/IJEDR1402214.pdf>.

- **Розроблення рішень для знайдених проблем.** Для кожного сформульованого протиріччя у вимогах необхідно знайти рішення, яке буде адекватне з погляду вартості його реалізації та термінів.
- **Дослідження.** У проектах трапляються ситуації, коли для уточнення вимог аналітику потрібно провести додаткове дослідження. Наприклад, клієнт скаржиться на те, що його сайт приносить дедалі менше лідів. Аналітику потрібно провести аналіз сайту та виробити пропозиції щодо його покращення.
- **Прогнозування.** Для деяких ІТ-проектів важливо, щоб аналітик міг підказати клієнту які існують тренди у його сфері діяльності та прогнози щодо розвитку цього бізнесу. Наприклад, аналітик може розповісти клієнту про нові бізнес-моделі його конкурентів.

Докладніше про бізнес-моделі можна прочитати у книзі **«Побудова бізнес-моделей. Настільна книга стратега і новатора»**.

Для вирішення усіх цих завдань потрібні такі якості, якими повинен володіти професійний бізнес-аналітик:

- аналітичне мислення;
- системне мислення;
- комунікабельність;
- уважність та організованість;
- креативність;
- навчаність.

За статистикою **The Standish Group** (світового лідера у дослідженнях з галузі розробки програмного забезпечення) одна з ключових причин провалу ІТ-проектів – це неповні вимоги до продукту:

Причина провалу ІТ-проекту	% відповідей
Неповні вимоги	13,1
Відсутність залучення користувачів	12,8
Нестача ресурсів	10,6
Нереалістичні очікування	9,9
Відсутність підтримки з боку керівництва	9,3
Зміни вимог та специфікацій	8,7
Недостатнє планування	8,1
Низький попит результатів проекту протягом тривалого часу	7,5

Слабке керування проєктом	6,2
Неправильний вибір технологій	4,3
Інше	9,9

© The Standish Group. Chaos report, 2015

Оскільки опрацювання вимог стає одним із ключових факторів успіху ІТ-проєкту, розглянемо в наступних розділах процес роботи з вимогами та класифікацію вимог.

2. КЛАСИФІКАЦІЯ ВИМОГ ДО ПРОГРАМНОГО ПРОДУКТУ

Для того щоб не упустити важливих вимог до програмного продукту, бізнес-аналітику слід використовувати класифікації вимог до програмного забезпечення. Один із способів класифікації вимог був запропонований К. Вігерсом (рис. 2):



© Карл І. Вігерс «Розробка вимог до програмного забезпечення»

Рисунок 2

Бізнес-вимоги (*Business requirements*) відповідають на питання: для чого потрібен програмний продукт?

Вимоги користувачів (*User requirements*) описують цілі та завдання, які користувачам дозволить вирішувати програмний продукт.

Функціональні вимоги (*Functional requirements*) описують, що має робити інформаційна система, щоб користувачі змогли виконувати свої завдання і досягти своїх цілей.

Системні вимоги (*System requirements*) описують високорівневі характеристики інформаційної системи, зокрема:

- вимоги до роботи під певними операційними системами;
- вимоги до апаратного забезпечення;
- вимоги до ліній підтримки після застосування програмного продукту.

Бізнес-правила (*Business rules*) описують корпоративну політику, стандарти, процедури тощо.

Атрибути якості (*Quality attributes*) описують інформаційну систему з погляду характеристик, важливих для користувачів.

Приклади атрибутів якості:

- легкість і простота використання (*usability*);
- масштабування;
- інтегрованість з іншими інформаційними системами компанії.

Обмеження – умови, що обмежують вибір можливих рішень щодо реалізації окремих вимог або їх наборів.

Зовнішні інтерфейси – опис аспектів взаємодії з іншими системами та операційним середовищем. До них належать вимоги до API-продукту або системи, а також вимоги до API інших систем, з якими здійснюється інтеграція.

Знаючи про існуючі класи вимог до програмного продукту, бізнес-аналітик навряд чи упустить якийсь клас вимог. Це означає, що він хоча б поцікавиться у керівника IT-проєкту, чи потрібно здійснювати збір вимог даного класу і, при позитивній відповіді на питання, запланує збір та аналіз вимог для даного класу.

Помічено, що бізнес-аналітики найчастіше фокусуються на зборі та аналізі бізнес-вимог, користувацьких та функціональних вимог, забуваючи при цьому інші класи вимог.

Якості, які повинні мати гарні вимоги

Як можна зрозуміти, наскільки хороші вимоги до програмного продукту? Для цього потрібно розуміти, якими якостями мають бути наділені вимоги. В літературі зустрічаються такі якості гарних вимог:

- повні (тобто достатні);
- правильні (з погляду авторів);
- здійснювані (в даних умовах);
- необхідні (для чогось і комусь);

- пріоритезовані (за важливістю);
- однозначні (не двозначні);
- які можна перевірити (спробуйте самі).

