

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ КІБЕРБЕЗПЕКИ, КОМП'ЮТЕРНОЇ ТА ПРОГРАМНОЇ
ІНЖЕНЕРІЇ
КАФЕДРА ЗАСОБІВ ЗАХИСТУ ІНФОРМАЦІЇ**

ДОПУСТИТИ ДО ЗАХИСТУ
Завідувач кафедри ЗЗІ
_____ В.В. Козловський

«_____» _____ 2022 р.

КВАЛІФІКАЦІЙНА РОБОТА

**ЗДОБУВАЧА ОСВІТНЬОГО СТУПЕНЯ
«БАКАЛАВР»**

Тема: Удосконалення процесів проектування програмного забезпечення шляхом мінімізації ризиків

Автор: Ю.О. Кравчук

Науковий керівник: д.т.н., професор А.В. Міщенко

Нормоконтролер: д.т.н., професор М.О. Шутко

Київ 2022

НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ**Факультет:** Кібербезпеки, комп'ютерної та програмної інженерії**Кафедра:** Засобів захисту інформації**Освітнього ступеня:** «Бакалавр»**Спеціальність:** 125 Кібербезпека**Освітньо-професійна програма:** «Системи технічного захисту інформації, автоматизація її обробки»

ЗАТВЕРДЖУЮ

Завідувач кафедри ЗЗІ

_____ В.В. Козловський

«_____» _____ 2022 р.

ЗАВДАННЯ**на виконання кваліфікаційної роботи
студента Кравчука Юрія Олександровича**

1. Тема: Удосконалення процесів проектування програмного забезпечення шляхом мінімізації ризиків
затверджена наказом ректора від 06.05.2022 р. № 483/ст.
2. Термін виконання: з 16 травня 2022р. по 19 червня 2022р.
3. Вихідні дані: теоретичні засади удосконалення процесів проектування програмного забезпечення; шляхи мінімізації ризиків при проектуванні програмного забезпечення; впровадження удосконалення процесів проектування програмного забезпечення шляхом мінімізації
4. Зміст пояснювальної записки (перелік питань, що підлягають розробці):
 1. *Теоретичні засади удосконалення процесів проектування програмного забезпечення*
 2. *Шляхи мінімізації ризиків при проектуванні програмного забезпечення*
 3. *Впровадження удосконалення процесів проектування програмного забезпечення шляхом мінімізації*

КАЛЕНДАРНИЙ ПЛАН
виконання кваліфікаційної роботи

№ п/п	Етапи виконання кваліфікаційної роботи	Термін виконання етапів	Примітка
1.	Уточнення постановки задачі		Виконано
2.	Аналіз літературних джерел		Виконано
3.	Обґрунтування рішення		Виконано
4.	Збір інформації		Виконано
5.	Теоретичні засади удосконалення процесів проєктування програмного забезпечення		Виконано
6.	Шляхи мінімізації ризиків при проєктуванні програмного забезпечення		Виконано
7.	Впровадження удосконалення процесів проєктування програмного забезпечення шляхом мінімізації		Виконано
8.	Оформлення і друк пояснювальної записки		Виконано
9.	Оформлення презентації		Виконано
10.	Отримання рецензій від опонентів		Виконано
11.	Захист в ЕК		

Дипломник

(підпис, дата)

Ю.О. Кравчук

Дипломний керівник

(підпис, дата)

А.В. Міщенко

РЕФЕРАТ

Кваліфікаційна робота складається зі вступу, трьох розділів, загальних висновків, списку використаних джерел, загальний обсяг роботи складає 79 сторінок, має 14 рисунків, 5 таблиць. Список використаних джерел містить 19 найменування і займає 2 сторінки.

Метою є побудова та інтеграція моделі мінімізації ризикових збитків для вдосконалення процесів проектування ПЗ.

Ключові слова: ПРОЄКТНИЙ МЕНЕДЖМЕНТ, МЕНЕДЖЕР, УПРАВЛІННЯ, РИЗИК, РИЗИК МЕНЕДЖМЕНТ, ОПТИМІЗАЦІЯ, ПЛАН РЕАГУВАННЯ.

ЗМІСТ

СПИСОК УМОВНИХ СКОРОЧЕНЬ.....	6
ВСТУП.....	7
РОЗДІЛ 1 ТЕОРЕТИЧНІ ЗАСАДИ УДОСКОНАЛЕННЯ ПРОЦЕСІВ ПРОЄКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	8
1.1 Аналіз сучасних методологій проектування програмного забезпечення	8
1.2 Проблеми процесів проектуванні програмного забезпечення за різними методологіями	14
1.3 Ризик менеджмент як складова удосконалення процесу проектування програмного забезпечення.....	17
1.4 Висновки до першого розділу.....	20
РОЗДІЛ 2 ШЛЯХИ МІНІМІЗАЦІЇ РИЗИКІВ ПРИ ПРОЄКТУВАННІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	21
2.1 Сучасні методи ідентифікації ризиків.....	21
2.2 Методи управління ризиками	25
2.3 Висновки до другого розділу.....	31
Розділ 3 ВПРОВАДЖЕННЯ УДОСКОНАЛЕННЯ ПРОЦЕСІВ ПРОЄКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ШЛЯХОМ МІНІМІЗАЦІЇ РИЗИКІВ	32
3.1 Ідентифікація ризиків проєкту на підприємстві	32
3.2 Планування та впровадження заходів для зниження ризиків	37
3.3 Експериментальна перевірка ефективності запропонованих заходів мінімізації ризиків.....	43
3.4 Висновки до третього розділу	46
ВИСНОВКИ	47
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	49
ДОДАТОК А	51

СПИСОК УМОВНИХ СКОРОЧЕНЬ

ІОП	–	Ідентифікація основних причин
ПЗ	–	Програмне забезпечення
ПМ	–	Проектний менеджмент
ЧПД	–	Чистий приведений дохід
SDLC	–	Software Development Life Cycle
SWOT	–	Strengths, Weaknesses, Opportunities та Threats

ВСТУП

Актуальність. Впродовж останніх 30 років сформувалася нова культура управлінської діяльності – управління проєктами. Це відбулося, так як переважна більшість бізнесу є проєктно-орієнтованим (інноваційна, інвестиційна сфери, штучний інтелект, консалтинг, інжиніринг та інше). Для виробництв вищевказаного типу актуальними є професійне управління проєктами. Наразі існує проблема ефективного управління ризиками. Ключове завдання, котре реалізовано в рамках управління IT-проєктами, є управління ризиками проєктної діяльності чи управління ризиками проєкту.

Метою є побудова та інтеграція моделі мінімізації ризикових збитків для удосконалення процесів проєктування ПЗ.

У процесі підготовки кваліфікаційної роботи були поставлені наступні **задачі:**

1. Теоретичні засади удосконалення процесів проєктування програмного забезпечення
2. Шляхи мінімізації ризиків при проєктуванні програмного забезпечення
3. Впровадження удосконалення процесів проєктування програмного забезпечення шляхом мінімізації

Об'єкт дослідження. Процес проєктування ПЗ.

Предмет дослідження. Мінімізація ризикових збитків для вдосконалення процесів проєктування ПЗ.

Практична цінність. Необхідність інтеграції ризик-менеджменту в процес планування розробки ПЗ для ранньої ідентифікації та аналізу можливих проблем на проєкті.

Наукова новизна. Необхідність аналізу ризиків та інтеграції плану реагування на ризики згідно до властивостей проєкту, котрий зменшує вірогідність появи негативних факторів на проєкті та формування сприятливих умов на проєкті, котрі позначаються на продуктивності команди на краще.

РОЗДІЛ 1 ТЕОРЕТИЧНІ ЗАСАДИ УДОСКОНАЛЕННЯ ПРОЦЕСІВ ПРОЄКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

1.1 Аналіз сучасних методологій проєктування програмного забезпечення

Напрямок проєктного менеджменту (ПМ) бере початок з формування в останні десятиліття, загально визнаної методології здійснення проєктів та перетворення на невід'ємну частину ведення бізнесу та загальнолюдської культури. Керування проєктами можна вважати окремою економікою, котра розвивається областю знань та практики в управлінні [11].

Проєкт – це обмежена в часі, ресурсах та вимогах якості унікальна сукупність процесів, котра є направленою на досягнення унікальних цілей та завдань для створення нової цінності. Методологія управління проєктами - це методологія об'єднання, планування, керівництва, координації трудових, фінансових, і матеріально - технічних ресурсів, спрямованих на ефективне досягнення його цілей шляхом застосування сучасних методів, техніки і технологій керування для досягнення результатів за складом і обсягом робіт, вартості, часу і якості [11].

Більшість проєктів є складними з точки зору прийняття та виконання управлінських рішень. Успіх великих проєктів залежить від правильності стратегічних рішень, прийнятих на початкових стадіях, обґрунтованістю та оперативністю на наступних етапах реалізації. Інтеграція прогресивних підходів, методів та систем при управлінні проєктами підприємства є актуальними [11].

В умовах сучасної економіки, тоді як конкуренція у всіх сферах зростає, здається, до межі, а тривалість життя окремих товарів обчислюються місяцями чи навіть тижнями, використання технологій управління проєктами є ключовим не лише для процвітання, але і для виживання комерційних підприємств [11].

Підходи до розробки ПЗ визначають успіх проєкту, так як методології складно досягти стабільності у роботі продукту, безпеки та стійкості функціональних особливостей. Проєктний підхід до керування доводить власну ефективність на практиці та застосовується провідними світовими компаніями, його успішно застосовують IBM, Motorola, Boeing, Intel і інші компанії не залежно

від розміну [11].

ПЗ має власний життєвий цикл (Software Development Life Cycle, SDLC) – умовна схема, котра включає фази створення ПЗ. Цикл розробки пропонує шаблон, застосування якого полегшує проектування, формування та випуск якісного ПЗ. На виході має бути отриманий економічно вигідний продукт, котрий відповідає вимогам замовника [19].

Правильне планування функціональності майбутнього продукту та аналіз вимог грають ключову роль для всього проєкту [19]:

- Менеджер проєкту (відповідає за успіх всього процесу розробки). Важливо максимально точно та однозначно визначити вимоги до проєкту для команди проєкту та бізнесу.

- Аналітики визначають кінцеві цілі та завдання роботи.
- UX/UI дизайнери – фахівців, які проєктують інтерфейси (вивчення поведінки користувачів та вибудовування зрозумілого людині інтерфейсу, візуальний вигляд продукту

- Розробка – використовується одна з методологій, в основному Agile (розробники діють послідовно, розділяючи проєкт на більш дрібні завдання).

- Тестувальники – пошук помилок та готовність до релізу. Види тестування: модульне, інтеграційне, тестування інтерфейсу і інші.

- Супровід – доопрацювання системи з урахуванням змін середовища та відповідно до бажань користувачів, підтримка працездатності системи ПЗ.

- Передача ПЗ клієнту.

- Підтримка – виявлення аномалій, зміна робочого середовища, поява нових вимог.

Причиною закриття великої кількості програмних проєктів – застосування конкретних методів управління розробкою.

