

Міністерство освіти і науки України
Запорізький національний технічний університет

МЕТОДИЧНІ ВКАЗІВКИ

до самостійних робіт
з дисципліни

“Аналіз вимог до програмного забезпечення”

для студентів
напряму підготовки 6.050103
“Програмна інженерія ”
всіх форм навчання

2016

Методичні вказівки до самостійних робіт з дисципліни “Аналіз вимог до програмного забезпечення” для студентів напряму підготовки 6.050103 “Програмна інженерія” всіх форм навчання / Г. В Табунщик, Т. І. Каплієнко, Н. О. Миронова. – Запоріжжя: ЗНТУ, 2012. – 35 с.

Автори: Г. В. Табунщик, к.т.н., професор
 Т. І. Каплієнко, к.т.н., доцент
 Н. О. Миронова, к.т.н., доцент

Рецензент: Р.К.Кудерметов, к.т.н., доцент

Відповідальний
за випуск: Т. І. Каплієнко, к.т.н., доцент

Затверджено
на засіданні кафедри
програмних засобів

Протокол №
від “ ” _____ 2016 р.

ЗМІСТ

Вступ	4
Самостійна робота №1 Розробка Use Case Specification	5
Самостійна робота №2 Інтеграція Requisite Pro з MS Project 2003	16
Додаток А Перелік тем для самостійних робіт	35

ВСТУП

Дисципліна "Аналіз вимог до програмного забезпечення" спрямована на отримання студентом базових знань та практичних навичок з виявлення, аналізу та керування вимогами.

Успішний досвід розробки показує, що ефективне керування вимогами є ключовим фактором всього процесу розробки програмного забезпечення. Вимоги визначають те, що повинна робити система. Тому протягом всього життєвого циклу проекту необхідно організувати ефективну роботу з ними. Першим кроком у цьому напрямку є організація зберігання всіх виявлених вимог, що обумовлює актуальність вивчення даної дисципліни.

Особливу увагу треба приділяти правильному оформленню звітів з самостійних робіт. Звіт повинен містити: титульний аркуш (на ньому вказують назву міністерства, назву університету, назву кафедри, номер, вид і тему роботи, виконавця та особу, що приймає звіт, рік); мету, варіант і завдання роботи; лаконічний опис теоретичних відомостей; повний текст всіх версій розроблених документів.

Звіт виконують на білому папері формату А4 (210 x 297 мм). Текст розміщують тільки з однієї сторони листа. Поля сторінки з усіх боків – 20 мм. Аркуші скріплюють за допомогою канцелярських скріпок. Для набору тексту звіту використовують редактор MS Word: шрифт Times New Roman, 14 пунктів. Міжрядковий інтервал: полуторний – для тексту звіту, одинарний – для листингів програм і роздруровок даних.

Під час співбесіди студент повинний виявити знання про мету роботи, з теоретичного матеріалу, зі змісту основних розділів розробленого звіту з демонстрацією результатів на конкретних прикладах.

Для виконання самостійних робіт рекомендується використовувати IBM Rational Requisite Pro.

САМОСТІЙНА РОБОТА №1 РОЗРОБКА USE CASE SPECIFICATION

Мета роботи

Навчитись розробляти специфікації для сценаріїв використання з використанням *IBM Rational Requisite Pro*.

Основні теоретичні відомості

Створення сценаріїв використання (Use Cases). Сценарій використання (Use Cases) або прецеденти – це опис системи в термінах послідовності дій. Він повинен мати значимий результат або певне значення для особи, яка взаємодіє із системою (actor – діюча особа). Характеристики прецедентів:

- ініціюються діючою особою;
- є моделлю взаємодій між діючою особою й системою;
- описують послідовність дій;
- містять у собі функціональні вимоги;
- повинні мати деяке значення для діючої особи;
- надають повний, що має зміст потік подій.

Призначення прецедентів – сприяти згоді між розроблювачами, замовниками й користувачами в питаннях функціональності системи. Прецедент стає свого роду контрактом між розроблювачами й замовниками. Він також є основою для реалізації прецедентів, які розробляються при проектуванні системи. Більше того, із прецедентів можна скласти посібник користувача. Прецеденти можуть також бути корисні в плануванні формального змісту ітерацій і дозволяють розроблювачам системи краще зрозуміти призначення системи.

При розробці прецедентів будемо визначати сценарії (алгоритми, scenarios) – особливі шляхи проходження прецеденту. Із прецедентів можна створювати діаграми послідовностей, діаграми зв'язків і діаграми класів. Вони також використовуються як вихідні дані для тестових сценаріїв (test cases).

Прецеденти – це гарний спосіб вираження функціональних вимог до системи. Модель прецедентів містить всі сценарії використання, діючі особи (хто взаємодіє зі сценаріями використання), а також відносини між ними. Модель визначає взаємини системи й діючих осіб.

Визначення діючих осіб. Діюча особа або актор (actor) – це зовнішній стосовно системи користувач, взаємодіючий із системою. Актором може бути як людина, так і інша система. Кілька прикладів діючих осіб: користувачі системи; адміністратори; особа, що надає інформацію для системи; зовнішні системи, що надають дані; зовнішні системи, що одержують повідомлення.

Всі зацікавлені особи також є кандидатами в актори. В нашому прикладі визначені наступні зацікавлені особи: Користувач, Розробник, Власник.

Розглянемо, хто з них може бути діючою особою. Власник – може бути діючою особою, якщо він надасть кілька сценаріїв використання, специфічних тільки для нього. Це залежить від того, чи є у власника особливі привілеї, та чи є якась функціональність, доступна тільки для власника. Якщо він має доступ до тієї ж функціональності, що й адміністратор, то створювати діючу особу для власника, тому що адміністратор покриває його функціональність.

Користувач 1 і Користувач 2 можуть бути з'єднані в одну діючу особу «Користувач». Діюча особа визначає роль, а не певну особу, так що існує тільки одна діюча особа для всіх осіб, що мають однакову роль.

Також потрібна роль із назвою користувач, під яким ми будемо розуміти всіх осіб, що мають доступ до системи. Із цією діючою особою будуть зв'язані такі сценарії використання, як рестрація або вхід у систему. Багато діючих осіб успадковують функціональність від користувача.

Розроблювач не є діючою особою, тому що після створення системи, розроблювачі не взаємодіють із нею.

Адміністратор – це діюча особа, що виконує різні завдання по адмініструванню системи. В нашому прикладі його не має.

Визначення прецедентів. Кілька питань, які можуть допомогти у визначенні прецеденту:

Яку функціональність кожна діюча особа очікує від системи?

Чи повинні діючі особи бути інформовані про події, що відбуваються в системі?

Яку інформацію діючі особи повинні надавати системі?

Яку інформацію діючі особи повинні одержувати від системи?

Про які події, що відбуваються поза системою, система повинна бути сповіщена?

Сценарії використання можуть бути визначені впродовж семінару по вимогах.

Кілька рекомендацій по створенню сценаріїв використання:

– кожний сценарій використання повинен взаємодіяти, принаймні, з однією діючою особою;

– кожний сценарій використання повинен ініціюватися діючою особою.

Назви прецедентів повинні бути значеннєсвими. Ніколи не називайте прецеденти однаковими назвами. Назви повинні бути зрозумілі не тільки команді розроблювачів, але також замовникам і користувачам.

Прецеденти повинні описувати функціональність, а не реалізацію. Створення компонента сесії – не зовсім гарний сценарій використання. Повинно бути зрозуміло, хто саме ініціює прецедент.

Не забувайте, що сценарії використання не можуть бути занадто малими або занадто більшими. Наприклад, Надання інформації із кредитної карти є некоректним прецедентом, тому що він не описує повний потік подій і не надає ніякого значення для діючої особи. У прецеденті Замовлення квитка – тільки один крок. Якщо прецедент містить тільки один або два кроки, то можливо, що це некоректний сценарій використання.

Інший приклад – Виконання завдань по адмініструванню. Цей прецедент занадто загальний і може бути розділений на кілька прецедентів, що мають більше точний зміст, наприклад Генерація звітів або Відновлення інформації користувача.