З одного боку, якості гарних вимог очевидні, проте на практиці розробити такі вимоги виявляється дуже непросто.

3. ЩО ТАКЕ РОЗРОБКА ВИМОГ?

Процес роботи з вимогами складається з трьох активностей (рис. 3):

1. Планування роботи з вимогами.
2. Розробка вимог.
3. Керування змінами вимог.

У свою чергу, **розробка вимог** складається з:

- збору (виявлення) вимог;
- аналізу;
- документування вимог;
- перевірки.



Рисунок 3

Планування вимог до програмного продукту полягає в тому, щоб:

- продумати, які інструменти (підходи) для отримання вимог використати;
- узгодити використання даних інструментів з керівником проєкту;
- визначити, в яких саме груп користувачів для даного продукту потрібно виокремлювати вимоги та узгоджувати це з керівником проєкту.

Дії з розробки вимог можна описати так (рис. 4):

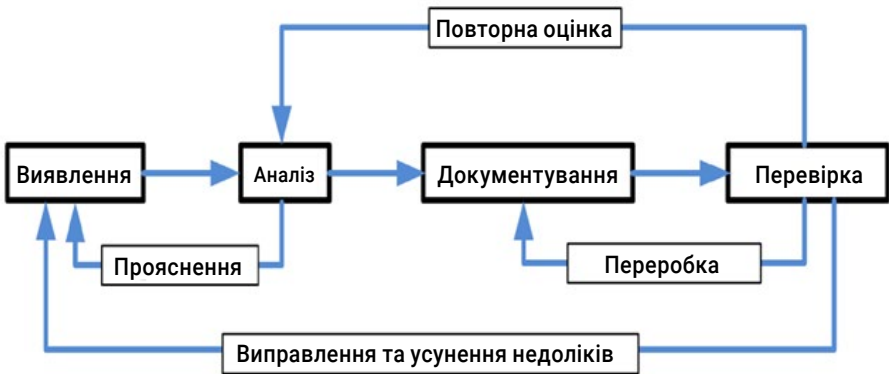


Рисунок 4

Аналітик починає свою роботу з виявлення вимог.

На етапі виявлення бізнес-аналітик має добути вимоги у зацікавлених сторін проєкту (*про підходи до вилучення вимог йдеться нижче*).

Кожну вимогу необхідно проаналізувати на предмет її реалізованості, оцінки обсягів робіт на реалізацію, вартості реалізації і обговорити із замовником проєкту, чи варто реалізовувати дану вимогу з урахуванням її вартості. В ході аналізу може знадобитися прояснення деяких моментів у вимогах.

Після цього вимогу та підхід до її реалізації слід помістити до документа про вимоги.

На фінальному етапі бізнес-аналітик повинен переконатися, що вимога була реалізована розробниками правильно. Якщо реалізація відрізняється від опису в документі, можлива переробка документа за вимогами. Також за підсумками реалізації, замовник проекту може внести якісь уточнення до вимог, а це призведе до необхідності повернутися на етап аналізу або навіть на етап виявлення вимог.

До дій з керування вимогами можуть входити:

- підтримка версії документа щодо вимог в актуальному стані;
- аналіз запропонованих змін у вимогах та оцінку впливу кожної зміни на обсяг робіт проекту;
- розробку завдань з реалізації схвалених змін вимог;
- узгодження завдань із плану-проекту з вимогами;
- відстеження статусів вимог.

4. РИЗИКИ У РОЗРОБЦІ ВИМОГ

1. Недостатнє залучення зацікавлених сторін (у т. ч. користувачів).

Розробники продукту не завжди розуміють важливість взаємодії з користувачами продукту щодо вимог, а користувачі часто не уявляють, чого саме вони очікують від програмного продукту. Крім формулювання: «хочу, щоб було зручно та швидко», вони рідко можуть самостійно сформулювати вимоги. Недостатнє залучення зацікавлених сторін у проекті може призвести до пропуску важливих вимог, що призведе до необхідності витратити незапланований час на їх обговорення та реалізацію. Усі ці фактори призведуть до різкого зниження ймовірності проекту, щоб вкластися у запланований термін та бюджет.

2. Розростання вимог користувачів.

Користувачі продукту та замовник проекту мають схильність до того, щоб генерувати ідеї про нові вимоги в ході реалізації проекту. Частина цих вимог має цінність і їх цілком можна було б реалізувати, проте якщо проект має фіксовану вартість, то додавання нових вимог знижуватиме прибутковість цього проекту для команди виконавців. Замовник проекту та команда виконавців повинні на старті проекту домовитися про те, як вони керуватимуть змінами вимог (*про це див. нижче*). Інакше команда виконавців перешкоджатиме проникненню нових вимог до проекту.

