

ПРОГРАМНІ ЗАСОБИ ДЛЯ УПРАВЛІННЯ ВИМОГАМИ

Рябокін Ю.М., кандидат технічних наук, доцент, Національний авіаційний університет, Київ, Україна

SOFTWARE TOOLS FOR REQUIREMENTS MANAGEMENT

Ryabokin Y.M., Ph.D., associate professor, National Aviation University, Kyiv, Ukraine

ПРОГРАММНЫЕ СРЕДСТВА ДЛЯ УПРАВЛЕНИЯ ТРЕБОВАНИЯМИ

Рябокін Ю.Н., кандидат технических наук, доцент, Национальный авиационный университет, Киев, Украина

Вступ. На сьогоднішній день розроблена велика кількість технологій ведення проектів по створенню програмного забезпечення (ПЗ). Незалежно від підходу, який використовується, на початковому етапі життєвого циклу (ЖЦ) будь-якого проекту, проводиться аналіз проблемної області, формування і узгодження вимог до майбутньої системи. Саме на цьому етапі мають бути зафіксовано реальні потреби користувачів, що стосуються функціональних, операційних та сервісних можливостей, які береться реалізувати розробник. Правильно сформовані вимоги є гарантією того, що система буде задовольняти вимогам зацікавлених у її розробці осіб [1].

Ціна помилок та нечітких неоднозначних формулювань на цьому етапі дуже висока, бо час і засоби витрачаються на непотрібну замовникові програму. Внесення необхідних коректив при цьому може вимагати значних переробок, а інколи й повного перепроєктування і, відповідно, перепрограмування. Статистика показує [1-2], що відсоток помилок у постановці завдань значно перевищує відсоток помилок кодування. А усунення помилки у вимогах на стадії супроводу готового ПЗ обходиться в 200 разів дорожче, ніж на стадії специфікації вимог.

Останніми роками спостерігається підвищений інтерес до питань документування та управління вимог до ПЗ. Необхідність цього обумовлена зростанням вимог до якості ПЗ, змінами в методології його проектування і розробки, в сучасній організації проектних робіт.

Постановка задачі дослідження. Робота з вимогами є найбільш відповідальним і найменш визначеним етапом при розробці ПЗ. Найбільш відповідальним – тому що успіх всього проекту залежить від того, наскільки повно і коректно виявлені справжні потреби замовника. Найменш визначеним – тому що дізнатися ці потреби можна тільки від працівників замовника, які не завжди добре розуміють тонкощі розробки ПЗ [1-2].

Але для успіху проекту недостатньо просто дізнатися потреби замовника. Їх необхідно перетворити в добре структурований перелік чітких і однозначних вимог, достатній для проектування архітектури системи, для розробки і тестування програмних модулів. Визначення вимог, формулювання вимог, і управління вимогами – наріжні камені успіху будь-якого ІТ-проекту.

За даними досліджень, проведених IBM в області ІТ, 60% витрат часу організації-розробники ПЗ несуть в результаті неефективного підходу до управління вимогами [1-3]. В організаціях, які не мають достатніх можливостей бізнес-аналізу, проекти в три рази частіше закінчуються невдачею, ніж успіхом. Типовими причинами зриву термінів і бюджетів проектів є неспроможність повністю виявити, чітко сформулювати вимоги замовників та відстежити зміни вимог. В результаті помилки у вимогах, які виявляються на пізніх етапах проекту посягають на 30-40% загальної вартості бюджету проекту.

При правильному визначенні вимог і управлінні ними перевитрати по проекту можна знизити на 20% завдяки скороченню числа неточних, неповних та втрачених вимог. А тому задача організації ефективного процесу управління вимогами є актуальною.

Мета дослідження. Виявлення проблем пов'язаних із процесом управління вимог та визначення рішень для його ефективної організації.