Модель розробки ПЗ описує, які стадії життєвого циклу проходять та дії на кожній з них.

Існують моделі розробки ПЗ [19]:

1. Code and fix – модель кодування і усунення помилок;

2. Waterfall Model – каскадна модель, або «водоспад»;
3. V-model – образна модель, розробка через тестування;
4. Incremental Model – інкрементна модель;
5. Iterative Model – ітеративна (або ітераційна) модель;
6. Spiral Model – спіральна модель;
7. Chaos model – модель хаосу;
8. Prototype Model – прототипна модель.

Популярними є: каскадний, V-образна, інкрементна, спіральна та гнучка [19]:

- Методологія Waterfall (каскадна модель) – формалізує структуру етапів розробки ПЗ, котра підтримує стратегію одноразового проходу етапів розробки ПЗ, ґрунтується на повному формулюванні вимог на початку життєвого циклу [19].

Рис. 1.1 наведені етапи розробки за використанням каскадної моделі.

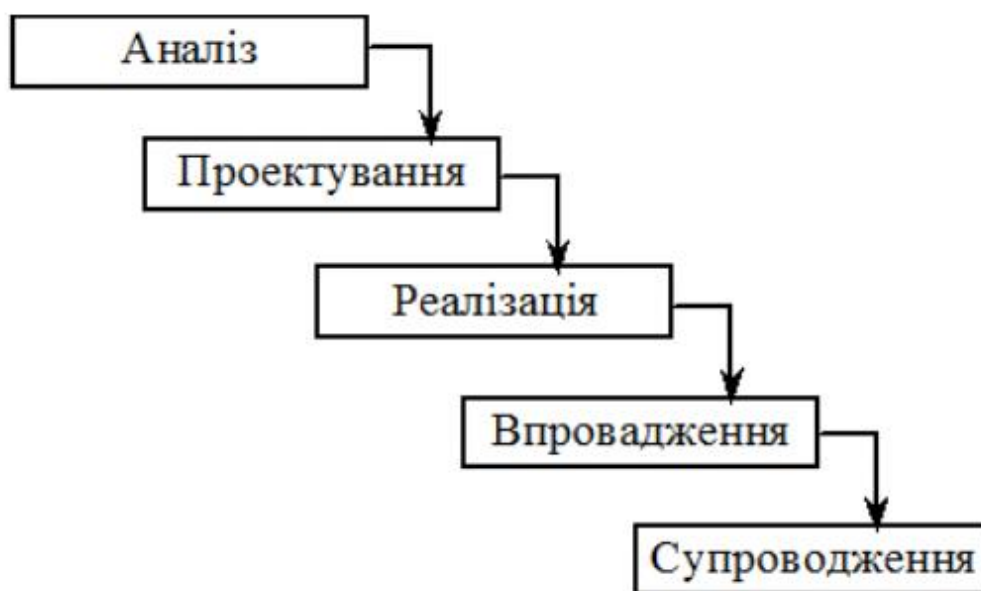


Рисунок 1.1 – Етапи каскадної моделі

Переваги Waterfall [19]:

1. Простота використання;
2. Структурність;
3. Документація.

Недоліки Waterfall [19]:

1. Проект залежить від конкретних виконавців;
2. Виконавці працюють за чітким планом;

3. Вартість та тривалість проєкту заздалегідь розраховують та затверджують, у процесі роботи їх не змінюють;

4. Вимоги не змінюються під час роботи.

- V-образна модель – вдосконалена каскадна модель, де замовник з командою розробників складають вимоги до системи та описують, яким чином здійснюється тестування на кожному етапі. Призначення V-подібної моделі – забезпечення планування тестування системи на ранніх стадіях проєкту. Рис. 1.2, етапи розробки за використанням V-подібної моделі [19].



Рисунок 1.2 – Етапи V-подібної моделі

Переваги V-подібної моделі [19]:

1. Кількість помилок в архітектурі ПЗ зводиться до мінімуму;
2. Велика роль приділяється верифікації та атестації ПЗ, починаючи з ранніх стадій його розробки;
3. Передбачається верифікація ПЗ, внутрішніх та зовнішніх даних;
4. Хід виконання робіт легко відстежується.

Недоліки V-подібної моделі [19]:

1. Недостатня гнучкість моделі;
2. Власне створення ПЗ здійснюється на етапі написання коду;
3. Недостатній аналіз ризиків;
4. Немає роботи з паралельними подіями та можливості динамічного

внесення змін.

- Інкрементна модель – метод, де проєкт формується, реалізується і тестується інкрементного до самого закінчення розробки. У випадку застосування інкрементної моделі життєвого циклу реалізується процес одноразової часткової реалізації системи. Рис. 1.3, етапи розробки за використанням інкрементної моделі [19].



Рисунок 1.3 – Етапи інкрементної моделі

Перевагами інкрементної моделі [19]:

1. Робоча програма виходить на ранній стадії життєвого циклу продукту;
2. Гнучкість;
3. Невеликі ітерації спрощують тестування і внесення правок;
4. Простіше ідентифікувати ризики;
5. Простота в управлінні контрольна точка проєкту;
6. Запобігання формуванню об'ємний вимог.

Недоліками інкрементної моделі є [19]:

1. Вимоги мають бути сформульовані заздалегідь;
2. Потрібне ретельне планування та проектування;
3. Поява тенденції до відтягування розв'язання проблем на майбутнє задля демонстрації керівництву успіху, який досягнуто на ранніх етапах розробки.

- Спіральна модель – кожен цикл розробки є набором операцій, який відповідає крокам у каскадній моделі, де враховується кожен компонент продукту

чи системи та кожен рівень складності, починаючи із загального аналізу вимог і до програмування кожного компонента. Рис. 1.4, основні етапи спіральної моделі [19].

Переваги спіральної методології [19]:

1. Найкращий аналіз ризиків;
2. Хороша документація процесу розробки;
3. Гнучкість;
4. Раннє створення робочих прототипів.

Негативні риси спіральної моделі [19]:

1. Може бути досить дорогою у використанні;
2. Управління ризиками вимагає залучення висококласних фахівців;
3. Успіх процесу великою мірою залежить від стадії аналізу ризиків;
4. Не підходить для невеликих проєктів.



Рисунок 1.4 – Етапи спіральної моделі

- Методологія управління з акцентом на розробці ПЗ – застосування швидкого та гнучкого підходу (Рис. 1.5) [19].



Рисунок 1.5 – Етапи Agile моделі

Переваги Agile методології є [19]:

1. Гнучкість і свобода;
2. Знижений ризик.

Недоліки Agile методології є [19]:

1. Слабка увага до документації;
2. Проблеми із реалізацією комплексних продуктів;
3. Імовірність відмови клієнтів у процесі розробки;
4. Вплив ієрархії.

Є велика кількість методологій управління проектами, де кожна з яких має свої завдання та можливості, котрі допомагають при реалізації успішних проектів із високою цінністю. Ефективність одного обраного Вами методу варіюється у залежності від галузі та конкретного проекту. Їх сильні сторони та переваги працюють найкраще, якщо застосувати їх правильно [19].

1.2 Проблеми процесів проектуванні програмного забезпечення за різними методологіями

Є велика кількість різноманітних методологій для побудови процесу розробки ПЗ, і кожна з них має свої переваги та недоліки, сфери застосування, в яких певні з них найбільш ефективні. Вищенаведені методології мають на меті поліпшення виробничого процесу, котрі дозволяють ефективно та якісно розробляти програмні продукти. Однак, варто враховувати, що при розробці ПЗ за

будь-якої методології можуть виникнути проблеми, з якими зіштовхнеться команда у процесі роботи. Важливо заздалегідь знати про ці проблеми, спираючись на них при виборі методології для конкретного проєкту. Це допоможе обрати оптимальний варіант методології розробки ПЗ [19].

Проблеми у процесі розробки ПЗ при застосуванні [19]:

- Waterfall.

Виявлення помилок чи внесення змін, так як доведеться починати проєкт спочатку, це великий ризик. Складність у завершенні проєкту вчасно. Висока ймовірність виявлення критичних проблем на завершальному етапі розробки, і при цьому будуть наслідки досить вартісні.

- V-образною методологією.

Основою моделі є послідовна лінійна структура, призводить до того, що кожна спроба повернутися на одну або дві фази назад. Тобто, при не виявленні проблеми вчасно, призводить до збільшення витрат та збою в графіку. Для кожної фази формуються результативні дані, котрі після завершення вважаються замороженими. Важливо, що модель не розрахована на динамічні зміни в вимогах протягом усього життєвого циклу. Надмірна кількість документації, що призводить для користувачів, що вони не можуть переконатися як розроблений продукт до закінчення всього процесу розробки. Навчання здійснюється після введення в експлуатацію.

- Інкрементною методологією.

Необхідність чітко визначених інтерфейсів між модулями, то проблема пов'язана з різними термінами їх формування та складністю формального аналізу, перевірки окремих інкрементів. Можлива поява непередбачених ітерацій в рамках кожного інкременту моделі, є можливість змін у технологіях робіт, які порушують графік робіт.

- Спіральною методологією.

Визначення моменту переходу наступного етапу, для її вирішення впроваджують тимчасові обмеження на кожен із етапів життєвого циклу. Тобто, процес розробки перетворюється на постійне вдосконалення вже розробленого.

Висока вартість моделі за рахунок вартості та додаткових часових витрат на планування, визначення цілей, виконання аналізу ризиків та прототипування під час проходження кожного циклу спіралі.

- Agile.

Неможливість кількісно оцінити необхідні зусилля, особливо на початку життєвого циклу розробки об'ємного ПЗ. Вимоги до ПЗ уточнюються у процесі розробки, документація не надто докладна. Не вимагається детальне планування для початку роботи та передбачає, що потреби замовника постійно змінюються. У такому випадку, із-за недостатніх вихідних даних обмежується використання Agile [19].

Здійснивши аналіз проблем кожної з моделей розробки ПЗ, робимо висновок, що застосування будь-якої з моделей розробки ПЗ принципово залежить від характеру вимог, які стоять перед системою, яка проектується [19].

При застосуванні низки моделей успіх реалізації проєкту залежить від ступеня злагодженості роботи єдиної команди розробників та користувачів, то вже на початкових етапах роботи, обираючи модель життєвого циклу, слід отримати повне уявлення про характеристики колективу користувачів як комплексний фактор, що впливає на вибір моделі [19].

При виборі моделі життєвого циклу відіграється тип передбачуваного проєкту та ризику, пов'язані із застосуванням проєкту. Процедура вибору моделі життєвого циклу має здійснюватися з урахуванням розгляду окремих критеріїв та комплексу [19].

У табл. 1.1 наведена застосування різних моделей розробки ПЗ у залежності від характеристик набору вимог до проєктованої системи [4].

За прибутком ключові проблеми розробки ПЗ йдуть на другий план – терміни перемагають у боротьбі за високу якість програмних рішень. Однією зі складових вдосконалення процесу проєктування ПЗ є ризик-менеджмент, котрий надає можливість спланувати роботу в команді, ґрунтуючись на передбачуваних ризиках та контролювати процес розробки, ґрунтуючись на раніше виявлених проблемах. Дану процедуру найчастіше виконує менеджер проєкту, і це є важливою складовою

процесу розробки ПЗ [4].

