УДК 004.41:004.056 **doi: 10.26906/SUNZ.2018.3.116**

О. В. Коваленко

Центральноукраїнський національний технічний університет, Кропивницький, Україна

МЕТОДИ ЯКІСНОГО АНАЛІЗУ ТА КІЛЬКІСНОЇ ОЦІНКИ РИЗИКІВ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

В роботі визначено і вирішено одне з протиріч, що виникають при розробці ПЗ, яке полягає в нехтуванні фірмами-розробниками ПЗ факторів вразливості безпеки ПЗ. Усі ризики при розробці програмного забезпечення, з більшим або меншим допущенням, можна вважати суб'єктивним результатом виконання процесу, який пов'язаний з нестачею кількісної або якісної інформації про процес, а також її невизначеністю. Зазначені фактори можна вважати головною причиною, яка породжує і супроводжує ризики у всьому їхньому життєвому циклі. В якості вирішення зазначеної проблеми запропоновано використання розроблених методів якісного аналізу та кількісної оцінки ризиків розробки програмного забезпечення.Його відмінною особливістю є врахування факторів експлуатаційних ризиків, особливо ризику невиявлення вразливостей ПЗ і оцінки довільного несуперечливого кінцевого набору «квантів інформації». Доведено, що методика якісної оцінки ризиків проекту є описовою і являє собою процес, спрямований на виявлення конкретних ризиків проекту, а також причин, що їх породжують, з подальшою оцінкою можливих наслідків і вироблення заходів для роботи з ризиками. У процесі якісного аналізу ризиків відбувається вироблення метрик, що відповідають за визначення граничних показників факторів, які сигналізують про прояв ризиків.Розроблено метод кількісної оцінки ризиків розробки ПЗ. Його відмінною особливістю є комплексне використання методики «Аналізу дерева відмов» і способу оцінки показника чистої приведеної вартості проекту розробки ПЗ з урахуванням негативних факторів можливого невиявлення вразливостей безпеки ПЗ. У той же час використання способу оцінки показника чистої приведеної вартості проекту розробки ПЗ дозволяє розглядати проект комплексно, з урахуванням необхідності врахування безпеки і тестування вразливості ПЗ із залученням інструментів, які дозволяють подолати складність, невизначеність і довгостроковість проектів.

Ключові слова: оцінка ризиків, розробка програмного забезпечення.

Постановка проблеми дослідження та аналіз літератури

В даний час у більшості організацій і підприємств різних форм власності все більше уваги приділяється питанням аналізу та оцінки ризиків. Але, не дивлячись на це, проблеми та питання, що відносяться до загальної теорії та методології аналізу, оцінки й управління ризиками, потребують адаптації до підходів і положень сучасного менеджменту, врахування нових факторів становлення і розвитку технологій, об'єднання відомих «усталених» положень теорії ризиків з новими, прогресуючими підходами аналізу і синтезу.

Аналіз літератури [1-8] показав, що, не дивлячись на достатньо глибоку історію розвитку поняття «ризик» і спроби ряду відомих авторів сконцентрувати свої розробки в області управління ризиками окремих галузей і напрямів діяльності, розробка нових, перспективних наукових положень в цій області все ж дещо «звужена» фінансовою діяльністю. В той же час широке використання в нашій роботі інформаційних технологій потребує підвищеної уваги до цього напряму, і відповідно, більш глибокого висвітлення питань ризик-менеджменту ІТ-індустрії.

Суттю будь-якого процесу, явища або об'єкта (в тому числі й інформаційної складової) є діяльність, яка приводить до формування результатів. У застосуванні до такого напряму діяльності, як розробка програмного забезпечення, кінцевим результатом, в більшості практичних випадків, є виконання вимог замовника і впровадження розробленого продукту. Сучасні автори [3-8] дуже часто результат оцінюваного ризику зводять до негативного типу ефекту, забуваючи, що навіть сам термін «ризик» означає можливість або ймовірність настання подій з конкретними наслідками в результаті певних рішень або дій. Доцільність такого подання понять в теорії ризику особливо підкреслюється закономірностями, що виникають в інформаційних відношеннях при розробці програмного забезпечення, де складність і динаміка взаємозв'язків, нечіткість зовнішніх факторів, а також гетерогенність у структурній та функціональній побудові систем дозволяє розширити класифікацію результатів інформаційної діяльності.

Варто зауважити, що об'єктивний результат є наслідком цілеспрямованого і явного виконання процесу, який пов'язаний з його суттю. Суб'єктивні результати проявляються у тих випадках, коли виконання процесу проходить з недостатнім рівнем визначеності і повноти інформації.

На практиці у сфері ІТ-індустрії, переважна кількість ризиків пов'язана саме з суб'єктивними результатами здійснення ходу або виконання процесу.

Таким чином, можна відмітити, що усі ризики при розробці програмного забезпечення, з більшим або меншим допущенням, можна вважати суб'єктивним результатом виконання процесу, який пов'язаний з нестачею кількісної або якісної інформації про процес, а також її невизначеністю. Зазначені фактори можна вважати головною причиною, яка породжує і супроводжує ризики у всьому їхньому життєвому циклі. Кожен ризик циклу розробки програмного забезпечення (ПЗ) можна пов'язати з одним із наступних компонентів: дані; людина; система. При цьому слід врахувати ступінь впливу і відповідальності результатів оцінки ризиків для різних методологій розробки програмного забезпечення.

Аналіз літератури [1-8] показав, що в даний час існує множина $R=\{x_1, \ldots, x_n\}$ різних методик розроб-

ки ПЗ. Слід зауважити, що вибір безпосередньо методики при реалізації проекту має суттєвий вплив на результати аналізу, оцінки та управління ризиками. Відомо, що однією з широко використовуваних методологій розробки ПЗ є спіральна методологія - ця методологія керується інкрементними розробками на основі ризиків. Більше 15% тимчасових витрат управління ІТ-проектами йде на аналіз та оцінку ризиків. Аналіз літератури [3-8] показав, що сучасні автори в своїй більшості виділяють п'ять основних ризиків: помилки, властиві розкладу, поява нових вимог, зміна співробітників, декомпозиція специфікації, низька продуктивність.

Проведені дослідження показали, що дана позиція спірна, оскільки не враховує ряд важливих аспектів розробки ПЗ. Аналіз нормативної документації ряду відомих фірм-розробників ПЗ показав, що на етапі оцінки ризиків, як правило, не враховуються ризики, пов'язані з можливою наявністю помилок в моделях, алгоритмах, програмах обробки інформації, які використовуються для вироблення керуючих рішень, нехтуються ризики безпеки (можливих помилок, що впливають на вразливість ПЗ). Це часто призводить до помилок і, відповідно, необґрунтованих втрат (часових, економічних, іміджевих та ін.).

Таким чином, проведені дослідження показали, що, незважаючи на важливість вирішення завдання управління ризиками при розробці ПЗ, на даний момент немає чітко сформованої, стандартизованої методологічної бази опису даного процесу. В даний час спостерігається:

– відсутність єдиного, комплексного і системного підходу до проблеми виникнення ризиків при розробці ПЗ;

– відсутність ясності та прозорості у розумінні кінцевих результатів впливу ризиків, їх недостатнього врахування при розробці ПЗ;

– значні різночитання у розумінні методик аналізу, оцінки та управління ризиками;

– недостатність урахування важливих факторів, що виникають з вдосконаленням технологій і засобів розробки ПЗ.

Аналіз літератури [1-8] та проведені дослідження показали, що загальна послідовність оцінки ризиків найчастіше включає в себе наступні дії:

– виявлення джерел і причин ризику розробки ПЗ, етапів і робіт, при виконанні яких виникає ризик.

– ідентифікація всіх можливих ризиків, властивих розглядуваному проекту.