В нашому проєкті для діючих осіб були визначені наступні прецеденти, які наведені в табл.1.1.

Таблиця 1.1 – Прецеденти проєкту

Актор	Прецедент
Користувач	Реєстрація Вхід у систему Відправлення електронного листа Перегляд електронних листів Редагування адресної книги Довідка

Документ Use Case Specification. Для специфікації прецедентів будемо використовувати шаблон Use Case Specification, що містить наступні розділи:

- 1 Use Case Name (Найменування прецеденту)
 - 1.1 Brief description (Короткий опис)
- 2 Flow of Events (Потоки подій)
 - 2.1 Basic flow (Основний потік)
 - 2.2 Alternative flows (Альтернативні потоки)
 - 2.2.1 Alternative flow 1 (Альтернативний потік 1)
 - 2.2.2 Alternative flow 2 (Альтернативний потік 2)
- 3 Special requirements (Особливі вимоги)
- 4 Preconditions (Передумови)
- 5 Postconditions (Постумови)
- 6 Extension points (Крапки розширення)

Короткий опис (Brief Description). Короткий опис повинний зрозуміло пояснювати призначення документа. Також необхідно вказати всіх діючих осіб, взаємодіючих із системою.

Основний Потік (Basic Flow). Основний потік містить основний позитивний сценарій.

Будемо використовувати наступні правила для позначення потоків:

Основний потік (Basic flow): B.

Альтернативний потік (Alternative flows): A1, A2, A3...

Кроки в основному потоці: B1, B2, B3...

Кроки в альтернативному потоці 1: A1.1, A1.2, A1.3...

Кроки в альтернативному потоці 2: A2.1, A2.2, A2.3...

Не існує універсального стандарту для нумерації цих кроків у сценарії використання. Деякі використовують послідовну нумерацію 1,2,3..., а інші використовують 2.1, 2.2, 2.3... (тому що вони перебувають у пункті 2 документа). Деякі взагалі не використовують нумеровані кроки.

В нашому проекті *Поштовий клієнт*, основний потік сценарію використання *Відправлення електронного листа* може виглядати таким чином

- B1. Система відображає головне вікно програми.
- B2. Користувач натискає кнопку «Відправити лист».
- B3. Система відображає вікно відправлення повідомлення.
- B4. Користувач вводить: поштову адресу одержувача.

B5. Користувач вводить: ступінь важливості, тему й текст повідомлення.

B6. Користувач натискає кнопку «Відправити».

B7. Система відображає повідомлення про успішність відправлення повідомлення.

B8. Система відображає головне вікно програми.

Альтернативний Потік (Alternative Flow). Альтернативні потоки (Alternative flows) – це можливі варіанти, включаючи помилкові ситуації.

Які інші дії можуть бути виконані на кожному кроці основного потоку?

Які помилки можуть виникнути на кожному кроці (невірні дані, пропущена інформація, проблеми з'єднання)?

Чи існує таке поводження системи, що може виникнути в будь-який час (таке як вихід, друк, допомога)?

Чи можуть умови (такі як певні комбінації даних, що вводяться,) істотно змінити потік?

Кілька альтернативних потоків для сценарію використання *Відправлення електронного листа*:

A1. Додавання файлу

A1.1. На кроці B3 Користувач вирішує прикріпити файл до повідомлення.

A1.2. Користувач натискає кнопку «Додати файл».

A1.3. Користувач вибирає файл у себе на машині.

A1.4. Система відображає повідомлення про прикріплення файлу до листа.

A1.5. Потік повертається на крок B4 основного потоку.

A2. Помилка в адресі одержувача.

A2.1. На кроці B4 Користувач увів невірний/неіснуючу адресу одержувача.

A2.2. Система відображає повідомлення про необхідність введення поштової адреси одержувача.

A2.3. Користувач виправляє/уписує поштова адреса одержувача.

A2.4. Потік вертається на крок B7 основного Потоку.

A3. Чернетка

A3.1. На кроці B6 Користувач натискає кнопку «У Чернетки»

A3.2. Система відображає повідомлення про успішність збереження листа в чернетках.

A3.3 Потік вертається на крок B8 основного Поток.

A4. Скасування

A4.1. На кроці B4 (або B5, або B6) Користувач натискає кнопку «Скасувати»

A4.1 Потік вертається на крок B8 основного Поток

Альтернативні потоки повинні мати унікальну послідовність дій. Вони не можуть бути різними тільки в даних.

Особливі вимоги (Special Requirements). Даний пункт містить всі вимоги, що відносяться до прецедентів, які не були покриті потоком подій. Звичайно це нефункціональні вимоги. Однак якщо вимога досить загальна й зустрічається в багатьох сценаріях використання, вона повинна бути описана в *Додатковій Специфікації*.

Передумови (Preconditions). Передумова – це стан, у якому повинна перебувати система перед початком сценарію використання. Часто це позитивний результат іншого прецеденту. Наприклад, передумовою для сценарію використання Пошук замовлення може бути: «Адміністратор повинен виконати вхід у систему».

Постумова (Postconditions). Постумова – це стан, у якому повинна перебувати система після закінчення сценарію використання. Поки постумова не зазначена в кожному конкретному потоці, вона повинна бути однаковою для всіх альтернативних потоків, а не тільки для основного.

Крапки Розширення (Extension Points). Крапка розширення – це місце, звідки може бути витягнутий розширений сценарій використання. Нариклад, сценарій використання *Відправлення електронного листа* може мати наступне розширення:

Назва: Додавання файлу

Місце розташування: Після кроку B3 основного потоку.

Порядок виконання роботи

Створення Use Case Specification. Для створення документа *Use Case Specification* в *RequisitePro*, необхідно виконати дії,

аналогічні діям при створенні документа *Stakeholder Request* і *Документа Vision*.

1. У *Провіднику RequisitePro*, виберіть папку Use Cases.
2. Виберіть File⇒New⇒Document (Файл⇒Новий⇒Документ).
3. Заповніть поля в діалоговому вікні Document Properties (Властивості Документа), як показано на рис.1.1. Натисніть ОК.
4. Ескіз документа відкриється в Microsoft Word.
5. Заповніть всі пункти документа.

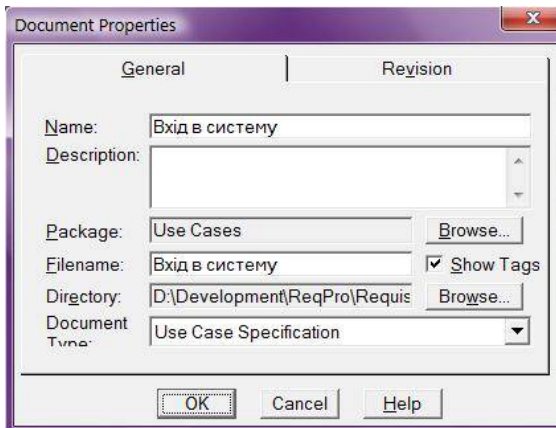


Рисунок 1.1 – Діалогове вікно Document Properties (Властивості Документа)

Створення вимог сценарію використання. Перед створенням вимог в *RequisitePro* необхідно вирішити, що буде розумітися під однією вимогою: цілий сценарій використання; кожний альтернативний потік; група відповідних кроків; кожний крок.

Це рішення залежить від рівня бажаної деталізації в трасуванні (установки зв'язку). Єдиною перевагою збереження цілих сценаріїв використання є обмеження витрат у цілому, але такий підхід може бути не зовсім гарним для відстеження коректного призначення всіх завдань. Збереження кожного кроку як окрема вимога забезпечить гарне трасування (зв'язок), але це вимагає величезних витрат і може затьмарити основну функціональність і багато деталей. Найбільш кращим підходом є зберігання основного й альтернативного потоків у

якості окремих елементарних вимог. Проте, це рішення може змінюватися для різних проектів.

Процес створення вимог сценарію використання – такий же, що був використаний для створення вимог STRQ і FEAT.

1. Виділіть назву сценарію використання.

2. Правою кнопкою мишки натисніть на назві й виберіть New Requirement (Нова вимога), або виберіть RequisitePro⇒ Requirement⇒New (Вимога⇒Нова), або натисніть на іконку New Requirement (Нова вимога).