Таблиця 1.1 – Характеристика набору вимог моделей розробки ПЗ

Характеристика набору вимог	Модель				
	Каскадна	V-подібна	Інкрементна	Спіральна	Agile
Чи є вимоги легко визначними та/або добре відомими?	Так	Так	Ні	Ні	Ні
Чи можуть бути вимоги наперед визначені?	Так	Так	Так	Ні	Так
Чи часто змінюватимуться вимоги?	Ні	Ні	Ні	Так	Так
Чи потрібно демонструвати вимоги з метою їхнього визначення?	Ні	Ні	Ні	Так	Так
Чи потрібна для демонстрації можливостей перевірка концепції?	Ні	Ні	Ні	Так	Так
Чи вимоги відобразатимуть складність системи?	Ні	Ні	Так	Так	Ні
Чи відображають вимоги ранньому етапі функціональні	Ні	Ні	Так	Так	Так

1.3 Ризик-менеджмент як складова удосконалення процесу проєктування програмного забезпечення

В інженерії ПЗ є ризики у процесі розробки незалежно від того, як наскільки проєкт матеріально залежний та технічно забезпечений, нормативно-правовими актами та професійними фахівцями [4].

Управління ризиками – невід’ємна складова ефективного управління будь-яким програмним проєктом, керівник має ефективно планувати, керувати та здійснювати контроль етапами виконання завдань проєкту [4].

Більшість ІТ-компанії стають конкурентоспроможними лише за рахунок

інноваційної діяльності, котра пов'язана з різними ризиками, а саме ймовірністю появи збитків чи недоотримання прибутків у порівнянні з прогнозованими. Ризик є одночасно як причиною можливих збитків, так і джерелом потенційних прибутків. Ризик – невід'ємна складова процесу керування, неврахування якого призводить до вироблення, аналізу та прийняття необґрунтованих та малоефективних управлінських рішень. Ключовим завданням управління ризиками – не відмовитись від ризику як такого взагалі, а приймати ризикові рішення, ґрунтуючись на об'єктивних критеріях та допустимих втратах. Прийняті керівником проекту ризик-орієнтовані рішення часто призводять до більш ефективної реалізації програмних проєктів, котрі отримують свою вигоду як замовники та розробники ПЗ, так користувачі [4].

Ризики появи потенційних загроз та небезпек існують практично в усіх програмних проєктах, однак не завжди вони відбуваються та наносять шкоду ПЗ. Прогнозування потенційних загроз та небезпек у багатьох випадках – передбачення появи певних ризикових подій, котрі мають негативно вплинути на хід реалізації ПЗ та на його остаточні результати – продукти проєкту. Рис. 1.6 відображається план управління ризиками. Ризики розробки ПЗ поділяється на кілька категорій, у кожному випадку додаються й інші типи ризиків [4].

Основні категорій ризиків відповідно до джерела належать [4]:

- Технологічні ризики, пов'язані з незнанням технологій, котрі плануються застосовувати працівниками для розробки ПЗ, або з низькою апробацією та відпрацюванні технологій в колективі виконавців проєкту.
- Ризики наявних суперечностей у вимогах до ПЗ, пов'язується з виявленням суперечностей у вимогах замовника на етапі декомпозиції користувацьких вимог у системні вимоги чи на етапі конструювання програмного коду чи інтеграції продуктів проєкту.
- Ризики, пов'язуються з низькою кваліфікацією персоналу.
- Ризики поганої взаємодії між замовником ПЗ та його виконавця, які пов'язані з відсутністю комунікації між їхніми керівниками чи їх представниками.



Рисунок 1.6 – Управління ризиками проєкту

- Ризики неефективного планування етапів реалізації проєкту.
- Ризики недостатності системи контролю за реалізацією завдань проєкту.
- Ризики появи нових користувацьких вимог до ПЗ.
- Ризики наявних суперечностей у вимогах до ПЗ.
- Ризики неправильно сформульованих системних вимог.
- Ризики нездатності команди впоратися зі складністю розробки ПЗ.
- Ризик низької продуктивності праці команди розробників ПЗ.
- Ризик зміни працівників програмного проєкту.
- Ризики розкрадання продуктів проєкту.
- Ризики порушення закону про авторське право.
- Ризики розробки ПЗ віднести ризики фінансових обмежень.

Рис. 1.7, класифікація ризиків проєкту. У період активного розвитку інформаційних технологій застосовуються різні інформаційні системи, набувають масового характеру. Некеровані системи спричиняють серйозні ризики [4].



Рисунок 1.7 – Класифікація ризиків

1.4 Висновки до першого розділу

У першому розділі кваліфікаційної роботи було проаналізовано сучасні підходи до проектування ПЗ. Виявляються переваги та недоліки кожного з цих методів. Всі методології мають загальні проблеми при проектуванні ПЗ, а саме підвищений ризику, які можуть знадобитися для внесення змін, необхідність у жорсткому управлінні та контролі, проблеми щодо проектування архітектури системи, підвищення вартості проєкту за рахунок додаткових часів на планування, прототипування, документування та визначення цілей проєкту.

Розглянуто сутність поняття ризик, досліджено його класифікацію, виділено зовнішні та внутрішні, технічні та нетехнічні, передбачувані та непередбачувані, юридичні, управлінські, фінансові, маркетингові та ризики персоналу.

РОЗДІЛ 2 ШЛЯХИ МІНІМІЗАЦІЇ РИЗИКІВ ПРИ ПРОЄКТУВАННІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

2.1 Сучасні методи ідентифікації ризиків

Для оцінки ризику та прийняття пов'язаних рішень варто зібрати вихідну інформацію щодо суб'єкта, який несе ризик. Основний етап має назву – ідентифікацією ризику, котра збирає інформацію щодо структури об'єкта та виявляє небезпеки [9].

Ідентифікація – визначає ризики, які ґрунтуються на визначенні факторів, які їх продукують, документальне оформлення параметрів цих ризиків. Якісний та кількісний аналіз виявлення причин та можливості негативних наслідків, які є необхідними для формування оціночної процедури. Планування реагування на знайдені ризики передбачає створення комплексу заходів, які направлені на зниження негативного впливу ризиків на параметри та результати проєкту. Однак, високе місце у даній системі займає моніторинг ризиків та контроль над ними, який здійснюється впродовж всього життєвого циклу проєкту [9].

Є велика кількість способів, котрі допомагають отримати інформацію щодо характеристик індивідуальних ризиків, які притаманні тому чи іншому виду діяльності. Для вирішення даної проблеми рекомендується застосувати набір методів. Задokumentовані ризики та їх характеристики, відображаються в реєстрі ризиків, дозволяють досліджувати причини та можливі суміжні ризики, котрі можуть вплинути та взаємозамінювати один одного у залежності від параметрів оточення діяльності. Розмір реєстру ризиків залежить від масштабу діяльності, однак якщо організація має на меті підвищити якість своєї діяльності та створення якісного продукту чи послуги, то реєстр ризиків має бути максимально повним, незалежно від ймовірності та значення подій ризику [9].

Управляти всіма можливими ризиками є мало ймовірним, так як це потребує великих фінансових та кадрових витрат. Обов'язкова умова для реєстру ризиків – це наявність у ньому пріоритетів [9].

Пріоритет ризику – це параметр, який ідентифікується важливими та значущими з них на даний момент часу. Важливість коректного визначення

пріоритетів ризиків, може забезпечити надійні «тили» значущих процесів та проєктів, які проводять в організації [9].

Важлива частина організації діяльності у даній сфері є створення спеціальних програм з керування та виявлення нових ризиків. Є безліч способів класифікації, відповідно, ідентифікації ризиків проєкту [9].

На вибір конкретного методу чи комплексу методів роботи з ризиками впливають різні фактори, такі як доступність кваліфікованих ресурсів, характер та ступінь невизначеності даних та інформації, складність методу його застосування. Перед тим, як обрати метод, слід провести дослідження, метою якого є обґрунтування вибору конкретних методів ідентифікації ризику із зазначенням їх прийнятності, придатності та застосування у заданих умовах функціонування. Варто забезпечити відповідність застосовуваних методів та вихідних даних для об'єднання результатів різних аналізів при роботі над великими проєктами [9].

1. Brainstorming

Мозковий штурм – простий та поширений метод ідентифікації, котрий існує досить давно. Мета мозкового штурму – це створення докладного списку всіх можливих ризиків проєкту чи процесу. Проводяться збори, які присвячені мозговому штурму, не розробляється реєстр ризиків. Мета зборів – розробка список ризиків. У зборах може брати участь від 5 до 12 осіб. Учасники зборів виявляють та ідентифікують потенційні ризики, котрі не допускаються до обговорення висунутих ідей, група експертів озвучує всі ідеї, надалі мають бути детально та ретельно проаналізовані, структуровані з подальшим відображенням ризиків [9].

Плюси методу – відносна швидкість, реалізованість, легка досяжність кінцевого результату. Недолік порівняно з методом мозкового штурму – це нижча соціальна орієнтованість методики карток кроуфорду. Методики є життєздатними та мають практичну значимість і цінність [9].

2. Метод Delphi

Розроблений у процесі розпалу холодної війни у США, групою експертів, котрі представляють одну з корпорацій, які працюють на урядові структури. Даний метод надає можливість проаналізувати ризики кілька разів, систематизувати їх.

Метод Delphi виключає тиск з боку колег та страх незручного становища при висловленні ідеї. Основні переваги способу є виключення можливості домінування однієї особи, спосіб проводиться дистанційно. Недоліки методу – вимагають участі кожного члена групи, який займає багато часу, високе навантаження лягає на плечі ведучого та адміністративний персонал. Рис. 2.1, продемонстрована процедура ідентифікації ризиків за методом Delphi [9].



Рисунок 2.1 – Метод Delphi

3. Ідентифікація основних причин

Ідентифікація основних причин (ІОП) – це метод, який має чітко сформульований алгоритм, який використовується при ідентифікації ризиків [9].

Метод полягає у докладному розгляді потенційних ризиків, котрі є наслідком певної діяльності та побудовою причинно-наслідкових зв'язків. Завдяки зафіксованих закономірностей можна виявити основні та головні причини ризиків, галузі та активності, де вони з'являються, враховуючи різноманітні суміжні процеси, котрі впливають на появу ризиків. Причини виявлені та зафіксовані, далі ухвалюють рішення про те, що і як слід коригувати [9].

Перевага методу ідентифікації є можливість його проведення без додаткового залучення дорогого ресурсу додаткових експертів, окремо взятим фахівцем, що робить цей метод менш «експертним» у порівнянні з розглянутими раніше методами, однак є менш ефективним та швидким [9].

Недоліки – наявність певної документарної бази, котра буде аналіз ідентифікації та виявлення ризиків [9].