– документування результатів та їх подальша приорітизація.

– оцінка рівня окремих ризиків і ризику проекту, що визначає його економічну доцільність.

– визначення допустимого рівня ризику розробки ПЗ.

– розробка заходів зі зниження ризику.

Відповідно до даного алгоритму, оцінка ризику підрозділяється на три взаємодоповнюючих напрямки: якісний (етапи 1-3) та кількісний аналіз (етапи 4, 5) ризиків розробки ПЗ, а також управління (етап 6).

Метою роботи є розробка комплексу методів якісного аналізу та кількісної оцінки ризиків розробки програмного забезпечення, який дозволить вирішити протиріччя, що виникає при розробці ПЗ.

Дослідимо більш докладно методики якісного та кількісного аналізу ризиків розробки ПЗ.

Результати досліджень

Проведені дослідження показали, що методика якісної оцінки ризиків проекту є описовою і являє собою процес, спрямований на виявлення конкретних ризиків проекту, а також причин, що їх породжують, з подальшою оцінкою можливих наслідків і вироблення заходів для роботи з ризиками. У процесі якісного аналізу ризиків відбувається вироблення метрик, що відповідають за визначення граничних показників факторів, які сигналізують про прояв ризику(-ів).

Розглядаючи перший пункт наведеного вище переліку дій щодо якісного та кількісного аналізу ризиків, зауважимо, що початкові дані для виявлення і опису характеристик ризиків можуть братися з різних джерел: база знань організації; інформація з відкритих джерел, наукових праць; маркетингова аналітика; опитування експертів та ін.

Ряд відомих авторів [1-8], провівши дослідження, виявили найбільш поширені ризики при розробці ПЗ. Наприклад, автори Демарко і Лістер [1] наводять свій список з п'яти найбільш важливих джерел ризиків будь-якого проекту розробки ПЗ: вади календарного планування; плинність кадрів; роздування вимог; порушення специфікацій; низька продуктивність. Можна відзначити, що даний перелік має узагальнений характер, що в значній мірі ускладнює метричну оцінку наведеного списку.

Баррі Бом в своїй роботі [2] розширює список до 10 найбільш поширених ризиків програмного проекту: дефіцит фахівців, нереалістичні терміни і бюджет, реалізація невідповідної функціональності, розробка неправильного користувальницького інтерфейсу, «золота сервіровка», перфекціонізм, непотрібна оптимізація і відточування деталей, безперервний потік змін, брак інформації про зовнішні компоненти, що визначають оточення системи або залучені в інтеграцію, недоліки в роботах, виконуваних зовнішніми (по відношенню до проекту) ресурсами, недостатня продуктивність одержуваної системи, «розрив» в кваліфікації фахівців різних галузей знань. Однак цей перелік не повний, і неструктурований. Це ускладнює процес оцінки взаємовпливу наведених ризиків один на одного.

Досить докладно ризики були оцінені і класифіковані в роботах [1-10]. Відповідно до даних досліджень ризики класифікуються за наступними ознаками: середовище (внутрішній, зовнішній ризики); природа (економічний, технічний, технологічний); сфера (ризик проекту, процесу, продукту); рівень (від критичного до незначного ризику); галузь впливу (ризик невиконання бюджету проекту, ризик невиконання плану проекту, ризик невиконання якості проекту); ланка управління ризиком (ризик окремого процесу, ризик проекту, ризик компанії). Однак подібна класифікація робить акцент на проектах розробки програмних систем, які не пов'язані з процесами їх подальшого впровадження та адаптації систем в умо-

вах конкретної організації, експлуатації в умовах можливих зовнішніх зловмисних впливів.

Використовуючи результати досліджень наведених вище авторів, думки експертів, маркетингові дані, а також бази знань відомих фірм представляється доцільним ідентифікувати ризики розробки ПЗ у вигляді сукупності множин:

– організаційних ризиків $Z = \{Id1, \dots, Id5\}$:

Id 1 – невиконання стратегії розробки ПЗ;
Id 2 – невідповідність організаційної структури;
Id 3 – неадекватний вибір стратегії розробки ПЗ;
Id 4 – низький рівень підтримки топ-менеджментом;
Id 5 – культурні проблеми та проблеми оточення;

– управлінських ризиків $U = \{Id6, \dots, Id9\}$:

Id 6 – неефективний менеджмент проектів;
Id 7 – неадекватний менеджмент змін при розробці ПЗ;
Id 8 – невикористання груп лідерів розробки ПЗ;
Id 9 – недостатній супровід зі сторони менеджерів;

– операційних ризиків $Y = \{Id10, \dots, Id15\}$::

Id 10 – неадекватний фінансовий менеджмент;
Id 11 – неадекватне навчання та інструктування;
Id 12 – неефективні комунікації;
Id 13 – неефективний консультаційний сервіс;
Id 14 – нестабільність дій постачальників;
Id 15 – неадекватний реінжиніринг бізнес-проектів;

– технологічних ризиків $T = \{Id16, \dots, Id20\}$:

Id 16 – непередбачена технічна складність;
Id 17 – неадекватне забезпечення системи розробки ПЗ технічними потужностями;
Id 18 – неефективне обслуговування і переоснащення системи;
Id 19 – неадекватний менеджмент існуючих систем;
Id 20 – неадекватна взаємодія з іншими системами;

– експлуатаційних ризиків $E = \{Id21, \dots, Id24\}$:

Id 21 – неадекватна складність інтерфейсів;
Id 22 – недостатній експлуатаційний супровід ПЗ;
Id 23 – неадекватний взаємовплив експлуатованого і впроваджуваного ПЗ;
Id 24 – невиявлення вразливостей безпеки ПЗ;

– соціальних ризиків $C = \{Id25, \dots, Id27\}$:

Id 25 – недостатня комунікація з користувачами;
Id 26 – неефективний менеджмент при створенні груп розробників ПЗ;
Id 27 – неефективна взаємодія між стейкхолдерами;

– правових ризиків $W = \{Id28, Id29\}$: :

Id 28 – зміна законодавства;
Id 29 – неадекватність дій менеджменту органів державного управління.

Відмінною особливістю представленої класифікації є врахування експлуатаційних ризиків. Особливої важливості ці ризики набувають в умовах підвищеного рівня кіберзлочинності, коли нехтування вразливостями програмного забезпечення може призвести до експлуатаційних проблем, а часто і до неможливості експлуатації («краху») ПЗ. Крім цього, в умовах українського правового поля спостерігаються окремі випадки неадекватності і невідповідності правовим нормам дій посадових осіб державного апарату. Практика ряду відомих фірм-розробників ПЗ показує, що зазначений фактор ризику доцільно враховувати при розробці ПЗ, нарівні з фактором можливої зміни українського законодавства.

Вплив зазначених вище ризиків на основні фактори успіху розробки, впровадження та тривалої експлуатації ПЗ проілюстровано наступними залежностями. Провал при розробці ПЗ можливий при реалізації наступних ситуацій:

– провал при експлуатації ПЗ (включає в себе наступну групу ситуацій: непідтримка бізнес-процесів або нереалізація очікувань; недосягнення цілей; незадоволеність клієнтів/стейкхолдерів; погіршення іміджевого портрета клієнтів; погіршення показників інформаційної та функціональної безпеки) (через реалізацію ризиків: {*Id* 1,…,*Id* 5}, {*Id* 6,…, *Id* 9}, {*Id* 10}, {*Id* 12}, {*Id* 14,…,*Id* 20}, {*Id* 25,…, *Id* 29}).

