

ВИКОРИСТАННЯ СИСТЕМ УПРАВЛІННЯ ВИМОГАМИ У ПРОЦЕСІ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Vinnitsia National Technical University

Анотація

У науковій тезі розглянуто особливості використання систем управління вимогами у процесі розробки програмного забезпечення. Досліджено значення вимог як основи успішного створення програмного продукту, а також проаналізовано роль спеціалізованих систем у зборі, документуванні, відстеженні та зміні вимог. Охарактеризовано основні переваги застосування таких систем у командній роботі, зокрема підвищення узгодженості дій учасників проєкту, покращення контролю змін і забезпечення простежуваності вимог. Визначено основні етапи використання систем управління вимогами та їхній вплив на якість кінцевого програмного продукту. Результати дослідження підтверджують, що системи управління вимогами є важливим інструментом підвищення ефективності процесу розробки програмного забезпечення.

Ключові слова: вимоги до програмного забезпечення, управління вимогами, розробка ПЗ, системи управління вимогами, специфікація вимог, простежуваність вимог, зміни вимог, командна робота.

Abstract

This scientific paper examines the use of requirements management systems in the software development process. The importance of requirements as the basis for successful software product creation is investigated, and the role of specialized systems in collecting, documenting, tracking, and modifying requirements is analyzed. The main advantages of using such systems in teamwork are characterized, including improved coordination among project participants, better change control, and ensured requirements traceability. The main stages of using requirements management systems and their influence on the quality of the final software product are identified. The results of the study confirm that requirements management systems are an important tool for increasing the efficiency of the software development process.

Keywords: software requirements, requirements management, software development, requirements management systems, software requirements specification, requirements traceability, requirements changes, teamwork.

Вступ

У сучасних умовах розробка програмного забезпечення є складним і багатоетапним процесом, у якому особливе значення має правильне визначення вимог до майбутнього продукту. Саме вимоги формують основу для проєктування, реалізації, тестування та подальшого супроводу програмної системи. Якщо вимоги сформульовані нечітко, неповно або суперечливо, це може призвести до помилок у розробці, перевищення термінів виконання проєкту, збільшення витрат і невідповідності кінцевого продукту очікуванням замовника.

У зв'язку з цим важливого значення набуває управління вимогами як процес організації, аналізу, документування, погодження та контролю змін вимог протягом усього життєвого циклу програмного забезпечення. Особливо актуальним це є для сучасних проєктів, у яких над одним продуктом працюють декілька учасників команди, а самі вимоги можуть змінюватися залежно від потреб бізнесу, ринку або користувачів.

Для підвищення ефективності цього процесу використовуються спеціалізовані системи управління вимогами. Вони дозволяють централізовано зберігати вимоги, контролювати їх зміни, підтримувати зв'язки між окремими елементами документації та забезпечувати зручну взаємодію між усіма учасниками проєкту [1]. Саме тому дослідження використання систем управління вимогами у процесі розробки ПЗ є актуальним і важливим.

Результати дослідження

Насамперед доцільно визначити сутність поняття вимог до програмного забезпечення. Вимоги – це сукупність умов, характеристик, функцій і обмежень, яким має відповідати програмний продукт для задоволення потреб замовника, користувачів та інших зацікавлених сторін. Вони можуть бути функціональними, тобто описувати, які саме дії повинна виконувати система, і нефункціональними, тобто визначати характеристики якості, продуктивності, безпеки, зручності використання та інші параметри [2]. Саме вимоги є базою для створення програмного продукту, тому якість їх опрацювання безпосередньо впливає на успішність усього проекту [3].

Управління вимогами є безперервним процесом, що охоплює збір, аналіз, документування, перевірку, узгодження та відстеження змін вимог. У традиційних підходах ці дії часто виконувалися вручну або за допомогою окремих текстових документів і таблиць. Проте з розвитком складних програмних систем та переходом до командної і гнучкої розробки такий підхід став недостатньо ефективним [4]. Велика кількість вимог, необхідність частого оновлення інформації, участь кількох виконавців та потреба у збереженні історії змін обумовили потребу у використанні спеціалізованих систем управління вимогами.

Системи управління вимогами – це програмні засоби, призначені для централізованого опрацювання вимог на всіх етапах життєвого циклу ПЗ. Вони дають змогу фіксувати вимоги у структурованому вигляді, групувати їх за категоріями, визначати пріоритети, встановлювати зв'язки між вимогами, завданнями, тестами та іншими артефактами проекту. Крім того, такі системи забезпечують контроль версій, журнал змін, спільний доступ до документації та зручну взаємодію між аналітиками, розробниками, тестувальниками, менеджерами та замовниками.

Роль систем управління вимогами у процесі розробки програмного забезпечення є надзвичайно важливою. Передусім вони сприяють підвищенню впорядкованості роботи з вимогами. Замість розрізнених документів, листування та неструктурованих нотаток команда отримує єдине інформаційне середовище, у якому зберігаються всі актуальні вимоги [5]. Це зменшує ризик втрати даних, дублювання інформації або використання застарілих версій документів.

Важливим аспектом є і підвищення якості кінцевого програмного продукту. Якщо вимоги чітко задокументовані, погоджені й доступні всім учасникам розробки, зменшується ймовірність неправильного розуміння завдань, пропуску важливих функцій або появи невідповідностей між очікуваннями замовника та реалізованим рішенням. Таким чином, системи управління вимогами сприяють не лише організації роботи, а й безпосередньо впливають на результативність усього процесу розробки.