Виклад основного матеріалу дослідження. Вимога – це будь-яка умова, якій повинна відповідати система або програмний засіб, що розробляється [3]. Такою вимогою може бути і

можливість, якою система повинна володіти, і обмеження на її продуктивність або призначений для користувача інтерфейс, і нормативні вимоги, яким вона повинна задовольняти.

Вимоги до ПЗ оформлюються у вигляді окремого документа. Кожна з вимог вищого рівня породжує одну або декілька детальних вимог, в якій заявлена функціональність деталізується до рівня класів, властивостей і методів. Між вимогами вищого рівня та детальними вимогами організуються зв'язки. Цей підхід забезпечує послідовне накопичення і уточнення вимог на кожному етапі проекту, аналіз, формалізацію і подальший їх контроль при розробці ПЗ, а також ефективне управління змінами вимог впродовж всього ЖЦ проекту.

Виявлені вимоги повинні характеризуватися [3]:

- повнотою – вимога повинна описувати всю функціональність, яку необхідно реалізувати;
- коректністю – вимога повинна точно описувати бажану функціональність;
- виконуваністю – вимога повинна бути реалізована за даних умов, обмежень системи та операційного середовища;
- необхідністю – вимога повинна відображати можливість, яка дійсно необхідна користувачам, або яка потрібна для відповідності зовнішнім системним вимогам, або стандартам;
- пріоритетністю – вимоги повинні мати пріоритети для визначення послідовності їх реалізації;
- однозначністю – вимоги повинні трактуватися однаково всіма користувачами;
- можливістю бути перевіреними – необхідно розробити декілька тестів або застосувати інші прийоми для перевірки, щоб встановити, чи дійсно в програмній системі реалізована кожна вимога. Неповні, неузгоджені, нездійсненні або двозначні вимоги не перевіряються.

Аналітик вимог здійснює виявлення та збір вимог з різних джерел, систематизацію вимог, аналіз вимог для контролю їх якості, документування вимог, узгодження вимог із зацікавленими особами, обробку запитів на зміну вимог до системи.

При виявленні, документуванні, перевірці вимог здійснюється постійний взаємозв'язок аналітика вимог із зацікавленими особами проекту: представниками замовника та розробника (рис. 1).

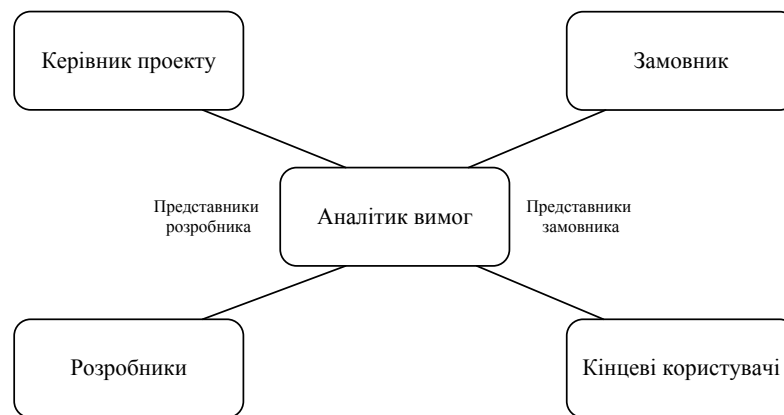


Рис. 1. Комунікація аналітика вимог

Збір усіх вимог, які пред'являються до розроблюваного ПЗ і управління ними – це дуже складне завдання. Управління вимогами – необхідна умова успішної розробки ПЗ.

Управління вимогами – контроль за виконанням вимог і планування використання ресурсів (людських, програмних, технічних, періодичних, вартісних) в процесі розробки проміжних робочих продуктів на етапах ЖЦ [3].

Під управлінням вимогами розуміють усі дії, що підтримують цілісність, точність і своєчасність оновлення угоди про вимоги в ході проекту [1, 3].

