

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет кораблебудування імені адмірала Макарова

Навчально науковий інститут комп'ютерних наук та управління проектами

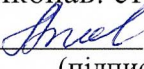
Кафедра управління проектами

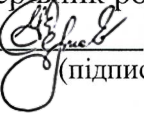
«Допущений до захисту»
Завідувач кафедри
 Чернов С.К.
«» 2023 р.

КВАЛІФІКАЦІЙНА РОБОТА
на здобуття ступеня вищої освіти «магістр»

на тему:

**Удосконалення моделі управління IT-проєкту для розробки web-
додатку**

Виконав: студент 6171м групи
 Аполонова А.В.
(підпис)

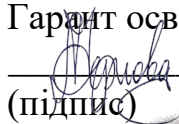
Керівник роботи: д.т.н., професор
 Чернов С.К.
(підпис)

Миколаїв – 2023 р.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет кораблебудування імені адмірала Макарова

Навчально науковий інститут комп'ютерних наук та управління проєктами

Кафедра управління проєктами
Спеціальність 122 Комп'ютерні науки
Освітня програма «Управління проєктами»

«ЗАТВЕРДЖУЮ»
Гарант освітньої програми
 Чернова Л.С.
(підпис)
«___» _____ 2023 р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

на здобуття ступеня вищої освіти «**магістр**»

Студенту Аполоновій Алесі Володимирівні
(Прізвище, ім'я, по батькові)

1. Тема роботи: Удосконалення моделі управління ІТ-проєкту для розробки web-додатку

Керівник роботи д.т.н., професор Чернов С.К.

Затверджені наказом ректора № 1215-уч від «13» жовтня 2023 року

2. Термін подання роботи: до 05.12.2023 року

3. Вихідні дані по роботі: дослідити, проаналізувати та удосконали існуючі моделі управління ІТ-проєкту для розробки web-додатку

4. Перелік питань, що належать до розробки (найменування розділів)

Розділ 1. Теоретичні аспекти в управлінні ІТ- проєктами

Розділ 2. Аналіз методології в управлінні ІТ- проєктами

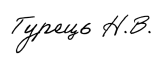
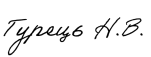
Розділ 3. Удосконалення моделі управління ІТ-проєкту

Розділ 4. Охорона праці

Розділ 5. Охорона навколишнього середовища

5. Перелік презентаційних матеріалів виконаний в програмі Power Point

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Розділ 4. Охорона праці	Гурець Н.В., старший викладач		
Розділ 5. Охорона навколишнього середовища	Гурець Н.В., старший викладач		

7. Дата видачі завдання 15.09.2023р.

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Вивчення літературних джерел з предмету дослідження	Вересень 2023	Виконано
2	Складання розгорнутого плану магістерської роботи та ознайомлення керівника з планом кваліфікаційної роботи	Вересень 2023	Виконано
3	Написання розділу 1	Вересень 2023	Виконано
4	Написання розділу 2	Жовтень 2023	Виконано
5	Написання розділу 3	Листопад 2023	Виконано
6	Написання розділів з Охорони праці та навколишнього середовища (4,5)	Грудень 2023	Виконано
7	Оформлення магістерської роботи	До 5 грудня	Виконано
8	Передача магістерської роботи рецензенту для рецензування	Грудень 2023	Виконано
9	Передача магістерської роботи науковому керівникові для написання відгуку	Грудень 2023	Виконано
10	Попередній захист магістерської роботи	5.12.2023	Виконано
11	Захист магістерської роботи	19.12.2023	Виконано

Студент


(підпис)

Аполонова А.В.
(ПІБ)

Керівник роботи


(підпис)

Чернов С.К.
(ПІБ)

АНОТАЦІЯ

Аполонова А.В. Удосконалення моделі управління ІТ-проєкту для розробки web-додатку. Кваліфікаційна робота зі спеціальності 122 «Комп'ютерні науки», освітньої програми «Управління проєктами». Миколаїв: НУК ім. адм. Макарова. 2023 рік. 113 сторінок.

У кваліфікаційній роботі було досліджено, проаналізовано та удосконалено модель управління ІТ-проєктом для розробки web-додатку. Результатом роботи стала вдосконалена, на основі методу Agile, модель управління, яка враховує усі аспекти процесу розробки Web-додатку програмного забезпечення, для подальшого використання у проєктуванні подібних проєктів.

Ключові слова: управління проєктами, ІТ-проєкти, управління командою, Agile, Kanban, Scrum, Waterfal, PERT, RACI, бюджет, якість.

ANNOTATION

Alesia Apolonova. Improvement of IT project management model for the development of web application. Qualified work in specialty 122 “Computer Science”, major program “Project Management”. Mykolaiv: Admiral Makarov National University of Shipbuilding. 2023. 113 pages.

A qualified robot conducted an investigation of the process of managing IT projects from the development of a Web browser. The result was a thoroughly developed, based on Agile, management model that covers all aspects of the process of developing Web-based software, for further development of similar projects.

Keywords: project management, IT projects, team management, Agile, Kanban, Scrum, Waterfall, PERT, RACI, budget, quality.

ЗМІСТ

	Стор.
ВСТУП	6
РОЗДІЛ 1. ТЕОРЕТИЧНІ АСПЕКТИ В УПРАВЛІННІ ІТ-ПРОЄКТАМИ	8
1.1. Сутність області знань та етапи в управлінні проєктами .	8
1.2. Загальна характеристика та класифікація ІТ- проєктів ...	24
1.3. Основні поняття методології в управлінні проєктами	27
Висновок до розділу 1	34
РОЗДІЛ 2. АНАЛІЗ МЕТОДОЛОГІЇ УПРАВЛІННЯ ІТ-ПРОЄКТІВ	35
2.1. Аналіз методів управління ІТ-проєктів сімейства Agile ...	35
2.2. Аналіз інструментів в управлінні ІТ-проєктів	45
2.3. Формування команди в ІТ-проєктах для розробки web-додатку	58
Висновок до розділу 2	65
РОЗДІЛ 3. УДОСКОНАЛЕННЯ МОДЕЛІ УПРАВЛІННЯ ІТ-ПРОЄКТУ	66
3.1. Вдосконалення моделей управління командою в ІТ-проєкті для розробки web-додатку програмного забезпечення на базі Agile	66
3.2. Удосконалення структури розподілу роботи ІТ-проєкту для розробки додатку програмного забезпечення	75
3.3. Впровадження розробленої моделі в соціальній сфері	87
Висновок до розділу 3	89
РОЗДІЛ 4. ОХОРОНА ПРАЦІ	90
4.1. Основні аспекти заходів з охорони праці користувачів ПК.....	90
4.2. Нормативна база при роботі за комп'ютером	91
4.3. Санітарно-гігієнічні норми і правила роботи за комп'ютером	92
Висновок до розділу 4	96
РОЗДІЛ 5. ОХОРОНА НАВКОЛИШНЬОГО СЕРЕДОВИЩА	97
5.1. Забрудненням навколишнього середовища	98
5.2. Заходи щодо запобігання забруднення навколишнього середовища	100
Висновок до розділу 5	100
ВИСНОВКИ	101
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ	102
Додатки	104

ВСТУП

Актуальність теми: тема розробки проєкту веб-додатку для соціальних компаній залишається актуальною в сучасному світі. Соціальні компанії відіграють ключову роль у вирішенні соціальних проблем, і наявність веб-додатків може сприяти їхній ефективності та здатності до впливу. Розвиток технологій веб-розробки дозволяє створювати потужні та інтуїтивно зрозумілі додатки, які полегшують роботу соціальних компаній та сприяють збору даних, комунікації з аудиторією та залученню ресурсів. Загальний інтерес громадськості до соціальних питань зростає, тому інструменти, що допомагають спрощувати та підтримувати роботу соціальних компаній, набувають великого значення.

Мета дослідження полягає в розробці та імплементації інноваційного web-додатку для соціальних компаній з використанням Agile методології розробки програмного забезпечення.

Для досягнення мети необхідно вирішити **основні задачі:**

- дослідити теоретичні аспекти в управлінні IT-проєктами;
- проаналізувати існуючі методології управління IT-проєктів;
- удосконалити моделі управління проєктом розробки web-додатків програмного забезпечення;
- обґрунтувати доцільність використання удосконаленої модель.

Об'єкт дослідження: процеси управління проєктами розробки web-додатків програмного забезпечення. **Предмет дослідження:** моделі та методи управління командою в проєктах розробки web-додатків програмного забезпечення.

Методи дослідження: узагальнення теоретичних основ, методи системного аналізу. Проведення дослідження та аналізу наукової літератури, що стосується web-додатків для соціальних компаній. Теоретичну і методологічну основу проведеного дослідження склали роботи вітчизняних і зарубіжних практик і фахівців в галузі управління проєктами.

У роботі також використовувалися міжнародні стандарти, статистичні дані, що детально розкривають стан і динаміку світового ринку в ІТ-сфері: наукові статті, огляди, матеріали міжнародних конференцій з інформаційних технологій та матеріали міжнародних організацій, що спеціалізуються на управлінні проєктами та розробки web-додатків, на самперед Інституту управління проєктами (PMI), Міжнародної Організації Стандартів (ISO).

Наукова новизна одержаних результатів: розроблена структурна модель управління проєктами web-додатку для програмного забезпечення на основі метода Agile, яка дозволяє вдосконалити процес управління проєктом, шляхом введення гнучких етапів роботи і дозволить підвищити якість кінцевого продукту. Отримала подальший розвиток визначення структури розробки проєкту програмного забезпечення, що дозволило скоротити термін виконання робіт і паралельно виконувати науково-дослідницьку роботу у компанії, завдяки збільшенню кількості одночасно виконуваних робіт.

Практичне значення одержаних результатів: впровадження виконаної методики зі створення мобільного додатку дозволить побудувати такий процес розробки управління проєктом, який дозволить суттєво зменшити ризики провалу проєкту, покращити якість кінцевого продукту та вдосконалити процес управління проєктами.

Робота складається із змісту, вступу, трьох основних розділів, розділу «Охорона праці» та розділу «Охорона навколишнього середовища», висновків, переліку посилань та додатків.

РОЗДІЛ 1. ТЕОРЕТИЧНІ АСПЕКТИ В УПРАВЛІННІ ІТ-ПРОЄКТАМИ

1.1. Сутність області знань та етапи в управлінні проєктами

Щоб зрозуміти термін «проєкт», було б доцільно ознайомитися з його визначенням. Це «тимчасове зусилля, спрямоване на створення унікальної послуги або результату проєкту». Іншими словами, проєкт – це починання, яке починається і закінчується в певний часовий проміжок, зосереджуючись на отриманні конкретного результату у формі заздалегідь встановлених результатів [0].

Управління проєктами — це використання конкретних знань, навичок, інструментів і методів для надання людям чогось цінного. Розробка програмного забезпечення для вдосконалення бізнес-процесів, будівництво будівлі, надання допомоги після стихійного лиха, розширення продажів на новий географічний ринок — усе це приклади [0].

Проєкти — це тимчасові спроби створити цінність за допомогою унікальних продуктів, послуг і процесів. Деякі проєкти створені для швидкого вирішення проблем. Інші вимагають подовжених часових рамок для отримання результатів, які не потребуватимуть серйозних покращень за межами запланованого технічного обслуговування (наприклад, дороги загального користування) [0].

Звичайно, деякі проєкти будуть сумішшю обох цих речей. Це стосується всього, починаючи від розробки нового програмного забезпечення і закінчуючи плануванням ліквідації наслідків стихійних лих. Тим не менш, це все дуже загальна інформація про те, що таке проєкт. Коли розбити їх більш конкретно, то видно, що проєкти є об'єднанням завдань, заходів і результатів, які повинні бути ретельно структуровані та виконані для досягнення бажаного результату.

Перш ніж буде досягнуто результату, кожен аспект проєкту повинен пройти через етапи ініціювання, планування та виконання. Цей процес

відомий як життєвий цикл управління проєктами, і він є джерелом життя успішних проєктів. Крім того, цей цикл дозволяє керівникам проєктів ретельно планувати кожне завдання та діяльність, щоб забезпечити найвищі шанси на успіх. Загалом, проєкт — це добре спланована робота, яка проходить протягом життєвого циклу з певним початком і кінцем.

Усі проєкти — це тимчасова спроба створити цінність за допомогою унікального продукту, послуги чи результату. Усі проєкти мають початок і кінець. У них є команда, бюджет, графік і набір очікувань, яким команда повинна відповідати. Кожен проєкт є унікальним і відрізняється від звичайних операцій — поточної діяльності організації, — оскільки проєкти завершуються після досягнення мети.

Зміна характеру роботи через технологічний прогрес, глобалізацію та інші фактори означає, що все частіше робота організовується навколо проєктів, а команди об'єднуються на основі навичок, необхідних для виконання конкретних завдань.

Деякі проєкти потрібні для швидкого вирішення проблем із розумінням того, що протягом певного періоду часу будуть внесені покращення. Інші проєкти мають довшу тривалість і/або створюють продукт або інший результат, який не потребуватиме значних покращень за межами запланованого технічного обслуговування, наприклад, шосе.

Ще інші будуть сумішшю обох цих типів проєктів. Професіонали проєкту використовують різноманітні навички та знання, щоб залучати та мотивувати інших для досягнення цілей проєкту. Професіонали проєктів мають вирішальне значення для успіху проєктів, і вони дуже затребувані, щоб допомогти організаціям досягти їхніх цілей.

Нині загальновизнано, що базові знання з управління проєктами можуть стати цінними для людей, які виконують різноманітні ролі в різноманітних справах. Навички управління проєктами можуть допомогти молодому студенту, який працює над науковим проєктом, досягти успіху, або керівнику компанії вирішити особисті суперечки. Ці навички можуть

допомогти медсестрі оптимізувати зміни змін, щоб покращити час реагування пацієнтів у своїй палаті. Вони можуть допомогти ІТ-фахівцю розробити інноваційне програмне забезпечення в рекордно короткі терміни або допомогти урядовій установі покращити послуги, які вони надають, економнішим способом.

Проекти мають певні визначальні характеристики. Вони унікальні, специфічні, цілеспрямовані та обмежені у часі. Хоча специфіка кожного починання унікальна, є певні елементи, в яких всі вони спільні. Ось короткий перелік найважливіших:

- мета;
- критерії успіху;
- вимоги;
- команда;
- обсяг робіт;
- бюджет;
- ресурси;
- розклад.

Без цих аспектів було б важко відслідковувати прогрес, оцінювати навантаження та часові рамки, розраховувати витрати або перевіряти результат.

Підприємства використовують різні способи підходу до управління проектами, також відомі як методології управління проектами. На практичному рівні вибір методології має величезні наслідки. Кожен із них має різні правила, процеси та інструменти.

Природно, проекти не обов'язково стосуються фізичних об'єктів. Управління проектами завжди присутнє в розробці програмного забезпечення, тому розглянуто приклад з цієї галузі. У програмному забезпеченні проєкт може бути новою функцією для одного з хмарних рішень.

Організації будь-якого розміру виконують кілька проєктів одночасно. Незважаючи на те, що це окремі організації, деякі з них обмежуються однією сферою діяльності. Щоб краще регулювати дії в цих сферах, потрібна сходинка вище проєкту. Ось тут програма вступає в дію.

У визначенні Програми зазначено, що це «група пов'язаних проєктів, якими скоординовано керують для отримання переваг і контролю, недоступних при індивідуальному управлінні ними». Таким чином, програма має на меті зібрати всі відповідні проєкти та цілісно контролювати їх. Крім того, впровадження програми полегшує визначення того, чи надають проєкти бажані можливості для бізнесу чи вирішують проблеми. Як і проєкти, програми є тимчасовими. Як тільки завершується останній фрагмент, програма завершується.

Можна визначити чотири основні риси, спільні для всіх програм:

- великий;
- довгий строк;
- генеральний;
- стратегічний;

Перша характеристика стосується того факту, що програми включають кілька проєктів. Тому логічно, що масштаби програми будуть широкими. Другий також впливає з багатьох елементів у програмі. Переваги для всього бізнесу, як правило, займають багато часу. Тому керівники програм повинні готуватися до того, що така ініціатива триватиме довго.

Хоча проєкти є специфічними та зосереджені на окремих результатах, охоплюючи програма має бути достатньо загальною, щоб охоплювати всі проєкти в певній галузі.

Програми підтримують виконання стратегічних цілей організації. Іншими словами, обсяг програми стосується ширшої картини бізнесу порівняно з результатами проєкту нижчого рівня.

Інститут управління проектами визначає портфоліо як «сукупність» проектів, програм та іншої роботи, згрупованої разом для полегшення ефективного управління цією роботою для досягнення стратегічних цілей бізнесу. Іншими словами, портфоліо – це набір різноманітних ініціатив під однією парасолькою.

Управління портфелем – це безперервний процес. Оскільки це стосується всіх цілей бізнесу, нові програми та проекти включаються до портфоліо, коли вони з'являються. Важливо, що елементи портфоліо повинні узгоджуватися з довгостроковою стратегією та цілями бізнесу.

Ключові процеси в управлінні портфелем. Нагляд за цілим портфоліо програм і проектів є важким завданням, незалежно від розміру бізнесу. Однак існують процеси, які допомагають керівникам і менеджерам портфелів приймати правильні рішення щодо нагляду за портфелем.

Зазначено, що усі три терміни стосуються управління ініціативами в організації. Отже, варто підсумувати, що вони пов'язані один з одним у такому порядку:

1. Портфоліо: всеохоплююче, що складається з програм, проектів та інших типів ініціатив.
2. Програма: складається з проектів.
3. Проект: складається із завдань.

Ефективне управління проектами вимагає планування, спілкування та управління завданнями. Але чи відомі всі основні області знань про управління проектами та як вони працюють разом?

Отже, щоб відповісти на це питання було розглянуто 10 областей знань з управління проектами, які надають важливу інформацію та набори навичок, які знадобляться для ефективнішого виконання проектів, задоволення зацікавлених сторін і боротьби з меншою кількістю «пожеж». У посібнику «Project management knowledge areas» добре розкрито та проаналізовано 10 галузей знань, а саме:

1. Управління інтеграцією проектів.

2. Управління обсягом проєкту.
3. Тайм-менеджмент проєкту.
4. Управління вартістю проєкту.
5. Управління якістю проєкту.
6. Управління ресурсами проєкту.
7. Управління проєктними комунікаціями.
8. Управління ризиками проєкту.
9. Управління за купівлями проєкту.
10. Управління зацікавленими сторонами проєкту.

Галузі знань з управління проєктами – це, по суті, те, що потрібно знати про ефективне управління проєктами. Області знань з управління проєктами, які можна знайти в Посібнику до Зводу знань з управління проєктами (PMBOK® Guide), є хорошим місцем для початку.

Нижче розглянуто кожную з 10 областей знань про управління проєктами на високому рівні разом із кількома групами процесів або елементами дій, пов'язаними з кожною з них.

1. *Управління інтеграцією проєктів* – це парасолька, яка охоплює всі інші області знань з управління проєктами. Вона об'єднує окремі процеси та завдання в один проєкт із визначеними цілями та результатами.

Якщо дивитесь на загальну картину та на те, як проєкт вписується у більшу організацію, це область знань з управління проєктами, яка потрібна. Оскільки це найширша область, її можна зберегти наостанок або принаймні повернутися до неї в кінці плану проєкту. Управління інтеграцією проєкту допомагає координувати всі різні частини проєкту, гарантуючи дотримання різними членами команди єдиного загального плану. Це допомагає забезпечити плавну та ефективну роботу проєкту.

2. *Управління обсягом проєкту*. Обсяг проєкту має бути чітко визначений і захищений протягом усього процесу.

Заповнивши групи процесів обсягу, створюється план керування, який визначає, перевіряє та контролює обсяг. Ці процеси гарантують, що кожен буде виконувати завдання та що всі, включаючи запитувача проєкту, розуміють, які завдання будуть включені в проєкт, щоб запобігти неприємним змінам і невиправданим очікуванням. Керування обсягом проєкту допомагає бути в курсі додаткових завдань, які можуть бути додані під час виконання проєкту. Таким чином, це робить роботу з будь-яким очікуваним або несподіваним збільшенням витрат або робочого процесу набагато легшим.

3. *Тайм-менеджмент проєкту.* Майже всі проєкти покладаються на кілька різних часових шкал і розкладів кількох людей.

Деякі члени команди можуть переоцінити, скільки часу знадобиться для завершення проєкту, щоб залишити подушку і не відчувати поспіху. Інші можуть недооцінювати свій час. І, звісно, несподівані проблеми також зіб'ють із графіка. Але саме ці змінні є причиною того, чому ефективне управління часом є таким критичним. Тому плани проєкту визначатимуть, які завдання можна скоригувати, а також як розподілятимуться ресурси команди та керувати ними протягом усього проєкту. Коли ці складні проблеми з'являються, чудово мати план, до якого можна повернутися та вгамувати паніку.

Тайм-менеджмент проєкту допомагає керувати проєктом у встановлені терміни та графіки. Завдяки цьому можна краще забезпечити плавний рух проєкту до своєчасного завершення.

4. *Управління вартістю проєкту.* З бюджетом чи без нього проєкт коштуватиме грошей. Збереження низьких витрат або принаймні на очікуваному або прийнятному рівні є фундаментальною частиною відображення рентабельності інвестицій у проєкт.

Ось чому роль проєкт-менеджера в управлінні витратами — це не просто одноразове завдання зі створення бюджету. Менеджеру потрібно

буде постійно оцінювати свої витрати, щоб уникнути будь-яких сюрпризів наприкінці проєкту.

Управління витратами на проєкт має вирішальне значення для того, щоб залишалися в рамках бюджету. Фінансові труднощі, які виникають під час проєкту, можуть викликати незаплановані сюрпризи та вплинути на прибутковість — управління витратами на проєкт допоможе уникнути цього.

5. *Управління якістю проєкту.* В управлінні проєктами якість не означає досконалість. Непрактично витрачати час і ресурси на доведення проєкту до досконалості — а в багатьох випадках це навіть недосяжно. Метою управління якістю проєкту є досягнення узгодженості між існуючими проєктами.

Отже, якщо знати та розуміти очікування зацікавлених сторін і укласти розумні угоди з ними та командою, контроль якості гарантуватиме, що менеджер завжди виконуватимете чудову роботу. Якщо проєкти не досягають результатів, менеджер можете скорегувати курс і внести зміни в процес або продукт, щоб повернутися до потрібного курсу. Завершити проєкт вчасно — це одне, але, мабуть, важливіше виконати його на високому рівні. Управління якістю проєкту дозволяє переконатися, що те, що менеджер надає, схвалюється клієнтом.

6. *Управління ресурсами проєкту.* Найголовніший ресурс проєкту – це люди. Однією з найприємніших частин цього процесу є створення команд, які дають змогу допомагати окремим членам команди розвиватися та вивчати нові завдання. Ось чому ця область знань з управління проєктами — це більше, ніж просто встановлення графіків і призначення завдань.

Ефективне управління ресурсами вимагає знання та роботи з пропускною здатністю обраної команди. Необхідно визначити їх індивідуальні сильні та слабкі сторони та взаємодію з іншими членами команди. Усе повертається до тієї частини про допомогу членам команди у

зростанні. Також необхідно визначити прогалини в знаннях і можливості для продовження навчання окремих членів команди та всієї команди на основі поточних і майбутніх проєктів. Налаштувати команду на успіх і збільшити відданість, інвестуючи в її навички та розвиток.

Тому проаналізувавши, можна упевнитись, що управління ресурсами проєкту – це перевага, оскільки відомо, хто і що потрібно на місці, щоб проєкт був успішним. Можете перевірити, чи є достатньо людей, потрібних людей і проєктна команда — і шукати додаткові ресурси, якщо є якісь прогалини.

7. *Управління проєктними комунікаціями.* Скільки разів всі чули фразу: «Тримай мене в курсі?» І все ж, коли відбуваються зміни, можливо, важливі зацікавлені сторони залишаються поза увагою?

Існує тонка грань між недостатнім і надмірним спілкуванням. План управління комунікаціями має вирішальне значення, щоб допомогти визначити, хто має знати, що і коли перед початком проєкту. Тому у будь-якому великомасштабному проєкті комунікація є ключовою — переконатися, що всі учасники своєчасно інформуються про зміни, оновлення та проблеми. Порушення комунікацій може бути проблематичним, тому це не та сфера, де хочеться бути слабкими.

8. *Управління ризиками проєкту.* Правда полягає в тому, що жоден проєкт не проходить без проблем. І нереально дивитися на проєкт і вважати, що все пройде гладко.

Якщо можна керувати пожежогасінням, визначаючи основні ризики проєкту та плани пом'якшення, пов'язані з ними, команда та автори запитів проєкту будуть підготовлені та більш поблажливі, коли виникають проблеми в проєкті. Як додатковий бонус, буде перевага часу та енергії наперед, а не намагання усунути несправності об одинадцятій годині, коли команда перебуває у стресі та не відповідає встановленому терміну.