4. SWOT аналіз

SWOT (Strengths, Weaknesses, Opportunities та Threats), мета – оцінити можливість та оточення «ризикового» проєкту чи процесу. Дана методика набула широкого поширення серед різноманітних галузей бізнесу при проведенні консалтингових та управлінських досліджень за рахунок своєї привабливої суб'єктивності та легко інтерпретованості результатів, виконаних конкретними експертами [9].

Переваги і недоліки – це чинники внутрішнього середовища процесу чи проєкту, які впливають на появу ризиків. Можливість та загрози – це фактори зовнішнього середовища, які впливають на об'єкт та призводять до можливої появи ризиків. Рис. 2.2 наведено принцип SWOT аналізу [9].



Рисунок 2.2 – Принцип SWOT аналізу

5. Метод номінальних груп

Метод номінальної групи – це груповий процес, який включає ідентифікацію проблеми, генерація рішень та прийняття рішень. Їх використовують в групах різного розміру, котрі швидко ухвалюють рішення. Кожен член групи пропонує власне бачення рішення з короткими поясненнями, далі видаляються зі списку всіх

рішень та учасники переходять до ранжування рішень: 1, 2, 3, 4 і так далі [9].

6. Метод експертних оцінок

Метод застосовується у випадках відсутності чи недостатнього обсягу інформації. Він забезпечує виділення певних груп ризиків та отримання оцінки ступеня ризику на підставі знань спеціалістів і науковців, їх вміння узагальнювати власний та світовий досвід досліджень з певної проблематики [9].

За допомогою методів експертних оцінок розв'язують задачі якісного аналізу ризиків, так і кількісного. Розрізняють методи індивідуальних та колективних експертних оцінок. Кожен з методів має свої особливості, однак спільними для всіх є процедури, котрі забезпечують їх реалізацію (вибір експертної групи, проведення експертизи, оброблення експертної інформації, верифікація результатів експертизи) [9].

Основна перевага методу експертних оцінок полягає в можливості застосовувати досвід експертів для аналізу проєкту та вплив різноманітних якісних чинників [9].

Перевага експертного аналізу – відсутність необхідності в точних початкових даних та дорогих ПЗ, здійснення оцінки до розрахунку ефективності проєкту, простота розрахунку [9].

Недоліки – складність залучення незалежних експертів та суб'єктивність оцінки. Порівняльна характеристика методів ідентифікації ризиків наведена у табл. 2.1 [9].

2.2 Методи управління ризиками

Після розпізнавання ризику слід проаналізувати, для вибору правильного плану реагування на ризик. Якісний аналіз ризиків – процес подачі якісного аналізу ідентифікації ризиків та визначення ризиків, які вимагають швидкого реагування [15].

Оцінка ризиків визначає ступінь важливості ризику та обирає спосіб реагування. Доступність документації надає легше розставити пріоритети для різних категорій ризиків [15].

Таблиця 2.1 – Порівняльна характеристика методів ідентифікації ризиків

Метод ідентифікації	Переваги	Недоліки
Мозковий штурм	Сприяє взаємодії усієї команди розробки. Швидкий. Недорогий.	Може проявитися переважання одного члена команди, наприклад ліда чи менеджера проекту. Можна зосереджуватися тільки в конкретних модулях програми. Можливий відхід від реальних ризиків
Метод Delphi	Може проводитися дистанційно (через електронну пошту або в онлайн зв'язках), що зараз дуже актуально. Виключається проблема ранньої оцінки. Вимагає участі кожного члена команди розробки.	Займає багато часу. Високе завантаження менеджера проекту чи бізнес аналітика.
Метод номінальних груп	Зменшується ефект домінуючою особистості. Забезпечує взаємодію команди. Дає упорядкований список ризиків що можуть виникнути під час розробки.	Високе завантаження менеджера проекту чи бізнес аналітика.
Картки Кроуфорда	Швидкий. Легко реалізується. Повинен брати участь кожен у команді. Виробляється велика кількість ідей по реалізації. Можна проводити з групами більше звичайного розміру.	Потребує багато часу.
Картки Кроуфорда	Швидкий. Легко реалізується. Повинен брати участь кожен у команді. Виробляється велика кількість ідей по реалізації. Можна проводити з групами більше звичайного розміру.	Менше взаємодія між командою розробки програмного забезпечення.
Метод експертних оцінок	Використовується минулий досвід розробки проекту та проблеми які вже виникали.	Експерт може бути упередженим. Потребує багато часу. Може бути затратно фінансово через те що експерт - людина з досвідом, витрачає час на цю роботу, а час роботи дуже високооплатний.
Контрольні списки	Конкретний і упорядкований Легко використовувати упередженість	Може не містити конкретних елементів для даного проекту
Метод аналогії	Використовує минулий досвід для виключення проблем в майбутньому подібні проекти містять багато схожих рис	Потребує багато часу. легко отримати результати, яких не слід цього випадку. Аналогія може бути некоректної
Метод SWOT аналізу	Дозволяє виявити всі фактори, що впливають на організацію, а саме дозволяє узагальнити і зіставити дані абсолютно різного типу і призначення.	Відсутність динаміки в часі

Завдання якісного аналізу ризиків полягає у [15]:

1. Виявлення та ідентифікація можливих видів ризиків;
2. Дослідження причин появи ідентифікованих ризиків та наслідків їх дій;
3. Встановлення потенційних меж окремих видів ризиків;
4. Надання вартісної оцінки потенційних втрат від прояву ризиків;
5. Розробка системи заходів для зменшення та уникнення ризиків.

Також існує кількісний аналіз ризиків, котрий визначає ймовірність їх появи та вплив наслідків ризиків на проєкт, які допомагають групі менеджменту проєкту правильно приймати рішення, уникати невизначеностей. Кількісний аналіз передбачає визначення окремих ризиків та ризику проєкту в цілому у конкретних числових показниках [15].

Кількісний аналіз проєктних ризиків передбачає [15]:

1. Вибір системи показників для оцінювання ризиків;
2. Обґрунтування та вибір методів кількісного оцінювання ризиків;
3. Формування інформаційної бази для кількісного аналізу ризиків;
4. Побудову економіко-математичних моделей для оцінок альтернативних варіантів рішень;
5. Вибір підмножини пріоритетних рішень.

У процесі кількісного аналізу ризику встановлюється співвідношення між ступенем ризику і очікуваним економічним результатом. На ступінь допустимого ризику впливають параметри економічної системи, де в момент обґрунтування рішення вони відомі, а також величина очікуваних збитків чи прибутків. Рис. 2.3, узагальнена блок-схема комплексної оцінки ризиків [15].

Існує велика кількість методів для аналізу ризиків [15]:

1. Визначення параметрів граничного рівня.

Застосовується для визначення групи параметрів, які характеризують ступінь стійкості проєкту щодо можливих змін його реалізації. Граничним є значення параметра, де чистий прибуток = нулю. Основний показник групи – точка беззбитковості. Значення точки беззбитковості обов'язково розраховується у процесі формування нової організації, модернізації виробничих потужностей, інтеграція створення нового виду продукції. Значення точки беззбитковості

розраховується у процесі формування нової організації, модернізації виробничих потужностей, інтеграція на створення нового виду продукції. Мета розрахунку точки безбитковості – знаходження обсягів реалізації, які необхідні для відшкодування витрат, основа для аналізу безбитковості є дані бухгалтерської звітності.

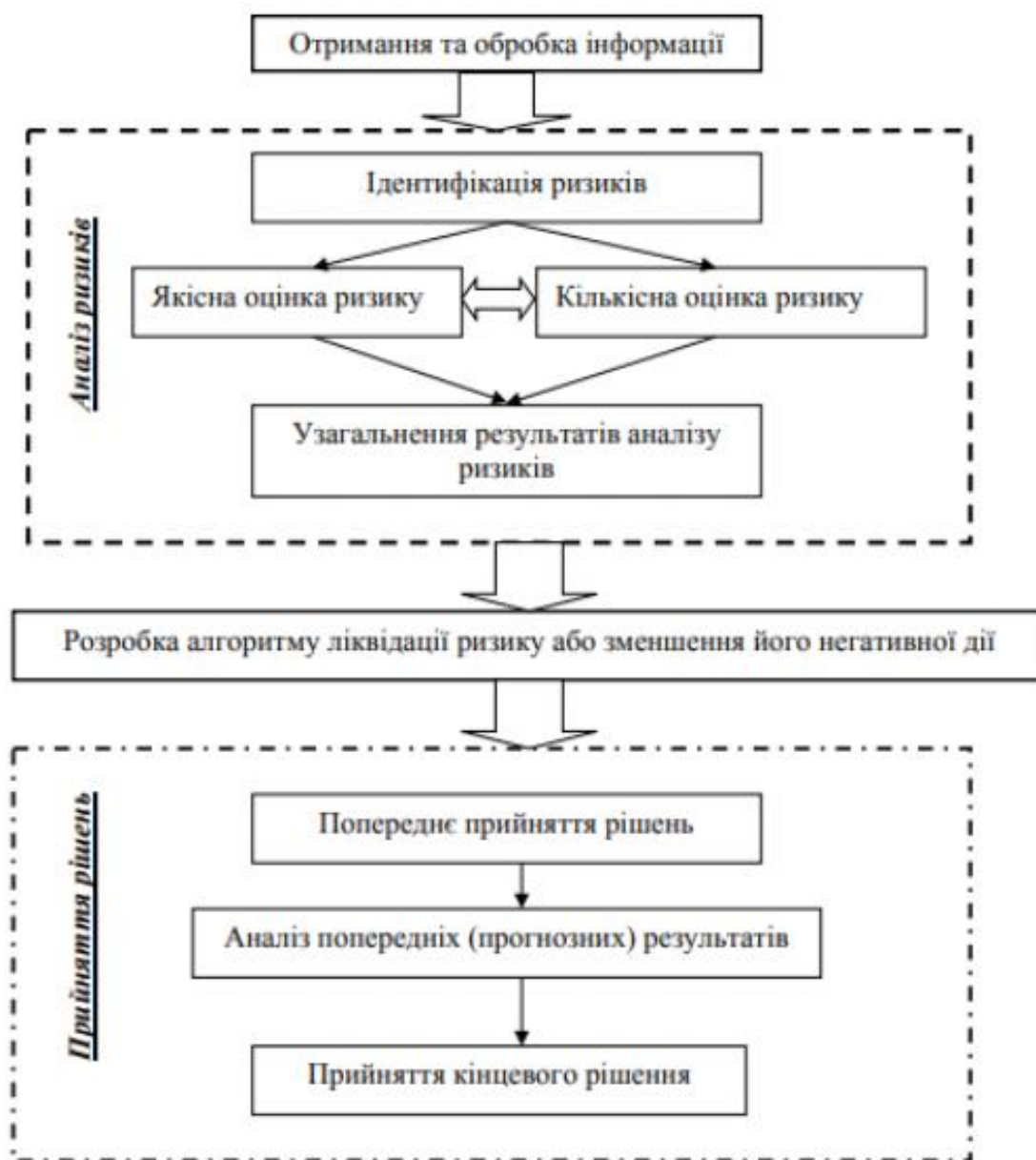


Рисунок 2.3 – Блок-схема комплексної оцінки ризиків

2. Оптимізаційні методи.