– група ситуацій: перевищення бюджету, перевищення термінів, незавершення проекту (через реалізацію ризиків: {*Id* 1,…,*Id* 4}, {*Id* 6,…,*Id* 8}, {*Id* 11,…, *Id* 14}, {*Id* 16,…, *Id* 20}, {*Id* 21,…, *Id* 24}, {*Id* 25,…, *Id* 27}, {*Id* 29})

Як видно з цих залежностей, більшість з розглядуваних ризиків розробки ПЗ (організаційні, операційні, управлінські та ін.) можуть безпосередньо впливати як на процес розробки ПЗ, так і на процес його експлуатації. У той же час, наприклад, експлуатаційні ризики безпосереднього впливу на процес розробки ПЗ не мають. Але нехтування цими ризиками часто веде до провалу експлуатації ПЗ і втрати майбутніх замовлень і проектів (простоїв розробників ПЗ). Саме цим фактором викликаний зв'язок між блоками «Провал при експлуатації ПЗ» і «Провал при розробці ПЗ». Однак, незважаючи на це, в цілому можна виділити множину ризиків, що безпосередньо впливають на процес розробки ПЗ:

$$MR = \{Z, U, Y, C, T, W\},$$

і множину ризиків, що безпосередньо впливають на процес експлуатації ПЗ:

$$ME = \{Z, U, Y, C, T, W, E\}, (Id9, Id10, Id15, Id29 \notin ME).$$

Слід зауважити, що виділені вище фактори в достатній мірі описують перелік можливих ризиків розробки ПЗ. Однак вони не дають уявлення про взаємний вплив і, відповідно, можливу зміну кінцевого результату. Крім цього наведені множини ризиків розробки ПЗ в різному ступені впливають на кінцевий результат. Тому наступним кроком ідентифікації ризиків розробки ПЗ доцільно виконати процедури ранжування і виділення найбільш пріоритетних (важливих) ризиків розробки ПЗ. Проведені дослідження показали, що для вирішення задачі визначення взаємовпливу ризиків доцільно використовувати інструмент аналізу причинно-наслідкових зв'язків між різними факторами і ризиками, розроблений Каору Ісікава [11] (діаграма Ісікави). Відповідно до відомого принципу Парето [1], серед множини потенційних причин (причинних факторів, за Ісікавою), що породжують проблеми (наслідок), лише дві-три є найбільш значущими, їх пошук і повинен бути організований. Для цього здійснюється:

– збір і систематизація усіх причин, що впливають на досліджувану проблему;

– групування цих причин за смисловими і причинно-наслідковими блоками;

– їх ранжування всередині кожного блоку;
– аналіз отриманої картини.

Тому даний інструмент дозволяє прояснити і врахувати усі суттєві фактори, що впливають на результат розробки ПЗ.

Застосування діаграми Ісікави дозволяє з'ясувати причини будь-яких проблем в організації або, наприклад, причини виникнення експлуатаційних «багів» ПЗ. При цьому діаграма Ісікави має ряд переваг: допомагає наочно показати зв'язки між отриманим результатом і причинами, що його викликали; дозволяє провести аналіз ланцюжка факторів, що впливають на проблему.

Однак задачу вибору найбільш пріоритетних ризиків діаграма вирішити не може. Для вирішення цієї задачі в роботі пропонується використовувати математичний апарат багатокритеріальної оптимізації, що базується на локальній геометрії множини Парето.

Аналіз літератури [1-8] показав, що існують принаймні три формулювання багатокритеріальної оптимізації, що базується на локальній геометрії множини Парето:

– локальна. Знайти одне Парето-оптимальне рішення (найближче до заданої початкової точки).
– глобальна. Знайти кінцеву множину Парето-оптимальних рішень, які достатньо добре описують (покривають) істинний Парето-фронт.
– інтерактивна. Знайти Парето-оптимальне рішення, яке максимально задовольняє перевагам особи, що приймає рішення (ОПР).

Проведені дослідження показали, що в процесах, побудованих на принципах постійних комунікацій між учасниками, використання «мозкових штурмів» із залученням думок експертів, доцільним видається використання інтерактивного формулювання багатокритеріальної оптимізації.

У цих умовах абстрактну задачу вибору найбільш важливих ризиків розробки ПЗ з наявної початкової множини можливих (допустимих) варіантів (рішень) *X* можна сформулювати наступним чином.

Позначимо множину усіх заздалегідь визначених ризиків розробки ПЗ через *S(X)*. Вочевидь, *S(X)*∈*X*. Таким чином, в задачі вибору дано множину *X*, що містить принаймні два елемента, а вимагається знати деяку його не порожню множину *S(X)*. Припускається, що вибір проводиться ОПР, в ролі якого може виступати як окрема людина, так і цілий колектив розробників. Для того, щоб здійснюваний вибір найбільшою мірою відповідав досягненню наявної мети (тобто був «найкращим» або «оптимальним» для даної ОПР), необхідно в процесі вибору враховувати думку експертів. Проведені дослідження показали, що в даний час існує безліч підходів врахування думки експертів, проте всі вони мають істотні недоліки, головний з яких полягає в тому, що, не дивлячись на різноманіття і детальну вивченість ієрархій та «штучних» відношень, вкрай рідко будь-яке з них можна вважати задовольняючим певну ОПР в повній мірі. Характерним прикладом, що підтверджує даний факт є нехтування оцінкою вразливостей розробленого ПЗ (недостатність або повна відсутність pen-тестування).

Тому для вирішення задачі вибору найбільш пріоритетних ризиків (звуження множини Парето) пропонується використовувати «кванти інформації». Для цього розглянемо довільні оцінки ризиків розробки ПЗ

$$y' = (y_1', \ldots, y_m') \text{ і } y'' = (y_1'', \ldots, y_m''), y',$$

що належать множині Парето-оптимальних векторів $f(P_f(X))$. За визначенням множини Парето повинні знайтися такі дві непустих підмножини номерів критеріїв $A, B \subset I = \{1, 2, \ldots, m\}$, що

$$y_i' > y_i'', \quad y_i' - y_i'' = w_i > 0, \quad \forall i \in A, \qquad (1)$$

$$y_j'' > y_j', \quad y_j'' - y_j' = w_j > 0, \quad \forall j \in B, \qquad (2)$$

$$y_s'' = y_s', \quad \forall s \in I \setminus (A \cup B). \qquad (3)$$

Згідно з умовами (1-3), перший вектор перевершує другий за компонентами групи критеріїв *A*, тоді як другий перевершує перший за компонентами групи критеріїв *B*. За іншими компонентами (якщо такі є) два зазначених вектора збігаються. Звуження множини Парето, тобто видалення деяких Парето-оптимальних векторів, зазвичай відбувається на основі порівняння. Людині найпростіше порівнювати пари. Якщо при порівнянні фіксованої пари Парето-оптимальних векторів y′ і y″ виду (1-3) ОПР «вибраковує» один із цих векторів (наприклад, другий), то це означає, що для неї перший вектор переважніший другого, тобто $y' \succ y''$, де $\succ$ – відношення переваги, визначене на усьому критеріальному просторі $\Re^m$ і, яке збігається на множині *Y* з відношенням $\succ y$.

Співвідношення $y' \succ y''$, задає «квант інформації» про відношення строгої переваги, який свідчить про готовність ОПР до компромісу – вона згодна піти на втрати за всіма критеріями групи *B* у розмірі w_j для того, щоб отримати прибавки у розмірі w_i за критеріями групи *A*, зберігши при цьому значення всіх інших критеріїв.

