

ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД  
«УНІВЕРСИТЕТ ЕКОНОМІКИ ТА ПРАВА «КРОК»»

**КВАЛІФІКАЦІЙНА РОБОТА**

Тема: «ГНУЧКЕ УПРАВЛІННЯ РОЗРОБКОЮ ОПЕРАЦІЙНОЇ СИСТЕМИ  
«VW OS» ДО ТАКТИЧНОГО ГОДИННИКА  
«VIGILANT WATCHMAN»

Ступінь вищої освіти – магістр

Спеціальність – 073 «Менеджмент»

Освітня програма «Agile-технології розробки програмного забезпечення»

**ПОЯСНЮВАЛЬНА ЗАПИСКА**

Керівник: к.т.н., доцент  
Веніамін ГІТІС

Керівник: викладач.  
Олег МУШИНСЬКИЙ

Виконав: здобувач  
групи МЕН/Agile-23м  
Іван ПИГА

Київ, 2024 р

ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД  
«УНІВЕРСИТЕТ ЕКОНОМІКИ ТА ПРАВА «КРОК»»

ЗАТВЕРДЖУЮ:

завідувач кафебри  
інформаційного менеджменту,  
математики та статистики

\_\_\_\_\_ Денис БАЛДИК

«\_\_»\_\_\_\_20\_\_ р.

ЗАВДАННЯ  
НА КВАЛІФІКАЦІЙНУ РОБОТУ

***ПИГА ІВАН МИКОЛАЙОВИЧ***

Тема роботи	Гнучке управління розробкою операційної системи «vw os» до тактичного годинника «vigilant watchman»
Номер та дата наказу про затвердження теми	№56-3 від 27.06.2024
Коротка постановка завдання	Впровадження гнучкого підходу до управління створенням розробкою операційної системи «VW OS» до тактичного годинника «vigilant watchman, призначеної для людей які працюють в екстремальних умовах
Посилання на джерела інформації (не більше п'яти найменувань, які рекомендує науковий керівник)	Рач, Д.В. Управління невизначеністю та ризиками в проєкті: термінологічна основа / Д.В. Рач // Управління проєктами та розвиток виробництва: Зб. наук. пр. – Луганськ: вид-во СЛУ ім. В. Даля, 2013. – №3(47). – С. 146–164. – Режим доступу: <a href="http://www.pmdp.org.ua">http://www.pmdp.org.ua</a> .
Вимоги до кваліфікаційної роботи	Кваліфікаційна робота має містити теоретичне та/або практичне дослідження за темою роботи, яку слід розглядати як складне спеціалізоване завдання або практичну проблематику в галузі управління та адміністрування, яка характеризується комплексністю та невизначеністю умов і потребує застосування теорій і методів Agile технологій.

Дата видачі завдання «14» липня 2024 р.

Керівник

Веніамін ГІТІС

Керівник

Олег МУШИНСЬКИЙ

Студент

Іван ПИГА

## КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання	Примітка
<b>Підготовчий етап</b>			
1	Вибір напрямку дослідження та керівника	01.07.2024 р.	Виконано
2	Формування теми та призначення керівника	08.07.2024 р.	Виконано
3	Затвердження теми кваліфікаційної роботи	09.07.2024 р.	Виконано
4	Затвердження завдання на кваліфікаційну роботу	15.07.2024 р.	Виконано
<b>Основний етап</b>			
5	Розробка концепції кваліфікаційної роботи	22.07.2024 р.	Виконано
6	Підбір та вивчення джерел інформації з напрямку дослідження. Огляд існуючих аналогів.	29.07.2024 р.	Виконано
7	Затвердження розширеної постановки завдання. Підготовка та подання керівнику розділу 1 кваліфікаційної роботи	18.09.2024 р.	Виконано
8	Проектування інформаційної системи. Підготовка та подання керівнику розділу 2 кваліфікаційної роботи	18.09.2024 р.	Виконано
9	Реалізація інформаційної системи. Підготовка та подання керівнику розділу 3 кваліфікаційної роботи	25.09.2024 р.	Виконано
10	Підготовка та подання керівнику першого варіанту всієї кваліфікаційної роботи	01.10.2024 р.	Виконано
11	Доопрацювання кваліфікаційної роботи з урахуванням зауважень керівника та представлення керівнику доопрацьованого варіанту кваліфікаційної роботи	04.10.2024 р.	Виконано
<b>Завершальний етап</b>			
12	Представлення рукопису для перевірки на плагіат	07.10.2024 р.	Виконано
13	Підготовка презентації та доповіді на передзахист	07.10.2024 р.	Виконано
14	Передзахист кваліфікаційної роботи	08-11.10.2024 р.	Виконано
15	Технічна самооцінка роботи на відповідність вимогам до оформлення та виправлення недоліків	08-11.10.2024 р.	Виконано
16	Експертиза роботи керівником та зовнішнім експертом	14.10.2024 р.	Виконано
17	Доопрацювання доповіді та презентації для захисту	18.10.2024 р.	Виконано
18	Захист кваліфікаційної роботи	21-25.10.2024 р.	Виконано

Керівник

Веніамін ГІТІС

Керівник

Олег МУШИНСЬКИЙ

Студент

Іван ПИГА

## ЗМІСТ

<b>ВСТУП .....</b>	<b>7</b>
<b>РОЗДІЛ 1. ДИЗАЙН БІЗНЕСУ «VIGILANT WATCHMAN» .....</b>	<b>11</b>
1.1. Опис предметної області для якої розробляється «VW OS»...	11
1.2. Постановка цілей та завдань .....	16
1.3. Визначення вимог до продукту.....	18
Висновки розділу 1 .....	23
<b>РОЗДІЛ 2. ГНУЧКЕ УПРАВЛІННЯ СТВОРЕННЯМ ПРОДУКТУ «VW OS» .....</b>	<b>25</b>
2.1. Обґрунтування вибору фрейму гнучкого управління проектом 25	
2.2. Планування проекту з розробки продукту «VW OS» .....	29
2.3. Моніторинг виконання проекту.....	37
Висновки розділу 2.....	41
<b>РОЗДІЛ 3. РЕЗУЛЬТАТИ РОБОТИ ТА ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ КОМАНДИ З РОЗРОБКИ ПРОДУКТУ «VW OS» ....</b>	<b>42</b>
3.1. Огляд інкременту продукту .....	42
3.2. Ретроспектива роботи команди.....	44
3.3. Сучасний інструментарій менеджера в Agile-середовищі .....	46
Висновки розділу 3.....	54
<b>ВИСНОВКИ .....</b>	<b>56</b>
<b>СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....</b>	<b>58</b>
<b>ДОДАТОК А.....</b>	<b>61</b>

## **Пига І. М. ГНУЧКЕ УПРАВЛІННЯ РОЗРОБКОЮ ОПЕРАЦІЙНОЇ СИСТЕМИ «VW OS» ДО ТАКТИЧНОГО ГОДИННИКА «VIGILANT WATCHMAN»**

Кваліфікаційна випускна робота на здобуття ступеня вищої освіти магістра за спеціальністю 073 – Менеджмент. – ВНЗ «Університет економіки та права «КРОК», Навчально-науковий інститут інформаційних та комунікаційних технологій, кафедра математичних методів та статистики, Київ, 2024.

У кваліфікаційній роботі розглядаються поточні проблеми гнучкого управління, пов'язані з розробкою операційної системи «VW OS» до тактичного годинника «VIGILANT WATCHMAN», використовуючи методи та інструменти гнучкого управління проєктами. На основі аналізу ринку та вимог клієнтів розроблено план процесу створення продукту. Використовуючи фреймворк Scrum, було встановлено особливості гнучкого управління розробкою операційної системи до тактичного годинника. Окрім того, для спрощення гнучкого управління розробкою операційної системи, розглянуто характеристики лідерства, комунікації та управління взаємодією.

Ключові слова: Agile, Scrum, тактичний годинник, операційна система, розробка, менеджмент.

Табл. 3. Рис. 19. Бібліограф.: 7 найм.

## **Pyha I. M. FLEXIBLE DEVELOPMENT MANAGEMENT OF THE «VW OS» OPERATING SYSTEM TO THE «VIGILANT WATCHMAN» TACTICAL WATCH**

Qualifying final work for obtaining a master's degree in higher education by specialty 073 – Management. – «KROK» University, Educational and Scientific Institute of information and communication technologies, Department of Mathematical Methods and Statistics, Kyiv, 2024.

The qualification work considers the current problems of agile management related to the development of the operating system “VW OS” for the tactical watch “VIGILANT WATCHMAN”, using methods and tools of agile project management. Based on market analysis and customer requirements, a product development process plan was developed.

Using the Scrum framework, the features of agile management of operating system development to the tactical clock are identified. In addition, the characteristics of leadership, communication and interaction management are considered to simplify the agile management of operating system development.

Keywords: Agile, Scrum, tactical clock, operating system, development, management.

Tabl. 3. Fig. 19. Bibliography: 7 Items.

## ВСТУП

### **Актуальність теми.**

Сучасний світ характеризується стрімкими технологічними змінами, що вимагають від компаній гнучкості та адаптивності. Особливо це стосується сфери розробки програмного забезпечення, де традиційні методи управління проектами часто виявляються недостатніми. Одним із перспективних підходів є гнучке управління (Agile), яке дозволяє оперативно реагувати на зміни вимог клієнтів та ринку.

Дана магістерська робота присвячена дослідженню застосування методології гнучкого управління при розробці операційної системи «VW OS» для тактичного годинника «Vigilant Watchman» і є надзвичайно актуальною з кількох причин, зазначених нижче.

Технологічний прогрес, вимоги ринку та специфіка тактичних годинників роблять гнучку розробку незамінною. Швидкі зміни в технологіях, зростаючі вимоги користувачів та висока конкуренція на ринку вимагають від розробників гнучкого підходу. Особливо це стосується таких спеціалізованих пристроїв, як тактичні годинники. Ці пристрої працюють в екстремальних умовах і постійно доповнюються новими функціями. Гнучка розробка дозволяє оперативно реагувати на зміни, швидко інтегрувати нові технології та адаптувати продукт до еволюціонуючих потреб користувачів.

Завдяки гнучким методологіям можна досягти значних переваг: швидкий вихід на ринок, адаптація до змін, висока якість продукту, задоволеність клієнтів та зниження ризиків. Швидка розробка та отримання зворотного зв'язку від користувачів дозволяють постійно вдосконалювати продукт. Гнучкий підхід забезпечує можливість оперативно реагувати на зміни вимог та ринкових умов, що є критично важливим в динамічному технологічному середовищі. Фокус на якості кожного інкременту продукту гарантує високу надійність і функціональність. Тісна співпраця з

користувачами дозволяє створювати продукти, які максимально відповідають їхнім потребам. Регулярна оцінка прогресу та адаптація до змін знижують ризику провалу проекту.

Наукова новизна роботи полягає в адаптації загальних принципів гнучкого управління до специфічних вимог проекту, а також в оцінці ефективності різних практик гнучкого управління в умовах обмежених ресурсів та високих вимог до надійності. Практична значущість роботи полягає в розробці рекомендацій для розробників, які планують використовувати гнучке управління в подібних проектах.

У контексті розробки операційної системи для годинника «Vigilant Watchman», гнучке управління може забезпечити такі переваги.

В першу чергу, оптимізація використання ресурсів, що включає регулярну оцінку ефективності розробки і дозволяє оптимізувати використання обчислювальних ресурсів і енергії годинника.

Наступний етап підвищення надійності, що описує часте тестування і дозволяє виявляти та усувати помилки, яке підвищує надійність операційної системи.

Потрібно зауважити, що краща інтеграція з новими датчиками та модулями у розрізі гнучкої методології дозволяє легко інтегрувати нові датчики та модулі в годинник.

Вирішення цієї актуальної проблеми вимагає прояву компетентності з гнучкого управління процесами створення інноваційних програмних продуктів, а також застосування сучасних підходів в управлінській діяльності. Це визначило мету і задачі дослідження.

**Мета дослідження** – виявити особливості та реалізувати функції гнучкого управління розробкою операційної системи до тактичного годинника «Vigilant Watchman».

Таке дослідження має практичну значущість, оскільки його результати можуть бути використані для покращення процесу розробки операційної

системи «VW OS», збільшення ефективності та якості продукту, зменшення ризиків провалу проекту та підвищення задоволеності клієнтів та команди розробників.

**Завдання дослідження.** Для досягнення мети необхідно виконати наступні завдання:

- обґрунтувати візію створюваного продукту для бізнесу;
- детально описати особливості гнучкого управління створенням операційної системи до тактичного годинника «Vigilant Watchman» з використанням обраного фреймворку;
- дослідити діяльність команди розробки операційної системи та описати підходи для підвищення ефективності та результативності досягнення цілей бізнесу.

**Об'єкт дослідження** - процеси реалізації функцій менеджменту при створенні програмних засобів, як операційна система.

**Предмет дослідження** - процеси управління розробкою програмного забезпечення (операційної системи) з використанням Agile-підходів (на прикладі продукту «Vigilant Watchman»).

**Методи дослідження.** Для обґрунтування візії створюваного продукту, як операційної системи, для розв'язання проблеми в діяльності замовника, а також виявленні викликів у діяльності команди розробки у Agile-середовищі використані загальнонаукові методи аналізу, синтезу, критичного мислення, системного підходу та аналізу. Для реалізації процесів гнучкого управління створення операційної системи до тактичного годинника «**Vigilant Watchman**» застосовано професійний інструментарій, як Jira, Confluence, Excel, PowerPoint.

**Новизна та практичне значення результатів дослідження.** Результати дослідження гнучкого управління при розробці операційної системи тактичного годинника мають значний практичний вплив. Ось деякі з них:

1. Команда розробників - оптимізація процесів за рахунок використання ефективної практики гнучкого управління, підвищення мотивації шляхом бачення прямого зв'язку між своєю роботою та успіхом проєкту, розвиток компетенцій членів команди;

2. Компанія - покращення якості продукту за рахунок тісної співпраці з клієнтом та швидке реагування на зміни, регулярна оцінка прогресу та адаптація до змін для зменшення ризиків, успішна реалізація проєкту прямо пропорційна підвищенню репутації компанії на ринку;

3. Галузь - сприяння розробці нових підходів до застосування гнучкого управління, сформування нових стандартів та рекомендації розробникам, стимулювання інновацій та експериментів до появи нових продуктів та послуг.

