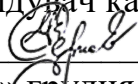
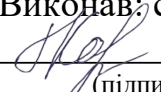


МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет кораблебудування імені адмірала Макарова
Навчально науковий інститут комп'ютерних наук та управління проектами
Кафедра управління проектами

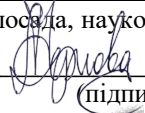
«Допущений до захисту»
Завідувач кафедри
 Чернов С.К.
«12» грудня 2022 р.

КВАЛІФІКАЦІЙНА РОБОТА
на здобуття ступеня вищої освіти «магістр»

на тему:
Формування команди розробки програмного забезпечення в Agile проектах

Виконав: студент 6171мз групи
 Казимір К.С.
(підпис)

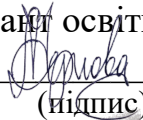
Керівник роботи:

Д.Т.Н., доцент
(посада, науковий ступень вчене звання)
 Чернова Л.С.
(підпис)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет кораблебудування імені адмірала Макарова

Навчально науковий інститут комп'ютерних наук та управління проектами

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

«ЗАТВЕРДЖУЮ»
Гарант освітньої програми
 Чернова Л.С.
(підпис)
«29» вересня 2022 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ
на здобуття ступеня вищої освіти «магістр»

Студенту **Казимір Катерині Сергіївні**
(Прізвище, ім'я, по батькові)

1. Тема роботи: Формування команди розробки програмного забезпечення в Agile проектах

Керівник роботи **д.т.н., доцент Чернова Людмила Сергіївна**

Затверджені наказом ректора № 806-уч. від «29» вересня 2022 року

2. Термін подання роботи: 12.12.2022р.

3. Вихідні дані по роботі: виконати дослідження з формування команди розробки програмного забезпечення в Agile проектах

4. Перелік питань, що належать до розробки (найменування розділів):
Розділ 1. Теоретичні засади формування команди проекту розробки програмного забезпечення

Розділ 2. Принципи та методології формування команди розробки програмного забезпечення

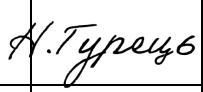
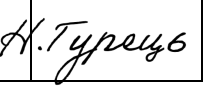
Розділ 3. Формування команди розробки програмного забезпечення в Agile проектах

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 05.09.2022 р.

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Вивчення літературних джерел з предмету дослідження	протягом навчання на 5 курсі	виконано
2	Складання розгорнутого плану магістерської роботи та ознайомлення керівника з планом кваліфікаційної роботи	Вересень 2022	виконано
3	Написання розділу 1	Вересень 2022	виконано
4	Написання розділу 2	Жовтень 2022	виконано
5	Написання розділу 3	Листопад 2022	виконано
6	Написання розділів з Охорони праці та навколишнього середовища (4,5)	Листопад 2022	виконано
7	Оформлення магістерської роботи	Грудень 2022	виконано
8	Передача магістерської роботи рецензенту для рецензування	Грудень 2022	виконано
9	Передача магістерської роботи науковому керівникові для написання відгуку	Грудень 2022	виконано
10	Попередній захист магістерської роботи	12.12.2022	виконано
11	Захист магістерської роботи	20.12.2022	виконано

Студент


(підпис)

Казимір К.С.

Керівник роботи


(підпис)

Чернова Л.С.

Введіть текст

АНОТАЦІЯ

Казимір К.С. «Формування команди розробки програмного забезпечення в Agile проектах». Магістерська робота за спеціальністю 122: Комп'ютерні науки, освітня програма «Управління проектами». Національний університет кораблебудування ім. адм. Макарова. Миколаїв, 2022. 77 сторінок.

В магістерській роботі розглянуті теоретичні аспекти та особливості формування команди проєкту, досліджено принципи та методології формування команди розробки програмного забезпечення. Розглянуті переваги та недоліки використання гнучкої методології Agile. Сформовано команду розробників програмного забезпечення в Agile проектах.

Ключові слова: програмне забезпечення, команда, формування, методологія управління проєктами, Agile.

ABSTRACT

Kazimir K.S. «Forming a team of software development in Agile projects» . Master's thesis on specialty 122: Computer science, educational program «Project management». National University of Shipbuilding named after Adm. Makarov. Mykolaiv, 2022. 77 pages.

The master's thesis examines the theoretical aspects and peculiarities of the formation of the project team, investigates the principles and methodologies of the formation of the software development team. Advantages and disadvantages of using Agile flexible methodology are considered. A team of software developers has been formed in Agile projects.

Keywords: software, team, formation, project management methodology, Agile.

ЗМІСТ

ВСТУП.....	6
РОЗДІЛ І.ТЕОРЕТИЧНІ ЗАСАДИ ФОРМУВАННЯ КОМАНДИ ПРОЕКТУ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	8
1.1. Проєкт з точки зору системного підходу.....	8
1.2.Поняття команди проєкту.....	10
1.3.Розподіл командних ролей та функцій.	16
РОЗДІЛ 2.ПРИНЦИПИ ТА МЕТОДОЛОГІЇ ФОРМУВАННЯ КОМАНДИ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	20
2.1. Методології розробки ПЗ	20
2.2. Особливості застосування методології Agile при розробці програмного забезпечення.....	26
2.3. Проблеми формування та управління командою розробників програмного забезпечення.	32
РОЗДІЛ 3.ФОРМУВАННЯ КОМАНДИ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ В AGAILE ПРОЕКТАХ.....	36
3.1 Формування та (управління) команди розробки програмного забезпечення.....	36
3.2 Практичне застосування удосконалених моделей формування команди	51
3.3 Аналіз апробації.	55
РОЗДІЛ.4. ОХОРОНА ПРАЦІ	57
4.1 Аналіз шкідливих та небезпечних факторів на робочому місці програміста.....	57
4.2 Система штучного освітлення приміщення.	63
4.3.Основні положення техніки безпеки під час експлуатації ЕОМ.....	64
РОЗДІЛ 5.....	67
ОХОРОНА НАВКОЛИШНЬОГО СЕРЕДОВИЩА	67
5.1.Загальні питання охорони навколишнього середовища	67
5.2. Вплив комп'ютеризації на наволишне середовище.	68
5.3. Переробка комп'ютерного обладнання	70
ВИСНОВОК	74
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ	75

ВСТУП

Актуальність теми дослідження. З розвитком науково-технічного прогресу зростає роль програмного забезпечення. Програмне забезпечення використовується для управління бізнесом як складова КІС, для управління технічними пристроями (автомобілями, літаками, поїздами), побутовими пристроями, в телекомунікаційній індустрії.

Програмна інженерія невблаганно рухається у бік нових методів управління проектами дедалі більшої складності. Технології програмування, процеси та методи швидко розвиваються, а індустрія розробки програмного забезпечення залишається процесом з інтенсивним використанням людської праці.

Вибрати метод управління ІТ-проектами - Kanban, Scrum або Agile - важливе завдання для будь-якого керівника. Але важливі і деталі: як правильно вибудувати комунікацію між співробітниками, особливо в умовах тотального видалення, які настали, які програми допоможуть організовувати наради, обмінюватися кодом або стежити за ходом виконання проекту, а також як зберігати мотивацію у розробників, що сидять вдома.

Мета дослідження. Підвищення ефективності розробки програмного забезпечення, шляхом удосконалення моделей формування команди проекту.

Завдання дослідження

- провести теоретичне дослідження моделей управління командою в проектах розробки програмного забезпечення;
- провести аналіз особливостей формування команди розробників ПЗ;
- провести підбір та обґрунтування методичних основ управління командою на базі розроблених моделей;
- розробити рекомендації щодо практичного використання запропонованих моделей формування команди розробки програмного забезпечення.

Об'єкт дослідження - проекти розробки програмного забезпечення.

Предмет дослідження є процес формування команди в проектах розробки програмного забезпечення.

Наукова новизна. Удосконалено моделі управління командою за принципами формування та розвитку команд, що дає змогу підвищити ефективність та якість розробки програмного забезпечення.

Методи дослідження. В якості методів наукового дослідження використано: системний аналіз, процесне моделювання, імітаційне моделювання, планування експерименту, порівняльний аналіз, методологію Agile.

Магістерська робота викладена на 77 сторінках, складається з п'яти розділів, вступу та висновків та переліку використаних джерел, який складається з _ джерела.

РОЗДІЛ I

ТЕОРЕТИЧНІ ЗАСАДИ ФОРМУВАННЯ КОМАНДИ ПРОЕКТУ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

1.1. Проєкт з точки зору системного підходу

Для управління будь-яким проєктом на період його здійснення створюється специфічна тимчасова команда проєкту, що відображає існуючу організаційну структуру управління проєктом, поділ функцій, обов'язків та відповідальність за рішення, що приймаються у процесі його реалізації. На верхньому рівні структури знаходиться менеджер проєкту, а на нижніх – виконавці, відділи та фахівці, які відповідають за окремі функціональні сфери.

За змістом команда проєкту є групу фахівців високої кваліфікації, які володіють знаннями та навичками, необхідними для ефективного досягнення цілей проєкту.

Основними інтегруючим фактором створення та діяльності команди виступає стратегічна мета реалізації проєкту. Команда проєкту виступає, як соціальний організм, який має свій початок, здійснює процес життєдіяльності (реалізація проєкту) та завершує своє існування розформуванням чи трансформацією в іншу управлінську команду. З одного боку, команда проєкту зумовлює створення певного організаційного середовища проєкту, формуючи цінності, принципи та норми поведінки персоналу. З іншого боку, діє в ній, підкоряючись єдиній меті та філософії управління проєктом.

Проєкт може розглядатися як деяке завдання з певними вихідними даними та необхідними результатами (цілями), визначальними способи її розв'язання. Проєкт містить у собі задум (проблему), кошти його реалізації (вирішення проблеми) та одержувані в процесі реалізації результати. З погляду системного підходу проєкт може розглядатися як процес переходу з вихідного стану в кінцевий результат при участі низки обмежень та механізмів (рисунок 1.1). У сучасному розуміння проєкти – те, що змінює наш світ: будівництво

об'єкту, програма науково-дослідних робіт, реконструкція підприємства, створення нової організації, розробка нової техніки та технології, розвиток регіону тощо.[2].

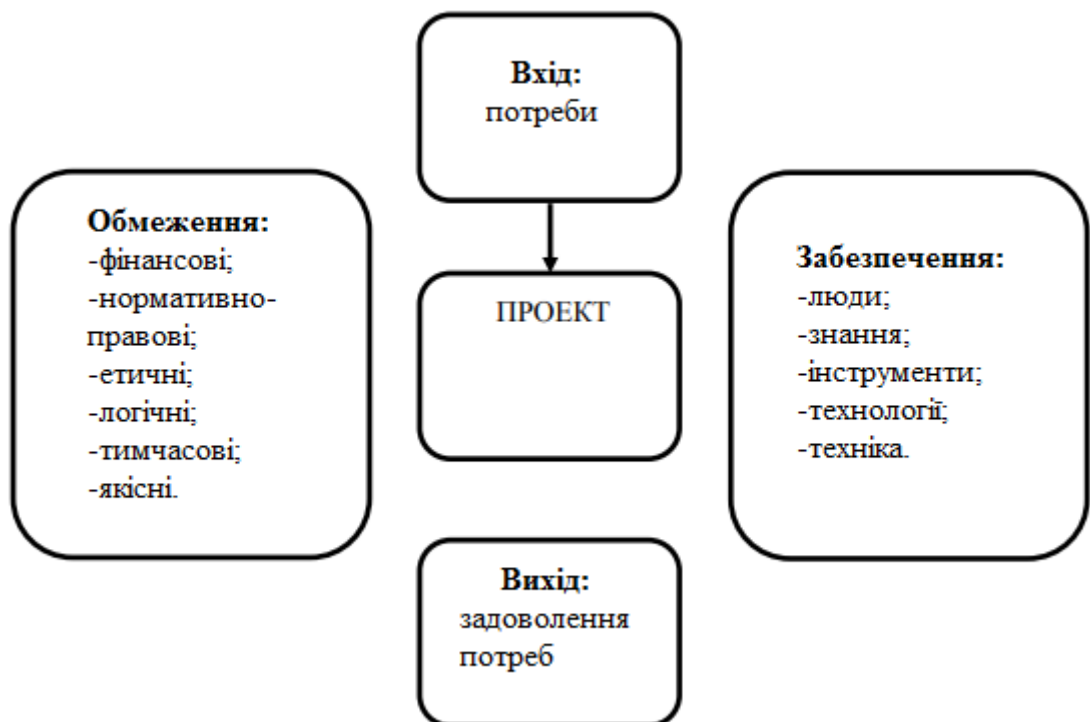


Рис.1.1 – Проєкт з точки зору системного підходу

При організації роботи над проектом необхідно вирішити дві основні завдання:

- формування команди проекту;
- організація ефективної роботи команди.

Залежно від специфіки, розміру та типу проекту у його реалізації можуть брати участь від однієї до кількох десятків (іноді сотень) організацій та окремих фахівців. У кожній їх свої функції, ступінь участі у проекті та міра відповідальності за його реалізацію.

Спеціалістів та організації, залежно від виконуваних ними функцій, прийнято об'єднувати у абсолютно конкретні групи (категорії) учасників проекту, до складу яких входять: замовники, інвестори, проектувальники, підрядники, консультанти, ліцензіари, фінансові інституції. Зрештою, існує

команда проекту, яку очолює керівник проекту – менеджером проекту, а також, залежно від специфіки, у проекті можуть бути та інші учасники. Слід зазначити, що учасники проекту – категорія ширша, ніж команда [2].

1.2.Поняття команди проекту.

Команда проекту – це група співробітників безпосередньо працюючих над здійсненням проекту та підлеглих керівнику останнього; основни елемент його структури, оскільки саме вона забезпечує реалізацію задуму. Ця група створюється на період реалізації проекту та після його завершення розпускається. Кількість людей у команді визначається обсягом робіт, передбачених проектом. Як правило, лідери функціонально та предметно орієнтованих груп фахівців та становлять команду управління проектом.

Лідери груп – це керівники, координатори зусиль усіх членів групи; члени групи – безпосередні виконавці.

Для команди проекту потрібна наявність у її членів комбінації взаємодоповнюючих навичок (компетенцій), що становлять три категорії:

- технічні та/або функціональні, тобто. професійні навички;
- навички щодо вирішення проблем та прийняття рішень;
- навички міжособистісного спілкування (прийняття ризику, корисна критика, активне слухання тощо).

Основними факторами, що визначають принципи формування команди, є: специфіка проекту, організаційно-культурне середовище та особливості взаємодії керівника коїться з іншими членами команди.

Команда проекту організується для його реалізації, тому така Характеристика як специфіка проекту – одна з головних в освіті команди. Специфіка визначає:

- формальну структуру команди, яка затверджується
- керівництвом;
- рольовий склад;
- перелік знань, умінь та навичок, якими мають володіти члени

команди;

- терміни, етапи, види робіт із проекту.

Очевидно, що склад команди для реалізації будівельного проекту повинен включати проектувальників, будівельників, постачальників тощо. А до складу команди наукового проекту повинні входити науковці, експерти, фахівці у галузях відповідних знань тощо.

Організаційно-культурне середовище команди проекту ділиться на зовнішнє та внутрішнє. Зовнішнє середовище включає оточення проекту у всіх аспектах. Внутрішнє середовище, чи організаційна культура самої команди, включає такі характеристики:

- прийняті та розділені всіма учасниками норми команди;
- способи розподілу влади;
- згуртованість та пов'язаність членів команди;
- організація рольового розподілу;
- характерні способи організації та протікання командного взаємодії (координації, комунікації, ухвалення рішень).

Особливості взаємодії лідера проекту з іншими членами команди визначають всю систему взаємовідносин керівника із підлеглими.

Ефективний проект-менеджер допомагає членам команди пройти всі стадії розвитку та повністю розкрити свій потенціал. Він повинен представляти і аналізувати можливі труднощі на шляху становлення команди, вміти передбачати настання чергового етапу розвитку та вести команду вперед. Сучасна концепція лідерства наголошує на такій його цінності, як підвищення у підлеглих здатності до самоорганізації та самоконтролю.

Найбільш адекватний лідер - той, хто може керувати іншими в такому напрямі, щоб вони керували собою [5].

Таким чином, при організації роботи над проектом необхідно вирішити дві головні завдання: сформувати команду проекту та організувати її ефективну роботу. Розрізняють чотири основні підходи до формування команди:

- цілеспрямований (заснований на цілях);
- міжособистісний;
- рольовий;
- проблемно-орієнтований.

Цілеспрямований підхід дозволяє членам команди краще орієнтуватися у процесах вибору та реалізації загальних групових цілей проекту.

Міжособистісний підхід сфокусований на покращенні міжособистісних відносин у команді і заснований на тому, що міжособистісна компетентність збільшує ефективність діяльності команди. Його мета – збільшення групової довіри, заохочення спільної підтримки, а також збільшення внутрішньоконандних комунікацій.