Наявність вказаного «кванта інформації» дозволяє скоротити множину Парето на один вектор y″. Для того, щоб домогтися більшого скорочення можна прийняти, що $y' \succ y''$, має місце не тільки для даної пари векторів, але і для всіх тих векторів, які задовольняють умовам (1-3) за незмінних значеннях w_i і w_j. В цьому випадку запропоновано говорити, що група критеріїв *A* важливіша за групу *B*. При зазначеному розширенні дії «кванта інформації» можна розраховувати на більш помітне звуження множини Парето, хоча нерідко і воно виявляється недостатнім для остаточного вибору. У таких випадках має сенс накласти додаткові вимоги на відношення переваги так, щоб дія «кванта інформації» у звуженні множини Парето виявилася більш ефективною. Пізніше було встановлено, що вони представляють собою подальше посилення системи двох згадуваних раніше аксіом, що гарантують виконання принципу Еджворта-Парето.

Аксіома 1 (аксіома виключення).

Аксіома 2. Відношення $\succ$ визначено на усьому критеріальному просторі $\Re^m$ і транзитивнt на ньому.

Аксіома 3 (аксіома узгодження). Із двох векторів, що відрізняються один від одного єдиною компонентою, для ОПР переважніший вектор, що має більшу компоненту.

Аксіома 4 (аксіома інваріантності). Відношення переваги $\succ$ інваріантне відносно лінійного позитивного перетворення (тобто є лінійним). Нехай один критерій (або група критеріїв) важливіший за інший критерій (іншої групи критеріїв), якщо має місце деяка умова Ξ, яка містить певну інформацію про відношення переваги ОПР. Звідси ясно, що без визначення важливості критеріїв завжди можна обійтися, оперуючи в процесі прийняття рішень безпосередньо з умовою Ξ. Щоб скористатися визначенням важливості, що базується на «кванті інформації» і, яке використовує в якості умови Ξ співвідношення (1-3), спочатку слід пояснити ОПР це визначення важливості, переконатися, що вона його «засвоїла», після чого для виявлення переваг ОПР задати їй питання «мовою важливості»: чи є група критеріїв *A* важливішою за групу *B* з параметрами w_i і w_j (для $i \in$ A і $j \in$ B). Відомо, що бінарне відношення $\succ$, задане на векторному просторі $\Re^m$, називається конусним, якщо існує такий конус $K \subset \Re^m$, що співвідношення $y' \succ y''$, має місце тоді і тільки тоді, коли $y' - y'' \in K$.

Аксіома 5. Будь-яке бінарне відношення $\succ$, задане на векторному просторі $\Re^m$, і яке задовольняє аксіомам 2-4, є конусним з гострим опуклим конусом (без початку координат), який містить всі вектори з невід'ємними компонентами. Зворотно, всяке конусне відношення $\succ$ до зазначеного конусом, задовольняє аксіомам 2-4. Аксіома 5 відкриває можливість використання апарату опуклого аналізу і побудови змістовної математичної теорії для врахування різного набору «квантів інформації». Найбільш простий випадок одного «кванта» розглядається в наступному твердженні, доказ якого опирається на факти з теорії подвійності опуклого аналізу.

Аксіома 6. Нехай виконані аксіоми 2-4 і є «квант інформації» про відношення переваги $\succ$. Тоді для будь-якої множини вибираних варіантів *S(X)*, що задовольняє аксіомі 1, справедливі включення

$$S(X) \subset P_g(X) \subset P_f(X),$$

причому «новий» векторний критерій g може бути утворений з функцій f_i для всіх $i \in I \setminus B$:

$$g_{i,j} = w_j f_i + w_i f_j \text{ для всіх } i \in A, j \in B, \qquad (4)$$

або із функцій f_i для всіх $i \in I \setminus B$:

$$f_0 = \min_{i \in A}\left(f_i / w_i\right) + \min_{j \in B}\left(f_j / w_j\right). \qquad (5)$$

Важлива особливість аксіоми 6 полягає у відсутності будь-яких вимог до множини *X* і векторному критерію *f*: ці об'єкти можуть бути будь-якими. Обмеження накладаються лише на поведінку ОПР в процесі прийняття рішень та виражаються вони у формі аксіом 1-4.

Аксіомою 6 вказується оцінка зверху $P_g(X)$ для невідомої множини вибираних варіантів *S(X)*, більш точна, ніж множина Парето $P_f(X)$. Сама оцінка являє собою множину Парето-оптимальних варіантів, але відносно «нового» векторного критерію *g*.

Для того, щоб сформувати *g*, зі «старого» векторного критерію *f* слід видалити всі компоненти групи критеріїв *B* і додати один нелінійний критерій f_0 виду (5), або $|A| \cdot |B|$ «нових» лінійних критеріїв виду (4), де $|L|$ позначає число елементів кінцевої множини L.

Варіант з нелінійної функцією f_0 виду (5) можна застосовувати для кількісних критеріїв, значення яких вимірюються шкалою відношень, тоді як варіант (4) допускає використання ще й в шкалі інтервалів. Нелінійну функцію f_0 виду (5) можна використовувати для вивчення випадку, коли одна група критеріїв важливіша за іншу, де, на відміну від наведеної вище аксіоматики, використовується операція транзитивного замикання бінарного відношення і деякі інші припущення.

Як показали дослідження, врахування декількох «квантів інформації» повинне в більшій мірі сприяти звуженню множини Парето. Однак, може трапитися ситуація, коли ряд «квантів інформації» матиме суперечливий зміст, і їх використання буде неможливим. Тому важливою є задача вибору несуперечливих «квантів інформації». В рамках роботи несуперечливою названо таку множину, яка «породжує» іррефлексивне відношення. Побудова оцінки зверху, для невідомої множини вибираних векторів

$$S(Y) = f(S(X))$$

у вигляді множини

$$\overline{P}(Y) = f\left(P_g(X)\right)$$

за наявності довільного несуперечливого кінцевого набору «квантів інформації» у випадку кінцевої множини *Y*, зводиться до перевірки співвідношення

$$y' \succ_m y'' \qquad (6)$$

для всіх пар допустимих векторних оцінок $y', y'' \in Y$, де $\succ_m$ – бінарне відношення, яке будується на основі наявної несуперечливої множини «квантів інформації».

Таким чином отримала подальший розвиток методика структурної ідентифікації ризиків розробки ПЗ, що відрізняється від відомих побудовою оцінки ризиків розробки ПЗ «зверху» у вигляді множини за наявності довільного несуперечливого кінцевого набору «квантів інформації».

Використовуючи, наведену вище методику проведемо оцінку рангу ризиків розробки ПЗ. Після того, як ризики розробки ПЗ виявлені і включені до реєстру ризиків, виникає необхідність оцінки і визначення їх рангу окремо для кожної мети процесу/проекту (наприклад, для рамок функціональності, часу або інших ресурсів), і побудови матриці ймовірності і наслідків. Ранг ризику дозволяє оперативно управляти реагуванням на ризики, що розташовані в різних зонах матриці. Зони матриці грають роль пріоритетів.

Як було зазначено, на прийняття рішення про ранг ризику впливають пріоритети ОПР, сформовані багато в чому на основі експертних оцінок або результатів мозкового штурму (характерно для гнучких методик розробки ПЗ). З урахуванням цих факторів бла побудована матриця якісної оцінки рангу ризиків розробки ПЗ відповідно до даних і експертних оцінок фахівців ряду відомих фірм-виробників ПЗ.

У табл. 1. наведені результати якісної оцінки рангу ризиків розробки ПЗ. Слід зауважити, що зони матриці грають роль пріоритетів. Наприклад, для ризиків, розташованих в зоні високого ризику (виділено темно-сірим кольором і складають множину D) матриці, необхідні попереджувальні операції і агресивна стратегія реагування. Для загроз, розташованих в зоні низького ризику (виділено білим кольором і складають множину G), здійснення запобіжних операцій може не знадобитися, якщо тримати під контролем весь зміст виконуваної діяльності. У свою чергу множина загроз середнього ризику (виділені світло-сірим кольором (множина F)) вимагають обов'язкової стратегії управління та реагування.