Ось чому не часто буває так, що проєкт не натрапляє на один або два каменів спотикання. Управління ризиками проєкту може позначити

майбутні проблеми та надати засоби для їх вирішення та вирішення, замість того, щоб спричиняти серйозні ускладнення.

9. *Управління закупівлями проєкту.* У деяких випадках або частинах проєкту не буде ресурсів або членів команди для виконання завдання. Якщо наймати підрядників або постачальників для виконання певних завдань, тоді захочеться, щоб вони безперешкодно інтегрувалися в команду.

Ця область знань про управління проєктами надає план, для якого завдання чи послуги виконуватимуть сторонні підрядники. Він також завчасно створює та планує юридичну документацію та процес узгодження. Можливо, це не та сфера знань, яку використовують щоразу чи навіть дуже часто, але вона неймовірно цінна, коли вона потрібна.

Проаналізувавши, можливо, і не маючи необхідних знань і досвіду для кожного аспекту проєкту — управління за купівлями проєкту дозволяє швидко увійти в роботу, щоб будь-які підрядники почали працювати.

10. *Управління проєктом зацікавлених сторін.* Зрештою, успіх чи провал проєкту залежить від того, чи буде проєкт доступний зацікавленим сторонам. Але хто зацікавлені сторони?

Зацікавлені сторони включають не лише заявника проєкту, але й членів команди, які працювали над проєктом, підрядників, постачальників, клієнтів або громадськість, а також багатьох інших внутрішніх і зовнішніх людей організації. Не всі зацікавлені сторони рівні в очах проєкту.

Визначення того, хто є зацікавленою стороною в проєкті та як вони залучені до процесу, гарантує, що кожен отримає необхідну інформацію — ні більше, ні менше. Ось чому у ході проєкту є багато людей, які повинні бути задоволені, і будь-які зацікавлені сторони знаходяться прямо тут з точки зору найважливіших. Важливо переконатися, що вони мають всю необхідну інформацію, і це може позбавити проєкт від проблем надалі.

Розуміння областей знань про управління проєктами. Ці області знань з управління проєктами охоплюють багато питань. Дивитися на цей список

процесів і завдань може бути страшно, і навіть не розуміти, як вписати будь-яке з них у свій графік. Але застосування цих навичок у проєктах убереже від кризового управління або, принаймні, різко зменшить його, і підштовхне до перспективного мислення та ініціативного прийняття рішень. Удосконалюючи та повторюючи їх у проєктах, удосконалим управління проєктами та людьми, які в них беруть участь.

Крім 10 областей знань, згідно з посібником PMBOK® існує 49 процесів, які об'єднані в п'ять етапів управління проєктом:

- 1) *Фаза ініціації*. Основна мета фази ініціації полягає в тому, щоб переконатися, що проєкт відповідає бізнес-потребам і що зацікавлені сторони та команди проєкту узгоджуються за критеріями успіху проєкту протягом життєвого циклу проєкту.

Щоб досягти мети проєкту, найкраще залучати внутрішніх і зовнішніх зацікавлених сторін на етапі ініціації. Таким чином ви можете ефективно узгодити очікування та збільшити ймовірність виконання всіх результатів протягом життєвого циклу управління проєктом.

Під час фази ініціації вся команда проєкту визначає ідею проєкту, а спонсор проєкту оцінює її та дає дозвіл на продовження проєкту. Керівник проєкту починає процес документування, який включає обґрунтування, результати, ризики, оціночну вартість і вимоги до ресурсів.

Статут проєкту є ключовим результатом етапу ініціації проєкту та містить усю цю інформацію. Це перше формальне визначення проєкту. Він дає дозвіл на існування проєкту, встановлює повноваження керівника проєкту та документує вимоги високого рівня, етапи проєкту та критерії успіху.

Іншим важливим документом на етапі ініціації є Реєстр зацікавлених сторін. Цей документ містить інформацію про всіх зацікавлених сторін проєкту. Він визначає людей, групи та організації, які зацікавлені в завданні, проєкті та його результатах. Затвердження статуту проєкту означає перехід проєкту до наступної фази, фази планування проєкту.

2) *Фаза планування.* Після того, як очікування та критерії успіху зрозумілі, наступна фаза життєвого циклу управління проєктом зосереджується на плануванні кожного завдання, яке команда повинна виконати, щоб охопити обсяг, досягти кінцевих результатів і досягти загальної мети.

На етапі планування проєкту члени команди проєкту занурюються в конкретні вимоги, завдання, часові рамки та дії. Менеджер проєкту працює з усією командою, щоб створити дизайн, перерахувати список завдань і оцінити бюджет. Команда проєкту створює план ресурсів, план комунікацій і початковий графік проєкту. Керівник проєкту також визначає ролі та обов'язки команди та зацікавлених сторін. Обсяг проєкту остаточно визначається залежно від затверджених доступних ресурсів і пріоритетів клієнта.

На етапі планування команда проєкту завершує структуру розподілу робіт, план проєкту, перелік вимог, план управління комунікаціями та інші відповідні документи для налагодження робочого процесу та координації із залученими сторонами.

План проєкту є ключовим результатом і містить детальну структуру розподілу робіт (WBS) або список завдань із датами початку та завершення, а також передбачуваними зусиллями та тривалістю. Він визначає етапи, ресурси та графік. Він також містить залежності завдань, які дозволять групі проєкту використовувати метод критичного шляху, якщо вона вибере.

Іншими важливими результатами є План управління комунікаціями, який допомагає сприяти ефективній комунікації із зацікавленими сторонами, і План розподілу ресурсів, який визначає графік ресурсів команди проєкту щодо їх доступності протягом усього життєвого циклу проєкту.

Про що слід пам'ятати менеджерам проєктів: коли дізнавшись більше інформації, можливо, доведеться скорегувати попередній план проєкту та

відповідні процедури. Більш складні проєкти вимагатимуть більше повторних схвалень для кожного створеного завдання.

Планування проєкту — це повторюваний процес, тому керівник проєкту повинен переглядати, переглядати та переглядати всі плани принаймні раз на місяць до завершення проєкту. Для команди проєкту також важливо залучати відповідних зацікавлених сторін на цьому етапі життєвого циклу проєкту.

3) *Фаза виконання.* Етап виконання проєкту — це етап, на якому команда проєкту виконує завдання на основі плану проєкту. На цьому етапі команда витрачає більшу частину свого часу на координацію з людьми, допомагаючи забезпечити якість роботи, відстежуючи ресурси та інформуючи зацікавлених сторін.

Цей етап, який іноді називають етапом впровадження, є етапом, коли керівник проєкту намагається керувати кожним завданням і аспектом реалізації проєкту, щоб утримувати проєкт у потрібному плані протягом усього життєвого циклу проєкту, що залишився.

Команда проєкту зосереджується на досягненні всіх цілей, поставлених на попередніх етапах. На цьому етапі керівник проєкту, швидше за все, використовує програмне забезпечення для управління проєктом, щоб призначити кожне завдання членам команди. Інструменти, які централізують інформацію про завдання, а також доступність ресурсів і командне спілкування можуть спростити й оптимізувати необхідні процеси управління проєктами.

Документація із забезпечення якості, протоколи зустрічей і робочі замовлення — це деякі документи, створені на етапі виконання життєвого циклу управління проєктом.

Цілком імовірно також, що ви відкриєте нову інформацію, яка вимагатиме перегляду та оновлення початкових планів управління проєктом. Необхідно має увагу щодо запитів на зміни та переконатися, що необхідні коригування виконуються.

- 4) *Етап моніторингу та контролю.* Найкращий спосіб забезпечити прогрес і вдосконалення — відстежувати та переглядати ефективність проєкту.

Одночасно під час виконання проєкту команда ретельно відстежує прогрес проєкту на основі Плану проєкту, встановленого раніше. Відстеження ефективності проєкту за допомогою різних показників має вирішальне значення для того, щоб забезпечити виконання проєкту за графіком, у межах бюджету та обсягу.

Команда проєкту відстежує документи управління змінами, записи про витрати, контрольні списки контролю якості та відлік робочого часу команди. Вони можуть виміряти, куди спрямовуються зусилля та ресурси протягом життєвого циклу проєкту, порівнюючи це з планом проєкту.

І фаза виконання, і фаза моніторингу та контролю є критичними періодами, які можуть визначити успіх проєкту. Окрім моніторингу виконання завдань, керівник проєкту також намагається виявити проблеми чи ризики, разом із командою створює план пом'якшення наслідків і регулярно звітує про статус проєкту зацікавленим сторонам. Старанність у записі та вимірюванні прогресу проєкту ставить команду проєкту в стратегічне положення. Вони можуть виявити вузькі місця та ініціювати важливі обговорення або вдосконалення процесу управління проєктом.

Проактивний підхід дозволить команді проєкту швидко реагувати на будь-які зміни в плані. Якщо потрібне додаткове планування, час або ресурси, тоді потрібно буде повідомити про це відповідним зацікавленим сторонам проєкту, поки не стане надто пізно. Також мати дані та результати для резервного копіювання запитів, тож буде більше шансів виправдати запити та зберегти їхню довіру, незважаючи на обставини.

- 5) *Фаза закриття.* На останній фазі життєвого циклу управління проєктом завершуються всі дії, пов'язані з його завершенням. Це може включати подання остаточного результату, виконання

договірних зобов'язань, розірвання відповідних угод і вивільнення ресурсів проєкту.

Причинами закриття проєкту можуть бути завершення, скасування, припинення або передача новій організації. Документація, необхідна для завершення закриття проєкту, буде відрізнятися залежно від ситуації.

На цьому етапі менеджер проєкту повідомляє всім зацікавленим сторонам про остаточний порядок і статус проєкту. На цьому етапі також забезпечується інформування учасників і зацікавлених сторін про будь-які подальші дії або продовження життєвого циклу продукту, щоб вони могли спілкуватися та координувати дії з відповідальними людьми.

Незалежно від результату життєвого циклу проєкту, було б добре, щоб команда провела ретроспективу проєкту. Під час цієї посмертної діяльності команда проєкту може обробити нові уроки та забезпечити вдосконалення поточних процесів управління проєктом для майбутнього проєкту.

Під час завершення проєкту документи, які потрібно передати, можуть включати різноманітну проєктну документацію, протоколи заключних нарад та інші звіти про закриття. Ці документи можуть ідентифікувати та фіксувати отримані уроки та найкращі практики для подальшого використання та повторного використання.

Доцільно організувати та зберігати матеріали проєкту в спільній груповій папці. Ці матеріали можуть бути довідковими під час оцінювання ефективності. Можливість постійно тестувати, вдосконалювати або винаходити способи управління всім життєвим циклом проєкту може сприяти розвитку організації та її бізнесу.

Посібник PMBOK® містить перелік 49 процесів у 10 областях знань, з різними процесами, що підпадають під кожен область. Звернувшись до посібника, щоб дізнатися про всі 49, але прикладом може бути управління вартістю проєкту, яке складається з чотирьох процесів:

1 процес - Оцінка вартості – це процес складання та прогнозування витрат проєкту протягом його життєвого циклу. Він охоплює три фази,

включені в початкову функціональну схему; а саме економічна оцінка, інвестиційна вартість проєкту та прогнозування витрат.

2 процес - Заявки на вартість. Це рівень процесу, пов'язаний зі спеціальними застосуваннями методів витрат, які не включені в інші процеси витрат. Він також включає пов'язані теми, які впливають на управління витратами, такі як комп'ютерні програми, аналіз вартості тощо.

3 процес - Визначення бюджету – це процес створення бюджетів, стандартів і системи моніторингу, за допомогою яких можна виміряти інвестиційну вартість проєкту та керувати нею. Цей процес є етапом планування після отримання схвалення проєкту. Він включає всі облікові функції, необхідні для встановлення процедур і систем моніторингу проєкту.

4 процес - Контрольні витрати. Хоча функції контролю з'являються в усіх діаграмах функцій управління проєктами, дії, пов'язані з кожною з них, не є універсальними, тому термін «Контроль витрат» використовується для відмінності від інших функцій. Процес контролю витрат - це збір, накопичення, аналіз, моніторинг, звітування та управління витратами на постійній основі.

Оскільки технології продовжують розвиватися, а методи вдосконалюються та вдосконалюються, не дивно, що процеси управління проєктами та передовий досвід також змінюватимуться. Посібник PMBOK® часто оновлюється, щоб відобразити це, тому сучасні маркетологи та менеджери проєктів повинні стежити за останніми інноваціями.

1.2. Загальна характеристика та класифікація ІТ- проєктів

ІТ-проєкти – це проєкти, пов'язані з розробкою, впровадженням, підтримкою або оптимізацією інформаційних технологій. Вони можуть бути різними за масштабом, складністю та тривалістю. ІТ-проєкти мають ряд загальних характеристик, які відрізняють їх від інших типів проєктів.[2]

Загальні характеристики ІТ-проектів:

1. *Інформаційно-технологічний характер.* ІТ-проекти передбачають розробку та впровадження інформаційних систем, які використовують комп'ютерні технології.

Інформаційні системи – це сукупність взаємопов'язаних компонентів, які обробляють інформацію та забезпечують її доступ користувачам. Комп'ютерні технології – це сукупність методів і засобів, які використовуються для розробки, впровадження та використання комп'ютерів та комп'ютерних систем.

ІТ-проекти можуть бути різними за масштабом, складністю та тривалістю. Вони можуть включати розробку нового програмного забезпечення, впровадження нової інформаційної системи, модернізацію існуючої інформаційної системи або впровадження нових технологій, таких як хмарні або мобільні технології.

ІТ-проекти є важливими для успіху бізнесу або організації. Вони можуть допомогти підвищити ефективність, продуктивність та конкурентоспроможність.

Ось деякі приклади інформаційно-технологічних компонентів, які можуть бути включені до ІТ-проектів:

- програмне забезпечення: операційні системи, бази даних;
- програмне забезпечення для бізнес-аналізу;
- програмне забезпечення для електронної комерції;
- обладнання: комп'ютери, сервери, периферійні пристрої;
- мережі: локальні мережі, глобальні мережі;
- сховища даних: дані, які зберігаються в інформаційній системі.

ІТ-проекти вимагають спеціальних знань та навичок у галузі інформаційних технологій. Персонал, який працює над ІТ-проектами, повинен мати глибокі знання про інформаційні системи, комп'ютерні технології та бізнес-процеси.

2. *Комплексність*. ІТ-проєкти часто є комплексними завданнями. Вони передбачають взаємодію різних елементів, таких як обладнання, програмне забезпечення, люди та процеси. Ця взаємодія може бути складною, що може призвести до труднощів у управлінні ІТ-проєктами. Приклади елементів, які можуть бути включені до ІТ-проєктів:

- обладнання: комп'ютери, сервери, периферійні пристрої;
- невизначеність: ІТ-проєкти часто характеризуються невизначеністю, оскільки можуть виникати непередбачені труднощі та проблеми;
- важливість: ІТ-проєкти часто є важливими для успіху бізнесу або організації;
- програмне забезпечення: операційні системи, бази даних;
- програмне забезпечення для бізнес-аналізу;
- програмне забезпечення для електронної комерції;
- люди: розробники, тестувальники, менеджери проєктів, користувачі;
- процеси: бізнес-процеси, процеси розробки, процеси тестування.

Управління взаємодією цих елементів є важливою частиною успішного управління ІТ-проєктами.

Деякі з проблем, які можуть виникнути внаслідок складності ІТ-проєктів, включають:

- 1) Невизначеність: складність ІТ-проєктів може призвести до невизначеності щодо їх масштабу, складності та тривалості. Це може ускладнити планування та управління проєктами.
- 2) Зміни: вимоги до ІТ-проєктів можуть змінюватися протягом усього проєкту. Це може призвести до затримок, перевитрат та інших проблем.
- 3) Невідповідність: ІТ-проєкти можуть не відповідати вимогам користувачів або бізнесу. Це може призвести до незадоволеності користувачів та невдачі проєкту.

ІТ-проекти можуть бути успішними, незважаючи на їх складність. Однак менеджери проєктів повинні бути обережними, щоб правильно керувати складністю та уникати проблем, які можуть виникнути.

Приклади того, як ІТ-проекти можуть допомогти бізнесу або організації досягти успіху:

- розробка нового веб-сайту або додатків: новий веб-сайт або додатки можуть допомогти бізнесу або організації підвищити свою присутність в Інтернеті та залучити нових клієнтів;
- впровадження нової інформаційної системи: нова інформаційна система може допомогти бізнесу або організації автоматизувати завдання, покращити ефективність і продуктивності та зменшити витрати;
- модернізація існуючої інформаційної системи: модернізація існуючої інформаційної системи може допомогти бізнесу або організації підтримувати конкурентоспроможність і відповідати новим вимогам;
- впровадження хмарних технологій: хмарні технології можуть допомогти бізнесу або організації скоротити витрати на ІТ-інфраструктуру і забезпечити більш гнучке та масштабоване рішення;
- впровадження мобільних технологій: мобільні технології можуть допомогти бізнесу або організації взаємодіяти з клієнтами в будь-який час і в будь-якому місці.

ІТ-проекти можуть бути складними та вимагати значних інвестицій. Однак вони також можуть бути дуже корисними для бізнесу або організації. Важливо правильно планувати та управляти ІТ-проектами, щоб забезпечити їх успішне завершення.

Класифікація ІТ-проектів. ІТ-проекти можна класифікувати за різними критеріями. Ось деякі з найпоширеніших класифікацій:

За масштабом: визначається бюджетом, тривалістю та кількістю залучених людей.

За складністю: визначається кількістю елементів, які необхідно розробити та впровадити.

За тривалістю: визначається часом, необхідним для їх завершення.

За типом: визначається їх метою.

Управління ІТ-проєктами є важливим завданням, оскільки воно допомагає гарантувати, що ІТ-проєкти будуть успішно завершені вчасно, в рамках бюджету та відповідно до вимог. Управління ІТ-проєктами включає в себе такі завдання, як:

1. Планування: розробка плану проєкту, який визначає цілі, завдання, ресурси та терміни виконання проєкту.
2. Виконання: виконання завдань проєкту відповідно до плану.
3. Контроль: моніторинг прогресу проєкту та виявлення та вирішення проблем.
4. Завершення: завершення проєкту та передача його в експлуатацію.

Управління ІТ-проєктами є складним завданням, яке вимагає спеціальних знань та навичок. Однак ефективне управління ІТ-проєктами може забезпечити успіх бізнесу або організації.

1.3. Основні поняття методології в управлінні проєктами

Якщо розуміти детально методології управління проєктами, може бути важко відрізнити дошку Kanban від дошки Scrum або зрозуміти, чому важливі відмінності між методологіями Agile та Waterfall. І з такою кількістю подібних термінів може здатися, що не має значення, який метод управління проєктом потрібно обрати.

Не варто пропустити явних переваг, які можуть запропонувати ці методології. Нижче наведено блок-схеми, щоб отримати короткий огляд методології управління проєктами та краще розуміти та порівнювати Agile проти Waterfall проти Kanban проти Scrum.

Хоча ці методології мають значні відмінності, важливо визнати, що кожна методологія управління проєктами в кінцевому підсумку має ту саму мету: сприяти завершенню проєктів. З цією метою кожна методологія допомагає керувати робочими процесами команди через структуру та комунікацію. Хоча реалізація кожної з цих методологій різна, Agile, Waterfall, Kanban і Scrum усі мають багато спільного. Хоча відмінності між методологіями можуть здатися незначними, але все рівно вони існують. Насправді ці, здавалося б, дрібні деталі значною мірою впливають на функціонування методу. Маючи це на увазі, розглянуто, що відрізняє кожну методологію управління проєктами.

Управління проєктом Waterfall, проєкти розбиваються на лінійні та послідовні етапи, де кожна частина проєкту залежить від завершення попередніх результатів. Таким чином, Waterfall має дві унікальні риси: дискретні, кінцеві фази. Нижче розглянуто більш конструктивне значення даної методології.

Waterfall (рис.1.1). Управління водоспадними проєктами зародилося в будівництві та виробництві — галузях, де один етап має бути завершений перед початком іншого. Не можна розпочати покрівлю, наприклад, якщо не закінчили каркас. Цей акцент на лінійному завершенні є центральним у робочому процесі Waterfall [3].

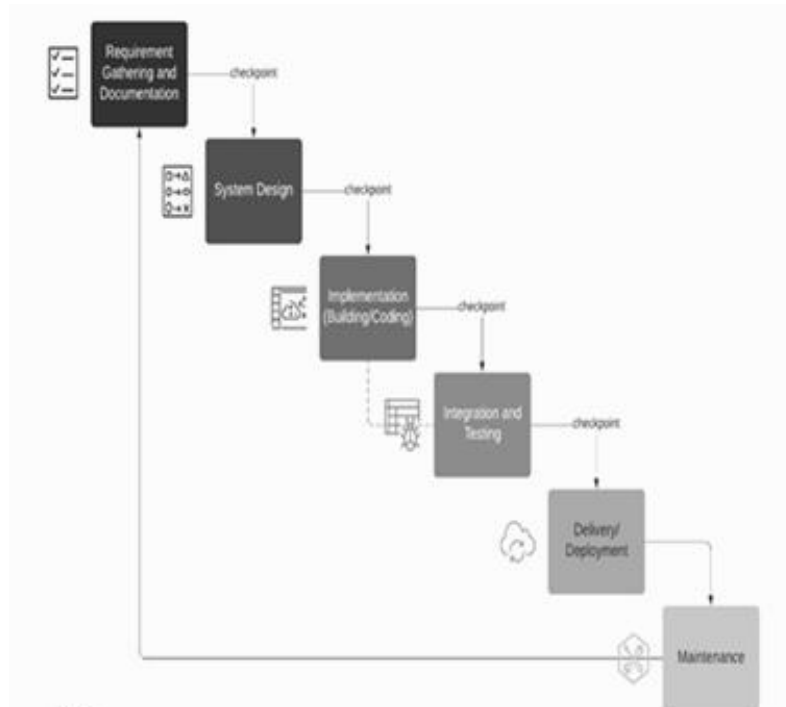


Рис.1.1. Waterfall

Waterfall використовує різні фази, а не одночасну роботу. Якщо використовувати методологію Waterfall, то необхідно завершити кожен етап перед початком наступного. Так само не можна повернутися до попередньої фази. Будь-який перегляд вимагає перезапуску всього процесу.

Якщо розглядати Agile проти Waterfall в управлінні проєктами, необхідно пам'ятати, що Waterfall пропонує меншу гнучкість. Оскільки водоспад не дозволяє повернутися до попередньої фази, вимоги до проєкту мають бути чіткими наперед. Ця методологія починається зі збору та документування вимог, а потім робить ці вимоги доступними для членів команди. Члени команди також документують свою роботу, оскільки проєкт продовжується на кожному етапі. В ідеалі члени команди можуть виходити з проєкту або входити в нього, не порушуючи робочий процес, що робить Waterfall хорошим рішенням для команд, які очікують змін у пропускній здатності.

Agile (рис.1.2). Гнучка методологія, яка зазвичай використовується в налаштуваннях розробки програмного забезпечення, являє собою спільний, самоорганізований, міжфункціональний підхід до виконання роботи та

виконання вимог. Хоча можна почути, що про це говорять як про окрему методологію, гнучке управління проєктами правильніше відноситься до категорії методологій, яка включає Scrum і Kanban. І все-таки існують принципові відмінності між управлінням проєктами Agile та Waterfall.

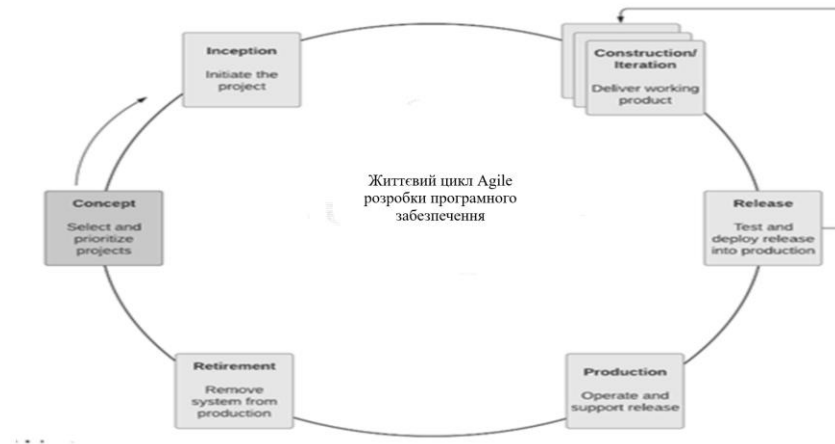


Рис.1.2 Agile

Agile зосереджується на адаптивних одночасних робочих процесах — це дуже далеко від лінійної природи Waterfall. Кожна методологія Agile матиме такі характеристики:

- 1) *Одночасна, поступова робота.* Це найбільш відмінна риса при порівнянні методології Waterfall і Agile. Гнучкі методи розбивають проєкти на менші ітераційні періоди, що особливо добре працює для продуктів, які виграють від тестування користувачами та регулярного перегляду (наприклад, програмного забезпечення). Оскільки гнучкі методи працюють поступово, команди можуть коригувати свої процеси з певною частотою. Там, де Waterfall використовує встановлений, негнучкий процес, методології Agile заохочують команди вдосконалювати та коригувати робочий процес за потреби.
- 2) *Адаптивність Agile* особливо добре підходить для проєктів, у яких очікується зміна вимог або обмежень. Хоча необхідно уникати таких змін, коли це можливо, методології Agile дозволяють командам адаптувати свій процес, щоб компенсувати такі зміни.