3. З'явиться діалогове вікно Requirement Properties (Властивості Вимоги). На закладці General (Загальне) не потрібно нічого вводити. Тип показується за замовчуванням: «UC: Use Case». Не потрібно вводити Name (Назва), тому що для визначення вимоги може використовуватися Text (Текст).

4. Натисніть OK на діалоговому вікні Requirement Properties (Властивості вимоги).

Замість виділення назви сценарію використання можна включити у вимогу весь потік. Це дозволить автоматичне відстеження змін, якщо хтось додасть, видалить або змінить будь-який крок. Будь-які зміни будуть відображені в підозрілих зв'язках. Для вимоги можна використовувати заголовок як ім'я, а потік як опис. Якщо використовувати заголовок як вимогу, тоді при зміні опису потоку буде потрібно вручну позначати зв'язки як підозрілі.

RequisitePro надає можливість установки ієрархії вимог. Це дуже зручно, особливо при структуруванні вимог сценаріїв використання. Можна визначити кожний сценарій використання як батьківський і приєднати всі альтернативні потоки як дочірні вимоги. Для створення дочірньої вимоги виконайте наступні дії:

1. Виділіть назву альтернативного потоку. Правою кнопкою мишки натисніть на назві й виберіть New Requirement (Нова вимога), або виберіть RequisitePro⇒ Requirement⇒New (Вимога⇒Нова), або натисніть на іконку New Requirement (Нова вимога).

2. З'явиться діалогове вікно Requirement Properties (Властивості Вимоги).

3. Виберіть закладку Hierarchy (Ієрархія).

4. У списку Parent (Батьківський), виберіть «choose parent» («вибрати батьківський»). З'явиться діалогове вікно Parent Requirement Browser (Огляд батьківської вимоги), як показано на рис.1.2.

5. Виберіть батьківську вимогу.

6. Натисніть ОК у діалоговому вікні Parent Requirement Browser.

7. Натисніть ОК у діалоговому вікні Requirement Properties (Властивості Вимоги), як показано на рис.1.3.

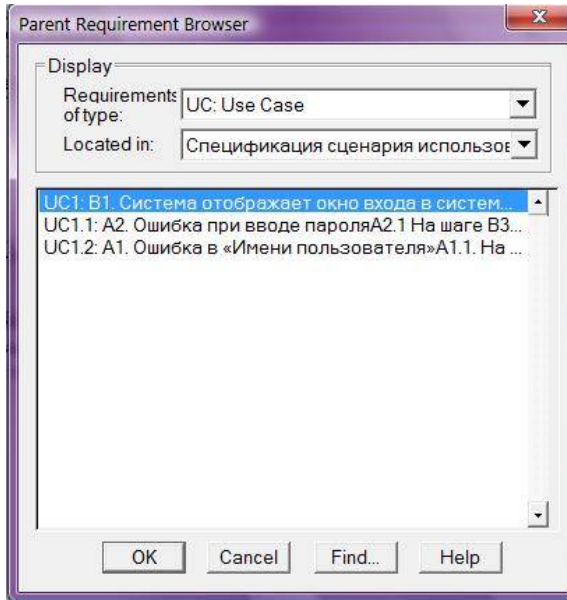


Рисунок 1.2 – Вибір назви сценарію використання в діалоговому вікні Parent Requirement Browser (Огляд батьківської вимоги)

Після збереження документа, *RequisitePro* призначає унікальний номер вимозі. У випадку дочірньої вимоги номер складається з номера батьківської вимоги, крапки й номеру дочірньої вимоги. Ієрархія вимог може бути переглянута із *Провідника*, як показано на рис.1.4.

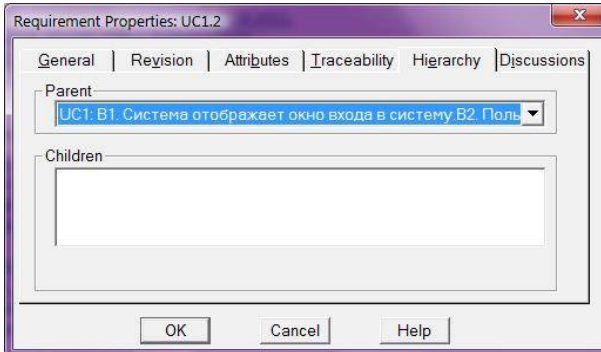


Рисунок 1.3 – Закладка Hierarchy (Ієрархія) у діалоговому вікні Requirement Properties (Властивості вимоги)

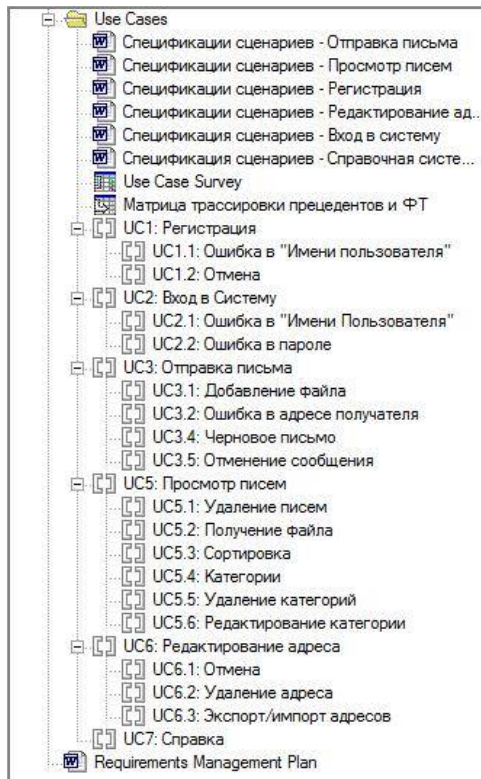


Рисунок 1.4 – Ієрархія сценаріїв використання в *Провіднику*

Завдання на самостійну роботу

1. Використовуючи рекомендовану літературу та дані методичні вказівки, ознайомитися з основними принципами розробки прецедентів.
2. Сформулювати акторів та прецеденти сценарії використання для вашої системи.
3. Розробити таблицю прецедентів для своєї задачі.
4. Розробити специфікацію для 5 прецедентів.
5. Оформити звіт (розроблені специфікації).
6. Відповісти на контрольні запитання.

Контрольні запитання

1. Визначення прецеденту.
 2. Які існують характеристики прецедентів?
 3. Призначення прецеденту.
 4. Визначення діючих осіб.
 5. Яким чином створюється документ Use Case Specification?
- Наведіть склад документу.
6. Поняття основного потоку.
 7. Поняття альтернативного потоку.
 8. Поняття особливих вимог, посту мови, передумови, точки розширення.

САМОСТІЙНА РОБОТА №2

ІНТЕГРАЦІЯ REQUISITE PRO З MS PROJECT 2003

Мета роботи

Створення задач в MS Project з вимог в Requisite Pro 2003 і установка зв'язків між вимогами й задачами.

Основні теоретичні відомості

Рекомендації з інтеграції Requisite Pro в MS Project.

Рекомендується створювати задачі для вимог, виконання яких варто відстежити в MS Project, наприклад функціональних вимог до системи. Можна формувати ієрархію задач у відповідності ієрархії вимог до системи. Якщо у вимоги є атрибут «номер версії», то в графік задач можна включати вимоги з тим самим номером версії. Якщо у вимоги є атрибут призначений «assigned-to», то в графік задач можна включати вимоги, призначені кожному розробнику.

Вимоги типу MSPT. Вимоги й задачі зв'язані між собою використанням спеціального типу вимоги – MSPT. Цей тип вимоги використовується в проекті Requisite Pro, для відображення вимог, що є задачами в графіку робіт Microsoft Project. Тип MSPT включає атрибути вимоги: ім'я файлу Microsoft Project і унікальний ідентифікатор задачі.

Тип вимоги MSPT включає також задані за замовчуванням атрибути, які формуються на основі атрибутів у задачі в Microsoft Project. Це атрибути задач за замовчуванням – Cost (Ціна), Duration (Тривалість), Finish (Дата закінчення), Milestone (Дата контрольної крапки), Priority (Пріоритет), Resource Names (Назви ресурсів) і Start (Дата початку задачі).