3. Двозначність вимог.

Один із симптомів двозначності вимог полягає в тому, що користувач може інтерпретувати одну й ту ж вимогу двома або більше способами.

Інший симптом – у людей, які вивчають вимогу, виникає різне

уявлення про те, як це працюватиме в продукті. Часто після того, як вимоги реалізовані, вони кажуть: «Це працює не так, як записано» або «не так, як ми очікували».

Для зниження цього ризику керівнику проекту та бізнес-аналітику варто запропонувати кільком користувачам висловити свою думку стосовно розуміння вимоги. Другий спосіб зниження ризику полягає в тому, щоб розробити варіант тестування для вимоги та розробити прототип.

4. «Золочення» продукту.

Суть цього ризику полягає в тому, що розробники додають у продукт функції, яких немає в специфікації вимог, при цьому вони вірять в те, що нові функції сподобаються користувачам. Однак, найчастіше самі користувачі вимагають реалізувати функції або елементи інтерфейсу, які, на їх погляд, будуть затребувані, але насправді виявиться, що вони не мають особливої цінності для продукту.

Щоб знизити ефект «золочення», варто простежувати для кожної вимоги користувачів, до якої бізнес-вимоги вона входить.

5. Мінімальна специфікація.

Замовники проекту, не розуміючи важливості розробки детальних вимог, часто не готові платити за розробку

документа за вимогами. Це нашкодить команді виконавців створити спрощений варіант специфікації вимог або обмежитися списком бізнес-вимог. Такий варіант специфікації підвищує зростання ймовірності того, що обсяг робіт, терміни та бюджет проєкту будуть розраховані неправильно.

6. Пропуск груп користувачів.

Якщо бізнес-аналітик втратив з поля зору групу користувачів для вашого продукту, тоді деякі вимоги не потраплять у план робіт з проєкту, що призведе до помилок в оцінці термінів і бюджету проєкту. Для зниження цього ризику потрібно провести ідентифікацію всіх класів користувачів і переконатися, що представники цих класів будуть почуті бізнес-аналітиком.

7. Недбале планування.

Все починається з запитання: «Я хочу реалізувати ось таку штуку для нашого продукту. Коли ви зможете це зробити?». Необдумані відповіді на це питання може призвести до виникнення зобов'язання, яке потопить ваш проєкт. Відповідь на це запитання краще дати так: «На 80% я впевнений, що це можна реалізувати. Однак, мені потрібно порадитися з командою та оцінити вартість реалізації цієї штуки та планові терміни».

8. Повільна швидкість обговорень вимог з користувачами.

Часто вони зайняті оперативною роботою, не знаходять часу на участь в інтерв'ю, не хочуть читати документи про вимоги і т. д. Потрібно мінімізувати цей ризик закладан-

ням резервів часу в розклад проєкту на завдання за погодженням вимог, і в контракті із замовником прописувати санкції за перевищення термінів за погодженням вимог.

9. Повільний зворотний зв'язок від користувачів щодо реалізації вимог.

Після того як команда реалізує частину вимог, вона хоче отримати зворотний зв'язок від користувачів, але вони не хочуть брати участь у мітингах щодо огляду реалізованих вимог або переносять ці мітинги.

5. С-ВИМОГИ (ВИМОГИ ЗАМОВНИКА) ТА D-ВИМОГИ (ВИМОГИ РОЗРОБНИКА)

Існує ще один підхід до класифікації вимог (описаний у книзі «Технологія розробки програмного забезпечення» Еріка Дж. Брауде, 2004), який передбачає розподіл вимог на 2 групи:

1. **Вимоги замовника (С-вимоги)** – пишуться мовою, зрозумілою замовнику проекту та відображають його очікування та потреби.
2. **Вимоги розробника (D-вимоги)** – пишуться мовою, зрозумілою команді розробки та документуються у структурованій для команди формі.

Якщо використовувати вище наведену класифікацію вимог К. Вігерса, то можна умовно розділити С-вимоги та D-вимоги таким чином:

Класифікація на С та D-вимоги	Класифікація вимог за Вігерсом
С-вимоги	Бізнес-вимоги Користувацькі вимоги Бізнес-правила Атрибути якості
D-вимоги	Функціональні вимоги Системні вимоги Зовнішні інтерфейси Обмеження

Обидва класи вимог потрапляють у SRS-документ, структуру якого ми наводили раніше.

У разі використання шаблону SRS з IEEE, у розділи «Вступ» та «Загальний опис» слід включити C- вимоги, а в розділ «Конкретні вимоги» потрапляють D-вимоги.

6. ПІДХОДИ ДЛЯ ВИЛУЧЕННЯ С-ВИМОГ

Найпоширеніші підходи для вилучення вимог представлені на рисунку 5.