Канбан (рис.1.3) Хоча багато людей хочуть порівняти Kanban з Agile, Методика Канбан точніше, це конкретний тип методології Agile. Kanban прагне краще координувати та збалансувати роботу з можливостями та пропускнуою здатністю працівників. Він використовує принципи методології Agile, розглянуті вище, але реалізує їх певним чином.

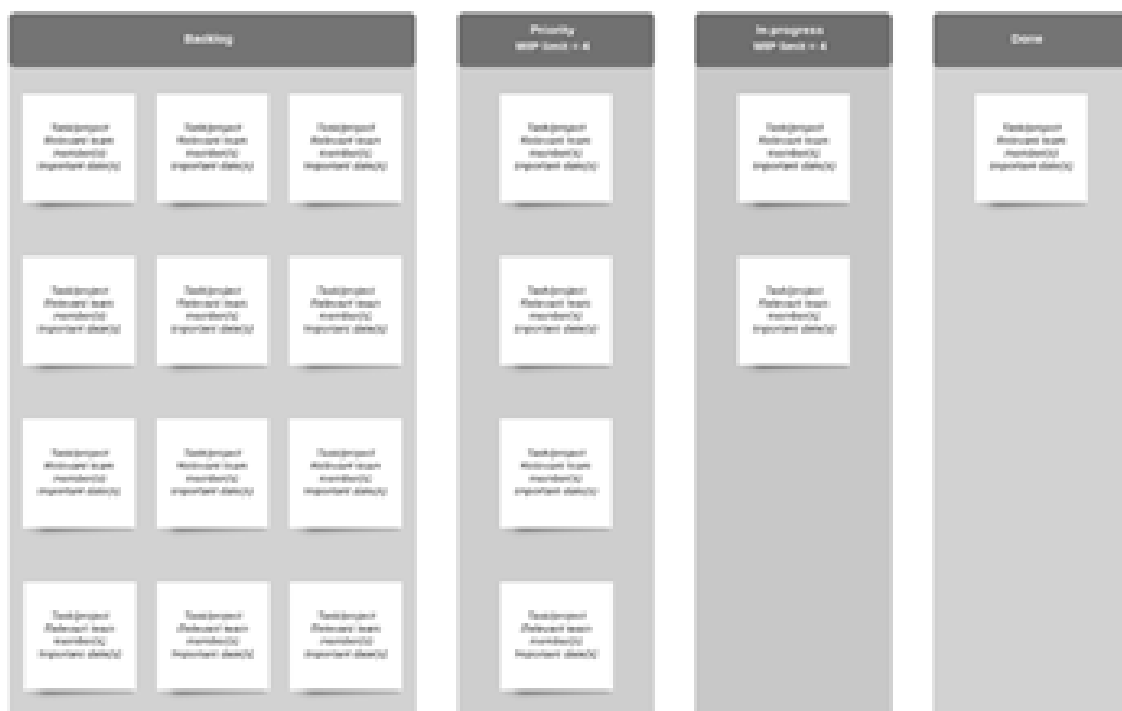


Рис. 1.3. Дошка Канбан

Однойменна дошка Kanban візуалізує робочий процес команди. Дошку поділено на категорії роботи, яку потрібно виконати, незавершеної роботи та завершеної роботи, і команди можуть додавати більше категорій за потреби, щоб краще візуалізувати свій процес.

Методологія Kanban вимагає суворих обмежень щодо обсягу незавершеної роботи в будь-який момент часу. Команди встановлюють обмеження на кількість карток у будь-яких колонках активної роботи. Коли ліміт досягнуто, нові роботи не можуть увійти до стовпця, доки завдання не буде виконано та переміщено до наступного стовпця. Знову ж таки, ця система допомагає командам виявляти вузькі місця та заохочує окремих учасників об'єднуватися для їх усунення.

Метою методології Kanban є покращення командного процесу. Команда періодично збирається для обговорення змін, які необхідно внести, і дані, які відображаються на дошці Kanban, керують цими обговореннями. Якщо такі зустрічі проводяться регулярно, вони допомагають команді постійно виправляти та коригувати свій процес. Це покращує робочий процес без будь-яких раптових або драматичних змін, забезпечуючи легке впровадження Kanban майже в будь-якій команді.

Scrum. Остаточна методологія, яку розглянуто, управління проєктами Scrum (рис.1.4), це ще одна методологія Agile, яка використовує поступовий підхід до роботи, щоб швидше завершувати проєкти. Методологія Scrum зазвичай займається комплексною роботою зі знаннями, такою як розробка програмного забезпечення.

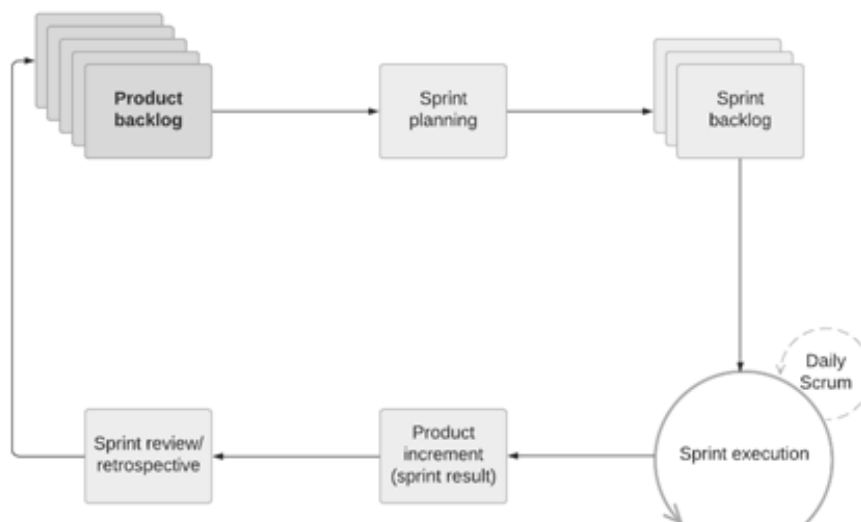


Рис. 1.4. Scrum

Якщо дивитися на Kanban проти Scrum, Kanban насамперед стурбований вдосконаленням процесів, тоді як Scrum стурбований тим, щоб більше роботи було виконано швидше.

Scrum використовує двотижневі спринти, щоб виконати роботу. Ці спринти плануються заздалегідь, виконуються, а потім переглядаються в кінці двотижневого періоду. Під час планування спринту команда створює

резерв спринту. Команда виконує ці невиконані завдання під час спринту, керуючи роботою між собою.

Члени команди також проводять 15-хвилинне засідання Scrum кожного дня спринту. Протягом цього часу учасники обговорюють будь-які потенційні перешкоди, які заважають успіху проєкту. Вони також переглядають роботу за попередній день і планують завдання на наступний день. Ця зустріч Scrum гарантує, що команда працює разом і підтримує синхронізацію.

Scrum використовує діаграму згорання під час спринтів, щоб дозволити членам команди миттєво бачити прогрес. Замість того, щоб відображати виконані завдання, діаграма згорання візуалізує те, що ще залишилося зробити. Його слід постійно оновлювати, щоб допомогти членам команди керувати робочим процесом.

Хоча кожна методологія має однакову мету - завершення проєкту, їхні другорядні цілі роблять їх справді різними.

Необхідно визначити, чого найбільше потрібно для команди. Якщо потрібно просто виконувати роботу швидше, краще обрати Scrum. Якщо потрібно покращити виробничий процес, використовуйте Kanban. Якщо проєкти потребують лінійного робочого процесу, запровадьте Waterfall. Якщо не має впевненості, можна дослідити інші варіанти Agile .

Відмінності в методологіях управління проєктами мають значення, лише якщо ви використовуєте методологію послідовно. Без обмежень WIP, наприклад, Kanban — це просто ще одна складна методологія Agile. І якщо ви не зберігаєте свої фази дискретними під час використання Waterfall, тоді ви можете просто використовувати методологію Agile.

Таким чином, найкраща методологія управління проєктами для команди – це та, яку можна бездоганно виконувати. Використання окремих частин методології лише призведе до того, що можуть втрачатись переваги, які популяризували методологію. Отже, хоча, звичайно, можна адаптувати

методології для використання командою, найкраще використовувати методологію за призначенням, коригуючи лише за необхідності.

Висновок до розділу 1. У розділі було розглянуто сукупність знань, навичок, інструментів, методів і технік, застосовуваних під час управління ІТ-проєктами. А також узагальнення основних висновків, які були зроблені під час аналізу та опису характеристик і класифікації ІТ-проєктів. Включає в себе висновки щодо типів ІТ-проєктів, їхніх специфічних рис, можливих переваг та обмежень. В розділі розглянуто, які концепції та методи вважаються найбільш ефективними або рекомендованими для успішного управління проєктами, можливі обмеження або переваги різних підходів, а також загальні висновки про важливість розуміння цих понять для успішного виконання проєктів.

РОЗДІЛ 2. АНАЛІЗ МЕТОДОЛОГІЇ УПРАВЛІННЯ ІТ-ПРОЄКТІВ

2.1. Аналіз методів управління ІТ-проектів сімейства Agile

Не маючи єдиної методології, яка б представляла відповідну альтернативу Waterfall, який був надто повільним і ризикованим, 11 лютого 2001 було створено маніфест Agile «Розробка програмного забезпечення».

Agile — це підхід до управління проектами, розроблений як більш гнучкий і ефективний спосіб виведення продуктів на ринок. Слово «спритність» означає здатність рухатися швидко та легко. Таким чином, гнучкий підхід дозволяє проектним командам адаптуватися швидше та легше порівняно з іншими проектними методологіями.

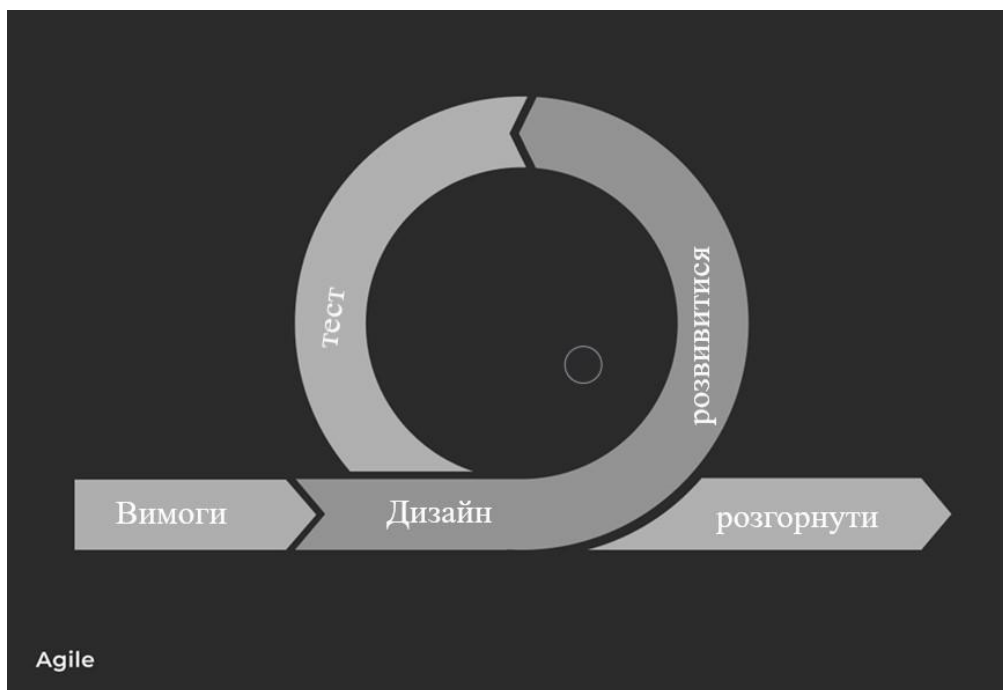


Рис.2.1. Життєвий цикл проекту Agile.

Гнучка методологія — це підхід до управління проектами, який використовує чотири цінності та 12 принципів для організації проектів.

Чотири цінності Agile Manifesto [4]:

1. Особи та взаємодія над процесами та інструментами.
2. Робота програмного забезпечення над повною документацією.
3. Співпраця клієнтів щодо переговорів про контракт.

4. Реагування на зміни за планом.

Гнучкий метод працює в постійних спринтах планування та виконання проєкту, дозволяючи вам постійно адаптувати та вдосконалювати свій план, обсяг і дизайн протягом усього проєкту.

Гнучкі проєкти вимагають ітераційного підходу, який підтримує поетапне, часте та послідовне надання працездатних продуктів замовнику чи клієнту. Цей інноваційний підхід гарантує, що проєктна команда може стабільно постачати конкретні продукти без затримок через зміни та зміну вимог.

Agile є основною методологією, яка використовується в сучасній розробці програмного забезпечення, і поширює свій вплив за межі кодування на багато аспектів розробки продукту, від ідеї до взаємодії з клієнтами.

Методологія Agile розбиває проєкт на кілька циклів, кожен з яких проходить через деякі або всі фази SDLC (Software Development Lifecycle). Основна увага приділяється людям і тому, як вони працюють разом, щоб виконати проєкт. Agile вимагає постійної співпраці між членами команди та зацікавленими сторонами з регулярними циклами зворотного зв'язку та ітерацій.

Гнучкі ролі. Ролі Agile розподіляють обов'язки між членами команди. Вони відрізняються від посад, оскільки одна особа може виконувати кілька ролей Agile залежно від обсягу проєкту. І навпаки, кілька людей можуть поділяти одну роль. Ось деякі з ролей, які можна побачити в проєкті Agile:

- власник продукту, також відомий як «голос клієнта», визначає бачення продукту на основі всіх ідей, відгуків та зібраних ідей. Він або вона є власником вимог до продукту та тісно співпрацює з командою розробників, щоб передати бачення, задокументувавши його в коротких розповідях під назвою «Історії користувачів». Історії користувачів зазвичай містять назву, опис, посилання на будь-які зовнішні документи та пояснення того, як перевірити реалізацію.

Власники продукту часто зберігають резерв історій користувачів, якщо їх забагато для одночасного виконання;

- Scrum Master - подібно до менеджера проєкту, ця роль полягає в тому, щоб переконатися, що команда дотримується принципів, цінностей і процесів Agile;
- член команди – усі члени команди розробників мають різні навички та співпрацюють разом для створення функціонального програмного забезпечення. Команди можуть включати інженерів із забезпечення якості, бізнес-аналітиків, дизайнерів, інженерів баз даних тощо залежно від обсягу проєкту.

Переваги гнучкої методології:

- пропонуйте програмне забезпечення, добре адаптоване до постійно зростаючого розуміння вимог клієнтів;
- програмне забезпечення розгортається швидше та регулярно вдосконалюється;
- краща гігієна коду, включаючи стиль, читабельність і структурування;
- гнучкий і адаптивний процес дає змогу робити повороти або зміни в середині проєкту;
- не вимагає повного списку вимог заздалегідь;
- звільняє місце для впливу на організаційне навчання під час просування проєкту;
- прозорість і постійна комунікація із залученими зацікавленими сторонами.

Agile Frameworks. Організації можуть прийняти єдину структуру Agile або поєднати елементи кількох структур відповідно до потреб проєкту та характеристик команди. Найпопулярніші фреймворки Agile:

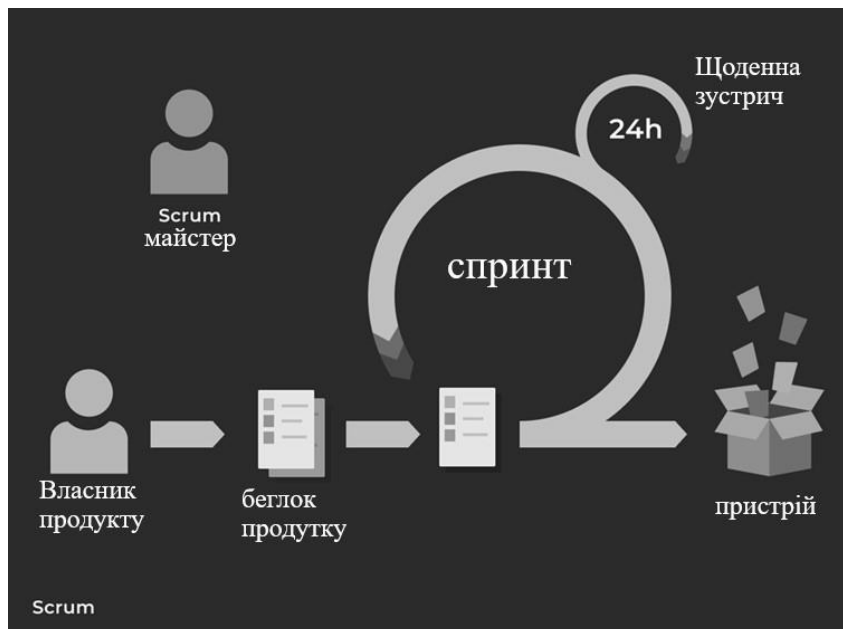


Рис.2.1. Scrum

Scrum — це легкий фреймворк, який допомагає людям, командам і організаціям створювати цінності за допомогою адаптивних рішень для складних проблем.[7] Він має дуже популярну гнучку структуру, яка характеризується безперервною співпрацею, частими поставками та спеціальними циклами розробки, які називаються «Спринтами». Scrum обертається навколо таких контрольних точок:

- 1) Зустрічі з планування, на яких команда визначає та обговорює пріоритети спринту.
- 2) Наради щодо зобов'язань, під час яких команда переглядає резервні історії користувачів, щоб визначити, скільки зусиль це потребує та скільки роботи можна виконати під час майбутнього спринту.
- 3) Щоденні стендап-зустрічі, які є особливо короткими, щоб переконатися, що всі узгоджені. У зв'язку з цим кожен член команди повідомляє про оновлення щодо статусу історії, блокувальників або проблем.
- 4) Демонстраційні зустрічі, які команда відвідує в кінці кожного спринту, щоб показати власнику продукту функції, реалізовані під час поточного спринту.

- 5) Ретроспективні зустрічі, які також проводяться наприкінці кожного спринту, щоб обговорити отримані уроки, те, що пройшло добре, а що потребує вдосконалення.

Scrum представляє роль Scrum Master у методі Agile. Робота Scrum Master полягає в тому, щоб керувати процесами та покращувати їх, допомагати команді залишатися автентичними цінностями Agile та зосереджуватися на максимальному підвищенні продуктивності. Хороший Scrum Master гарантує, що процес і прогрес є прозорими для всіх зацікавлених сторін.

Kanban — це популярний метод економічного керування робочим процесом для визначення, управління та вдосконалення послуг, які забезпечують роботу знань[8]. Він не має свого коріння в розробці програмного забезпечення, але дуже добре поєднується з Agile і став основним продуктом команд Agile.

Kanban розпочався з економічного виробництва, де Toyota застосовувала ті самі принципи «точно вчасно», які використовують супермаркети для управління рівнем запасів на основі попиту клієнтів. Kanban, що означає «вивіска» японською мовою, використовує картки для відстеження та підтримки виробничої системи, візуально показуючи етапи процесу та тривалість кожного кроку за допомогою карток. Kanban має безліч переваг у застосуванні до Agile. Можна обмежити WIP, зосередитися на тривалості циклу та використовувати методи «точно вчасно».

Канбан іноді порівнюють із Scrum, які в деякому роді схожі, але є різними фреймворками:

- у Scrum використовуються цикли спринтів фіксованої довжини, а в Kanban — безперервний потік;
- Scrum орієнтований на роль, тоді як Kanban не використовує ролі;
- Scrum вимірює швидкість, тоді як Kanban зосереджується на тривалості циклу

У рамках Kanban команда створює візуальне представлення своїх завдань і статусів за допомогою наліпок на фізичній дошці або за допомогою спеціального програмного забезпечення. Завдання переміщуються через попередньо визначені етапи, такі як «Справа», «Виконується», «Переглядається» або «Завершено».

Кілька прикладів популярних програм Kanban для продуктивності:

1. Trello
2. Джира
3. Proofhub
4. Проєкти Zoho
5. ZenHubs
6. Екстремальне програмування.

Екстремальне програмування (XP) — це гнучка структура, зосереджена на гнучкості проєкту та написанні високоякісного, добре перевіреного коду.

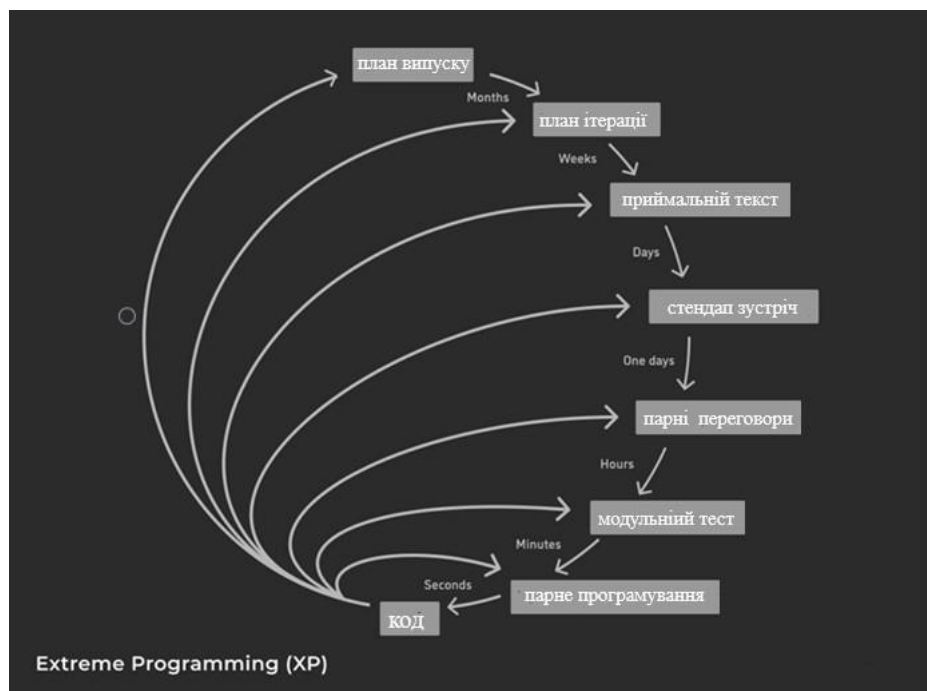


Рис.2.2. Екстремальне програмування

На офіційному веб-сайті Extreme Programming зазначено, що XP покращує проєкт програмного забезпечення 5 ключовими способами [7]:

- спілкування;
- простота;
- зворотній зв'язок;
- повага;
- мужність.

Екстремальне програмування найбільш відоме наступним:

Парне програмування — це техніка, коли два програмісти використовують одну робочу станцію та разом створюють програмне забезпечення. Один виконує роль водія, а інший — навігатора, потім вони міняються ролями. У парі перевірка коду може відбуватися миттєво, а дефекти, швидше за все, будуть негайно виявлені та виправлені. Парне програмування заохочує наставництво, обмін знаннями та навчання. І хоча створення нового коду може зайняти більше часу, коли два розробники працюють над одним завданням, отриманий код є вищої якості з меншою кількістю дефектів. У ХР наголошується на модульному та функціональному тестуванні. Тести мають бути всебічними й автоматизованими, зменшуючи технічний борг і гарантуючи, що код можна впевнено перевірити та повторно використовувати. Постійне спілкування між програмістами та зацікавленими сторонами для збору та реагування на їхні внески, відгуки та запити на зміни. ХР вимагає «розширеної групи розробників», яка може включати бізнес-менеджерів, клієнтів та інших ключових зацікавлених сторін.

Сучасні умови господарювання, що постійно супроводжуються викликами різного характеру, в тому числі, форс-мажорними, вимагають від власників та менеджменту підприємств обирати моделі управління, що дозволять мінімізувати вплив кризових процесів на діяльність підприємства.

LEAN-менеджмент ґрунтується на основних принципах, що були визначені J.K. Liker на основі багаторічного досвіду Toyota, і отримали назву «The Toyota Way Philosophy» [8].