Можна приєднати атрибути задач Microsoft Project до атрибутів вимог типу MSPT у файлі .rqs Requisite Pro.

Вимоги по забезпеченню безпеки. При першому використанні майстра MS Project Integration Wizard, потрібно мати права адміністратора Requisite Pro, при наступному використанні – права на створення й видалення типів вимог MSPT, на оновлення атрибутів, пов'язаних з типом вимог MSPT.

Порядок виконання роботи

Запуск майстра й додавання вимоги типу MSPT. MS Project Integration Wizard можна запускати в Requisite Pro або як незалежний додаток.

Запуск MS Project Integration Wizard в Requisite Pro. Для запуску майстра MS Project Integration Wizard потрібно виконати наступні дії:

1. На панелі інструментів, виберіть пункти Tools ⇒ Microsoft Project ⇒ Integration Wizard. На екрані з'являється вікно Introduction (рис.2.1).

2. Натисніть кнопку Next. На екрані з'являється вікно Select Project Files(рис.2.2).

3. Виберіть проект. Список відображає всі проекти з вашого списку проектів Requisite Pro.

4. Виконайте одну з наступних дій:

- введіть назву й шлях до існуючого проекту в Microsoft Project;
- натисніть на кнопку Browse щоб знайти, вибрати й відкрити існуючий проект в Microsoft Project;

- натисніть на кнопку New, щоб створити новий проект в Microsoft Project (На екрані з'являється вікно Project Info. Після створення нового проекту, натисніть на кнопку ОК у вікні Project Info. З'являється вікно для збереження нового проекту. Після збереження проекту, майстер згортає Microsoft Project і додає нову назву проекту в текстове поле Microsoft Project Schedule);

- натисніть на кнопку Next (якщо вибраний проект має систему безпеки, на екрані з'явиться вікно Logon. Введіть своє ім'я й пароль для з'єднання із захищеним проектом Requisite Pro. Майстер перевіряє проект і шукає вимоги типу MSPT. Якщо майстер не знаходить тип вимог MSPT, він додає їх і відображає у вікні MSPT Info (рис.2.3)).

5. Натисніть кнопку Next. Майстер відкриває проект і відображає екран Select an Action(рис.2.4).

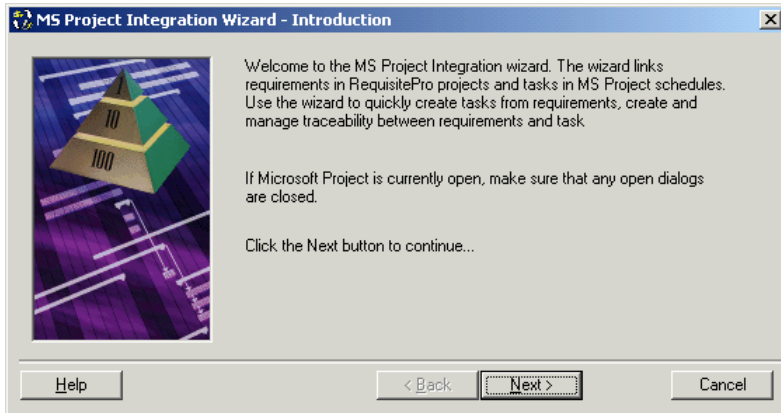


Рисунок 2.1 – Запуск майстра MS Project Integration Wizard

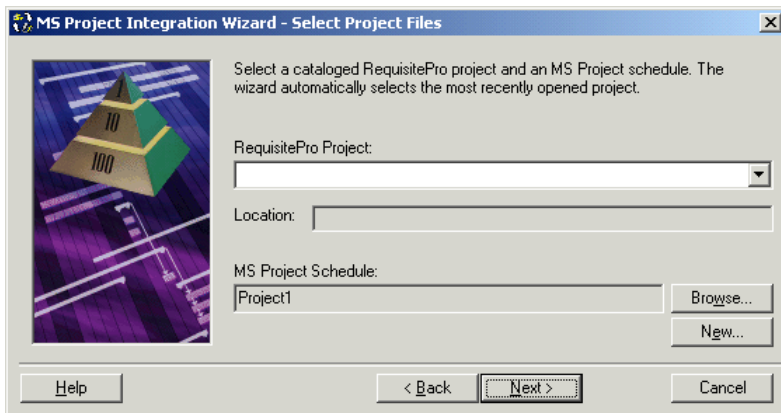


Рисунок 2.2 – Вікно вибору проекту Requisite Pro

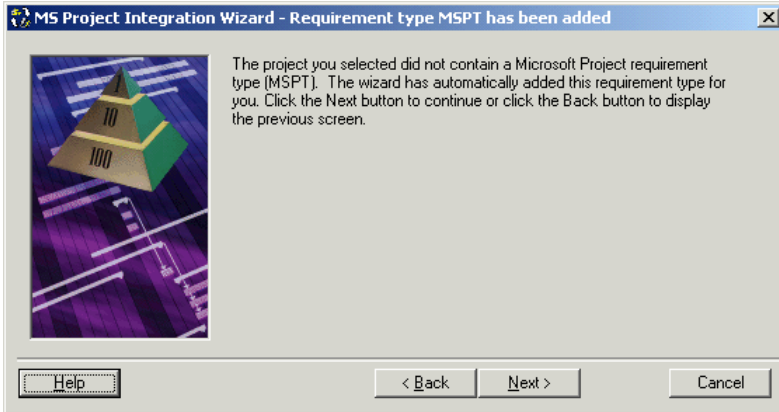


Рисунок 2.3 – Вікно додавання типу вимог MSPT

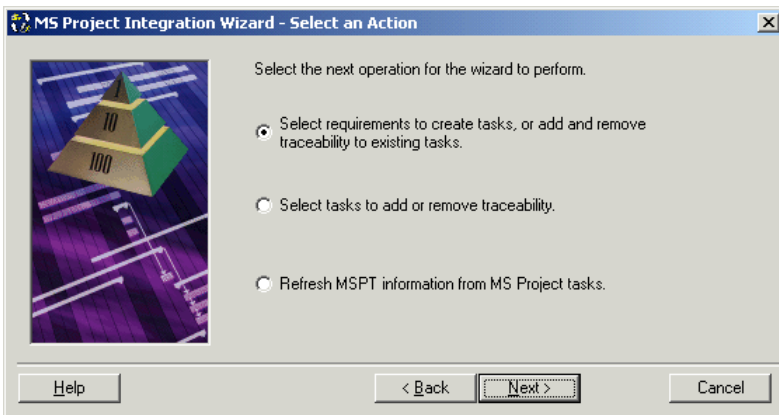


Рисунок 2.4 – Вікно вибору дії

Створення задач із вимог. Для створення задач із вимог потрібно виконати наступні дії.

1. Виберіть опцію «Select requirements to create tasks, or add and remove traceability to existing tasks».
2. Натисніть кнопку Next.
3. У списку Requirement Type виберіть тип вимог (рис.2.5):
– All requirements of selected type (Всі вимоги обраного типу);

- Requirements of this attribute value (Вимоги з таким значенням атрибута);
- у списку атрибутів виберіть атрибут, у поле Value виберіть значення й оператора;
- щоб шукати вимоги, що містять певні фрази (наприклад, слово «shall»), введіть рядок у поле Requirement Text Search. Можна використовувати частково заповнені рядки, але шаблони символів підтримуватися не будуть.

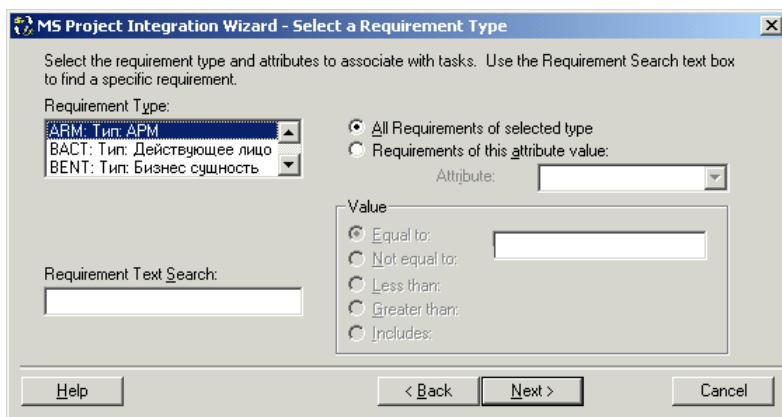


Рисунок 2.5 – Вікно вибору типу вимоги

4. Натисніть на кнопку Next. На екрані відобразитися дерево зв'язків задач, які були обрані на попередньому кроці.
5. Виберіть одну або більше вимог і ввімкніть перемикач Create Tasks.
6. Натисніть кнопку Next. На наступному екрані натисніть кнопку Finish (рис.2.7).