Рольовий підхід – проведення дискусії та переговорів серед членів команди щодо ролей; передбачається, що ролі членів команди частково перекриваються. Командна поведінка може бути змінена в наслідок зміни їх виконання, а також індивідуального сприйняття ролей.

Проблемно-орієнтований підхід передбачає організацію заздалегідь спланованих серій зустрічей із групою фахівців у рамках команди, мають спільні організаційні відносини та цілі. Підхід включає в себе послідовний розвиток процедур вирішення командних проблем і потім досягнення головного командного завдання.

Раціональне формування команди впливає на ефективність усієї наступної діяльності:

- покращуються керівництво та якість прийняття рішень;
- змінюється командна організаційна культура;
- утворюється розумна кооперація серед усіх членів команди [2].

На рис.1. 2 наведено модель формування ефективної команди проекту. У ході планування роботи команди проект-менеджер має чітко сформулювати цільові установки, забезпечити узгодженість у роботі всіх функціональних секторів команди для ефективного використання виділених на проект ресурсів. Для цього доцільно залучати членів команди до обговорення будь-

яких труднощів та перешкод на шляху до ефективної роботи, аналізувати та враховувати різні думки [4].

Поряд із заходами зі згуртування команди у високоефективний робітник колектив, проект-менеджер покликаний встановлювати і хороші робітники взаємини з вищими керівниками, забезпечуючи їх інформацією про хід робіт із проекту, створюючи сприятливий імідж проекту.

Основу концепції управління персоналом проекту в даний час складають зростаюча роль особистості працівника, знання його мотиваційних установок, вміння їх формувати та спрямовувати відповідно до завдань, що стоять перед організаційною структурою управління проектом.

Узагальнення вітчизняного та зарубіжного досвіду дозволяє сформулювати основні завдання системи управління персоналом у сучасних умовах:

- визначення загальної стратегії формування команди проекту;
- планування забезпечення проекту людськими ресурсами;
- залучення, відбір та оцінка персоналу;
- підвищення кваліфікації та перепідготовка персоналу команди проекту;
- система просування по службі (управління кар'єрою);
- управління заробітною платою та витратами на персонал [5].

Ефективне управління персоналом – це основа управління проектом. Система управління персоналом проекту включає методи, процедури, програми управління процесами, пов'язаними з людськими ресурсами та характеризується наступними параметрами:

- відповідність персоналу цілям та місії проекту;
- ефективність системи роботи з персоналом, вибір критеріїв оцінки результатів роботи з персоналом;
- надмірність або недостатність персоналу, розрахунок потреби, планування кількості та якості;
- збалансованість персоналу за певними групами професійної діяльності

та соціально-психологічних характеристик;

- структура інтересів та цінностей, що панують у групах персоналу управління, їх вплив на ставлення до праці та її результати;
- ритмічність та напруженість діяльності, що визначають психологічний стан та якість роботи;
- інтелектуальний та творчий потенціал персоналу управління, відбиває підбір та використання персоналу, організацію системи її розвитку.



Рис.1.2. Модель формування ефективної команди проекту

Ефективність управління персоналом визначається ступенем реалізації загальних цілей проекту. Ефективність використання кожного окремого члена команди залежить від його здатності виконувати необхідні функції та мотивації, з якою ці функції виконуються [3]. Під здатністю виконувати

відповідні функції розуміється наявність у спеціаліста необхідних для виконання своїх функцій професійних навичок, знань, досвіду, достатньої фізичної сили та витривалості, інтелектуальних можливостей загальної культури. Одним із принципів командної роботи виступає розподіл обов'язків та відповідальності за досягнення поставленої мети, а не жорстке закріплення виконуваних функцій. Для ефективної організації роботи команди необхідні:

- чіткий розподіл ролей та обов'язків;
- усвідомлення всіма членами команди цілей та поточних завдань проекту;
- облік і особистісних, і професійних якостей фахівців при поєднання їх у команду;
- увага менеджерів і до досягнення цілей проекту, і встановлення дружньої робочої атмосфери.

Також менеджеру не варто забувати і про те, що всі люди у проектній команді різні. Деякі вже мали досвід подібної роботи, а хтось, можливо, займається проектною діяльністю вперше. І, швидше за все, багато хто не був знайомий один з одним до того, як потрапив в одну команду. Не варто забувати і про те, що зустрічаються дуже складні проекти, як для здійснення, так і для розуміння. Тому для ефективної організації роботи команди над проектом менеджер повинен заздалегідь подбати про таку простий на перший погляд речі, як інструкція, тому що однією тільки мережевою діаграмою під час роботи над проектом може виявитися недостатньо.

Інструкції повинні бути якомога простішими та короткішими. Не варто намагатися укласти всі ідеї в один пункт. По можливості слід писати інструкції для кожного окремого етапу або виду робіт, щоб не наставала плутанина.

Для деяких проектів інструкції вам можуть не знадобитися, або їх має бути мінімальна кількість. Але для інших проектів може виявитись необхідним складання інструкції для кожного члена команди щодо кожної дії. Приймати рішення про написання інструкцій потрібно, виходячи зі складу

набраної вами команди, компетенції кожного її члена та числа фаз проекту, які мають бути виконані одночасно.

Доцільне формування команди впливає на ефективність усієї наступної діяльності:

- покращуються керівництво та якість прийняття рішень;
- змінюється командна організаційна культура;
- з'являються послідовність у відстоюванні своєї позиції та розумна кооперація серед усіх членів команди;
- рефлексивні моделі.

У колективах, які виконують програмні проекти, можливі самі різноманітні організаційні структури. Враховуючи, що розробка програмного забезпечення в більшості випадків може розглядатися як колективна праця фахівців, спрямована на задоволення потреби користувачів в автоматизації їх діяльності.

Інша колективна праця, вона вимагає координації та управління. Саме організація ефективної діяльності персоналу та є основною турботою управління персоналом організації загалом та кожного проекту окремо.

Найбільший ефект та якість системи управління персоналом досягаються в тому випадку, коли застосовується система методів у комплексі. Команда – це не просто соціальна група людей, передусім, засіб досягнення мети. Тому і формування команди має виходити із цілей організації, проекту та його стратегії.

1.3.Розподіл командних ролей та функцій.

Відкриті та чесні взаємини є важливою складовою процесу формування ефективної проектної команди. Підбір психологічно сумісних виконавців – інша ключова складова. Вкрай важливо, щоб менеджер проекту мав свободу дій під час виборів учасників команди.

Сьогодні все більшого значення набуває здатність фахівці працювати в команді, де тісно об'єднані дві складові: матеріальна та духовна. Матеріальну складову представляють люди, об'єднані в команду, духовну – ідеологія та психологія.

Ідеологія команди формується із сукупності ідей та поглядів, що відображають кінцеві цілі виконання трудової функції Психологія команди виражається в сукупності соціально-психологічних особливостей, що виявляються в ході її створення та розвитку, на основі взаємодії членів команди, форм, та способів взаємне задоволення потреб. У процесі роботи учасники мають спланувати спільну діяльність, організувати обмін інформацією, налагодити взаєморозуміння, виробити форми спільних дій. Це передбачає формування морально-психологічного клімату, спільного досвіду, громадської думки, а також вирішення питань лідерства, розуміння природи внутрішньогрупових конфліктів тощо. Велике значення при цьому мають особисті якості кожного та особистісні характеристики членів команди.

Коли люди працюють у складі однієї групи чи команди, кожен з них виконує ролі двох типів: функціональну, що базується на професійних навичках та практичному досвіді, та командну, основу якої лежать особистісні особливості.

Командну роль можна як характеристику якості застосування індивідуальних навичок та досвіду, що становлять зміст виконуваної функціональної ролі.

Розглянемо основні командні ролі з 3 класифікації та на прикладі дамо оцінку кожному учаснику відомої команди.

1 тип рольового розподілу:

«Професійна діяльність» Основні ролі тут цілком очевидні: "координатор" виконує організаторські функції в команді: підбирає членів команди, розподіляє завдання, організовує колективні обговорення, намічають терміни виконання завдань, контролює правильність виконання тощо;

«Головний спеціаліст» (2-3 чол.) – члени команди найбільш високої кваліфікації, що виконують ключові операції (процедури, роботи) технологічного циклу професійної діяльності;

«Напарник» - (1-2 чол.) Члени команди, що працюють у зв'язку з "головними фахівцями" під їх керівництвом; «завгосп-ремонтник» – член команди, відповідальний за спорядження, обладнання та інвентар команди; «агент із постачання» - член команди, найбільш успішно діючий із забезпечення команди всім необхідним.

2 тип рольового розподілу: «Неформальна командна взаємодія».

I тип ситуації: «Взаємодія із зовнішнім середовищем». Основні ролі: «аналітик» – найбільш врівноважений член команди системним складом мислення, «переговорник» - член команди, найбільш майстерний з усіх комунікації, "геній спілкування"; «скептик» – критикує пропоновані ідеї, має здатність до глибокого аналізу та послідовного логічного мислення, однак часом позбавленої уяви; «лобіст-діставала» - найбільш «пробивний» член команди, що «все може дістати», у якої великі зовнішні зв'язки; «душа команди» – найбільш емоційний та життєрадісний член команди, його Імпульс та Совість.

II тип ситуації: «Вирішення складних проблем». Основні ролі: «генератор ідей» – член команди з найбільш розкутим, образним складом мислення, з високою освітньою підготовкою і широким світоглядом, «генератор ідей» з добре розвиненим творчим мисленням; «концептуалізатор» – член команди зі схильністю та досвідом системного аналізу, широкого узагальнення та перспективного бачення; він вміє надати ідеї «закінчений вигляд», зробити її привабливою і зрозумілою як автору та її соратникам, а й «звичайним» людям; «оптимізатор» – член команди, який має здатність і навичками виділення критеріальних ознак та проведення порівняльних оцінок різних сценаріїв процесів; він вміє зіставити ідею та потреби практики, продумати технологічний цикл реалізації ідеї, оцінити ризики та наслідки, скласти загальний план спільних дій; «реалізатор» – член команди

прагматичного та практичного складу мислення з навичками планування та організації практичної діяльності (можливо це – керівник команди); він вміє практично втілити ідею, організувати всі спільні дії, правильно «розставити всіх на місця»; «критик» - аналізує проблеми з прагматичної точки зору, оцінюють його ідеї та пропозиції таким чином, щоб команда могла ухвалити збалансовані рішення. Важливо відзначити, що команда може покласти на свого найбільш видатного в особистісному плані члена команди не одну, а дві і більше рольових функцій, що природно підвищує його вагу та вплив на команду.

Щоб команда отримувала максимальну користь з усієї різноманітності командних ролей, кожному її члену повинні бути відомі особливості командних ролей своїх колег. Тільки в у цьому випадку можна встановити, чи немає серед цих ролей таких, які не належать до природних сильних сторін членів команди. Якщо це так, то тим членам команди, для яких відсутні природні командні ролі є вторинними, доведеться постаратися заповнити цю прогалину. Очевидно, що для цього знадобиться атмосфера відвертості та довірливості.

РОЗДІЛ 2

ПРИНЦИПИ ТА МЕТОДОЛОГІЇ ФОРМУВАННЯ КОМАНДИ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

2.1. Методології розробки ПЗ

Розробка якісного продукту починається з визначення життєвого циклу. Це чіткий план дій, що дозволяє зрозуміти, що має вийти, як досягти результату та які методи використати. Методологія розробки програмного забезпечення – це перевірені способи та практики, що дозволяють зробити діджитал продукт правильно та добре. Існує кілька різновидів методик, за допомогою яких робляться програми та програми.

Кожен продукт має власний життєвий цикл. Це етапи, якими проходить вся розробка. Починається він із створення ідеї та прийняття рішень. Ви вирішили зробити сайт або програму для своєї компанії. У цей момент життєвий цикл уже розпочато.

Далі проводиться підготовка, проводиться аналіз, вивчається, як цей продукт можна отримати. Коли стає зрозумілим, що ви хочете зробити, потрібно вирішити, як це зробити. На етапі проекту ідея формується у план дій, підбирається стек технологій, та вибираються методології розробки програмного забезпечення.

Це підхід, з якого розроблятиметься товар. І від вибору цього підходу залежить якість кінцевого продукту. Адже вибір моделі розробки ПЗ дозволяє визначити порядок виконання та реалізації завдань, розробити систему контролю та оцінки розробки, сформулювати терміни створення продукту, визначити вартість. Вибір методик дозволяє досягти стабільності в ході розробки, і це одне з основних завдань.

Сам процес розробки програмного забезпечення складається з приблизно однакових етапів. Про них ми поговоримо далі. Але як ці етапи реалізовуватимуться, як вони пов'язані між собою і за якими принципами працюють – визначає методологія розробки ПЗ. Її завдання такі:

- допомагає зрозуміти, як реалізувати ідею та створити продукт, корисний для бізнесу;
- дає чітке уявлення про методи роботи та план дій, що знижує ймовірність помилок у ході розробки;
- допомагає економити на етапі розробки, мінімізуючи кількість правок та прискорюючи процес за рахунок чіткого плану дій;
- дозволяє розподіляти завдання усередині команди залежно від обраного методу;
- утворює послідовність процесів, формує вимоги до продукту.

Необхідність вибирати методології розробки програмного забезпечення обґрунтована якісним підходом. В іншому випадку можна почати створювати продукт і ніколи його не закінчити, тому що не буде чіткого плану дій. Але навіть такий метод можна охарактеризувати в методології, просто вона буде менш ефективною. Існує досить багато моделей, деякі компанії пропонують роботу за унікальними методиками. Але за основу завжди беруться 7 базових, перевірених роками та працюючих методик, що гарантують гарний результат.

Кожна модель розробки програмного забезпечення має свої унікальні особливості, переваги та недоліки. Визначити, яка з них краща – не можна, тому що під різні завдання, продукти та ідеї вибирається свій принцип розробки. Давайте розберемо основні види, що використовуються у програмуванні.



Рис.2.1. Waterfall Model (каскадна модель або водоспад)

Це класична модель, яка використовувалася на зорі розробки, і продовжує активно застосовуватись і сьогодні. Її принцип роботи досить простий: кожен наступний етап виконується лише тоді, коли повністю закінчено попередній. Є чіткий поділ етапів, і технологія відбувається ніби каскадом, поступово спускаючись від першого до останнього.

Ця модель розробки ПЗ досить жорстка, має суворі правила. Вона чітко визначає терміни проходження кожного етапу. Але є недолік: зробити крок назад дуже складно. Внесення правок у існуючий проект коштує дуже дорого і проблематично. Такий спосіб підходить тільки для проектів, чітко розписаних, є повне розуміння, що створюється, для яких цілей і які вимоги виставляються.

Можна використовувати такий підхід у тому випадку, якщо є докладний прототип або існуючий подібний додаток. Крім того, каскадну модель потрібно використовувати при роботі з державними установами, де важливо суворо складати звітності, дотримуватись графіку та не відхилятися від заданого спочатку плану



Рис.2.2 «Incremental Model» (інкрементна модель)

Інкрементна модель розробки ПЗ підходить у тому випадку, якщо є чіткий план дій, але продукт потрібно запустити досить швидко, а зміни можна надалі вносити. Її суть полягає в тому, що спочатку розробляється план дій та поділяється на невеликі завдання. Далі кожен блок розробляється за традиційною каскадною моделлю. Спочатку робиться «базовий» продукт із мінімальними, але важливими функціями. Поступово він доповнюється з допомогою розробки інших компонентів, які називаються інкрементами. Процес зациклюється, доки буде повністю зібраної єдиної системи.

Кожна частина є готовим елементом. Іноді його можна використовувати окремо. Як правило, він розробляється так, щоб не переробляти його. Саме тому використовується каскадна модель всередині інкрементної моделі.

Що така методологія розробки програмного забезпечення? Насамперед, мінімізуються ризики, забезпечується швидкий вихід ринку і запуск продукту. Крім того, базовий функціонал вже працюватиме і приносить вигоди для бізнесу, завжди можна впровадити нові сформовані інструменти, якщо виникає потреба.



Рис.2.3. "Iterative Model" (ітеративна або ітераційна модель)

Такий підхід трохи за своєю конструкцією нагадує попередній. Суть ітеративної методології розробки програмного забезпечення полягає в тому, щоб створити базовий функціонал та поступово його покращувати. Поки що звучить схоже з попереднім варіантом, але є різниця. Інкрементну модель можна порівняти з пазлом, де дістаються всі елементи і поступово збираються в єдину картинку. Ітеративна модель - це ескіз художника, який спочатку робить малюнок олівцем, далі бере в руки фарби, а потім малює деталі і створює картину.