Головні причини складності управління вимог [3]:

- велика кількість потенційно «зацікавлених осіб», характерна для проектів розробки ПЗ, вимоги яких потрібно виявити і зафіксувати;

- різноманітність типів вимог, кожен з яких вимагає специфічного опису своїх атрибутів і деталізації;
- необхідність створення і підтримки складної ієрархічної структури;
- необхідність трасування вимог, тобто виявляти і фіксувати взаємозв'язок між вимогами різних типів;

– вимоги змінюються в ході виконання проекту.

З метою уникнення проблемних ситуацій під час розробки ПЗ необхідно в процесі управління вимогами дотримуватися виконання наступних дій [2]:

- 1) визначення основної версії вимог (моментальний зріз вимог для конкретної версії системи);
- 2) перегляд пропонуваніх змін до вимог і оцінка їх доцільності до прийняття;
- 3) включення схвалених змін до вимог в проект встановленим способом;
- 4) узгодження плану проекту з вимогами;
- 5) обговорення нових зобов'язань, основаних на оцінці впливу зміни до вимог;
- 6) відстеження окремих вимог до їх дизайну, вихідного коду і варіантів тестування;
- 7) відстеження статусу вимог та дії пов'язані з їх змінами впродовж всього проекту.

Серед основних процесів управління вимогами виділяють [1, 3]:

- розробку плану управління вимогами;
- розробку концепції побудови системи;
- побудову і деталізацію моделі сценаріїв використання системи, що розробляється.

Особливу увагу при управлінні вимогами слід приділяти інструментам і прийомам для управління версіями вимог; особливостям процедури формування базової версії вимог; статусам вимог, які будуть використовуватися, і особам, які мають право змінювати їх; процедурі контролю за статусом вимог; способам, за допомогою яких нові вимоги і зміни існуючих вимог пропонуються, обробляються, обговорюються і передаються усім зацікавленим особам; методам аналізу запропонованих змін; тому, яким чином на планах і зобов'язаннях проекту відіб'ється зміна вимог.

Виявлені, сформульовані та задокументовані вимоги можна зберігати в звичайних текстових файлах або електронних таблицях. Але у такого способу є ряд недоліків [3-4]:

– в текстових файлах незручно зберігати і використовувати в якості фільтра атрибути вимог (тип, статус, пріоритет);

- складно групувати вимоги;
- складно відстежувати взаємозв'язки вимог;
- складно відстежувати зміни вимог;
- складно організувати спільну роботу над вимогами кількох учасників проекту.

Використання спеціалізованих засобів для управління вимогами дозволяє уникнути цих труднощів. Багато засобів автоматично зберігають всі зміни вимог в базі даних, дозволяють відслідковувати вплив зміни однієї вимоги на інші пов'язані з нею вимоги, забезпечують спільну роботу над проектом кількох учасників і регулюють можливості по внесенню змін відповідно за присвоєними їм ролями. Засоби управління вимогами дозволяють значно знизити трудомісткість роботи з вимогами.

На цей час ринок засобів управління вимогами представлений такими продуктами як:

Rational RequisitePro, Rational Requirements Composer, Rational/Telelogic DOORS; Borland CaliberRM; Sybase PowerDesigner, Sparx RaQuest.

Вони розроблені з метою підвищення якості розробки програмних продуктів. Призначені для покращення взаємодії між учасниками проекту, спрощення аналізу впливу та процесу передачі інформації у сфері управління змінами початкових вимог, автоматизують процес збору, трасування, аналізу та управління змінами вимог.

Зазначені продукти відображають тенденцію до збільшення інтеграції з іншими інструментальними засобами, які використовуються при розробці ПЗ. Обираючи засіб для управління вимогами, необхідно з'ясувати, чи зможе він обмінюватися даними з іншими інструментами, що використовуються користувачем.