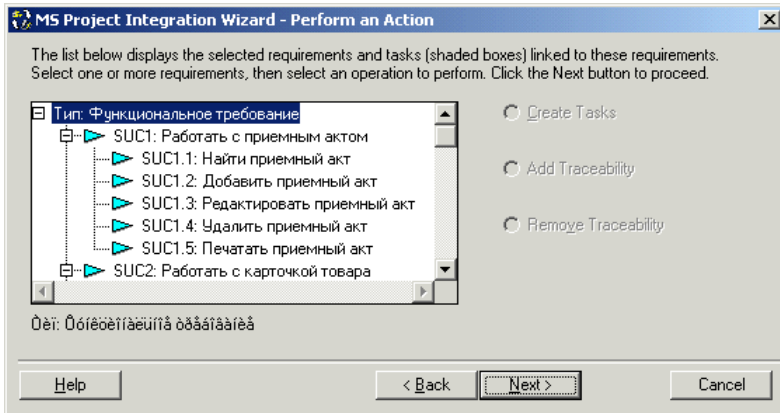


Рисунок 2.6 – Вікно дерева зв'язків задач

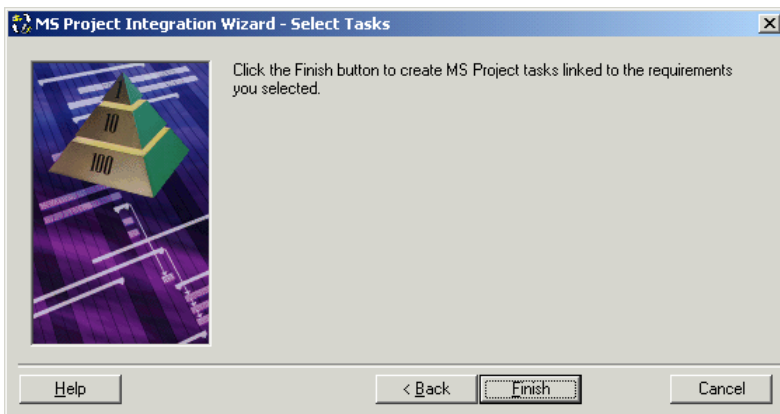


Рисунок 2.7 – Вікно завершення вибору задач

Примітка: При зміні в графіку робіт MS Project, рекомендується зберігати як базовий тільки після проведення всіх змін.

8. Натисніть кнопку Exit, щоб закрити майстер, або натисніть кнопку Another Action, щоб повернутися до вікна Select an Action, де можна створювати інші задачі, додавати або видаляти залежності й обновляти інформацію про задачі з Microsoft Project.

Результат імпорту вимог з Requisite Pro в MS Project, наприклад, може мати такий вигляд (рис.2.8).

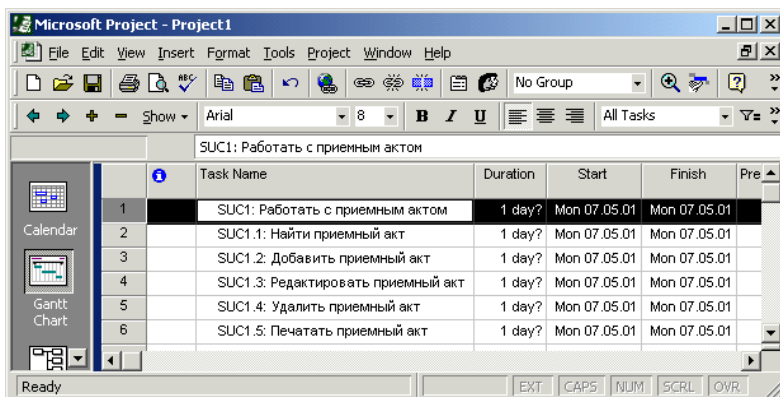


Рисунок 2.8 – Результат імпорту вимог з Requisite Pro в MS Project

Додавання або видалення залежності. Використання наступних команд припускає, що майстер запущений, вікно Select An Action відображене на екрані, і що створені задачі в Microsoft Project, які пов'язані з вимогами в Requisite Pro.

В майстрі MS Project Integration Wizard існують дві можливості для зміни залежності:

- вибрати вимогу в Requisite Pro і зв'язати її із задачею Microsoft Project;
- вибрати задачі в Microsoft Project і зв'язати їх з вимогою Requisite Pro.

Вибір вимог для зміни залежностей.

1. Виберіть опцію «Select requirements to create tasks, or add and remove trace ability to existing tasks» у вікні Select An Action.

2. Натисніть кнопку Next.

3. У списку Requirement Type виберіть тип вимоги.

4. Натисніть на кнопку Next. На екрані відобразиться дерево зв'язків задач, які були обрані на попередньому кроці.

Для видалення залежностей між обраною задачею і вимогою:

1. Виберіть одну або більшу кількість задач, далі натисніть Remove traceability і Finish. Залежність між обраними задачами й

вимогою видалитися. При видаленні залежностей задачі в графіку Microsoft Project зберігаються (рис.2.9).

2. Натисніть кнопку Exit, щоб закрити майстер, або натисніть кнопку Another Action, щоб повернутися до екрану Select an Action, де можна створювати інші задачі, додавати або видаляти залежності й обновляти інформацію про вимоги з Microsoft Project.

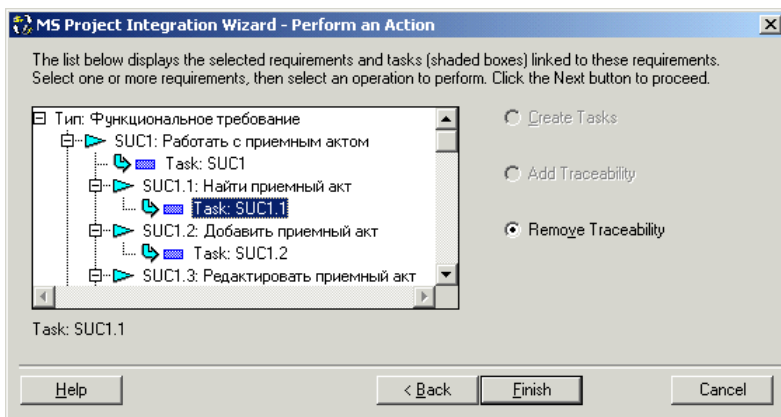


Рисунок 2.9 – Вікно дерева зв'язків задач (видалення трасування)

Для додавання залежності між обраними вимогами й задачами:

1. Виберіть вимогу й натисніть кнопку Add Traceability. Залежність створюється тільки для однієї вимоги.

2. Натисніть кнопку Next для переходу до наступного екрана й виберіть задачу.

3. Виконайте одну з наступних дій:

- якщо задоволені поточним вибором задач, натисніть Finish;
- щоб змінити вибір, натисніть MS Project для відображення графіка й виберіть одну або більшу кількість задач. Поверніться до майстра інтеграції й натисніть Refresh Task List, щоб обновити відображувані задачі;
- після цього натисніть Finish.

Вибір задач для зміни залежностей.

1. У вікні Select An Action виберіть опцію «Select requirements to create tasks, or add and remove traceability to existing tasks» і натисніть кнопку Next. На екрані з'явиться вікно Select Tasks (рис.2.10).

