**Лекція 6. Інженерія вимог. Визначення вимог до програмних систем.**

**На період** роботи в дистанційній формі навчання на надані в кінці запитання потрібно надати письмові відповіді, надіславши їх на електронну адресу викладача. Файл надавати з іменем у форматі

**OPI<Номер групи><Номер лекції / практичної / лабораторної>[-<Номер завдання>][літера позначення типу роботи L – лекція, P –практична, R – лабораторна]<Прізвищеанглійською>**. Наприклад, O**PI3104L**buts.doc. Кожна відповідь оцінюється в 0,5 балів. Відповіді повинні бути не довгими і змістовними. Не копіюйте фрагментів з різних інформаційних джерел, подумайте і викладіть свою точку зору. При наявності відповідей-"близнюків" відповідь буде зараховуватися першому за часом надсилання.

**Строк виконання цієї роботи ІПЗ-31,32,33 10.11.2024**

**Інженерія вимог** – це розділ програмної інженерії, що займається проблемами отримання вимог до програмного забезпечення і їх документування, а також проблемами управління вимогами. Тобто, вона полягає у виявленні вимог, їх описі (створенні специфікації) і валідації. Цієї теми ми торкалися на лекції №1. Пригадаємо розглянуту схему

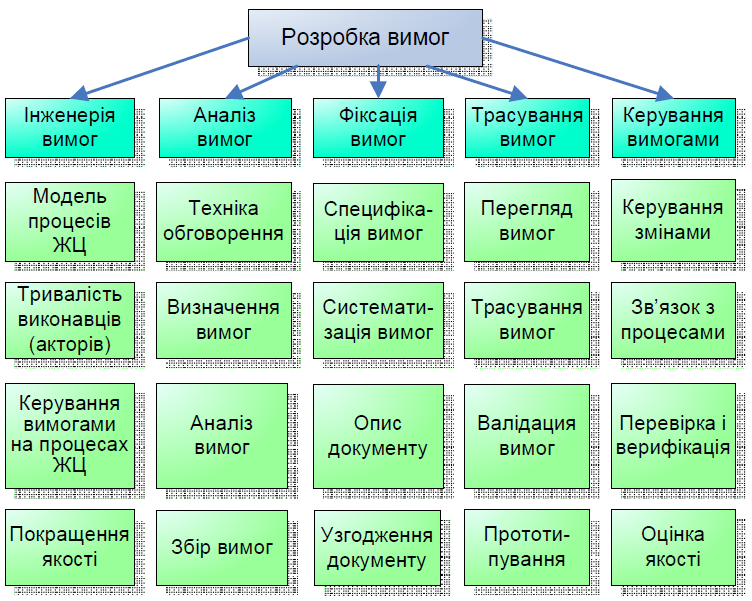


Рисунок 1. Схема процесу розроблення вимог.

Проблеми при виконанні проектів виникають через неформальний збір інформації, передбачуваної функціональності програмної системи (ПС), помилкових або неузгоджених нефункціональних вимог до системи, а також нерегламентованої процедури їх зміни, та інше. Кожна система — це певний перетворювач, чию поведінку і властивості будуватимуся в процесі створення програмної системи для вирішення конкретної задачі. Програмна система — це певна машина, яка вводиться у світ для того, щоб впливати на нього. Частини світу, які впливають на машину або піддаються її впливу, становлять так званий *домен предметної галузі (data domain)* або *домен застосування(application domain).* Опис цього впливу дає відповідь на запитання: "Що робить система?" і визначає вимоги до системи у формі угод між замовником і виконавцем. Як вона це робить, визначає опис машини.

В сучасних інформаційних технологіях фаза життєвого циклу, на якій фіксуються вимоги на розробку програмного забезпечення, є визначальною для його якості, термінів та вартості робіт. Саме на цій фазі мають бути зафіксовані реальні потреби користувачів, що стосуються функціональних, операційних та сервісних можливостей, які береться реалізувати розробник. Таким чином, на цій фазі домовляються замовник і виконавець, що визначає подальші дії виконавця.