Такий метод підходить для великих проектів, у якому визначаються основні завдання та є загальне уявлення, що має вийти. Але при цьому деталі не зрозумілі, не до кінця зрозуміло, як працюватиме та чи інша функція. За таким методом розробляються, наприклад, соціальні мережі. Загальний функціонал зрозумілий, але кількість функцій та можливості не визначені до кінця. Такий підхід дозволяє поступово додавати компоненти та покращувати існуючі.



Рис.2.4 "Agile Model" (гнучка методологія розробки)

Гнучка методологія розробки програмного забезпечення – це чудове рішення для створення продукту, який не до кінця сформований у своїй ідеї. Особливість даного методу полягає в тому, що замовник може одразу

спостерігати за змінами у розробці та коригувати дії. Це можливо завдяки визначенню спринтів – відрізків, які виконуються завдання. У ході обговорення ставляться цілі, визначається час відрізка та розробники виконують завдання. Далі відбувається обговорення, вносяться зміни, призначається новий відрізок.

Метод гнучкої розробки дуже ефективний, але має недоліки. Через те, що неможливо визначити точні результати та зрозуміти, скільки знадобиться часу на реалізацію ідеї, не можна визначити вартість. Якщо проект налаштований на тривалий життєвий цикл, повинен мати адаптивність до змін на ринку, Agile метод відмінно підходить. Він дозволяє підлаштовуватися під вимоги та вносити виправлення.



Рис.2.5 «Incremental Model» (інкрементна модель)

Інкрементна модель розробки ПЗ підходить у тому випадку, якщо є чіткий план дій, але продукт потрібно запустити досить швидко, а зміни можна надалі вносити. Її суть полягає в тому, що спочатку розробляється план дій та поділяється на невеликі завдання. Далі кожен блок розробляється за традиційною каскадною моделлю. Спочатку робиться «базовий» продукт із мінімальними, але важливими функціями. Поступово він доповнюється з

допомогою розробки інших компонентів, які називаються інкрементами. Процес зациклюється, доки буде повністю зібраної єдиної системи.

Кожна частина є готовим елементом. Іноді його можна використовувати окремо. Як правило, він розробляється так, щоб не переробляти його. Саме тому використовується каскадна модель всередині інкрементної моделі.

2.2. Особливості застосування методології Agile при розробці програмного забезпечення.

Детальніше розглянемо етапи проекту Scrum Agile. Agile – це набір методів управління проектами у галузях, які потребують прикладної роботи. Методологія застосовується збільшення швидкості створення продуктів, зменшення ризиків розробки, збільшення рівня взаємодії між членами команди. Вона забезпечує оперативну реакцію на зміни, що відбуваються, і дозволяє коригувати відхилення.

Вона не схожа на попередні підходи, що описують виробництво продукту в деталях. Agile короткий, у ньому 4 цінності та 12 принципів. RUP на відміну Agile – менш гнучка методологія, при цьому більш об'ємна, описує процес роботи на десятках сторінок. RUP не підходить для невеликих завдань, складається з ітерацій із тривалістю від 2 до 6 тижнів.

Методика застосовувалася в ІТ-індустрії та використовувалася для розробки ПЗ. Суть зводилася до впровадження адаптивних методів, що прискорюють виробництво продуктів через мікропланування та короткі виробничі цикли. Однак згодом Agile почала використовуватись і в інших прикладних областях. Agile зараз застосовують компанії: Netflix, Spotify, Magna International, General Electric, Accenture, M. Відео.

Подібні технології стали надбанням команд, які працюють над створенням клієнтських продуктів.

Переваги методики:

- адаптивні механізми, що дозволяють сфокусуватись на головному;

- відсутність бюрократії;
- скорочення термінів отримання результату;
- збільшення функцій товару.

Недоліки:

- зменшення технічної документації, що позначається на процесі розробки продукту та його експлуатації;
- прискорення планування не співвідноситься з масштабом продукту, що готується, через що можуть виникати помилки в його архітектурі;
- через прийняття швидких рішень формуються дефекти та страждає на якість.

У 2001 році групою авторів Agile Alliance був розроблений маніфест Agile. Вони заснували використання нових способів розробки програмного забезпечення на скасуванні усталених уявлень про менеджмент. Прихильники напряду поставили на перше місце цінність особистості та командної роботи замість інструментів, планів та документації. Після впровадження методики акценти змістилися у бік нових цілей, завдань і принципів, що дозволило адаптуватися до ринку, що змінюється, і споживчих запитів.

Ідеї Agile.

Цінності Agile кажуть, що: люди важливіші за процеси та стандартні інструменти; створення товару важливіше, ніж підбір та узгодження документації; почути позицію клієнта та співпрацювати з ним понад контракт; вносити корективи важливіше, ніж розроблена стратегія.

Методи організації роботи ґрунтуються на каскадній моделі, в якій процеси реалізуються поетапно. Якщо вимоги до розробки продукту, кінцевої мети змінюються, необхідно переробляти виконану частину роботи. На це готові не всі співробітники: деякі до останнього не повідомляють начальство про необхідність коригування в непрацюючому плані. Agile методологія вирішує цю проблему за допомогою наявної гнучкості та адаптивності.

При виборі цієї системи організація швидше працює, мобільніша, поділяє співробітників на невеликі команди. Принципи Agile висловлюють так: розподіл завдань на невеликі блоки, автономність співробітників, прозорість роботи, обробка зворотного зв'язку від клієнтів.

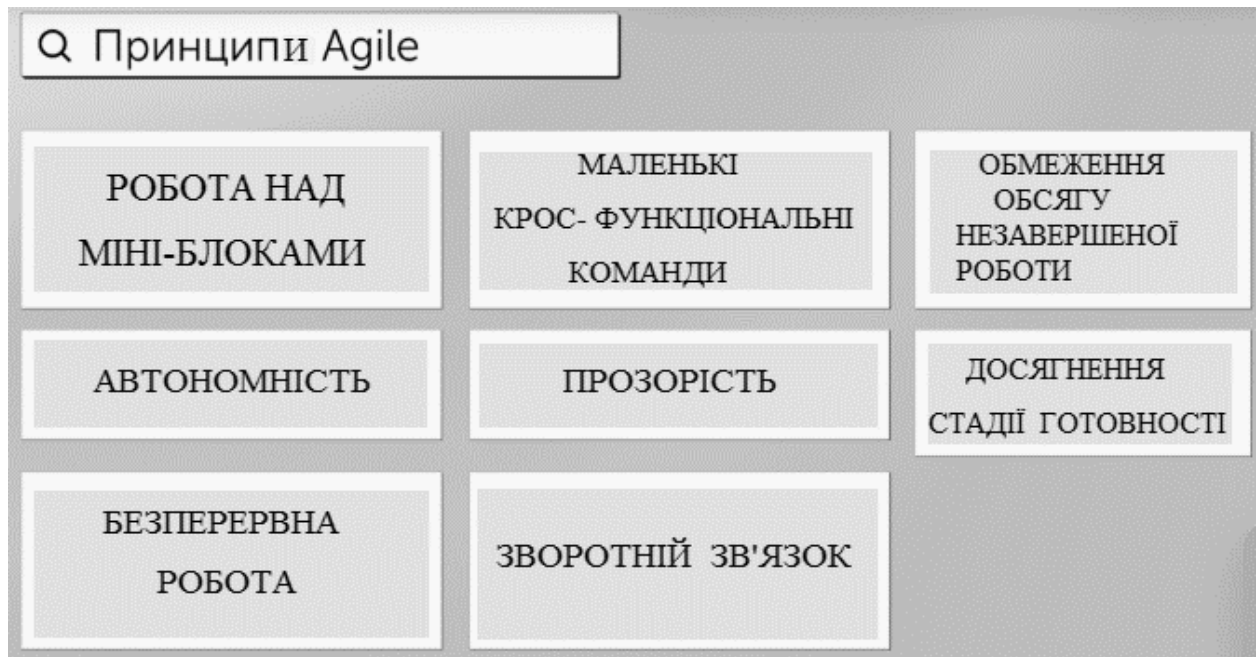


Рис. 2.6 Принципи Agile

Робота над міні-блоками

Складні проекти поділяють на дрібні завдання, кожна з яких міститься в окремий блок. Цілей досягають за короткий цикл, тому навіть у багатоступінчастому проекті видно прогрес у роботі.

Маленькі крос-функціональні команди

Співробітники працюють у невеликих командах. Завдання кожної реалізації однієї з функцій, яка важлива для клієнта. Чисельність і склад команд різняться залежно від своїх завдань. Число співробітників в одній команді до 12 осіб.

Обмеження обсягу незавершеної роботи

Agile допомагає командам концентруватись на завданнях, які можна вирішити за невеликий проміжок часу. Зменшення обсягу роботи допомагає

швидше впоратися із міні-завданнями, що позначається на загальній продуктивності.

Автономність команд

Перед початком роботи над завданням складають план. Потім кожна команда вирішує, як розпочати його виконання. Завдання керівника – визначити базові правила, співробітники самостійно обирають темп роботи, умови, координують дії.

Досягнення стадії готовності

Перевірка методології – завершення завдання, підсумки якого підбиваються наприкінці кожного циклу. Завдяки поділу на невеликі блоки команди можуть повністю завершити завдання, а не відзначати його як «практично закінчену». Причина повільності у великих проектах – завдання, які завершені частково, але ще ряд проблем. В результаті вони тягнуть час, ресурси, увагу та відволікають співробітників від зобов'язань.

Безперервна робота

Завдання, поділені на короткі цикли, мають пріоритети, яких потрібно прагнути кожному етапі. Завдяки чому робота безперервна і співробітники не відволікаються на суміжні завдання.

Повна прозорість та використання дощок зі стікерами

Це допомагає коротко, але ємно описати роботу, зафіксувати актуальну стадію, де знаходиться команда, подивитися на процес очима співробітників, за необхідності визначити джерело проблем.

Зворотній зв'язок від користувачів на кожному циклі

Команди отримують зворотний зв'язок від клієнтів наприкінці кожного циклу. На основі даних оцінюють досягнення, реалізацію завдання. Інформація враховується надалі для планування.

Ключові моменти у застосуванні

Методи Agile застосовуються для вирішення різних бізнес-процесів. Тому перед впровадженням важливо розібратися, як виглядає методологія на практиці. Які існують ролі по Agile

Ролі у системі:

Власник товару – не присвячений технічні подробиці реалізації, але має загальне бачення, розуміє, для якої аудиторії робиться товар, які завдання він повинен виконувати.

Координатор дій – відповідає за процеси у команді та спрямовує потенціал співробітників.

Команда розробників - бере участь у створенні продукту, реалізує його технічну складову.

Ієрархія компетенції в Agile

Структура команди горизонтальна, але є ієрархія. Керівник задає вектор, яким співробітники реалізують завдання. Особливість системи – ієрархія побудована на компетенції, а чи не влади, що визначає взаємодія співробітників із начальством.

Що таке пропускна спроможність

Пропускна здатність – реалізована кількість «історій користувача». Це побажання клієнтів, які формуються у завдання. Наприклад, встановлення фільтрів пошуку у додатку, покращення зворотного зв'язку з клієнтами, робота над службою техпідтримки. Пропускна здатність вимірюється кількістю відпрацьованих історій користувача на тиждень.

Як визначити послідовність та пріоритетність завдань

Пріоритетність завдань залежить від напряму компанії. Наприклад: Value Based – оцінка цінності бізнесу. Кожне завдання вивчається з точки зору її прибутковості, підвищення репутації та загального рівня задоволеності користувачів.

Technology Risk Based – оцінка технологічних ризиків. Пріоритет розподіляється виходячи з ризику реалізації вимог. Ризик у роботі компанії буває через велику кількість поставлених умов, зовнішньої взаємодії.

Як скласти графік розв'язання задач

Для складання графіка використовують програми із шаблонами для планування проектів. Наприклад, GanttPRO – сервіс для постановки завдань та

їхнього контролю. Він містить шаблони графіків, відстежує цілі за рівнем просування, відзначає слабкі місця.

Для застосування методології виконують комплекс заходів. Він ґрунтується на виборі головного методу в системі, після чого встановлюються завдання, цілі, терміни, чисельність команди. Співробітники мають бути навчені застосуванню методів на практиці, а керівництво має розуміти, що впровадження системи стане новим поворотом у розвитку бізнесу.

Важливо використовувати досвід фахівців, які вже працювали із системою та знають, як реалізувати її. Їхній досвід допомагає у формуванні команди, підборі інструментів, аналітиці.

Поширені проблеми під час реалізації (рис.2.7)



Рис.2.7 Основні проблеми при впровадженні Agile

Основні проблеми при впровадженні Agile:

Відмова від концепції «керівник – підлеглий». У методології команди будуються відповідно до особливої філософії. Свободу дій та зниження формальності у робочих відносинах витримують не всі компанії.

Відсутність згуртованості команди. Комуś зручніше виконувати завдання поодиночі. За системи Agile комунікувати доведеться постійно.

Марно витрачені зусилля. Іноді концепція проекту змінюється, через що доводиться брати курс на зміну підходу та завдань. Робити це доводиться швидко, через що не кожна команда виявляє мобільність та відпускає витрачений час.

Методологія Agile допомагає бізнесу досягати цілей. Її принципи дозволяють у стислі терміни готувати для клієнтів нові продукти з урахуванням їхніх думок та реакцій, що передаються через зворотний зв'язок.

2.3. Проблеми формування та управління командою розробників програмного забезпечення.

Основними факторами для створення команди є:

- Необхідність підвищення швидкості прийняття рішень. Забезпечення більшої гнучкості організаційної структури;
- Зростання складності діяльності. Посилення значення спеціалізації та, як наслідок, необхідність підвищення якості взаємодії;
- Зростання конкуренції на ринки праці. Необхідність високонадійних систем відбору, підготовки, адаптації, розвитку та стабілізації персоналу;
- Зростання значення негрошової мотивації, розвиток теорії що залучає менеджменту;
- Збільшення навантаження (інформаційної, емоційної, фізичної), що призводить до необхідності розвитку системи дублювання, взаємодопомоги та взаємозамінності.

Кожна команда має свій професійний почерк, є свої специфічні внутрішньокомандні (внутрішньогрупові) відносини:

- за наявності команди з управлінського середовища йде стиль роботи у вигляді затяжних та часто безрезультативних нарад.

Короткі ділові робочі зустрічі – перший показник командного стилю роботи;

– практично знімається питання про виконавську дисципліну, бо працює морально-психологічний механізм «взаємного відповідальності» (взаємозалежності через загальну ставку) тощо.

Таблиця 2.1

Основні проблеми, що впливають на процес формування команд

Проблема № 1.	Якість підбору персоналу (рекрутинг).
Проблема № 2.	Комунікативні канали та їх якість.
Проблема № 3.	Якісний та кількісний склад команди.
Проблема № 4.	Розмір команди.
Проблема № 5.	Формування проектної команди із окремих професіоналів.
Проблема № 6.	Мультикультурні та міжетнічні взаємовідносини

Задачі, які необхідно розв'язати на етапі створення та на етапі гуртування команди наведено у табл.2.2.

Причинами доцільності командної організації виконання проектів є їх комплексність, часто міждисциплінарний характер розв'язуваних проблем, новизна а, особливо гостра необхідність збільшення творчого потенціалу співробітників як за рахунок ефекту синергії, так і за рахунок горизонтальних зав'язків, відсутність таких стримуючих творчості факторів, як стереотипи, рутинні процедури.

Основними факторами, які впливають на формування команди є:

- зовнішні фактори, такі як професійні вимоги до членів команди і відповідність рольового складу, що визначаються на етапі інноваційного макета проекту;
- внутрішні чинники, такі як соціотипи особистості претендентів в проектній команді з обмеженого числа співробітників компанії, які визначають взаємовідносини членів команди.

Таблиця 2.2

Декомпозиція процесу формування команди

Етап	Опис	Вхід	Вихід	Відповідальний
Аналіз необхідних ресурсів розробників.	Визначення кваліфікаційних (професійних) вимог.	Технічне завдання.	Перелік ресурсів включно із складом та кваліфікацією окремих виконавців.	Менеджер проекту.
Аналіз наявних ресурсів.	Відображення вимог на існуючий склад розробників.	Перелік ресурсів включно із складом та кваліфікацією окремих виконавців.	Перелік наявних Спеціалістів включно із їх Поточним завантаженням на інших проектах, перелік відсутніх ресурсів.	Менеджер проекту разом із HR (Human Resources).
Визначення вакансій та підбір персоналу.	Рекрутинг нових спеціалістів, перерозподіл наявних спеціалістів.	Перелік наявних та відсутніх ресурсів включно із складом та кваліфікацією окремих виконавців.	Укомплектована команда.	Менеджер проекту разом із HR (Human Resources), технічні спеціалісти (інтерв'ю).
Планування роботи команди.	Ознайомлення команди із технічним завданням, планом розробки, визначення відповідальних та термінів робіт.	Технічне завдання, склад команди.	План робіт узгоджений із усіма членами команди.	Менеджер проекту, керівник групи.
Підготовка та координація заходів щодо навчання.	Організація необхідних технічних тренінгів (onboarding).	Технічне завдання, ресурси для розробки.	Матеріали і план для технічних тренінгів.	Менеджер проекту, інструктори, система дистанційної освіти (при наявності).
Навчання співробітників.	Проведення тренінгів згідно планів та графіків.	Матеріали для технічних тренінгів.	Документи, що підтверджують проходження тренінгів та відповідну набуту кваліфікацію.	Менеджер проекту, інструктори, система дистанційної освіти (при наявності).
Гуртування команди, підтримання мікроклімату, розвиток команди.	План заходів щодо гуртування команди (team buildings).	Склад команди, аналіз професійних та людських якостей.	Команда.	Менеджер проекту.