Забезпечення пошуку найкращих варіантів для функціонування економічних систем та процесів при раціональному застосуванні обмежених ресурсів. Показники функції мети обсяги виробництва та реалізації продукції, витрати на

виробництво, прибуток, рівень задоволення потреб тощо.

Область допустимих рішень ґрунтується, враховуючи фінансові, трудові, технологічні, ринкові обмеження витрат ресурсів. Розрізняють завдання оптимізації з однією та кількома функціями мети. Методи оптимізації використовуються при відборі проєктів з безлічі, які забезпечують певний рівень ризику, котрі не перевищують встановленого рівня.

3. Теорії ігор та статистичних рішень.

Теорія ігор застосовується для дослідження конфліктних ситуацій, де зіштовхуються супротивні сторони, кожна з яких переслідує власну мету, результат довільної дії кожної із сторін залежить заходів, які будуть вжиті супротивником.

Теорія ігор — це теорія математичного моделювання прийняття рішень в умовах конфлікту, задача якої полягає у виробленні ефективної поведінки учасників конфлікту.

4. Метод ставки дисконту з поправкою на ризик.

Метод надає можливість збільшити безризикову ставку відсотку на величину надбавки за ризик, враховуючи фактори ризику при розрахунку ефективності проєкту. У випадку інноваційних проєктів надбавка за ризик може = 10-20% .

Мета – коригування базової норми дисконту, котра вважається безризиковою або мінімально прийнятною.

Коригування реалізується завдяки додавання обсягу необхідної премії за ризик, після чого здійснюється розрахунок критерії ефективності проєкту (NPV, IRR) за отриманою нормою. Рішення приймається відповідно до обраного критерію.

5. Дерево рішень.

Для побудови «дерева рішень» аналітик має визначати склад та тривалість фаз життєвого циклу проєкту, виокремлюючи ключові події, котрі можуть вплинути на подальший розвиток проєкту та можливий час їх настання, аналітик обирає рішення, котрі можуть бути прийнятими в результаті настання кожної із подій, визначаючи ймовірність кожного із них, останній етап аналізу даних для побудови «дерева рішень» – встановлення вартості кожного етапу здійснення проєкту в

поточних цінах.

На основі даних формується «дерево рішень». Вузли представляють ключові події, а стрілки, які поєднують – перелік робіт по реалізації проєкту. Наводиться інформація відносно часу, вартості робіт та ймовірності розвитку того чи іншого рішення.

Як результат, побудова дерева рішень визначається ймовірність кожного сценарію розвитку проєкту та чистий приведений дохід (ЧПД) по кожному сценарію та по проєкту в цілому.

6. Метод достовірних еквівалентів.

Метод достовірних еквівалентів на відміну від попереднього надає можливість коригувати очікувані значення потоку. Єдиний метод, який би повністю відповідав вимогам до оцінки проєктних ризиків, виокремлюється важко та використовується їх комбінація. Основні методи в рамках кількісного аналізу проєктів є аналіз чутливості проєкту, сценарний підхід, імітаційне моделювання.

Вибір методу залежить від багатьох факторів (масштабу проєкту, складності, застосування інформації, оточення, де реалізується інвестиційний проєкт та ін). Наявність характеристик певних методів оцінки ризиків, виділяючи переваги та недоліки дозволяють орієнтуватися та обрати правильний для певного інвестиційного проєкту.

Результати якісно проведеного аналізу ризиків:

1. Ранжування загального ризику проєкту;
2. Список ризиків по пріоритету;
3. Список ризиків для додаткового аналізу та управління;
4. Тренди в результатах якісного аналізу ризику.

Ранжування ризику означає, що загальний ризик проєкту відповідно до інших проєктів, високий чи низький. Порівнюючи ризики різних проєктів по відношенню один до одного, то ризики розбиваються за пріоритетом, за різною кількістю критеріїв. Включаючи рейтинг: високий, низький, середній або рівень ієрархічної структури робіт.

Ризики, які потрапляють в категорію високих чи середніх, є головними

кандидатами для подальшого аналізу, включаючи кількісний аналіз ризиків та іншими діями з управління ризиками. При повторенні аналізу проявляється тенденція – тренд в результатах аналізу. Тренд робить відгуки на ризик чи подальший аналіз більш-менш терміновим і важливим.

2.3 Висновки до другого розділу

У другому розділі було розглянуто сучасні методи ідентифікації ризику, а саме метод Delphi, метод карт Кроуфорд, SWOT аналіз та метод експертної оцінки, котрий застосовується для ідентифікації ризиків на проекті. Було розглянуто ризик менеджмент як невід’ємну складову вдосконалення процесу проектування ПЗ та сучасні методи управління ризиками (визначення параметрів граничного рівня, теорія статистичних рішень, метод ставки дисконту, метод достовірних еквівалентів та оптимізаційний метод).

РОЗДІЛ 3 ВПРОВАДЖЕННЯ УДОСКОНАЛЕННЯ ПРОЦЕСІВ ПРОЄКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ШЛЯХОМ МІНІМІЗАЦІЇ РИЗИКІВ

3.1 Ідентифікація ризиків проєкту на підприємстві

Для аналізу ризиків розглядається проєкт в ІТ компанії IT-SEC, котра надає послуги для різних етапів діяльності компанії, починаючи від підготовки до відкриття бізнесу до розширення та зміни сфери його діяльності. IT-SEC розробляє мобільні рішення [6].

Проєкт Team Stream – це мобільне застосування для онлайн занять спортом. Онлайн платформи для тренувань є актуальними і можна здійснювати тренування не виходячи з дому. Основний напрямок занять – це йога, однак є інші напрямлення. Дана платформа має велику перспективу подальшого розвитку для клієнтів. Перевага – команда тренерів отримувати зворотний зв'язок від клієнтів, розуміючи одразу про недоліки та слабкі сторони системи та варіанти її поліпшення [6].

Ключовий функціонал програмного застосування для тренера є [6]:

1. Реєстрація облікового запису та заповнення даних профілю, опис про себе, створення тарифних планів та умов їх застосування;
2. Складання графіка онлайн занять та додавання клієнтів на них;
3. Проведення онлайн конференцій з групою чи персонально;
4. Спілкування з клієнтами, можливість підтримувати зв'язок;
5. Можливість завантажувати тренування для офлайн доступу, створювати добірки відео та групувати їх за темами.

Користувачу платформи надаються такі функції [6]:

1. Реєстрація акаунту та підписка на улюблених тренерів;
2. Відображення стрічки з усіма відео тренерів та перегляд їх персональної інформації;
3. Перегляд власного графіка занять та графіка тренера, можливість додавати собі тренування у календар;
4. Онлайн заняття у групі та наодинці з тренерів з відеозв'язку;
5. Онлайн чат з тренером;

6. Перегляд добірки тренера з відео для офлайн переглядів у вільний час.

Основними проблемами при розробці ПЗ є [6]:

- великий обсяг роботи, вимог, який часто змінюється;
- погана комунікація в команді процес розробки значно затягується;
- команда не встигає розробити функціонал в обговорені строки;
- команда витрачає занадто багато часу на непотрібні процеси.

Таким чином, з'являються непередбачені ризики, реагування на які не було сплановано зараніше, поява проблем із замовником через зірвані терміни, перевитрати бюджету, втрату прибутку та негативний відгук від клієнтів через пропущені недоліки перед релізами, через тиск із боку замовника, це призводить до негативного впливу на проєкт [6].

Недоліками проєкту є [6]:

Розробка первинної версії застосування було з використанням методології Waterfall, це призвело до [6]:

- частих змін у процесі розробки;
- тестування здійснювалося на останніх етапах розробки великих модулів;
- складно вносити зміни в документацію та доповнювати її;
- команда брала велику кількість задач та не вкладалась у терміни через великий обсяг роботи.

Для доопрацювання ПЗ, було вирішено застосувати саме ітеративний метод розробки. Принцип методу полягає у поділенні циклу на більш дрібні легко створювані модулі. Кожен модуль має проходити через фази визначення вимог, проєктування, кодування, інтеграції та тестування. Метод дуже підходить до цього проєкту так як процедура розробки по інкрементній моделі передбачає випуск на першому великому етапі продукту в базовій функціональності, послідовне додавання нових функцій – інкрементів [6].

В інкрементній моделі повні вимоги до системи поділяються на різні складання. Термінологія часто застосовується для опису поетапного складання ПЗ, який є актуальним для проєкту та активно розробляється. Існує кілька циклів

розробки та становлять життєвий цикл «мульти-водоспад». Цикл поділяється на дрібніші модулі, які легко формуються, де кожен модуль проходить через фази визначення вимог, проектування, кодування, інтеграції та тестування [6].

Це дає змогу допомогти команді не загубити існуючі та майбутні етапи. Процедура розробки по інкрементній моделі передбачає випуск першому великому етапі продукту у базовій функціональності та послідовне додавання нових функцій. Для ідентифікації ризиків проектування означеного ПЗ було залучено комбінацію двох методів – метод експертної оцінки та SWOT аналіз [6].

На проєкті є провідні розробники з великим досвідом роботи в даній галузі розробки, які беруть активну участь у розробці поточного проєкту з початку.

Проведений SWOT аналіз на підприємстві дав результати, Рис. 3.1 [6].



Рисунок 3.1 – SWOT аналіз проєкту Team Stream

Виходячи з результатів SWOT аналізу проектування ПЗ, ідентифіковано такі ризики на проєкті [6].

При розробці ПЗ важливо одразу коректно розпланувати завдання та

погодити їх з замовником, у залежності від формування бюджету на розробку. Погане планування впливає на бюджет, і це призводить до перевитрат, тобто витрачено на не пріоритетний функціонал, неправильно викладені вимоги, зайві комунікації та інше. Бюджет виділений на команду розподіляється не завжди коректно, не приділяється достатньо уваги на документування вимог, тестування чи налаштування ПЗ, перевірку коду, рефакторинг та виправлення технічних проблем [6].

Непорозуміння з замовником – це велика проблема при розробці ПЗ, такими як недостатня комунікація, чи надмірна комунікація, неправильно задокументовані вимоги чи їх відсутність [6].

Важлива проблема, на яку варто звернути увагу – погана якість коду через відсутність коректного тестування. Бюджет розподіляється не коректно, і на тестування виділяється мало часу. Результат, тестування покриває код не повністю та пропускається велика кількість проблем та помилок, котрі потрапляють до кінцевого користувача. Користувачі – залишають негативні відгуки [6].

Реліз ПЗ із запізненням також є поширеною проблемою при розробці, так як часто команда не вкладається в раніше погоджені терміни, надаючи неправильну оцінку за часом на розробку чи не закладається час на непередбачені ситуації, чи поповнюється список функціоналу котрий має бути випущений та не узгоджений раніше [6].

На проєкті працює не велика кількість кваліфікованих співробітників. Може бути так, що працівники є досвідченими та висококваліфікованими, однак виникають непередбачувані ситуації, коли співробітник відсутній на робочому місці. Це викликає проблему передачі його завдання іншим розробникам та призводить до перевитрат часу та затримки строків розробки ПЗ [6].