Таким чином, практичне значення полягає в тому, що спряє підвищенню ефективності розробки програмного забезпечення, покращення якості продуктів та зміцнення конкурентних позицій компанії.

**Структура та обсяг роботи.** Магістерська кваліфікаційна робота має вступ, три розділи, висновки до них, загальний висновок, список посилань та додатки. Загальний обсяг сягнув 60 сторінок, а основний текст розмістився на 54 сторінках.

## **РОЗДІЛ 1. ДИЗАЙН БІЗНЕСУ «VIGILANT WATCHMAN»**

### **1.1. Опис предметної області для якої розробляється «VW OS»**

Компанія «Vigilant Watchman» спеціалізується на виробництві тактичних годинників, призначених для використання у складних та екстремальних умовах. Її продукти розроблені для військових, рятувальників, спортсменів та інших професіоналів, які потребують високої надійності, функціональності та витривалості техніки. Тактичні годинники від «Vigilant Watchman» зазвичай мають міцну конструкцію, підвищену водо- і ударостійкість, а також підтримують спеціальні функції, як-от GPS-навігація, моніторинг фізичних показників і синхронізація з мобільними пристроями.

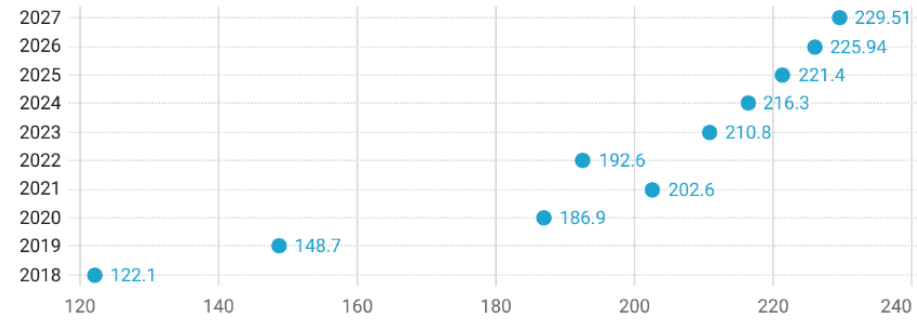
Компанія фокусується на інноваціях у дизайні та інтеграції передових технологій, зокрема у розробці операційних систем, що забезпечують ефективне управління функціями годинників і зручний інтерфейс для користувача. «Vigilant Watchman» відома своєю націленістю на безпеку, точність і багатофункціональність продуктів, що робить їх популярними серед користувачів, які потребують надійних пристроїв у найскладніших умовах.

Цілі та завдання прикріплюються статистичними даними [1], котрі розкривають та доповнюють значення «розумних» годинників (включає у себе тактичні годинники) у повсякденному житті людини.

Отож, прогноз використання смарт-годинників з 2018 по 2027 роки наведено на Рисунку 1.

## Worldwide Smartwatch Users

2018 to 2027



(Users in Million)

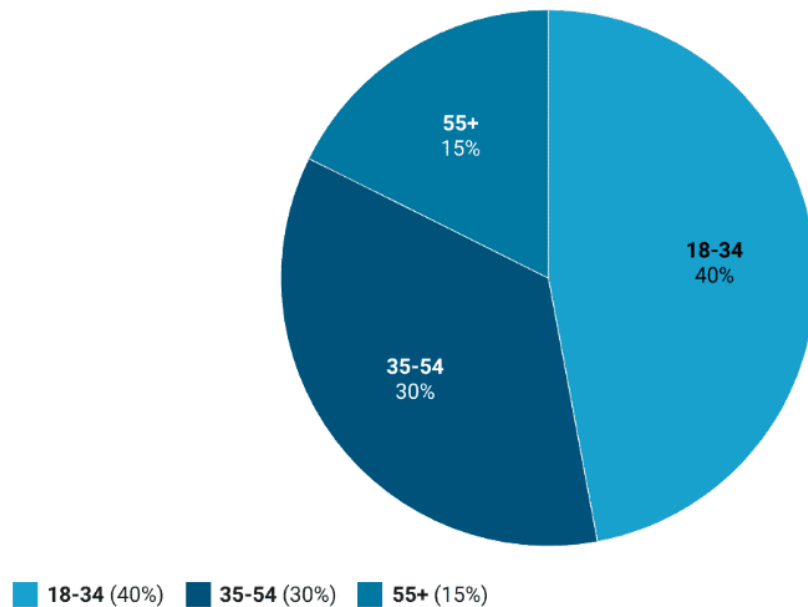
Рисунок 1. Світове використання смарт-годинників.

Джерело: Smartwatch Statistics 2024 By Wearables, Technology, Devices [1].

Важливо, що носимі гаджети користуються популярністю не лише у вікової категорії 18-28 років, а у старших людей. Рисунок 2 показує повну статистику використання носимих гаджетів.

## Smartwatch Ownership Percentage

By Age Group



(Smartwatch Ownership in %)

Рисунок 2. Відсоткове співвідношення користувачів смарт-годинників.

Джерело: Smartwatch Statistics 2024 By Wearables, Technology, Devices [1].

Варто відмітити, що чоловіча стать випереджає жіночу стать на 7% у використанні носимих гаджетів, що продемонстровано на Рисунку 3.

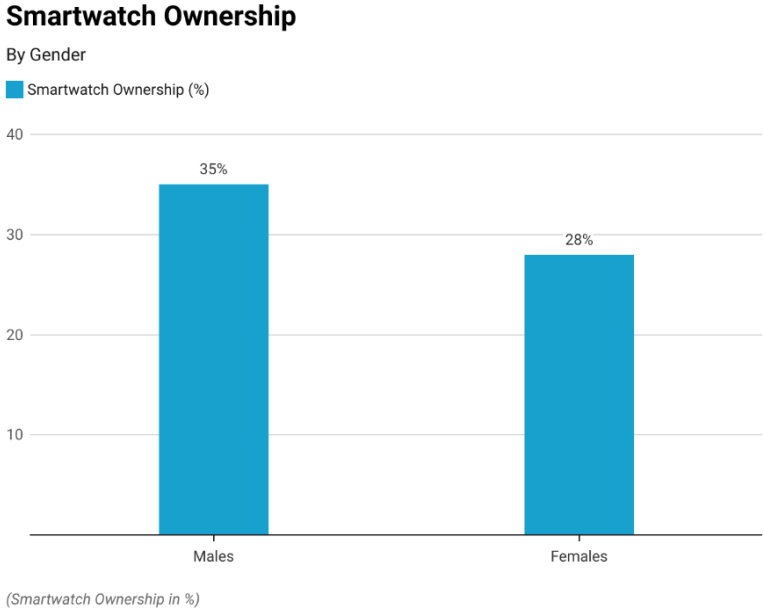


Рисунок 3. Користувачі смарт-годинників.

Джерело: Smartwatch Statistics 2024 By Wearables, Technology, Devices [1].

Важливою ознакою використання смарт годинників є функції, котрі безпосередньо впливають на життя людини, що демонструє Рисунок 4.

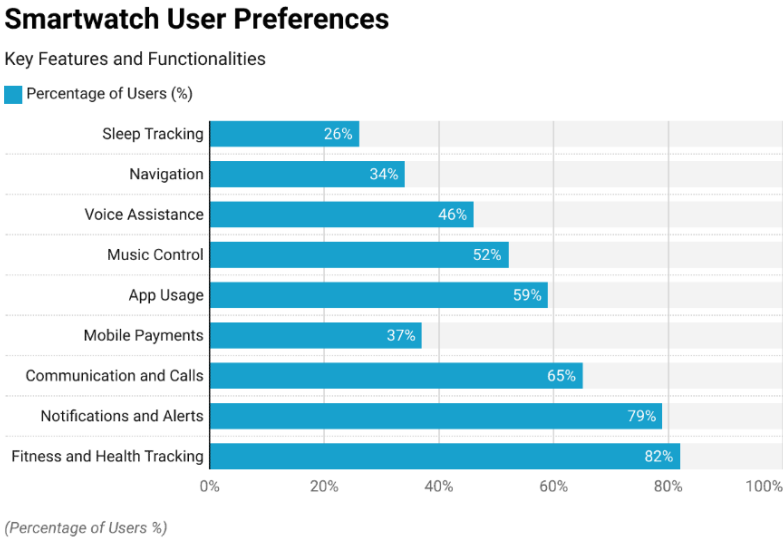


Рисунок 4. Функції, котрими користуються користувачі.

Джерело: Smartwatch Statistics 2024 By Wearables, Technology, Devices [1].

Аналіз конкурентів включає дослідження прямих та непрямих конкурентів на ринку тактичних годинників, їхніх сильних та слабких сторін, стратегій та ринкових позицій. Визначення зацікавлених сторін допоможе зрозуміти, хто буде впливати на процес виробництва та просування тактичних годинників, а також чиї інтереси потрібно враховувати.

Таблиця 1. – Аналіз конкурентів

Критерії оцінки	Оцінка конкурентів		
	Casio (G-Shock) [2]	Suunto [3]	Garmin [4]
Досвід роботи	41 рік	88 років	35 років
Імідж	Високий, міжнародна компанія з якісними продуктами		
Тенденції розвитку	Високі, постійне оновлення та використання по світу		
Потенціал розвитку	Високий, нові продукти з'являються раз на 3 роки	Високий, нові продукти з'являються раз на 4 роки	Високий, нові продукти з'являються раз на 4 роки
Доля на ринку	На ринку України офіційно присутні і зареєстровані		

Таблиця 2. – Аналіз стейкгоल्дерів

Критерії оцінки	Оцінка стейкгоल्дерів		
	Міністерство внутрішніх справ України [28]	Міністерство оборони України [29]	Державна прикордонна служба України [30]
Досвід роботи	2 роки 5 місяців	5 років 2 місяців	3 роки 9 місяців
Імідж	Середній, присутні корупційні моменти, відносна довіра, і внутрішня і зовнішня, співпраця з міжнародними партнерами		
Тенденції розвитку	Повільні, потребують вкладень	Стрімкі, постійне покращення та осучаснення	Стрімкі, постійне покращення та осучаснення
Потенціал розвитку	Середній	Високий	Високий

Компанія «Vigilant Watchman» має організаційну структуру, де ключовими фігурами є генеральний директор (CEO), технічний директор (CTO), фінансовий директор, які керують стратегією, технологіями, продуктовою лінійкою та затратами на вищеперераховані процеси. Відділи розробки (R&D), виробництва, маркетингу, фінансів та підтримки клієнтів забезпечують повний цикл створення та просування тактичних годинників. Також компанія включає HR і юридичний відділи для управління персоналом і дотримання правових норм.



Рисунок 1. Організаційна структура «Vigilant Watchman».

Джерело: Розроблено автором.

**Місія компанії:** Створювати інноваційні, надійні та функціональні тактичні годинники, які допомагають професіоналам у екстремальних умовах досягати своїх цілей, забезпечуючи безпеку, точність та зручність використання.

**Ціль компанії:** Стати лідером на ринку тактичних пристроїв, пропонуючи високотехнологічні рішення з передовими функціями, які відповідають потребам військових, рятувальників та інших користувачів, що потребують найкращих інструментів у складних умовах.

Цей підрозділ висвітлює стратегічні напрями розвитку компанії «Vigilant Watchman», шляхом визначення ключових цілей і завдань. Постановка цілей є критично важливою для орієнтації діяльності компанії для досягнення успіху на ринку тактичних годинників. Цілі формують довгострокову стратегію розвитку, тоді як завдання - конкретні кроки, необхідні для досягнення цих цілей. Цей підрозділ визначає пріоритети компанії в сферах інновацій, якості продукції, розширення ринку та поліпшення взаємодії з клієнтами.

## **1.2. Постановка цілей та завдань**

Розробка операційної системи (ОС) для тактичного годинника – складний процес, що змушує вимагати глибокого розуміння як специфічних потреб військових та інших користувачів, так і сучасних технологій. Давайте розглянемо ключові завдання, які необхідно вирішити при розробці такої ОС:

### **1. Функціональність:**

- Базові функції годинника: Точний час, будильник, секундомір, таймер;
- Навігація: Компас, GPS, ГЛОНАСС, можливість завантаження карт;
- Комунікація: Bluetooth, Wi-Fi (для синхронізації з мобільними пристроями), можливість передачі даних у режимі реального часу;
- Датчики: Барометр, термометр, акселерометр, гіроскоп (для визначення фізичної активності, положення в просторі);

Спеціалізовані функції:

- Лічильник калорій;
- Моніторинг пульсу, артеріального тиску;
- Нічне бачення;
- Запис голосових нотаток;
- Функції для спеціальних операцій (наприклад, таймер підриву, кодування повідомлень).

## **2. Програмне забезпечення:**

- Інтерфейс користувача: Інтуїтивно зрозумілий, адаптований для роботи в екстремальних умовах, можливість керування годинником в рукавичках;
- Операційна система: Висока надійність, низьке енергоспоживання, можливість роботи в автономному режимі;
- Програмне забезпечення для розробки додатків: SDK (Software Development Kit) для створення додатків сторонніми розробниками.

## **3. Апаратне забезпечення:**

- Процесор: Висока продуктивність, низьке енергоспоживання;
- Пам'ять: Достатній об'єм оперативної та постійної пам'яті для зберігання даних та додатків;
- Дисплей: Яскравий, чіткий, з високою контрастністю, можливість роботи в різних умовах освітлення;
- Батарея: Довгий час автономної роботи, швидка зарядка;
- Корпус: Міцний, водонепроникний, ударостійкий.

## **4. Безпека:**

- Захист даних: Шифрування даних, захист від несанкціонованого доступу;
- Надійність: Захист від збоїв, можливість відновлення даних;
- Захист від перешкод: Стабільна робота в складних електромагнітних умовах.

## **5. Інтеграція з іншими системами:**

- Синхронізація з мобільними пристроями: Обмін даними, віддалене управління;
- Інтеграція з іншими гаджетами: Смартфони, фітнес-трекери, безпілотники;
- Сумісність зі стандартними протоколами: Bluetooth, ANT+, Wi-Fi.