Рисунок 5

Підходи до вилучення вимог розташовані на шкалі складності: зліва розташовані прості підходи, при просуванні шкалою зліва направо складність підходів зростає. Відмітимо, що таке ранжування – це суб'єктивна думка кількох бізнес-аналітиків.

- **Спостереження** – передбачає отримання інформації про те, як користувачі виконують свої завдання.

Способи спостереження можуть бути різними: спостереження через отримання статистики про частоту використання тих чи функцій у програмному продукті, або спостереження з присутністю бізнес-аналітика.

Наприклад, бізнес-аналітик може, знаходячись в офісі замовника проекту, спостерігати за тим, як виконується робота співробітників. За підсумками спостережень, він формулює вимоги про те, як цю роботу можна автоматизувати у програмному продукті.

- **Анкети** – використання заздалегідь розроблених питань та отримання письмової відповіді від зацікавлених сторін проекту на ці питання.

Цей підхід використовується в тому випадку, коли учаснику потрібно лише вибрати з кількох варіантів одну або кілька відповідей, або коли вартість інтерв'ю виявляється дуже дорогою.

- **Аналіз документів** – передбачає вивчення стандартів підприємства, процедур та регламентів, що стосуються правил та послідовності виконання бізнес-процесів, які команда планує автоматизувати.
- **Інтерв'ю** – найпоширеніший підхід для збору С-вимог, що передбачає особисту зустріч бізнес-аналітика із зацікавленою проектом особою.

Аналітик заздалегідь готує питання для інтерв'ю, призначає місце та час зустрічі, а під час інтерв'ю ставить питання, що цікавлять, стежить за невербальними сигналами зацікавленої особи, ставить уточнюючі питання і записує відповіді. Після закінчення інтерв'ю аналітик дає можливість опоненту ознайомитися з письмовим результатом інтерв'ю і внести свої коригування.

- **Фокус-групи** – цей інструмент прийшов з маркетингу. Він дозволяє зібрати кілька представників користувачів та інших зацікавлених осіб проекту та провести мітинг, в результаті якого можна отримати перелік вимог до продукту.
- **Мозковий штурм** – цей підхід давно відомий і використовується для того, щоб зібрати широкий спектр ідей щодо вимог.

- **Інноваційні ігри (бізнес-ігри)** – підхід розроблений Л. Хоманном та популярний у США. Ідея полягає в тому, щоб залучити зацікавлені сторони проекту до процедури збору вимог і перетворити цю дію на веселу, але при цьому корисну гру.
- **Прототипування** – найпоширеніший в **Agile** спосіб збору вимог. Команда розробки створює прототип декількох функцій програмного продукту і демонструє його майбутнім користувачам. Прототип дозволяє отримати зворотний зв'язок та уточнити вимоги.
- **Моделювання процесів** – передбачає використання графічних моделей діяльності користувачів.

Для збору інформації перед початком моделювання часто використовуються інтерв'ю, анкети та фокус-групи. Існує величезна кількість нотацій моделювання процесів, але найбільшу популярність набули такі: **IDEF0**, **eEPC**, **BPMN 2.0**, **UML** (особливо варіанти використання – **use case**), **DFD** (діаграми потоків даних).

- **SQFD (Software quality function deployment)** – підхід, що зародився в Японії як QFD, який почали використовувати в розробці програмного забезпечення.

У QFD використовується складний аналіз матриць причинно-наслідкових зв'язків між побажаннями споживача та технічними характеристиками продукту.

SQFD можна використовувати й для аналізу вимог.

7. ПІДХОДИ ДО АНАЛІЗУ С-ВИМОГ

Після того як С-вимоги зібрано, бізнес-аналітику слід їх проаналізувати на предмет повноти, відсутності протиріч та можливості реалізації. Під час аналізу вимог, бізнес-аналітик може використовувати такі інструменти:

- реверсивний аналіз вимог;
- аналіз систем-аналогів;
- TPB3;
- Root Conflict Analysis Plus (RCA+);
- Value-Conflict Mapping +;
- швидке прототипування.

Підходи до розробки D-вимог

Для реалізації С-вимог, команда розробки повинна сформулювати детальні D-вимоги.

Мепінг розділів SRS з IEEE 830 за D-вимогами представлений далі, на рисунку 6.

Для розробки D-вимог до продукту, розробники можуть використовувати наступні моделі:

- діаграми варіантів використання (**use-case diagram**) UML;
- діаграми потоків даних (**DFD**);
- діаграми станів (**statechart diagram**) UML.