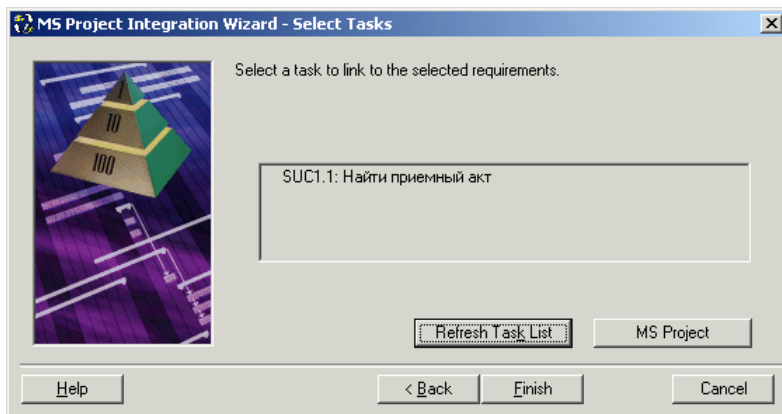


Рисунок 2.10 – Вікно вибору задачі

2. Виконайте одну з наступних дій:

– якщо задоволені поточним вибором задачі, натисніть кнопку Next;

– щоб змінити вибір клацніть по кнопці MS Project для відображення графіка проекту й виберіть одну або більшу кількість задач у проекті;

– поверніться в майстер інтеграції й клацніть по кнопці Refresh Task List для оновлення відображуваних задач. Після цього натисніть на кнопку Next. Екран відображає дерево зв'язку задач для задач, які були обрані в попередньому вікні й вимог, які пов'язані із цими задачами (рис.2.11).

3. Виконайте одну з наступних дій:

– для відображення задачі в графіку Microsoft Project виберіть задачу й клацніть по кнопці Go to Task. Можна переглядати або модифікувати задачі в проекті. Після перегляду або модифікації задачі в проекті, поверніться до майстра інтеграції;

– виберіть вимогу, а потім опцію – Remove traceability. Натисніть на кнопку Finish. Залежність між обраним задачею і вимогою видалиться;

– виберіть задачу, а потім опцію – Add Traceability (рис.2.11). Натисніть кнопку Next. З'являється вікно Select a Requirement Type.

4. У списку Requirement Type виберіть тип вимоги.

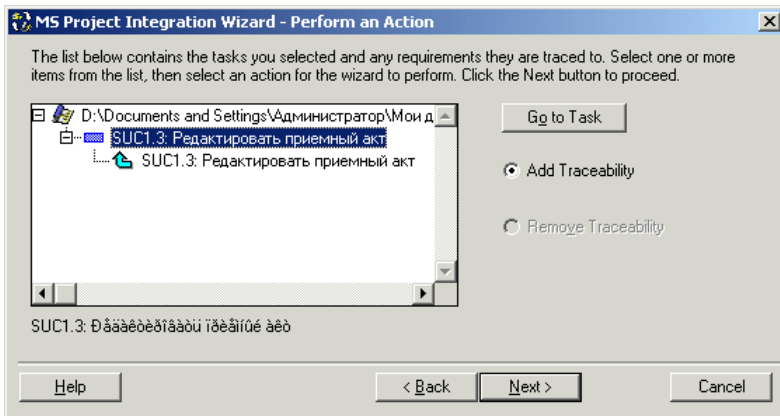


Рисунок 2.11 – Вікно дерева зв'язків задач (додавання трасування)

5. Для переходу до наступного вікна майстра натисніть на кнопку Next. На екрані відобразяться вимоги, обрані на попередньому кроці.

6. Виберіть вимогу. Щоб відображати вимоги без задач, виберіть перемикач «Show only requirements with no tasks».

7. Натисніть на кнопку Finish.

Оновлення інформації про вимоги MSPT із задач. Використання наступних команд припускають, що на екрані відображене вікно «Select An Action» в майстрі інтеграції з MS Project, і що створені задачі в Microsoft Project, пов'язані з вимогами MSPT.

Вимоги з типом MSPT представляють задачі в Requisite Pro. Повинна бути встановлена залежність між вимогами до програми й вимогами MSPT.

Коли значення в полях задач Microsoft Project змінюються, варто використовувати можливість оновлення функцій у майстрі інтеграції з

MS Project для оновлення значень відповідних атрибутів у вимогах MSPT. У результаті зміни значень атрибутів MSPT залежності між вимогами до програми й вимогами MSPT відзначаються як підозрілі. Ці підозрілі залежності попереджають про зміни в графіку проекту Microsoft Project.

Можна переглянути підозрілі зв'язки в матриці зв'язків між вимогами до програми й вимоги MSPT або можна переглянути зміни значень атрибутів MSPT, використовуючи матрицю атрибутів вимог для вимог з типом MSPT.

Для оновлення вимог MSPT із задач icrosoft Project:

1. У вікні Select an Action виберіть опцію «Refresh MSPT information from MS Project tasks» і натисніть на кнопку Next.

2. На екрані відображається вікно Update зі списком змін, які відбулися після того, як Requisite Pro був модифікований задачами Microsoft Project (рис.2.12). Для продовження натисніть на кнопку Finish.

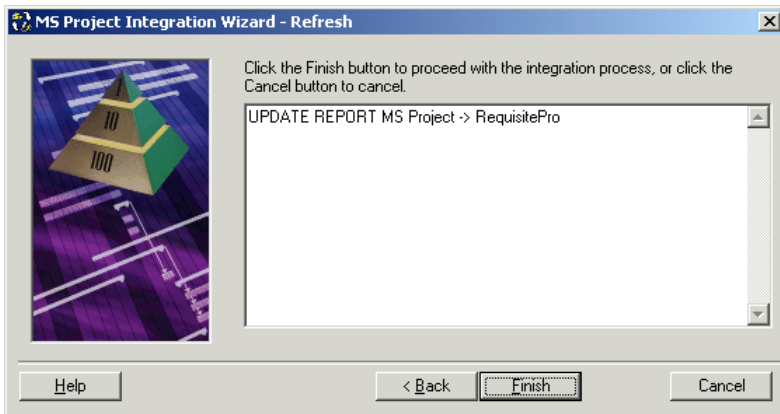


Рисунок 2.12 – Вікно Update зі списком змін

Робота з вимогами типу MSPT. Переглядати вимоги типу MSPT можливо у вікнах переглядів Requisite Pro.

1. В *Області Подань* виберіть пункти меню File ⇒ New View або натисніть на іконку New View.

2. Виберіть один з наступних варіантів переглядів:

- виберіть *Матрицю Атрибутів* і виберіть тип вимог MSPT для перегляду змін у значеннях атрибутів вимог MSPT (рис.2.13);
- виберіть *Матрицю Трасування*. Виберіть типи вимоги в рядках – вимоги до програми, у стовпцях вимоги MSPT. У матриці відображаються зв'язки між вимогами (рис.2.14);
- виберіть *Дерево Трасування*. Виберіть тип вимоги до програми й розгорніть відображення для перегляду зв'язків з вимогами типу MSPT(рис.2.15).