Таблиця 1. **Результати якісної оцінки рангу ризиків розробки ПЗ**

Якісна оцінка ймовірності заподіяння шкоди	**Тяжкість наслідків при заподіянні шкоди**				
	Дуже висока	**Висока**	**Середня**	**Низька**	**Незначна**
Висока	***Id 1***	***Id 6, Id 7, Id 24***	***Id 15***	*Id 23, Id 27*	*Id 25*
Середня	***Id 19***	***Id 3, Id 10, Id 16***	*Id 4, Id 8*	*Id 21*	*Id 22*
Низька	***Id 9***	*Id 2, Id 17*	*Id 12, Id 14*	*Id 18*	*Id 28*
Мала	*Id 26*	*Id 11*	*Id 20, Id 29*	*Id 13*	*Id 5,*

Як видно з табл. 1 основна частина організаційних, операційних, управлінських та експлуатаційних ризиків знаходиться в «зафарбованій» зоні. Це говорить про важливість врахування цих ризиків (особливо в сучасних умовах застосування гнучких методологій розробки ПЗ).

Слід зауважити, що багато ризиків (наприклад, *Id 18*, *Id 20*), на початку певної активності можуть перебувати в зоні низького рангу, а ближче до відповідальних віх переміститися в приграничні або більш критичні зони. У той же час ряд існуючих ризиків незалежно від початкового рівня рангу можуть переміститися в більш «критичну» область (наприклад, *Id 23*, *Id 24* та ін.).

Таким чином, запропонований апарат ідентифікації та якісної оцінки рангу ризиків розробки ПЗ дозволяє до 17% звузити множину важливих ризиків і, відповідно, першочергових рішень управління.

Наступним етапом якісного аналізу ризику є процес документування.

Процес аналізу ризиків слід документувати протягом життєвого циклу всього проекту/процесу. Обсяг документування і його форма, що містить результати аналізу, залежить від конкретних цілей проведеного аналізу ризику.

Аналіз документації відомих фірм розробників ПЗ показав, що в підсумковому документі доцільно фіксувати такі дані: титульний аркуш; список учасників процесу якісного аналізу ризиків розробки ПЗ; анотацію; зміст; цілі та завдання проведеного якісного аналізу ризиків розробки ПЗ; опис аналізованого об'єкта; методологію якісного аналізу ризиків розробки ПЗ – початкові припущення і обмеження, що визначають межі аналізу ризику; опис використовуваних методів аналізу і обґрунтування їх застосування; початкові дані та їх джерела; результати ідентифікації; результати якісного аналізу ризику; аналіз невизначеностей результатів оцінки ризику; рекомендації по роботі з ризиками; висновок; список використаних джерел інформації.

Таким чином, в результаті проведених досліджень на основі класифікації та структурної ідентифікації ризиків розробки ПЗ розроблено метод якісного аналізу ризиків розробки програмного забезпечення. Відмінною особливістю розробленого методу є врахування факторів експлуатаційних ризиків, особливо ризику невиявлення вразливостей ПЗ та оцінка довільного несуперечливого кінцевого набору «квантів інформації». Це дозволить до 17% звузити множину важливих ризиків і знизити можливі фінансові та іміджеві втрати організацій-розробників ПЗ.

Тільки після того, як накопичено досвід і необхідний масив даних, від якісного аналізу ризиків доцільно переходити до їх кількісної оцінки. Причому концентрувати увагу слід саме на тих ризиках, які в процесі якісної класифікації були включені в категорію високих (особливо з високим ступенем шкоди при високій ймовірності реалізації).

Як вказано вище, для ефективного управління проектами потрібно не тільки ідентифікувати ризики, але й оцінювати їх кількісно. При цьому особливості сучасних фірм-розробників ПЗ як надсистем, особливості окремих етапів розробки ПЗ, бізнес-процесів та їх груп як підсистем, визначають ряд проблем. До цих проблем можна віднести: відсутність статистичних даних про вдалі і невдалі проекти впровадження систем, особливо на операційному рівні; відсутність статистичних даних про провали безпеки при експлуатації ПЗ; унікальність кожного проекту впровадження; довгостроковість подібних проектів; високу вартість подібних проектів; значну складову несистемних факторів ризику, пов'язаних з внутрішніми факторами фірми-розробника ПЗ.

З огляду на наведені фактори можна відзначити, що для оцінки ризиків розробки ПЗ можна використовувати три основні підходи: формалізований опис невизначеності ризиків розробки ПЗ; корегування показників проекту шляхом заміни їх проектних значень на очікувані; перевірку стійкості.

Формалізована оцінка невизначеності, яка виникає в процесі реалізації проектів при відсутності статистичних даних, може спиратися на два методи: експертних оцінок і нечітких множин.

Проведені дослідження показали, що використання суб'єктивно-аксіологічної імовірності (експер-

тних оцінок) є вимушеним відступом науки перед нарощуванням несистемних факторів ризику розробки ПЗ, але це вимагає подальшої верифікації моделі та обчислених показників ризику. У зв'язку з цим доцільним видається перехід від суб'єктивних експертних методів до методів, які використовують теорію нечітких множин. Корегування показників проекту (процесу розробки ПЗ) шляхом заміни їх проектних значень на значення з урахуванням ризиків викликає додаткові складності, пов'язані з невизначеністю всіх факторів, що впливають на фінансові, іміджеві та інші здобутки і втрати.

Для подолання цих проблем можна використовувати методи, які спираються на опис бізнес-процесів і дозволяють виявляти зміни окремих їх параметрів, пов'язаних з розробкою, впровадженням та експлуатацією ПЗ.

Проведені дослідження показали, що адекватним інструментом для таких досліджень є «Аналіз дерева відмов» (*Fault Tree Analysis, FTA*), запропонований в роботах [9, 10, 12-15].

Аналіз даного підходу кількісної оцінки ризиків показав доцільність використання графічної моделі *FTA* в термінах математичної логіки. Це допоможе формалізувати умови впливу факторів ризику в різних їх комбінаціях на кінцеві показники проекту розробки ПЗ.

Наведемо приклад дерева ризиків розробки ПЗ. Групи ризиків формуються з урахуванням розробленої класифікації ризиків розробки ПЗ і результатів якісної оцінки рангу ризиків (табл. 1).

Крім того, для наочності оцінки груп ризиків, пропонується використовувати логічні елементи «*and*» і «*or*», які дозволяють використовувати методи математичної логіки для розрахунку коефіцієнтів груп ризиків і загального ризику. Очевидно, що для цієї схеми загальний коефіцієнт ризиків розробки ПЗ можна розрахувати за формулою:

$$P(Un\,13) = \\ = P(Un\,2) + P(Un\,12) + P(Un\,6) + P(Un\,7) + \\ + P(Un\,8) + P(Un\,9) + P(Un\,10) + P(Un\,11), \qquad (7)$$

де $P(Un\,2) = P(Un\,1)\cdot P(Id\,2)$ – коефіцієнт ризику вибору неправильної методики розробки ПЗ;

$P(Un\,1) = P(Id\,1)\cdot P(Id\,3)$ – коефіцієнт ризику вибору неправильної методики розробки ПЗ;

$P(Un\,12) = P(Un\,3) + P(Un\,4) + P(Un\,5)$ – коефіцієнт ризику неадекватного менеджменту проекту;

$P(Un\,3) = P(Id\,4)\cdot P(Id\,9)$ – коефіцієнт ризику нехтування топ-менеджментом розробки ПЗ;

$P(Un\,4) = P(Id\,6)\cdot P(Id\,7)\cdot P(Id\,8)$ – коефіцієнт ризику неадекватного менеджменту активної стадії розробки ПЗ;

$P(Un\,5) = P(Id\,10) + P(Id\,26)$ – коефіцієнт ризику неадекватного соціального менеджменту;

$P(Un\,6) = P(Id\,11) + P(Id\,14) + P(Id\,15)$ – коефіцієнт ризику невідповідності професійного рівня;

$P(Un\,7) = P(Id\,16) + P(Id\,17) + P(Id\,18) + \\ + P(Id\,19) + P(Id\,20)$ – коефіцієнт технологічних ризиків розробки ПЗ;

$P(Un\,8) = P(Id\,21)\cdot P(Id\,22)$ – коефіцієнт ризику невідповідності складності ПЗ рівню підготовки експлуатанта;

$P(Un\,9) = P(Id\,23) + P(Id\,24)$ – коефіцієнт ризику безпеки експлуатації ПЗ;

$P(Un10) = P(Id\,12)\cdot P(Id\,25)\cdot P(Id\,27)$ – коефіцієнт ризику неадекватної комунікації;

$P(Un11) = P(Id\,28) + P(Id\,29)$ – коефіцієнт настання правових ризиків.

