

На основі отриманих алгоритмів, результатів моделювання, та враховуючи сучасні вимоги щодо використання групи БПЛА (надійність живучість формації БПЛА як системи вцілому) найбільш підходящим методом для реалізації керування групою БПЛА є метод віртуального лідера; алгоритм роботи такого методу буде розроблено та досліджено в наступних роботах авторів.

Список використаних джерел:

1. Soleymani T., Saghafi F. «Behavior-Based Acceleration Commanded Formation Flight Control» International Conference on Control, Automation and Systems 2010 Oct. 27-30, in KINTEX, Gyeonggi-do, Korea. 2010.
2. Das A.K., Fierro R., Kumar V., Ostrowski J.P., Spletzer J. and Taylor C.J. «A Vision-Based Formation Control Framework», IEEE Transaction on Robotics and Automation, Vol. 18, № 5, 2002.
3. Hammer J., Piper G., Thorp O. and Watkins J. «Investigating Virtual Structure Based Control Strategies for Spacecraft Formation Maneuvers», AIAA Guidance, Navigation, and Control Conference and Exhibit, Providence, Rhode Island, 2004.
4. Gosiewski Z., Ambroziak L. «UAV Autonomous Formation Flight Experiment with Virtual Leader Control Structure», Solid State Phenomena, Vol. 198, pp. 254-259, 2013.
5. Paul T., Krogstad T.R. and Gravdahl J.T. «Modelling of UAV Formation Flight Using 3D Potential Field», Simulation Modelling Practice and Theory, vol. 16, № 9, pp. 1453–1462, Oct. 2008.
6. Randal W. Beard, Timothy W. McLain «Small Unmanned Aircraft: Theory and Practice» by Princeton University Press, 2012.

Пулик М.І., Петросаняк Т.П.

студенти,

Науковий керівник: Вовк Р.Б.

кандидат технічних наук, доцент,

Івано-Франківський національний технічний університет нафти і газу

АНАЛІЗ ТА УСУНЕННЯ РИЗИКІВ ПРИ УПРАВЛІННІ ПРОЕКТАМИ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

У сучасному світі галузь інформаційних технологій набула глобального поширення у людському житті, тому її розвиток посідає визначальне місце в усіх сферах суспільної діяльності. Важливим аспектом є правильна організація управління розробкою програмного продукту, адже допущення невизначеностей і неоднозначностей може спричинити низькі показники успішності проекту.

Вибір певної моделі життєвого циклу забезпечує вдале керівництво процесом створення програмного продукту, як і врахування можливих наслідків, зумовлених ризиками. Останнім часом популярною стала технологія управління проектами Scrum, яка ставить на перше місце регулярне узгодження вимог за допомогою спілкування між виконавцем та замовником. Від вдалої ідентифікації, виявлення та знешкодження загроз коректній роботі кінцевого

продукту залежить ступінь його відповідності до вимог. Тому доцільним є аналіз виникнення та усунення ризиків у методології Scrum, що і є метою нашого дослідження.

Scrum – це одна з найпопулярніших гнучких методологій управління розробкою програмних продуктів класу Agile, що побудована на принципах залучення в процес всіх учасників, в кожного з яких є певна роль, та передбачає регулярну співпрацю із замовником задля узгодження створеного на даний момент функціоналу [1]. Її особливістю є ітеративний процес розробки, що полягає у виконанні командою фіксованого списку завдань протягом спринту – визначеного періоду часу. Через високу конкуренцію компанії-розробники повинні створювати програмні продукти швидко, якісно, звертаючи особливу увагу на задоволення потреб користувача. Оперативне прийняття рішень, встановлення пріоритетів та чітке розуміння вимог клієнта забезпечує саме ця методологія. Однак немає ідеального вирішення будь-якої проблеми, тому, як і в споріднених технологіях, в Scrum також існує ймовірність виникнення ризиків. Термін ризик програмного продукту пов'язується з наявністю невизначеностей, які можуть призвести до зниження якості кінцевого проекту, підвищення вартості його розробки, затримки календарного плану або зриву проекту.

Існує чотири основних категорій ризиків. Перша з яких – це внутрішні складнощі календарного планування, які полягають у суттєвих відмінностях між запланованими і реальними термінами реалізації (іноді досягає 50-80% [2]), що пов'язані із створенням планів менеджерами, які керуються побажаннями замовника або некоректною оцінкою можливостей команди розробників.

Друга категорія – збільшення вимог з боку замовника під час розробки проекту виникає через те, що часто потреби замовників змінюються, і як наслідок це призводить до додаткових трудовитрат.

Третій різновид ризиків – це плінність кадрів, оскільки заміна досвідчених розробників під час проекту, які мають навички роботи у колективі, а також знають специфіку виконання завдання, спричиняє збій у роботі всієї команди.

Останньою категорією ризиків є низька продуктивність. Ефективність роботи колективу важко оцінювати як одне ціле, а згідно з наслідком закону Паркінсона відомо, що показник продуктивності праці розробників зростає тільки до кінця терміну здачі проекту.

Коли йдеться про управління ризиками, в Agile-проектах, порівняно з традиційними моделями такими, як «водоспадна», є декілька ризиків, які є більш притаманними для даних типів проектів. Проведемо детальне дослідження цих ризиків і основні способи їх усунення.

Низька продуктивність, окрім некоректної працездатності виконавців, може також бути спричинена тим, що команда зайнята відлагодженням недоліків попередніх спринтів. Після того, як результати виконання спринту демонструються замовнику, команда продовжує тестувати їх, в той час як наступний набір завдань перебуває в процесі розробки. Часто високий відсоток невідповідності технічному завданню означає, що команда не звертає належної уваги на поточну мету спринту, а натомість цього фокусується на виправленні проблем попередньої ітерації. Для запобігання цього може створюватися один

додатковий спринт перед випуском, який називають «hardening sprint» [3] (зміцнюючий, стабілізуючий), що призначений для усунення недоліків. У випадку розробки програмного продукту, що потребує великого об'єму роботи, тільки протягом певних періодів може бути створена окрема команда для усунення невідповідностей.