Ціна помилок та нечітких неоднозначних формулювань на цій фазі дуже висока, бо час і засоби витрачаються на непотрібну замовникові програму, оскільки він мав на увазі зовсім інше, але не зумів сформулювати свої потреби. Внесення необхідних коректив при цьому може вимагати значних переробок, а інколи й повного перепроектування і, відповідно, перепрограмування. Тим часом статистика показує, що відсоток помилок у постановці завдань перевищує відсоток помилок кодування, і це є наслідком суб’єктивного характеру процесу формулювання вимог та майже повної відсутності засобів його формалізації. Так, згідно зі статистикою, в Америці витрачається щорічно до 82 млрд. доларів на проекти, визнані після реалізації такими, що не відповідають вимогам замовника, інакше кажучи, непотрібними. ***Помилки, допущені на стадії збору вимог, складають від 40 до 60% всіх дефектів проекту.***

**Визначення вимоги**

**Вимоги** – це умови або можливості, якими повинна володіти система або системні компоненти з метою виконання контракту або задоволення відповідним формальним документам та стандартам; це сукупність тверджень відносно атрибутів, властивостей або якостей програмної системи, що підлягають реалізації.

**Поняття «Вимога до ПЗ»**

*Вимоги до ПЗ – сукупність властивостей, які повинно мати ПЗ. Вимоги призначені* для адекватного визначення функцій, умов і обмежень на використання програмного продукту (ПП), а також обсягів даних, технічного забезпечення і середовища для його функціонування. Визначення вимог конкретизується таким чином.

* ***SWEBOK*** *(Software Engineering Body of Knowledge)* Програмні вимоги (Software Requirements) – властивості програмного забезпечення, які повинні бути належним чином представлені в ньому для вирішення конкретних практичних завдань.
* ***RUP*** (*Rational Unified Process*) Вимога – це умова або можливість, якій повинна відповідати система.
* ***IEEE*** (*I triple E, Institute of Electrical and Electronics Engineers, Інститут інженерів з електротехніки та електроніки)* ***Standard Glossary of Software Engineering Terminology (1990)***
  1. Умови або можливості, необхідні користувачу для вирішення проблем або досягнення цілей;
  2. умови або можливості, якими має володіти система або системні компоненти, щоб виконати контракт або задовольняти стандартам, специфікаціям або іншим формальним документам;
  3. документоване уявлення умов або можливостей для пунктів 1 та 2.
* ***Ian Sommerwille &Pete Sawyer*** Вимога – це специфікація того, що повинно бути отримано. Вимоги описують поведінку системи або атрибути та властивості системи. Вимоги можуть бути і обмеженнями на процес розробки системи.

**Властивості вимог:**

* Ясність, недвозначність;
* повнота і несуперечність;
* необхідний рівень деталізації;
* простежуваність;
* тестованість і перевірюваність;
* модифікованість.

Виявленням вимог займається аналітик; інколи у ролі аналітика виступають інші спеціалісти, наприклад, менеджер.

Продуктом процесу виявлення вимог є неформалізований опис, що є фактично контрактом на розробку між замовником та виконавцем. Обидві сторони мають розуміти його зміст, оскільки це розуміння гарантує, що система, у розробку якої буде вкладено працю виконавця, задовольнить замовника. Виконавцем даного процесу виступає розробник, завданням якого є подальше уточнення та формалізація вимог, і їхнє документування, що є однозначно зрозумілими колективу розробників для подальшого проектування, реалізації, тестування та документування програмного продукту. Вимоги можуть представлятися у вигляді текстових тверджень або графічних моделей.

**Класифікація вимог**

Першим кроком аналізу вимог є їх класифікація; потім встановлюється пріоритетність вимог.

***SWEBOK***  не описує підходи до класифікації вимог, а описує можливості угрупування вимог відповідно до їх характеристик таким чином.

* Вимоги до продукту та процесу, які потрібно реалізувати;
* Функціональні та нефункціональні вимоги
* Незалежні властивості
* Вимоги з кількісною оцінкою
* Системні вимоги та програмні вимоги

*Вимоги до продукту і до процесу* визначають умови функціонування і режими роботи ПЗ в операційному середовищі, обмеження на структуру і пам'ять комп'ютерів, на принципи взаємодії програм і комп'ютерів та інше.

Найбільш часто модель вимог поділяють на дві моделі:

* *модель функціональних вимог* - містить вимоги і властивості, що визначають очікувану функціональну поведінку системи. Вимоги, що відображають можливості, які повинна забезпечити система, назвали *функціональними вимогами (functional requirement);*
* *модель нефункціональних вимог* - містить обмеження на рівні продуктивності, очікуваної від системи (наприклад, час відгуку, стійкість шифрування, число транзакцій за секунду). Вимоги, що відображають обмеження, пов’язані з функціонуванням системи, назвали *нефункціональними вимогами (notfunctional requirement).*

Перша з наведених категорій дає відповідь на запитання: "Що робить система?", а друга визначає характеристики її роботи, наприклад, що ймовірність збою системи протягом години не може перевищувати однієї мільйонної.