Rational RequisitePro допомагає робочим групам управляти вимогами, створювати якісні сценарії використання, розширювати можливості відстеження, підвищувати ефективність спільної роботи, зменшувати потребу в доробках і підвищувати якість [3, 5]. Продукт орієнтований на:

- зниження складності завдяки детальним представленням з можливістю трасування;

- зниження ризиків пов'язаних з проектом, відображаючи вимоги, які можуть бути порушені змінами вимог нижчого або більш високого рівня;
- забезпечення спільної роботи географічно розподілених робочих груп завдяки застосуванню повнофункціонального, масштабованого Web-інтерфейсу і ланцюжків обговорення;
- забезпечення збору та аналізу відомостей про вимоги з можливістю точного налаштування атрибутів і фільтрації.

- підвищення продуктивності праці, дозволяючи відслідковувати зміни шляхом порівняння версій проекту з початковими характеристиками, описаними за допомогою XML.

- забезпечення відповідності результатів проекту поставленим завданням і бізнес-цілям завдяки інтеграції з засобами IBM Rational для розробки і випуску ПЗ.

Rational Requirements Composer допомагає робочим групам ефективно визначати і застосовувати вимоги на всіх етапах ЖЦ проекту, скорочує необхідність доопрацювання проекту, прискорює вихід на ринок і покращує бізнес-результати [3-5]. Вимоги створюються і використовуються всіма учасниками робочої групи. Продукт орієнтований на:

- включення широкого кола замовників та інших зацікавлених осіб у процес розробки вимог;
- підвищення гнучкості і передбачуваності завдяки застосуванню спрощених методів розробки вимог;

- забезпечення аналізу і впорядкування вимог, а також ефективне управління ними за допомогою атрибутів, наборів, міток, фільтрів, готових шаблонів звітів (включаючи специфікацію користувальницького інтерфейсу, історію аудиту та звіти про відстеження);

- забезпечення організації роботи розробників і тестувальників відповідно до пріоритетів бізнесу;

- забезпечення можливості конкретизації вимог за допомогою візуальних і текстових засобів;

- забезпечення спільної роботи географічно розподілених робочих груп.

- забезпечення інтеграції з іншими продуктами.

IBM Rational/Telelogic DOORS – сімейство рішень, призначене для керування вимогами і розробкою складних наукомістких систем (авіа, суднобудування, ракети, автомобілі). Він є потужним інструментом управління вимогами, що полегшує збір, відслідковування і аналіз змін, а також управління ними [3-4]. Продукт орієнтований на забезпечення:

- скорочення витрат;

- підвищення ефективності та покращення очікуваного результату за рахунок підвищення якості роботи з вимогами;

- організації процесу управління вимогами в масштабах всієї організації;

- процесу управління змінами вимог;

- можливості обміну вимогами між зацікавленими сторонами;

- можливості зв'язування вимог з елементами проекту, планами тестування, наборами тестових даних;

- можливості масштабування у відповідності із змінюваними потребами по управлінню вимогами;

- можливостей обговорення вимог.

Borland Caliber RM – система, яка розроблена з метою підвищення якості створюваних продуктів і призначена для покращення взаємодії між учасниками робочої групи проекту, спрощення аналізу впливу і процесу передачі інформації в сфері управління змінами початкових вимог [3-4]. Продукт орієнтований на забезпечення:

- планування робіт по проекту;

- управління вимогами;

- оцінки проекту на основі вимог;

- сортування і призначення пріоритетності вимог;

- можливостей трасування вимог;

- можливостей обміну та обговорення вимог;

- інтеграції з іншими продуктами;

Sybase PowerDesigner – набір інтегрованих засобів моделювання, який поєднує в собі декілька стандартних технологій: управління вимогами, моделювання даних і моделювання застосувань за допомогою уніфікованої мови моделювання (UML) та дозволяє компаніям створювати гнучкі IT-рішення [6].

PowerDesigner дозволяє створювати моделі бізнес-процесів, що необхідні аналітикам вимог, перетворювати ці моделі в моделі бізнес-процесів, моделі даних і моделі архітектури застосувань; управляти вимогами; інтегруватися із середовищами розробки (Eclipse, Microsoft Visual Studio .NET).