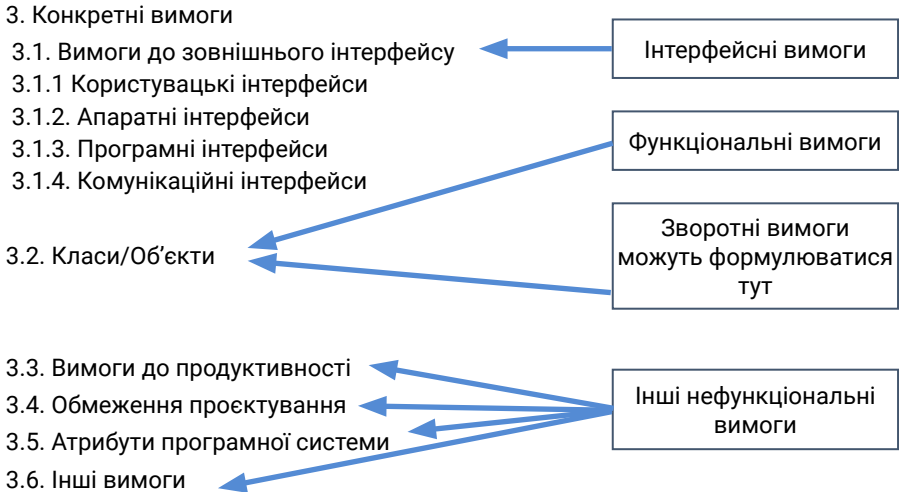


Рисунок 6

Правила використання діаграм включають наступні кроки:

1. Якщо вимога проста і впливає на інші вимоги, то діаграми не використовують, а вимогу формують словісно у відповідному розділі SRS.
2. Якщо вимога являє собою взаємодію між користувачем і додатком, то його висловлюють з допомогою діаграми варіанта використання (use-case diagram).
3. Якщо вимога стосується елементів обробки, кожен з яких отримує і видає дані, використовують діаграму потоків даних (DFD).
4. Якщо вимога зачіпає стани, в яких може перебувати програма, створюють діаграму станів (statechart diagram).

Приклад перетворення С-вимоги на D-вимогу

Бізнес-аналітик сформулював **С-вимогу** так:

Для реєстрації потреб замовника у перевезенні вантажу, експедитор повинен створити у системі документ «Заявка».

Перелік реквізитів документа «Заявка» представлений нижче.

Таблиця 1. Реквізити документа «Заявка»

№	Назва реквізиту	Короткий опис
1	Номер документа	
2	Дата створення документа	
3	Замовник (клієнт)	
4	Експедитор	
5	Відправник вантажу	
6	Отримувач вантажу	
7	Сповіднення	Текст: «Повідомити сторону про прибуття вантажу»
8	Доручення	Текст: «Замовник доручає, а Експедитор бере на себе зобов'язання з доставки вантажу»
9	Назва вантажу	
10	Країна походження вантажу	
11	Товарний код	
12	Маркування	
13	Маса (вага) вантажу (брутто, нетто)	
	Об'єм	

№	Назва реквізиту	Короткий опис
	Вартість	
14	Кількість вантажних місць	
15	Вид (розмір) упаковки	
16	Пункт відправки	Назва пункту відправлення або передачі вантажу експедитору
17	Пункт призначення	Назва пункту призначення вантажу
18	Крайній термін доставки	Дата доставки вантажу одержувачу
19	Спосіб перевезення	Методи та способи доставки вантажу: вид(и) транспорту, повагонні відправки, суднові партії, збірні відправки, перевезення у контейнерах тощо
20	Інструкція з перевезення вантажу	Властивості вантажу, особливі умови його перевезення, інша інформація про вантаж
21	Необхідні документи	
22	Страхування вантажу	Відомості про страхування вантажу: застрахований, доручається застрахувати тощо
23.1	Обов'язки сторін	Текст: «Обов'язки сторін, крім перелічених у договорі»
23.2	Обов'язки Експедитора	
24	Обов'язки Замовника	
24.1	Відповідальність сторін	Текст: «Відповідальність сторін, крім перелічених у договорі»
24.2	Відповідальність Експедитора	
25	Відповідальність Замовника	

№	Назва реквізиту	Короткий опис
25.1	Порядок розрахунків	
25.2	Вартість послуг Експедитора	
26	Порядок та строки оплати послуг	
27	Термін дії доручення	
28	Інші умови (особливі позначки)	
29	Назва Експедитора	
30	Підпис та печатка Експедитора	
31	Назва Замовника	
32	Підпис та печатка Замовника	

На наступній діаграмі подана модель стану заявки в системі (рис. 7):

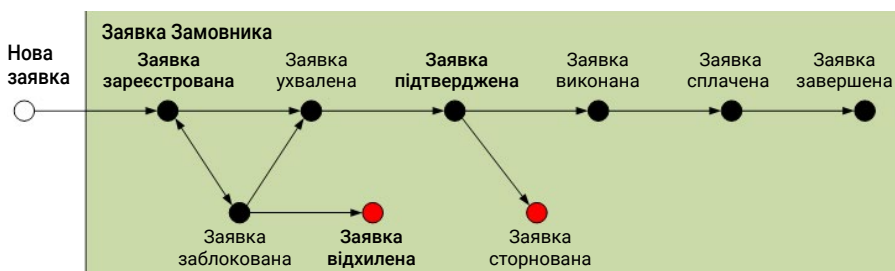


Рисунок 7

Заявка потрапляє до системи в процесі реєстрації.