Функціональні вимоги пов’язані із семантичними особливостями проблемної галузі, в межах якої планується розробка. Проблема термінологічних розбіжностей для них є досить впливовим фактором ускладнення. На жаль, робляться спроби розв’язання її шляхом стандартизації термінології лише для кількох проблемних галузей, для яких властивий інтенсивний розвиток комп’ютерних систем, наприклад, для авіоніки та медицини. Але можна засвідчити наявність стійкої тенденції до створення стандартизованого понятійного базису більшості проблемних галузей, які набувають певного досвіду комп’ютеризації. Помилки і різночитання, які виникають при виявленні вимог до системи, виявляються одними з найдорожчих. **Вимоги** - це те вихідне розуміння завдання розробниками, яке є основою всієї розробки.

Нефункціональні вимоги можуть виступати як окремий чисельний показник, наприклад, час чекання відповіді абонента не може перевищувати півсекунди. Інколи вони можуть мати комплексний характер і потребувати для свого втілення сукупності деталізованих властивостей, наприклад, "підвищити кількість обслуговуваних клієнтів удвічі".

***Класи нефункціональних вимог***

Є кілька класів нефункціональних вимог, суттєвих для більшості програмних систем, які виражають обмеження, актуальні для багатьох проблемних галузей. Серед них назвемо такі:

* вимоги конфідеційності;
* відмовостійкість;
* число клієнтів, які мають одночасно доступ до системи;
* вимоги безпеки;
* час очікування відповіді на звернення до системи;
* виконавські якості системи (обмеження щодо ресурсів пам'яті, швидкість реакції на звернення до системи та інше).

***Незалежні властивості*** – це вимоги, пов’язані зі специфікою предметної області.

***Вимоги з кількісною оцінкою*** визначають показники якості ПП.

***Показник якості (продукції)*** - це кількісна характеристика одного або декількох властивостей продукції, що входять до її якості, то розглядається відповідно до певних умов її створення та експлуатації або споживання.

***Програмні вимоги*** - Software Requirements - властивості програмного забезпечення, які повинні бути належним чином представлені в ньому для вирішення конкретних практичних задач.

Схематично класифікація та зв’язок вимог показано на рис.2, 3.

*Функціональні вимоги* *Нефункціональні вимоги*

Документ про образ та межи проекту

Документ про варіанти використання

Специфікація вимог до ПЗ

Рисунок 2. Класифікація та зв’язок вимог

Рисунок 3. Піраміда вимог за Леффінгуелом [7]

Бізнес-вимоги та бізнес цілі - це вимоги, що відображають високорівневі цілі організації або замовників системи. Як правило, їх висловлюють ті, хто фінансує проект, замовники системи.

***Яких вимог не повинно бути:***

* Деталей дизайну або реалізації
* Даних про планування проекту
* Відомостей про тестування

**Стадія формування вимог до П**З

Стадія формування вимог до ПЗ - це найважливіша стадія, оскільки вона визначає успіх усього проекту. Ця стадія складається з таких етапів:

1) планування робіт включає визначення мети розробки, попередню економічну оцінку проекту, створення плану-графіка виконання робіт, навчання спільної робочої групи;

2) проведення обстеження діяльності об'єкта (організації) автоматизації, у рамках якого здійснюються: попереднє виявлення вимог до майбутньої системи; визначення структури організації; визначення переліку цілей організації; аналіз розподілу функцій за підрозділами і між співробітниками; виявлення функціональних взаємодій між підрозділами, інформаційних потоків усередині підрозділів і між ними, зовнішніх стосовно організації об'єктів і зовнішніх інформаційних взаємодій; аналіз наявних засобів автоматизації діяльності організації;

3) побудову моделей діяльності організації, що передбачає обробку матеріалів обстеження;

4) побудову двох видів моделей:

* + моделі ***"як є",*** що відображає наявний на момент обстеження стан справ і допомагає зрозуміти, як саме функціонує певне підприємство, а також виявити вузькі місця і сформулювати пропозиції щодо поліпшення ситуації;
  + моделі ***"як має бути",*** що відображає схему про нові технології роботи підприємства. Кожна з моделей містить повну функціональну й інформаційну модель діяльності організації, а також у разі потреби модель, що описує динаміку поведінки організації, і може включати відомості про
  + відмовостійкість;
  + кількість клієнтів, що одночасно мають доступ до системи;
  + вимоги безпеки;
  + час очікування відповіді на звернення до системи;
  + виконавські властивості системи (обмеження щодо ресурсів пам'яті, швидкість реакції на звернення до системи тощо).