Розвиток передових інформаційних технологій, світова практика комплексного управління якістю, необхідність застосування інновацій потребує зміщення акценту з індивідуальної роботи окремих виконавців на діяльність багатофункціональних управлінських команд і робочих груп, сконцентрованих на комплексних проблемах і задачах. Згідно методології управління командами після розподілу ролей і відповідальності, кожний член команди має приймати участь у плануванні проекту і прийнятті рішень.

Склад команди в процесі розвитку проекту не є сталим, на різних етапах до неї підключаються спеціалісти для вирішення конкретних короткотермінових задач, наприклад, налаштування служб СІ, написання драйвера пристрою тощо. Це створює додаткові виклики перед менеджментом та вимагає толерантного ставлення до новачків з боку інших членів команди. Також поширеною є практика часткової зайнятості спеціалістів.

РОЗДІЛ 3

ФОРМУВАННЯ КОМАНДИ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ В AGILE ПРОЕКТАХ

3.1 Формування та (управління) команди розробки програмного забезпечення.

Незалежно від того, яка методологія розробки програмного забезпечення обрана, вони дотримуються основних стадій створення будь-якого діджиталу продукту. Залежно від обраного методу стадії можуть змінюватися, використовуються інші підходи, але принцип залишається таким самим. Розробка ПЗ відбувається за такими етапами:

Аналіз вимог. Спершу вивчається ідея. Важливо зрозуміти, що клієнт хоче одержати, як він бачить свій майбутній продукт. Це дозволяє сформулювати ідею та оцінити її перспективи. Далі вивчаються конкуренти, подібні продукти на ринку, цільова аудиторія, можливості та проводиться низка аналітичних досліджень, що дозволяють зрозуміти, як перетворити ідею на працюючий продукт.

Проектування. Складається план завдань, створюється проект, що дозволяє вже цьому етапі побачити, як продукт виглядатиме, як його буде реалізовано. Враховується дизайн елементів, продумується структура, створюється архітектура програми. Для розробників складається технічне завдання.

Розробка та програмування. Дизайнери приступають до створення зовнішньої частини програми та розробки UX. Вони створюють модулі, зв'язки компонентів, продумують повну структуру та роблять верстку. Програмісти втілюють це у робочий інструмент, використовуючи обраний стек технологій. На цій стадії розробки програмного забезпечення виходить готовий продукт, який можна впроваджувати в бізнес.

Документація. Кожна дія описується. Створюється технічна документація для програмістів, які надалі надаватимуть технічну підтримку, а

також керівництво користувача. У документації детально описуються основні засади роботи програми та її функціонал.

Тестування. Коли програма готова, вона відбувається ретельне тестування. Методологія розробки програмного забезпечення визначає варіанти тестів, способи оцінки ефективності розробки. Якщо під час тестування визначаються помилки, вони виправляються. Також розробляється система подальшої оцінки працездатності програми.

Впровадження та супровід. Як тільки продукт пройшов тести, він впроваджується у бізнес. Навчаються співробітники, технічна команда, яка надаватиме супровід. Якщо такої немає, ви можете замовити послугу повного супроводу та бути впевненим, що програмне забезпечення працює без збоїв.

Це основні стадії розробки програмного забезпечення. Кожен етап – це комплекс заходів, що дозволяє створити продукт бізнесу. Залежно від обраної моделі розробки ПЗ вони можуть змінюватися, проходити циклами або бути гнучкими. Саме в цьому полягає головне завдання вибору методу – визначити найбільш ефективний і раціональний шлях створення продукту.

Кожна з представлених методологій має свої сильні та слабкі сторони та підходить для різних проектів розробки. На рис. представлена таблиця, яка допоможе вирішити, який із них вибрати для вашої програми.



	Agile	Iterative Model	Waterfall	Incremental Model
Project size	Small and medium	Large and medium	Any	Medium and small
Product owner involvement	High	High	Low	High
Timeline	Flexible	Varied	Strict	Short
Budget	Flexible	Varied	Fixed and unchangeable	Flexible
Documentation	Light	Light	Elaborated	Light
Possibility to change requirements	Yes	Yes	No	Yes
Depend on the user feedback	Yes	-	No	Yes
Team collaboration	High	Extremely high	Low	High
Automation level	Varied	High	Low	High

Рис.3.1. Вибір потрібної моделі

Успішна робота над проектом потребує злагодженої команди та кваліфікованих кадрів. Нижче наведено приклад інтеграційного проекту, в якому об'єднані зусилля всіх учасників. Такий варіант міститиме безліч різноманітних завдань, що потребують досвіду та різних ресурсів.

Завдання можуть призначатися різним спеціалістам та організаціям як усередині, так і за межами компанії. Структура такої робочої групи ґрунтується на проектно-орієнтованій ієрархії.



Рис.3.2. Структура проєктної команди

Розглянемо загальну модель формування та розвитку команди (рис.3.3). Вона описує сутність процесу управління командою.



Рис. 3.3 Загальна модель формування та розвитку команди

На рисунку 3.4 зображена модель «Підбір членів команди». Вона описує сутність підбору спеціалістів для комплектації групи розробників програмного забезпечення.

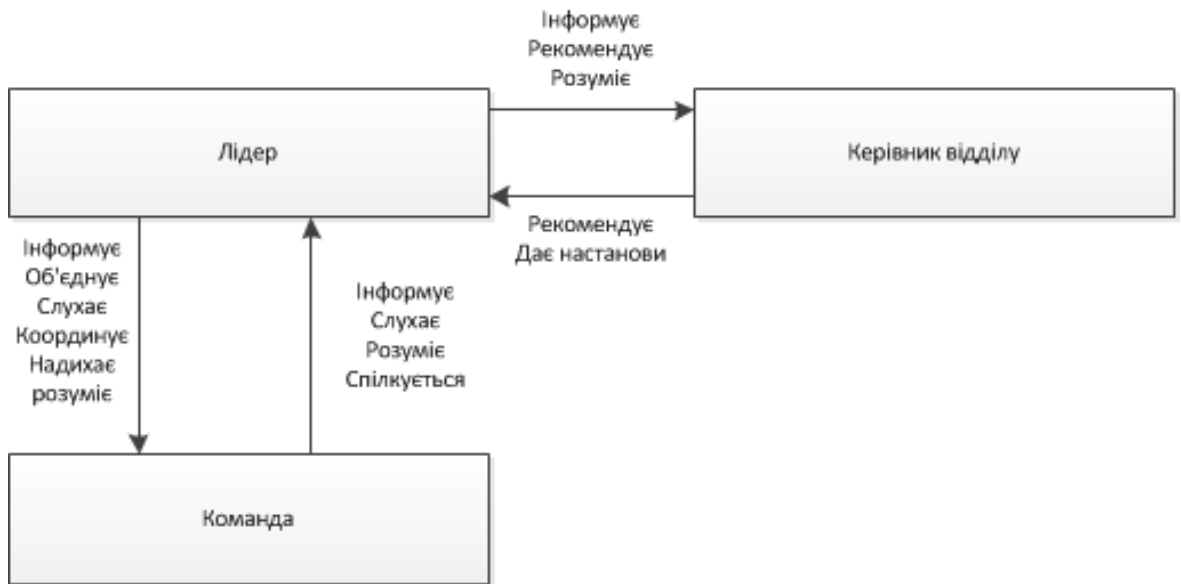


Рис. 3.4 Підбір членів команди

На рисунку 2.3 зображена модель «Впровадження комунікацій». Вона описує процеси які потрібно залучити для впровадженні комунікацій.



Рис. 3.5 Впровадження комунікацій

На рисунку 3.6 зображена модель «Конфлікт виробничий». Вона описує сутність процесів управління виробничими конфліктами у команді.



Рис.3.6 Конфлікт виробничий

На рисунку 3.7 зображена модель «Конфлікт особистий». Вона описує сутність процесів управління особистими конфліктами у команді.



Рис. 3.7 Конфлікт особистий

На рисунку 3.8 Зображена модель «Залучення нового члена команди». Вона описує сутність процесів необхідних при залученні нового члена у команду. Базується на моделі «Підбір членів команди», але має свої особливості.

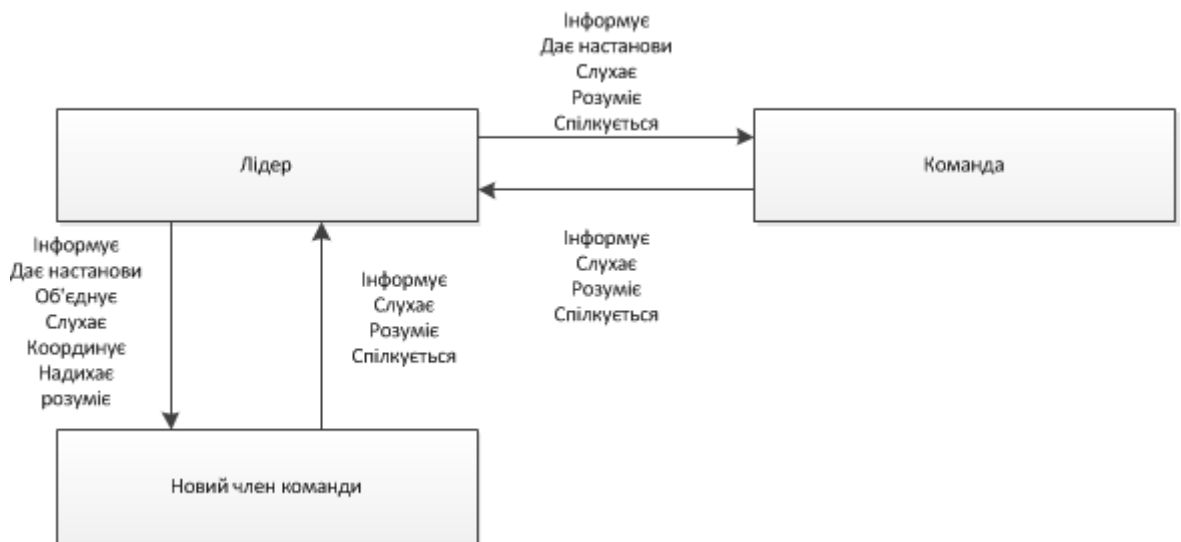


Рис. 3.8 Залучення нового члена команди

На рисунку 3.9 Зображена модель «Атестація». Вона описує сутність процесів управління мотивацією команди після закінчення проекту. Важливо компаніям які «вирощують» кваліфікованих співробітників.



Рис. 3.9 Атестація

На рисунку 3.9 зображений «Загальний механізм формування та розвитку команди». Він описує сутність поверхневого алгоритму управління командою.



Рис. 3.9 Загальний механізм формування та розвитку команди

На рисунку 3.10 зображений механізм «Підбір членів команди». Він описує сутність алгоритму підбору команди.

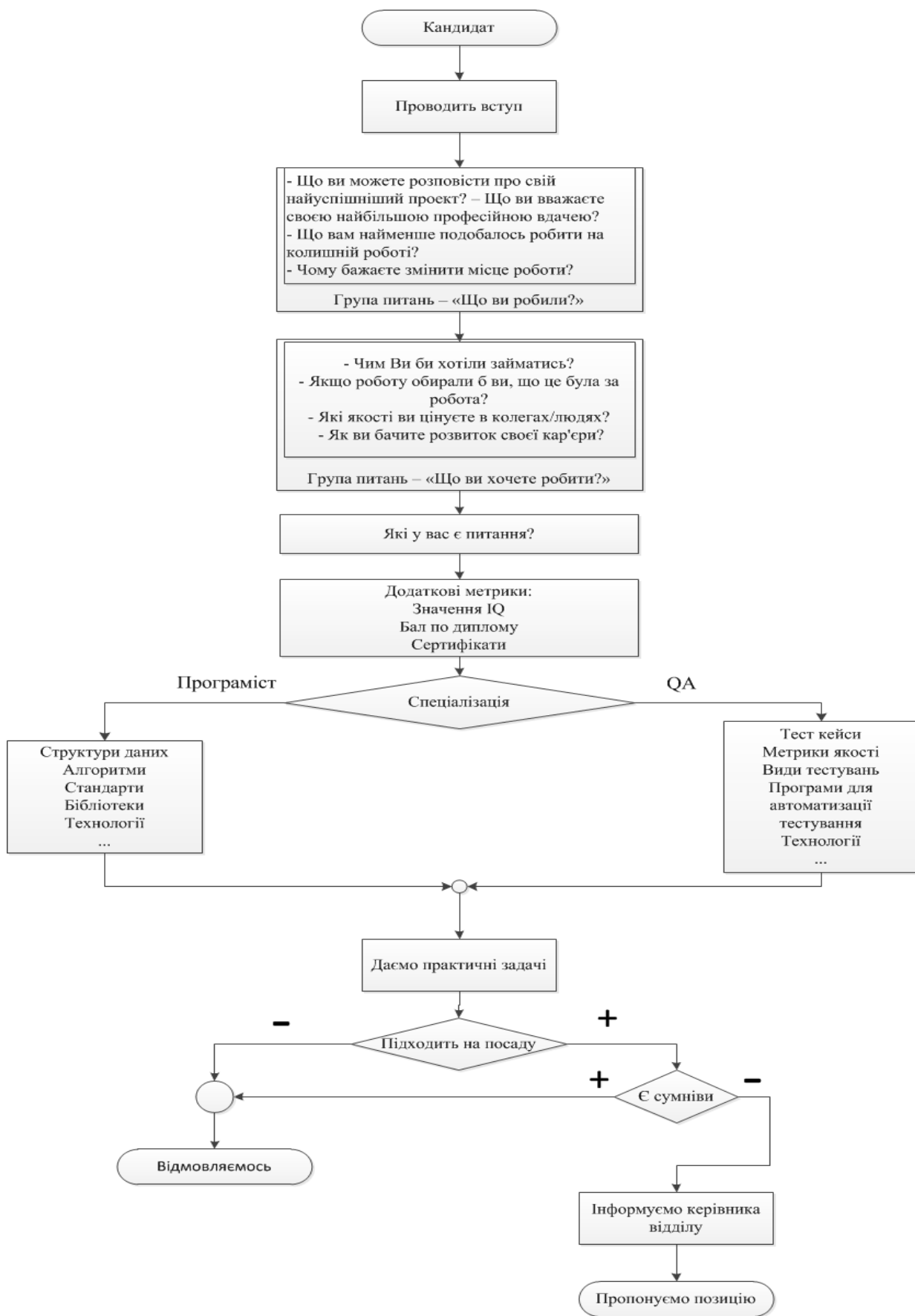


Рис.3.10 Підбір членів команди

На рисунку 3.11 зображений механізм «Впровадження комунікацій». Алгоритм показує сутність усіх процесів необхідних для впровадження ефективних комунікацій в команді, що є фундаментом команди.