Lean – це не методологія розробки програмного забезпечення. Витоки Lean сягають виробничого методу виробництва, який був винайдений у 1930-х роках, офіційно названий у 80-х роках і більш формально визначений у 90-х роках. Lean – система, яка спрямована на отримання більшого з меншими витратами. Багато хто нещодавно виявив, що Lean надзвичайно добре працює з розробкою програмного забезпечення, особливо Agile.

У той час як Agile зосереджується на забезпеченні безперервної цінності, метою Lean є збільшення швидкості та зниження вартості розробки продукту. З Lean найвищий ризик – це втрачений час і зусилля. Lean перешкоджає багатозадачності та спонукає членів команди зосереджуватися на тому, що є важливим у поточний момент. Таким чином, відходи, пов'язані з непотрібною документацією, зустрічами чи плануванням, усуваються.



Рис.2.3. Lean Workplace

Lean зосереджується на наступних принципах «точно вчасно»:

- усунення марних витрат у вартості, масштабах і плануванні;
- посилення навчання;
- прийняття рішень якомога пізніше;
- швидка доставка;

- розширення можливостей команди;
- побудова цілісності;
- оптимізація всього проєкту.

DevOps – це підхід до розробки програмного забезпечення, який дозволяє командам створювати, тестувати та випускати ПЗ швидше та надійніше завдяки впровадженню принципів та методів Agile, таких як розширення автоматизації та взаємодії між командами з розробки та експлуатації. Розробка, тестування та розгортання відбуваються як у Agile, так і в DevOps. Однак в рамках традиційного підходу Agile виключаються операції, які обов'язково входять до DevOps [9].

Технічно DevOps не є методологією SDLC, але вона поділяє мету максимізації успіху програмного проєкту та включає концепції, натхненні Agile. У DevOps визначається як «набір практик, що поєднує розробку програмного забезпечення та ІТ-операції. Він спрямований на скорочення життєвого циклу розробки системи та забезпечення безперервної доставки з високою якістю програмного забезпечення. DevOps доповнює розробку програмного забезпечення Agile; кілька аспектів DevOps виникли з методології Agile.

Підхід DevOps потрібен, щоб об'єднати розробників, які створюють прикладне програмне забезпечення, та фахівців з експлуатації, які запускають програмне забезпечення у робочому середовищі, а також щоб створити та підтримувати робочу інфраструктуру. DevOps замінює старий підхід, в рамках якого команди розробників створювали програми, а потім передавали їх команді з експлуатації, яка займалася розгортанням та керуванням ПЗ без достатніх відомостей про його розробку. У середовищі DevOps команди з розробки та експлуатації працюють спільно протягом усього процесу розробки та розгортання програм, а також управління ними.

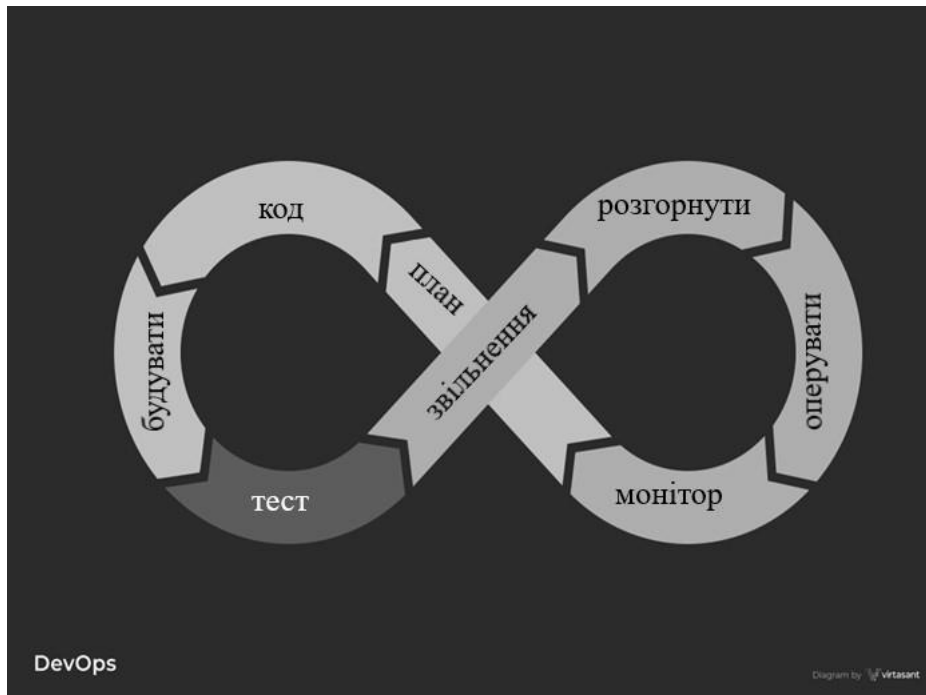


Рис.2.4. DevOps

DevOps, як і Lean, може працювати разом з Agile для створення інфраструктури, яка усуває бар'єри, що сповільнюють розробку та доставку кінцевого програмного продукту. DevOps повністю включає розгортання та роботу програмного забезпечення в процес розробки Agile так само, як Agile внесла тестування та бізнес-аналіз у розробку програмного забезпечення. Зрештою, команда має право бути самодостатньою та брати на себе відповідальність за розробку програмного забезпечення, доставку та підтримку. Вони використовують безперервну доставку (CD) для частих випусків і підтримки добре перевіреної та високоякісної кодової бази.

Переваги DevOps. Групи розробників програмного забезпечення є самодостатніми; доставка та підтримка програмного забезпечення незалежно від ІТ-технічних операцій:

- процес розгортання автоматизований і оптимізований. Молодший розробник може навчитися безпечно розгортати з меншими зусиллями;
- команди впроваджують безперервну інтеграцію / безперервну доставку (CI/CD);

- використовуючи правильні інструменти, інженери економлять час на розгортанні, щоб вони могли зосередитися на кодуванні.

2.2. Аналіз інструментів в управлінні IT-проєктами

Інструменти управління проєктами – це відповідь керівника проєкту на управління проєктами. Для простих проєктів потрібен лише контрольний список, тоді як інші складні вимагають належного планування, призначення завдань, встановлення термінів виконання, переконання, що всі їх дотримуються, і відстеження витраченого часу. Необхідність кількісного визначення, об'єктивізації, розподілу та пропорційного розподілу завдань є надзвичайно важливою, і саме тут на допомогу приходять використання сучасних інструментів. Існує багато програмних інструментів для керування проєктами.

Нижче представлено 15 найкращих інструментів управління проєктами у 2023 році [00]:

1. *Kissflow*. Цей потужний універсальний інструмент є одним із найкращих безкоштовних програм для керування проєктами.

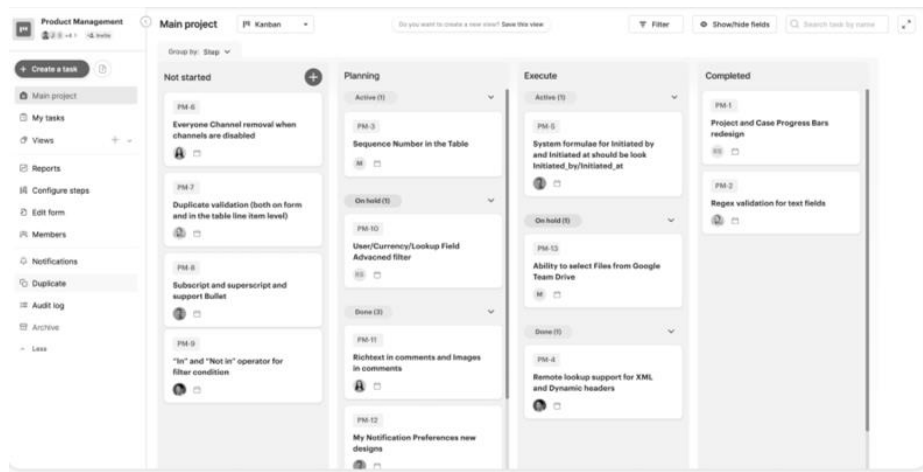


Рис. 2.5. Kissflow

Kissflow Project — це ідеальний варіант інструменту управління проєктами для функціональних менеджерів проєктів і людей, які тільки починають керувати проєктами. Проєкт Kissflow перевіряє всі параметри завдяки комплексному набору функцій керування проєктами. Це простий у

використанні та естетичний графічний інтерфейс користувача, який робить це універсальне рішення чудовим вибором для всіх типів організацій. Він пропонує всі можливості сучасного інструменту проєкту, але орієнтований на функціональних менеджерів із невизначеною відповідальністю за керування проєктами, а не на сертифікованих менеджерів проєктів.

Проєкт Kissflow пропонує широкий набір функцій для різноманітних випадків використання. Нижче наведено деякі з пропонованих особливостей;

- настроюванні кроки проєкту та стани «Активний», «У очікуванні» та «Готово»;
- можливість створювати підзавдання та перетворювати їх на окремі завдання;
- кілька способів візуалізації проєктів (список, Канбан і матричні перегляди);
- спеціальні поля та розширені фільтри для сегментації завдань;
- детальні журнали аудиту, повна історія активності для кожного завдання та контекстні примітки в завданнях;
- можливість додавати спостерігачів за завданнями для забезпечення детальної видимості та можливість встановлення термінів виконання;
- автоматичні сповіщення електронною поштою, коли крайні терміни наближаються;
- інтеграція Guide і Microsoft 365.
- глибокі звіти на основі даних і можливість створювати власні звіти;
- можливість розподіляти ролі членів команди та можливість дублювати проєкти;
- настроюванні та попередньо визначені шаблони керування проєктами

2. *Trello* – це провідне програмне забезпечення для керування онлайн-проєктами з вбудованими інструментами для спільної роботи з командами. Він легкий, простий і зрозумілий у використанні. Він використовує дошки, списки та картки, щоб створювати завдання та підтримувати порядок.

Користувачі можуть співпрацювати в рамках карток, обмінюватися файлами та залишати коментарі.

Не вистачає функцій звітування та функцій відстеження проєктів, таких як Kissflow Project. Він ідеально підходить для окремих людей або невеликих команд, які одночасно керують кількома проєктами.

3. *Asana* — це гнучкий інструмент управління проєктами SaaS з акцентом на співпрацю. Існують робочі процеси, які автоматизують повторювані завдання. Він інтуїтивно зрозумілий із різними дисплеями завдань і підтримує спеціальні поля та форми. Навіть можна запросити необмежену кількість гостей. Є також варіанти для створення спеціальних правил і робочих процесів затвердження.

Asana пропонує багато для продуктивності та співпраці, тоді як багато основних функцій управління проєктами обробляються через інтеграцію, яка може бути ідеальною для не всіх користувачів. Це добре для компаній будь-якого розміру, але не ідеально для фрілансерів. Деякі користувачі вважають, що інтерфейс *Asana* надто простий, а система занадто жорстка, але вона має потужні функції керування завданнями та співпраці. Якщо стиль програмного забезпечення для організації проєкту добре збігається з *Asana*, то це чудово.

4. *Zoho Projects* — популярна пропозиція від *Zoho*, яка допомагає компаніям відстежувати завдання, співпрацювати, відстежувати помилки та створювати інтуїтивно зрозумілі звіти. Онлайн-інструмент керування проєктами пропонує такі функції, як діаграми Ганта та можливість оплачувати години для кількох проєктів одночасно. *Zoho Projects* також може похвалитися дуже вражаючим набором комунікаційних інструментів, включаючи чат у реальному часі та сторінки форуму.

Багато користувачів кажуть, що діаграми Ганта громіздкі у використанні, а розподіл звітів може бути недостатнім для великих команд.

5. *Wrike* — це хмарний інструмент керування проєктами, який спрощує планування проєктів, допомагає відстежувати роботу команди,

контролює терміни та легко співпрацює з усіма зацікавленими сторонами. Він пропонує багато розширених функцій, таких як керування завданнями, діаграми Ганта, стрічка новин у реальному часі тощо. Можна розставити завдання за пріоритетністю, щоб просунути найважливіші завдання вперед і забезпечити дотримання термінів. Існує також багато варіантів інтеграції, що робить його потужним пакетом для управління проєктами.

Величезна кількість функцій може бути трохи переважною для більшості нових користувачів, а не інтуїтивно зрозумілий інтерфейс мало допоможе. Він ідеально підходить для різних команд всередині організації, таких як маркетинг, розробка програмного забезпечення та професійні послуги.

6. *Monday.com* поєднує візуально привабливий інтерфейс і корисні інструменти візуалізації проєкту в інтуїтивно зрозуміле, просте програмне забезпечення для керування проєктами. Він гнучкий і пропонує прозорий спосіб організації роботи. Зверху є шаблони проєктів, за якими ви можливо створювати власні дошки. Доступно багато варіантів відображення, наприклад перегляд карти та календаря.

Однією з поширених причин невдоволення клієнтів є відсутність повторюваних завдань. Мобільний додаток для деяких користувачів незручний для навігації. Вони також пропонують високий рівень безпеки, надійні функції та легке налаштування.

7. *ProofHub* надає централізований робочий простір для списків завдань, робочих процесів, діаграм Ганта, обговорень, календарів і документів. Це допомагає планувати, організовувати та відстежувати завдання команди. Це також полегшує співпрацю всередині команд, а також із зовнішніми клієнтами. Є багато звітів, таких як звіти про навантаження та ресурси.

З наголосом на простоті ProofHub має дуже мало інтеграцій і дуже обмежене керування завданнями. Це добре як для фрілансерів, так і для компаній будь-якого розміру.

8. *Clarizen* — це високо оцінене програмне забезпечення для керування онлайн-проектами, призначене для підприємств середнього та корпоративного розміру. Система керує проектами лише через Інтернет, тому відсутність мобільного додатку може обмежити корисність для деяких команд. Користувачам подобаються надійні функції звітування та можливість відстежувати використання ресурсів, але йому бракує списку справ і відстеження часу.

Крива навчання є дещо крутою, оскільки вона має специфічні способи керування робочими процесами. *Clarizen* найкраще підходить для менеджерів проектів корпоративного рівня, IT-організацій і команд професійних послуг, яким потрібне потужне, гнучке та інтуїтивно зрозуміле програмне забезпечення для керування проектами на основі Інтернету.

9. *Airtable* поєднує найкращі функції електронних таблиць і баз даних. Він налаштовується, простий у використанні та добре працює для підприємств будь-якого розміру. У ньому відсутні надійні функції звітування, що може бути проблематичним для деяких користувачів.

Можна зберігати інформацію в базі даних, яку можна використовувати для керування завданнями та планування проекту. Багато користувачів люблять універсальність організації даних, яка можлива за допомогою електронних таблиць *Airtable*. Є кілька режимів перегляду, наприклад Канбан, список, сітка та календар.

10. *Інструмент Kanban*. Інструмент Kanban є хорошою точкою входу до гнучкої розробки. Це дозволяє візуалізувати робочий процес і пропонує безперебійні функції відстеження часу. Це полегшує співпрацю та відстеження завдань за допомогою сповіщень, звітів і функцій керування командою.

Він надає такі функції, як звітування, обмін документами, власний мобільний додаток тощо, які ідеально підходять для середнього та великого бізнесу. Інструмент Kanban також пропонує корпоративним користувачам можливість розмістити програмне забезпечення на місці.

11. *Zenkit* — це широко поважне рішення для керування проєктами, яке дозволяє вам бути в курсі своїх даних, починаючи з досліджень і мозкового штурму й закінчуючи спільною роботою над ними. Він спеціально орієнтований на команди, які віддають перевагу гнучкому підходу до своєї роботи. Однак він також підтримує традиційні підходи до управління проєктами.

Zenkit — це надійний інструмент управління проєктами, який пропонує численні перегляди, звіти та можливості керування ресурсами, тому завжди на радарі тих, хто шукає універсальну відповідь на потреби своїх проєктів.

12. *MeisterTask* — це онлайн-інструмент управління проєктами для команд, які працюють разом у режимі реального часу. Він пропонує інформаційну панель активності, керування резервами, налаштування поля, можливості обміну файлами тощо.

Він пропонує візуально привабливий інтерфейс користувача, хоча його можливості керування версіями можуть бути покращені.

13. *ClickUp* — це хмарна пропозиція, яка представляє себе як універсальний інструмент для управління проєктами. Він виділяється своїм широким набором функцій, який включає навіть документи, цілі та папку "Вхідні". Модульність є головною перевагою *ClickUp*. Він пропонує додатки під назвою *Click Apps*, які дозволяють багато налаштовувати.

14. *Rayo* представляє себе як сучасне та інтуїтивно зрозуміле рішення для керування роботою для малого бізнесу та віддалених команд. Він включає розширені функції керування завданнями, планування, співпрацю, виставлення рахунків і відстеження часу. По суті, це дозволяє відстежувати весь життєвий цикл проєкту без необхідності використовувати та платити за декілька програм.

15. *Командна робота* — це інструмент управління проєктами на основі Kanban. Він пропонує більш розширені функції звітування, ніж *Trello*, у подібному інтерфейсі перетягування. Це дозволяє створювати

брендинг облікових записів із користувальницькою ідентифікацією компанії.

Крім того, щоб облегшити роботу в управлінні проєктами, розглянуто список із семи основних функцій інструментів управління проєктами, без яких абсолютно не можна жити, якщо керувати успішним проєктом:

- 1) *Канбан дошка*. Дошки Kanban — це робочі простори, які використовуються для візуалізації проєкту, і вони надзвичайно корисні для успішного управління проєктами. На дошці канбан використовується проста система дощок і карток, що дозволяє кожному в команді бачити не лише окремі завдання, а й ширшу картину командної діяльності. Citizen development є чудовим прикладом використання дошки канбан без коду. Створений Toyota Motors для своїх виробничих потужностей, Kanban був адаптований і застосовувався в багатьох секторах протягом останніх кількох десятиліть. Дошка Kanban може бути у фізичному просторі, як-от біла дошка, або в цифрових просторах, які можна створити в програмному забезпеченні дошки Kanban.
- 2) *Управління завданнями*. Проєкт — це набір завдань, які необхідно виконати, щоб досягти однієї мети. Деякі з цих завдань є послідовними або залежними. Деякі з них можна завершити одночасно. Кожне завдання потрібно призначити особі чи особам із команди проєкту. Чим складніший проєкт, тим більше завдань постає перед ним. Хоча це звучить не дуже гламурно, управління завданнями є важливою частиною управління проєктами, і важливо мати інструменти, які допомагають ефективно керувати завданнями.
- 3) *Списки справ*. Старий добрий список справ. Хоча це може бути просто, але воно допомагає вам і вашій команді виконувати роботу. Звичайно, набагато ефективніше зберігати список справ у центральному видимому місці, до якого кожен член команди матиме легкий доступ. Гарне програмне забезпечення для управління

проектами має включати список справ, який гарантує, що кожен член команди може перевірити, що було зроблено та що ще залишилося зробити.

- 4) *Відстеження часу.* Кожен аспект проекту потребує часу, але коли процес підходить до кінця проекту, іноді важко зрозуміти, хто скільки часу витратив на які завдання. Це стає складнішим, якщо є схожий проект; відчувається, що необхідно знати, скільки часу виділити на кожне завдання та фазу проекту, але якщо команда раніше не могла відстежити свій час, це трохи вгадування. Хороший пристрій відстеження часу допомагає з точним виставленням рахунків, проникливим керуванням командою та ретельним плануванням.
- 5) *Співпраця.* Незалежно від того, чи є команда з двох, двадцяти чи двохсот людей, співпраця має життєво важливе значення для успішного завершення проекту. Хороша система управління проектами повинна включати потужні функції співпраці, щоб розробники-громадяни та інші члени команди могли спілкуватися та добре працювати разом як у будь-якій кімнаті, так і по всьому світу. Онлайн-інструменти спільної роботи над проектами включають такі речі, як оновлення в реальному часі, спілкування та автоматичні сповіщення.
- 6) *Інтеграція.* Проекти складаються з багатьох рухомих частин. Потрібно знати вплив однієї змінної на інші щоразу, коли щось змінюється. Іншими словами, шукайте функції, які допомагають аналізувати компроміси та те, як розподіл ресурсів впливає на загальний графік проекту.
- 7) *Детальна звітність.* Звіти — найкращий спосіб отримати стратегічний огляд статусу проектів. Зокрема, потрібно знати, як все відбувається до кінця кожного тижня або місяця. Крім того, варто звернути увагу на програмне забезпечення, яке збирає всю

інформацію, пов'язану з графіком, вартістю, продуктивністю команди та зручністю використання ресурсів, у структурований документ.

Хороший онлайн-інструмент керування проєктами дозволяє створювати звіти про проєкти, які дають команді уявлення про завершення проєкту, стан бюджету, неефективність тощо. Багато програмних засобів управління проєктами мають стандартні функції звітності. Ще корисніше, коли ці звіти можна налаштувати відповідно до вашого проєкту.

Отже, у даному випадку, найкраще для управління проєктом розробки веб-додатку до телефону, буде обрати інструмент Kanban. За його допомогою можна призначати завдання, створювати списки, відстежувати час, співпрацювати та створювати звіти в простому, інтуїтивно зрозумілому інтерфейсі. Тому нижче більш ширше проаналізовано та розглянуто обраний інструмент. Дошка Kanban — це інструмент для візуалізації робочого процесу, розроблений, щоб допомогти надати чіткості робочому процесу та підвищити ефективність шляхом обмеження незавершеної роботи. Завдяки цьому новому рівню прозорості можна швидко виявляти проблемні етапи роботи, а покращуючи їх, команда незабаром працюватиме ефективніше. На дошках Канбан використовуються ліміти карток, стовпців, доріжок і WIP, щоб команди могли візуалізувати робочі процеси та ефективно керувати ними.

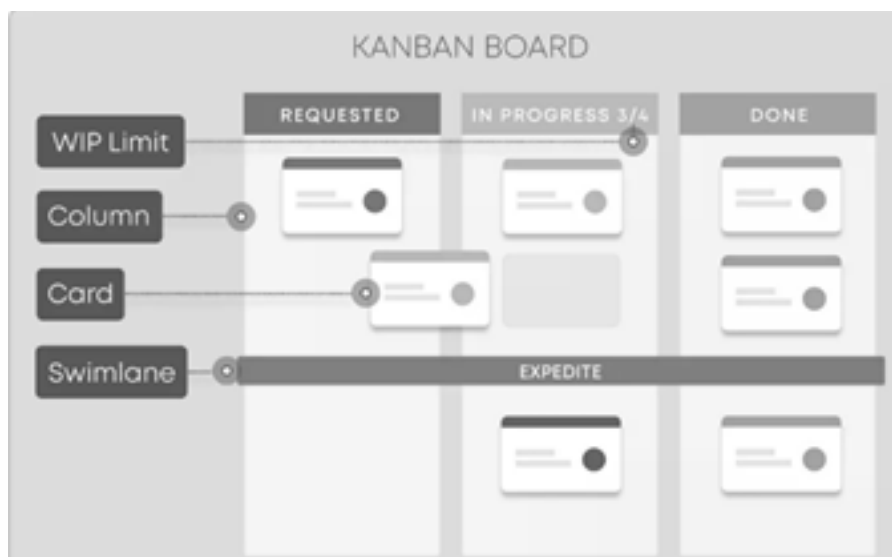


Рис. 2.6. Дошка Канбан

Картки Kanban – це візуальне представлення завдань. Кожна картка містить інформацію про завдання та його статус, наприклад термін виконання, виконавця, опис тощо. Стовпці Kanban – кожен стовпець на дошці представляє окремий етап робочого процесу. Картки проходять робочий процес до повного завершення.

Обмеження незавершеної роботи – вони обмежують максимальну кількість завдань на різних етапах робочого процесу. Обмеження WIP дозволяє швидше завершувати робочі завдання, допомагаючи команді зосередитися лише на поточних завданнях.

Kanban Swimlanes – це горизонтальні доріжки, які можна використовувати для розділення різних видів діяльності, команд, класів обслуговування тощо.

Точка зобов'язання – зобов'язання позначає точку в робочому процесі, де робочий елемент готовий до завантаження в систему. Точка доставки – точка в робочому процесі, де робочі елементи вважаються завершеними.

Впроваджуючи Kanban, можна розпочати невелике та поетапне створення стійких, довгострокових покращень у процесах, забезпечуючи тим самим вищий рівень продуктивності, зменшення стресу та покращену якість, серед інших переваг:

- покращена видимість;
- підвищення продуктивності;
- більша гнучкість;
- покращена концентрація команди;
- зниження рівня відходів ;
- краща співпраця ;
- покращена передбачуваність ;
- відповідність цінностям компанії.