## **6. Розробка та тестування:**

- Створення прототипу: Швидка розробка та тестування нових функцій;
- Тестування в реальних умовах: Перевірка працездатності годинника в екстремальних умовах;
- Оптимізація: Постійне вдосконалення програмного забезпечення та апаратного забезпечення.
- Ключові виклики при розробці ОС для тактичного годинника:
- Обмежені ресурси: Невеликий розмір екрану, обмежений об'єм пам'яті, обмежена потужність процесора;
- Високі вимоги до надійності: Годинник повинен працювати без збоїв в екстремальних умовах;
- Швидкий розвиток технологій: Необхідність постійно оновлювати програмне забезпечення та апаратне забезпечення.

Розробка ОС для тактичного годинника – складний і багатогранний процес, який вимагає залучення фахівців різних профілів: програмістів, електроніків, дизайнерів та військових експертів.

### **1.3. Визначення вимог до продукту**

Розробка сучасних тактичних годинників – це не просто інженерний процес, а складне поєднання технології та людських потреб. Ринок тактичних годинників, за оцінками аналітиків, демонструє стрімке зростання, що обумовлено як розвитком військових технологій, так і зростанням інтересу до активного відпочинку. Історично, годинники використовувалися не лише для визначення часу, а й як інструменти навігації та управління. Сучасні тактичні годинники успадкували цю багатофункціональність, додавши до неї новітні технології. Як зазначав відомий військовий стратег Карл фон Клаузевіц, "війна – це продовження політики іншими засобами" [5].

У контексті сучасних конфліктів, де точність і швидкість прийняття рішень мають вирішальне значення, тактичний годинник стає незамінним інструментом.

Тому розробка операційної системи для тактичного годинника "Vigilant Watchman" – це виклик, який стоїть перед сучасною мікроелектронікою. Мініатюризація електронних компонентів, розвиток бездротових технологій та зростаючі обчислювальні потужності створюють нові можливості для розробки інтелектуальних пристроїв. Однак, щоб реалізувати весь потенціал сучасних технологій, необхідно чітко визначити функціональні та нефункціональні вимоги до програмного забезпечення. Адже навіть найпотужніший процесор буде марним, якщо програмне забезпечення не відповідає потребам користувача.

Неправильно визначені вимоги можуть призвести до затримок у розробці, перевищенню бюджету та, як наслідок, до незадоволеності кінцевого користувача. Тому, на початковому етапі розробки операційної системи для тактичного годинника «Vigilant Watchman» було проведено детальний аналіз потреб потенційних користувачів та визначено комплекс вимог, яким має відповідати розроблюваний продукт. Від повноти і точності сформульованих вимог залежить не лише функціональність пристрою, а й безпека тих, хто ним користується.

**Функціональні вимоги** до годинника сформовані на основі потреб користувачів.

Годинник повинен забезпечувати точний хід часу в різних часових поясах та виконувати функції будильника, секундоміра і таймера із вібрацією та звуковим сигналом. Крім того, пристрій має мати вбудований календар з відображенням фаз Місяця та даними сонячного та місячного календарів. Для навігації передбачені компас, барометр, альтиметр з автоматичною калібровкою, а також вбудований GPS-приймач з високою точністю визначення координат. Годинник дозволить здійснювати навігацію за

маршрутом, створювати та зберігати треки, а також інтегруватися з онлайн та офлайн картами.

Для спеціалізованих потреб, таких як використання в польових умовах, годинник буде обладнаний нічним режимом з високою контрастністю екрану та червоним підсвічуванням. Також передбачені функції стеження за фізичною активністю, такі як крокомір і пульсометр. Для роботи з цифровими картами та іншими захищеними даними передбачений спеціальний модуль з функціями шифрування та захисту від несанкціонованого доступу. Для автономної роботи годинник буде мати модуль з оптимізацією енергоспоживання.

Щодо комунікаційних можливостей, годинник має підтримувати Bluetooth для синхронізації з мобільними пристроями та підключення зовнішніх датчиків, а також бездротові стандарти для передачі даних на великі відстані.

Інтерфейс користувача інтуїтивно зрозумілий, оптимізований для роботи в рукавичках і підтримує можливість налаштування під потреби користувача. Годинник підтримуватиме декілька мов, включаючи мови програмування, а також забезпечувати зручну візуалізацію даних.

**Нефункціональні вимоги** можна описати декількома наступними абзацами.

Годинник має бути надійний та стійкий до механічних пошкоджень, вологи, пилу та екстремальних температур. Також буде захищений від електромагнітних перешкод і забезпечує автоматичне резервне копіювання важливих даних.

Пристрій має бути ефективним, тобто швидко реагувати на команди користувача та мати оптимізоване енергоспоживання для тривалої роботи без підзарядки. Безпека даних користувача є пріоритетом, тому годинник має бути захищений від несанкціонованого доступу та шкідливих програм.

Годинник повинен бути сумісним з різними операційними системами мобільних пристроїв та різними типами датчиків і аксесуарів. Інтерфейс користувача має бути інтуїтивно зрозумілим і не вимагати тривалого навчання.

Годинник повинен бути розширюваним, тобто допускати встановлення додаткових програмних модулів та мати відкритий API для розробки сторонніх додатків. Крім того, операційна система годинника має бути масштабованою, доступною для розробки та модифікації, і супроводжуватися повною і детальною документацією для розробників і користувачів.

Також потрібно додати, що у Agile-підходах широкого розповсюдження набули User Case [6]. Тож, випадок використання (User Case) – це детальний опис взаємодії користувача з системою для досягнення конкретної мети. У контексті розробки операційної системи для тактичного годинника, use case допомагають визначити конкретні сценарії використання пристрою, які забезпечують необхідну функціональність для військових, спортсменів та туристів.

Для продукту операційної системи тактичного годинника було розроблено 5 основних сценаріїв використання, представлених у таблиці 3.

*Таблиця 3. Приклади Use Case для операційної системи.*

Use Case	Опис	Актор	Попередні умови	Основний сценарій	Альтернативний сценарій
Відображення часу	Користувач може легко перевірити точний час у будь-яких умовах, навіть при слабкому освітленні.	Користувач	Годинник увімкнений	1. Користувач натискає кнопку або піднімає руку. 2. Годинник показує час на дисплеї.	1. При слабкому освітленні годинник автоматично вмикає підсвітку або переходить в нічний режим відображення.
Вимірювання пульсу	Годинник дозволяє відстежувати частоту серцевих скорочень під час	Користувач	Годинник на зап'ясті користувача	1. Користувач активує функцію вимірювання пульсу.	1. Якщо пульс занадто високий, годинник відправляє попередження користувачу.

Use Case	Опис	Актор	Попередні умови	Основний сценарій	Альтернативний сценарій
	фізичних навантажень.			2. Годинник проводить вимірювання і відображає частоту пульсу на екрані.	
Навігація за допомогою GPS	Користувач може визначити своє місцезнаходження, прокласти маршрут і отримати вказівки на місцевості.	Користувач	GPS увімкнений	1. Користувач відкриває меню навігації. 2. Вводить кінцеву точку. 3. Годинник прокладає маршрут і надає покрокові інструкції.	1. Якщо сигнал GPS втрачено, годинник використовує останнє відоме місцезнаходження або відображає повідомлення про відсутність сигналу.
Сповіщення про низький заряд	Годинник попереджає користувача про необхідність підзарядки.	Користувач	Низький заряд батареї	1. Коли заряд падає до критичного рівня, годинник відправляє сповіщення про низький заряд.	1. Годинник автоматично переходить у режим енергозбереження для подовження часу роботи.
Підрахунок кроків та активності	Годинник відстежує фізичну активність користувача і відображає статистику.	Користувач	Годинник на зап'ясті користувача	1. Годинник автоматично рахує кроки та фізичну активність користувача. 2. Користувач може переглянути статистику на екрані годинника або у додатку на смартфоні.	1. У разі тривалої неактивності годинник може надіслати нагадування про необхідність рухатися.

Кожен з цих use case детально описує, як користувач взаємодіє з годинником, які дії він виконує і які результати очікує отримати. Це дозволяє

розробникам чітко визначити функціональні вимоги до пристрою та створити продукт, який буде корисним і зручним у використанні.

Окрім вищеперерахованого, потрібно звернути увагу на верифікацію вимог, тому що для операційної системи тактичного годинника це критичний етап у процесі розробки. Вона гарантує, що розроблюваний продукт повністю відповідає, заздалегідь, визначеним функціональним та нефункціональним вимогам. З огляду на специфіку використання такого годинника (екстремальні умови, висока точність, надійність), верифікація набуває особливої ваги.

Мета верифікації - переконатися, що всі функції годинника працюють коректно, а його характеристики відповідають очікуванням користувача та вимогам замовника.

Для верифікації вимог необхідно провести ряд тестів, включаючи:

- Функціональне тестування: Перевірка кожної функції на відповідність вимогам;
- Тестування на надійність: Перевірка стійкості ОС до різних несприятливих факторів;
- Тестування на безпеку: Перевірка захищеності ОС від різних типів атак;
- Тестування продуктивності: Перевірка швидкодії ОС в різних умовах;
- Тестування зручності використання: Оцінка зручності користувацького інтерфейсу.

## **Висновки розділу 1**

Аналіз проведеного дослідження та визначення вимог до операційної системи тактичних годинників "Vigilant Watchman" дозволяє зробити кілька важливих висновків. По-перше, ринок тактичних годинників демонструє стійке зростання. По-друге, годинники "Vigilant Watchman" позиціонуються як багатофункціональний інструмент для професійного використання. По-третє, вимоги до операційної системи є високими, вона повинна забезпечувати

не лише базові функції годинника, але й спеціалізовані можливості для тактичного використання. Ключовими факторами успіху проєкту є надійність, функціональність, ергономічність, безпека та використання сучасних технологій. На основі проведеного аналізу сформульовані стратегічні цілі проєкту: створення продукту, який задовольнить потреби як професійних військових, так і любителів активного відпочинку, зайняття лідируючої позиції на ринку та створення екосистеми пристроїв. Для досягнення цих цілей необхідно продовжити деталізацію вимог до операційної системи, розробити детальний план розробки програмного забезпечення, створити прототип пристрою та забезпечити ефективне маркетингове просування продукту.

## **РОЗДІЛ 2. ГНУЧКЕ УПРАВЛІННЯ СТВОРЕННЯМ ПРОДУКТУ «VW OS»**

Даний розділ має на меті представити ключові аспекти процесу розробки цього продукту, що відбувається в умовах сучасного динамічного середовища. У наш час розробка операційних систем для тактичних пристроїв, зокрема годинників, вимагає високого рівня адаптивності та швидкого реагування на зміни вимог користувачів та технологічного контексту. Саме тому гнучкі методології, такі як Agile, стають основою для ефективного управління такими проєктами.

Процес створення операційної системи «VW OS» для тактичного годинника включає тісну взаємодію між командами розробників, замовником та користувачами, що дозволяє швидко реагувати на зворотний зв'язок і коригувати напрямок розробки відповідно до нових вимог. Використання нових інструментів (JIRA, Confluence), забезпечує прозорість у плануванні, моніторингу прогресу та вирішенні проблем, що виникають у ході реалізації проєкту.

Цей розділ описує основні етапи створення операційної системи, обґрунтовується підхід до управління проєктом у гнучкому середовищі та аналізується роль команди у забезпеченні успішної розробки продукту. Особлива увага приділяється важливості регулярної комунікації, ідентифікації ризиків та впровадженню постійного вдосконалення продукту на основі ітераційного підходу.

### **2.1. Обґрунтування вибору фрейму гнучкого управління проєктом**

Гнучке управління проєктами – сучасний підхід до організації роботи, який дозволяє командам бути більш адаптивними та ефективними, особливо в умовах постійних змін. На відміну від традиційних методів, де весь проєкт

детально планується наперед, гнучкі підходи передбачають ітеративну розробку та постійну адаптацію до нових вимог і умов.

Ключові ідеї гнучкого управління:

- *Короткі ітерації*: Проєкт розбивається на короткі ітерації (спринти), що дозволяє регулярно отримувати працюючий продукт та отримувати зворотний зв'язок від замовника;
- *Фокус на цінності*: Команда концентрується на створенні найбільш цінних функціональних можливостей для користувача;
- *Співпраця*: Тісна співпраця між командою розробки та замовником забезпечує розуміння потреб та очікувань;
- *Адаптація до змін*: Гнучкі методи дозволяють легко вносити зміни в проєкт відповідно до нових вимог або обставин;
- *Постійне вдосконалення*: Команда регулярно аналізує свою роботу та вносить зміни для підвищення ефективності.

Чому гнучке управління є ефективним?:

- *Швидкість*: Завдяки коротким ітераціям продукт виводиться на ринок швидше;
- *Якість*: Постійний зворотний зв'язок дозволяє покращувати якість продукту на кожному етапі;
- *Гнучкість*: Можливість адаптуватися до змін робить проєкт більш стійким до ризиків;
- *Задоволеність замовника*: Замовник має можливість впливати на процес розробки та отримувати бажаний результат.

Гнучкі методології стали важливим інструментом для управління проєктами, зокрема в сфері розробки програмного забезпечення. Вони допомагають командам адаптуватися до змінних умов, швидко реагувати на нові вимоги і ефективно співпрацювати. Серед найпопулярніших гнучких підходів виділяються такі методології, як Scrum, Kanban та Extreme

Programming (XP) та інші [7]. Кожна з них має свої особливості, але всі вони базуються на загальних принципах гнучкого управління.

Кожна з цих методологій має свої переваги. Scrum надає чітку структуру для виконання завдань і добре підходить для проєктів з чіткими ітераціями. Kanban є більш гнучким підходом, орієнтованим на безперервне вдосконалення і управління потоком завдань. Extreme Programming відзначається увагою до деталей та технічної досконалості, що робить його ідеальним для проєктів, де критичним є якість програмного забезпечення. Ці методології використовуються для різних типів проєктів та команд, але всі вони спрямовані на підвищення продуктивності, гнучкості та якості результатів.

Підсумовуючи, гнучке управління – ефективний інструмент для управління проєктами, який дозволяє командам бути більш продуктивними, адаптивними та орієнтованими на задоволення потреб клієнтів.

Розглядаючи різні методології гнучкого управління, вибір пав на Scrum. Чому? Тому що вважається кращою методологією для багатьох проєктів через низку особливостей, які роблять його ефективним. Наприклад, він забезпечує високу гнучкість та адаптивність.

Scrum – це один з найпопулярніших фреймворків гнучкої методології, який ідеально підходить для розробки операційної системи до тактичного годинника [8].