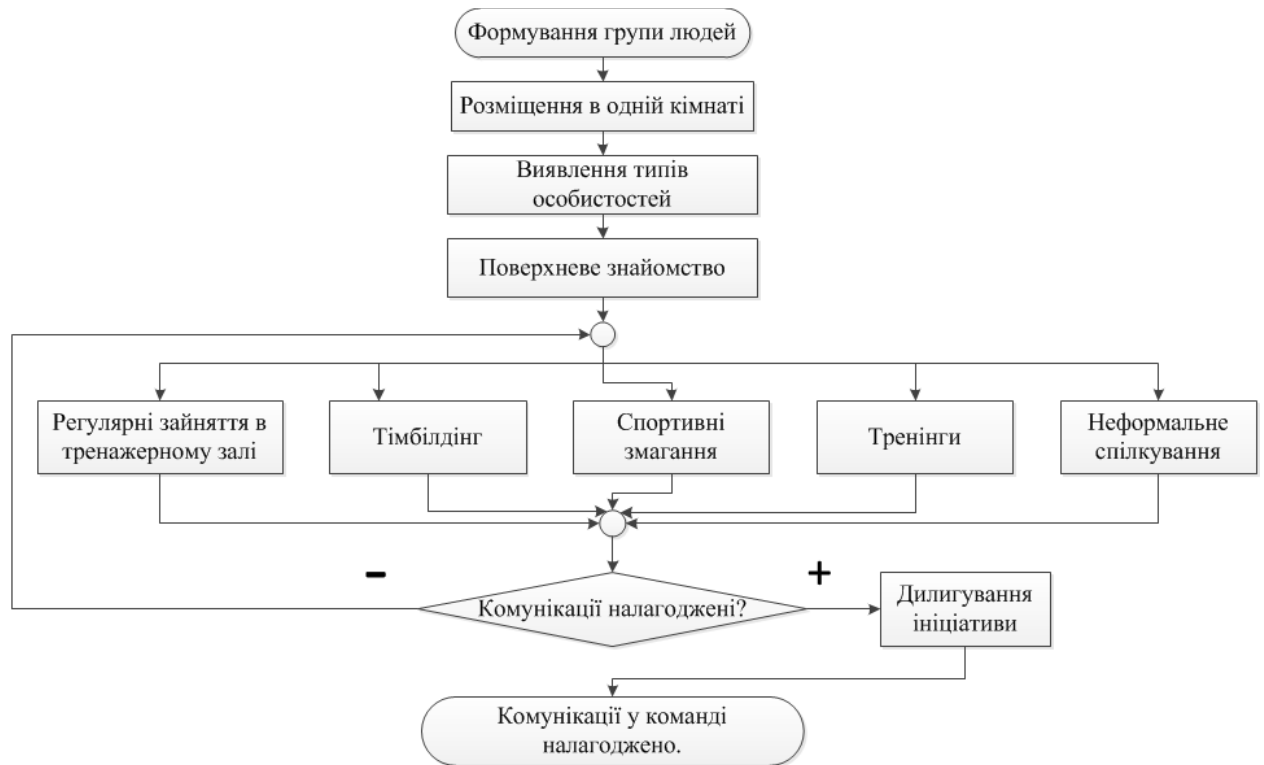


Рис.3.11 Впровадження комунікацій

На рисунку 3.12 зображений механізм «Мотивація команди». Алгоритм покриває усі типи особистості членів команди та кваліфікацію, що дає змогу ефективно мотивувати команду на протязі розробки програмного забезпечення.

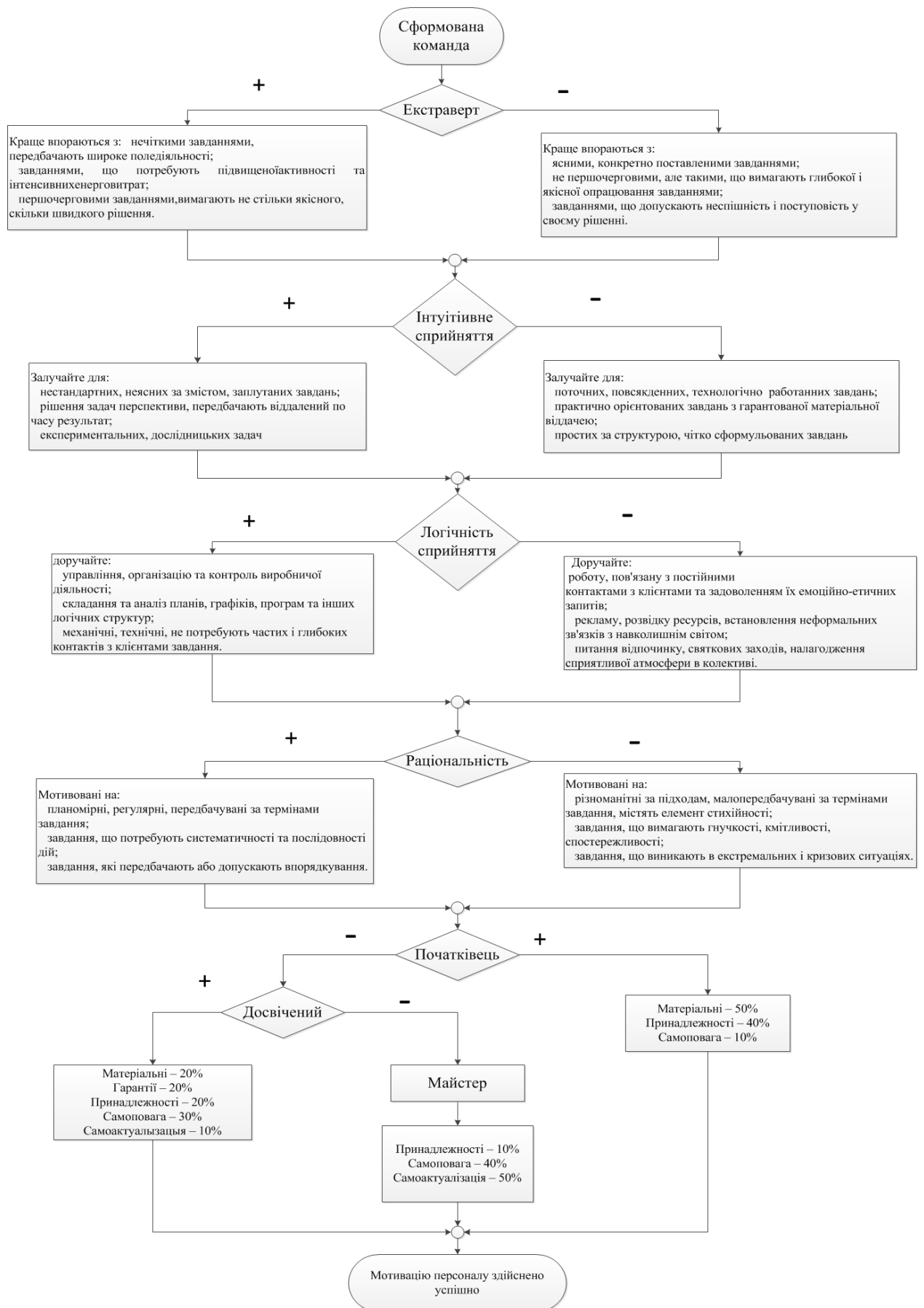


Рис. 3.12 Мотивація команди

На рисунку 3.13 Зображений механізм «Конфлікт виробничий». Алгоритм вирішує проблему розрішення виробничих конфліктів за правилом «виграв/вигрва».

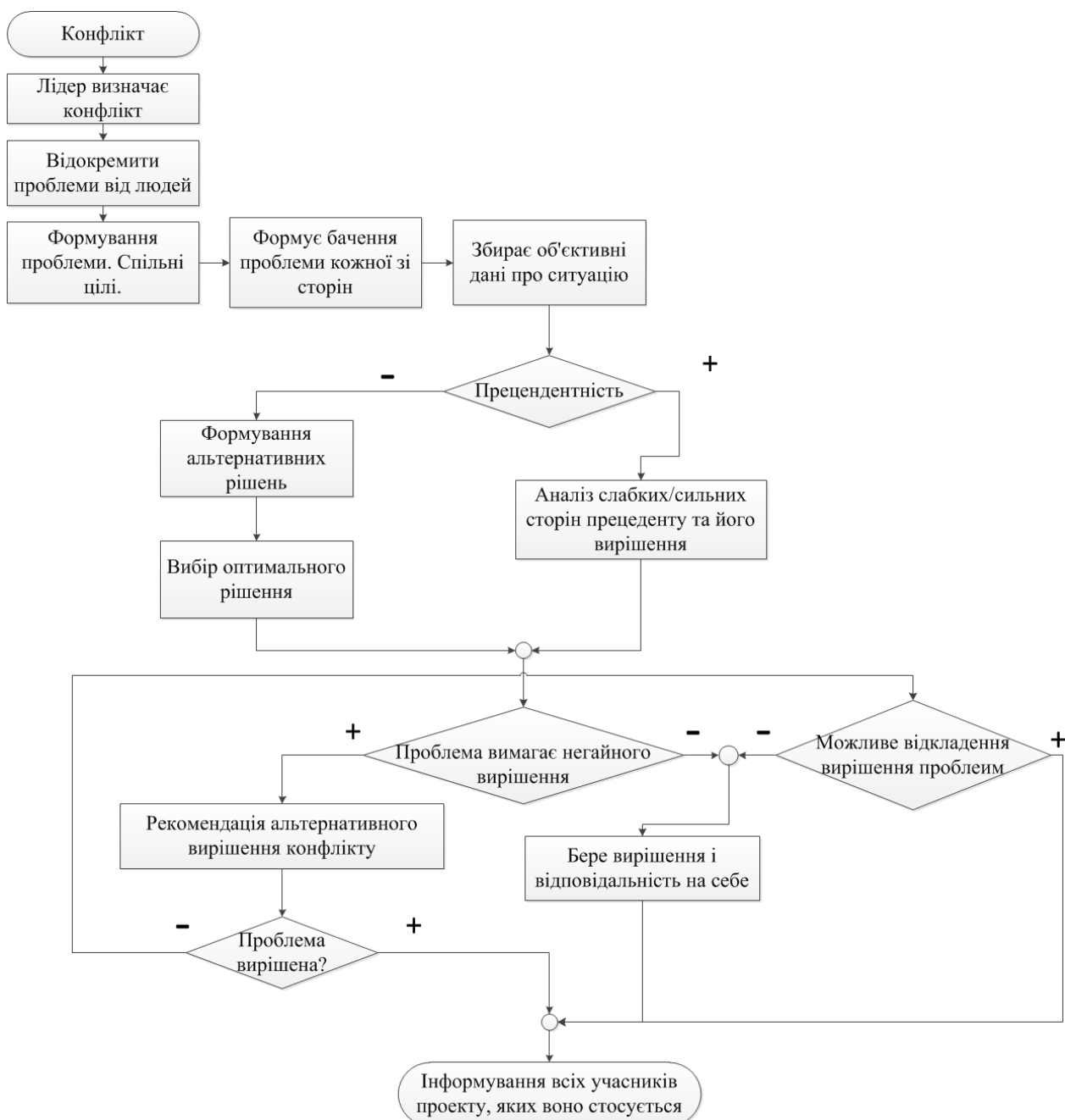


Рис. 3.13 Конфлікт виробничий

На рисунку 3.14 зображений механізм «Конфлікт особистий». Алгоритм вирішує проблему розрішення особистих конфліктів, які в принципі не мають місця у команді. Якщо проблему не вдається вирішити одразу – джерело особистих конфліктів повинно покинути команду якомога швидше. При цьому

необхідно одразу ж почати пошук заміни, бо втрата одного члена команди призведе до тимчасового перевантаження (якщо був запас часу то цього може і не статись, просто витратиться резерв часу).

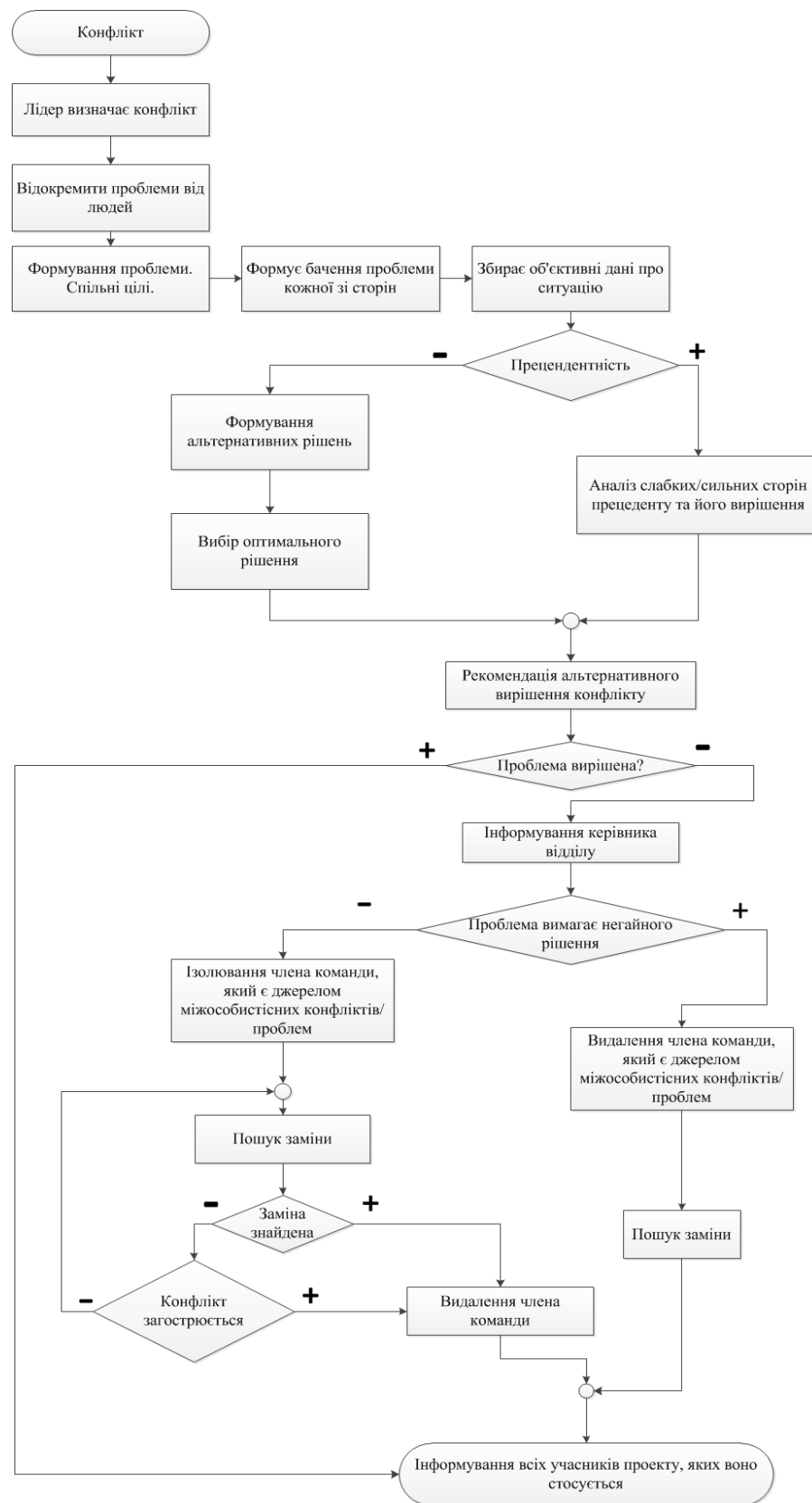


Рис. 3.14 Конфлікт особистий

На рисунку 3.15 зображений механізм «Залучення нового члена команди». Алгоритм описує процеси які треба впровадити одразу коли нова людина прийшла у команду.

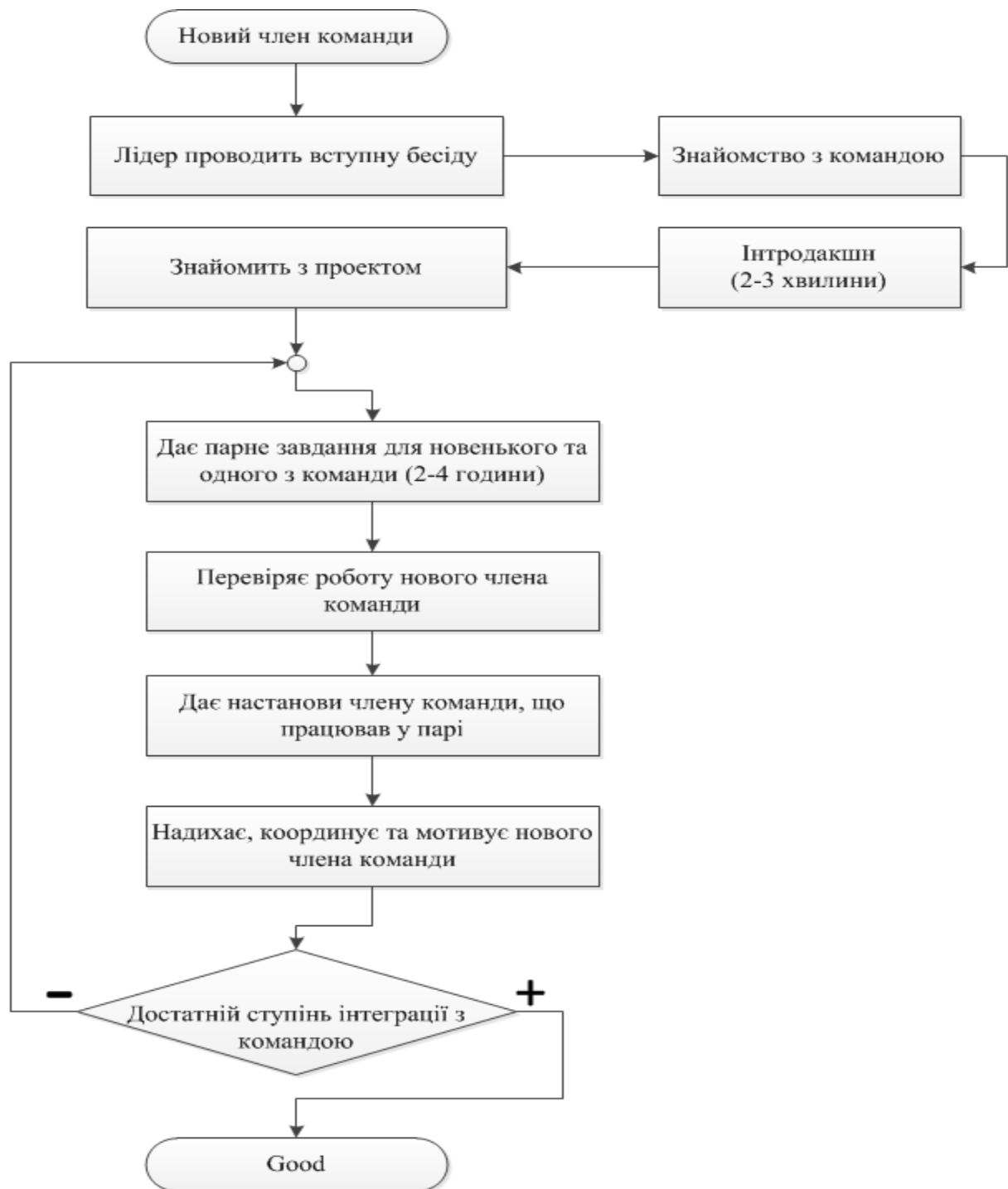


Рис. 3.15 Залучення нового члена команди

На рисунку 3.16 зображений механізм «Атестація». Алгоритм описує процеси які мотивують людину на роботу у компанії та допомагають їй зростати як спеціалісту.