**Зацікавлені особи на стадії збирання вимог**

Ніде більше, як на стадії збору вимог, так тісно не пов'язані інтереси всіх зацікавлених у проекті осіб з успіхом проекту. До зацікавленим у проекті осіб належать:

1. ***Замовники***, які фінансують проект або набувають продукт для вирішення якихось бізнес-завдань.

2. ***Користувачі***, які взаємодіють безпосередньо чи не безпосередньо з застосуванням (підклас замовників).

3. ***Аналітики вимог***, які пишуть вимоги і передають їх розробникам.

4. ***Розробники***, які створюють, розгортають і підтримують продукт.

5. ***Тестери***, які визначають відповідність поведінки ПЗ бажаному.

6. ***Технічні письменники***, які відповідають за створення керівництва користувачів, тренувальні матеріали та довідкову систему.

7. ***Менеджер по проекту***, який планує процес і керує командою розробників від початку і до успішного випуску продукту.

8. ***Співробітники правового відділу***, які стежать, щоб продукт не суперечив усім чинним законам і постановам.

9. ***Розробники***, які повинні побудувати продукт, що містить дане ПЗ.

10. ***Співробітники відділу продажів і маркетингу***, виїзної служби підтримки та інші, кому доведеться працювати з продуктом і його користувачами.

***Ідентифікація зацікавленої особи***

Діючими особами процесу формулювання вимог є носії інтересів замовників (досить часто замовника репрезентують кілька професійних груп, потреби яких можуть мати не лише відмінності, а й навіть суперечності), оператори, котрі здійснюють обслуговування під час функціонування системи, розробники системи. Загалом можна так ідентифікувати зацікавлених в ПП осіб:

* кожен, хто користується системою (користувачі та обслуговуючий персонал);
* будь-який, хто отримує вигоду з системи (функціональну, політичну, фінансову і соціальну);
* залучений в покупку або забезпечення системи організації, які регулюють аспекти системи (фінансові, безпека, та інші);
* люди або організації, налаштовані проти системи (негативні зацікавлені особи);
* організації, відповідальні за систему, які взаємодіють з системою згідно з проектом;
* ті організації, які об'єднуються горизонтально з організацією, для якої аналітик проектує систему.

**Особливості збирання та аналізу бізнес-вимог**

***Продукт під замовлення***

Характеристики процесу:

* сектор ринку з самого початку може бути не визначено;
* цілі продукту ґрунтуються на конкурентному аналізі.

Особливості збору та аналізу бізнес-вимог:

* відразу за визначенням вихідних передумов (стимулів) йде огляд конкурентів;
* далі йде визначення цільового сегмента ринку і потреб його замовників;
* в кінці визначаються цілі продукту і критерії його успіху.

***Встроєні застосунки***

Характеристики процесу залежать від того, чи є це продуктом під замовлення (наприклад, ПЗ для мікрохвильової печі) або продуктом для відкритого ринку (наприклад, ПЗ мобільних телефонів)

***Визначення стимулів*** є важливою ланкою в процесі формування вимог для проведення в подальшому їх пріоритетності. Стимулами можуть бути:

* потреба ринку;
* виробнича необхідність;
* потреба замовника;
* технічний прогрес;
* юридичні обмеження або норми.

Не менш важливою ланкою є також ***визначення цілей продукту і критеріїв успіху,*** які зазвичай поділяються на фінансові та нефінансові таким чином:

***фінансові:***

* Досягти обсягу продажів X одиниць або доходу, рівного $ Y, за Z місяців.
* Отримати Х% прибутку або доходу з інвестицій протягом Y місяців.
* Заощадити Х $ на рік, які зараз витрачаються на обслуговування системи.
* Зменшити витрати на підтримку на Х% за Z місяців.
* Отримати не більше X дзвінків у службу обслуговування по кожній одиниці товару і Y дзвінків по гарантії кожної одиниці товару протягом Z місяців після випуску товару.