Використання гнучкого управління, як Scrum, при розробці операційної системи для тактичного годинника є надзвичайно важливим з кількох причин:

- Високі вимоги до адаптивності: Вимоги до тактичного годинника можуть змінюватися протягом розробки, оскільки військові та інші спеціалізовані користувачі часто мають нові потреби. Гнучкі методи (Scrum) дозволяють швидко адаптуватися до змін, не переробляючи весь проєкт;

- Складність проекту: Годинник має виконувати безліч функцій і інтегрувати різні технології, що ускладнює розробку. Гнучкі методології (Scrum) дозволяють розбити проєкт на невеликі ітерації, що полегшує управління складністю;
- Швидкість виведення продукту на ринок: Конкуренція на ринку електроніки висока, тому важливо швидко випустити продукт. Гнучкі методи (Scrum) дозволяють отримувати перші версії продукту швидко і постійно їх вдосконалювати;
- Покращення якості продукту: Регулярні ретроспективи та фокус на створенні цінності для користувача дозволяють постійно покращувати якість продукту;
- Збільшення задоволеності замовника: Прозорість процесу розробки і можливість вносити зміни «на ходу» дозволяють замовнику відчувати себе залученим до проєкту і бути впевненим у результаті.

Ключові переваги Scrum для розробки ОС для тактичного годинника:

- Спринти: Розбивка проєкту на короткі ітерації дозволяє швидко отримувати працюючий продукт;
- Беклог: Створення пріоритезованого списку задач дозволяє фокусуватися на найбільш важливих функціональних можливостях;
- Ролі: Чітко визначені ролі (продукт-оунер, скрам-майстер, команда розробки) забезпечують ефективну співпрацю;
- Щоденні зустрічі: Регулярні зустрічі дозволяють синхронізувати роботу команди та оперативно вирішувати проблеми.

Використання Scrum при розробці ОС для тактичного годинника дозволяє:

- Зменшити ризики: Раннє виявлення та усунення проблем;
- Покращити якість продукту: Постійне вдосконалення на основі зворотного зв'язку;

- Збільшити задоволеність замовника: Прозорість, гнучкість, фокус на цінності;
- Швидке виведення продукту на ринок: Короткі ітерації та швидка адаптація до змін.

## **2.2. Планування проєкту з розробки продукту «VW OS»**

Розробка операційної системи для тактичного годинника – це складний і багатоаспектний процес, який вимагає ретельного планування та координації зусиль всієї команди. Scrum, як одна з найпопулярніших гнучких методологій, дозволяє ефективно проводити процес управління таким проєктом.

Планування розробки операційної системи для тактичного годинника це складний процес, який вимагає точного визначення вимог, чіткої структури та гнучкості для адаптації до змін. Оскільки такий годинник використовується в екстремальних умовах і має виконувати широкий спектр функцій, планування стає особливо важливим. Тож, далі представлені основні етапи планування [9].

Перше завдання – визначення продукту. Потрібно визначити мінімально життєздатний продукт (MVP) [10], який набір функцій повинен бути реалізований в першій версії годинника. Пріоритезація функцій допоможе визначити, які з них є найважливішими для користувача. Далі йде створення беклогу – детального списку всіх необхідних функцій і завдань.

Наступний етап – створення команди. Визначаються ролі, такі як продукт-оунер, скрам-майстер та команда розробки. Після цього потрібно набрати команду, залучаючи фахівців з необхідними компетенціями та досвідом, таких як програмісти, тестувальники та дизайнери.

Далі йде планування спринтів. Спочатку потрібно визначити тривалість спринту, яка зазвичай триває від 2 до 4 тижнів. Після цього планується спринт: обираються завдання з беклогу для реалізації в поточному спринті. Оцінка складності завдань виконується за допомогою методів оцінки, наприклад, Story Points [11], для визначення обсягу роботи.

Під час проведення спринтів щоденно проводяться короткі зустрічі (Daily Scrum) [8] для обговорення прогресу та вирішення проблем. В кінці кожного спринту демонструються результати замовнику, де показуються закінчені функції. Після завершення кожного спринту проводиться ретроспектива для аналізу пройденого спринту, визначення сильних і слабких сторін та визначення напрямків для покращення.

На завершення, важливим є адаптація та зміни. Регулярний перегляд беклогу є необхідним, оскільки пріоритети можуть змінюватися, і беклог потрібно регулярно оновлювати. Також можливі зміни плану спринту, якщо виникають непередбачені обставини, що впливають на його виконання.

Ефективне планування є ключовим фактором успішної розробки будь-якого програмного забезпечення, особливо такого специфічного, як операційна система для тактичного годинника. Для забезпечення чіткої координації, відстеження прогресу та оптимізації процесу розробки використовується широкий спектр інструментів.

Для ефективного управління проектом можна використовувати різноманітні інструменти:

- Дошки для планування: Trello, Jira, Asana, Confluence.
- Системи контролю версій: Git.
- Інструменти для проведення зустрічей: Zoom, Google Meet, WebEx, Teams, Skype.

Враховуючи відомості про Scrum [8] була створена Scrum-команда, що працює над розробкою операційної системи «VW OS» тактичного годинника. Вона має ряд специфічних особливостей, пов'язаних з вимогами до такого продукту і сконцентрована на надійності, функціональності, компактності та енергоефективності. Склад Scrum-команди показано на Рисунку 5.

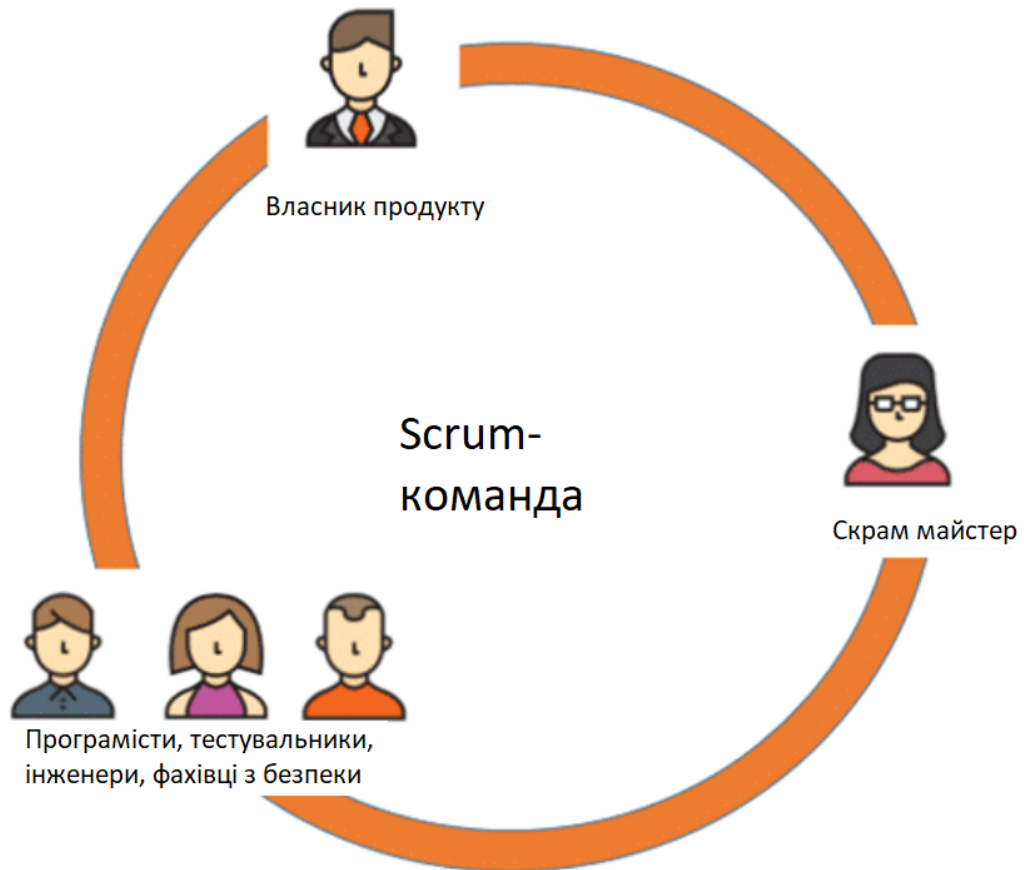


Рисунок 5. Класична структура Scrum-команди.

Джерело: Розроблено автором.

Використовуючи Посібник зі Скраму [8], Scrum-командою було утверджено всі необхідні наради, які дають можливість формально перевіряти та адаптувати Скрам артефакти і створені вони для того, щоб забезпечити прозорість. При чому, відмові від таких нарад призведе до втрати можливостей перевірки та адаптації. Проведення цих нарад дозволяє систематизувати процес розробки та звести до мінімуму потребу у нарад, які не передбачені Скрамом.

Спринт є основою Скраму, тому що в ході Спринту прості ідеї перетворюються на цінність. Спринт це нарада (подія) з фіксованою тривалістю. Тому Scrum-команда ініціювала Планування спринтів,

визначивши саме ту роботу, яку слід виконати за Спринти. В результаті чого, отримала план, до створення кого була залучена уся Scrum-команда.

Scrum-команда була поставлена у короткі терміни розробки операційної системи, тому було створено беклог вимог, котрі складаються з 6 інкрементів (ітерацій) і відповідно 6 спринтів, котрі розбиті у часі, внесені у Jira і показані на Рисунку 6.

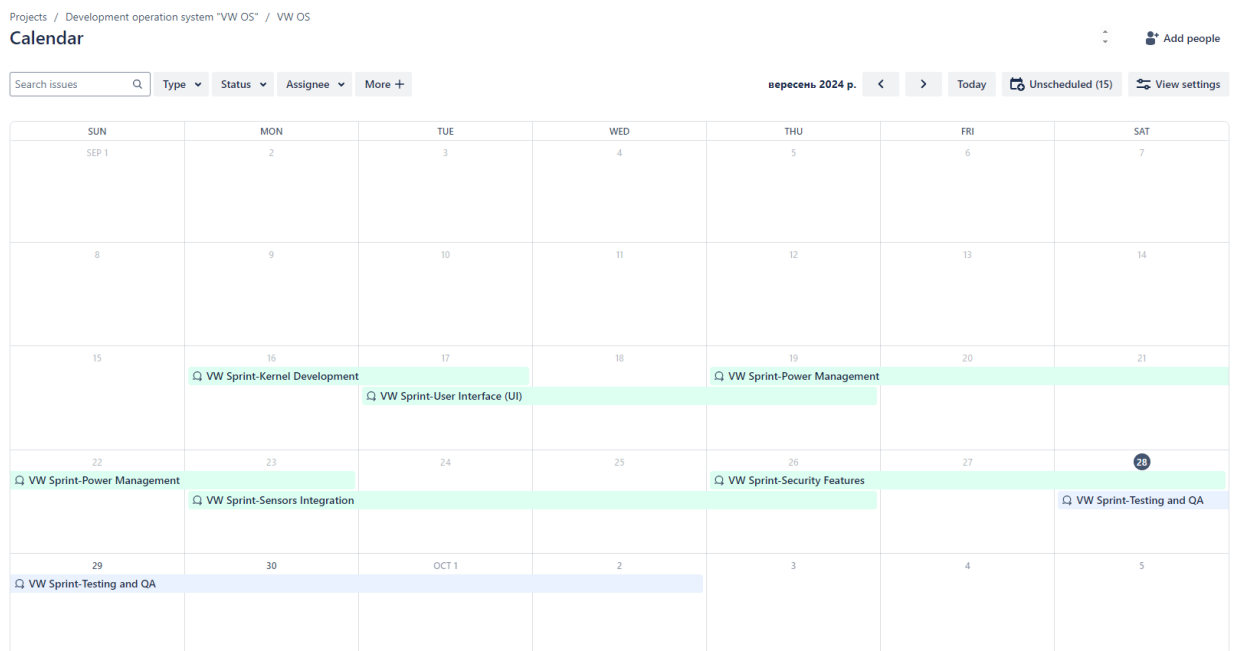


Рисунок 6. Календар ітерацій/спринтів.

Джерело: Розроблено автором за допомогою Jira.

1 Спринт це Створення ядра, до якого увійшло 4 Сторі та 2 Завдання, продемонстровано на Рисунку 7.

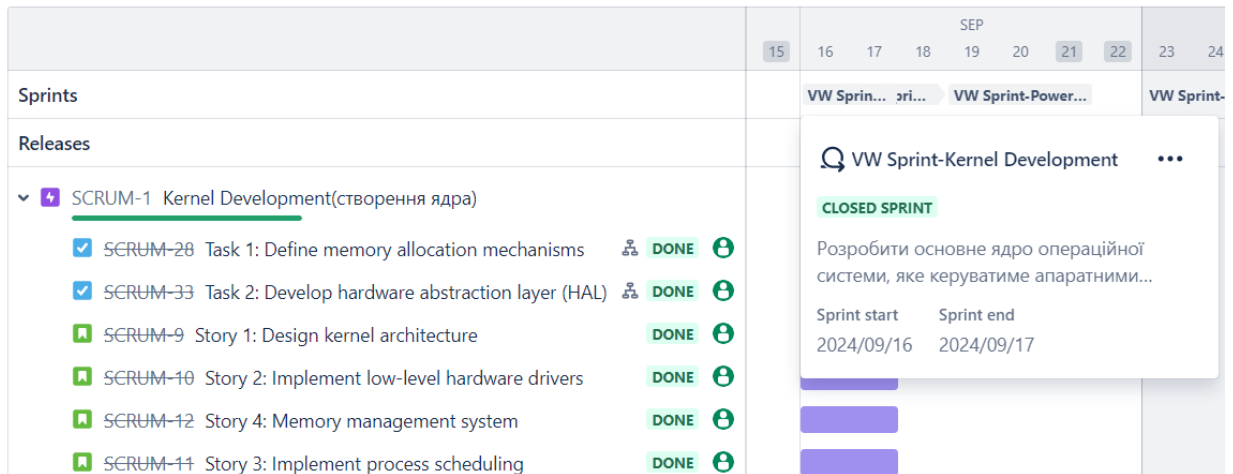


Рисунок 7. 1 Спринт.

Джерело: Розроблено автором за допомогою Jira.

2 Спринт – розробка Користувачького інтерфейсу, який складається з 3 Сторей та 2 завдань, показані на Рисунку 8.