Процес використання систем управління вимогами можна умовно поділити на кілька етапів. Першим етапом є збір і фіксація вимог. На цьому етапі інформація надходить від замовника, користувачів, аналітиків та інших зацікавлених сторін. Отримані вимоги вносяться до системи у структурованому вигляді, де можуть бути класифіковані за типами, пріоритетами або модулями системи [5].

Другим етапом є аналіз і уточнення вимог. На цьому кроці перевіряється повнота, однозначність, несуперечливість і реалізовуваність сформульованих вимог. У системі можуть додаватися коментарі, уточнення та зв'язки з іншими елементами документації. Такий підхід дозволяє ще до початку реалізації виявити слабкі місця у вимогах і своєчасно усунути неточності [6].

Третім етапом є погодження та затвердження вимог. Після аналізу вимоги мають бути узгоджені між усіма зацікавленими сторонами. У системах управління вимогами це часто супроводжується зміною статусу вимоги, фіксацією рішень та збереженням історії узгодження [1]. Це дозволяє уникнути ситуацій, коли одна й та сама вимога по-різному трактувалася різними учасниками проекту [4].

Наступний етап – відстеження реалізації вимог. Після затвердження вимоги можуть бути пов'язані із завданнями розробки, елементами дизайну, модулями системи або тест-кейсами [6]. Це забезпечує контроль за тим, чи справді вимога була врахована в процесі розробки та перевірена під час тестування. У разі виявлення дефектів або невідповідностей система дозволяє швидко визначити, до якої саме вимоги вони належать.

Останнім важливим етапом є керування змінами та актуалізація вимог. Упродовж життєвого циклу продукту вимоги можуть доповнюватися або змінюватися. Система управління вимогами дозволяє вносити зміни контрольовано, зберігати попередні версії, аналізувати наслідки оновлень і підтримувати

документацію у актуальному стані. Саме це забезпечує гнучкість процесу розробки та дає змогу адаптувати продукт до нових умов без втрати керованості.

Серед прикладів таких систем можна виділити Jira, Confluence, IBM Engineering Requirements Management DOORS, Trello, Azure DevOps та інші інструменти, які можуть використовуватися як окремо, так і в поєднанні між собою [1; 4]. Вибір конкретної системи залежить від масштабу проєкту, кількості учасників команди, складності вимог і прийнятої методології розробки. Проте незалежно від конкретного програмного засобу, головною метою його використання залишається впорядкування роботи з вимогами та підвищення ефективності розробки [5].

Отже, системи управління вимогами є важливою складовою сучасного процесу розробки програмного забезпечення. Вони забезпечують централізоване зберігання інформації, підтримують узгодженість дій учасників команди, спрощують контроль змін та дозволяють підтримувати простежуваність вимог на всіх етапах життєвого циклу ПЗ [1; 6]. Саме завдяки цим властивостям вони сприяють зниженню ризиків, покращенню якості продукту та підвищенню ефективності командної роботи [4].

Висновок

Отже, у результаті дослідження було встановлено, що використання систем управління вимогами є важливим чинником успішної розробки програмного забезпечення. Вимоги становлять основу будь-якого програмного проєкту, а їх правильне збирання, документування, узгодження та супровід визначають якість кінцевого продукту. Саме тому застосування спеціалізованих систем дозволяє зробити цей процес більш упорядкованим, прозорим і керованим.

Дослідження показало, що системи управління вимогами забезпечують централізоване зберігання вимог, підтримують простежуваність зв'язків між різними артефактами проєкту, покращують взаємодію між учасниками команди та спрощують контроль змін.

Таким чином, системи управління вимогами слід розглядати як один із ключових інструментів сучасної інженерії програмного забезпечення. Їх використання сприяє підвищенню якості розробки, оптимізації командної роботи та створенню більш надійних, гнучких і конкурентоспроможних програмних продуктів.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Ліщинська Л. Б. Використання CASE-засобів для керування вимогами до програмних систем // Тези Всеукраїнської науково-практичної on-line конференції здобувачів вищої освіти і молодих учених «Комп'ютерні технології: інновації, проблеми, рішення». Житомир, 2021.
2. Козаченко М. Р., Ліщинська Л. Б. Аналіз вимог до інтелектуального програмного застосунку для адаптивного керування параметрами екрана // Матеріали Всеукраїнської науково-практичної інтернет-конференції «Молодь в науці: дослідження, проблеми, перспективи (МН-2025)», Вінниця, 15 жовтня 2024 р. – 15 червня 2025 р. Вінниця, 2025. С. 396–397.
3. Sommerville I. Software Engineering. 10th ed. Boston : Pearson, 2016.
4. Pressman R. S., Maxim B. R. Software Engineering: A Practitioner's Approach. 9th ed. New York : McGraw-Hill Education, 2019.
5. A Guide to the Business Analysis Body of Knowledge (BABOK Guide). Version 3. Toronto : International Institute of Business Analysis, 2015.
6. Wiegers K., Beatty J. Software Requirements. 3rd ed. Redmond, WA : Microsoft Press, 2013..

Бабенко Софія Павлівна – студентка групи 5ПІ-22б, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, м. Вінниця, e-mail: sofia.kudran@gmail.com.

Ліщинська Людмила Броніславівна – д-р техн. наук, професор, професор кафедри програмного забезпечення, Вінницький національний технічний університет, м. Вінниця, e-mail: llb@vntu.edu.ua

Babenko Sofia Pavlovna – student of the 5PI-22b group, faculty of Information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: sofia.kudran@gmail.com.

Lishchynska Lyudmyla Bronislavivna – Dr. Sc. (Eng.), Full Professor, Professor of Program Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: llb@vntu.edu.ua