У наступній таблиці наведено опис станів заявки Замовника в системі.

Таблиця 2. Стан заявки Замовника в системі

№	Стан	Опис стану
1	Заявка зареєстрована	Дані заявки Замовника введені в систему. Надано номер заявки
2	Заявка схвалена	Заявка пройшла перевірку і може виконуватися. Схвалення заявки не означає, що заявку буде виконано. Див. також п. 5
3	Заявка заблокована	Заявка не пройшла перевірку умов виконання заявки
4	Заявка відхилена	Заявка може бути відхилена з однієї з наступних причин: <ol style="list-style-type: none"> 1. Ліміт дебіторської заборгованості Замовника перевищено, і умови виконання заявки не можуть бути змінені відповідним чином. 2. Ліміт дебіторської заборгованості з урахуванням виконання заявки буде перевищено, і умови виконання заявки не можуть бути змінені відповідним чином.
5	Заявка підтверджена	Замовнику надіслано підтвердження заявки у вигляді документа «Доручення експедитору»
6	Заявка сторнована	Зміни, які вносяться до заявки з цього моменту, фіксуються, зберігаються та додатково узгоджуються із Замовником
7	Заявка виконана	Сторнування означає скасування виконання заявки з ініціативи компанії або Замовника та, у разі виникнення фінансових зобов'язань (наприклад, штрафів), їх задоволенню
8	Заявка сплачена	Усі замовлення, пов'язані із заявкою, виконані
9	Заявка завершена	Усі взаєморозрахунки між Замовником та компанією за заявкою завершено

Для того щоб перетворити **С-вимогу** на **Д-вимогу**, бізнес-аналітики прийняли рішення створити **Use-case** роботи із заявками у системі (рис. 8):

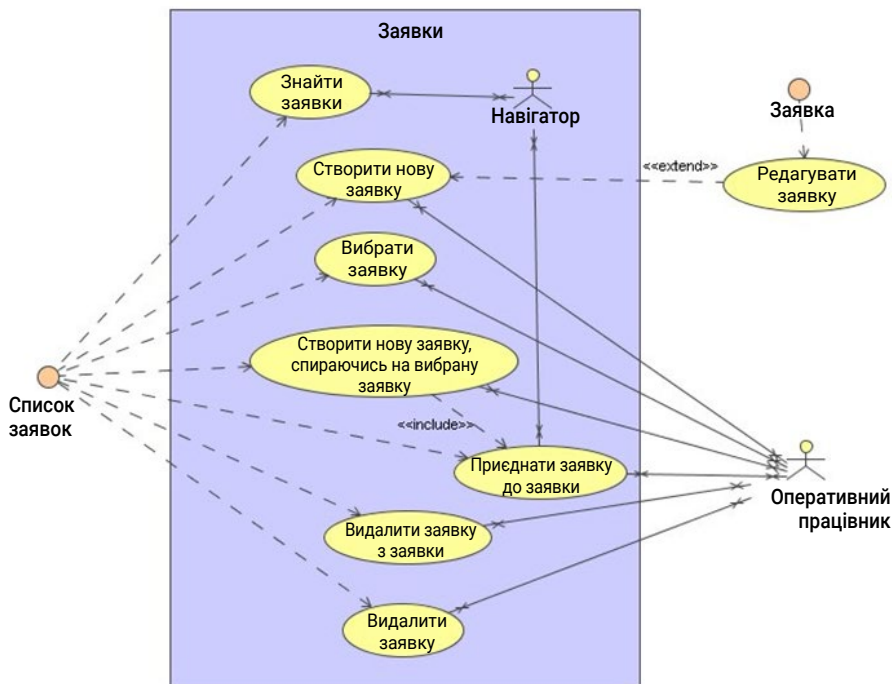


Рисунок 8

Для створення об'єкта **Заявка** в системі та опису його взаємодії з іншими об'єктами системи, бізнес-аналітики створили опис структури даних заявки.

Таблиця 3. Структура даних об'єкту Заявка

Загальні параметри

№	Реквізит документа	Опис
1	Номер документа	Номер документа в системі

№	Реквізит документа	Опис
2	Дата	Дата та час реєстрації заявки в системі (надається автоматично)
3	Реф. №	№ у заявці замовника
4	Договір	
5	Контактна особа	
6	Валюта	

Вантаж

Представлення опису вантажу. Викликається підпорядкований довідник **Вантажі**.