Ідентифіковані ризики та ресурси проєкту, впливають на ризики, у табл. 3.1 [6].

Після ідентифікації ризиків оцінюється вірогідність P_i їх появи на проєкті для кожного із ресурсів x_i та вартість C_i ризикових збитків на одиницю кожного ресурсу [6].

Експертами виступають: Менеджер проєкту, Замовник, Бізнес-аналітик, Менеджер з тестування, Провідний розробник та Дизайнер. Саме ці співробітники мають великий досвід роботи та працюють на проєкті з початку, тому вони зможуть об'єктивно оцінити [6].

Таблиця 3.1 – Ідентифіковані ризики та ресурси проєкту

Ідентифікатор ризику	Ризик	Ресурс (x)
R ₁	Невірна оцінка часових меж проєкту	Час на оцінювання робіт
R ₂	Недостатня взаємодія із замовником	Час на взаємодію с замовником на етапі розробки
R ₃	Недоліки тестування	Час на планування тестування
R ₄	Зміна вимог до підзадач проєкту у процесі його реалізації	Час на документування вимог до підзадач
R ₅	Недостатня кількість кваліфікованих розробників	Час роботи команди розробників

Кожен з них висловлює власну думку щодо вірогідності появи кожного з ризиків. Далі, результати порівнюються, обчислено середньо-арифметична оцінка вірогідності, котре застосовується для формування цільової функції, котру виражає загальну вартість ризикових збитків. Таким чином, оцінено вартість C_i ризикових збитків на одиницю кожного ресурсу [6].

експерт	Верогідність виникнення					Вартість компенсації				
	R1	R2	R3	R4	R5	R1	R2	R3	R4	R5
1	0,25	0,22	0,75	0,5	0,5	18	36	24	10	50
2	0,45	0,75	0,23	0,6	0,1	15	25	36	11	47
3	0,24	0,55	0,5	0,75	0,2	28	14	30	29	36
4	0,6	0,5	0,33	0,6	0,4	13	34	21	14	25
5	0,2	0,2	0,7	0,88	0,3	25	16	40	13	40
	0,348	0,444	0,502	0,666	0,3	19,8	25	30,2	15,4	39,6

Рисунок 3.2 – Результати оцінки кожного з експертів щодо вірогідності P_i виникнення ризиків на проєкті для кожного із ресурсів x_i та вартості C_i ризикових збитків на одиницю кожного ресурсу

Рис. 3.2, наведено результати оцінки кожного з експертів вірогідності P_i виникнення ризиків на проєкті для кожного із ресурсів x_i та вартості C_i ризикових

збитків на одиницю кожного ресурсу [6].

Далі, дані будуть застосовані для розрахунків коефіцієнтів капіталовкладення на одиницю ресурсу для побудови цільової функції задачі мінімізації ризикових збитків [6].

3.2 Планування та впровадження заходів для зниження ризиків

На підставі аналізу вирішується реагування на ризики.

Для мінімізації ймовірності появи ризиків та зниження вартості ризикових збитків при розробці ПЗ, варто сформувавши план реагування на ризик. План допомагає знизити ймовірність появи ризиків та уникнути критичних наслідків на проєкті під час розробки ПЗ [13].

Неправильна оцінка проєкту. При формуванні оцінки проєкту, іноді, вона не виправдовує очікувань. Команда обирає тривалість ітерації проєкту, стек технологій та інше. Між клієнтом та командою часто виникають розбіжності, які призводять до збільшення тривалості завдання, витрат, через які у клієнта закінчуються гроші і він не може завершити проєкт [13].

Для мінімізації цього ризику [13]:

1. Виконувати тільки найважливіші завдання;
2. Надати час розробникам для вивчення та зниження ризиків в частинах нового проєкту;
3. Надати час розробникам на ознайомлення з функціоналом до проведення оцінювання;
4. Надати час на планування роботи та обговорення її оцінки;
5. Надати передбачуваний період для команди розробників впродовж тижня для завдання, який виходить за рамки проєкту;
6. В управлінні проєктами конус невизначеності описує розвиток невизначеності в кращому випадку під час проєкту. На початку проєктів, мало що відомо про продукт або результати роботи, тому оцінки схильні до великої невизначеності.

Для відстеження роботи в команді, кількість витраченого часу та швидкість

виконання роботи, слід застосовувати автоматизовану систему для управління проектами та завданнями (Jira, Trello, Asana і інші). Часто застосовувана система для управління проектами та завданнями серед розробників є Jira [13].

JIRA – платформа для управління проектами, завданнями та відстеження помилок. Платформа призначена, для розробників та ведення agile-проектів.

Платформа JIRA – гнучкий інструмент, який може бути адаптований під потреби компанії. Для кожного окремого проекту адміністратор Jira визначає тип завдань з унікальними складовими елементами, прив'язуючи набір статусів. До кожному проекту визначається право доступу для учасників [13].

Структура JIRA формується з трьох елементів: проєкт, завдання та підзадачі. Проєкт – основний елемент платформи, де зберігаються завдання та інформація по роботі над програмою. Користувачі мають можливість створити проєкт з нуля чи використовувати готовий шаблон. Для відстеження ходи роботи над проєктом автоматично створюється дорожня карта. Дорожня карта проєкту – це ієрархічна структура, яка дозволяє планувати робочий процес в різних часових перспективах, відстежуючи процес виконання завдань та систематизуючи роботу кількох команд над одним проєктом [13].

Завдання JIRA – це структуровані інструменти для управління проєктом, де міститься інформація щодо необхідних дій, фіксується час для її виконання, встановлюється виконавець, прикріплюються додаткові файли. Користувач може отримувати повідомлення щодо внесених змінах завдання, ведеться журнал виконання, формуюючи підзадачі та залишати коментарі [13].

Особливості JIRA [13]:

1. Kanban-дошка – допомагає команді забезпечити прозорість роботи над проєктом, оптимізувати робочий процес, розподілити завдання з беклога, Рис. 3.3.
2. Scrum-дошка – дозволяє управляти складним проєктом, об'єднуючи команди з різних напрямків розробки продукту для досягнень однієї мети;
3. Прив'язка програмного коду до завдань завдяки Bitbucket та спільна над ним;
4. Ведення документації, протоколів та інших документів завдяки Confluence;

5. Спільна робота – обмін інформацією по проєкту, спільне вирішення питань та звернення до колег;

6. Звітність в JIRA – звіти формуються завдяки віджетів на панелі дашборда та містять інформацію щодо проєкту в цілому чи про окремі його елементи. Звіти візуалізуються в графіки та діаграми;

7. Підтримка інтеграцій з безліччю інструментів для розробки та інших сервісів.

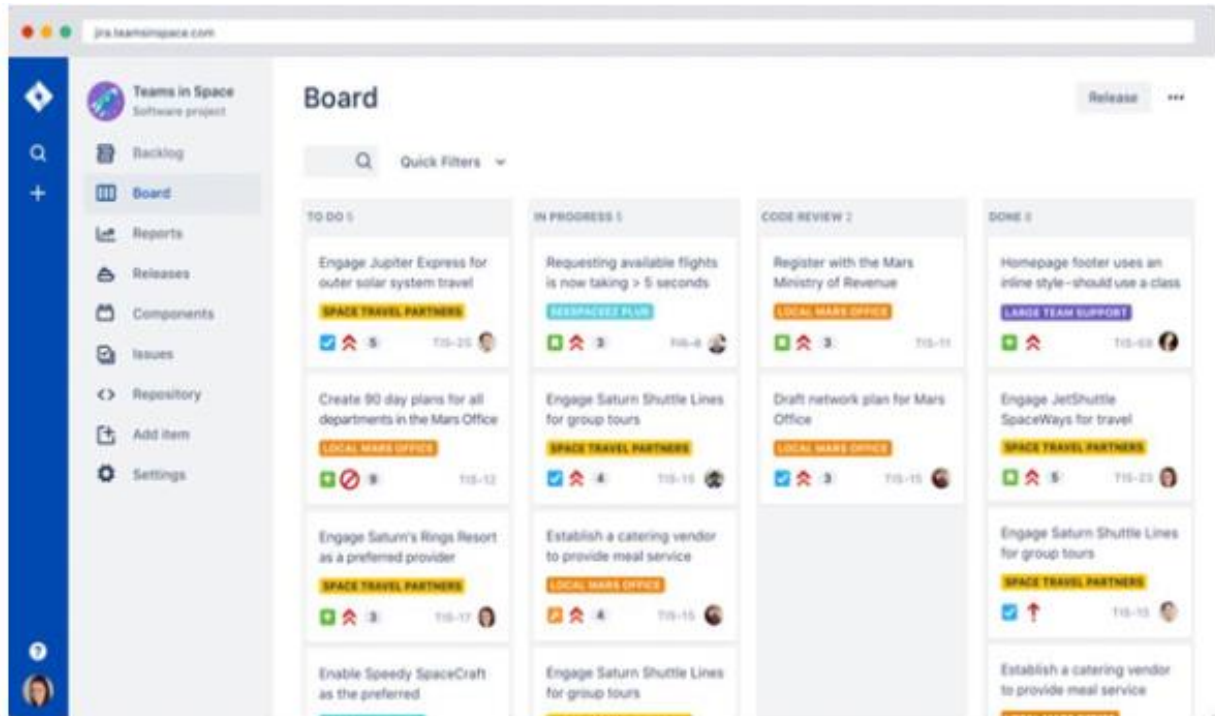


Рисунок 3.3 – Kanban дошка

Ризики поганої взаємодії між замовником ПЗ та його виконавцем у процесі розробки ПЗ. Пов'язуючи з відсутністю комунікації між керівниками чи їх представниками. Недостатнє обговорення користувацьких вимог до ПЗ чи його архітектури, чи після старту процесу реалізації може негативно вплинути на його майбутньому функціоналі та якість самого ПЗ. Щоб команда розробників закінчила вчасно проєкт слід часто спілкуватися з клієнтом чи менеджером проєкту для уточнення деталей по проєкту навіть вже на етапі реалізації. У випадку не дотримання цих рекомендацій, є ризик до непорозуміння з обох сторін та збільшення тривалості ітерацій. Слід враховувати, що надмірна комунікація про свою ефективність прирівнюється до недокомунікації. Якщо витратити

спілкування забагато багато часу, його потім може бути замало для виконання основного запланованого обсягу роботи. Важливо максимально точно для поточного проєкту обрати оптимальну кількість часу на початкову та проміжну комунікацію із замовником [13].

Для мінімізації цього ризику [13]:

1. Необхідно чітко узгодити, коли клієнт та замовник зможуть провести Тестування користувача;
2. Важливо обумовити допустимий час відповіді, якщо у кожної зі сторін є проблема або питання по проєкту;
3. Погодити часові межі для комунікації;
4. Слід визначати чіткий вибір цілей та пріоритетів проєкту.

Код низької якості та недоліки тестування. Низька якість коду – одна з найпоширеніших проблем в розробці та проблем клієнта. Часто, замовник не розуміє код та не може визначити його якість. До завершення розробки може з'ясуватися, що у додатки не якісно розроблений код, проблеми в роботі і відсутність грамотного тестування [13].