До складу PowerDesigner входять різноманітні моделі (Business Process Model, Requirements Model, Conceptual Data Model, Logical Data Model та ін.). Requirements Model (RQM) – модель, що призначена для збору і управління вимогами. Дозволяє здійснювати детальний аналіз вимог, будувати ієрархію вимог і пов'язувати їх з будь-якими об'єктами інших моделей, що надає можливість відслідковувати хід виконання і покриття вимог.

Останнім часом широкої популярності набула надбудова над Enterprise Architect для роботи з вимогами – Sparx RaQuest [7].

Вона дозволяє відображати списки вимог у ієрархічному та лінійному вигляді з деякими правилами відбору: по пакету, статусу і ін.; задавати певний ЖЦ роботи з вимогами; здійснювати графічне представлення зв'язаних (залежних) вимог з метою проведення аналізу впливу змін вимог; додавати коментарі, задавати кольорові налаштування вимог, фільтрувати вимоги.

Важлива відмінність наявних інструментальних засобів полягає в способі збереження даних. Одні зберігають всі вимоги, атрибути і інформацію трасування в базі даних (наприклад, RequisitePro, Requirements Composer, DOORS, CaliberRM, Sybase PowerDesigner), а інші – ні (наприклад, MS SharePoint). В залежності від продукту, база даних може бути комерційною або розробленою власними силами. Вимоги можуть імпортуватися із різноманітних документів, але потім вони зберігаються в базі даних. Деякі продукти (наприклад, RequisitePro, Requirements Composer та ін.) дозволяють встановлювати зв'язки з зовнішніми файлами Microsoft Word, Microsoft Excel, графічними файлами в яких міститься інформація, яка доповнює вміст бази.

При виборі інструментального засобу, необхідно враховувати наступні параметри: платформа, ціна, режим доступу і способи збереження даних для вимог – документальний або в БД, які найбільш підходять середовищу розробки і культурі користувача.

Висновки. Основою успіху при створенні надійного і корисного ПЗ завжди є чітке розуміння потреб його користувачів. В той час, коли технології розробки програм постійно удосконалюються й нарощують свої можливості, неправильне або неповне розуміння потреб користувачів створюваного ПЗ усе ще залишається однією із причин провальних результатів проектів при його розробці.

В управлінні вимогами присутня значна організаційна складова, яка допомагає управляти не тільки стадією проектування, але і проектом загалом і в цілому. Організація ефективного процесу управління вимогами допомагає знизити обсяг повторно виконуваних дій і прискорити виведення продуктів на ринок за рахунок співпраці із зацікавленими особами; підвищити продуктивність за рахунок контролю змін у вимогах; знизити ризики і витрати шляхом оцінки впливу змін у міру їх виникнення і довести відповідність вимогам шляхом повного їх відстеження.

ПЕРЕЛІК ПОСИЛАНЬ

1. Лаврищева Е.М. Методы и средства инженерии программного обеспечения. Учебник / Е.М. Лаврищева, В.А. Петрухин // Московский физико-технический институт. – 2006. – 304с.
2. Мезерний А.М. Застосування концепції автоматизованої побудови специфікації вимог до процесу обліку та формалізації вимог до програмного забезпечення / А.М. Мезерний, Д.В. Кукленко // Вісник Нац. техн. ун-ту «ХПІ»: НТУ «ХПІ». – 2011. – № 9. – С. 82-89.
3. Вигерс К. Разработка требований к программному обеспечению / К. Вигерс. – М.: Русская редакция, 2004. – 576с.
4. Automating requirements management. [Електронний ресурс]. – Режим доступу: http://www.processimpact.com/articles/rm_tools.html
5. Управление требованиями. [Електронний ресурс]. – Режим доступу: http://cmcons.com/articles/upravlenie_trebovanijami_instrument_ibm_rational_r/
6. PowerDesigner. [Електронний ресурс]. – Режим доступу: <http://www.sybase.ru/products/powerdesigner>
7. RaQuest - Requirements Management Tool. [Електронний ресурс]. – Режим доступу: <http://www.raquest.com/products/>

REFERENCES

1. Lavrishcheva, E.M., Petruhyn, V.A. (2006) Methods and tools software engineering. Textbook. Moscow, Physical-technical institute, 304 p. [in Russian].
2. Mezernyy, A.M. (2011) Applying the concept of automated build process specification requirements for accounting and formalize software requirements. Visnuk Nacionalnogo tehnicnogo universutety «KPI», (9), pp. 82-89 [in Ukrainian].