***нефінансові:***

* Досягти показника задоволення покупців, рівного, принаймі, X, протягом Y місяців з часу випуску товару.
* Збільшити продуктивність обробки транзакцій на Х% і знизити рівень помилок даних до величини не більше Y%.
* Розробити надійну платформу для сім'ї пов'язаних продуктів

**Джерела вимог**

Джерелами відомостей про вимоги можуть бути:

- мета та завдання системи, які формулює замовник. Потрібно зазначити, що при безумовній пріоритетності думки замовника треба мати на увазі небезпеку неоднозначного розуміння цих формулювань замовником і розробником, а також властивість людини замовчувати багато визначальних подробиць, що є не ознакою злої волі, а лише наслідком або забудькуватості, або впевненості, що це "загальновідомо". Тому формулювання замовника підлягають глибокому осмисленню з боку виконавця;

- діюча система або колектив, який виконує її функції. Досить часто система, яку замовляють, має замінити собою попередню систему, що вже не задовольняє замовника, або певні функції діючого персоналу. Вивчення та фіксація наявних функціональних можливостей створює базу, розширення якої дозволить сформулювати вимоги до потрібної системи, в яких урахується набутий досвід замовника. Для цього джерела також є певна небезпека перенесення вад організації попередньої системи в нову. Наприклад, розподіл обов’язків серед персоналу певного відділу склався історично впродовж послідовного розширення кадрів і є недоцільним відносно функцій відділу. Тому, вивчаючи діючу систему, треба вміло відокремити потреби проблеми, яку розв’язує система, від закладених у стару систему невдалих організаційних рішень. З погляду на це, процес урахування діючої системи під час складання вимог до нової доцільно провести за три кроки:

а) за першим кроком вивчаємо фізичну структуру діючої системи (незалежно від того, автоматизована вона чи "людська");

б) за другим кроком робимо логічне узагальнення виявлених на першому етапі функцій та особливостей, виділяючи ті, що відображають поставлену перед розробкою проблему, на відміну від закладених у стару систему окремих шляхів розв’язання проблеми (інколи не зовсім вдалих);

в) за третім кроком визначаємо логічне розширення функцій, виявлених на другому кроці, яке відповідає потребам нової системи як розвитку існуючої в заданому напрямі; визначені функції формулюємо як вимоги до нової системи;

- загальні знання щодо проблемної галузі замовника. Він формулює і розуміє свої проблеми в термінах понять певної проблемної галузі і, як згадувалося вище, замовчує подробиці, котрі належать саме до загальновідомих знань (але, на жаль, тільки серед професіоналів відповідної галузі);

- відомчі стандарти замовника, котрі стосуються організаційних вимог, середовища функціонування майбутньої системи, її виконавських та ресурсних можливостей.

**Методи виявлення вимог**

* Інтерв’ю, опитування, анкетування;
* мозковий штурм, семінар;
* спостереження за виробничою діяльністю, «фотографування» робочого дня;
* спостереження за роботою діючої системи з метою відокремлення її органічних властивостей від тих, що зумовлені структурою кадрів;
* фіксація сценаріїв усіх можливих випадків використання системи, виконуваних при цьому системою функцій, ролей осіб, котрі запускають ці сценарії або взаємодіють з системою під час її функціонування
* аналіз нормативної документації;
* аналіз моделей діяльності;
* аналіз конкурентних продуктів;
* аналіз статистики попередніх версій системи.

**Першочергові перешкоди, які виникають при виявленні вимог**

1. Образ і межі проекту ніколи не визначені ясно.

2. Замовники дуже зайняті, щоб працювати з аналітиками і програмістами над вимогами.

3. Заступники користувачів - менеджери по продуктах, по розробці, менеджери користувачів або маркетологи - викликаються говорити від імені клієнтів, але не точно представляють їхні потреби.

4 Вимоги існують тільки у головах "експертів", які працюють у вашій організації, і ніколи не фіксуються у письмовому вигляді.

5. Замовники наполягають, щоб критикувалися всі вимоги, без урахування їх пріоритетів.

6. Розробники отримують двозначну і неповну інформацію, тому при кодуванні їм доводиться робити припущення.

7. Взаємодія між розробниками і замовниками обмежується зовнішнім виглядом користувальницького інтерфейсу і не зачіпає того, що ж дійсно клієнти збираються робити за допомогою програми.

8. Ваші замовники підписують вимоги і потім постійно змінюють їх.