Як мінімізувати цей ризик [13]:

1. Розробникам важливо слідувати розробленим стандартам коду;
2. Клієнт може знайти менеджера проєкту чи технічного директора, який може перевіряти якість коду та контролювати команду розробників;
3. Дотримання системи Стандартів Якості Коду;
4. Тестування після кожної ітерації коду.

Найбільш важливу роль грає правильне тестування системи. Дане тестування проводиться для проєкту на основі ризиків. Тестування ризику використовує ризик для визначення пріоритету та підкреслення відповідних тестів у процесі виконання тесту. Ризик – це ймовірність настання небажаного результату. Результат пов'язаний із впливом, так як може не вистачити часу на тестування всіх функціональних можливостей, тестування на основі ризиків передбачає тестування функціональних можливостей, котрі впливають [13].

1. Тестування на основі ризиків починається на початку проєкту, визначає

ризика для якості системи та використовує дані знання про ризики для планування, специфікації, підготовки та виконання тестування;

2. Тестування на основі ризику включає як пом'якшення – тестування;

3. Тестування на основі ризиків передбачає вимір того, наскільки добре здійснюється пошук та усунення дефектів у критичних зонах;

4. Тестування на основі ризику включає застосування аналізу ризиків для виявлення можливостей для усунення чи запобігання дефектів завдяки тестування.

Мета тестування на основі ризику практично не може бути – проєкт без ризику. Тому, слід провести тестування із застосування найкращих практик з управління ризиками, щоб досягти результату проєкту, котрий урівноважує ризики з якістю, функціями, бюджетом і графіком [8].

Часті зміни у вимогах вже після початку виконання підзадачі. Вносячи зміни, слід проаналізувати вплив на поточний стан проєкту. Завдяки аналізу, можна грамотно розподілити обов'язки, внести зміни в пріоритети та надати клієнту точну інформацію про те, що може бути виконано. Нерозділена реакція означає беззастережне прийняття тих чи інших змін. Дана практика є поширеною в процесі розробки ПЗ. Важливо всім учасникам проєкту усвідомлювати наслідки змін та спільно йти на компроміси, за необхідності [8].

Результатом проблеми є те, що команда даремно витрачає час на питання стосовно базової інформації щодо проєкту. Відсутність хороших контрольних показників для застосування членами команди. Недостатні знання членів команди, котрі приєдналися до проєкту у процесі розробки.

Мінімізація ризику [8]:

1. Наявність мінімальної проєктної документації може зіграти велику роль в запобіганні наслідків.

2. Застосування спеціалізованих інструментів (JIRA, Confluence або QA Touch), які дозволяють знайти базову проєктну документацію, членів команди, їх ролі та іншу важливу інформацію за властивостями проєкту, його середовища, опису користувачів і переліку функцій.

3. Застосування спеціальних умовних позначень для опису завдань.

Документація не має бути об'ємною. Її задача – опис проєкту зрозумілою мовою.

Рис. 3.4, інтерфейс користувача системи Confluence – це простір для команд, де накопичено знання об'єднані з можливостями для спільної роботи. Динамічні сторінки – майданчик для творчості, збору інформації та спільної роботи учасників команди над будь-якими проєктами та ідеями. У розділах можна структурувати та організувати роботу в команді, надавати загальний доступ до бази знань організації та інформації, необхідної учасникам для ефективної роботи [8].

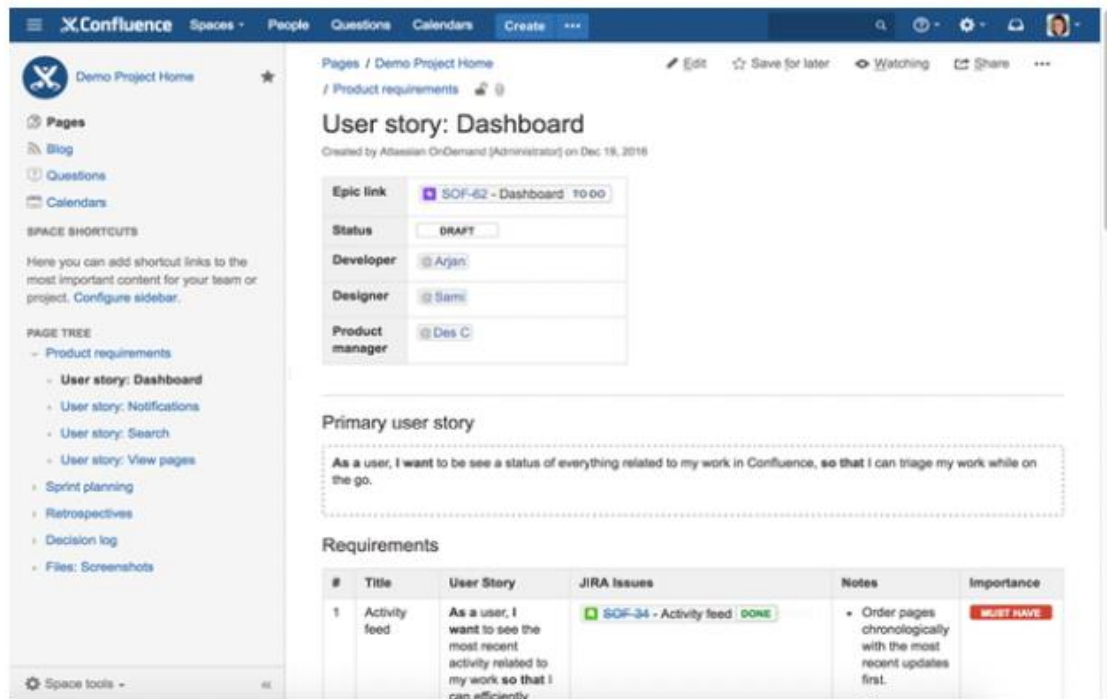


Рисунок 3.4 – Документування функціоналу

Confluence застосовується командами будь-якого розміру та типу (великі проєкти, критично важливі проєкти, ті, які шукають простір для формування командної культури та більш відкритого, природного способу взаємодії один з одним. Далі, Confluence, команда зможе швидко приймати рішення, домогтися погодження та отримати великі результати при спільній роботі [8].

Недостатня кількість кваліфікованих співробітників, залучаючи всіх членів команди є обов'язковою умовою для успіху кожного проєкту. Важливо, щоб кожен учасник процесу відданий для спільної мети, розуміючи власну роль та підтримувати інших членів команди. Нестача знань про проєкт, у випадку якщо фахівець був головним членом команди [8].

Мінімізація цього ризику [8]:

1. Всі члени команди мають однакові ключові знання про проєкт. У залежності від того, як довго відсутній співробітник та на якому етапі знаходиться проєкт, проджект-менеджер має прийняти рішення про заміну;

2. Новому співробітнику простіше приступити до роботи якщо з ним діляться інформацією про проєкт та передадуть наявну документацію;

3. Розширення штату співробітників на проєкті та навчання нового персоналу, однак потрібно мати на увазі що це потребує багато часу.

3.3 Експериментальна перевірка ефективності запропонованих заходів мінімізації ризиків

Після того як на проєкті виявлено ризики, оцінені ймовірні витрати на їх компенсацію збитків, отримані результати мінімізації ризикових збитків та інтеграції плану реагування, проаналізовано та виявлено, частота появи непередбачених ризиків значно знизилася, результат та зменшення витрат йде на покриття компенсації цих ризиків. Продуктивність команди зросла та збільшено кількість, незважаючи на те, що часу витрачається стільки ж, іноді навіть менше [17].

Покращена здатність команди об'єктивно оцінювати обсяг передбачуваної роботи, так як чітко визначаються вимоги та кількість часу для виконання поставленої задачі. Колективні зустрічі та обговорення допомагають покращити взаємини у команді. Наперед виокремлений час на ознайомлення з вимогами відіграє важливу роль у прийнятті рішення щодо оцінки обсягу робіт [17].

Ключовими змінами вважається покращення якості ПЗ та коду загалом, так як відповідає всім стандартам якості. Кількість позитивних відгуків збільшилася через те, що всі релізи проходять якісніше та частіше, зменшення витрат часу на виправлення всіх недоліків. Важливу роль даного поліпшення грає тестування після кожної ітерації розробки, де за виділений час команда тестувальників повною мірою покриває код тестами та перевіряє весь функціонал. Тому, тепер команда не витрачає на повторне виправлення дефектів та рефакторинг, а лише на тестування [17].

Проблеми з недостатньою комунікацією із замовником практично усунуто. Команда запроваджує додаткові мітинги із замовником вже на етапі розробки ПЗ, а не лише на етапі планування роботи. Мітинги відіграють велику роль для того, щоб переконатися, що за підсумком розробки продукту відповідає бажанню замовника. Команда вносить обмеження за часом на зустрічі щоб не перевитрати часу, як вже з'ясовано, надлишок комунікації позначається на проєкті як її нестача. Замовник бере участь у перевірці приймального тестування після фази розробки. Це допомагає команді продемонструвати необхідний результат, а замовнику надати можливість визначити чи відповідає це його вимогам [17].

Покращення становища з вимогами на проєкті, так як вони вже задокументовані ще до початку розробки функціоналу. Були проведені розмови із замовником щодо зміни вимог у процесі розробки підзавдання та потенційні збитки. На вимогу замовника, команда зможе підлаштуватися під нові вимоги, а де вже ні. Інтегрована система, завдяки якій бізнес-аналітика та проджект менеджер чітко прописується вимога до функціоналу, котра раніше узгоджена із замовником. У замовника також є доступ до системи, за необхідності може бути самостійно перевірена [17].

Питання з недостатньою кількістю висококваліфікованих розробників все ще відкрите, так як даний ризик складно передбачити завчасно. Склад співробітників може змінитися, співробітники можуть захворіти, перейти на інший проєкт, піти у відпустку чи звільнитися. Вирішено виокремити частину часу на навчання нових стажерів розробників, котрі наділені достатнім рівнем досвіду на проєкті. Досвідчені розробники здійснюють менторство над джуніорами розробниками. Поліпшення роботи у даному напрямі складно оцінити, для отримання результату, необхідна велика кількість часу на навчання нового персоналу. Однак, незважаючи на це, команда здійснює дії, які допомагають звести даний ризик до мінімуму [17].

Отримані результати після інтеграції плану реагування на ризики були обговорені повторно з експертами. Прийнято рішення, що показники ймовірності появи ризиків на проєкті значно зменшились. У табл. 3.1 наведено порівняльну характеристику результатів оцінки експертів щодо вірогідності появи ризиків [17].

Можна зробити висновок, що запропонований план управління ризиками є ефективним для конкретно розглянутого проекту, тому як показники ймовірності появи ризику є значно нижче [17].