Наведемо значення коефіцієнтів для елементів дерева ризиків, що були отримані в результаті якісного аналізу ризиків розробки ПЗ: *Id* 1 (0,08), *Id* 2 (0,05), *Id* 3 (0,06), *Id* 4 (0,05), *Id* 5 (0,01), *Id* 6 (0,07), *Id* 7 (0,07), *Id* 8 (0,05), *Id* 9 (0,06), *Id* 10 (0,06), *Id* 11 (0,04), *Id* 12 (0,04), *Id* 13 (0,02), *Id* 14 (0,04), *Id* 15 (0,6), *Id* 16 (0,06), *Id* 17 (0,05), *Id* 18 (0,03), *Id* 19 (0,07), *Id* 20 (0,03), *Id* 21 (0,04), *Id* 22 (0,03), *Id* 23 (0,05), *Id* 24 (0,07), *Id* 25 (0,04), *Id* 26 (0,05), *Id* 27 (0,05), *Id* 28 (0,02), *Id* 29 (0,03).

Використовуючи вираз 7 і наведені вище значення коефіцієнтів, можна визначити, що загальний коефіцієнт ризиків розробки ПЗ $P(Un\,13) = 0{,}558$. Слід зауважити, що нехтування ризиком невиявлення вразливостей безпеки ПЗ (P(Id24))) може знизити точність кількісної оцінки ризиків розробки ПЗ до 13% ($P(Un\,13) = 0{,}488$), а нехтування коефіцієнтом ризику безпеки експлуатації ПЗ (P(Un9)) знизить точність оцінки до 22% P (Un13)) = 0,438). Це підтверджує необхідність врахування ризиків невиявлення вразливостей безпеки ПЗ і неадекватного взаємовпливу експлуатованого та впроваджуваного ПЗ.

Ще однією особливістю проектів розробки ПЗ є їх довгостроковість і необхідність врахування інформації, яка виникає на чергових стадіях прийняття рішень. Для вирішення цієї задачі в [16], наприклад, пропонується, використовуючи модульність програмного забезпечення, оцінювати їх інвестиційну привабливість з урахуванням розроблених компонент. Проведені дослідження дозволили знайти такий підхід імовірнісної оцінки ризиків, який дозволив використовувати основні положення нечіткомножинної теорії при оцінці ключових показників результативності проекту. Можливість його використання оцінимо на прикладі показника вартості C_{NPV} *(Net Present Value)*.

Аналіз ряду робіт, пов'язаних з економічною теорією ризиків показав, що для розрахунку показника вартості часто використовується класичний метод визначення чистої приведеної вартості проекту:

$$C_{NPV} = \sum_{i=1}^{n} \frac{D_i}{(1+r)^i} - \sum_{i=1}^{n} \frac{C_i}{(1+r)^i}, \qquad (8)$$

де n – кількість періодів реалізації проекту; D_i – фінансовий потік доходів від проекту в період *i*; C_i –

фінансовий потік витрат на проект в період *i*; r – ставка дисконтування.

Однак з урахуванням того, що проект розробки ПЗ є довгостроковим і залежить від багатьох факторів, існує необхідність розглядати показники повернення інвестицій в ці проекти спільно з процесами розвитку проекту. Крім цього важливо враховувати фактори динамічної зміни ризикових компонентів в процесі проектування, кодування, тестування та експлуатації на всіх «витках спіралі» розробки ПЗ.

Проведені дослідження показали, що з урахуванням додаткових факторів ризику, а також (8) для розрахунку C_{NPV} доцільно використовувати вираз:

$$C_{NPV} = \sum_{i=1}^{n} \frac{\left(B - AC_i\right)}{\left(1+r\right)^i} - \sum_{i=1}^{n} \left(\sum_{k=1}^{\ell} C_i^{(k)} \right) \Bigg/ \left(1+r\right)^i - \\ - \sum_{i=1}^{n} \left(\sum_{k=1}^{\ell} e^{-\lambda \times QC_i^{(k)}} \right) \Bigg/ \left(1+r_1\right)^i, \qquad (9)$$

де B – фінансові інвестиції, що надходять в процесі розробки ПЗ в період *i*; AC_i – поточні витрати на підтримку і розвиток системи розробки ПЗ в період *i*; l – кількість додаткових видів витрат на придбання, налаштування і модернізацію технічної, технологічної, програмної та інших складових в процесі розробки ПЗ; $QC_i^{(k)}$ – витрати на врахування безпеки і тестування вразливості ПЗ; λ – параметр впливу факторів безпеки і тестування вразливості ПЗ на чисту приведену вартість проекту; $C_i^{(k)}$ – додаткові витрати в процесі розробки ПЗ.

Відмінною особливістю математичного виразу (9) є введення додаткових складових

$$\sum_{i=1}^{n} \left(\sum_{k=1}^{\ell} C_i^{(k)} \right) \Bigg/ \left(1+r\right)^i ; \sum_{i=1}^{n} \left(\sum_{k=1}^{\ell} e^{-\lambda \times QC_i^{(k)}} \right) \Bigg/ \left(1+r_1\right)^i ,$$

що характеризують врахування додаткових витрат на апаратну і програмну модернізацію фірми, удосконалення системи її управління, а також врахування безпеки і тестування вразливості ПЗ.

Оцінка ефективності проекту полягає в порівнянні C_{NPV} з деяким значенням $C_{NPVreference}$, яке визначає мінімально допустимий рівень приведеної вартості проекту розробки ПЗ.

З урахуванням невизначеності складових, на основі теорії нечітких множин можна розглядати трикутні нечіткі значення складових для вираження (9): $\overline{\overline{B_i}} = \left(B_i^{(\min)}, \overline{B_i}, B_i^{(\max)} \right)$, якщо неможливо визначити фінансові доходи, які виникнуть від модифікації в процесі розробки ПЗ технологічних етапів; $\overline{\overline{AC_i}} = \left(AC_i^{(\min)}, \overline{AC_i}, AC_i^{(\max)} \right)$, якщо існує невизначеність витрат, необхідних для підтримки і розвитку системи розробки ПЗ; $\overline{\overline{C_i^{(k)}}} = \left(C_i^{(k)(\min)}, \overline{C_i^{(k)}}, C_i^{(k)(\max)} \right)$, якщо існує невизначеність щодо додаткових витрат *k*-го призначення.

Крім того, подібним же чином потрібно представити коефіцієнт дисконтування:

$$\overline{\overline{r_i}} = \left(r_i^{(\min)}, \overline{r_i}, r_i^{(\max)} \right),$$

якщо інвестор не може оцінити вартість капіталу, який буде використовуватись в проекті.

Для приведення формули (9) до виду, який може використовуватися для обчислень, скористаємося сегментним способом, наведеним в роботі [17].