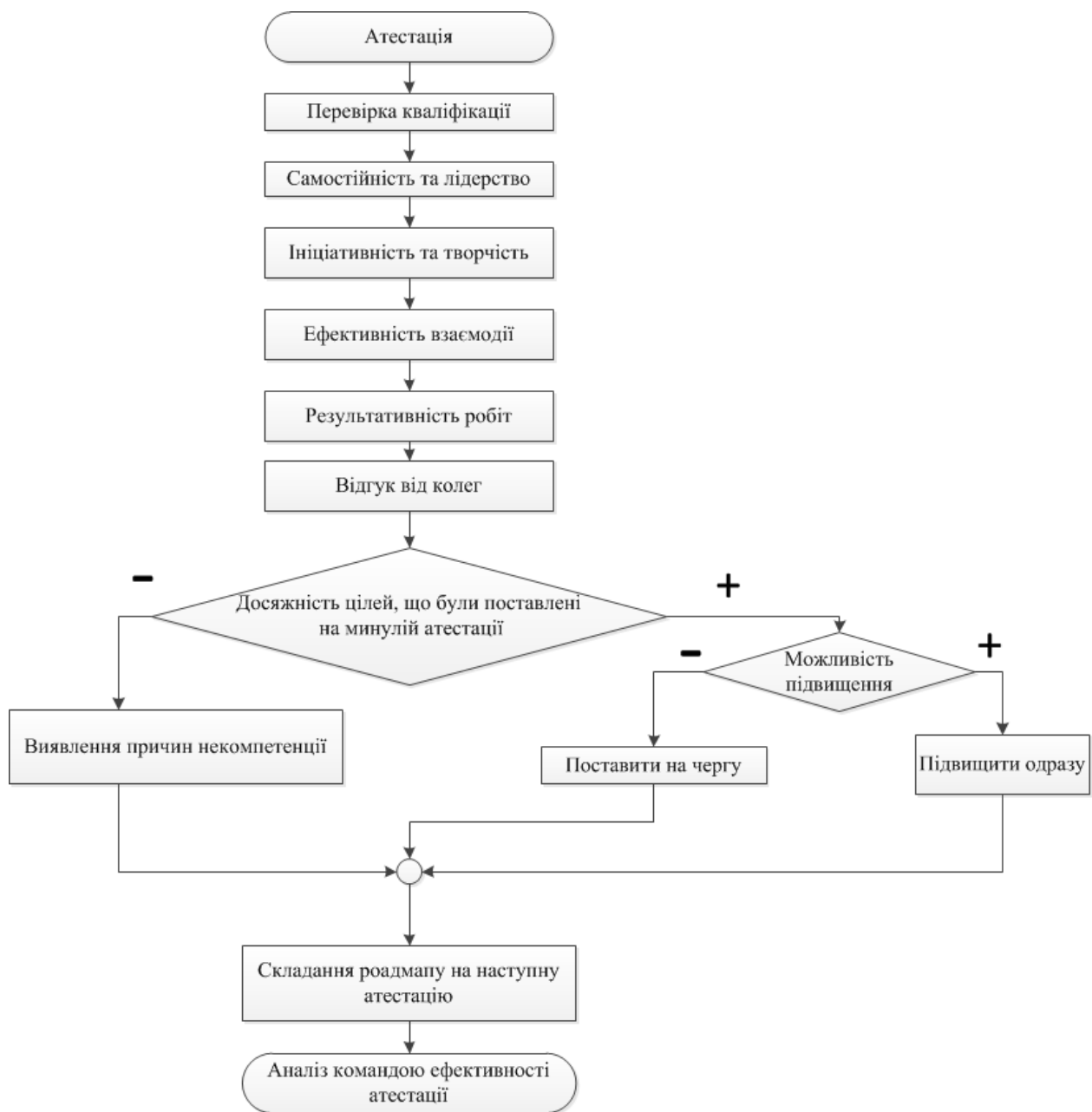


Рис. 3.16 Атестація

3.2 Практичне застосування удосконалених моделей формування команди

В процесі реалізації проекту розробники стикались з проблемами, які було вирішено використовуючи удосконалені моделі управління командою при розробці програмного забезпечення.

Розроблявся мобільний додаток «Спортивна країна». Любої людині бисто та зручно можливо записатися до спортивної секції, зареєструватися на різні заходи та вести історію своєї активності.

З самого початку проекту було використано модель «Загальна модель формування та розвитку команди» для того щоб визначитись з загальною стратегією та сформувані базові цілі.

Сформувавши загальну стратегію, можна було перейти до її реалізації. Першим пунктом якої був знаходження потрібних людей для формування команди розробників програмного забезпечення. Було використано модель «Підбір членів команди» для формування ефективної групи розробників програмного забезпечення. Керівник проекту провів з кожним кандидатом співбесіду на якій, використовуючи удосконалену модель, дізнався про всі необхідні данні для прийняття рішення щодо залучення кандидата у проєкті.

Спочатку керівник провів вступну промову. Розповів про компанію, її історію, чим займається, як досягла успіхів, що цінується, які принципи та інше. Потім розповів про проєкт який потенційно може чекати кандидата, якщо той вдало пройде співбесіду. Розповів про задачі які ставляться перед розробниками, які терміни на розробку і як буде проходити розробка програмного забезпечення. Розповів про очікування від кандидата і те як він бачить майбутню команду.

Отримавши відповіді на всі питання була дана можливість задати питання кандидату для того щоб усі мали повну картину бачення ситуації.

Далі керівник поцікавився у кандидата де той вчився, наявність диплому і про те як він його захистив. Що подобалось і не подобалось у в процесі навчання, чим кандидат може пишатись. Значення IQ, якщо той його

проходив, якщо ні то можна було запропонувати кандидату його пройти на місці, використавши спеціально розроблені програми. Також були зібрані інші дані такі як: сертифікати, доклади, виступи на різних конференціях та ступінь начитаності (особливо за спеціальністю).

Далі в залежності від того яка спеціалізація у кандидата було проведено поглиблену технічну співбесіду. У ході якої було задано сотню питань, відповіді на які діли змогу зрозуміти теоретично-практичний рівень підготовки та базових знань кандидата.

Одразу після питань були задані практичні задачі. Це дало змогу зрозуміти наскільки кандидат вміє застосовувати свої знання та досвід на практиці.

Особливу увагу було приділено тому як кандидат вміє висловлювати свою думку, всіх хто не міг виразно та лаконічно виразитись було майже одразу відсіяно. Бо якщо він не може виразити свою думку на співбесіді, де треба показати максимальний рівень знань та вмінь, то у процесі розробки програмного забезпечення він просто не зможе працювати з іншими членами команди, бо не зможе виразити свою думку.

Одразу після цього був проведений аналіз інформації, що дало змогу зробити висновок: «чи підходить кандидат у команду?».

Якщо не підходив, або підходив, але були вагання то йому відмовляли. Якщо підходив і не було вагань - інформувався керівник відділу, який разом з керівником проекту робив висновок яку саме робити пропозицію, який рівень компенсаційних виплат, бонусів та інших умов праці робити.

Коли все це було узгоджено (на це виділялось до пів години часу) - кандидату пропонували посаду у команді[27].

Завдяки цьому, вдалося вдало підібрати усіх членів групи і забезпечити можливість використання інших моделей, що ґрунтувались на цій.

Другим пунктом було впровадження комунікацій, для переходу із групи розробників до команди розробників програмного забезпечення. Було використано модель «Впровадження комунікацій». Спочатку для групи було

виділено окрему кімнату, щоб вони могли працювати у «своєму світі». Далі лідер виявив типи особистостей кожного члена команди, проявивши ініціативу поверхневого знайомства. Далі було проведено цілий цикл різних заходів для впровадження комунікації: регулярні заняття в тренажерному залі, тимбілдинги, спортивні змагання, тренінги і багато неформального спілкування. Коли комунікації були налагоджені лідер делегував ініціативи, забезпечивши підтримку комунікацій на нормальному рівні.

Третім пунктом була мотивація команди, на початку проекту всіх мотивував сам початок проекту але після того як були впроваджені комунікації – мотивація потрохи почала у деяких членах команди згасати, тому для підтримки мотивації на достатньому рівні було вирішено використати модель «Мотивація команди». Лідер почав рекомендувати, залучати та доручати завдання в залежності від типу особистості. Також в залежності від типу кваліфікації лідер корегував інші види мотивації: компенсаційний рівень, гарантії, приналежність, самоповага та самоактуалізація. Також важливим кроком було своєчасне впровадження комунікацій, що дозволило делегувати задачі, а делегування – це вища форма мотивації.

В процесі розробки програмного забезпечення часто виникали виробничі конфлікти між членами команди, які розглядались використовуючи модель «Вирішення виробничих конфліктів». Основою моделі лежить принцип «виграв/виграв» який дозволяє вирішити конфлікт так, щоб усі залишились у виграші.

На практиці була ситуація коли на «скрамі» обидва програмісти виявили бажання зробити одну важку задачу і кожен при цьому хотів зробити його своїм методом, вважаючи саме його більш ефективним. Спочатку лідер визначив конфлікт. Відокремив проблеми від людей. Знайшов проблеми (корінь зла), сформував спільні цілі. Далі подивився на проблему з точки зору кожної сторони (опинився в їх шкурі). Потім зібрав об'єктивні дані про ситуацію, опитав інших членів команди. Оскільки раніше на практиці такого

не було формовано альтернативні рішення проблеми. Далі було обрано оптимальне рішення. Оскільки проблема вимагала негайного рішення - лідер взяв вирішення і відповідальність на себе, запропонувавши свою альтернативу як наказ до її виконання. Це було парне програмування, де обидва програмісти могли об'єднати свої ідеї та зробити реалізацію завдання максимально ефективним. Це звичайно забрало часу стільки ж скільки б його робив один програміст, але це дало ефективне вирішення конфлікту без його загострення де усі залишились к переможцям. Бо проект отримав ефективну реалізацію важкої частини коду, програмісти залишились задоволеними, а лідер «зберіг» команду та зменшив ризики виникнення дефектів у складній частині коду.

Також в процесі розробки два рази виникали особисті конфлікти між членами команди, які розглядались використовуючи модель «Вирішення особистих конфліктів».

Спочатку лідер визначив конфлікт. Відокремив проблеми від людей. Знайшов проблеми (корінь зла), сформував спільні цілі. Далі подивився на проблему з точки зору кожної сторони (опинився в їх шкурі). Потім зібрав об'єктивні дані про ситуацію, опитав інших членів команди. Оскільки раніше на практиці такого не було формовано альтернативні рішення проблеми. Далі було обрано оптимальне рішення. Але конфлікт не вдалось вирішити. Було проінформовано керівника відділу і разом з ним було вирішено план дій. Оскільки проблема вимагала негайного рішення – було вирішено видалити члена команди який був джерелом міжособистісних проблем/конфліктів. Далі було розпочато пошук заміни. Також були проінформовані члени команди та керівництво компанії.

Оскільки команда втратила одного члена команди, що вплинуло на завантаження команди, було кинуте багато зусиль на пошук заміни. Процедура підбору кандидата було реалізовано використовуючи модель «Підбір членів команди», яку вже розглядали нижче. Коли обрали підходящого кандидата у команду на заміну – постала необхідність ефективного та швидкого залучити його у команду. Для вирішення цієї

проблеми було знову ж використано удосконалені моделі та механізми управління командою, а саме модель «Залучення нового члена команди». Спочатку лідер провів вступну бесіду. Потім познайомив з командою. Далі новий член команди розповів коротко про себе, щоб команда могла знати приблизну уяву про новенького і мала базові поняття на які теми можна точно заводити теми під час неформальної бесіди. Далі проводилось ознайомлення з проектом, де розповідалось, що було зроблено, що зараз робиться і що буде зроблено.

Потім лідер дав парне завдання (на 2-4 години) для новенького та одного із команди, для того щоб налагодити комунікації між новеньким та командою (та зробити це «точково», а не масово, для початку). Після того як робота була виконана лідер перевіряв роботу нового члена команди. Зробивши висновки, лідер давав настанови члену команди, що працював у парі для того щоб той виніс висновки та підвищив свою ефективність при роботі у парі у майбутньому. Потім лідер проводить бесіду з новим членом команди, надихає, координує та мотивує його.

На даний момент проект вже успішно завершено, але скоро постане необхідність у атестації членів команди – тому було вирішено використати удосконалену модель «Атестація», яка повинна вирішити проблемо ефективного збору та аналізу даних про підвищення рівня кваліфікації членів команди.

3.3 Аналіз апробації.

Проаналізувавши результати успішно завершеного проекту, можна провести аналіз. На практиці удосконалені моделі показали себе дуже добре і дозволили ефективно сформувати та розвивати команду на протязі розробки програмного забезпечення. З самого початку була підібрана ідеальна група розробників. На відміну від старих проектів, коли з самого початку підбирались групи виключно враховуючи кваліфікацію співробітника та його досвід у подібних проектах. Було вирішено десяток виробничих конфліктів,

раніше це робилося хаотично, що давало нормальну якість коду у місцях де потрібна висока якість. Було вирішено два особистих конфлікти, один із яких дуже вдало, конфлікт зам'явся. Інший було також вирішено використовуючи удосконалену модель, але працівника, який був джерелом конфлікту довелося спочатку ізолювати від команди, а згодом, після того як він зробив передачу інформації по проекту видалено, а на його місце майже одразу знайшли гарного командного гравця. Раніше такі випадки або замовчувались, або лідер команду намагався замаскувати та приховати конфлікт, що дуже погано відображалось на стосунках та комунікаціях у команді.

Нові моделі дозволили реалізувати проект вчасно, зробивши сам процес максимально приємним і комфортним для усіх членів команди. Залишились у плюсі усі хто мав причетність до проекту. Замовник – вдалий продукт, керівництво – хороший відгук від замовника, гроші і ще один вдалий проект у портфолію, команда – отримала задоволення від реалізації проекту, хороший відгук від керівництва, бонуси від компанії та нових друзів, а керівник проекту окрім усіх благ описаних вище, довів ефективність впровадження удосконалених методів управління командою в проектах розробки програмного забезпечення.

РОЗДІЛ.4.

ОХОРОНА ПРАЦІ

4.1 Аналіз шкідливих та небезпечних факторів на робочому місці програміста

Однією із характерних особливостей сучасного розвитку суспільства є зростання сфер діяльності людини, в яких використовуються інформаційні технології. Широке розповсюдження отримали персональні комп'ютери. Однак їх використання загостило проблеми збереження власного та суспільного здоров'я, вимагає вдосконалення існуючих та розробки нових підходів до організації робочих місць, проведення профілактичних заходів для запобігання розвитку, негативних наслідків впливу ПК на здоров'я користувачів.

Шкідливі та небезпечні фактори діляться на:

- а) Фізичні;
- б) Хімічні;
- в) Біологічні;
- г) Психофізіологічні.