Транспорт

№	Реквізит	Короткий опис
1	Вид транспорту	
2	Вид ТЗ	Викликається довідник Вид ТЗ
3	Обладнання	
4	Коментар	

Сервіси

№	Реквізит	Короткий опис
1	Сервіс	
2	Входить у ставку	
3	Сума	
4	Сума, EUR	

Контрольні точки

Викликається довідник **Контрольні точки**.

Параметри заявки

№	Реквізит	Короткий опис
1	Організація	
2	Підрозділ	
3	Офіс	
4	Куратор основний	
5	Куратор напряму	
6	Група заявок	
7	Угода	
8	Напрямок	
9	Програма угод	
10	Вид перевезення	
11	Дата навантаження (план)	

Умови заявки

Заявка виконується виходячи з умов, визначених у відповідному договорі. Експедитор може редагувати умови договору у заданих межах, а також додавати інші умови.

Умови заявки представлені списком, в якому окремий елемент списку є окремою умовою. Структура даних, що описує умову заявки, представлена у наступній таблиці.

Структура даних умови заявки

№	Реквізит	Короткий опис
1	Умова оплати	
2	Умова штрафу	
3	Додаткові умови	
4	Коментар	

Умова оплати

Для заявки може бути визначена лише одна умова оплати. Структура даних, що описує умову оплати заявки, представлена у наступній таблиці.

Структура даних параметрів умови оплати в заявці

№	Реквізит	Короткий опис
1	Текст умови	В заявці не редагується. Змінюється автоматично під час редагування окремих параметрів умови оплати в заявці.
2	Вид оплати	Визначається договором. В заявці не редагується. Можливі значення: за документом, за період.
Розділ «Попередня оплата»		
3	Частка попередньої оплати, %	Визначається договором. В заявці редагується в межах дозволеного інтервалу.
4	Вид рахунку на оплату	Визначається договором. В заявці не редагується. Можливі значення: оригінал, копія (факс), копія (електронна пошта).

№	Реквізит	Короткий опис
5	КТ періоду оплати	Визначається договором. В заявці не редагується.
6	Напрямок періоду оплати (до, після)	Визначається договором. В заявці не редагується. Можливі значення: до, після.
7	Строк періоду оплати	Визначається договором. Редагується в рамках дозволеного інтервалу.
Розділ «Оплата»		
8	Вид рахунку на оплату	Визначається договором. В заявці не редагується. Можливі значення: оригінал, копія (факс), копія (електронна пошта).
9	КТ періоду оплати	Визначається договором. В заявці не редагується.
10	Напрямок періоду оплати (до, після)	Визначається договором. В заявці не редагується. Можливі значення: до, після.
11	Строк періоду оплати	Визначається договором. В заявці редагується в межах дозволеного інтервалу.

Дану **D-вимогу** бізнес-аналітики включили до розділу SRS «Конкретні вимоги».

8. ДОКУМЕНТУВАННЯ ВИМОГ

Для документування вимог зазвичай використовуються шаблони документа SRS, структура яких була наведена в Уроці 2.

Для автоматизації документування вимог та спільної роботи з документами за вимогами можна використовувати спеціальні програмні продукти, такі як: **Confluence**, **Zoho Wiki** або безкоштовний **Google Docs**.

9. КЕРУВАННЯ ЗМІНАМИ ВИМОГ

Розглянемо, як здійснюється керування змінами вимог при використанні **водоспадної моделі ЖЦ проекту** (*waterfall*) та при роботі за ітераційною моделлю, яка використовується в Agile.

У випадку, коли **Замовник проекту** розуміє, що з початкової концепції продукту не обов'язково всі вимоги до продукту мають потрапити до його першої версії, а терміни та бюджет проекту при цьому фіксуються, команда проекту повинна мати перелік пріоритезованих вимог до продукту проекту.

Пріоритети для вимог потрібні щоб, у разі, якщо затверджені у договорі терміни підходять до завершення, а бюджету не вистачає на реалізацію всіх вимог, **Замовник проекту** може швидко прийняти рішення, від яких вимог до товару можна відмовитися.

При цьому, замовник проекту має бути впевнений у тому, що при відмові від частини вимог, вийде продукт, здатний вирішувати виявлені проблеми майбутніх споживачів.

При такому підході керування змінами у вимогах в ході реалізації проекту зводиться до того, щоб постійно ранжувати решту вимог.

Такий підхід використовується у фреймворку **Scrum**. Розглянемо варіант проекту, в якому вимоги є фіксо-

ваними, а термін та бюджет проєкту можуть змінюватися в ході проєкту.