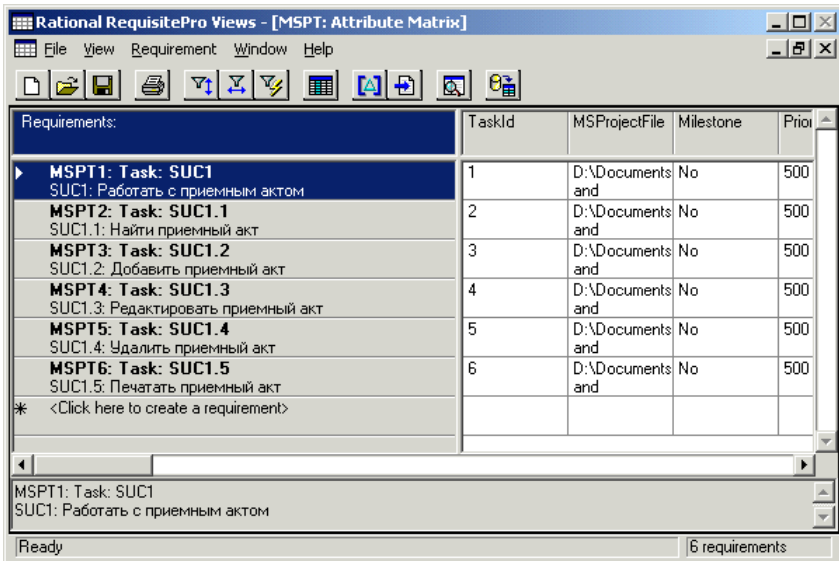


Рисунок 2.13 – *Матриця Атрибутів* зі списком змін

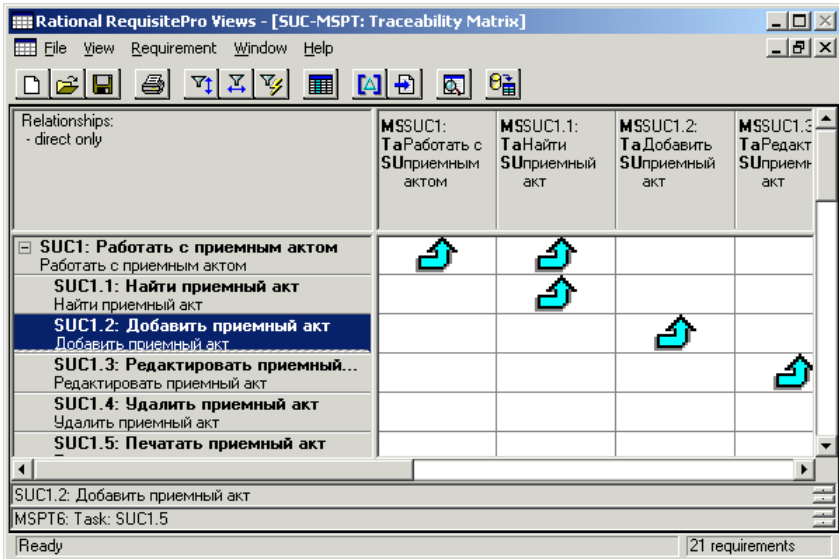


Рисунок 2.14 – Матриця Трасування зі списком змін

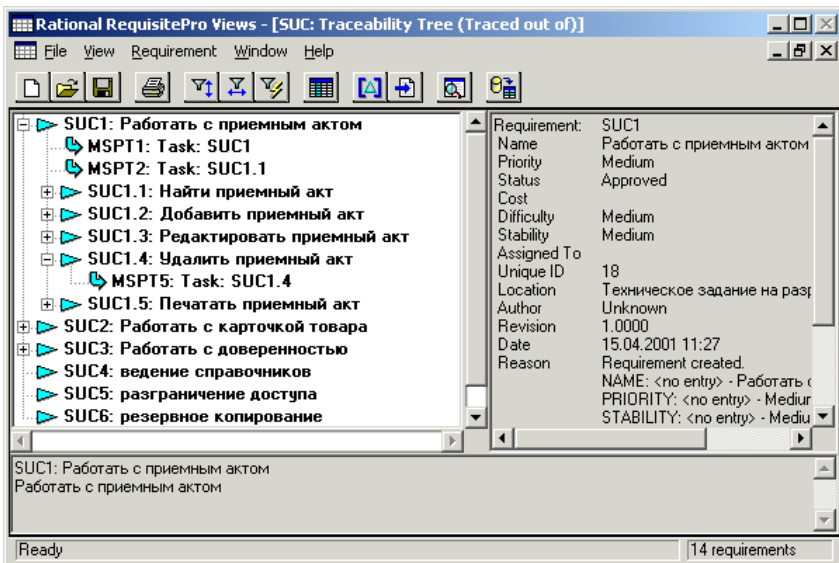


Рисунок 2.15 – Дерево Трасування зі списком змін

Додавання полів атрибутів до вимог типу MSPT. На відміну від інших типів вимог, не можна додати новий атрибут до вимог з типом MSPT з використанням закладки Attributes у вікні Project Properties. Створювати атрибути для вимог типу MSPT можна у файлі .rqs проекту Requisite Pro. Наприклад, якщо назва проекту – «ТТ», можна використовувати файл tt.rqs для створення нових атрибутів для вимог з типом MSPT.

Якщо встановлено систему безпеки проекту, то в особи, що додає поля до атрибутів вимог з типом MSPT, повинні бути відповідні повноваження.

Ідентифікатори полів Microsoft Project. Використовуйте наступні константи як ідентифікатори полів, щоб відображати атрибути для вимог типу MSPT Requisite Pro. Використовуючи текстовий редактор, додайте рядок в розділ [MSProjectMappings] у файл <ім'я проекту>.rqs.

Використовуйте наступний формат:

ідентифікатор поля=найменування атрибута,

де <ідентифікатор поля> – числова константа з перерахованих нижче, а <найменування атрибута> – це мітка, яку потрібно привласнити атрибуту в Requisite Pro. Ідентифікатори полів, показані жирним шрифтом, уже забезпечуються як задані за замовчуванням атрибути для вимог типу MSPT.

Для додавання полів до атрибутів:

1. У текстовому редакторі відкрийте файл .rqs проекту, з яким працюєте.

2. У розділі [MSProjectMappings] файлу .rqs введіть наступне:

ідентифікатор поля=найменування атрибута,

3. Збережіть й закрийте файл проекту.

4. На панелі інструментів Requisite Pro виберіть пункти Project
⇒ Open.

5. Виберіть проект, ввімкніть прапор Exclusive і натисніть кнопку ОК. Введіть ім'я та пароль, якщо буде потрібно.

6. На панелі інструментів виберіть пункти меню Tools
⇒ Microsoft Project (Integration Wizard).

7. Натисніть кнопку Next. Переконайтесь, що поточний проект Requisite Pro і пов'язаний з ним проект Microsoft Project обрані. Натисніть кнопку Next.

8. Виберіть опцію «Refresh MSPT information from MS Project tasks». Натисніть на кнопку Next.

9. Після того, як обробка закінчена, натисніть на кнопку Finish. Натисніть кнопки Exit і Yes для виходу й підтвердження.

Видалення атрибутів у вимог типу MSPT. У особи, що видаляє атрибути у вимог типу MSPT, повинні бути відповідні повноваження. Для видалення атрибутів потрібно виконати наступні дії:

1. Відкрийте файл проекту Requisite Pro (tt.rqs) з використанням текстового редактора.

2. У файлі проекту в розділі [MSProjectMappings] знайдіть атрибут вимоги для видалення.

3. Видаліть весь рядок, що містить атрибут. Збережіть й закрийте файл проекту.

4. На панелі інструментів Requisite Pro виберіть пункти Project ⇒ Open.

5. Виберіть проект, ввімкніть прапор Exclusive і натисніть кнопку ОК. Введіть ім'я й пароль, якщо буде потрібно.

6. На панелі інструментів виберіть пункти меню Tools ⇒ Microsoft Project ⇒ Integration Wizard.

7. Натисніть кнопку Next. Переконайтесь, що поточний проект Requisite Pro і пов'язаний з ним проект Microsoft Project обрані. Натисніть кнопку Next.

8. Виберіть опцію «Refresh MSPT information from MS Project tasks». Натисніть на кнопку Next.

9. Після того, як обробка закінчена, натисніть на кнопку Finish. Натисніть кнопки Exit і Yes для виходу й підтвердження.

Імпорт документів. Деякі документи можуть бути створені до створення проекту RequisitePro. У такому випадку необхідно імпортувати їх в інструмент. Виконаємо імпорт документа RMP. Для цього виконайте наступні дії:

1. Виберіть файл у *Провіднику*.

2. Виберіть File⇒Import (Файл⇒Імпорт). З'явиться Import Wizard (Майстер налаштування імпорту), як показано на рис.2.15.

3. Виберіть Microsoft Word Document.

4. Виберіть шлях до файлу, який необхідно імпортувати.

5. Натисніть Next.

6. Виберіть «Document only», тому що документ RMP не містить ніяких вимог (рис.2.16).

7. Натисніть Next. З'явиться діалогове вікно Document Properties (Властивості Документа), як показано на рис.2.17.

8. Введіть Name (Назву) і Description (Опис) документа, а також виберіть Document Type (Тип документа).

9. Якщо необхідно, можна змінити встановлені за замовчуванням значення Filename (Назви файлу) і Directory (Каталогу).