Для програмістів можна виділити наступні фактори, що негативно впливають на здоров'я людини:

- а) небезпечний фізичний фактор - ураження електричним струмом при взаємодії з токопроводимими частинами обладнання;
- б) шкідливі фізичні фактори:
 - 1) недостатня освітленість робочого місця;
 - 2) температура зовнішньої середовища;
 - 3) шум;
- в) шкідливі психофізіологічні фактори.
 - 1) постійне напруження зору;
 - 2) постійне статичне навантаження;
 - 3) нервово-психічне перевантаження;

4) напружена зорова робота;

Освітлення - отримання, розподіл та використання світлової енергії для забезпечення нормальних умов праці. Таким чином, освітлення повинне бути достатнім, тобто відповідати вимогам санітарних норм, рівномірним, не повинно осліплювати очі та не створювати відблиски на робочій поверхні, за спектральним складом має наближатись до сонячного світла. Оптимальним вважається таке освітлення, при якому втома зору найменша. Недопустимою величиною освітлення вважається така, нижче якої відбувається порушення зорової функції.

Дія світла на очі людини викликає трансформацію енергії електромагнітного випромінювання у нервові імпульси, які по зоровому нерву передаються в головний мозок - до зорової зони кори великих півкуль.

Слід зазначити, що світло має надзвичайно важливе значення для життєдіяльності людини. Воно впливає на стан вищих психічних функцій та на фізіологічні процеси в організмі людини. Відомо, що нормальне освітлення тонізуюче діє, викликає гарний настрій, покращує діяльність вищої нервової системи людини.

Рациональне влаштоване освітлення виробничих приміщень позитивно впливає на працівників, підвищує ефективність та безпеку праці, знижує втому та травматизм, забезпечує високу працездатність.

Санітарні та гігієнічні норми щодо виробничого освітлення, котрі базуються на психологічних особливостях сприйняття світла та його вплив на організм людини, можуть бути зведені до наступного:

а) Спектральний склад світла, який забезпечується електричним джерелом, повинен бути близьким до сонячного;

б) Рівень освітлення повинен бути достатнім і відповідати гігієнічним нормам, які враховують зорові роботи;

в) Освітлення повинно забезпечувати рівномірність та стабільність світла у приміщенні, щоб запобігти частій переадаптації та втоми зору.

Гігієнічними нормами вимагається максимально використовувати природне освітлення, оскільки денне світло краще сприймається органами зору.

Штучне освітлення у виробничих та адміністративно-громадських приміщеннях має здійснюватись системою загального рівномірного освітлення, допускають застосування системи комбінованого освітлення.

Значення освітленості на поверхні робочого столу в зоні розміщення документів має становити 300-500лк. Як джерела світла для штучного освітлення мають застосовувати переважно люмінесцентні лампи типу ЛБ.

Шум, як фізичне явище, являє собою хвильові коливання матеріальних тіл - твердих, газоподібних або рідких. Виникнення звукових відчуттів людини пов'язане з коливаннями повітря.

Шум - один з найнесприятливіших факторів на підприємствах, що знижує працездатність робітників, їх уважність і створює передумови для виробничого травматизму та професійних захворювань.

Шум, як фізіологічне явище, являє собою несприятливий фактор для людини, викликає гіпертонію, глухоту, розлад центральної нервової системи, злоякісні пухлини тощо.

Інтенсивність шуму та вібрації є тими шкідливими факторами виробничого середовища, що негативно впливають не лише на слух, але і на нервову, серцево-судинну та інші системи організму. Судинні розлади та захворювання нервової системи в більшій мірі обмежують працездатність людини, ніж зниження слуху.

Виходячи з гігієнічних нормативів, витікає, що знижувати треба не будь-яку віброакустичну активність машин, а тільки ту, яка шкідливо впливає на людський організм, тобто за рівнем вище за гранично допустимий норматив. Таким нормативом є гранично допустимий спектр шуму та вібрації. Саме граничний спектр може бути взятий за «умовний віброакустичний нуль» (УВАН). УВАН - такий рівень віброакустичної активності машини і процесу, який забезпечує віброакустичну нешкідливість і безпеку.

Усунення промислового шуму може бути досягнуто лише в результаті проведення комплексу заходів, особливо в галузі вдосконалення технологічних процесів і обладнання. При цьому, в першу чергу, повинні бути усунені найсильніші джерела шуму. Знешумлення окремої одиниці обладнання в загальному технологічному процесі при великій кількості комп'ютерів, що продовжують випромінювати шум, недоцільно. Необхідне знешумлення тих комп'ютерів, шумові характеристики яких виходять за рамки вимог санітарних норм.

Допустимі рівні звуку, еквівалентні рівні звуку і рівні звукового тиску в октавних смугах частот мають відповідати вимогам СН 3222-85, ГОСТ 12.1.003-85, ГР 2411-81. Вони наведені в таблиці 4.1.

Таблиця 4.1
Допустимі рівні звуку

Вид трудової діяльності	Рівні звукового тиску в дБ октавних смугах із середньо геометричними частотами, Гц									
	31,5	63	125	250	500	1000	2000	4000	8000	Рівні звуку, еквівалентні і рівні звуку, дБа/дБАекв
Програмісти ЕОМ	86	71	61	54	49	45	42	40	38	50
Оператори в залах обробки інформації на ЕОМ ті оператори комп'ютерного набору	96	83	74	68	63	60	57	55	54	65
В приміщеннях для розташування шумних агрегатів ЕОМ	103	91	83	77	73	70	68	66	64	75

Для нормальної життєдіяльності людини важливим є забезпечення нормальних метеорологічних умов у виробничих приміщеннях, котрі суттєво впливають на самопочуття людини.

В поняття «метеорологічні умови», або «мікроклімат», виробничих приміщень входять ті фізичні фактори виробничого середовища, які впливають на тепловий стан організму і котрі необхідно постійно контролювати. Це температура, вологість, швидкість руху повітря, барометричний тиск і теплове випромінювання. Від стану виробничого приміщення залежить самопочуття і здоров'я людини.

Якщо кількість кисню в повітрі зменшиться до 12% то утруднюється дихання. В таких умовах людина напружує дихальний апарат, дихає частіше; такий стан людина витримує до 0,5 години.

Перегрівання організму відбувається за умов надлишкового конвективного випромінювання тепла нагрітих поверхонь. Розрізняють 2 фази перегрівання.

Перша - фізіологічна, при котрій вступають в активну реакцію пристосувальні функції організму. При цьому активізується робота серцево-судинної та дихальної систем, відбувається інтенсивне потовиділення, котре сягає 5л за зміну. З потім втрачається велика кількість мінеральних солей та вітамінів.

Охолодження організму можливе найчастіше в зимовий і перехідний періоди року при виконанні робіт на відкритому повітрі, або робіт, які проводяться в неопалюваних виробничих і складських приміщеннях.

Тривала дія низької температури може викликати різні небажані зміни в організмі людини. Охолодження пов'язане зі значними тепловиділеннями через кінцівки рук і ніг. На охолодження впливає вологість і швидкість руху повітря.

Вологість повітря визначається ступенем його насиченості водяною парою.

Вологість повітря суттєво впливає на терморегуляцію людського організму. Підвищення відносної вологості повітря у виробничому приміщенні ускладнює терморегуляцію, зменшення тепловиділення організмом. Фізіологічно оптимальною є відносна вологість в межах 40..60%. Норми мікроклімату для приміщень з ЕОМ представлені в таблиці 4.2

Таблиця 4.1

Норми мікроклімату для приміщень з ЕОМ

Пора року	Категорія робіт	Температура повітря , С, не більше	Відносна вологість повітря, %	Швидкість руху повітря, м/с
Холодна	Легка – Іа	22-24	40-60	0,1
	Легка – Іб	21-23	40-60	0,1
Тепла	Легка – Іа	23-25	40-60	0,1
	Легка – Іб	22-24	40-60	0,2

Робота працівників невиробничої сфери характеризується тривалою багатогодинною працею в одноманітному напруженому положенні, малою руховою активністю при значних локальних динамічних навантаженнях.

Робоче положення «сидячи» супроводжується статичним навантаженням значної кількості м'язів ніг, плечей, шиї та рук, що дуже втомлює. М'язи перебувають довгий час у скороченому стані і не розслаблюються, що погіршує кровообіг. В результаті, виникають больові відчуття в руках, шиї, верхній частині ніг, спині та плечових суглобах. Внаслідок динамічного навантаження на кістково-м'язовий апарат кистей рук виникають больові відчуття різної сили в суглобах та м'язах кистей рук; оніміння та уповільнена рухливість пальців; судом м'язів кисті; ниючий біль в ділянці зап'ястя.

У результаті, виникають локальні м'язові перенапруження, хронічні розтягнення м'язів травматичного характеру, що можуть викликати професійні захворювання: дисоціативні моторні розлади, захворювання периферійної нервової та кістково-м'язової систем.

Крім того, робота «сидячи» призводить до зниження м'язової активності - гіподинамії. За браком рухів відбувається зниження споживаного кисню

тканинами організму, сповільнюється обмін речовин. Це сприяє розвитку атеросклерозу, ожиріння, може стати причиною дистрофії ' міокарда, хронічного головного болю, запаморочення, безсоння, роздратування.

Трудова діяльність працівників невиробничої сфери належить до категорії робіт, які пов'язані з використанням великих обсягів інформації, із застосуванням комп'ютерних робочих місць, із частим прийняттям відповідних рішень в умовах дефіциту часу, безпосереднім контактом із людьми різних типів темпераменту тощо. Це зумовлює високий рівень нервово-психічного перевантаження, знижує функціональну активність центральної нервової системи, призводить до розладів в її діяльності, розвитку втоми, перевтоми, стресу.

Тривала робота на комп'ютеризованому робочому місці призводить до значного навантаження на всі елементи зорової системи і зумовлює втому та перевтому зорового аналізатора. Напружена зорова робота викликає «очні» (біль, печія та різь в очах почервоніння повік та очей, ломота у надбрівній частині тощо) та «зорові» (пелена перед очима, подвоєння предметів, мерехтіння, швидка втома під час зорової роботи) порушення органів зору, що може викликати головний біль, посилення нервово-психічного напруження, зниження працездатності.

4.2 Система штучного освітлення приміщення.

Освітлення відіграє важливу роль у житті людини. Біля 90% інформації сприймається через зоровий канал, тому правильно виконане раціональне освітлення має важливе значення для виконання всіх видів робіт. Раціональне освітлення повинно відповідати таким умовам: бути достатнім (відповідним нормі і: рівномірним; не утворювати тіней на робочій поверхні; не засліплювати працюючого; напрямок світлового потоку повинен відповідати зручному виконанню робіт.

За своєю природою світло - це видиме випромінювання електромагнітних хвиль довжиною від 380 до 780нм. Випромінювання вище

780нм називають інфрачервоним, нижче - ультрафіолетовим. Залежно від джерела світла виробниче освітлення може бути 3-х видів:

а) природне - пряме або відбите світло сонця (небосхила), що освітлює приміщення через світлові прорізи в зовнішніх відгороджувачих конструкціях.

б) штучне - здійснюється штучними джерелами світла і призначене для освітлення приміщень у темні години доби, або таких приміщень, які не мають природного освітлення.

в) сполучене - одночасне поєднання природного і штучного освітлення.

4.3.Основні положення техніки безпеки під час експлуатації ЕОМ.

До роботи з персональними ЕОМ і зовнішніми пристроями ЕОМ допускаються особи, які пройшли медичний огляд, вступний інструктаж і навчання на робочому місці, перевірку знання по охорони праці.

Оператори і користувачі ЕОМ зобов'язані:

1) Дотримуватись правил внутрішнього розпорядку
2) Знати і дотримуватися правил з охорони праці в обсязі виконуваних вимозі.

3) Виконувати тільки ту роботу, яка визначена інструкцією: з експлуатації устаткування та посадовими інструкціями, затвердженими адміністрацією підприємства, і за умови, що безпечні способи її виконання добре відомі

4) Знати й уміти надавати першу медичну допомогу при ураження електричним струмом і при інших нещасних випадках

5) Дотримуватися інструкції про заходи пожежної безпеки.

Якщо не виконувати профілактичні заходи і не дотримуватися режиму роботи, робота з ЕОМ, як правило, супроводжується зоровим і загальним перевтомою.

Організація робочого місця:

1) Приміщення повинно мати природне і штучне освітлення.

2) Площа на одне робоче місце з ЕОМ для дорослих користувачів повинна бути не менше 6,0 кв.м, обсяг - не менш 20,0 куб.м.

3) У виробничих та адміністративно-громадських приміщеннях у випадках переважної роботи з документами допускається застосування системи комбінованого освітлення.

4) Освітленість на поверхні столу в зоні розміщення робочого та документа повинна бути 300-500 лк.

5) Робочі місця по відношенню до світлових отворів повинні розташовуватися так, щоб природне світло падало з боку переважно сторінки. З метою профілактики перевтоми і перенапруження при роботі з дисплеями необхідно виконувати під час регламентованих перерв комплекси спеціальних вправ (вони наведені нижче).

6) Схеми розміщення робочих місць з ЕОМ повинні враховувати відстані між робочими столами з відеомоніторами (в напрямі тилу поверхні одного відеомонітора і екраном іншого), яка повинна бути не менше 2,0 м відстань між бічними поверхнями відеомонітора не менше 1,2 м. 7) Робочий стіл повинен регулюватися по висоті в межах 680-760 мм, за відсутності такої можливості його висота має складати 1600х900 мм. Під стільницею робочого столу має бути вільний простір для ніг з розмірами по висоті не менше 600 мм, по ширині 500 мм, по глибині 650 мм.

7) Розмір екрана повинен бути не менше 31 см (14 дюймів) по діагоналі, при цьому відстань від очей до екрану повинна бути в межах 40-80 см.

8) Робочий стіл (крісло) повинен бути підйомно-поворотним і регульованим по висоті та кутах нахилу сидіння і спинки, а також - відстані спинки від переднього краю сидіння. Конструкція його повинна забезпечувати:

а) ширину і глибину поверхні сидіння не менше 400 мм;

б) поверхню сидіння з заокругленим верхнім краєм;

в) регулювання висоти поверхні сидіння у межах 400-550 мм і кутом нахилу вперед на 15 град., і тому до 5 град.;

г) висоту опорної поверхні спинки 300 плюс-мінус 20 мм, ширину не менше 380 мм та радіус кривизни горизонтальної площині - 400 мм;

д) кут нахилу спинки у вертикальній площині в межах $0 + / - 30$ град.;

е) врегулювання відстані спинки від переднього краю сидіння в межах 250-400 мм;

є) стаціонарні або знімні підлокітники довжиною не менше 250 мм: і шириною 50-70 мм; регулювання підлокітників по висоті над сидінням у межах $230 + / - 30$ мм і внутрішнього відстані: між підлокітниками в межах 350-500 мм.

Робоче місце має бути обладнане підставкою для ніг, що має ширину не менше 300 мм, глибину не менше 400 мм, регулювання по висоті в межах до 150 мм і кутом нахилу опорної поверхні підставки до 20 град. Поверхня підставки має бути рифленою і мати по передньому краю бортик висотою 10 мм.

Робоче місце з ЕОМ повинне бути оснащене легко переміщуваним пюпітром для документів.

Клавіатуру слід розташовувати на поверхні столу на відстані 100-300 мм від краю, зверненого до користувача, або на спеціальною регульованою по висоті робочої поверхні, відокремленої від основний стільниці. Про кожний нещасний випадок на виробництві потерпілий або очевидець негайно сповіщає безпосереднього керівника.

За невиконання даної інструкції винні притягуються до відповідальності згідно з правилами внутрішнього трудового розпорядку або стягненням, визначених Кодексом законів про працю України.

РОЗДІЛ 5.

ОХОРОНА НАВКОЛИШНЬОГО СЕРЕДОВИЩА

5.1. Загальні питання охорони навколишнього середовища

Запобігання екологічних ризиків - пріоритетний напрямок внутрішньої і зовнішньої політики держав. Починаючи з 60-х років XX століття, із загостренням проблем навколишнього природного середовища, виокремлюються такий напрямок державної політики, як «екологічна політика».

Держави диференціюють екологічну політику з прив'язкою до окремих галузей: зміна клімату, підвищення ефективності використання природних ресурсів, охорона біорізноманіття тощо. Національна екологічна політика будується на уніфікованому підході до організації природоохоронних заходів, але у різних країнах зберігаються свої особливості.

За способом організації екологічної політики виокремлюють дві групи країн: перша - західноєвропейські країни і Японія; друга - США і Канада. У країнах першої групи основний акцент при реалізації екологічної політики здійснюється на систему законів. У країнах другої групи для обґрунтування політичних заходів основна увага приділяється різним програмам, планам, рекомендаціям [1]

Основним нормативно-правовим актом України в сфері охорони НС є закон України «Про охорону навколишнього середовища», уведений в дію Постановою Верховної Ради № 1268-12 від 26.06.1991 року. Верховною Радою України прийняті також закони, кодекси та інші нормативно-правові акти, метою яких є регулювання процесу використання та охорони природних ресурсів. Насамперед, це закони України «Про охорону атмосферного повітря» (1992), «Про природно-заповідний фонд» (1992), «Про забезпечення санітарного та епідеміологічного благополуччя населення» (1994); «Про використання ядерної енергії та радіаційну безпеку» (1995), «Про екологічну експертизу» (1995), «Про відходи» (1998), «Про рослинний світ» (1999), «Про

зону надзвичайної екологічної ситуації» (2000), «Про тваринний світ» (2001), «Про питну воду та питне водопостачання» (2002), «Про основи національної безпеки України» (2003); кодекси «Про надра» (1994), «Лісовий кодекс України» (1994), «Водний кодекс України» (1994), «Земельний кодекс України» (1994) тощо.