Якщо вибрати фіксований рівень приналежності нечітких чисел α (ординату функції приналежності нечітких чисел), то можна застосувати операції нечіткої арифметики, які дозволять перетворити вираз (9) в систему рівнянь:

$$\left[C_{NVP1}, C_{NVP2}\right] = \\ = \begin{cases} \sum_{i=1}^{n} \frac{\left(B_{i_1} - AC_{i_1}\right)}{\left(1+r_1\right)^i} - \sum_{i=1}^{n} \frac{\left(\sum_{k=1}^{\ell} C_{i_1}^{(k)} \right)}{\left(1+r_1\right)^i} - \sum_{i=1}^{n} \frac{\left(\sum_{k=1}^{\ell} e^{-\lambda \times C_{i_3}^{(k)}} \right)}{\left(1+r_1\right)^i}; \\ \sum_{i=1}^{n} \frac{\left(B_{i_2} - AC_{i_2}\right)}{\left(1+r_2\right)^i} - \sum_{i=1}^{n} \frac{\left(\sum_{k=1}^{\ell} C_{i_2}^{(k)} \right)}{\left(1+r_2\right)^i} - \sum_{i=1}^{n} \frac{\left(\sum_{k=1}^{\ell} e^{-\lambda \times C_{i_3}^{(k)}} \right)}{\left(1+r_1\right)^i}; \end{cases} \qquad (10)$$

$$R_{NPV} = \begin{cases} 0, \quad C_{NVP} < C_{NVP}^{(\min)}; \\ R \cdot \left(1 + \frac{1-\alpha}{\alpha} \cdot \ell n\left(1-\alpha\right) \right) \\ \text{при} \quad C_{NVP}^{(\min)} \le C_{NVP} < C_{NVP}^{(av)}; \\ 1 - \left(1-R\right) \cdot \left(1 + \frac{1-\alpha}{\alpha} \cdot \ell n\left(1-\alpha\right) \right), \\ \text{при} \quad C_{NVP}^{(av)} \le C_{NVP} < C_{NVP}^{(\max)}; \\ 1, \quad C_{NVP} \ge C_{NVP}^{(\max)}, \end{cases} \qquad (11)$$

де $$R = \begin{cases} \frac{C_{NPV} - C_{NPV}^{(\min)}}{C_{NPV}^{(\max)} - C_{NPV}^{(\min)}}, C_{NVP} < C_{NVP}^{(\max)}; \\ 1, C_{NPV} \ge C_{NPV}^{(\max)}, \end{cases} \qquad (12)$$

$$\alpha = \begin{cases} 0, \qquad C_{NPV} < C_{NPV}^{(\min)}; \\ \frac{C_{NPV} - C_{NPV}^{(\min)}}{C_{NPV}^{(av)} - C_{NPV}^{(\min)}}, C_{NPV}^{(\min)} \le C_{NPV} < C_{NPV}^{(av)}; \\ \frac{C_{NPV}^{(\max)} - C_{NPV}}{C_{NPV}^{(\max)} - C_{NPV}^{(av)}}, C_{NPV}^{(av)} \le C_{NPV} < C_{NPV}^{(\max)}; \\ 1, \qquad C_{NPV} \ge C_{NPV}^{(\max)}; \end{cases} \qquad (13)$$

$$P_{NVP} =$$

$$= \begin{cases} \dfrac{\left(R_{NVP_{\min}}\right)^2}{\left(R_{NVP_{av}} - R_{NVP_{\min}}\right) \times \left(R_{NVP_{\max}} - R_{NVP_{\min}}\right)}, \\ \quad \text{якщо } R_{NVP_{av}} > 0; \\ 1 - \dfrac{\left(R_{NVP_{\max}}\right)^2}{\left(R_{NVP_{\max}} - R_{NVP_{av}}\right) \times \left(R_{NVP_{\max}} - R_{NVP_{\min}}\right)}, \\ \quad \text{якщо } R_{NVP_{av}} < 0. \end{cases} \quad (14)$$

$C_{NVP}^{(av)}$ – задає середній рівень нечіткого числа.

Дослідимо і проведемо оцінку ризиків розробки ПЗ за наступних умов: n=3; B_1=15; B_2=16; B_3=17; r=(0,1; 0,125; 0,15); AC_1=AC_2=AC_3=1; $\ell = 3$ (1 – витрати на придбання, налаштування та модернізацію програмного забезпечення в ході реалізації проекту, 2 – витрати на модернізацію апаратного забезпечення фірми та реорганізацію системи управління, яка необхідна після модернізації програмного забезпечення та устаткування, 3 – витрати на урахування безпеки ПЗ і додаткове тестування його вразливості); C_{i1} = =(1,1; 1,2; 1,3); C_{i2} = (0,5; 0,6; 0,7); C_{i3} = (0,1; 0,2; 0,3), λ=1, $C_{NPVreference}$ = 0. Як показали проведені дослідження, ведення додаткових витрат в процесі розробки ПЗ до 1,5 разів підвищує чисту приведену вартість проекту. Результати розрахунку R_{NPV} для рівнів приналежності α від 0,1 до 1 з кроком 0,1 можуть бути представлені в нечіткій трикутній формі. Ризик проекту розробки ПЗ визначається ймовірністю попадання точки в область, обмежену графіком нечіткого числа при R_{NPV} <0. Ця ймовірність визначається відношенням площі під цією частиною графіка до площі під всім графіком. Виходячи з цього, за допомогою формули визначимо рівень ризику для розглядуваного прикладу проекту розробки ПЗ. Для приведеного прикладу проекту розробки ПЗ рівень ризику складе 0,713. Таким чином, запропонований в підпункті 3.2 загальний спосіб оцінки показника чистої приведеної вартості проекту розробки ПЗ є засобом подолання проблем, пов'язаних з неефективним завершенням проектів розробки ПЗ. Спосіб пропонує розглядати проект комплексно, з урахуванням необхідності врахування безпеки і тестування вразливості ПЗ, з використанням інструментів, які дозволяють подолати складність, невизначеність і довгостроковість проектів. Для подолання проблем відсутності статистичних даних і підвищення достовірності експертної оцінки пропонується використовувати методи теорії нечітких множин.

З огляду на всі описані вище етапи кількісної оцінки ризиків розробки ПЗ можна відзначити, що відмінною рисою розробленого методу є комплексне використання методики «Аналізу дерева відмов» і способу оцінки показника чистої приведеної вартості проекту розробки ПЗ з урахуванням негативних чинників можливого невиявлення вразливостей безпеки ПЗ. Результати кількісної оцінки ризиків розробки ПЗ можуть бути використані в якості вхідних даних при управлінні ризиками безпосередньо на наступних етапах розробки ПЗ (прототипування, кодування, тестування і т.д.).

Висновки

В роботі визначено і вирішено одне з протиріч, що виникають при розробці ПЗ, яке полягає в нехтуванні фірмами-розробниками ПЗ факторів вразливості безпеки ПЗ. В якості вирішення зазначеної проблеми запропоновано використання розроблених методів якісного аналізу та кількісної оцінки ризиків розробки програмного забезпечення. В ході вирішення поставленої задачі на першому етапі розроблено метод якісного аналізу ризиків розробки програмного забезпечення. Його відмінною рисою є врахування факторів експлуатаційних ризиків, особливо ризику невиявлення вразливостей ПЗ та оцінка довільного несуперечливого кінцевого набору «квантів інформації». Це дозволить до 17% звузити множину важливих ризиків і знизити можливі фінансові та іміджеві втрати організацій-розробників ПЗ. Однією з основних складових методу є методика структурної ідентифікації ризиків розробки ПЗ, що відрізняється від відомих побудовою оцінки ризиків розробки ПЗ «зверху» у вигляді множини за наявності довільного несуперечливого кінцевого набору «квантів інформації». На другому етапі розроблено метод кількісної оцінки ризиків розробки ПЗ. Його відмінною рисою є комплексне використання методики «Аналізу дерева відмов» і способу оцінки показника чистої приведеної вартості проекту розробки ПЗ з урахуванням негативних факторів можливого невиявлення вразливостей безпеки ПЗ. Використання вдосконаленої методики «Аналізу дерева відмов» дозволить до 22% підвищити точність кількісної оцінки ризиків розробки ПЗ. У той же час використання способу оцінки показника чистої приведеної вартості проекту розробки ПЗ дозволяє розглядати проект комплексно, з урахуванням необхідності врахування безпеки і тестування вразливості ПЗ із залученням інструментів, які дозволяють подолати складність, невизначеність і довгостроковість проектів.