Таблиця 3.1 – Результати опитування експертів щодо вірогідності виникнення ризиків після впровадження плану реагування

Ризик	Вірогідність виникнення ризику до впровадження плану реагування на ризики	Вірогідність виникнення ризику після впровадження плану реагування
Невірна оцінка часових меж проекту	0,3	0,15
Недостатня взаємодія із замовником	0,4	0,15
Недоліки тестування	0,5	0,1
Зміна вимог до підзадач проекту у процесі його реалізації	0,6	0,2
Недостатня кількість кваліфікованих співробітників	0,3	0,05

Застосовані нові дані оцінок по ймовірність появи ризиків на проекті, котрі отримані в результаті здійснення додаткової експертної оцінки (табл. 3.1) [17].

Була побудована цільова функція з новими коефіцієнтами [17]:

$$F(x) = \sum_{i=5}^n R_i x_i = 3x_1 + 3,75x_2 + 3x_3 + 3x_4 + 2x_5 \rightarrow \min$$

Задача мінімізації ризикових збитків для удосконалення процесів проектування застосунок на підприємстві													
Числові значення для системи обмежень		Формули для системи обмежень											
500	500												
700	120												
Кількість одиниць ресурсів						Цільова функція	0,15	0,15	0,1	0,2	0,05	PI	вірогідність виникнення ризику
x1	x2	x3	x4	x5			20	25	30	15	40	CI	вартість ризику, якщо він трапиться, на одиницю ресурсу x
15	15	60	30	380	1131,25		3	3,75	3	3	2	RI	вартість ризику з урахуванням вірогідності виникнення

Рисунок 3.5 – Результати повторного розв'язання задачі мінімізації вартості ризикових збитків

Повторно розв'язана задача мінімізації вартості ризикових збитків, Рис. 3.5.

У даний момент, ресурси розподілені іншим чином. Можна побачити, що тепер на оцінювання робіт витрачається менше часу, у порівнянні з першим результатом, а на написання коду – більше [17].

$$x_1 = 15 \text{ год. } x_2 = 15 \text{ год. } x_3 = 60 \text{ год. } x_4 = 30 \text{ год. } x_5 = 380 \text{ год.}$$

Рис. 3.6, порівняльний графік розподілення ресурсів, одержано в результаті розв’язання задачі мінімізації ризикових збитків, до та після інтеграції заходів для зниження ризиків [17].

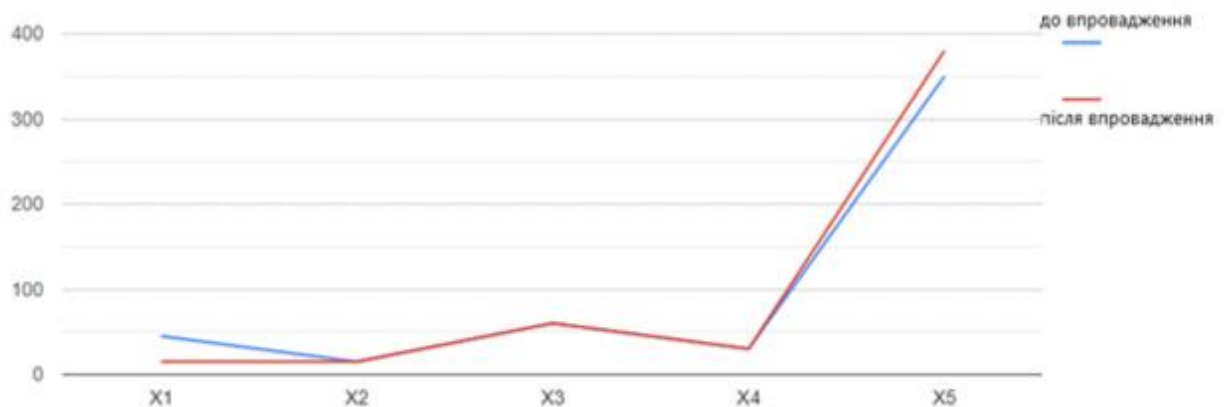


Рисунок 3.6 – Порівняльний графік розподілення ресурсів

3.4 Висновки до третього розділу

У третьому розділі кваліфікаційної роботи було застосовано оптимізаційний метод для управління ризиками при проєктуванні ПЗ. Побудовано математичну модель задачі мінімізації ризикових збитків для вдосконалення процесів проєктування ґрунтуючись на результатах при отриманні в ході експертної оцінки. Дану модель мінімізації ризикових збитків було інтегровано для вдосконалення процесів проєктування ПЗ в організації.

Стан на підприємстві після інтеграції було оцінено повторно, результати показали, що за допомогою запропонованих заходів для зниження вірогідності ризиків та ризикових збитків можна досягти зниження вартості ризикових збитків.

ВИСНОВКИ

У процесі підготовки кваліфікаційної роботи були виконані наступні **задачі**:

1. Теоретичні засади удосконалення процесів проєктування програмного забезпечення
2. Шляхи мінімізації ризиків при проєктуванні програмного забезпечення
3. Впровадження удосконалення процесів проєктування програмного забезпечення шляхом мінімізації

У першому розділі кваліфікаційної роботи було проаналізовано сучасні підходи до проєктування ПЗ. Виявляються переваги та недоліки кожного з цих методів. Всі методології мають загальні проблеми при проєктуванні ПЗ, а саме підвищений ризику, які можуть знадобитися для внесення змін, необхідність у жорсткому управлінні та контролі, проблеми щодо проєктування архітектури системи, підвищення вартості проєкту за рахунок додаткових часів на планування, прототипування, документування та визначення цілей проєкту.

Розглянуто сутність поняття ризик, досліджено його класифікацію, виділено зовнішні та внутрішні, технічні та нетехнічні, передбачувані та непередбачувані, юридичні, управлінські, фінансові, маркетингові та ризики персоналу.

У другому розділі було розглянуто сучасні методи ідентифікації ризику, а саме метод Delphi, метод карт Кроуфорд, SWOT аналіз та метод експертної оцінки, котрий застосовується для ідентифікації ризиків на проєкті. Було розглянуто ризик менеджмент як невід'ємну складову вдосконалення процесу проєктування ПЗ та сучасні методи управління ризиками (визначення параметрів граничного рівня, теорія статистичних рішень, метод ставки дисконту, метод достовірних еквівалентів та оптимізаційний метод).

У третьому розділі кваліфікаційної роботи було застосовано оптимізаційний метод для управління ризиками при проєктуванні ПЗ. Побудовано математичну модель задачі мінімізації ризикових збитків для вдосконалення процесів проєктування ґрунтуючись на результатах при отриманні в ході експертної оцінки. Дану модель мінімізації ризикових збитків було інтегровано для вдосконалення процесів проєктування ПЗ в організації.

Стан на підприємстві після інтеграції було оцінено повторно, результати показали, що за допомогою запропонованим заходам для зниження вірогідності ризиків та ризикових збитків можна досягти зниження вартості ризикових збитків.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. «PersonPro 2.0» htp та «PersonPro 2.0 та SQL» Renaisc [Електронний ресурс] – Режим доступу: Oracle <http://personpro.ami.ua> 2011. 20–27 pp
2. 20 базових HR-метрик, котрі допоможуть виміряти ефективність [Електронний ресурс] – Режим доступу: <https://hurta.work/ru/blog/20-bazovyh-hr-metrik-kotorye-pomogut-izmerit-effektivnost-raboty-kompanii/>
3. Бондаренко Е. HR-аналітика в українських компаніях // HR ЛІГА співтовариство кадровиків та фахівців з управління персоналом. [Електронний ресурс] – Режим доступу :<https://hrliga.com/index.php?module=profession&op=view&id=1844>
4. Батенко Л.П. Управління проєктами «Принципи, запропоновані авторами». Л.П. Батенко, О.А. Загородніх, В.В. Ліщинська. – К., 2020. – 231с
5. Биков В. Ю. Хмарні технології, ІКТ-аутсорсинг і нові функції ІКТ підрозділів освітніх і наукових установ / В. Ю. Биков // Інформаційні технології в освіті. – 2018. – № 10. – С. 8–23.
6. Блек Р. Ключові процеси тестування / Рекс Блек., 2020. – 112 с.
7. Винокурова В. К. Дослідження ризиків при проєктуванні ПЗ за різними методологіями / В. К. Винокурова, Л. Е. Гризун // Інформаційні технології та системи: зб. тез Міжнар. наук-практ. конф. (X – X грудня 2021 р.; ХНЕУ ім. С. Кузнеця). – Х. : ФОП Бровін О. В., 2021. – С. 45.
8. Винокурова В. К. Управління ризиками при проєктуванні програмного забезпечення за різними методологіями / В. К. Винокурова, Л. Е. Гризун // Інформаційні технології та системи: зб. тез Міжнар. наук-практ. конф.(14 – 15 квітня 2022 р.; ХНЕУ ім. С. Кузнеця). – Х. : ФОП Бровін О. В., 2022. – С. 5.
9. За загальною редакцією Шапіро В.Д. Управління проєктами. Підручник.: «Два Три», 2020.
10. Канер С. Тестування програмного забезпечення / С. Канер, Д. Фолк, Е. Нгуен., 2020. – 421 с.

11. Керівництво до Зводу знань з управління проєктами. Третє видання (Посібник РМВОК)/. Американський національний стандарт ANSI/PMI 99-001-2019).

12. Кобилянський Л.С. Управління ризиками [навч.посібник] / Л.С. Кобилянський. – К.: Мир управління проєктами: основи, методи, організація, применение / Под. ред. Х. Решке, Х.Шелле.

13. Майерс Г. Мистецтво тестування програм / Г. Майерс, К. Сандлер, Т. Баджетт., 2019. – 272 с.

14. Методичні рекомендації до виконання магістерської дипломної роботи для студентів ОПП «Комп'ютерні науки» спеціальності 122 «Комп'ютерні науки» другого (магістерського) рівня: [Електронне видання] / уклад. С.В.Мінухін, І.О.Ушакова, Д.Ю. Голубничий, О.В.Щербаков. – Х. : ХНЕУ ім. С. Кузнеця, 2021. – 59 с. (Укр. мов.)

15. Методичні рекомендації до оформлення звітів, курсових ктів та дипломних робіт (ктів) для студентів спеціальності 121 «Інженерія програмного забезпечення», 122 «Комп'ютерні науки», 126 «Інформаційні системи і технології»: [Електронне видання] / уклад. І.О.Ушакова, Г.О. Плеханова, О.М. Беседовський. – Х. : ХНЕУ ім. С. Кузнеця, 2021. – 48 с.

16. Палеха Ю.І., Горбань Ю.І. Інформаційний бізнес : підручник / Ю.І. Палеха, Ю.І. Горбань – К.: Вид-во Ліра-К. 2015. – 492 с.

17. Первушин Т.Л. Стратегія розвитку підприємства з урахуванням ризиків. Економіка та управління народним господарством: збірка статей.

18. Посібник до Зводу знань з управління проєктами. Третє видання (Посібник РМВОК)/. Американський національний стандарт ANSI/PMI 99-001-2019.

19. Середа Г. В. Гейміфікація в менеджменті персоналу: зарубіжний та український досвід. Економіка і організація управління. 2017. Вип. 4. С. 216–223. [Електронний ресурс] – Режим доступу:http://nbuv.gov.ua/UJRN/eiou_2018_4_22.

Оформлення слайдів та роздаткового матеріалу