5.2. Вплив комп'ютеризації на навколишнє середовище.

Важко уявити сьогоденне життя без персонального комп'ютера (ПК), електроніки, мобільного телефону. ПК всюди - вдома, на роботі, школі, вузі, аптеці, банку, на вокзалі, магазині, лікарні і т.д. Інформатизація суспільства в повному обсязі. Бази даних, величезні потоки різної інформації, Internet - звичні слова і поняття. Кількість персональних комп'ютерів зростає з кожним роком.

На сьогоднішній день (за різними оцінками фахівців), обсяг випущених в світі ПК становить 1 мільярд. Багато це чи мало? Це коли у Вас ПК всього один-це мало, а уявіть цілий комп'ютерний клас. Це звичайно багато. Якщо врахувати, що кожна школа має комп'ютерний клас і в місті цих шкіл 15 (це близько 375 комп'ютерів), то цілком можна уявити який обсяг все це займає. Це навіть без урахування муніципальних організацій та комерційних фірм. Парк персональних комп'ютерів в місті величезний. Думаю, що помилюся не набагато, якщо назву цифру кількості ПК в місті це більше 3000 штук. Це дуже багато для одного міста.

Але будь-яка техніка стрімко застаріває, їй на зміну приходять нові, більш потужні, більш сучасні ПК та оргтехніка. Людство, хоче воно того чи ні, втягнуте в постійний процес модернізації та заміни електронної техніки. Ми радіємо новим моделям персональної техніки з новими можливостями. Поступово виникає проблема: а що робити зі старою технікою, морально застарілою або з тих чи інших причин, що вийшла з ладу, яка захащує підсобні приміщення та склади.

За період з 1991 року по теперішній час в Україну завезено різними постачальниками близько 10 млн. одиниць (близько 400000 т.) персональної й оргтехніки (це за найскромнішими підрахунками), мобільних телефонів - 37-40 млн. шт. (Близько 4800 т.). І це приблизні дані. Точного підрахунку ніхто не проводить.

Стурбованість громадськості проблемами екології, а також нові, більш жорсткі закони по захисту навколишнього середовища змушують великих виробників устаткування створювати мережі по збору вийшла з обігу техніки 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. Так, наприклад, у Західній Європі, де екологічні закони дуже суворі, компанія Hewlett-Packard створила вельми значну інфраструктуру зі збору та переробки застарілих комп'ютерів та оргтехніки. Всього в Європі продукцію HP переробляють 30 заводів, один з яких знаходиться в Росії. Справедливості заради потрібно відзначити, що ці заводи не є власністю HP. Вони належать партнерським компаніям американського вендора, які беруть участь у програмі утилізації списаної техніки.

Вся оргтехніка включає в свій склад як органічні складові (пластик різних видів, матеріали на основі полівінілхлориду, фенолформальдегіда), так і майже повний набір металів.

Всі ці компоненти не є небезпечними в процесі експлуатації виробу. Однак ситуація докорінно змінюється, коли виріб потрапляє на звалище. Такі метали, як свинець, сурма, ртуть, кадмій, миш'як входять до складу електронних компонентів переходять під впливом зовнішніх умов у органічні і розчинні сполуки і стають найсильнішими отрутами. Утилізація пластиків, що містять ароматичні вуглеводні, органічні хлорпохідні з'єднання є насущною проблемою екології Тому вся оргтехніка повинна утилізуватися за методикою затвердженою Державним комітетом України по телекомунікаціям. Завдяки комплексній системі утилізації оргтехніки

зводяться до мінімуму не перероблювані відходи, а основні матеріали (пластмаси, кольорові і чорні метали) і цінні компоненти (рідкісні метали, люмінофор, ферити і інші) повертаються у виробництво. Дорогоцінні метали, що містяться в електронних компонентах оргтехніки концентруються і після переробки на афінажних заводах здаються в Держфонд.

Майже у всіх комп'ютерах, електронній техніці вітчизняного або імпортного виробництва є деяка кількість золота, срібла та інших дорогоцінних металів. Це загальновідомий факт. А ось про те, що всі без винятку фірми повинні вести їх облік знають небагато. Тим не менш, за відсутність такого обліку можна поплатитися штрафом. Будь-яка організація зобов'язана документально оформляти надходження, рух, інвентаризацію та вибуття дорогоцінних металів, що містяться в складових частинах офісної техніки (комп'ютерів, телевізорів і т.д.).

Незважаючи на те, що ці нормативні акти ніхто не відміняв, вони якось випали з поля зору фірм, діяльність яких безпосередньо не пов'язана з дорогоцінними металами. Чим же обернеться фірмі відсутність обліку «дорогоцінних» запчастин? Відповідають фахівці Центральної державної інспекції пробірного нагляду Російської державної пробірної палати (саме цей держорган перевіряє, чи правильно фірми ведуть облік дорогоцінних металів). «Вести облік дорогоцінних металів, у тому числі що містяться в комп'ютерній техніці, зобов'язані всі фірми, - відповіли нам у інспекції. - Порушникам загрожує штраф за статтею 19.14. КоАП України.».

5.3. Переробка комп'ютерного обладнання

Комп'ютерами та оргтехнікою в нашій країні володіють фізичні та юридичні особи. Утилізація оргтехніки фізичними особами відбувається дуже прозаїчно - викинув на смітник у гіршому випадку, в кращому - розбирання на частини для подальшого застосування, що в кінцевому рахунку закінчується тією ж звалищем. Іноді просунуті керівники, щоб не зв'язуватися з утилізацією, просто дарують стару техніку своїм підшефним організаціям

(дитячі будинки, школи, заклади культури). Але сама проблема залишається. Цей процес в даний час не хвилює ні більшу частину населення, ні держава. Утилізацією в цьому випадку займаються шукачі кольорових і чорних металів. З юридичними особами все набагато складніше. Згідно із законодавством, персональні комп'ютери відносяться до основних засобів і підлягають бухгалтерському обліку на підприємстві із зазначенням кількості дорогоцінних металів, які в них містяться. Мало того, на цю техніку поширюється правило про амортизацію протягом 10 років.

Іншими словами, списувати і утилізувати дане обладнання можна тільки через 10 років. Однак ми знаємо, що персональна техніка морально застаріває набагато раніше. Це пов'язано зі стрімким розвитком електроніки і програмного забезпечення. У такому випадку юридична особа повинна звернутися до спеціалізованим підприємствам, які займаються ремонтом і обслуговуванням персональної техніки для проведення технічної експертизи обладнання з отриманням висновку про те, що персональна техніка морально застаріла, знята з виробництва, ремонтна база відсутня, підлягає списанню та утилізації в установленому порядку. Тільки після цього необхідно укласти договір з підприємством, яке займається роботою з відходами.

Як нам відомо, персональний комп'ютер містить усі види відходів. В даному випадку обладнання, що містить дорогоцінні метали, кольорові, чорні метали, полімери, здаються на підприємства, які мають ліцензію на роботу з дорогоцінними металами. Ці підприємства - переробники дорогоцінних металів, в свою чергу, укладають договори з підприємствами, які мають інші ліцензії і таким чином персональна техніка розбирається і утилізується згідно із законодавством.

В принципі, будь-який комп'ютер або телефон можна переробити і пустити у вторинне використання. При грамотній утилізації близько 95% відходів техніки здатні повернутися до нас в тому чи іншому вигляді, і приблизно 5% відправляються на звалища або федеральні заводи з переробки твердих побутових відходів.

Співвідношення ручного та автоматизованого праці на фабриках по переробці комп'ютерної техніки залежить від її типу. Для монітора це співвідношення приблизно 50 на 50 - розбирання старих кінескопів є досить трудомістким заняттям. Для системних блоків та оргтехніки частка автоматичних операцій вище.

НР вперше запропонувала переробку відслужили свій термін продукції ще в 1981 році. Сьогодні НР володіє інфраструктурою по збору й переробки використаних ПК та оргтехніки в 50 країнах світу. У рік утилізації піддається близько 2,5 млн. одиниць продукції. В одному тільки 2007 році НР переробив близько 100 тис. тонн списаного обладнання та витратних матеріалів, - майже в півтора рази більше, ніж роком раніше.

Перший етап завжди проводиться вручну. Це - видалення всіх небезпечних компонентів. У сучасних настільних ПК і принтерах таких компонентів практично немає. Але переробці піддаються, як правило, комп'ютери і техніка, випущені в кінці 90-х - початку 2000-х років, коли плоских рідкокристалічних моніторів просто не існувало. А в кінескопних моніторах міститься чимало сполук свинцю. Інша категорія продукції, що містить небезпечні елементи, - ноутбуки. В акумуляторах і екранах застарілих моделей є певна кількість ртуті, яка також дуже небезпечна для організму. Важливо відзначити, що в нових моделях ноутбуків від цих шкідливих компонентів позбулися.

Потім видаляються всі великі пластикові частини. У більшості випадків ця операція також здійснюється вручну. Пластик сортується в залежності від типу і подрібнюється для того, щоб надалі його можна було використовувати повторно. Що залишилися після розбирання частини відправляють у великий подрібнювач-шредер, і всі подальші операції автоматизовані. Багато в чому технології переробки запозичені з гірничої справи - приблизно таким же способом витягують цінні метали з породи.

Подрібнені в гранули залишки комп'ютерів піддаються сортуванню. Спочатку за допомогою магнітів витягуються всі залізні частини. Потім

приступають до виділення кольорових металів, яких в ПК значно більше. Алюміній добувають з брухту за допомогою електролізу. У сухому залишку виходить суміш пластику і міді. Мідь виділяють способом флотації - гранули поміщають в спеціальну рідину, пластик спливає, а мідь залишається на дні. Сама ця рідина не отруйна, проте, робочі на заводі використовують захист органів дихання - щоб не вдихати пил.

У даний момент можна з упевненістю сказати про те, що закон про утилізацію комп'ютерів необхідний, так як в дійсності ми не повинні негативно впливати на навколишнє нас середовище. Але в нашій країні все набагато простіше, існує дуже багато компаній, які із задоволенням приймуть замовлення на утилізацію старого устаткування, вам не доведеться тривалий час проводити за документами, які необхідні для утилізації, ваші комп'ютери зможуть забрати компанії, що займаються утилізацією. Все що від вас буде потрібно - це правильно оформлене списання, пам'ятайте, що списання комп'ютерів можна без особливих складнощів здійснити за допомогою спеціальних компаній, а також складання всього одного документа на утилізацію оргтехніки. Вартість таких послуг не велика, і ви безумовно зможете дозволити собі зробити навколишній світ чистішим.

ВИСНОВОК

Багато компаній займаються розробкою програмного забезпечення, гостро стоїть питання оптимізації праці. Розробка дуже часто буває неефективною з багатьох причин, наприклад у компаніях застосовуються «важковагові» методології, але ієрархічні методи управління не дають потрібного результату.

Також проблемою переходу є відсутність проінформованості про інші, не ієрархічні підходи, наприклад «гнучкі», і найголовніше, про способи впровадження «гнучких» методологій у технологічний процес.

Існує безліч різних методологій розробки програмних продуктів, розрахованих на великі та невеликі проекти, з різним складом команди розробників. Слід точно визначити, яка методологія підходить у даному конкретному випадку для цієї компанії.

В магістерській роботі проведено дослідження сучасних методологій формування команди розробників програмного забезпечення. Виявлені переваги та недоліки методологій : Waterfall Model, Agile, Incremental Model, Iterative Model та Incremental Model.

Розглянуті особливості використання методологія Agile, її принципи дозволяють у стислі терміни готувати для клієнтів нові продукти з урахуванням їхніх думок та реакцій, що передаються через зворотний зв'язок.

Сформовано команду проекту розробки програмного забезпечення в Agile простах з використанням удосконаленої моделі. Що дозволило реалізувати проект вчасно, зробивши сам процес максимально приємним і комфортним для усіх членів команди.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Вартамян А. А. Управление командой и организацией в бизнес-среде. - М.: Доброе слово, 2009.- 472 с.
2. Мазур И.И. Управление проектами: учебное пособие/ Мазур И.И., Шапиро В.Д., Ольдерогге И.Г.- М.:2-е издание, 2004.- 290 с.
3. Пособие по управлению проектами / Ю.Н. Забродин, А.М. Саруханов и др. – М., Омега-Л 2004.- 932 с.
4. Уэбстер, Г. Планирование и управление проектами для менеджеров:пер. с англ. Гордон Уэбстер. М.: Дело и Сервис, 2008. - 272 с.
5. Фоппель К. Создание команды. Психологические игры и упражнения. - М.: Генезис, 2009.- 484 с.
6. Николай Большаков Agile: что это такое и где используется, принципы и методология <https://www.calltouch.ru/blog/agile-cto-eto-takoe-i-gde-ispolzuetsya-principy-i-metodologiya/>
7. Agile Processes in Software Engineering and Extreme Programming. Lecture Notes in Business Information Processing: Springer International Publishing. – 2018. – № 314. –314 p.
8. Schmidt C. Agile Software Development Teams: the Impact of Agile Development on Team Performance / C. Schmidt // Progress in IS. – Springer. – 2016. – 184 p.
9. Гулага Я.С. Критерії оцінки якості в проектах, що використовують Agile / Я.С. Гулага, О.В. Мнушка // Комп'ютерні технології і мехатроніка. Збірник наукових праць за матеріалами міжнародної науково-практичної конференції. – Харків: ХНАДУ, 2019. – С. 82-85.
10. Кириленко Е.Г. Обоснование содержания обучения в рамках методологии преподавания профессионально-ориентированной дисциплины "Групповая динамика и коммуникация" / Е.Г. Кириленко, О.В. Лучшева // Інженерія програмного забезпечення. – 2010. – № 1. – С. 71-78.
11. Буч Ю. Навчальний курс "Групова динаміка і комунікації" в контексті підготовки сучасних фахівців у галузі інформаційних технологій / Ю. Буч //

- Вісник Львівського університету. Серія педагогічна. – 2017. – Вип. 32. – С. 511-519.
12. Савченко В.Н. Использование виртуализации и облачных технологий при обучении информационным технологиям / В.Н. Савченко, О.В. Мнушка, И.А. Сасимова // Нові технології навчання. – Київ: Інститут інноваційних технологій і змісту освіти МОН України. – 2016. – Вип. 88. – Частина 1. – С. 97-101.
13. Мнушка О.В. Аналіз використання хмарних технологій для формування компетенцій під час навчання в галузі інформаційних та комп'ютерних технологій / О.В. Мнушка // Вісник ХНАДУ. – Харків – 2017. – Вип. 76. – С. 123-127.
14. Computer Engineering Curricula 2016. Curriculum guidelines for undergraduate degree programs in computer engineering // IEEE Computer Society and Association for Computing Machinery. – 2016
15. Управління інноваційними проектами: навч. посібник / Уклад.: Н.Н. Пойда-Носик, І.І. Черленяк. Ужгород: Вид-во УжНУ «Говерла», 2017. 360 с.
16. Курс МВА по менеджменту: пер. с англ. / под ред. А.Р.Коэн . М.: Альпина Бизнес Букс, 2004. 507 с. С.197–205.
17. Бушуев С., Бушуева Н. Управление проектами: основы проф. знаний и система оценки компетентности проектных менеджеров (National Competence Baseline, NCB UA Version 3.1). Изд. 2-е. К.: ІРІДІУМ, 2010. 208 с.
18. Кон М. Agile: оценка и планирование проектов / Майк Кон; Пер. с англ. — М.: Альпина Пабlishер, 2018.
19. Введення в гнучку розробку програмного забезпечення. Частина 2. Маніфест і принципи гнучкої розробки - <http://www.kv.by/index2008354201.htm>
20. Закон України «Про охорону праці» від 14.10.92 у редакції 21.11.2002. <https://zakon.rada.gov.ua/cgi-bin/laws/main.cgi?nreg=2694-12>
21. Яскілка В.Я., Олійник М.З. Конспект лекцій з курсу «Охорона праці в галузі» /Укладачі:Тернопіль: Вид-во ТНТУ імені Івана Пулюя, 2016.56 с.

22. Закон України «Про охорону навколишнього природного середовища». Документ 1264-ХІІ, чинний, поточна редакція 01.01.2022. <https://zakon.rada.gov.ua/laws/show/1264-12>
23. Конституція України. <https://zakon.rada.gov.ua/laws/show/888-09#Text>