Створюючи дошку Kanban, краще почати із простої структури, поступово вдосконалюючи дошку та перетворюючи її на систему керування робочим процесом, виконавши ці сім кроків:

1. Візуалізація робочого процесу на дошці Kanban. Використовуючи дошку Kanban, щоб відобразити всі етапи робочого процесу. Дивлячись на дошку, необхідно мати можливість зрозуміти, як виконується робота. Почати із базової структури дошки Kanban і розділити її на кілька основних розділів, які показують різні етапи роботи.

2. Виявити вузькі місця робочого процесу на дошці Kanban. Дошка Kanban є ідеальним інструментом для візуалізації потенційних проблем у процесі. Логіка проста: якщо дивитися на колонку, в якій завдання приходять швидше, ніж відходять, робота почне накопичуватися, і проблема стане видимою для всієї команди. Це може бути спричинено тимчасовою проблемою або вузьким місцем у процесі.

Наприклад, на простій дошці Канбан із лише одним розділом «Виконується» та багатьма картками, які очікують на активність, буде складно визначити, чому робота зупиняється та як можна полегшити ситуацію. І навпаки, детальне відображення робочого процесу за допомогою спеціальних стовпців для всіх робочих етапів дозволить з першого погляду зрозуміти, де можна в нести покращення.

Як тільки помітно проблемний стовпець/вузьке місце, обов'язково потрібно діяти швидко, щоб вирішити його та запобігти його повторенню. Найпростіше, що можна зробити, це обмежити незавершену роботу раніше в потоці. Таким чином надати більше часу людям, які «володіють» цією стадією, дозволяючи їм усунути вузьке місце. Іншим варіантом може бути перерозподіл зусиль команди для задоволення нових вимог. Kanban створено для забезпечення максимальної гнучкості, таким чином залишаючись за кермом у прийнятті всіх важливих рішень.

3. Використовувати дошку Канбан, щоб обмежити поточну роботу та зосередитися. Дошка Kanban — це чудовий спосіб позбутися команді від

багатозадачності, застосовуючи обмеження WIP (Work In Progress) відповідно до можливостей. Можна використовувати обмеження на загальну кількість завдань, які можуть виконуватися одночасно на дошці, або встановити індивідуальні обмеження на кожному етапі робочого процесу. Якщо є бажання збільшити відсоток надання цінності клієнтам, слід зосередити кожен команду на завершенні роботи, а не починати нову .

Таким чином вони перестануть починати нову роботу і зосередяться на завершенні завдань, які вже виконуються. Знання того, що кожен бачить, що робить кожна людина, також є чудовим мотиватором для постійного досягнення кращих результатів.

4. Використовувати інформаційну панель Kanban, щоб заощадити час, витрачений на непотрібні зустрічі. Значною перевагою впровадження дощок Kanban є те, що вони економлять час, витрачений на зустрічі, звіти про хід роботи та багато інших непотрібних перерв. Використовуючи дошку Kanban як спеціальне сховище інформації. Це поширюватиме знання про те, хто що робить у будь-який час. Завдяки дошці Канбан всі члени команди будуть в курсі того, як просуваються завдання. Таким чином, не потрібно буде так часто планувати зустрічі, щоб отримувати оновлення прогресу.

Крім того, дошка Kanban дозволить забути про ті довгі звіти про прогрес, які готуються багато років і вже застаріли до моменту, коли їх представили зацікавленим сторонам.

5. Візуалізувати робочі блокувальники на дошці Kanban. Дошка Kanban дозволяє візуалізувати проблеми, які заважають команді виконати будь-яке завдання. На типовій фізичній дошці блокери зазвичай позначаються червоними магнітами (або шпильками, якщо використовувати пробкову дошку). Заблокована картка на дошці Kanban. Використовувати візуальні блокери на дошці Kanban.

Програмне забезпечення Digital Kanban має більш просунуті способи візуалізації блокувальників (наприклад, знаки зупинки на заблокованих картках). Коли щось заважає команді продовжувати роботу над завданням,

вони можуть позначити його як заблоковане та почати працювати над іншим завданням, не порушуючи обмежень WIP.

6. Автоматизувати робочі процеси на Agile Kanban Board. Автоматизація робочих операцій — чудовий спосіб побудувати зрілу систему керування робочим процесом. Часто командам доводиться вирішувати повторювані завдання, швидко сповіщати своїх колег про невідкладні зміни, встановлювати нагадування про крайні терміни та плавно передавати роботу між собою. Усе це може зайняти багато часу, створити марнотратство та хаос у системі, якщо обробляти їх вручну.

Ось чому сучасні дошки Kanban дозволяють застосовувати правила автоматизації до ваших завдань і оптимізувати їх потік у системі. Зазвичай вони мають форму правил «якщо-це-то-то», які дозволяють запускати попередньо визначені дії на дошці Канбан щоразу, коли щось трапляється.

Правило автоматизації на дошці Kanban -застосувати автоматизацію до дошки Kanban. Таким чином можна автоматизувати повторювані процеси, де це можливо, скоротити марнотратну діяльність і підвищити ефективність системи управління робочим процесом.

7. Збирати ключові показники робочого процесу та вдосконалювати. Сучасні цифрові дошки Kanban можуть автоматично збирати інформацію про тривалість циклу завдань, час виконання та інші ключові показники ефективності. Це допоможе приймати керовані даними рішення щодо будь-яких змін у вашому процесі та заощадить багато часу, яке б витрачали на збір показників вручну.

Використовуючи, наприклад, Kanbanize by Businessmap, дає можливість використовувати кілька діаграм на дошці Kanban, де можна вимірювати дані робочого процесу та постійно оптимізувати виконання завдань і проєктів. Показники надання послуг. Використовувати показники Lean/Agile і діаграми. Крім того, знання того, як користуватися дошкою Kanban, допоможе набагато краще розставляти пріоритети завдань. Завдяки

візуалізації всього в одному місці вся команда завжди буде на одному шляху.

2.3. Формування команди в ІТ-проєктах для розробки web-додадку

Проаналізувавши різні методології, вибравши рамки, які найбільше відповідають даному проєкту, команді та цілям проєкту, і визначивши інструмент управління проєктом, який відповідає даним потребам. Протягом усього життєвого циклу проєкту спільна робота з кожним членом команди забезпечить проєкту функціонування як згуртовану одиницю.

Але вибрати правильних людей для виконання роботи, не означає, що вони завжди співпрацюватимуть найкращим чином. Різні робочі звички, стилі спілкування, короткострокові та довгострокові цілі можуть збивати всю команду (і проєкт) з колії. Роз'яснення поведінки, яку потрібно практикувати у команді, і встановлення ключових цінностей з самого початку змушує всіх стати на правильну ногу.

Розглянемо кілька найкращих методів створення та керування командою проєкту, щоб заохотити співпрацю для досягнення оптимальної продуктивності.

Важливість співпраці в управлінні проєктами. Перш ніж перейти до створення командного середовища для спільної роботи, визначино, чому співпраця важлива. Без співпраці компанія застоїться. Потрібні об'єднані ідеї та робота команди, щоб реалізувати складний проєкт, впроваджувати інновації та створювати продукти, які перевершують конкурентів.

Співпраця важлива з двох основних причин: перша – це внутрішня співпраця команди підвищує продуктивність. Коли внутрішні команди використовують інструменти віддаленої співпраці та робочі процеси, які роблять спілкування та співпрацю ефективнішими, вони можуть швидше досягати цілей і виконувати роботу кращої якості. Внутрішня продуктивність підвищується завдяки плавній співпраці. Використовуючи

інструменти спільної роботи Wrike, каліфорнійська технологічна інноваційна компанія Tactus змогла централізувати інформацію про свій продукт і скоротити періоди Scrum на 80% . Друга причина – це співпраця із зовнішніми зацікавленими сторонами підвищує інноваційність.

Сьогодні компанії, які намагаються робити все своїми силами, обмежені — неможливо бути хорошим у всьому постійно. Їм потрібні сторонні ідеї та відгуки, щоб розвиватися. Те саме стосується команди проєкту. Коли команди працюють разом, вони досягають неймовірних результатів .

Створення команди проєкту. Команда проєкту – це група людей, які працюють над спільною метою, використовуючи цінні та унікальні навички. Ідентифікація членів команди проєкту, визначення ідентичності команди та стандартизація її робочих практик є критичними для успішного проєкту.

Для створення команди проєкту враховують:

- потреби проєкту – розуміння обсягу проєкту насамперед дає змогу стратегічно вибрати, хто має бути в команді;
- набір навичок – вибір членів команди, які можуть запропонувати різноманітний набір навичок, має вирішальне значення. Якщо команді бракує певного набору навичок, завдання може бути виконано неправильно. Занадто багато людей з однаковими навичками можуть спричинити плутанину щодо власності;
- спроможність – навіть якщо знайти ідеальну людину для частини проєкту, якщо вона перевантажена роботою, вона може стати перешкодою. Знайти членів команди, які мають доступність у своїх майбутніх розкладах проєктів.
- стилі роботи – різні люди мають різні стилі роботи та характери. Важливо зрозуміти, як ці відмінності можуть вплинути на динаміку команди, і прийняти їх після початку проєкту.

Проектна команда може відрізнятись залежно від обсягу проекту та необхідних навичок, але елементи успішної проектної команди більш-менш узгоджені за всіма напрямками:

- сильне керівництво командою – кожній команді потрібен лідер, який може виявити найкраще у своїх товаришах по команді. Розуміння індивідуальних сильних сторін кожного члена вашої команди та знання того, як їх виявити, є життєво важливими для успіху команди в цілому;
- чіткі цілі та мета – коли дорожня карта та цілі чіткі, команді набагато легше зрозуміти, куди вони рухаються та наскільки їхній внесок матиме значення надалі. Потрібно заохочувати їх об'єднатися навколо справи з чітким баченням;
- стандартизовані робочі процедури – коли виникають конфлікти або потрібно призначити завдання, команди повинні знати, як діяти.
- різноманітність – дослідження показують, що різноманітність покращує продуктивність. Сильна проектна команда включатиме різноманітні навички та досвід;
- час спілкування – знаходження часу для членів команди проекту, щоб пізнати один одного та налагодити стосунки, сприяє зростанню довіри, руйнуванню бар'єрів і відкриває шляхи спілкування.

Після початку проекту, робота проектного менеджера — підтримувати мотивацію, підтримувати всіх у курсі та працювати разом.

Робоче середовище для спільної роботи — це не лише наявність інструментів, процесів і робочих процесів для спільної роботи. Це також прагнення до спільних цілей, незважаючи на відмінності в стилях роботи, особистих пріоритетах або культурному середовищі. Робоче середовище для співпраці має бути безпечним простором, де кожен відчуває підтримку та можливість висловити свою думку.

У міру того як організації ростуть і стають все більш розпорошеними, все більше працівників співпрацюють на великих відстанях. Як наслідок,

культурний інтелект — здатність приймати різні погляди в складних ситуаціях — стає все більш і більш важливим.

Практичні поради щодо формування команди:

- дізнатися один про одного: знайти час поговорити про речі, які не стосуються роботи чи проєкту;
- використовувати різні стилі навчання та спілкування;
- завжди повторювати основні моменти наради;
- завжди чітко визначати обов'язки та наступні кроки для кожного члена команди;
- залучати всіх: у різних людей різні стилі спілкування. Те, що хтось не говорить, не означає, що у нього немає хороших ідей;
- додавати гумор (де це доречно): це не означає, що потрібно весь час жартувати про все, але підняття настрою може допомогти людям відкритися та покращити робочі стосунки.

Може бути легко втратити з поля зору найкращі методи співпраці через навислі терміни, але впровадження їх протягом усього проєкту допоможе оптимізувати роботу команди. Не менш необхідним кроком у формуванні команди над проєктом – це зустрічі .

Хоча зустрічі можуть бути цінним часом для співпраці, надмірно тривалі зустрічі можуть вбити продуктивність. Із глобальним розвитком віддаленої роботи організація ефективних і ефективних віртуальних зустрічей може бути особливо складною.

Віддалена співпраця стає все більш поширеною, і тепер вона вважається нормою для тисяч компаній у всьому світі. Сучасним працівникам і робочим місцям необхідно адаптуватися до дистанційної роботи та заохочувати співпрацю, незалежно від того, де вони знаходяться чи в якому часовому поясі.

Під час керування проєктом розподіленої команди в різних місцях програмне забезпечення для співпраці (або інструмент керування проєктами) є ефективним способом тримати всіх на одній сторінці та всю

інформацію про проєкт в одному місці. Однак використання чудового інструменту не допоможе чарівним чином розпочати співпрацю. Безсумнівно, співпраця важлива не лише для успіху проєктів, але й для подальшого зростання організації. Це важливий інгредієнт для повсякденної роботи над проєктами, але, що важливіше, це X-фактор, який визначає культуру даної організації та залучає до компанії найкращих виконавців. Також залишається фактом: складні проєкти дозволяють командам легко втратити коло завдань. Може бути герметичний план проєкту та зіркова команда для його підтримки. Але якщо не має чіткого розуміння завдання — або навіть участь — на рівні завдання, у проєктну команду вкрадеться плутанина, капризність і навіть демотивація.

Діаграма RACI — це матриця, яка використовується в управлінні проєктами для чіткого розподілу ролей і відповідальності за завдання проєкту та результати. RACI означає відповідальний, підзвітний, консультований та поінформований.

Діаграма RACI, також відома як матриця розподілу відповідальності, допомагає усунути плутанину в проєкті, визначаючи, хто що робить на більш детальному рівні, ніж звичайні призначення завдань. На прикладі розробки web-додадку для андроїд представлено діаграму RACI (додаток 1).

По суті, діаграма RACI допомагає чітко визначити очікування щодо проєктних ролей і обов'язків. Таким чином не буде кількох людей, які працюють над одним завданням або один проти одного, оскільки завдання не були чітко визначені на інтерфейсі.

Матриця RACI також заохочує членів команди брати на себе відповідальність за свою роботу або довіряти комусь іншому, коли це необхідно. По суті, це усуне особисті судження та політику зі свого процесу та концентрується на здатності команди діяти відповідально в рамках, які було створено.

Кожна літера в аббревіатурі RACI означає рівень відповідальності кожної особи, яка бере участь у проєкті, щодо окремого завдання, етапу чи рішення. Ця проста діаграма дає миттєве уявлення про значення RACI та кількість людей, які потрібно призначити на кожну роль у матриці RACI.

	визначення	номер для призначення
відповідальний	виконує роботу для виконання завдань	принаймні 1 на завдання
підзвітний	уповноважені особи працюють і є останніми, хто переглядає завдання або результат до того, як вони будуть вважатися виконаними	обмеження до 1 на завдання
проконсультувався	Надає вхідні дані на основі того, як це вплине на їх сферу знань щодо результату	ні максимуму, ні мінімуму
інформований	треба бути в курсі прогресу проєкту, а не вдаватися в деталі кожного результату	ні максимуму, ні мінімуму

Рис. 2.7. Пояснення визначення RACI

Характеристика кожної ролі у RACI [13]:

- Відповідальні (R • Responsible): особи, які будуть виконувати роботу над завданням.
- Підзвітний (A • Accountable): особа, яка в кінцевому підсумку несе відповідальність за успішне виконання завдання і є особою, яка приймає рішення. Тільки одна зацікавлена сторона отримує таке завдання.
- Зацікавлена сторона (C • Consulted): зацікавлена сторона або група зацікавлених сторін, яку попросять надати думку або інформацію про завдання. Таке завдання надається профільним експертам (the subject matter experts - SMEs).
- Поінформована (I • Informed): зацікавлена сторона або група зацікавлених сторін, яка постійно знаходиться в курсі завдання та повідомляється про його результати. Інформування відрізняється від консультування оскільки у випадку з Informed комунікація є

односпрямованою (бізнес-аналітик - зацікавлена сторона), а у випадку з Consulted - односпрямованою (бізнес-аналітик - зацікавлена сторона), а при консультаціях - двосторонній.

Діаграма RACI добре підходить практично кожному проєкту. Але це особливо корисно, коли завдання вимагають кількох ресурсів, виконуються одночасно або залежать від інших завдань. Наведемо кілька сценаріїв, коли така модель корисна:

- процес прийняття рішень або затвердження може затримати проєкт;
- існує конфлікт щодо володіння завданням або прийняття рішень;
- складається враження, що навантаження на проєкт розподілене нерівномірно;
- відчувається мінливість у команді, і потрібно швидко залучити когось до нової ролі.

Звичайно, не всі команди та проєкти створюються однаково. Можна працювати з командою, яка справді добре спілкується та стежить за своєю роботою. Або, можливо, проєкт настільки малий, що було б нерозумно витрачати час на виконання цієї справи. Щодо конкретного прикладу розглянутого в даній роботі, розробка web-додадку для Андроїд, така діаграма дуже доречна та має місце на існування в такому IT- проєкті (додаток 1). Одним із головних завдань є створення «реєстру зацікавлених осіб», де чітко зазначено усіх зацікавлених осіб при виконанні проєкту та їх роль і обов'язки в ньому (додаток 2).

Висновки до розділу 2. Отже, в даному розділі було проаналізовано деякі методи управління IT-проєктами сімейства Agile і доведено, що ці методи надають гнучкість і можливість швидко реагувати на зміни вимог клієнтів чи умов ринку. Також розглянуто можливі інструменти в управлінні цих проєктів. Розглянуто важливі аспекти для формування команди з розробки web-додадку.

РОЗДІЛ 3. УДОСКОНАЛЕННЯ МОДЕЛІ УПРАВЛІННЯ ІТ-ПРОЄКТУ ДЛЯ РОЗРОБКИ WEB-ДОДАТКУ

3.1. Удосконалення моделей управління командою в ІТ- проєкті для розробки web-додатку програмного забезпечення на базі Agile

Agile-моделі управління командою в ІТ-проєкті для розробки web-додатку сприяють покращенню якості та ефективності розробки, оскільки вони засновані на принципах гнучкості та адаптивності. В Agile-методологіях команда працює над проєктом у коротких ітераціях, що дозволяє швидко реагувати на зміни та покращувати продукт на кожному етапі. Важливим елементом Agile-моделей є постійний обмін інформацією між членами команди, що дозволяє прискорити прийняття рішень і зменшити кількість помилок. Також Agile-моделі надають можливість постійної оцінки якості продукту та процесу розробки, що дозволяє команді швидко вносити зміни для поліпшення результатів. Загалом, Agile-моделі управління командою в ІТ-проєкті для розробки web-додатку сприяють більш ефективній та якісній роботі команди, що призводить до покращення результатів проєкту.

Гнучкість та адаптивність: Agile-методології дозволяють командам швидко реагувати на зміни у вимогах та умовах проєкту. Вони дозволяють гнучко змінювати пріоритети, додавати нові функції та вносити зміни у процес розробки.

Участь замовника: Agile-моделі активно залучають замовника до процесу розробки. Замовник має можливість бачити результати роботи команди на ранніх стадіях проєкту та давати зворотний зв'язок. Це дозволяє уточнювати вимоги та вносити зміни в реальному часі, що збільшує шанси на створення кінцевого продукту, що повністю відповідає потребам замовника.

Поліпшена комунікація: Agile-моделі ставлять особливий акцент на комунікацію всередині команди та із замовником. Регулярні зустрічі,

демонстрації проміжних результатів та зворотний зв'язок допомагають покращити розуміння вимог та очікувань усіх учасників проєкту.

Швидка доставка цінності: Agile моделі дозволяють доставляти цінність замовнику на ранніх етапах проєкту. Команди розробляють продукт поетапно, надаючи працездатні версії товару кожної ітерації.

Тому для того щоб ще більш удосконалити управління командою на базі Agile проаналізовано та розглянуто метод лідерство – слуга.

Лідерство-слуга – це стиль лідерства, який підкреслює потреби команди та здоров'я організації вище потреб лідера. Такий підхід дає змогу членам команди досягати своїх цілей і рости особисто та професійно. Розробка стратегії розвитку співробітників, яка акцентує увагу на лідерстві слуги, дозволить визначити пріоритети їхніх потреб, зміцнити довіру, надати підтримку та створити позитивне робоче середовище.

Нещодавнє дослідження, проведене Business Perspectives, показало, що 82,5% тих, хто керував слугою, повідомили про підвищений рівень задоволеності роботою та розширення можливостей. Але більшості компаній не вдається запровадити даний спосіб керування.

Швидше за все, це тому, що багато власників і менеджерів бізнесу все ще застрягли в традиційному «командно-контрольному» стилі управління змінами. Щоб по-справжньому запровадити стиль керівництва слуги, потрібно спочатку поставити потреби членів команди вище власних. Це важко для багатьох менеджерів і власників бізнесу, оскільки вимагає надзвичайної самовіддачі.

Коли беруться на себе зобов'язання обслуговувати членів команди, необхідно переконатися, що надано їм необхідну підтримку та контекстне керівництво. Це включає в себе надання їм чітких очікувань щодо їхньої ролі, надання зворотного зв'язку та чіткі вказівки щодо їхньої роботи, одночасно створюючи позитивне робоче середовище, де члени команди відчують для себе комфортне ставити запитання та висловлювати ідеї.

У цьому розділі розглянуто, як можна запровадити стиль керівництва слуги у бізнесі та його переваги для членів команди. Такий стиль удосконалив управління командою щодо успішного розширення можливостей команди за допомогою лідерства в контексті сучасного робочого середовища.

Сервісне лідерство — це філософія лідерства, яка підкреслює відповідальність лідера за пріоритетність потреб і розвитку інших, таких як співробітників або клієнтів, над власними інтересами.

Цей підхід спрямований на досягнення особистих і організаційних цілей шляхом надання повноважень і служіння тим, хто знаходиться під владою лідера, а не контролюючи їх. Ті, хто практикує лідерство слуги, створюють сприятливе робоче середовище для співпраці, чуйно слухаючи свою команду, розвиваючи навички та здібності своїх співробітників і надихаючи їх сприяти досягненню цілей ефективності організації.

Ставлячи пріоритетом потреби інших, лідери-слуги культивують культуру довіри, поваги та зростання у своїй організації.



Рис. 3.1. лідерство слуги.

У сучасному швидкоплинному діловому світі все більш необхідним для ефективного управління командою стає лідерство на основі – служіння, яке:

1. Заохочує розширення можливостей працівників. Лідерство надає працівникам повноваження приймати рішення та брати на себе відповідальність за свою роботу. Надання їм відчуття самостійності та довіри робить їх більш залученими та відповідальними за свої дії, що зрештою призводить до підвищення задоволеності роботою та продуктивності.

2. Виховує культуру довіри. Коли лідери визначають пріоритети потреб членів своєї команди, це сприяє розвитку культури довіри. Вони відчують, що їх цінують і поважають, що створює середовище, де люди частіше діляться ідеями, співпрацюють і ризикують.

3. Сприяє кращому спілкуванню. Лідерам-слугам слід надавати пріоритет активному слуханню та заохочувати відкрите спілкування. Лідери можуть отримати цінну інформацію та прийняти обґрунтовані рішення, створивши середовище, у якому співробітникам буде комфортно ділитися своїми думками та проблемами.

4. Покращує утримання співробітників. Співробітники, які відчують цінність і підтримку, швидше за все, залишаться в компанії надовго. Керівництво слугою сприяє відчуттю спільності та спільній меті, що може допомогти створити лояльність і зменшити плинність кадрів.

5. Підвищує загальну продуктивність. Зрештою, лідерство слуги може призвести до кращої загальної продуктивності організації. Лідери можуть створити більш зацікавлену та мотивовану робочу силу, сприяючи розвитку культури співпраці, довіри та розширення можливостей.

Отже, лідерство на службі та традиційне лідерство — це два різні стилі лідерства, які привернули значну увагу та обговорювалися в діловому світі.

У той час як традиційне лідерство наголошує на владі, контролі та владі прийняття рішень, лідерство-слуга зосереджується на служінні потребам інших і наданні їм повноважень для досягнення своїх цілей. У цьому розділі порівняно обидва стилі лідерства.

Традиційне лідерство - це підхід зверху вниз, коли лідер має абсолютний контроль над прийняттям рішень. Більше направлений на метод WaterFall. Керівники, які приймають цей стиль, часто ставлять свої власні інтереси та цілі вище інтересів своїх підлеглих. Вони покладаються на свою владну позицію, щоб віддавати накази та очікують дотримання від членів своєї команди. Цей стиль керівництва ефективний у ситуаціях, коли потрібно швидко приймати рішення, або в дуже структурованому середовищі, наприклад у військових чи правоохоронних органах.

Лідерство слуги, яке використовується на основі методу Agile, наголошує на тому, що потреби інших ставлять на перше місце та дають їм змогу реалізувати свій потенціал. Лідери-слуги діють як фасилітатори, тренери та наставники, допомагаючи членам команди розвивати свої навички, знання та здібності. Цей стиль лідерства вимагає високого рівня емпатії, емоційного інтелекту та терпіння. Лідери-слуги частіше делегують відповідальність і повноваження приймати рішення членам своєї команди, що створює відчуття власності та мотивації.