3. Vigers, C. Development of software requirements (2004). Moscow, Russian edition, 576 p. [in Russian].
4. Automating requirements management. [Electronic resource]. – Access mode: http://www.processimpact.com/articles/rm_tools.html
5. Management requirements. [Electronic resource]. – Access mode: http://cmcons.com/articles/upravlenie_trebovanijami_instrument_ibm_rational_r/
6. PowerDesigner. [Electronic resource]. – Access mode: <http://www.sybase.ru/products/powerdesigner>
7. RaQuest - Requirements Management Tool. [Electronic resource]. – Access mode: <http://www.raquest.com/products>

РЕФЕРАТ

Рябокін Ю.М. Програмні засоби для управління вимогами / Ю.М. Рябокін // Управління проектами, системний аналіз і логістика. Науковий журнал: в 2 ч. Ч. 1: Серія: „Технічні науки” – К. : НТУ, 2014. – Вип. 14.

Розглянуті загальні проблеми встановлення, формулювання та документування вимог.

Об’єкт дослідження – програмні засоби управління вимогами.

Мета роботи – розкрити особливості процесу управління вимогами.

Метод дослідження – аналіз існуючих програмних засобів, які використовуються для роботи з вимогами.

Визначені особливості організації зв’язків вимог різних рівнів, характеристики вимог, посадові обов’язки аналітика вимог, зв’язки аналітика вимог із зацікавленими особами проекту, складові процесу управління вимогами. Встановлені основні причини складності управління вимогами: велика кількість потенційно «зацікавлених осіб», різноманітність типів вимог, змінюваність та необхідність трасування вимог; проблеми, пов’язані з представленням та зберіганням вимог. Наведено перелік спеціалізованих програмних засобів, що використовуються для управління вимогами, подано їх загальну характеристику; представлені результати їх аналізу та порівняння.

КЛЮЧОВІ СЛОВА: АНАЛІТИК ВИМОГ, ЗБІР ВИМОГ, УПРАВЛІННЯ ВИМОГАМИ, ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ, ПРОГРАМНИЙ ЗАСІБ.

ABSTRACT

Ryabokin Y.M. Software tools for requirements management / Y.M. Ryabokin // Management of projects, system analysis and logistics. Science journal: In Part 2. Part 1: Series: "Technical sciences" - Kyiv: NTU, 2014. - Vol. 14.

In the article are considered general problems definition, formulation and documentation requirements.

Object of study – software requirements management.

Purpose of study – to reveal features of the process control requirements.

The method of study – analysis of existing software used for the requirements.

It was determined that the work with requirements is of the most responsible and least certain stage in the development of software which determines the success of the whole project. Any software requirements must be characterized by found, completeness, correctness, feasibility, need, priority, uniqueness and ability to be checked. Analyst of requirements must to identify and to capture requirements from different sources, to systematize requirements, to analysis requirements for quality control, to document requirements, to coordinate requirements with stakeholders, processing requests for change of system requirements. In identifying, documenting, verifying requirements is realized the constant connection analyst of requirements with project stakeholders: representatives of the customer and the developer. Capture and management of all software requirements are very difficult tasks. Requirements management is a necessary condition for successful software development. The main reasons related to the complexity of requirements: a large number of potentially "interested stakeholders", a variety of types of requirements and the need for tracing requirements; problems associated with the presentation and storage requirements. The basic processes of manage requirements are development of a plan manage requirements, development of the concept of building system, construction and detailing of model scenarios use of system. Identified, defined and documented requirements can be stored in plain text files or spreadsheets that are quite inconvenient and not practical when grouping, tracking links and changes the requirements, organization of joint work of the working group, as recently became popular use of specialized tools for managing requirements. This paper

provides a list of specialized software tools used to manage requirements, describes their general characteristics; presents the results of their analysis and comparison.