9. Проект розростається, коли ви приймаєте зміни вимог, графік при цьому порушується, тому що ніяких додаткових ресурсів не виділяється і ніякі функції не видаляються.

10. Запити на зміну вимог губляться по дорозі: ні ви, ні ваші клієнти не знають статус кожного запиту.

11. Замовники наполягають на певній функціональності, яку розробники і створюють, однак ці можливості системи клієнтам не потрібні.

**Встановлення пріоритетності вимог**

Наступним кроком аналізу вимог є встановлення їхньої пріоритетності, бо, як було вказано вище, вимоги, висунуті різними носіями інтересів у системі, можуть конфліктувати між собою. Крім того, кожна з вимог потребує для свого втілення певних ресурсів, надання яких може залежати також від визначеного для неї пріоритету.

Визначення впливу вимог на потреби в ресурсах є також кроком процесу аналізу вимог.

Ще одним із важливих завдань аналізу є передбачення можливих змін у зібраних вимогах і забезпечення можливостей внесення передбачуваних змін без суттєвого перегляду всієї системи. Такі можливості мають забезпечити живучістьсистеми та її здатність до адаптації.

Нарешті, в процесі аналізу вимог має бути перевірено правдивість та відповідність їх інтересам замовника, висловленим усіма носіями інтересів цього замовника.

Продуктом процесу аналізу є побудована модель проблеми, орієнтована на її розуміння, якого має досягнути виконавець до початку проектування системи.

Аналіз вимог може бути довгим та важким процесом що вимагає використання тонких психологічних навиків. Нові системи змінюють середовище і відношення між людьми, тому важливо розпізнати всі зацікавлені сторони, взяти до уваги всі їхні потреби, і переконатись що вони розуміють наслідки які приносить нова система. Аналітики можуть використати кілька методів щоб отримати від споживача вимоги. Історично це включає проведення інтерв'ю, чи фокус-груп (яку в цьому контексті частіше називають як майстерня вимог) і створення списків вимог. До сучасніших підходів відносять прототипування, та прецеденти. За потреби аналітик використає комбінацію цих методів щоб встановити точні вимоги зацікавлених сторін, так щоб система відповідала бізнес-потребам.

*Для самостійного вивчення*: Вивчення лекційного матеріалу та додаткових джерел. Розгляд запитань і виконання завдань для самостійної роботи, запропонованих на лекції.

*Запитання для самоперевірки.*

1. В чому полягає важливість процесу збирання вимог ?
2. Надайте визначення вимоги та конкретизуйте його стосовно ПЗ.
3. Визначте класифікацію вимог.
4. Яким чином визначаються зацікавлені в програмному продукті особи на стадії збирання вимог?
5. Визначте джерела вимог.
6. Які Ви знаєте методи виявлення вимог
7. Які перешкоди виникають при виявленні вимог?
8. Для чого встановлюють пріоритети вимог?

*Запитання для письмової відповіді.*

1. Надайте наступні визначення, продовживши речення:

***Виявлення вимог*** – це …

***Модель процесу визначення вимог*** – це схема процесів ЖЦ, які …

1. Поясніть властивості вимог, в разі потреби, надайте приклади.

*Література*.

1. Лавріщева К.М. Програмна інженерія. Електронний підручник. URL: http://csc.knu.ua/uk/library/books/lavrishcheva-6.pdf.
2. Лавріщева К.М. Програмна інженерія. – Підручник.–К.:Академперіодика, 2008.–415с.
3. Бабенко Л.П., Лавріщева К.М*.* Основи програмної інженерії.– Навч. посібник.–К.: Знання, 2001.– 269с.
4. Проектування інформаційних систем: Посібник // За редакцією Пономаренка В.С. – К.: Видавничий центр "Академія". 2002. ­ 488 с. URL: <http://www.dut.edu.ua/uploads/l_874_10304054.pdf>.
5. И. Соммервиль. Инженерия программного обеспечения, 6 изд. – И.д. "Вильямс", 2002. URL: https://www.studmed.ru/download/sommervill-ian-inzheneriya-programmnogo-obespecheniya\_4935164f089.html
6. Уокер Ройс. Управление проектами по созданию программного обеспечения. Пер. с англ. – М.: из-во «Лори», 2002
7. Леффингуэлл Дин, Уидриг Дон. Принципы работы с требованиями к программному обеспечению. Унифицированный подход. М.: Вильямс, 2002. — 448 с.