Одна з ключових відмінностей між лідерством-слугою та традиційним лідерством полягає в тому, як вони вимірюють успіх. Традиційні лідери можуть зосередитися на досягненні короткострокових цілей, таких як досягнення цілей продажів або збільшення прибутку.

Навпаки, керівники-слуги надають перевагу розвитку та добробуту членів своєї команди, а не миттєвим результатам. Вони прагнуть створити позитивне та сприятливе середовище, де кожен може рости та процвітати.



Рис.3.2. Впровадження стилю лідерства слуги

Включення стилю лідерства слуги в існуючі стратегії та процеси має вирішальне значення для виховання культури емпатії, співпраці та відповідальності. Розставляючи пріоритети для потреб членів команди та надаючи їм повноваження приймати рішення, службовці-лідери створюють відчуття власності та відданості, що сприяє продуктивності та інноваціям.

Такий підхід допомагає зміцнити довіру та лояльність всередині команди, що веде до довгострокового успіху та сталого зростання.

Команда для роботи над ІТ-проєктом, як правило складається з технічних інженерів та менеджерів.

Інженери використовують науку та математику для розробки нових продуктів, включаючи машини та програмні рішення. Вони займаються тестуванням, виробництвом і обслуговуванням на додаток до проєктування та розробки продукту. Вони повинні розрахувати кількість часу та грошей, необхідних для реалізації проєктів. Комунікативні навички, увага до деталей і здатність працювати разом і під напругою — все це важливі навички для інженерів.

Нижче наведено список деяких доступних посад технічного інженера в ІТ сфері:

1. Інженери-програмісти створюють і проєктують програмні додатки відповідно до вимог клієнтів. Після завершення їм також потрібно буде підтримувати та налагоджувати їх. Вони повинні знати різні мови програмування через різноманітність роботи, до якої вони залучені. Навички критичного мислення та навички спілкування також важливі.

2. Інженер з якості оцінює якість процесів і продуктів і вносить пропозиції щодо вдосконалення специфікацій. Вони відповідають за тестування продуктів, систем і процесів, щоб перевірити, чи відповідають вони технічним вимогам. Вони повинні надати документацію з якості та безпеки, а також мати можливість координувати різні команди всередині компанії з інженерів, виробників і клієнтів, щоб виявляти проблеми, пов'язані з якістю.

3. Інженер DevOps — це людина, яка поєднує розробку та кодування. Вони співпрацюють з різними відділами для проєктування та розвитку існуючих мереж. Інженер DevOps підвищує ефективність робочого місця, створюючи та розгортаючи системне програмне забезпечення, а також аналізуючи дані для покращення поточних. Вони відповідають за збалансування багатьох частин проєкту, більшість із яких є складними проблемами, такими як програмування та розробка мережі.

4. Інженери підтримки допомагають клієнтам і компаніям із технічними питаннями та внутрішніми проблемами. Вони повинні мати чудові комунікативні навички, оскільки вони часто мають справу зі дзвінками споживачів, щоб допомогти вирішити технологічні проблеми. Інженери служби підтримки також можуть повідомляти про недоліки продукту або надавати ідеї щодо вдосконалення. Вони сприяють розширенню знань про продукт і спрощують технічну підтримку в майбутньому.

5. Інженери даних відповідають за створення та керування платформами даних, вони допомагають забезпечити безперебійний потік і доступ до інформації. Інженери з даних повинні володіти такими мовами

програмування, як Python і Java, а також добре розуміти алгоритми та основи машинного навчання.

6. Інженери-механіки створюють механічне обладнання та системи, включаючи машини, інструменти та двигуни. Вони відповідають за їх проєктування, будівництво, випробування та інспектування. Інженери-механіки працюють на багатьох підприємствах, створюючи широкий асортимент продукції, оскільки машинобудування — це така велика дисципліна.

7. Веб-інженери є експертами у створенні інтернет-додатків і програмних засобів. Обов'язки включають розробку функціональності програми, впровадження мережових комунікаційних систем, створення зручних інтерфейсів і розробку веб-сайтів.

8. Інженери з безпеки відповідають за цифрову безпеку компанії або клієнта. Вони відповідають за встановлення процедур цифрової безпеки, адміністрування системи кібербезпеки та підтримку архітектури ІТ-безпеки для своєї компанії чи клієнтів.

9. Хмарний інженер відповідає за проєктування, управління та підтримку хмарної інфраструктури. Хмарний інженер відповідає за низку завдань і повинен мати глибоке розуміння розробки програмного забезпечення, мов сценаріїв і досвід створення хмарних веб-сервісів.

10. Інженер-випробувач виявляє потенційні проблеми та гарантує якість продукту до його офіційного запуску. Інженер-випробувач повинен володіти сильними комунікаційними навичками, оскільки він працює з різними відділами, включаючи мережових інженерів, службу підтримки та операцій, щоб забезпечити наявність плану для безперебійного запуску продукту. Вони відповідають за пошук інноваційних технологій і економічно ефективних варіантів. Інженер-випробувач прагне знайти баланс між багатьма компонентами проєкту, включаючи безпеку та дизайн.

На прикладі, розробки web-додатку до програмного забезпечення, розглянуто склад команди, який було залучено до проєкту.

Таблиця 3.1.

Зацікавлені сторони	Ролі
Розробники програмного забезпечення	відповідальний за створення додатків або програмного забезпечення, яке дозволить підключити телефон до пристрою.
Дизайнери інтерфейсу користувача	займається створенням зручного та інтуїтивно зрозумілого інтерфейсу, що дозволяє легко взаємодіяти з пристроєм
Інженери підтримки	щоб забезпечити ефективну підтримку та обслуговування для користувачів, які можуть зіткнутися з проблемами під час використання телефону з телефоном.
Виробники обладнання	забезпечить фізичну основу для підключення телефону до програми, тобто відповідне обладнання та компоненти.
Тестери	випробування компонентів, систем, транспортних засобів і схвалення регуляторів .
Відділ маркетингу	відповідає за просування продукту та позиціонування його на ринку для охоплення максимальної аудиторії.
Керівник проєкту	керує процесом розробки в цілому, забезпечуючи координацію роботи всіх залучених стейкхолдерів та виконання завдань згідно з графіком.

Відзначення успіхів команди та визнання їх наполегливої праці та досягнень є важливими для підвищення морального духу та мотивації. Визнавати внесок членів команди та давати їм знати, що їхні зусилля цінуються. Коли члени команди знають, що їх наполегливу роботу визнають і цінують, вони, швидше за все, відчують, що інвестують у свою роботу.

Таким чином розробка будь-якого проєкту буде мати більш велику ефективність та віддачу. Це не призведе до заторів у роботі. Завдяки злагодженості роботи можна уникнути додаткових витрат коштів, часу та вигорання членів команди. Удосконалене та гнучке управління командою завжди на крок попереду від традиційного методу.

3.2. Удосконалення структури розподілу роботи ІТ-проєкту для розробки додатку програмного забезпечення.

Work-breakdown structure (WBS) – розбиття проєкту на конкретні задачі, які повинні бути виконані для досягнення цілей проєкту.

Структура розподілу робіт (WBS): орієнтоване на продукт «генеалогічне дерево» компонентів проєкту, яке організовує та визначає загальний обсяг проєкту. Кожен низхідний рівень представляє все більш детальне визначення компонента проєкту. Компонентами проєкту можуть бути продукти або послуги [14].

До цього часу обравши, як приклад web-додаток до програмного забезпечення «Ментальне здоров'я», розглянемо удосконалення структури розподілу роботи. Визначивши характер статуту даного проєкту, що зазначений в додатку 4 та визначивши, хто буде ключовими людьми в команді проєкту. Знання їхніх конкретних навичок і талантів впливає на те, як саме удосконалити розробку WBS і як організувати проєкт. Подібним чином розмірковувати, де проєкт потрапляє в континуум від прогнозованої до слабкої матриці. Наприклад, якщо проєкт повністю спроектований, швидше за все, матиме більше гнучкості у розподілі обов'язків. У матричній організації можна мати менше контролю над тим, хто працює над деякими аспектами проєкту, і слід узгоджувати визначення робочого пакета з організаційними елементами, які фактично виконуватимуть роботу.

У проєкті «Ментальне здоров'я», відповідно до статуту проєкту, зазначено 7 членів команди. Ця кількість людей, які працюють над проєктом, також матиме вплив, оскільки меншими проєктами можна керувати з більшою гнучкістю, ніж більшими.

Іноді статут проєкту диктує склад команди. З іншого боку, РМ може мати можливість вибрати людей для своєї команди. Здібності людей у команді можуть вплинути на те, як саме організована робота, і, таким чином, вплинути на те, як визначити WBS. Оптимальний план має як найбільше використовувати сильні та слабкі сторони окремих осіб. Часто доводиться шукати компроміси через наявність ключових осіб, які повертаються з інших проєктів.

Проект складається завдяки зусиллям кількох субпідрядників і постачальників. Слід подбати про те, щоб WBS відповідав унікальним можливостям цих постачальників. Наприклад, часто бажано, якщо це можливо, щоб усі компоненти основних елементів постачалися за контрактом «під ключ». Це може суттєво зменшити клопоти та вказувати пальцями, коли функція не працює належним чином. Наприклад, для розробки даного проекту необхідно мати певну кількість оргтехніки та ліцензовані програми, та інше оснащення яке необхідне для роботи над проектом. Щоб запобігти затримки в роботі, тому що був один постачальник основного компонента, і більша частина його потужностей була поглинена контрактом, укладеним за кілька місяців до того, як контракт використання мобільного пристрою був зданий в оренду.

WBS, який додано до додатку 5, є засобом для керівника проекту визначити стратегію, за якою буде виконуватися проект. Частина стратегії продиктована технологією. Наприклад, при розробці додатка «Ментальне здоров'я» необхідно додатково залучити спеціалістів таких як психологи, дієтологи, фітнес-інструктори. Через специфіку додатку, який буде запущено в роботу. Це продиктувало послідовність додаткової іншої роботи для забезпечення використання продукту користувачами. Також у проекті програмного забезпечення може бути важливо розробити, закодувати та протестувати багато підпрограм допоміжних програм до завершення кодування для більшості підпрограм додатків.

Деякі стратегічні рішення є результатом ресурсів, які можна залучити до проекту. Деякі є відображенням політичного тиску, реального та потенційного, внутрішнього та зовнішнього, з яким стикається проект. Деякі з них є особистими уподобаннями керівника проекту. Особисті уподобання мають ґрунтуватися більше на слабких сторонах керівника проекту, ніж на сильних, міркуванні, яке стає важливішим із масштабістю проекту.

Одним із критеріїв хорошого WBS є те, що точки контакту між членами команди мають бути мінімальними. Чим більше таких точок контакту, тим більше часу доведеться витратити керівнику проєкту на забезпечення координації між елементами WBS [04].

Знайти та зазначити невиявлену або непризначену роботу, так звані «проміжки», яку потрібно виконати. Навіть у найбільш рутинному проєкті є такі проміжки, багато в чому через унікальну природу проєктів. Зі збільшенням розміру, складності, інноваційної технології, терміновості та інших характеристик, які сприяють унікальності проєкту, зростає ймовірність, масштаб і труднощі у визначенні змісту цих проміжків. Керівник проєкту повинен бути в першу чергу стурбований проміжками на наступному рівні нижче, певну увагу слід приділяти таким проміжкам на кожному рівні WBS, щоб, виявити нерозкритий робочий вміст проєкту. Тому використовуючи програмне забезпечення Jira, де детально фіксуються повний перелік задач, які чітко розподілені кожному члену команди та дати їх виконання.

Удосконалення структури розподілу робіт в IT-проєкті для розробки додатку "Ментальне здоров'я" може включати наступні зміни в розподілі задач між різними членами команди:

1. Аналіз та оцінка завдань: Перш за все, можуть бути внесені зміни в процес аналізу та оцінки завдань. Це може включати більш детальне визначення завдань, їх пріоритетів та обсягу. Також, можуть бути введені нові методи аналізу, які допоможуть краще розподіляти завдання між командою. Необхідно визначити технічні вимоги до програми. Це включає вибір платформи розробки, мови програмування, архітектури та інших технічних аспектів.

Для даного додатку, «Ментальне здоров'я», на телефон використовуючи платформу Android, що зазначено в статуті проєкту, було прийнято рішення використовувати краще мову програмування Java. Тому що найкращою мовою програмування для розробки додатків на телефон є

Java. Java є широко використовуваною мовою програмування для розробки мобільних додатків, оскільки вона є кросплатформеною. Це означає, що додатки, розроблені на Java, можуть працювати як на Android, так і IOS. Крім того, Java має велику кількість бібліотек і інструментів, що полегшують розробку додатків на телефон. Також обираючи, архітектуру для додатку було розглянуті деякі з них. Ось кілька популярних архітектур програмування, які використовуються для розробки додатків на телефон:

- Model-View-Controller (MVC): Ця архітектура розділяє додаток на три компоненти - модель, представлення та контролер. Модель відповідає за управління даними, представлення відображає інтерфейс користувача, а контролер обробляє взаємодію між моделлю та представленням;
- Model-View-ViewModel (MVVM): Ця архітектура також розділяє додаток на три компоненти - модель, представлення та модель представлення. Модель відповідає за управління даними, представлення відображає інтерфейс користувача, а модель представлення забезпечує зв'язок між моделлю та представленням;
- Clean Architecture: Ця архітектура базується на принципах чистого коду та розділяє додаток на рівні залежностей. Вона надає гнучкість та легкість утримання коду;

Удосконалюючи завдання проєкту, було прийнято рішення, що однією з найкращих архітектур програмування для розробки додатків на телефон є Model-View-Controller (MVC). Ця архітектура дозволяє розділити програмний код на три взаємодіючі складові - модель (Model), представлення (View) та контролер (Controller). Модель відповідає за обробку даних, представлення - за відображення інформації користувачу, а контролер - за керування потоком даних між моделлю і представленням.

2. Розподіл компетенцій: Залежно від складу команди та її навичок, можуть бути внесені зміни в розподіл компетенцій. Наприклад, якщо певний член команди має більше досвіду або знань у певній області,

йому можуть бути призначені більш складні або спеціалізовані завдання.

Провівши аналіз навичок та досвіду кожного члена команди, щоб визначити, хто найкраще підходить для виконання певних завдань. І визначивши, що в команді є розробник програмного забезпечення та інженер підтримки з великим досвідом роботи було прийнято рішення, що у рамках удосконаленого проєкту з розробки додатку "Ментальне здоров'я" до програмного забезпечення Android, програмістам можуть бути делеговані такі завдання:

- розробка та реалізація функціональності програми: програмісти можуть бути відповідальні за створення основних модулів, функцій та алгоритмів, необхідних для роботи програми;
- тестування та налагодження: програмісти можуть бути відповідальними за перевірку та виправлення помилок у коді програми, а також за тестування його працездатності та сумісності з різними операційними системами та пристроями;
- оптимізація та покращення продуктивності: програмісти можуть оптимізувати код програми, щоб він працював більш ефективно і швидко;
- інтеграція з іншими системами: програмісти можуть бути відповідальними за інтеграцію програми з іншими програмними системами або сервісами, щоб забезпечити ширший функціонал та зручність використання;
- забезпечення безпеки: програмісти можуть займатися захистом програми від злому та несанкціонованого доступу, а також забезпеченням конфіденційності даних користувачів;
- документація: програмісти можуть створювати документацію щодо використання та розробки програми, щоб полегшити його розуміння та підтримку.

Крім того, для удосконалення проєкту можна організувати навчання та тренінги для учасників команди, щоб підвищити їхню кваліфікацію та розширити кругозір. Це може допомогти знизити ризики та підвищити ефективність проєкту.

3. Комунікація та співпраця: важливим аспектом удосконалення структури розподілу робіт є комунікація та співпраця між членами команди.

Організувати регулярні зустрічі та обговорення - це один із головних аспектів для удосконалення управління командою в проєкті. Регулярне спілкування дозволяє своєчасно вирішувати проблеми, обговорювати завдання та узгоджувати плани. Також обов'язково потрібно застосовувати сучасні комунікаційні інструменти. Для спілкування та співробітництва в IT-проєкті можна використовувати такі інструменти, як Slack, Zoom, Trello та ін. Це допомагає спростити обмін інформацією та організувати роботу команди. Для даної команди краще обрати Slack, тому що саме Slack є найкращим інструментом для спілкування в команді із-за його широкого спектру функціоналу, який дозволяє легко обмінюватися повідомленнями, створювати канали для спільних проєктів, проводити відеодзвінки та зберігати документи, що сприяє ефективній комунікації між колегами.

Встановити ясні ролі та відповідальності. Кожен учасник проєкту має чітко розуміти свої обов'язки та знати, хто відповідає за ту чи іншу частину роботи. Це допомагає уникнути конфліктів та дублювання зусиль.

Активно використати процеси Agile-розробки. Agile-методології, такі як Scrum, допомагають забезпечити регулярну взаємодію між розробниками, тестувальниками та замовником, а також сприяють гнучкій адаптації до змін у проєкті.

Нова структура розподілу роботи може принести низку вигод для проєкту та його учасників:

1. Збільшення ефективності: поліпшення структури розподілу роботи може допомогти оптимізувати процеси та підвищити продуктивність

команди. Кожен учасник зможе зосередитись на своїх завданнях, що дозволить більш ефективно використовувати час та ресурси.

2. Поліпшення комунікації: нова структура розподілу роботи може сприяти більш ефективному обміну інформацією та комунікації між учасниками проєкту. Чітке визначення ролей та відповідальності допоможе уникнути непорозумінь та покращить взаємодію всередині команди.

3. Розширення навичок: нова структура розподілу роботи може надати учасникам проєкту можливість розвивати свої навички у різних галузях. Різноманітність завдань та відповідальності допоможе учасникам розширити свій досвід та компетенції.

4. Збільшення мотивації: чіткий розподіл роботи та визначення ролей дозволяють учасникам проєкту почуватися більш впевнено та відповідально за свої завдання. Це може підвищити їхню мотивацію та залученість до проєкту.

5. Гнучкіша адаптація: допоможе проєкту краще адаптуватися до змін.

Усі ці аспекти проєктів вимагають значної уваги з боку керівника проєкту, по-перше, щоб мінімізувати потенційні проблеми, а по-друге, щоб визначити їх на ранніх стадіях їх розробки. Керівник проєкту повинен розробити систему планування та контролю таким чином, щоб змусити визнати проблеми цього типу до того, як вони завдадуть шкоди проєкту. Управління шляхом обходу є одним із найефективніших способів виявити ці потенційні проблеми. Найефективнішим підходом до мінімізації цих проблем є розробка WBS таким чином, щоб мінімізувати їхній потенціал. У цьому полягає суть стратегічного планування проєкту.

Проаналізувавши та задокументувавши плюси та мінуси кожного, враховуючи наведене вище обговорення. Виконання цього аналізу чітко, у такий дисциплінований спосіб мінімізує природну тенденцію покладатися на успішний минулий досвід або особисті упередження та забезпечить адекватне врахування унікальних характеристик проєкту. Винятком є випадки, коли проєкт дуже схожий на минулий проєкт. У такому випадку

можна використовувати попередній WBS. Однак доцільно переглянути аналіз, який було виконано на попередньому проєкті, щоб переконатися, що ті самі критерії переважають у новому проєкті та що зроблені тоді судження залишаються застосовними.

Хоча WBS є важливим елементом для забезпечення стабільності проєкту, слід визнати, що проєкти використовуються для управління змінами. Зміни в проєкті зазвичай неминучі, оскільки навчання відбувається протягом життя проєкту. При розробці WBS слід звернути увагу на те, як змінені вимоги та зміст роботи будуть включені в затверджену WBS. Важливо контролювати зміни. Аномалія проєктів полягає в тому, що, хоча вони є засобом впровадження змін, самі зміни створюють багато проблем у проєктах.

Але також слід пам'ятати, що всі проєкти залежать від кошторису. Необхідно оцінити час, який знадобиться для виконання діяльності, і потрібно оцінити витрати, які будуть понесені під час виконання діяльності, етапу чи проєкту. Оцінки є наближеними або «найкращими припущеннями» і залежать від невизначеності, з якою стикаєшся, роблячи ці оцінки. З цього видно, що чим більше невизначеності, тим менш точним є припущення, а отже, тим вищий ризик того, що оцінки часу та витрат будуть нереалістичними. Якщо покладатися лише на одну оціночну цифру, наприклад, «для завершення діяльності знадобиться 8 днів», не брати до уваги невизначеність. Коротше кажучи, PERT — це техніка, яка допомагає враховувати цю невизначеність під час складання реалістичних оцінок проєктів.

Метод PERT — це аналітичний розрахунковий метод, що дозволяє спрогнозувати найоптимістичніші, найпесимістичніші та найвірогідніші терміни виконання робіт (у ході аналізу будується середньозважена оцінка), виключає при цьому повторення одних і тих же робіт в один і той же час. Він не допускає опису робіт з невідомою кількістю ітерацій, але може враховувати невизначені величини для підрахунку вірогідності виконання

як окремих завдань, так і всього проєкту у відведені терміни. Для кожного зі сценаріїв задається своя оцінка тривалості виконання робіт[16].

PERT — це триточкова методика оцінки діяльності, яка враховує невизначеність оцінки та ризик за допомогою трьох оцінок для визначення приблизної ймовірності вартості або тривалості діяльності(16). Використовуються три оцінки:

1. Швидше за все (M) — вартість/тривалість діяльності на основі реалістичної оцінки зусиль для необхідної роботи та будь-яких прогнозованих витрат.
2. Оптимістичний (O) — вартість/тривалість діяльності на основі аналізу найкращого сценарію для діяльності.
3. Песимістичний (P) — вартість/тривалість діяльності на основі аналізу найгіршого сценарію діяльності.

Припустімо, що є наступні три оцінки для діяльності А. Оптимістично можна завершити діяльність за п'ять (5) днів. За песимістичними оцінками, це може зайняти до п'ятнадцяти (15) днів, тоді як, швидше за все, це займе лише сім (7) днів.

PERT обчислює середньозважену величину як оцінку PERT за допомогою формули: $Pert\ Estimate = (оптимістичний + (4 \times найбільш\ вірогідний) + песимістичний) / 6$. Це означає, що отримано найвірогіднішу оцінку коефіцієнтом чотири (4), а потім визначено середнє значення зваженого найімовірнішого часу, найкращого та найгіршого сценаріїв. У випадку дії А це буде $(5 + (4 \times 7) + 15) / 6$. Таким чином, оцінка PERT розраховується як 8 днів. Якщо порівняти оцінку PERT у 8 днів зі звичайним середнім значенням трьох оцінок у 9 днів $(5 + 7 + 15) / 3 = 9$, зрозуміло, що оцінка PERT зважена до найбільш ймовірної оцінки. Використавши обчислення зваженого середнього значення, оскільки статистично виявили, що найбільша частина випадкової сукупності даних буде знайдена близькою до середнього значення, як можна побачити з наступної

дзвоноподібної кривої, яку також називають кривою нормального розподілу:

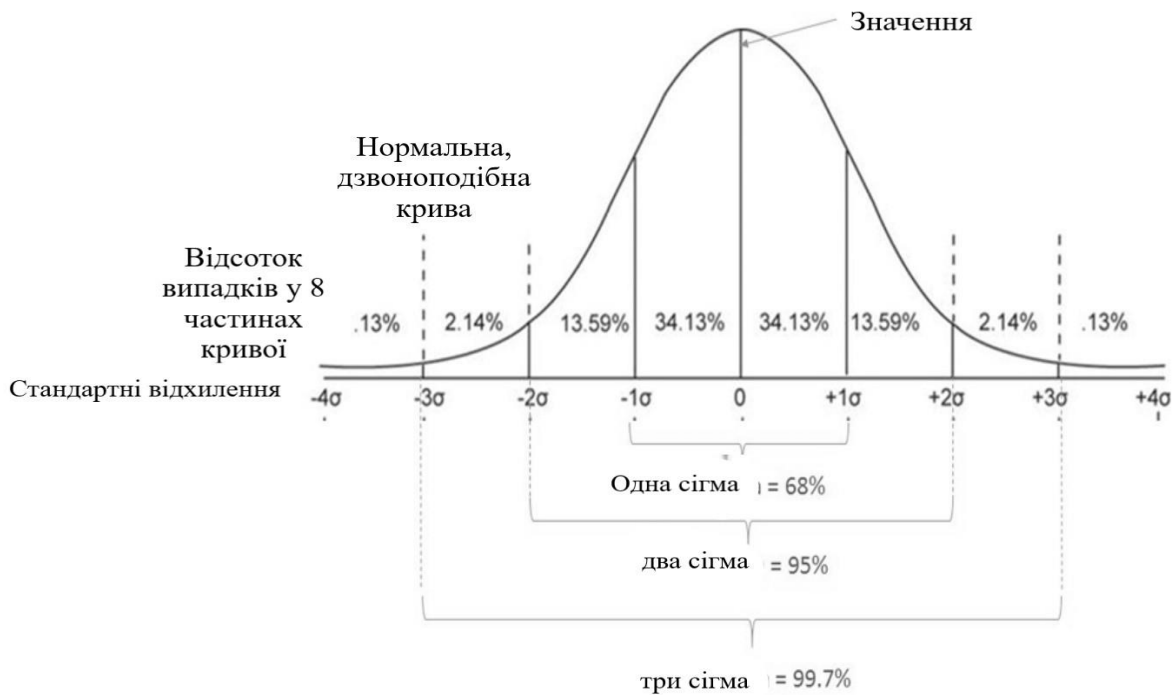


Рис3.3. крива нормального розподілу.