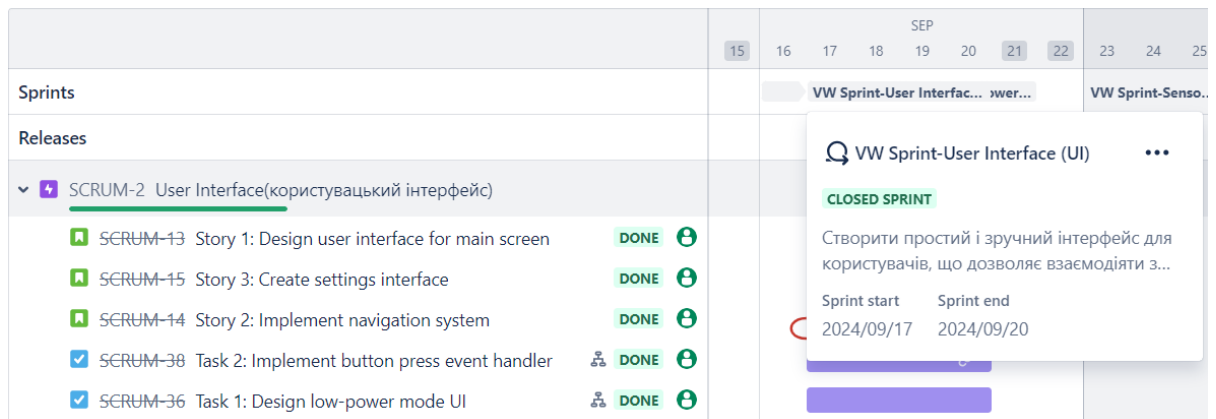


Рисунок 8. 2 Спринт.

Джерело: Розроблено автором за допомогою Jira.

3 Спринт – розробка Управління енергією, який включає 2 Сторі та 2 Завдання, продемонстровані на Рисунку 9.

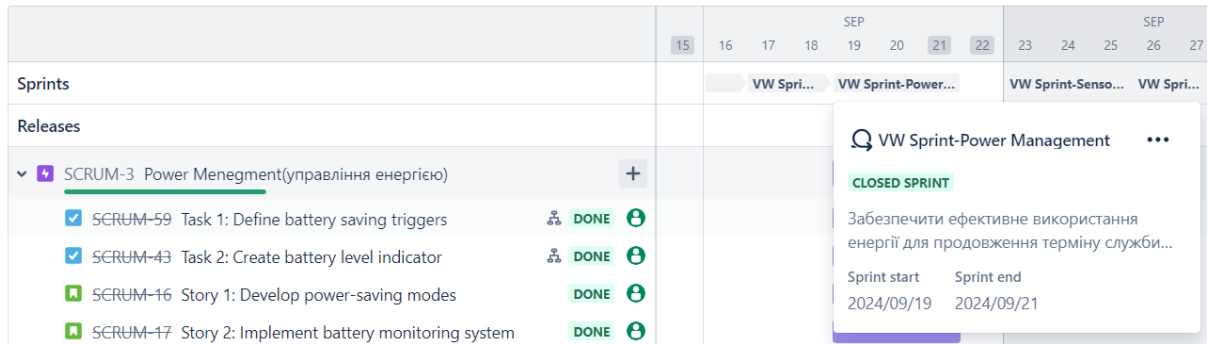


Рисунок 9. 3 Спринт.

Джерело: Розроблено автором за допомогою Jira.

4 Спринт – розробка Інтеграції з сенсорами, складається з 3 Сторей та 2 Завдань, показано на Рисунку 10.

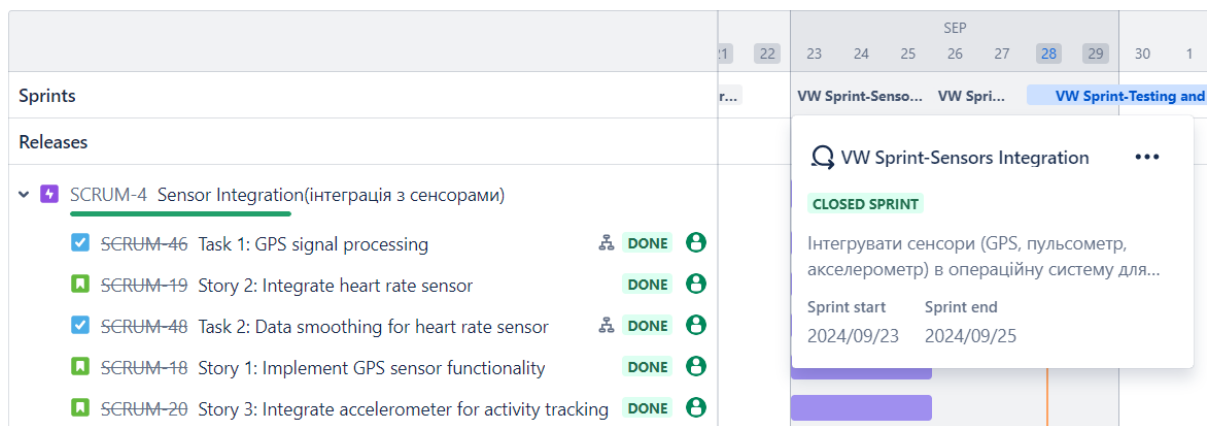


Рисунок 10. 4 Спринт.

Джерело: Розроблено автором за допомогою Jira.

5 Спринт – розробка Функцій безпеки, включає у себе 2 Сторі та 2 Завдання, показано на Рисунку 11.

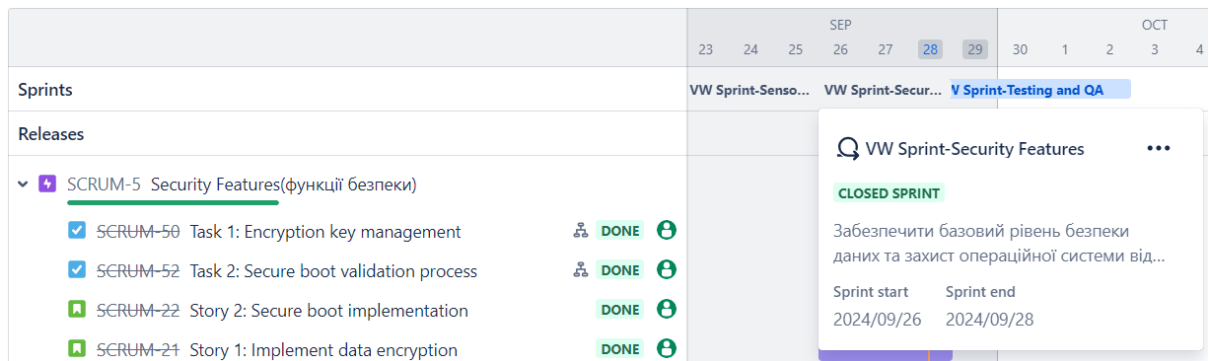


Рисунок 11. 5 Спринт.

Джерело: Розроблено автором за допомогою Jira.

6 Спринт, завершальний і об'ємний – Тестування та контроль якості, складається з 4 Сторей та 5 Завдань, продемонстровано на Рисунку 12.

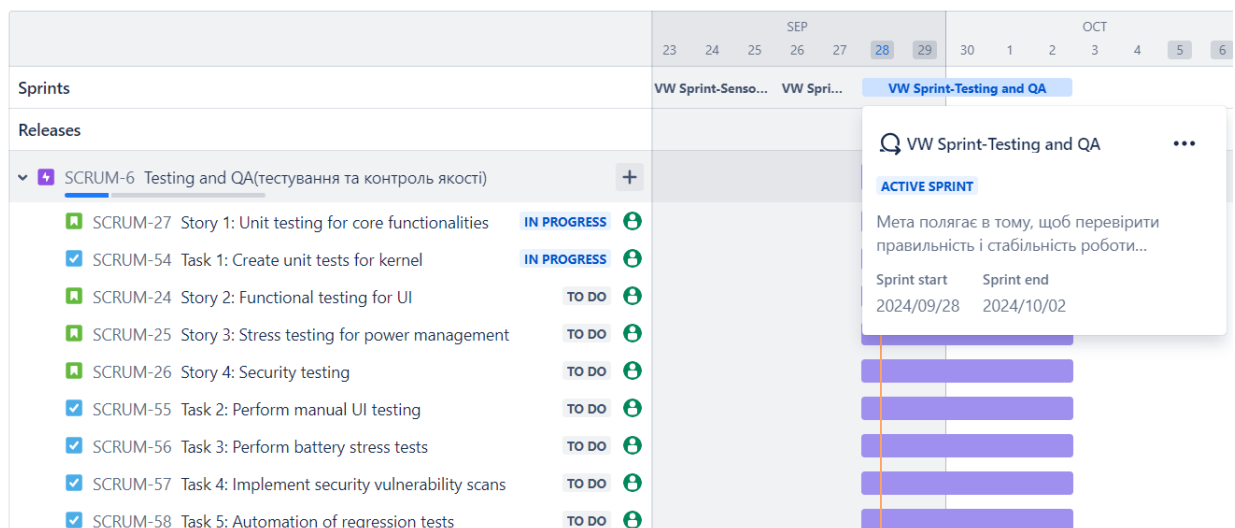


Рисунок 12. 6 Спринт.

Джерело: Розроблено автором за допомогою Jira.

В ході роботи Scrum-команди, було застосовано інструмент Jira. Jira - інструмент для команд, які використовують Scrum. Вона пропонує широкий спектр функцій, які допомагають ефективно керувати проєктами на всіх етапах - від планування та розробки до аналізу результатів. Завдяки Jira, Scrum-команда створила детальні беклоги, візуалізувала роботу на дошках,

відстежувала прогрес за допомогою діаграм та дозволила ефективно спілкуватися між собою. Крім того, Jira легко інтегрувалася з іншими інструментами, що використовуються в команді, та створила власні звіти для аналізу даних. В результаті, використання Jira підвищила прозорість, ефективність та якість роботи команди.

Окрім вищеперерахованих учасників та робіт потрібно спланувати бюджет проєкту. Тож використовуючи приблизну уявну середню вартість роботи спеціалістів, можна продемонструвати на Рисунку 12.

- Скрам майстер – 20 USD/годину;
- Власник продукту – 25 USD/годину;
  - Програмісти – 30 USD/годину;
  - Тестувальники – 20 USD/годину;
- Інженери/дизайнери – 20 USD/годину;
- Фахівці з безпеки - 20 USD/годину.

Рисунок 12. Середня вартість роботи спеціалістів.

Джерело: Розроблено автором.

Розрахована вартість роботи для кожного ресурсу показана на Рисунку 13.

- Скрам майстер – 180 годин \* 20 USD/годину = 3600 USD;
- Власник продукту – 200 годин \* 25 USD/годину = 5000 USD;
  - Програмісти – 220 годин \* 30 USD/годину = 6600 USD;
  - Тестувальники – 180 години \* 20 USD/годину = 3600 USD;
- Інженери/дизайнери – 160 годин \* 20 USD/годину = 3200 USD;
- Фахівці з безпеки – 160 годин \* 20 USD/годину = 3200 USD.

Рисунок 13. Розрахована вартість роботи.

Джерело: Розроблено автором.

Враховуючи загальну вартість роботи та хостингу, в бюджет проєкту потрібно закласти не менше 27000 USD для 220 години роботи та хостингу на 1 рік.

### **2.3. Моніторинг виконання проєкту**

Дана фаза проєкту складається з декількох ключових етапів, як показники стану виконання проєкту, зміна стейкхолдерів та аналіз результатів.

#### **Стан виконання проєкту.**

Стан виконання проєкту — це поточний етап або прогрес, якого досяг проєкт на шляху до завершення. Він відображає, наскільки проєкт наближається до досягнення своїх цілей, враховуючи заплановані завдання, строки та ресурси. Стан виконання проєкту дає уявлення про те, чи дотримуються встановлені дедлайни, чи виникають проблеми або затримки, і чи необхідно вносити коригування в процес.

Крім того, для ефективного моніторингу стану виконання проєкту було використано спеціалізовану програмну систему Jira. Вона допомагає стежити за кожним етапом проєкту та забезпечувати його відповідність плану.

Тож, Рисунок 14 показує інструментарій Jira для відображення стану виконаних завдань у спринтах, що в свою чергу несе наглядний огляд успішності чи то Епіків, чи Сторей.

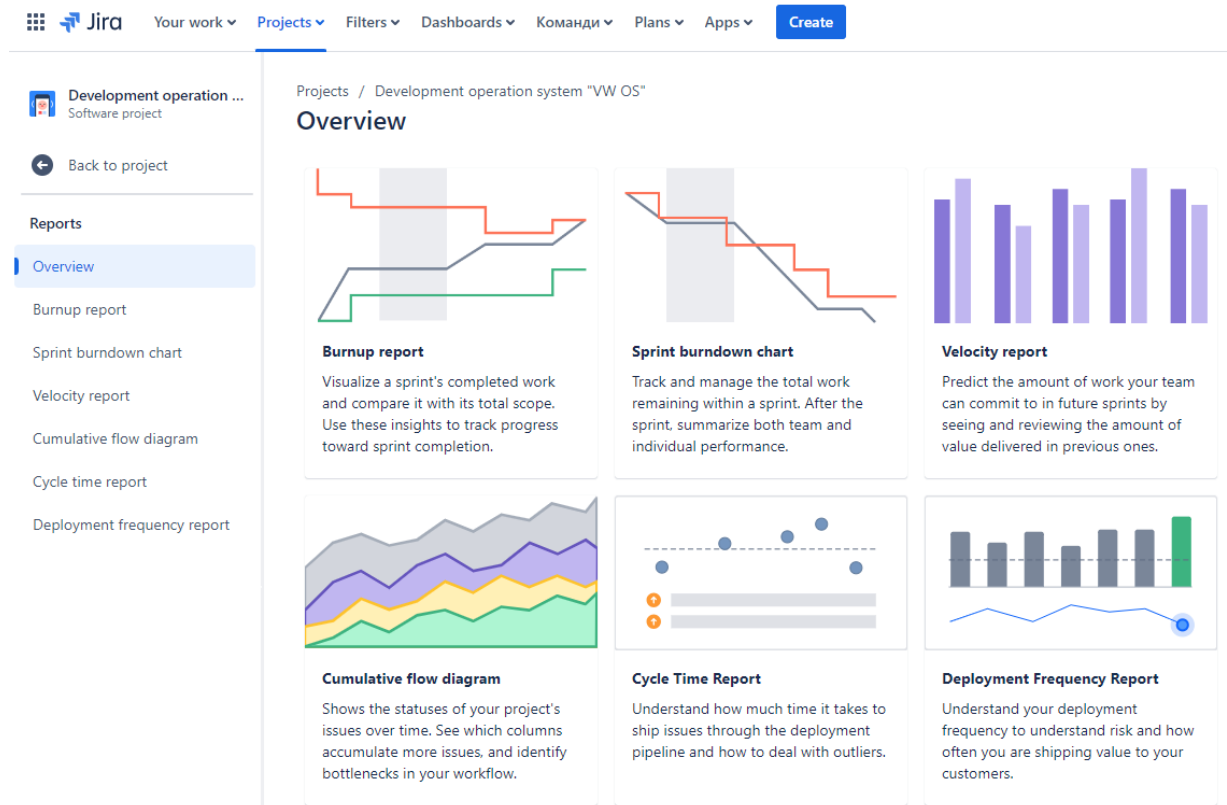


Рисунок 14. Інструментрй Jira для відображення успішності Спринта проєкту.