У цьому випадку опрацювання вимог до продуктів проєкту відбувається, як правило, до старту проєкту реалізації продукту і передбачається, що вимоги до продуктів не змінюватимуться. Однак, в ході реалізації вимог та появи прототипу продукту, початкові вимоги можуть здатися замовнику проєкту не актуальними, або у нього можуть з'явитися нові вимоги.

У цьому випадку керівник проєкту має бути готовим до переговорів із замовником щодо необхідності внесення змін в SRS, зміни вартості та термінів проєкту. У разі погодження змін до вимог або додавання вимог, йому слід ініціювати підписання додаткової угоди до договору.

Для роботи із змінами вводиться спеціальний документ «**Запит на зміни**», і алгоритм прийняття рішення виглядає приблизно так, як на рисунку 9 на сторінці 38.

Для того щоб ухвалити рішення про внесення змін, керівник проєкту, бізнес-аналітик та замовник проєкту повинні виконати багато дій (на кроках 2 та 3 – залучається бізнес-аналітик та замовник проєкту, на кроці 4 – потрібна участь замовника проєкту). В результаті, на обробку кожного запиту на зміну витрачається багато часу та енергії. Але якщо цього не робити, то обсяг робіт проєкту збільшиться без коригування строків та бюджету, що призведе до зриву спочатку затверджених термінів та бюджету проєкту.



Рисунок 9

Підіб'ємо підсумки уроку.

- Хороша робота з вимогами до продукту проекту різко підвищує ймовірність успіху проекту.
- Пропуск вимог, виявлений на пізніх етапах ІТ-проекту, швидше за все, призведе до серйозних витрат на його реалізацію.
- Основна особа, яка відповідає за роботу до вимог в ІТ-проекті – бізнес-аналітик.

- Усі вимоги до програмного продукту можна поділити на дві групи: С-вимоги та D-вимоги; додатково до цієї класифікації варто використовувати класифікацію К. Вігерса.
- Бізнес-аналітики повинні володіти великою кількістю методів та інструментів для отримання вимог.
- Вимоги мають бути записані в документ, наприклад, SRS, який обов'язково має затвердити замовник ІТ-проєкту.
- В ході реалізації вимог та появи першого прототипу майбутнього продукту у замовника може з'явитися бажання внести зміни до списку вимог. Є два варіанти керування змінами вимог, – до старту проєкту потрібно узгодити із замовником один із них.

ДОМАШНЄ ЗАВДАННЯ

1. Знайдіть у вашому оточенні особу, яка не задоволена вебсайтом своєї компанії. Проведіть інтерв'ю з цією особою та зафіксуйте проблему поточної версії вебсайту.
2. Проаналізуйте ці проблеми та опишіть бізнес-вимоги до покращення вебсайту.
3. Знайдіть кількох користувачів цього сайту та проведіть інтерв'ю з ними, за підсумками якого оформіть 10-15 вимог користувачів.
4. Знайдіть у вашому оточенні програміста з досвідом розробки вебсайтів і надішліть йому ваш список вимог користувача на оцінку трудомісткості.
5. За підсумками оцінки трудомісткості, у програміста, можливо, з'являться питання щодо суті вимог або за способом їх реалізації. Дайте відповідь на його питання.
6. Зафіксуйте список завдань щодо реалізації вимог та їх тестування, внесіть оцінки трудомісткості за кожним завданням.

Список завдань з реалізації вимог:

Вимога	Завдання з реалізації вимог	Трудовитрати, людино-годин	Виконавець



Урок 3

Робота з вимогами, інструментальні засоби, які використовуються в рамках проекту

© Комп'ютерна Академія ШАГ
www.itstep.org

Усі права на фото-, аудіо- і відеотвори, що охороняються авторським правом і фрагменти яких використані в матеріалі, належать їх законним власникам. Фрагменти творів використовуються в ілюстративних цілях в обсязі, виправданому поставленим завданням, у рамках учбового процесу і в учбових цілях, відповідно до законодавства про вільне використання твору без згоди його автора (або іншої особи, яка має авторське право на цей твір). Обсяг і спосіб цитованих творів відповідає прийнятим нормам, не завдає збитку нормальному використанню об'єктів авторського права і не обмежує законні інтереси автора і правовласників. Цитовані фрагменти творів на момент використання не можуть бути замінені альтернативними аналогами, що не охороняються авторським правом, і відповідають критеріям добросовісного використання і чесного використання.

Усі права захищені. Повне або часткове копіювання матеріалів заборонене. Узгодження використання творів або їх фрагментів здійснюється з авторами і правовласниками. Погоджене використання матеріалів можливе тільки якщо вказано джерело.

Відповідальність за несанкціоноване копіювання і комерційне використання матеріалів визначається чинним законодавством.