Дзвоноподібна крива показує, що якщо розділити дані на 6 рівних частин, тоді більшість випадків буде знайдено у 2 частинах поруч із середньою лінією та по обидва боки від неї (середнє значення), а ймовірність появи потрапляння в оптимістичний або песимістичний блок набагато нижчий. Розділяючи графік на 6 рівних блоків, обчислюючи стандартне відхилення, яке також називається сигмою (позначається символом сигми «σ»). Використовуючи просту формулу $\text{Sigma} = (\text{песимістичний} - \text{оптимістичний}) / 6$, ми ділимо графік на 6 рівних блоків. З цього видно, що формула оцінки PERT із використанням $(\text{оптимістичний} + (4 \times \text{найбільш вірогідний}) + \text{песимістичний}) / 6$ базується на цьому природному розподілі. Це також дає інструмент, за допомогою якого можна кількісно визначити ймовірність того, як дані будуть розподілені. Статистично відомо, що: 68% даних буде знайдено в межах одного

стандартного відхилення від середнього, іншими словами, між середнім значенням мінус одне стандартне відхилення та середнім плюс одне стандартне відхилення. Можна також сказати, що ймовірність того, що діяльність буде виконана в межах одного стандартного відхилення від середнього, становить 68%.

Також видно, що 95% даних буде знайдено в межах двох стандартних відхилень або, іншими словами, можна сказати з 95% упевненістю, що діяльність буде завершена в межах двох сигм.

При трьох сигмах є ймовірність 99,7%. Саме таку техніку розрахунку часу до структури розподілу роботи застосовано в даному проєкті та зазначено в додатку 5.

Після розрахунку часу, який буде використано для даного проєкту, слід також внести бюджет до WBS, який буде необхідно мати, щоб розробити додаток «Ментальне здоров'я».

Бюджет проєкту — це фінансовий документ, у якому вказано, що планується витратити на проєкт. Він покриває всі витрати, необхідні для реалізації проєкту.

Створення бюджету проєкту передбачає [17]:

- здатність ідентифікувати всі предмети, які будуть коштувати грошей;
- створення повної картини того, що потрібно витратити;
- отримати схвалення на цю суму.

Існує кілька способів класифікації вартості проєкту для бюджету проєкту:

1. Прямі витрати.
2. Непрямі витрати.
3. Капітальні витрати.
4. Експлуатаційні витрати.
5. Вартість виконання проєкту.

Витрати на управління проєктом більше використовувати прямі витрати. Прямі витрати охоплюють конкретні витрати, які пов'язані безпосередньо з розробкою, виконанням і підтримкою проєкту. Вони дозволяють більш

точно розрахувати витрати, що допомагає здійснювати контроль і ефективніше управляти бюджетом.

Прямі витрати включають в себе витрати, які безпосередньо пов'язані з розробкою додатку ментального здоров'я. Основні складові прямих витрат можуть включати наступне:

1. Заробітна плата розробників: це витрати, пов'язані з оплатою роботи програмістів, дизайнерів та інших фахівців, які працюють над розробкою додатків.
2. Витрати на програмне забезпечення: це включає витрати на придбання ліцензій на необхідне програмне забезпечення, таке як редактори коду, системи управління базами даних та інші інструменти розробки.
3. Витрати на апаратне забезпечення: це включає витрати на комп'ютери, сервери та іншу технічну інфраструктуру, необхідну для розробки та тестування додатків.
4. Витрати на тестування та якість: це включає витрати на проведення тестування додатків для перевірки їх функціональності та якості. Це може включати оплату тестувальників, придбання тестових пристроїв та програмного забезпечення для автоматизації.

Існують і непрямі витрати: це витрати, які не пов'язані безпосередньо з розробкою додатку, але необхідні для його успішного функціонування. Це можуть бути витрати на маркетинг, рекламу, підтримку користувачів, адміністрування та інші операційні витрати.

3.3. Впровадження розробленої моделі в соціальній сфері

Концепція та впровадження веб-додатків для підтримки ментального здоров'я набувають все більшої популярності в соціальній сфері. Ці додатки пропонують різноманітні функції, які спрямовані на підтримку психічного благополуччя та надання допомоги людям, які потребують психологічної або емоційної підтримки.

Впровадження моделі веб-додатку "Ментальне здоров'я" в соціальній сфері може включати такі аспекти:

1. Психологічна підтримка: додаток може надавати доступ до ресурсів, порад або онлайн-консультацій з психологами та терапевтами.
2. Трекінг емоцій та настрою: за допомогою додатку користувачі можуть відстежувати свої емоції, настрій, а також вести щоденник для управління стресом та тривогою.
3. Навчання навичкам саморегуляції: Веб-додаток може пропонувати вправи з медитації, релаксації, дихальні техніки або методику зниження стресу.
4. Соціальна підтримка: можливість спілкування з іншими користувачами, обмін досвідом та підтримкою через спеціальні форуми або чати.
5. Налагодження режиму сну та рекомендації щодо здорового способу життя: деякі додатки також надають поради з покращення сну та загального фізичного самопочуття.

Успішне впровадження моделі веб-додатку "Ментальне здоров'я" в соціальній сфері передбачає збалансований підхід до захисту даних користувачів, ретельне тестування та оцінку ефективності додатку, а також забезпечення доступності та легкості використання для різних категорій користувачів. Введення в суспільство моделі веб-додатку для підтримки ментального здоров'я може мати значний вплив на соціальну сферу. Нижче розглянуто можливі зміни у суспільстві при використанні даного додатку:

- *збільшення свідомості* – завдяки доступності цього веб-додатку більше людей можуть отримати інформацію про ментальне здоров'я, підвищуючи свою свідомість та розуміння щодо психічного благополуччя;
- *зменшення соціальної стигми* – використання такого додатку може сприяти усуненню стигми навколо психічних проблем, оскільки він допомагає відкрито говорити про ментальне здоров'я;

- *покращення доступності допомоги* – Веб-додаток може забезпечити доступ до психологічної підтримки та ресурсів у тих регіонах, де доступ до таких послуг обмежений;
- *сприяння самопізнанню та саморозвитку*: додаток може пропонувати інструменти для самодіагностики та самовдосконалення, що сприяє розвитку особистості та психічному самопізнанню.

За ініціативи О. Зеленської в Україні стартувала комунікаційна кампанія Всеукраїнської програми ментального здоров'я «Ти як?». Мета кампанії — навчити українців піклуватися про своє ментальне здоров'я та показати інструменти, які допоможуть подолати тривогу. «Ментальне здоров'я» дорівнює «сили для відновлення»[08].

По-перше, численні дослідження свідчать про значний вплив ментального здоров'я на продуктивність людей та громад і економіку в цілому: втрати від проблем пов'язаних з ментальним здоров'ям можуть сягати 4-5% ВВП. Тож коли ми говоримо про відновлення людини, що нерозривно пов'язано зі сферою ментального здоров'я, це має прямий вплив на відновлення економіки, відновлення країни в цілому.

По-друге, Україна відбудує нові школи, садочки, лікарні, багатоповерхівки. Але яке це матиме значення, якщо всередині цих будівель будуть поранені душі?! Тому говорити, промотувати культуру піклування про ментальне здоров'я, шукати дієві рішення, створювати послуги з психосоціальної підтримки – повсюдні й доступні для кожної людини, яка її потребує, треба вже сьогодні. Зволікання матиме ефект бомби уповільненої дії. «Детонації» не витримає ніяка медична система. Спрямовувати зусилля варто на плекання стійкості зараз, щоб не лікувати розлади потім.

Висновки до розділу 3. Порівнявши, дві моделі управління командою в IT- проєкті для розробки web-додатку програмного забезпечення на базі Agile, було чітко аргументовано, яка модель є кращою і чому. А також

удосконалено структуру розподілу роботи ІТ-проєкту для розробки додатку на прикладі додатку «Ментальне здоров'я» для подальшої більш продуктивної та злагодженої праці над проєктом. Розкрито концепцію та важливість таких додатків у соціальній сфері.

РОЗДІЛ 4. ОХОРОНА ПРАЦІ

Охорона праці – це система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних і лікувально-профілактичних заходів та засобів, спрямованих на збереження життя, здоров'я і працездатності людини у процесі трудової діяльності.

Головна мета охорони праці – створення на кожному робочому місці безпечних умов праці, безпечної експлуатації обладнання, зменшення або повна нейтралізація дії шкідливих і небезпечних виробничих факторів на організм людини і, як наслідок, зниження виробничого травматизму та професійних захворювань.

Сучасний розвиток технічного та технологічного стану виробництва передбачає постійну автоматизацію та оптимізацію виробничих процесів. Сьогодні, напевно, важко уявити компанію, господарська діяльність в якій здійснювалась би без використання комп'ютерної техніки. Через масовий характер робіт, що виконуються працівниками за допомогою комп'ютера, законодавством України чітко врегульовано норми та вимоги до використання комп'ютерної техніки на підприємстві, безпосередньо й охорона праці при роботі з комп'ютером.

4.1. Основні аспекти заходів з охорони праці користувачів ПК

Однією з характеристик розвитку сучасного суспільства є розширення сфери застосування інформаційних технологій у сфері людської діяльності. Персональні комп'ютери стали звичним явищем. Проте їх використання загострює проблеми збереження здоров'я людини та населення та потребує вдосконалення існуючих і розробки нових методів організації робочого місця та здійснення профілактичних заходів щодо запобігання негативним наслідкам впливу ПК на робоче місце. Заходи з охорони праці користувачів комп'ютерів враховують переважно соціальні, психологічні та медичні аспекти.

Соціальний аспект. З соціального боку вирішення цих проблем передбачає оптимізацію умов проживання, праці, відпочинку, харчування, способу життя, культурного розвитку, транспорту.

Психологічний аспект. Відіграє важливу роль у профілактиці захворювань. Належить до психології праці. Тому відповідні заходи щодо формування розумного виробничого колективу без психологічного

дискомфорту можуть допомогти зменшити надмірне нервово-психічне напруження та підвищити працездатність і ефективність. Психоемоційний стрес більшою чи меншою мірою проявляється в кожному з них і має особливе значення для користувачів відеотерміналів.

Медичний аспект. Велику роль у профілактиці захворювань серед користувачів комп'ютерів відіграє медицина. Існує низка профілактичних заходів для користувачів комп'ютерів, включаючи первинну профілактику здоров'я (професійні варіанти) та вторинну профілактику, спрямовану на зниження ймовірності перевтоми та надмірного стресу. Ці комплексні заходи спрямовані на відновлення функціонального стану органів зору та рухового апарату [20].

4.2. Нормативна база при роботі за комп'ютером

Перелік нормативно-правових актів, які регулюють це питання, досить широкий. Наприклад, ст. 21 Кодексу законів про працю України визначає обов'язки роботодавця щодо забезпечення працівникам комфортних та безпечних умов праці, а ст. 13 Закону України «Про охорону праці» закріплює це право з позиції охорони праці. Більшість нормативних актів — акти підзаконного рівня — правила, інструкції, державні санітарні правила і норми (ДСанПН) тощо, якими врегульовуються окремі моменти щодо власне конструкції комп'ютерної техніки, особливостей облаштування приміщень для роботи з нею та ряду інших вимог. Серед основних нормативів можна назвати [19]:

1. Наказ Держгірпромнагляду України «Про затвердження Правил охорони праці під час експлуатації електронно-обчислювальних машин» від 26 березня 2010 р. № 65
2. Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин ДСанПН 3.3.2.007-98, затверджені постановою Головного державного санітарного лікаря України від 10 грудня 1998 р. № 7.

3. Примірна інструкція з охорони праці під час експлуатації електронно-обчислювальних машин, затверджена наказом Міністерства доходів і зборів України від 5 вересня 2013 р. № 443.

4.3. Санітарно-гігієнічні норми і правила роботи за комп'ютером

Для забезпечення формування у здобувачів освіти відповідальної та безпечної поведінки в ситуаціях ризику, навичок здорового способу життя, запропоновано рекомендації щодо організації роботи за комп'ютером, без шкоди для здоров'я, під час дистанційного навчання [21].

Робота користувача ПК хоч і здається звичною та легкою справою, насправді може зашкодити здоров'ю, якщо занадто довго сидіти за монітором у одній позі і не давати очам відпочивати. Запобігти перевтомі можна, для цього потрібно:

- правильно облаштувати робоче місце;
- обмежити тривалість роботи за комп'ютером,
- робити гімнастику.

Вимоги до робочого місця. Робоче місце користувача ПК обладнується робочим столом і кріслом (рис. 4.1).

Висота робочого стола має бути в межах від 0,65 до 0,8 м, а ширина повинна забезпечувати можливість виконання операцій в зоні досяжності. Стіл повинен мати простір для ніг заввишки не менше ніж 600 мм, завширшки не менше ніж 500 мм, завглибшки (на рівні колін) не менше ніж 450 мм, на рівні простягнутої ноги – ніж 650 мм.

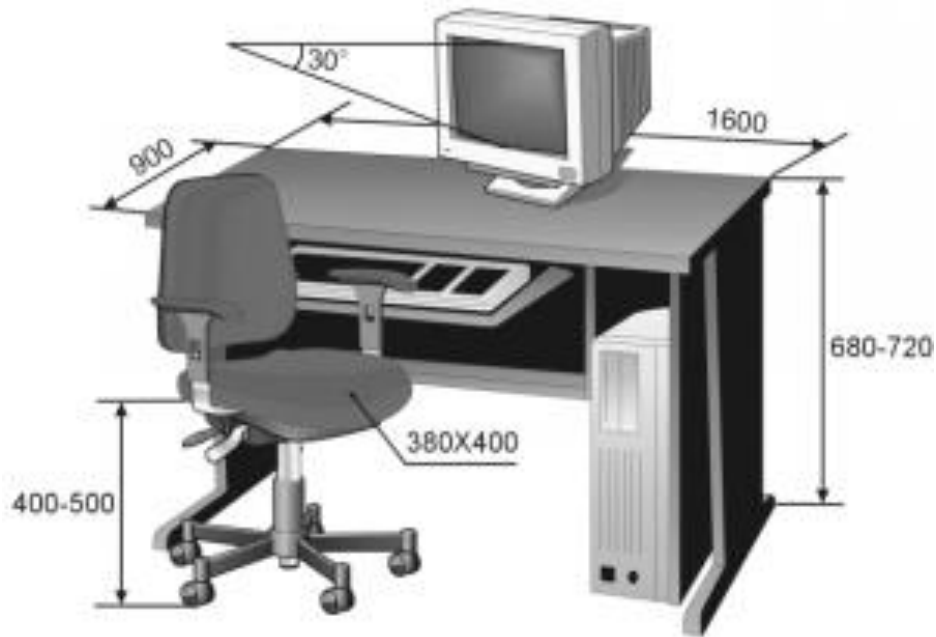


Рис. 4.1. Розташування робочого місця користувача ПК

Клавіатуру слід розташовувати на поверхні столу на відстані 200 мм від краю, звернутого до працюючого. У конструкції клавіатури має передбачатися опорний пристрій (виготовлений із матеріалу з високим коефіцієнтом тертя, що перешкоджає його переміщенню), який дає змогу змінювати кут нахилу поверхні клавіатури у межах 5...15°. Монітор необхідно встановлювати на відстані 60-100 см від вашого обличчя, і нижче рівня очей, з урахуванням розмірів екрану. Відстань від екрану до ока повинна складати:

- при розмірі екрану по діагоналі 35/38 см (14"/15")- 600 – 700 мм;
- 43 см (17") - - - - - 700 – 800 мм;
- 48 см (19") - - - - - 800 – 900 мм;
- 53 см (21") - - - - - 900 – 1000 мм.

Стіл необхідно розташовувати таким чином, щоб монітор був орієнтований боковою стороною до вікна, причому природне світло має бути переважно зліва.

Крісло повинно бути підйомно-поворотним, регульованим щодо висоти і кутів нахилу сидіння і спинки, а також відстані спинки від переднього краю сидіння, при цьому регулювання кожного параметра

крісла повинно бути незалежним, легко здійсненим і надійно фіксуватися. Висота поверхні сидіння має регулюватися в межах 400...500 мм, а ширина і глибина становити не менше ніж 400 мм. Кут нахилу сидіння – до 15° вперед і до 5° назад. Висота спинки стільця має становити (300±20) мм, ширина – не менше ніж 380 мм. Кут нахилу спинки має регулюватися в межах 1...30° від вертикального положення. Відстань від спинки до переднього краю сидіння має регулюватися в межах 260...400 мм.

Для зниження статичного напруження м'язів верхніх кінцівок слід використовувати стаціонарні або змінні підлокітники завдовжки не менше ніж 250 мм, завширшки 50...70 мм, що регулюються за висотою над сидінням у межах 230...260 мм і відстанню між підлокітниками в межах 350...500 мм. Поверхня сидіння і спинки стільця має бути напівм'якою з нековзким покриттям, що легко чиститься і не електризується.

Робоче місце користувача ПК має забезпечувати підтримання оптимальної робочої пози з такими ергономічними характеристиками:

- ступні ніг - на підлозі або на підставці для ніг;
- стегна - в горизонтальній площині;
- передпліччя - вертикально;
- лікті - під кутом 70-90° до вертикальної площини;
- зап'ястя зігнуті під кутом не більше 20° відносно горизонтальної площини;
- нахил голови - 15-20° відносно вертикальної площини.

Вимоги до освітлення. Як відомо, тривала робота за комп'ютером при недостатньому рівні освітленості може призвести до значного перенапруження зору, тому вимоги до освітлення є досить важливими.

Робоче місце необхідно розміщувати таким чином, щоб уникнути попадання прямого світла в очі. Вікна в приміщенні повинні бути обладнані регульованими пристроями: жалюзі, занавіски, тощо.

Для забезпечення захисту і досягнення нормованих рівнів комп'ютерних випромінювання необхідне застосування спеціальних окулярів для індивідуального захисту очей.

Приміщення, де розміщений комп'ютер повинно мати природне та штучне освітлення. Для штучного освітлення приміщення варто застосовувати переважно люмінесцентні лампи. Використання світильників без розсіювачів та екрануючих решіток не допускається.

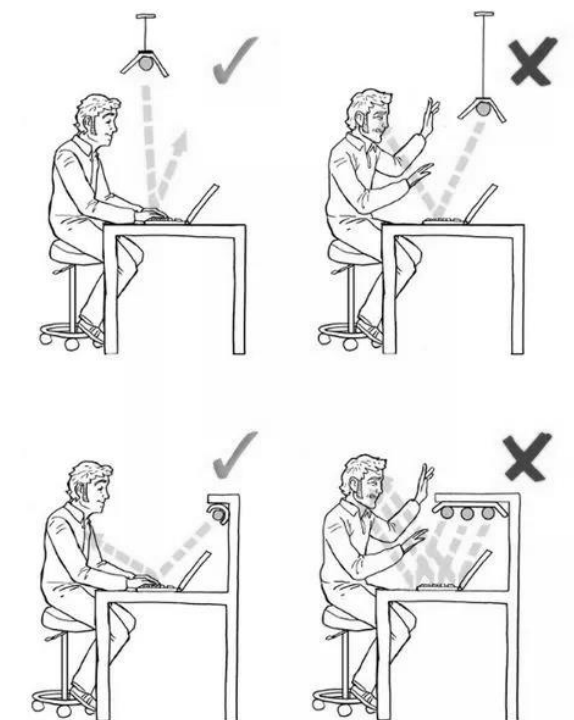


Рис. 4.2. Освітлення при роботі за комп'ютером

Світильники місцевого освітлення слід встановлювати таким чином, щоб не створювати відблисків на поверхні екрана, а освітленість екрана має не перевищувати 300 лк. Освітленість робочих поверхонь столів має становити не нижче 400 лк.

Мікроклімат. Приміщення, де розміщений комп'ютер повинно бути обладнано вентиляцією або кондиціонером для організованого повітрообміну. Допускаються параметри мікроклімату (температура — $19,5 \pm 0,5$ °C, відносна вологість повітря -60 ± 5 %, швидкість руху повітря не більше 0,1 м/с). Треба щоденно проводити вологе прибирання, тому підлогу

у кімнаті не слід накривати килимом. До і після роботи на комп'ютері слід протирати екран спеціальними серветками.

Режим праці з комп'ютером. Режим праці і відпочинку має передбачати додаткові нетривалі перерви в періоди, що передують появі об'єктивних і суб'єктивних ознак стомлення і зниження працездатності.

Безперервна робота перед екраном комп'ютера повинна бути не більше 50 - 60 хв. Через кожні 20–25 хвилин занять слід робити паузи для виконання вправ для очей, а через 50–60 хвилин роботи влаштовувати 10–15-хвилинну перерву з фізкультурними вправами. Під час перерви доцільно виконати спеціальний комплекс фізичних вправ чи просто походити, наприклад, у коридорі чи іншому приміщенні. Забороняється витрачати перерву на комп'ютерні ігри. Постійне сидіння та відсутність активного відпочинку викликає ряд хвороб – остеохондроз, захворювання дрібних суглобів, короткозорість, застій крові тощо. І щоб цього уникнути цих неприємностей, варто дотримуватися простих, але досить дієвих правил.

Висновки до розділу 4. У розділі «Охорона праці» було оглянуто основні аспекти заходів з охорони праці користувачів ПК, нормативна база при роботі за комп'ютером, та санітарно-гігієнічні норми і правила роботи за комп'ютером. Питання охорони праці складне і досить відповідальне. Адже за ним життя і здоров'я людей, які своєю працею створюють для держави та нації матеріальні блага.

РОЗДІЛ 5. ОХОРОНА НАВКОЛИШНЬОГО СЕРЕДОВИЩА

Охорона навколишнього середовища – є регулювання відносин у галузі охорони, використання і відтворення природних ресурсів, забезпечення екологічної безпеки, запобігання і ліквідації негативного впливу господарської та іншої діяльності на навколишнє природне середовище, збереження природних ресурсів, генетичного фонду живої природи, ландшафтів та інших природних комплексів, унікальних територій та природних об'єктів, пов'язаних з історико-культурною спадщиною.

Забрудненням навколишнього середовища це дії, які внесли в екологічну систему не властивих їй живих або неживих компонентів, фізичних або структурних змін, в результаті яких порушуються процеси круговороту і обміну речовин, а також відтоки енергії, унаслідок чого знижується продуктивність або руйнується дана екосистема.

Забруднюючі речовини зазвичай групуються по їх природі:

- фізичні забруднення, до них відносять: шумове забруднення і низькочастотна вібрація, електромагнітне забруднення, радіоактивні елементи;
- хімічні та біологічні забруднювачі, до них відносять: синтетичні органічні речовини, важкі метали, фтористі з'єднання;
- механічні, до них відносять: пил та тверді частки.

Для вирішення питань в галузі охорони навколишнього середовища займається наука – екологія. Екологія – це економіка природи й одночасне вивчення усіх взаємин живого з органічними і неорганічними компонентами середовища. Основними задачами екології є всебічна діагностика стану природи, розробка прогнозів по зміні стану природного середовища, формування профілактичних заходів щодо охорони навколишнього середовища.

5.1. Забрудненням навколишнього середовища

Під забрудненням навколишнього середовища слід розуміти зміну властивостей середовища (хімічних, механічних, фізичних, біологічних і пов'язаних з ними інформаційних), що відбуваються в результаті природних, або штучних процесів і призводять до погіршення функцій середовища по відношенню до будь-якого біологічного, або технологічного об'єкту. Використовуючи різні елементи навколишнього середовища у своїй діяльності, людина змінює її якість. Часто ці зміни виражаються в несприятливій формі забруднення. Сьогоднішній день а, тим більше завтрашній, важко представити без комп'ютерів, телевізорів, та іншої електронної техніки.

Інформаційні та телекомунікаційні технології, включивши в себе екологію в якості гуманних підвалин розвитку, перетворились на ідею інформаційного суспільства, стали способом життя людства, запорукою нового циклу розвитку цивілізації та планети.