Джерело: розроблено автором.

Для наглядності було вирішено продемонструвати закінчення одного з Спринтів проєкту і відобразити його за допомогою «Sprint burndown chart» Рисунок 15. Sprint burndown chart це візуальне відображення прогресу команди протягом спринту. Він показує, який обсяг роботи (завдань) залишається зробити до кінця спринту. Він допоміг візуалізувати прогрес, виявити потенційні проблеми і приймати обґрунтовані рішення для успішного завершення спринту.



Рисунок 15. Sprint burndown chart.

Джерело: розроблено автором.

Серед усього інструментарію, потрібно виділити «Velocity chart». Velocity chart дозволяє показати з якою швидкістю команда працює над завданнями протягом спринту і чи вона виконала усі завдання, які були покладені на неї у спринті. Рисунок 16 дозволяє продемонструвати не один такий спринт і швидкість команди.

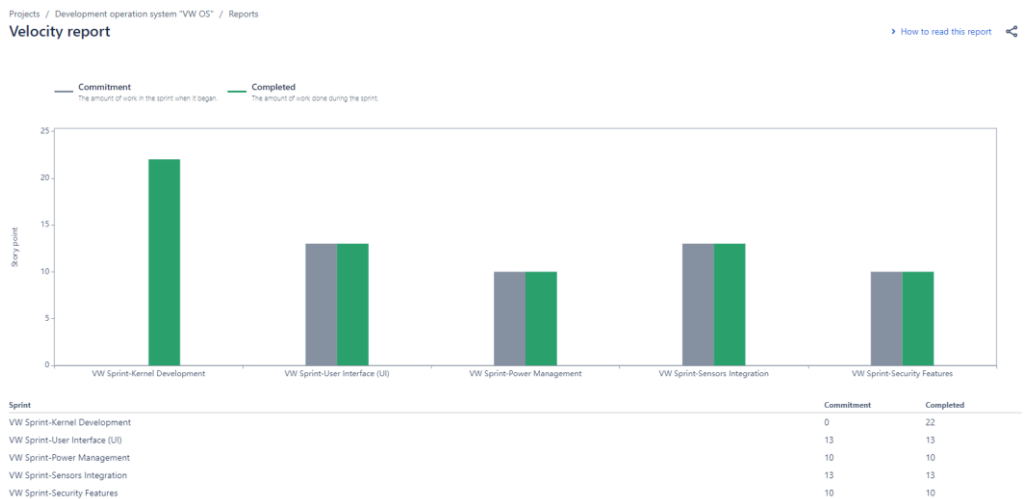


Рисунок 16. Velocity chart.

Джерело: розроблено автором.

Потрібно звернути увагу на те, що Jira дозволяє під'єднати до усього Беклогу план, котрий буде показувати узагальнену кількість Епіків або Сторей, залежно від налаштувань, що дозволяє показати проєкт у цілому, якщо потрібно у «стиислому» форматі показати прогрес. План дозволяє не лише команді, а й замовнику, отримати загальне уявлення про завдання та ініціативи і зв'язок між ними і як вони сприяють досягненню кінцевої мети. Рисунок 17 демонструє інструментарій «Plan».

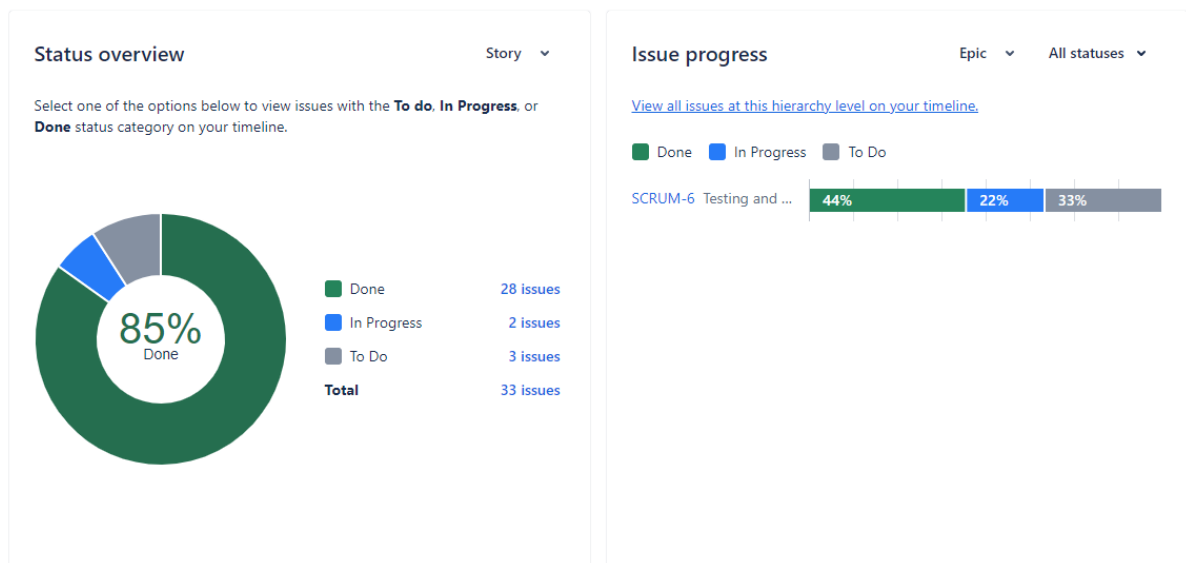


Рисунок 17. План усього проєкту.

Джерело: розроблено автором.

### **Зміна стейкголерів.**

Склад команди змінився, зменшившись на одного члена команди – тестувальника. Таким чином 1 тестувальник, а також функції тестувальника було перекладено на фахівця з безпеки у котрого був відповідний досвід, що дозволив перенести задачі 2 тестувальника на нього. Планування проєкту врахувало це і дало змогу впоратись з запланованою роботою.

### **Аналіз результатів.**

Аналіз результатів з Jira дозволив оцінити ефективність команди, процесів розробки та якість отриманого продукту, а також виявити потенційні

області для вдосконалення. Варто зазначити, що інструментарій Jira дозволив визначити середній час виконання завдань різних типів, кількість дефектів та підвищив надійність, допомогла оптимізувати роботу, підвищити мотивацію, а також отримані дані дали змогу прийняти рішення щодо подальшого розвитку продукту.

## **Висновки розділу 2**

Застосування Scrum для розробки операційної системи до тактичного годинника дозволило ефективно управляти проєктом, забезпечити високу якість продукту і задовольнити вимоги замовника. Регулярний перегляд і адаптація плану, а також тісна співпраця всіх членів команди є ключовими факторами успіху такого проєкту. Використання Jira для розробки операційної системи забезпечило детальний огляд процесу розробки, дозволило виявити потенційні проблеми та шляхи вдосконалення.

## **РОЗДІЛ 3. РЕЗУЛЬТАТИ РОБОТИ ТА ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ КОМАНДИ З РОЗРОБКИ ПРОДУКТУ «VW OS»**

### **3.1. Огляд інкременту продукту**

У цьому розділі представлені основні результати роботи команди з розробки операційної системи для тактичного годинника, а також заходи, що були вжиті для підвищення її ефективності. Успіх проєкту залежав не тільки від технічної реалізації, а й від оптимізації командної взаємодії та поліпшення процесів розробки. Аналіз показників продуктивності, змін у робочих процесах і впроваджених методик дає змогу оцінити, наскільки команда змогла досягти поставлених цілей, а також які підходи сприяли підвищенню ефективності та якості розробки. Цей розділ також висвітлює ключові аспекти вдосконалення комунікацій, планування й управління ресурсами, які допомогли оптимізувати роботу над операційною системою.

Огляд інкременту розробки операційної системи тактичного годинника починається з детального аналізу основних цілей, які команда ставила перед собою на цьому етапі розробки. Основними завданнями було створення базової функціональності для управління енергоспоживанням, значне покращення інтерфейсу користувача для тактичних режимів, інтеграція важливих сенсорів (GPS, альтиметр, компас) із системними процесами, а також оптимізація швидкодії системи при виконанні ключових завдань годинника в тактичних умовах. Цей інкремент став одним із ключових для подальшого розвитку проєкту, оскільки його завдання були спрямовані на створення базової архітектури та функціоналу для важливих аспектів операційної системи.

Як результат роботи в рамках цього інкременту, було досягнуто кілька важливих технічних і функціональних успіхів. Одним із найбільш помітних результатів стало успішне створення модуля для контролю споживання енергії, який дозволяє годиннику функціонувати в економному режимі без

втрати ключових функцій. Це дозволить користувачам продовжувати користуватися пристроєм протягом тривалого часу навіть у складних умовах без доступу до зарядки. Важливою частиною інкременту також стало завершення інтеграції сенсорів, таких як GPS, альтиметр і компас, із системними процесами, що дало змогу забезпечити їх стабільну роботу в різних режимах. Також було проведено суттєве оновлення інтерфейсу користувача: тепер взаємодія з годинником стала значно зручнішою та інтуїтивною. Додатково було впроваджено функцію швидкого перемикання між тактичними режимами, що значно спрощує процес використання годинника в умовах, коли необхідна миттєва реакція.

Тестування продукту підтвердило, що базові функції системи працюють відповідно до очікувань. Було проведено перевірку сенсорів у різних екстремальних умовах для забезпечення їх витривалості та надійності. Команда протестувала роботу системи під впливом високих та низьких температур, перевіряючи точність роботи сенсорів у складних кліматичних умовах. Користувацькі випробування також надали багато корисного зворотного зв'язку, завдяки якому інтерфейс користувача був доопрацьований для покращення навігації та зручності використання. Це дозволило ще більше підвищити якість взаємодії користувачів із пристроєм.

Однак попри досягнення, було виявлено кілька недоліків. Однією з найбільших проблем стала нестабільність роботи сенсорів при низьких температурах. Це викликало певні труднощі з точністю їх показань, тому в майбутніх інкрементах необхідно буде зосередитися на покращенні цієї функції. Також було виявлено затримку при перемиканні режимів, що негативно вплинуло на користувацький досвід. Важливо також зазначити, що проблема з підвищеним енергоспоживанням під час активного використання GPS залишилася нерозв'язаною, оскільки споживання енергії перевищувало заплановані показники, що також потребує додаткових зусиль для оптимізації.

Мета наступного інкременту зосередиться на подальшій оптимізації роботи сенсорів, особливо щодо зниження енергоспоживання. Необхідно вирішити питання із затримкою при перемиканні режимів та провести оптимізацію операційної системи для підвищення її швидкодії. Також планується провести додаткові тести у реальних умовах для перевірки стабільності роботи годинника, зокрема у польових умовах. Це дозволить отримати важливі дані для майбутніх етапів розробки і вчасно виявити потенційні проблеми.

Досягнення цих результатів стали можливими завдяки тісній співпраці всієї команди. Щоденні Scrum-зустрічі значно покращили комунікацію між інженерами й тестувальниками, що допомогло швидко вирішувати виникаючі проблеми. Важливим фактором стала тісна взаємодія з дизайнерами, яка сприяла швидкому впровадженню змін в інтерфейсі користувача. Завдяки регулярним покращенням у процесах інтеграції та розгортання нових функцій, команда змогла значно зменшити час на виправлення помилок та покращити загальний робочий процес, що відобразилося на якості продукту.

Оцінка ризиків на наступний етап показала, що питання енергоспоживання є одним із ключових для забезпечення тривалої роботи годинника в польових умовах. Тому вирішення цієї проблеми буде пріоритетним завданням. Крім того, інтеграція нових сенсорів або модулів у майбутніх інкрементах може створити певні труднощі, що вимагає детального планування і підготовки.

### **3.2. Ретроспектива роботи команди**

Ретроспектива [12] роботи команди при розробці операційної системи - важливий етап, що дозволяє оцінити виконану роботу, виявити успіхи та проблеми, а також розробити стратегії для подальшого вдосконалення.

На початку ретроспективи команда зібралася для обговорення різних аспектів процесу розробки. Це дозволило всім учасникам поділитися своїми

думками, враженнями та ідеями. Розглядалися як позитивні моменти, так і ті, що потребують поліпшення.

Серед позитивних, одним із головних успіхів стало досягнення командної синергії. Взаємодія між розробниками, тестувальниками та дизайнерами покращилася, завдяки регулярним зустрічам і комунікаційним інструментам. Кожен член команди відчував свою роль у процесі, що сприяло підвищенню мотивації та залученості. Зокрема, ефективне використання щоденних Scrum-зустрічей дозволило швидше виявляти та вирішувати проблеми, а також оперативно реагувати на зміни у вимогах.

Команда також відзначила значний прогрес у впровадженні нових функцій. Всі заплановані завдання були виконані вчасно, а випущені функції відповідають вимогам замовника. Це свідчить про високий рівень організації роботи та чітке розуміння цілей проєкту.

Що стосується викликів та проблем, то під час ретроспективи були виявлені й проблеми, які потребують уваги. Наприклад, деякі учасники команди відзначили, що недостатня документація ускладнювала процес інтеграції нових функцій. Це призводило до непорозумінь і затримок у роботі. Учасники запропонували впровадити систему документації, що дозволить легше слідкувати за змінами та зрозуміти контекст виконаних завдань.

Ще однією проблемою стало енергоспоживання, яке перевищувало очікувані показники під час активного використання сенсорів. Це вимагало додаткових зусиль для оптимізації системи, але також призвело до підвищення стресу в команді. У ході обговорення вирішили, що необхідно внести корективи у планування, щоб залишити більше часу для тестування і вдосконалення енергетичної ефективності.