СПИСОК ЛІТЕРАТУРИ

1. Tom DeMarco, Timothy Lister, Waltzing with Bears. Managing Risk on Software Projects. Dorset House Publ/, 2003. 542 p.
2. Boehm B.W., Egyed A. A. Spiral model of software development and enhancement. IEEE Computer.1988. May. P. 61-72.
3. Sandra M. N., Sandra M.N., Carlos S. da S. Eduardo Risk management applied to software development projects in incubated technology-based companies: literature review, classification, and analysis. Gest. Prod., São Carlos. 2016, №23(4). P. 798-814.
4. Yet, B, Constantinou, A., Fenton, N., Neil, M., Luedeling, E., & Shepherd, K. A Bayesian Network Framework for Project Cost, Benefit and Risk Analysis with an Agricultural Development Case Study. Expert Systems with Applic. 2016. № 60. P. 141–155.
5. James J. J., Jamie Y. T. Chang, Houn-Gee Chen, Eric T. G. Wang, G. KleinJiang J. J. Achieving IT Program Goals with Integrative Conflict Management. Journal of Management Information Systems. 2014. № 31(1). P. 79-106.

6. Tavares B. G., C. Eduardo S. da Silva, A. D. de Souza. Risk Management in Scrum Projects: A Bibliometric Study. Journal of communications software and systems. 2017. № 13(1). P. 25-41.
7. Tomanek M., Juricek J. Project Risk Management Model Based on PRINCE2 and Scrum Frameworks/ M. Tomanek. The International Journal of Software Engineering & Applications (IJSEA). 2015. № 6(1). P. 81-88.
8. Power K. Impediment Impact Diagrams: Understanding the Impact of Impediments in Agile Teams and Organizations. Agile Conference (Orlando, Florida, 28 July - 1 August 2014,), Orlando, 2014. P. 18-32
9. Коваленко А.В., Смирнов А.А.,Якименко Н.Н., Доренский А.П. Проблемы анализа и оценки рисков информационной деятельности. Збірник наукових праць "Системи обробки інформації". 2016. №3(140). С. 40-42.
10. Коваленко А.В., Смирнов А.А.. Использование псевдобулевых методов бивалентного программирования для управления рисками разработки программного обеспечения. Системи управління, навігації та зв'язку.2016. №1 (37). С. 98-103.
11. Исикава К. Японские методы управления качеством/ под ред. А. В. Гличева. Москва: Экономика, 1988. 214 с.
12. Clifton A Ericson II Fault Tree Analysis Primer Create space Inc., Charlestone, NC. 2011. 136 p.
13. Guidelines and Methods for Conducting the Safety Assessment Process on Civil Airborne Systems and Equipment. 1996. 331 p.
14. Military Handbook "MIL-HDBK-338B" Electronic. Reliability Design Handbook. 1998. 1046 p.
15. Fault tree analysis (FTA). First edition. Geneve, 1990. 56 p.
16. Krishnan M. Soumya Software Development Risk Aspects and Success Frequency on Spiral and Agile Model. International Journal of Innovative Research in Computer and Communication Engineering. 2015. № 3(1). P. 301-310.
17. Araszkiewicz K. Building information modelling: an innovative way to manage risk in construction projects. International Journal of Contemporary Management. 2015. № 14(3). P. 23–40.

Рецензент: д-р техн. наук, проф. О. А. Смірнов,
Центральноукраїнський національний технічний університет, Кропивницький
Received (Надійшла) 28.03.2018
Accepted for publication (Прийнята до друку) 23.05.2018

Методы качественного анализа и количественной оценки рисков разработки программного обеспечения

А. В. Коваленко

В работе определено и решено одно из противоречий, возникающих при разработке ПО, которое заключается в пренебрежении фирмами-разработчиками ПО факторов уязвимости безопасности ПО. Все риски при разработке программного обеспечения, с большим или меньшим допуском, можно считать субъективным результатом выполнения процесса, который связан с недостатком количественной или качественной информации о процессе, а также ее неопределенностью. Указанные факторы можно считать главной причиной, порождает и сопровождает риски во всем их жизненном цикле. В качестве решения данной проблемы предложено использование разработанных методов анализа и количественной оценки рисков разработки программного обеспечения. Его отличительной особенностью является учет факторов эксплуатационных рисков, особенно риска выявления уязвимостей ПО и оценки произвольного непротиворечивого конечного набора «квантов информации». Доказано, что методика качественной оценки рисков проекта является описательной и представляет собой процесс, направленный на выявление конкретных рисков проекта, а также причин, их порождающих, с последующей оценкой возможных последствий и выработке мер для работы с рисками. В процессе анализа рисков происходит выработка метрик, отвечающих за определение предельных показателей факторов, которые сигнализируют о проявлении рисков. Разработано метод количественной оценки рисков разработки ПО. Его отличительной особенностью является комплексное использование методики «Анализа дерева отказов» и способа оценки показателя чистой приведенной стоимости проекта разработки ПО с учетом негативных факторов возможного выявления уязвимостей безопасности ПО. В то же время использование метода оценки показателя чистой приведенной стоимости проекта разработки ПО позволяет рассматривать проект комплексно, с учетом необходимости учета безопасности и тестирования уязвимости ПО с привлечением инструментов, которые позволяют преодолеть сложность, неопределенность и долгосрочность проектов.

Ключевые слова: оценка рисков, разработка программного обеспечения.

Quality analysis and quantitative assessment of risks methods of software development

O. Kovalenko

In the article, one of the contradictions arising in the development of software is defined and solved, which consists in neglecting software developers of software vulnerability vulnerabilities. All risks in the development of software, with more or less admission, can be considered a subjective result of the process, which is associated with a lack of quantitative or qualitative information about the process, as well as its uncertainty. These factors can be considered the main reason, generates and accompanies risks throughout their life cycle. As a solution to this problem, the use of developed methods of analysis and quantitative assessment of the risks of software development is suggested. Its distinctive feature is the consideration of operational risk factors, especially the risk of identifying software vulnerabilities and evaluating an arbitrary, consistent, finite set of "information quanta". It is proved that the method of qualitative risk assessment of the project is descriptive and represents a process aimed at identifying specific project risks, as well as the causes that generate them, with subsequent assessment of possible consequences and development of measures to deal with risks. In the process of risk analysis, the development of metrics that are responsible for determining the limiting indicators of factors that signal the manifestation of risks. A method for quantifying software development risks has been developed. Its distinctive feature is the integrated use of the "Failure Tree Analysis" methodology and a method for estimating the net present value of a software development project taking into account the negative factors of possible detection of software security vulnerabilities. At the same time, the use of the method of estimating the net present value of the software development project allows the project to be considered in a comprehensive manner, taking into account the need to take into account security and vulnerability testing of the software with the use of tools that overcome the complexity, uncertainty and long-term nature of projects.

Keywords: risk assessment, software development.