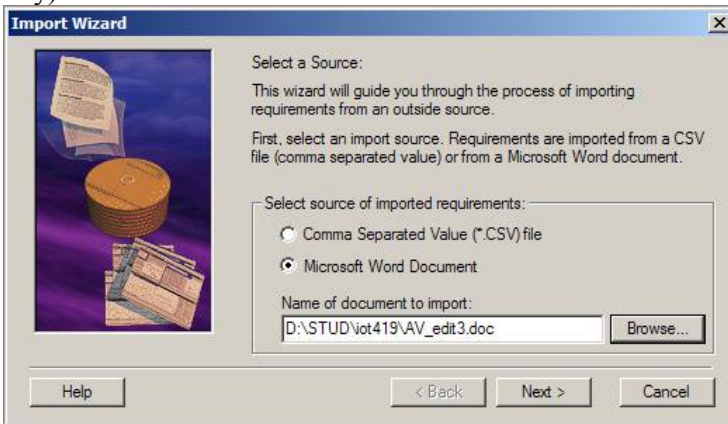


Рисунок 2.15 – Import Wizard (Майстер налаштування імпорту)

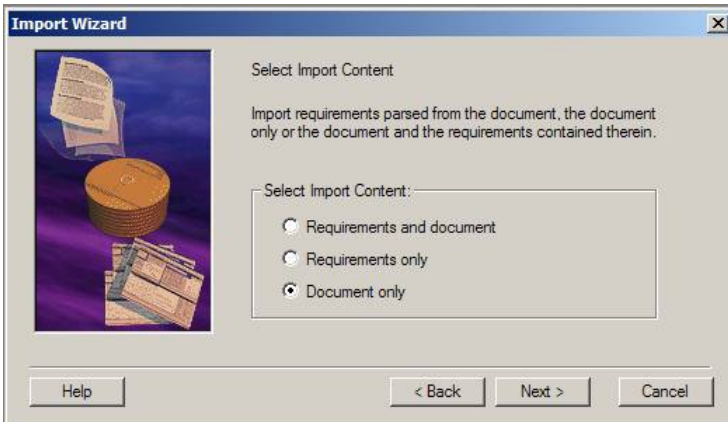


Рисунок 2.16 – Import Wizard (Майстер налаштування імпорту):
Вибір імпортованого документа

10. Натисніть Так (Yes) у діалоговому вікні, що попереджає про оновлення початкового формату документа.

11. Натисніть Commit у діалоговому вікні, підтверджуючи створення файлу.

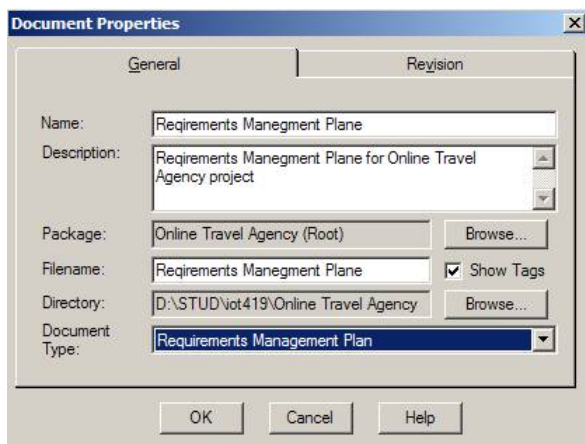


Рисунок 2.17 – Діалогове вікно Document Properties (Властивості Документа)

Експорт подань. Можна експортувати подання в документ Microsoft Word або в CSV-Файл (який можна відкрити за допомогою Microsoft Excel). Це дозволяє використовувати дану інформацію поза середовищем Requisite Pro.

Для експорту *Матриці Атрибутів* виконайте наступні дії:

1. Виберіть File⇒Export⇒Export to CSV (Файл⇒Експорт⇒Експорт в CSV).

2. Введіть назву файлу.

3. Натисніть кнопку Save (Зберегти).

Подання зберігається в CSV-Файлі, який можна відкрити з використанням Microsoft Excel, як показано на рис.2.18.

	A	B	C	D	E
23	STRQ22	System should provide the opportunity to book the flight, purchase a ticket, reserve hotel, reserve car, and provide info about attractions.	Main functionality	High	Travel Agency Owner
24	STRQ23	User should be able to purchase a ticket online, without the necessity of calling the travel agent.	Online purchasing	High	Travel Agency Owner
25	STRQ24	System should follow implementation guidelines set up in the chain of our travel agencies.	Implementation guidelines	Medium	Travel Agency Owner
26	STRQ25	Hotel Providers, Car Providers, and Airline Representatives should have their ids and passwords to the pages where they can submit their offers.	Usersids and passwords for providers	High	Travel Agency Owner
27	STRQ26	System must be available 24 hours a day with a reliability appropriate to commercial aspirations.	Reliability	Medium	Travel Agency Owner
28	STRQ27	System should be developed in 3 months.	Development time	Medium	Travel Agency Owner
29	STRQ28	Users shall pick ids and passwords while buying airline ticket.	Usersids and passwords for users	High	Travel Agency Owner
30	STRQ29	The search facility should allow customer service representative to find a reservation based on: Last Name, Date, Destination City	Reservation search	Medium	Customer Service Rep
31	STRQ30	The customer service representative should be able to change reservation details or cancel a reservation.	Changing and cancelling reservation	High	Customer Service Rep
32	STRQ31	While submitting the content information, the Content Manager should be able to submit plain text without HTML tags.	Submitting plain text	Low	Content Manager
33	STRQ32	Content information should be stored in a text file.	Storing content information	Medium	Content Manager
34	STRQ33	System shall be fully tested under specific versions of most popular browsers only.	Browser selection	Medium	Developer

Рисунок 2.18 – Матриця Атрибутів, експортована в Excel

Завдання на самостійну роботу

1. Використовуючи рекомендовану літературу та дані методичні вказівки, ознайомитися з основними принципами створення задач в MS Project з вимог.

2.

3. Відповісти на контрольні запитання.

Контрольні запитання

1. Визначення типу вимоги MSPT.
2. Яким чином створюються задачі з вимог?
3. Яким чином додається та видаляється залежність?
4. Яким чином відбувається оновлення інформації про вимоги MSPT із задач?
5. Як можна переглянути вимоги типу MSPT?
6. Яким чином виконуються операції додавання та видалення атрибутів у вимог типу MSPT?
7. Яким чином здійснюється імпорт документів з IBM Rational Requisite Pro?

ЛІТЕРАТУРА

1. Мацяшек Л.А. Анализ требований и проектирование систем. Разработка информационных систем с использованием UML. – М.:Изд. дом «Вильямс», 2002. – 432 с.
2. Петр Zielczynski Управление требованиями с помощью IBM Rational RequisitePro. – IBM Press, 2007. – 360 с.

Додаток А

Перелік тем для самостійних робіт

1. Розробка соціальної мережі.
2. Розробка відео-блогу
3. Розробка Wiki-системи
4. Розробка інтернет магазину
5. Розробка твіттера
6. Розробка системи для он-лайн трансляції телепередач
7. Розробка системи для дистанційної освіти
8. Розробка програмного забезпечення для голосового зв'язку (аналогу Skype).
9. Розробка безкоштовної системи миттєвого обміну повідомленнями (аналог Qip)
10. Розробка Dropbox.
11. Розробка поштового клієнта.
12. Розробка безкоштовного онлайн-офісу (аналог Google Docs)
13. Розробка довідково-пошукової системи
14. Розробка АРМ диспетчера таксі
15. Розробка АРМ бібліотекаря
16. Розробка системи деканат
17. Розробка системи бухгалтерського обліку
18. Розробка системи підтримки складського господарства
19. Розробка системи інтернет-банкінгу
20. Розробка системи онлайн-керування розробки ПЗ
21. Розробка аналога графічного редактора в Internet
22. Розробка системи розрахунку страхування для фізичних та юридичних осіб
23. Розробка web-орієнтованої системи підтримки групових рішень
24. Розробка експертної системи вибору методів прийняття рішень
25. Розробка системи оцінки якості програмного забезпечення
26. Оцінка надійності газотурбинних установок наземного використання.