KEYWORDS: ANALYST OF REQUIREMENTS, COLLECTION OF REQUIREMENTS, REQUIREMENTS MANAGEMENT, SOFTWARE, SOFTWARE TOOL.

РЕФЕРАТ

Рябокоть Ю.Н. Программные средства для управления требованиями / Ю.Н. Рябокоть // Управление проектами, системный анализ и логистика. Научный журнал: в 2 ч. Ч. 1: Серия: „Технические науки” – К. : НТУ, 2014. – Вип. 14/

Рассмотрены общие проблемы определения, формулирования и документирования требований.

Объект исследования – программные средства управления требованиями.

Цель работы – раскрыть особенности процесса управления требованиями.

Метод исследования – анализ существующих программных средств, используемых для работы с требованиями.

Определены особенности организации связи требований разных уровней, характеристики требований, должностные обязательства аналитика требований, связи аналитика требований из заинтересованными лицами проекта, составляющие процесса управления требованиями. Установлены основные причины сложности управления требованиями: большое количество потенциально «заинтересованных лиц», разнообразие типов требований, изменчивость и необходимость трассировки требований; проблемы, связанные с представлением и сохранением требований. Приведен перечень специализированных программных средств, которые используются для управления требованиями, приводится их общая характеристика; представлены результаты их анализа и сравнения.

КЛЮЧЕВЫЕ СЛОВА: АНАЛИТИК ТРЕБОВАНИЙ, СБОР ТРЕБОВАНИЙ, УПРАВЛЕНИЯМИ ТРЕБОВАНИЯМИ, ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ, ПРОГРАММНЫЙ ИНСТРУМЕНТ.

АВТОР

Рябокоть Юлія Миколаївна, кандидат технічних наук, Національний авіаційний університет доцент кафедри інженерії програмного забезпечення Інституту комп'ютерних інформаційних технологій, e-mail: Yulia.Ryabokin@livenau.net, тел. (044) 406 70 98, Україна, 03058, м. Київ, просп. Космонавта Комарова, 1, корп. 6, к. 303.

AUTHOR

Ryabokin Yulia M, Ph.D., National Aviation University, Associate Professor of Software Engineering Department of Computer Information Technology Institute, e-mail: Yulia.Ryabokin@livenau.net, tel. (044) 406 70 98, Ukraine, 03058, Kyiv, ave. Komarova, 1, building 6, r. 303.

АВТОР

Рябокоть Юлия Николаевна, кандидат технических наук, Национальный авиационный университет, доцент кафедры инженерии программного обеспечения Института компьютерных информационных технологий, e-mail: Yulia.Ryabokin@livenau.net, тел. (044) 406 70 98, Украина, 03058, г. Киев, просп. Космонавта Комарова, 1, корп. 6, к. 303.

РЕЦЕНЗЕНТИ

Сидоров Микола Олександрович, доктор технічних наук, професор, Національний авіаційний університет, завідувач кафедри інженерії програмного забезпечення Інституту комп'ютерних інформаційних технологій, Київ, Україна.

Зіатдінов Юрій Кашафович, доктор технічних наук, професор, Національний авіаційний університет, завідувач кафедри комп'ютерних інформаційних технологій Інституту комп'ютерних інформаційних технологій, Київ, Україна.

REVIEWER

Sidorov M.O., Ph.D., Engineering (Dr), Professor, National Aviation University, Head of Software Engineering Department of Computer Information Technologies Institute, Kyiv, Ukraine.

Ziatdinov Y.K., PhD, Engineering (Dr), Professor, National Aviation University, Head of Computer Information Technologies Department of Computer Information Technologies Institute, Kyiv, Ukraine.