Було заплановано, що відповідно до виявлених проблем, команда визначила кілька ключових напрямків для покращення у наступних інкрементах. По-перше, буде запроваджено більш структурований підхід до документації, щоб забезпечити доступність інформації для всіх учасників. По-

друге, необхідно зосередитися на оптимізації енергоспоживання, що передбачає проведення додаткових тестів і модифікацій.

Крім того, команда вирішила продовжити покращення комунікації та співпраці між різними підрозділами, щоб забезпечити ще ефективнішу інтеграцію нових функцій та зменшити ризики виникнення проблем у процесі розробки.

Тож, ретроспектива стала важливим інструментом для підведення підсумків роботи команди над розробкою операційної системи тактичного годинника. Завдяки відкритому обговоренню, команда змогла виявити як успіхи, так і виклики, з якими вона зіткнулася. Отримані висновки стануть основою для подальшого вдосконалення процесу розробки та підвищення ефективності команди у цілому.

### **3.3. Сучасний інструментарій менеджера в Agile-середовищі**

Сучасний інструментарій менеджера в Agile-середовищі включає різноманітні інструменти та практики, які допомагають командам адаптуватися до швидко змінюваних умов, поліпшити співпрацю та забезпечити високу продуктивність. Цей інструментарій варіюється в залежності від специфіки проєкту, проте існують загальні категорії та конкретні рішення, які широко використовуються в Agile-розробці.

Одним із ключових аспектів роботи Agile-менеджера є управління проєктами. Для цього використовуються інструменти, які дозволяють планувати, відстежувати та координувати роботу команди. Наприклад, Jira (яка і використовувалась у проєкті) є одним з найпопулярніших інструментів для управління проєктами, підтримуючи Scrum і Kanban. З його допомогою можна створювати та керувати завданнями, планувати спринти, відстежувати прогрес і генерувати звіти. Trello [13] та Asana [14] також популярні інструменти, які допомагають візуально управляти завданнями, термінами виконання та відповідальністю.

У Agile-оточенні, комунікація має критичне значення для успіху команди. Сучасні комунікаційні інструменти, такі як Slack [15] та Microsoft Teams, забезпечують швидку та ефективну взаємодію між учасниками команди. Slack дозволяє обмінюватися повідомленнями, створювати канали для різних тем та проєктів, а Microsoft Teams об'єднує чати, відеоконференції та інтеграцію з іншими сервісами Microsoft, що робить його зручним для організації зустрічей та спільної роботи над документами. Також Zoom [16] став популярною платформою для відеоконференцій, яка використовується для проведення онлайн зустрічей та вебінарів.

Автоматизація процесів розробки та безперервна інтеграція є важливими аспектами Agile. Інструменти, такі як Git [17] для контролю версій, і CircleCI [18] для швидкого тестування та розгортання, сприяють швидшому внесенню змін до продукту і зменшенню кількості помилок.

Управління вимогами є ще одним важливим аспектом, і для цього використовуються такі інструменти, як Confluence [19] для документування та спільної роботи, а також інструменти для створення карт користувачьких історій, що допомагають командам візуалізувати процеси та зосередитися на потребах користувачів.

Виходячи з вищеперерахованого, сучасний інструментарій менеджера в Agile-середовищі складається з широкого спектра інструментів для управління проєктами, комунікації, автоматизації, аналітики та управління вимогами. Вибір правильних інструментів залежить від специфіки команди та проєкту, проте їх ефективне використання може значно підвищити продуктивність команди, поліпшити співпрацю та адаптивність до змін, що, в свою чергу, сприяє успіху Agile-проєктів і задоволенню потреб замовників.

Окрім широкого інструментарію менеджера потрібно виділити, що у розділі 3.2 були відмічено негативні фактори, з якими команда зіткнулася у сучасному зовнішньому середовищі діяльності. Управління в сучасних умовах супроводжується значною невизначеністю [21] та ризиками, що підвищує

вимоги до адаптивності команд. Це знайшло своє відображення у формуванні різноманітних чотирьохкомпонентних аббревіатур, які характеризують стан зовнішнього середовища, таких як SPOD, VUCA, BANI та DEST.

Кожна з цих моделей включає власні фактори, які впливають на діяльність команд та підприємств. Проте, як свідчать останні дослідження, кількість нових аббревіатур зростає, оскільки з'являються нові фактори впливу. Це робить доцільним розгляд таких моделей у комплексі. Однією зі спроб узагальнення цих підходів - робота В. А. Рача, О. В. Россошанської та О. М. Медведєвої, які запропонували спрощену модель сучасного світу [22]. Згідно з цією моделлю, зовнішнє середовище змінюється непередбачувано, неоднозначно та в умовах нестабільності.

T E S D  
 B A N I  
 T U N A  
 V U S A

Рисунок 18. Приклад чотирьохкомпонентної аббревіатури.

Джерело: [22].

Для позначення всіх факторів впливу в рамках цієї моделі використовується термін «НЕ-фактори». До таких факторів відносяться: неоднозначність, невизначеність, нестійкість та непізнаваність. В умовах дії НЕ-факторів ключовим питанням управління стає прийняття рішень менеджером, яке має вирішальний вплив на успішність проєкту.

Більшість стандартів управління проєктами відносять до управління ризиками. Наприклад, PMBOK трактує ризик як невизначену подію або умову, що може позитивно або негативно вплинути на досягнення цілей проєкту, зокрема, його обсяг, розклад, вартість і якість.

Інший підхід до трактування ризику, який ми підтримуємо, запропонований Борульком Н. А., що спирається на терміносистему, розроблену у роботі Д. В. Рача. [23]. Відповідно до цієї системи, ризик визначається як стан діяльності об'єкта управління, що здійснюється в умовах недостатньої визначеності, коли можливий прояв подій, які можуть призвести до негативних відхилень від запланованого результату або очікуваної цінності [24].

Отже, при плануванні діяльності слід говорити про невизначенність. Тож, невизначеність при плануванні діяльності менеджера виникає через складність прогнозування майбутніх подій та умов, що впливають на виконання завдань і досягнення цілей. Менеджери, особливо в динамічних середовищах, таких як Agile або управління проєктами, часто зіштовхуються з такими факторами - непередбачувані зміни в вимогах, обмежені ресурси, зовнішні ринкові умови або вплив технологічних змін. Це ускладнює чітке визначення планів і цілей на майбутнє, вимагаючи від менеджерів гнучкості та готовності адаптуватися.

Однією з ключових причин невизначеності є відсутність повної інформації на момент планування. В процесі розробки або виконання проєкту можуть з'являтися нові дані, які змінюють попередні рішення або пріоритети. Для мінімізації ризиків невизначеності менеджеру необхідно використовувати такі підходи, як постійний моніторинг ситуації, перегляд планів та пріоритетів, залучення команди до пошуку рішень і обговорень, а також створення гнучких і адаптивних стратегій.

У плануванні важливо враховувати можливі сценарії розвитку подій, щоб мати альтернативні плани на випадок змін. Використання інструментів

для оцінки ризиків та управління ними, таких як SWOT-аналіз або матриця ризиків, допомагає менеджеру краще підготуватися до непередбачуваних ситуацій.

Завдання менеджера - ідентифікація загальних ризиків проєкту. При кожній ітерації спринта відбувається зміна станів менеджера від стану невизначеності, до стану недовизначеності та стану ризику і стану нової невизначеності.

Ідентифікація загальних ризиків проєкту є важливим етапом для менеджера, оскільки дозволяє передбачити потенційні проблеми, які можуть вплинути на досягнення цілей проєкту. Своєчасне виявлення ризиків допомагає знизити їхній негативний вплив і розробити стратегії для їхнього пом'якшення. Одним із ключових ризиків є зміна вимог, оскільки під час реалізації проєкту замовники можуть вносити нові коригування або вимоги, що призводить до затримок або перевищення бюджету. Також важливими є фінансові ризики, коли бюджет проєкту виявляється недостатнім або вартість матеріалів/послуг зростає. Це створює необхідність ретельного контролю витрат і передбачення можливих перевитрат.

Ризики часу також є важливим фактором, адже порушення термінів виконання проєкту може виникати через недооцінку необхідного часу для виконання завдань або затримки в постачанні ресурсів і послуг. Менеджер повинен враховувати можливі затримки та бути готовим до швидкого реагування. Крім того, технічні ризики можуть стати серйозною загрозою, особливо якщо проєкт передбачає використання нових або складних технологій. Непередбачувані технічні проблеми можуть спричинити необхідність додаткових ресурсів для їх вирішення.

Також важливо враховувати ризики, пов'язані з людським фактором. Відсутність компетентного персоналу, низька мотивація команди, конфлікти або недостатня комунікація можуть негативно вплинути на продуктивність і виконання завдань. Менеджеру потрібно звертати увагу на підтримку

команди, мотивацію та забезпечення належного рівня взаємодії між учасниками проєкту.

Зовнішні ризики, такі як економічні, політичні або регуляторні зміни, також можуть мати значний вплив на проєкт. Наприклад, зміни в законодавстві або нестабільність на ринку можуть вплинути на витрати або стратегії реалізації проєкту. Окрім цього, менеджеру варто враховувати ризики, пов'язані з постачальниками та партнерами, адже недотримання договорів, затримки в постачанні ресурсів чи несправності в роботі підрядників можуть серйозно вплинути на хід проєкту.

Для ефективного управління цими ризиками менеджеру потрібно регулярно аналізувати їх, залучати команду до обговорення потенційних загроз і розробляти плани дій, які допоможуть мінімізувати можливі наслідки.

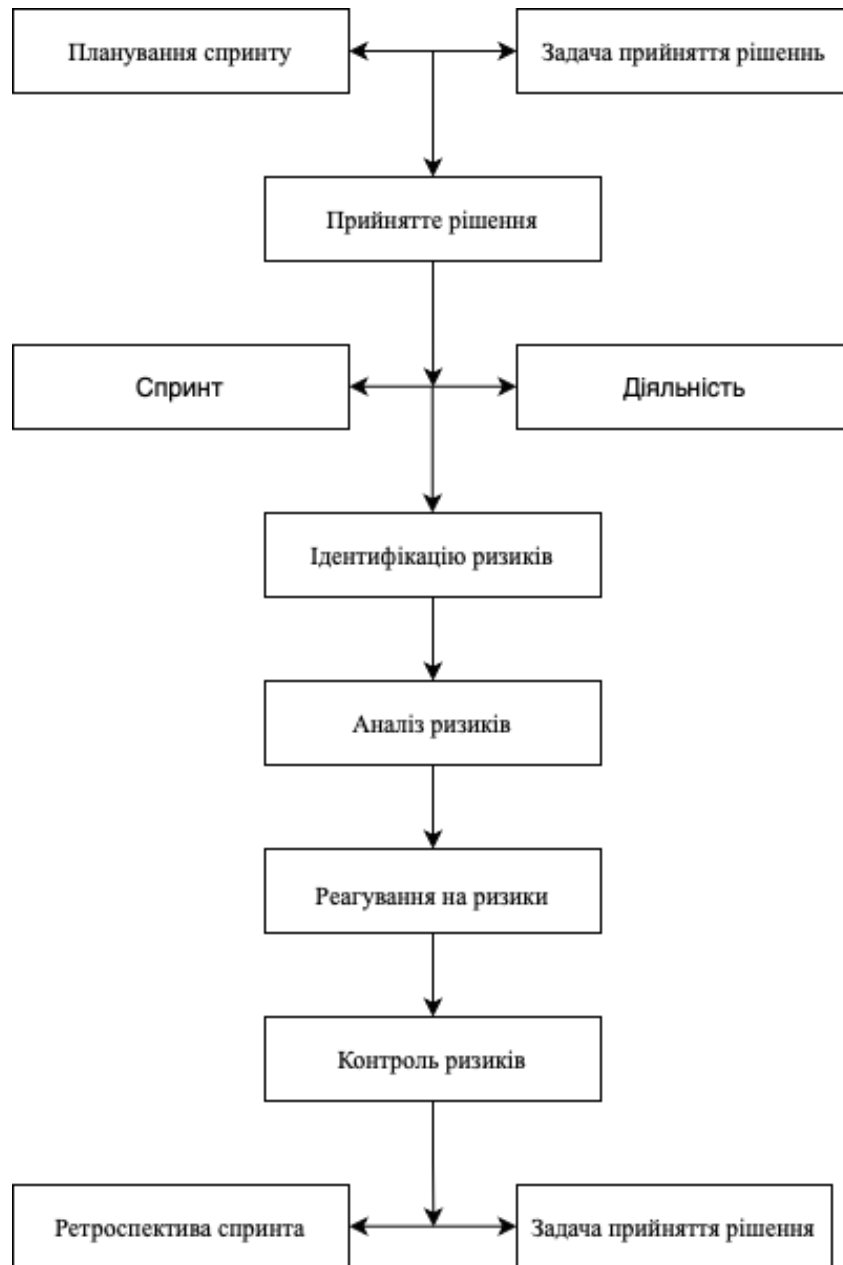


Рисунок 19. Алгоритм ідентифікації ризиків.

Джерело: Розроблено автором.

Нами представлено алгоритм (Рисунок 19), що складається з кількох етапів, кожен з яких відіграє важливу роль у процесі управління ризиками під час роботи в Agile-середовищі.

1. Планування спринту – це перший етап, де команда визначає завдання на спринт, оцінює обсяг роботи та встановлює цілі. Тут менеджери й команда мають враховувати можливі ризики, які можуть виникнути під час

реалізації спринту. На цьому етапі формується загальне бачення можливих ризиків, пов'язаних із завданнями та ресурсами.

2. Задача прийняття рішення – це момент, коли керівники або команда мають прийняти рішення щодо основних аспектів проєкту. Це може включати вибір методів виконання завдань, технологій або підходів. Важливо оцінити, чи можуть ці рішення призвести до появи ризиків.

3. Прийняття рішення – після визначення можливих ризиків і їхнього впливу, менеджери та команда приймають рішення, як діяти в умовах, що склалися. Вони можуть прийняти ризик, пом'якшити його або уникнути.

4. Спринт і діяльність – під час виконання спринту команда працює над завданнями, які були затверджені під час планування. У цей період важливо продовжувати моніторинг ризиків і негайно реагувати на зміни чи нові виклики, що можуть виникати.