Вищезгадану категорію ризиків доповнює проблема зниження прагнення розробників до самовдосконалення. За належної підготовки команда здатна до самоорганізації. Проте протягом певного проміжку часу принципи Scrum мають схильність ставати одноманітними, і команди більше не прагнуть постійного вдосконалення. Варіантами вирішення проблеми є проведення періодичних тренувань та занять, Agile-ігор, вікторин, впровадження нових ідей та тенденцій. Ці способи мотивують розробників виконувати вибрані завдання старанніше та краще.

Оскільки цілі спринту базуються на швидкості та оперативності команди-виконавця, зміна її складу часто призводить до втрати знань, набутих в даному проекті. У результаті це загрожує цілісному виконанню спринту чи навіть факту його завершення, що становить третю категорію ризиків – плинність кадрів. Варіантом запобігання є уникнення призначення конкретної особи на типові завдання з однієї і тієї ж предметної області, тобто здійснення короткострокової ротації кадрів. Це допоможе учасникам розробки орієнтуватись у всіх питаннях та завданнях проекту.

Однією з найважливіших бізнес-вимог до програмного продукту є вчасна його реалізація відповідно до попиту ринку. Для усунення ризику недотримання графіку в методології гнучкої розробки необхідно забезпечити наявність резерву часу на випадок помилок планування і виникнення непередбачуваних обставин, а також максимально залучати розробників до оцінки термінів.

Оскільки Agile – це клас методологій розробки ПЗ, в якій вимоги еволюціонують через тісну співпрацю між виконавцем та замовником, часто зустрічається проблема зміни вимог з боку клієнта в сторону їх збільшення. Замість ігнорування або неприйняття даних змін використовується призначення пріоритетів, що дозволяє раціонально виконати необхідні нововведення під час розробки, тобто задачі з низьким пріоритетом замінюються на нові завдання з вищими пріоритетами.

Отже, ідентифікація виникнення ризиків при розробці програмних проектів на основі гнучкої методології Scrum дозволила обґрунтувати необхідність створення плану запобігання ризикам під час розробки. У зв'язку з цим було запропоновано можливі шляхи вирішення складностей календарного планування та проблеми зростання вимог з боку замовника, уникнення плинності кадрів, запобігання низької продуктивності. Описані варіанти усунення невизначеностей допоможуть компанії-розробнику реалізувати якісне програмне забезпечення без додаткових затрат, що повністю задовольнятиме потреби користувача.

Список використаних джерел:

1. Sutherland J. J. Scrum: The Art of Doing Twice the Work in Half the Time. – NY: Crown Business, 2014.

2. DeMarco T., Lister T. *Waltzing with Bears: Managing Risk on Software Projects*. – NY: Dorset House Publishing, 2003.
3. *Building Real Software*. [Електронний ресурс] – Режим доступу: <http://swreflections.blogspot.com/2013/01/hardening-sprints-what-are-they-do-you.htm>

Рибальченко І.С.

студент;

Матвієнко О.А.

*кандидат технічних наук, асистент,
Сумський державний університет*

МЕТОДИ ПІДВИЩЕННЯ ЕНЕРГОЕФЕКТИВНОСТІ ВІДЦЕНТРОВИХ НАСОСІВ НИЗЬКОЇ ШВИДКОХІДНОСТІ

За різними оцінками насосне обладнання споживає 20-25% електроенергії, що виробляється у світі [1]. У зв'язку з цим, гостро постають питання підвищення енергетичної ефективності насосного обладнання. Вирішення цього питання потребує комплексного підходу, який включає в себе не тільки заміну старих насосів на нові, більш ефективні, а й модернізацію працюючих насосів шляхом використання змінного робочого колеса (за деякими даними [2] життєвий цикл корпусу насоса вище, ніж робочого колеса – частка зношення для робочого колеса складає 77%, для корпусу – 18%).

Не зважаючи на те, що відцентрові насоси досить детально досліджені, а їх робочий процес ґрунтовно описаний в багатьох наукових працях, пошуки шляхів впливу на їх напірні та енергетичні характеристики продовжуються. Особливо це стосується відцентрових насосів з нетипово низьким коефіцієнтом швидкохідності ($n_s < 45$). Такі насоси характеризуються високими напорами при відносно низьких витратах, при цьому їх коефіцієнт корисної дії здебільшого не перевищує 60%. Сферою застосування насосів такого типу, перш за все, є нафтовидобувна (насоси систем підтримки пластового тиску) та хімічна промисловість (перекачування легких речовин, газорідних сумішей тощо).

Виходячи з вищевикладеного, метою даної роботи є підвищення енергоефективності робочого колеса відцентрового насоса низької швидкохідності ($n_s < 45$).

На сьогодні існує декілька основних способів підвищення ефективності відцентрових робочих коліс низької швидкохідності.

Одним з найпростіших методів збільшення напору є затиловка лопатей робочих коліс [3]. Під затиловкою розуміється зміна вихідної кромки лопаті робочого колеса з тильного боку шляхом знімання металу. Різні модифікації затиловки, що застосовуються на практиці, показані на рис. 1. Така зміна дозволяє збільшити напір насоса на 5-8% при практично незмінному ККД на оптимальному режимі. Даний метод описаний в роботах Пфлейдера [4], проте певної методики розрахунку впливу затиловки на характеристики робочого