Інформаційні технології сьогодні є екологічнішими за більшість інших видів активної людської діяльності, проте їх ще не можна назвати справді екологічними. Скажімо, ефективність інформаційних мереж напряму залежить від кількості користувачів, тобто, від кількості комп'ютерів, включених до мережі. Але для виготовлення одного звичайного персонального комп'ютера потрібно від 15 до 19 тон матеріалів. Це порівнювано з 25 тонами, потрібними для виготовлення автомобіля. На кожен функціонуючий комп'ютер (використовуваний в середньому протягом 4 років) припадає 1,5 комп'ютери вироблених. А близько третини комп'ютерів ніколи не буває продано взагалі – через швидкість, з якою вони втрачають технологічну актуальність. Це означає, що затрачувані ресурси справді наближаються до рівня автомобіля.

Електронні пристрої містять дуже токсичні з'єднання, які, потрапляючи в навколишнє середовище, створюють серйозну небезпеку для життя людей. Так, наприклад, 22% ртуті, що видобувається щороку в усьому світі, йде саме на потреби електронної промисловості і, зокрема, міститься

в мобільних телефонах. Кадмій, який є канцерогеном, використовується практично в усіх напівпровідникових пристроях. Свинець, особливо токсичний для нервової системи, міститься в акумуляторах і екранах моніторів. У міру розкладання захисних покриттів з електронних пристроїв у навколишнє середовище виділяється діоксин та інші високотоксичні з'єднання.

Стурбованість громадськості проблемами екології, а також нові, більш жорсткі закони по захисту навколишнього середовища змушують великих виробників устаткування створювати мережі по збору, що вийшли з обігу техніку і заводи з її утилізації. Крім того, в конструкції обладнання максимально збільшується частка матеріалів, придатних для переробки. Розміри мережі з утилізації "електронного брухту" залежать від регіону і місцевого законодавства.

Вся оргтехніка включає в свій склад як органічні складові (пластик різних видів, матеріали на основі полівінілхлориду, фенолформальдегіда), так і майже повний набір металів. Згідно з довідкових даних і на підставі лабораторних хімічних аналізів в таблиці 5.1 наведено усереднені дані про вміст різних металів і матеріалів у персональному комп'ютері.

Таблиця 5.1

Шкідливі речовини які містяться в ПК

Найменування	Дорогоцінні метали		Кольорові і чорні метали			Полімери і скло	
	Au, г	Ag, г	Al, кг	Cu, кг	Fe, кг	Пластик, кг	Скло, кг
Персональний комп'ютер (монітор, системний блок, клавіатура, маніпулятор)	0,053-0,072	0,8-1,1	0,1-0,4	0,1-0,2	3-4	3-3,5	10-20

Отже, звичайний комп'ютер містить як цінні метали, такі як золото, срібло, алюміній, мідь, так і небезпечні, такі як кадмій, свинець, цинк, нікель, а тому при списанні та утилізації обладнання керівнику необхідно керуватися і законодавством в області охорони навколишнього середовища.

5.2. Заходи щодо запобігання забруднення навколишнього середовища

Персональні комп'ютери, ноутбуки та інша інформаційна техніка, як відомо широко використовується в галузі наукових досліджень, промисловості, а також у повсякденному домашньому користуванні. Але будь-яка техніка стрімко застаріває, їй на зміну приходять нові, більш потужні, більш сучасні ПК та оргтехніка. Поступово виникає проблема, що робити зі старою технікою, морально застарілою або з тих чи інших причин, що вийшла з ладу, яка захащує підсобні приміщення та склади.

Утилізація оргтехніки та комп'ютерів це процес, який проводиться в кілька етапів. Найперше дію це списання обладнання безпосередньо з підприємства. Етап другий це розбір техніки і сортування отриманих матеріалів. Якщо деталі здатні служити вихідною сировиною, наприклад, кінескоп, деталі, в складі яких є дорогоцінні метали, то їх відправляють на очищення.

Як відомо до складу комп'ютера входить безліч металів таких як золото, срібло, алюміній, мідь та інших. Ще етапом по утилізації персональних комп'ютерів можна досягнути завдяки вторинній переробці. Сутність даного процесу полягає в тому, що можна витягти з цієї сировини частку корисних та рідких матеріалів таких як іридію, міді та інших. Цей процес відтворити набагато легше аніж наприклад видобути тонну міді, яка міститься в тисячотонних гірських породах.

Одне з нововведень для утилізації друкованих плат придумали Співробітники з Національної фізичної лабораторії Великобританії,

продемонстрували можливість спеціального розчину який розчинюють у гарячій воді. Дія якого зумовлює відшарування електронних компонентів. Таким чином 90% компонентів нових друкованих плат можна використовувати знову, тоді як у випадку звичайним методам - тільки 2%. Практично жодне підприємство не зможе самостійно утилізувати комп'ютери та оргтехніку, так як цей процес вимагає сучасного обладнання та специфічних знань. Тому довірити таку роботу можна тільки професіоналам, які мають великий досвід у даній сфері.

Проблема утилізації використаних комп'ютерів, периферійного обладнання, стає гострішою з кожним роком. Обсяги виробництва продуктів інформаційно-телекомунікаційних технологій та частота їх заміни на нові моделі примушують компанії замислюватись над проблемою біодеградації. Успіхи в цій галузі допоможуть, серед іншого, компаніям-виробникам зменшити податки, котрі вони сплачують зараз за утилізацію застарілих моделей. Останнє тим більше важливо, оскільки робить екологізацію економічно вигідною, тож спрямовує у цю сферу дедалі більше зусиль дослідників та довгострокових капіталовкладень. Таким чином, подальше поширення інформаційних технологій не збільшить, а навпаки – зменшить техногенне навантаження на довкілля.

Висновок до 5 розділу. В розділі з «Охорони навколишнього середовища» можна стверджувати, що вдосконалення сучасних інформаційних технологій, слід направляти не тільки для того щоб створювати людині максимально комфортні умови життя теперішнього часу. Але і для досягнення безвідходного процесу утилізації відпрацьованої техніки, не завдаючи шкоду навколишньому середовищу.

ВИСНОВКИ

В представлений кваліфікаційній роботі було виконано наступне:

- досліджені теоретичні аспекти в управлінні IT-проектами, а саме їх сутність, загальна характеристика та основні поняття методологій в управлінні проектами;
- виконаний детальний аналіз методів управління, інструментів та формування команди в IT-проектах;
- проведено дослідження процесу управління проектом розробки web-додатку програмного забезпечення та його удосконалення.

Результатом стала вдосконалена, на основі методології Agile, модель управління, яка враховує усі аспекти процесу розробки web-додатків програмного забезпечення, для подальшого її використання.

В проекті також представлена документація, яка була розроблена на основі конкретного прикладу, а саме web-додатку «Ментальне здоров'я» на базі Android. На даному прикладі було удосконалено модель управління командою, структуру розподілу роботи та впровадження розробленої моделі в соціальній сфері. Така отримана структурна модель управління проектами розробки web-додатку дозволяє підвищити якість кінцевого продукту та вдосконалити процес управління командою.

Визначені основні поняття, методи, проблеми та ризики при проектуванні web-додатку. Проаналізовані і делеговані різні задачі в процесі розробки web-додатку, делегування яких збільшує кількість одночасно виконуваних робіт та приводить до зменшення термінів виконання портфелю проекту.

Увагу було приділено охороні праці, де розкрито основні аспекти заходів з охорони праці користувачів ПК, нормативна база при роботі за комп'ютером та санітарно-гігієнічні норми і правила роботи за комп'ютером. Розглянуто заходи щодо запобігання забруднення навколишнього середовища при роботі за комп'ютером.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. PMBOK(Project Management Body of Knowledge.). (7 видання.2023.ст 370) Посібник.URL: <http://surl.li/oafaz>
2. Близнюкова І.О., Данченко О.Б., Тесленко П.О. Аналіз сучасних визначень ІТ-проектів. Близнюкова І.О., Данченко О.Б., Тесленко П.О. URL: https://library.krok.edu.ua/media/library/category/materiali-konferentsij/danchenco_0015.pdf
3. Консалтинг. Методології управління проектами, або Що таке Waterfall, Agile та Scrum. 27 січня 2021. URL: <https://devisu.ua/uk/stattia/metodologii-upravlinnya-proktami-abo-shcho-take-waterfall-agile-ta-scrum>
4. Рауткерсон Дж. Скотт. Спритний маніфест: розуміння спритних цінностей та принципів: підручник. 2012. URL: <https://uk.myservername.com/top-30-network-testing-tools>
5. Кен Швабер та Джефф Сазерленд. The Scrum Guide: посібник 2020. URL: <http://surl.li/oaffc>
6. Эми Кристофф. Kanban 101: Повний посібник з Kanban: посібник. 2018. URL: <https://businessmap.io/kanban-resources/getting-started/what-is-kanban>
7. Extreme Programming: офіційний веб-сайт. URL: <http://surl.li/oafhp>
8. Денисюк О. Г. Lean-менеджмент як технологія управління вітчизняними підприємствами в умовах кризи. Випуск № 46. URL: <file:///C:/Users/Main%20User/Downloads/2023>
9. Tom Hall. DevOps та Agile: стаття URL: <https://www.atlassian.com/ru/devops/what-is-devops/agile-vs-devops>
10. Худенко Дмитро. Управління проектами. Планування. Школа РМ. Worksection. 2023. URL: <https://worksection.com/ua/blog/best-project-planning-tools.html>

11. Zosym Махум. Матриця відповідальності (RACI Matrix). 2022. URL: <https://www.maxzosim.com/raci-matrix/>
12. Посібник із сукупності знань про управління проєктами (PMBOK), чернетка для ознайомлення (особливо розділ 5). Інститут управління проєктами, Верхній Дарбі, Пенсільванія. 1994.
13. Лаволд Гаррі. Розробка та використання структури розподілу робіт. Нью-Йорк: Van Nostrand Reinhold, 1983. стор. 302-323.
14. Ноздріна Л. В., Ящук В. І., Полотай О. І. Управління проєктами: Підручник / За заг. ред. Л. В. Ноздріної. К.: Центр учбової літератури, 2010. 432 с. URL: <http://surl.li/cgjmc>
15. Керцнер, Гарольд. Управління проєктами: системний підхід до планування, планування та контролю, 3-є ред. Нью-Йорк: Van Nostrand Reinhold. 1989. стор. 597-607.
16. Всеукраїнська програма ментального здоров'я «Ти як?». Портал gov.ua. URL: <http://surl.li/oafqo>
17. Слободян Фелікс. Охорона праці при роботі з комп'ютерною технікою: стаття. Журнал «Охорона праці і пожежна безпека». 2017. URL: <http://surl.li/oafra>
18. Пойда В.Ю. Курс лекцій з «Охорони праці в галузі». URL: <http://surl.li/oafu>
19. Санітарно-гігієнічні норми і правила роботи за комп'ютером. URL: <http://surl.li/oafry>
20. Чернов С.К., Дюкова С.П., Майстер І.В., Чубчик Т.Т. Методичні рекомендації з підготовки кваліфікаційної роботи зі спеціальності 122 «Комп'ютерні науки» освітньої програми «Управління проєктами». Миколаїв: НУК ім. адм. Макарова. 2023. 59 с.

Responsible	Project	Designer	Developer	QA	BA	PM	CTO	CEO	Client
Accountable	Initiation								
Consulted	Creare project management documentation					R		I	C
Informed	Create High level documentation			I	R	A			C
	Planning								
						R		I	C
	Create WBS					R			C
	Make Test Strategy			R		I			
	Plan Architecture						A		
	Create QA documentation			R		I			
	Plan UX	R			C	I			C
	Estimate tasks			C	C	C	A		C
	Execution								
	Develop Tasks		R						
	Develop and support Desing files	R							
	Support documentation			R		A			I
	Support processes					R			
	Make a Demo		R			A			I
	Monitor and Control								
	EVM					R			
	Weekly status report					A			I
	PM documentation support					R			
	Check the quality			R		A			
	Review code						A		I
	Closing								
	Deliver code						A		I
	Create closing documentation					R			I

Project	Name	Project Manager	Active	Position	Role on the project	Expectations	Power	Current Interest	Planned Interest	Strategy to manage	Actions
Project 1	Mental health app	Alesya Apolonova	Yes	Software developers	створення високоякісного коду, що відповідає вимогам проєкту. активна участь у процесі проєктування та обговорення архітектури програмного забезпечення	Якісна документація Доступність даних Зрозуміння цільової аудиторії Забезпечення стабільної та масштабованої технічної інфраструктури	High	Medium	Medium	Тестування і зворотній зв'язок Стеження за аналітикою	описати чітку мету та значущість проєкту Забезпечення чіткості завдань Забезпечити комфортні умови для роботи розробників, дозволяючи їм мати можливість висловлювати свої ідеї та пропозиції
Project 1	Mental health app	Alesya Apolonova	Yes	User interface designers	створення та оптимізації користувацького інтерфейсу (UI) для різних цифрових продуктів	Розуміння потреб, бажань, проблем і вимог цільової аудиторії є важливим кроком. Розуміння основних психологічних принципів, таких як кольори, шрифти, композиція, мова та інші елементи Постійне тестування дизайну додатку Взаємодія із користувачем	Medium	Medium	Medium	Розробка прототипів та тестування Тестування використання та забезпечення конфіденційності даних	Розробити докладний план роботи, включаючи терміни, завдання і відповідальність за кожну стадію розробки інтерфейсу.
Project 1	Mental health app	Alesya Apolonova	Yes	Support engineers	підтримка та обслуговування програмного забезпечення або технічних систем. усунення помилок, забезпечення підтримки клієнтів, а	Розуміння технічних аспектів розробки програмного забезпечення, таких як програмування, бази даних, розробка інтерфейсу користувача та	Low	Medium	Medium	Збір вимог Комунікація із замовниками Тестування та зворотній зв'язок	Встановлення регулярних зборів або спільних форумів для обговорення проблем, з якими стикаються підтримуючі інженери, а також для обміну найкращими практиками.

					також консультування з технічних питань. забезпечення безперебійної роботи системи та задоволення потреб користувачів.	технології мобільних додатків.					
Project 1	Mental health app	Alesya Apolonova	Yes	Hardware manufacturers	відповідальний за проектування та розробку апаратних компонентів та пристроїв.	Комп'ютери Мобільні пристрої для тестування Хмарні послуги Тестувальне обладнання Засоби для віддаленого спілкування Засоби для збору даних	Low	Low	Low	Аналіз ринку та обрання відповідного обладнання Встановлення партнерства та співпраця з виробниками	Дослідження ринку апаратного забезпечення для психічного здоров'я для визначення найбільш популярних виробників і розуміння того, які пристрої або технології найбільш потрібні для цього проєкту.
Project 1	Mental health app	Alesya Apolonova	Yes	Testers	перевірка програмного забезпечення на відповідність вимогам, виявлення та документування помилок, а також взаємодія з розробниками для виправлення виявлених проблем. Тестувальники також відповідають за створення тестових планів та забезпечення високої якості продукту перед його випуском на ринок.	Розвиток навичок тестування Розуміння психологічних аспектів Співпраця з експертами психології	Medium	High	High	Чітке визначення завдань і очікувань Надання чіткої інформації про продукт Ефективне спілкування	Регулярне звітування про стан тестування та виявлення проблем для швидкого їх вирішення.
Project 1	Mental health app	Alesya Apolonova	Yes	Marketing department	зазвичай полягає у плануванні, розробці та реалізації маркетингових	Покращення репутації компанії Можливості монетизації Розширення аудиторії	High	High	High	Вивчення ринку та аудиторії Розробка маркетингової стратегії Співпраця	Забезпечення ефективної комунікації між проєктним управлінням та відділом маркетингу для

					стратегій і тактик для просування продукту чи послуги.					з маркетологами та PR-фахівцями	забезпечення взаєморозуміння та вироблення спільного плану дій
Project 1	Mental health app	Alesya Apolonova	Yes	СТО	керування технічними аспектами проєкту, розробку стратегії технологічного розвитку, прийняття технічних рішень, а також управління технічним персоналом.	Інноваційний імідж Розширення аудиторії Збільшення прибутку Залучення талановитих фахівців Бізнес-розширення	High	High	High	Розуміння ринку та аудиторії Визначення функціональності	Запропонувати СТО брати участь у ключових прийняттях рішень, пов'язаних з технічними аспектами проєкту. Показати його важливість у процесі прийняття рішень та враховувати його експертну думку.
	Mental health app	Alesya Apolonova		HR	Рекрутинг та підбір персоналу для проєкту. Організація навчання та розвитку співробітників.	Збільшення продуктивності Підвищення репутації компанії Збереження та привабливості талантів	High	High	High	Розуміння вимог проєкту Формування команди проєкту Управління ресурсами	Забезпечення постійного отримання і врахування зворотного зв'язку від HR для постійного вдосконалення процесів та залучення їх у діяльність проєкту.

Communication type & activities	Description & Goal	Frequency	Local Time	Client time	Format	Participants	Owner (Responsible)
Internal Communications							
планування спринтів	обговорення дедлайнів обсягів та поставлених задач	кожен день	на протязі дня	-	Jira	PM	
наради онлайн та офлайн зустрічі	підведення підсумків виявлення недоліків та їх вирішення	2 рази на тиждень	10.00	—	meet та офіс компанії Jira	розробники дизайнери PM	PM
дзвінки звіти	короткі звіти та обговорення наступних кроків роботи на проєкті	кожен день	11.30	—	корпоративна пошта та WhatsApp	тестувальники аналітики PM	PM
чат переписка	комунікація з командою Вирішення поточних питань та виконання нових завдань	3 рази на тиждень	9.00+17.00	—	Slack	команда проєкту PM	PM
External Communications							
дзвінки із замовником	обговорення побажань щодо роботи над проєктом та уточнення вимог до команди	1 раз у 2 неділі	10.00	—	meet та офіс компанії Jira	технічний директор та розробники	PM
наради замовника з розробником	звітування щодо проведеної роботи та результату на кожному етапі виконання задач	1 раз на місяць	14.00	—	meet та офіс компанії Jira	Team liders	PM

PROJECT CHARTER	
<i>This template can be modified to meet a project's specific needs. See sample content in gray. To use: (1). Download template (2). Delete examples (3). Input specific project information.</i>	
Project Title: Mental health app	Date Prepared: 01.10.2023.
Project Sponsor: Natalia Rvachova	Project Customer: Insurance Company
Project Manager: Alesya Apolonova.	
Project Purpose:	
Creating an application on Android version 10,11,12.	
Estimated to be ready by February 2024.	
High-Level Project Description:	
During the implementation of the project, the development of the application, its design and testing will be carried out.	
A mobile app that will provide employees with resources and tools to support their mental and physical well-being, stress management techniques, fitness challenges and mindfulness exercises	
Project Boundaries: (in/out of scope)	
The application will be implemented entirely in Ukrainian and English.	
It will work on the Android version.	
A product will not be able to provide full medical advice or prescribe treatment.	
Addition will be based on algorithms and questionnaires, and although it will be able to provide useful advice and strategies for psychological well-being, it will not be able to replace face-to-face communication with a professional psychologist	
Privacy of former employee' data.	
Security of the storage of the health data and unauthorized access.	
Key Deliverables:	
1. The application can collect data about the user's stress level based on survey results and other parameters. 2. The application may provide a survey page where the user can answer questions about his state of health, mood and other factors affecting psychological state. 3. Psychological advice and recommendations: Based on the data collected as a result of the surveys, the application can provide the user with individual advice and recommendations to reduce stress and improve the psychological state. 4. Exercises for stress relief: The application may contain various exercises and techniques for stress relief, such as meditation, breathing exercises, relaxation techniques, etc. 5. Statistics and Analytics: The Application may analyze data and statistics about the user's stress level over time and provide reports on the improvement or deterioration of the psychological state over time.	
Overall Project Risk:	
1. Lack of highly qualified specialists (developers, UI/UX designers, software, AQA engineers. 2. Changing the duration of the project 3. Change of the project budget 4. Change of the scope of the project 5. Launch delays are possible due to technical issues	
Project Objectives	Success Criteria

Scope:	
Improved performance of employees.	70%
Time:	
app Launch	December 2023
Cost:	
20000 EUROS	Actual cost within 110% of the budget.
Other:	
-	-
Summary Milestones	Due Date
1: Defining the main idea of the product, its goals and potential users.	05.10.2023
2: Creating a prototype, choosing technologies and system architecture.	01/01/2024
3: Development of core functionality that makes the product unique and useful for users.	15/01/2024
4: Testing and debugging	15/02/2024
5: Customer approval	01/03/2024

Додаток 5

#	Stages/Tasks	Estimations (hours)			PERT (hours)	USD/hour	Resources (staff)	Labor hours
		optimistic	realistic	pessimistic				
1	Ініціація проекту				67			4130
1.1	Розробка РС				20			600
1.1.1	робота зі стейкхолдерами	8	12	15	12	30	1	360
1.1.2	визначення цілей; обмежень та бюджету проекту	4	8	12	8	30	1	240
1.2	Розробка планової документації				41			3280
1.2.1	Підготовка статуту проекту	2	6	15	7	40	2	560
1.2.2	Піготовка плану проекту	24	32	48	34	40	2	2720
1.3	Нарада з командою				6			250
1.3.1	залучення додаткових спеціалістів-психологів	1	2	3	2	50	1	100
1.3.2	залучення дієтологів	1	2	3	2	30	1	60
1.3.3	консультації фітнес інструкторів	1	2	3	2	45	1	90
2.	Планування				19			1010
2.1	збір вимог				19			1010
2.1.1	формування списку вимог та продукту	3	6	8	6	20	3	360
2.1.2	пріоритизація вимог	2	4	12	5	25	2	250
2.1.3	графік виконаних робіт	4	6	16	8	50	1	400
2.2	План розробки							
2.3	Ризики маркетингового плану	1	2	3	2			
2.4	Оцінка часу та бюджету	5	7	9	7			
3.	Розробка				127			7310
3.1	Дизайн				15			2160
3.1.1	розробка інтерфейсу	1,5	2,5	3	3	30	2	180
3.1.2	розробка кнопок	1,5	2,5	3	3	30	2	180
3.1.3	розробка списку функцій	1,5	2,5	3	3	100	2	600
3.1.4	розробка стрічки додатку	1,5	2,5	3	3	120	2	720
3.1.5	SRS	1,5	2,5	3	3	80	2	480
3.2	Front-end				52			1610
3.2.1	модуль авторизації	4	7	15	8	20	1	160

3.2.2	модуль вводу параметричних даних	5	9	18	10	30	1	300
3.2.3	модуль ануєтування психологічного стану	5	8	13	9	10	1	90
3.2.4	модуль розрахунку результату даних	7	9	16	10	20	1	200
3.2.5	модуль рекомендацій текстового та відео	3	7	12	8	20	1	160
3.2.6	модуль контролю здоров'я	3	6	13	7	100	1	700
3.3	Back-end				39			1440
3.3.1	модуль авторизації	3	6	12	7	20	1	140
3.3.2	модуль вводу параметричних даних	4	6	12	7	30	1	210
3.3.3	модуль анкетування психологічного стану	5	8	13	9	10	1	90
3.3.4	модуль розрахунку результату даних	5	8	16	9	20	1	180
3.3.5	модуль рекомендацій текстового та відео	3	5	12	6	20	1	120
3.3.6	модуль контролю здоров'я	3	6	13	7	100	1	700
3.4	Testing				21			2100
3.4.1	тестування Front-end частини	4	6	8	6	50	2	600
3.4.2	тестування Back-end частини	4	7	9	7	50	2	700
3.4.3	звіт перевірки випробування	4	6	16	8	50	2	800
4.	Документація користувача							
5.	Закриття				18			3600
	готовий реліз продукту	8	12	24	14	100	2	2800
	закриття відносин з клієнтом	2	4	6	4	100	2	800
Total Labor hours, USD								9300
6.	Hardware							17650
6.2	Мобільні пристрої, такі як смартфони або планшети, для тестування та розробки мобільних додатків.					800	7	5600
6.3	Серверне обладнання для зберігання даних та запуску серверних додатків					200	1	200
6.4	Спеціалізоване обладнання для вимірювання показників фізіологічного стану користувачів					300	1	300
6.1	аренда комп'ютерів або ноутбуків				231	50	1	11550
7.	Software							22900
7.1	Інтегроване середовище розробки -Visual Studio або Visual Studio Code					200	1	200

7.2	ліцензія на Java для Android					100	1	100
7.3	Системи керування базами даних MySQL					150	1	150
7.4	Інструменти для тестування додатків JUnit (для тестування Java)					300	1	300
7.5	Інструменти для версіонування коду та спільної роботи: GitHub					21750	1	21750
7.6	Інструменти для розробки та тестування інтерфейсу користувача: Figma				6	400	1	400
8.	Busness trip	3	5	8	6	3500	2	7000
Total cost, USD								
Total project cost, USD								56850