5. Ідентифікація ризиків – постійний процес виявлення нових ризиків під час виконання завдань у спринті. Команда регулярно аналізує прогрес, обговорює можливі перешкоди та їх вплив на виконання проєкту.

6. Аналіз ризиків – після того, як ризики були ідентифіковані, проводиться їх детальний аналіз. Це включає оцінку їхнього ймовірного впливу на проєкт, рівня ризику та можливих наслідків для термінів, якості або бюджету проєкту.

7. Реагування на ризики – на основі проведеного аналізу команда приймає рішення щодо стратегії реагування на ризики. Це може бути план дій для мінімізації ризиків, або пошук способів уникнення проблем, пов'язаних із ризиками.

8. Контроль ризиків – упродовж усього процесу виконання спринту необхідно постійно контролювати наявні ризики та перевіряти, чи не з'являються нові. Це забезпечує готовність команди швидко відреагувати на непередбачувані ситуації та зберегти контроль над проєктом.

9. Ретроспектива спринта – на завершальному етапі команда проводить аналіз спринту, включаючи оцінку того, як були ідентифіковані, оцінені та вирішені ризики. Це дозволяє команді вчитися на помилках та краще підготуватися до майбутніх ризиків у наступних спринтах.

Такий підхід дозволяє більш точно оцінювати ризики і формувати ефективні стратегії їх мінімізації, що сприяє досягненню успіху в умовах сучасного зовнішнього середовища.

### **Висновки розділу 3**

Огляд інкременту, ретроспективи команди та сучасного інструментарію в Agile-середовищі свідчать про досягнення значних результатів у процесі розробки. Огляд інкременту показав, що команда успішно реалізувала заплановані функції, досягла ключових цілей та покращила продуктивність продукту. Цьому сприяв структурований підхід до планування та виконання завдань, що дозволив відповідати вимогам замовника.

Важливість командної взаємодії також була підтверджена під час ретроспективи, де команда продемонструвала здатність до відкритого обговорення та готовність ділитися ідеями. Це сприяло створенню атмосфери довіри, що є основою для подальшого розвитку команди.

Використання сучасних інструментів в Agile-середовищі, таких як Jira - суттєво вплинуло на успішне управління проєктом. Це допомогло оптимізувати робочі процеси, покращити координацію команди та зменшити ризик помилок. Орієнтація на користувача стала важливою темою в обох процесах — огляді інкременту та ретроспективі. Постійне вдосконалення продукту на основі потреб користувачів та зворотного зв'язку залишається ключовим елементом у досягненні успіху проєкту.

Завершення обговорення стало основою для чіткого планування подальших дій, що допоможе команді визначити наступні кроки, спрямовані на вдосконалення продукту та процесу розробки. У загальному контексті

успішна реалізація інкрементів, відкритість у ретроспективі та використання сучасного інструментарію вказують на готовність команди до адаптації, вдосконалення та досягнення нових вершин у роботі в Agile-середовищі.

У сучасному середовищі управління проєктами підвищується роль адаптивності команд через значну невизначеність і ризики. Невизначеність виникає через непередбачуваність подій, відсутність повної інформації та вплив різних змін, що ускладнює планування й управління. Для мінімізації ризиків менеджерам необхідно використовувати гнучкі стратегії, постійний моніторинг, оцінку ризиків і залучення команди до пошуку рішень. Це допомагає краще підготуватися до можливих змін та забезпечити успішність проєкту.

## ВИСНОВКИ

Застосування Scrum для розробки операційної системи «VW OS» до тактичного годинника «Vigilant Watchman» дозволило не лише ефективно управляти проектом, але й забезпечити високий рівень гнучкості та швидкого реагування на зміни. Основні переваги полягали у тому, що ітеративний підхід та спринти дозволили команді регулярно отримувати зворотний зв'язок, швидко адаптувати продукт до нових вимог і вдосконалювати його функціонал. Це особливо важливо для складних проектів, таких як розробка операційної системи для тактичного пристрою, де вимоги можуть змінюватися залежно від тестування та побажань користувачів.

Регулярний перегляд і адаптація плану, які є частиною Scrum-підходу, сприяли виявленню потенційних ризиків на ранніх етапах, що дозволило мінімізувати затримки та технічні проблеми. Тісна співпраця між усіма членами команди, включаючи розробників, тестувальників, дизайнерів і менеджерів проекту, забезпечила прозорість у процесі розробки і підвищила продуктивність. Кожен учасник мав змогу активно брати участь в обговоренні завдань і пропонувати власні ідеї, що сприяло кращому розумінню завдань та цілей.

Використання Jira для управління процесом розробки стало важливим інструментом для відслідковування прогресу та моніторингу кожного етапу. Ця платформа забезпечила детальний огляд завдань, дозволила відстежувати виконання спринтів, виявляти потенційні проблеми та шукати оптимальні шляхи їх вирішення. Крім того, Jira допомогла тримати всі зацікавлені сторони у курсі поточного стану проекту, що покращило комунікацію між командою та замовником.

В результаті, Scrum у поєднанні з Jira дозволив не тільки забезпечити високу якість продукту, але й створити операційну систему, яка максимально

відповідала вимогам замовника і кінцевих користувачів, що є важливим аспектом успіху проєкту.

Огляд інкременту, ретроспективи команди та сучасного інструментарію Agile продемонстрував суттєві досягнення у процесі розробки проєкту. Команда успішно реалізувала заплановані функції та досягла ключових цілей завдяки структурованому підходу до планування та виконання завдань, що дозволило відповідати вимогам замовника. Такий підхід сприяв підвищенню якості продукту та його продуктивності.

Ретроспектива показала, що відкриті обговорення та готовність команди ділитися ідеями допомогли створити атмосферу довіри й сприяли розвитку взаємодії всередині команди. Це забезпечило платформу для постійного вдосконалення як продукту, так і робочих процесів, що є важливою умовою для успішної роботи в Agile-середовищі.

У сучасних умовах управління проєктами зростає важливість адаптивності команд через високий рівень невизначеності та ризиків. Невизначеність зумовлена непередбачуваністю подій, нестачею повної інформації та постійними змінами, що ускладнюють процес планування та управління. Щоб знизити ризики, менеджерам необхідно застосовувати гнучкі підходи, постійно відстежувати ситуацію, оцінювати потенційні ризики та активно залучати команду до пошуку рішень. Це дозволяє ефективніше реагувати на зміни й забезпечувати успіх проєкту.

Представлений алгоритм ідентифікації ризиків дозволяє більш точно оцінювати ризики і формувати ефективні стратегії їх мінімізації, що сприяє досягненню успіху в умовах сучасного зовнішнього середовища.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Market, Smartwatch Statistics 2024 By Wearables, Technology, Device. URL: <https://scoop.market.us/smartwatch-statistics/>. (дата звернення 15.10.2024);
2. G-Shock, Casio. URL: <https://gshock.casio.com/europe/>. (дата звернення 15.10.2024);
3. Suunto. URL: <https://www.suunto.com/>. (дата звернення 15.10.2024);
4. Garmin. URL: <https://www.garmin.com/en-US/>. (дата звернення 15.10.2024);
5. Веденєєв Д. В., Гавриленко О. А., Кубіцький С. О. та ін. за заг. ред. Остроухова В. В., ЕВОЛЮЦІЯ ВОЄННОГО МИСТЕЦТВА, навчальний посібник. Київ: Національна академія СБУ, 2017. 276с. URL: [https://shron3.chtyvo.org.ua/Viedienieiev\\_Dmytro/Evoliutsiia\\_voiennoho\\_mystetstva\\_U\\_2\\_ch\\_Ch\\_1.pdf?PHPSESSID=2kev3to0n6ieva5bt1qg612110](https://shron3.chtyvo.org.ua/Viedienieiev_Dmytro/Evoliutsiia_voiennoho_mystetstva_U_2_ch_Ch_1.pdf?PHPSESSID=2kev3to0n6ieva5bt1qg612110). (дата звернення 15.10.2024);
6. QATestLab, Що таке USE CASE та для чого вони потрібні, URL: <https://training.qatestlab.com/blog/technical-articles/what-is-a-use-case-and-what-are-they-for/>. (дата звернення 15.10.2024);
7. AhaSlides, Agile методологія, URL: <https://ahaslides.com/uk/project-management/agile-methodology/>. (дата звернення 15.10.2024);
8. Кен Швабер та Джефф Сазерленд, Посібник зі Скраму, 2020, 17с. URL: <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Ukrainian.pdf>. (дата звернення 15.10.2024);
9. <https://iampm.club/ua/blog/etapi-upravlinnya-it-projektami-inicziacziya-projektu-dlya-menedzhera/>;
10. IAMPM, Етапи управління IT-проєктами: ініціація проєкту для менеджера. URL: <https://brainlab.com.ua/uk/blog-uk/shho-take-mvp->

- [yak-pravylnno-stvoryty-minimalno-zhyttyezdatnyj-produkt.](#) (дата звернення 15.10.2024);
11. Brainrain, Сторі поинти: як це працює. URL: <https://brainrain.com.ua/uk/story-points-2/>. (дата звернення 15.10.2024);
12. ATlassian, Ретроспективи Agile. URL: <https://www.atlassian.com/ru/agile/scrum/retrospectives>. (дата звернення 15.10.2024);
13. Trello. URL: <https://trello.com/>. (дата звернення 15.10.2024);
14. Asana. URL: <https://asana.com/>. (дата звернення 15.10.2024);
15. Slack. URL: <https://slack.com/>. (дата звернення 15.10.2024);
16. Zoom. URL: <https://www.zoom.com/ru>. (дата звернення 15.10.2024);
17. GIT. URL: <https://git-scm.com/>. (дата звернення 15.10.2024);
18. CircleCI. URL: <https://circleci.com/>. (дата звернення 15.10.2024);
19. Confluence. URL: <https://www.atlassian.com/software/confluence>. (дата звернення 15.10.2024);
20. Альтиметр. URL: <https://uk.wikipedia.org/wiki/%D0%90%D0%BB%D1%8C%D1%82%D0%B8%D0%BC%D0%B5%D1%82%D1%80>. (дата звернення 15.10.2024);
21. Щербак В. Г., Щербак О., В., Невизначеність і ризик у бізнес-діяльності промислового підприємства. Актуальні проблеми економіки №4(106), 2010. С. 156-161. URL: [https://er.knutd.edu.ua/bitstream/123456789/2706/1/20161025\\_110.pdf](https://er.knutd.edu.ua/bitstream/123456789/2706/1/20161025_110.pdf). (дата звернення 15.10.2024);
22. Rach, V., Rossoshanska, O., Medvedieva O. Management of different nature projects in the mind-economy of the bani-world. Information systems and innovative technologies in project and program

- management: collective monograph. Riga : ISMA, 2023. P. 290-308.  
DOI: <https://doi.org/10.30837/MMP.2023.290>;
23. Рач, Д.В. Управління невизначеністю та ризиками в проєкті: термінологічна основа / Д.В. Рач // Управління проєктами та розвиток виробництва: Зб. наук. пр. – Луганськ: вид-во СЛУ ім. В. Даля, 2013. – №3(47). – С. 146–164. URL: <http://www.pmdp.org.ua>;
24. Борулько Наталія Олександрівна, «Ідентифікація загальних ризиків в проєктах на основі м'якої інформації», Спецрада Д 26.056.0;
25. Чалюк Ю. О., Глобальний соціально-економічний розвиток в умовах VUCA, SPOD, DEST та BANI світу, Економіка та суспільство 36(36), 2020. С. 17. URL: [https://www.researchgate.net/publication/359634687\\_GLOBALNIJ\\_SO\\_CIALNO-EKONOMICNIJ\\_ROZVITOK\\_V\\_UMOVAN\\_VUCA\\_SPOD\\_DEST\\_TA\\_BANI\\_SVITU\\_GLOBAL\\_SOCIO-ECONOMIC\\_DEVELOPMENT\\_IN\\_VUCA\\_SPOD\\_DEST\\_AND\\_BANI\\_WORLD](https://www.researchgate.net/publication/359634687_GLOBALNIJ_SO_CIALNO-EKONOMICNIJ_ROZVITOK_V_UMOVAN_VUCA_SPOD_DEST_TA_BANI_SVITU_GLOBAL_SOCIO-ECONOMIC_DEVELOPMENT_IN_VUCA_SPOD_DEST_AND_BANI_WORLD). (дата звернення 15.10.2024);
26. Настанова ПМБОК. URL: <https://pmiukraine.org/pmbok7>. (дата звернення 15.10.2024);
27. SWOT-аналіз зі прикладами. URL: <https://esputnik.com/uk/blog/swot-analiz-iz-prikladami>. (дата звернення 15.10.2024);
28. Міністерство внутрішніх справ України. URL: <https://mvs.gov.ua/>. (дата звернення 15.10.2024);
29. Міністерство оборони України. URL: <https://www.mil.gov.ua/>. (дата звернення 15.10.2024);
30. Державна прикордонна служба України. URL: <https://mvs.gov.ua/contacts/state-border-guard-service>. (дата звернення 15.10.2024).

## ДОДАТОК А

### Термінологічний словник

1. SPOD – метод, який описує стан світового ладу за 4 ознаками: steady – стабільний, постійний, predictable – передбачуваний, очікуваний, ordinary – простий, постійний, definite – визначений, врегульований [25];
2. VUCA – метод, орієнтований на непередбачуваність зовнішніх та внутрішніх обставин розвитку суспільства з наступними ознаками: volatility - мінливість, uncertainty - невідомість, complexity -заплутаність, ambiguity – неясність [25];
3. DEST – новий метод, що має прийти на зміну VUCAз відповідними ознаками: disorder – суспільство, де панує безпорядок, egocentric – егоцентричність влади, suppression – замовчування інформації, turbulent – турбулентне бізнес-середовище [25];
4. BANI – структура для опису ситуацій, коли обставини не просто нестабільні, а хаотичні, коли результати не можна передбачити, бо вони несподівані. BANI, що означає: brittle – крихкість, ламкість, anxious – занепокоєння, тривога, nonlinear - нелінійність, incomprehensible – незбагненність, незрозумілість [25];
5. РМВОК - настільна книга менеджерів проєктів у всьому світі [26];
6. SWOT - надає можливість дізнатися про сильні та слабкі сторони компанії, а також зовнішні можливості й загрози [27];
7. Альтиметр – інструмент, який використовують для вимірювання висоти підйому або висоти на рівнем моря [20].