

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ**

ЕКОНОМІЧНИЙ ФАКУЛЬТЕТ

Кафедра економічної кібернетики

**Кваліфікаційна робота
магістра**

на тему Моделі управління проектами розробки web-ресурсів

Виконав: студент 2 курсу, групи 8.0518-ек

спеціальності 051 Економіка

(код і назва спеціальності)

освітньої програми Економічна кібернетика

(код і назва освітньої програми)

І. О. Тонких

(ініціали та прізвище)

Керівник доцент кафедри к.е.н., доцент Чеверда С. С.

(посада, вчене звання, науковий ступінь, прізвище та ініціали)

Рецензент зав. кафедри, д.е.н., професор Максишко Н. К.

(посада, вчене звання, науковий ступінь, прізвище та ініціали)

Запоріжжя
2020

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ**

Факультет _____

Кафедра _____

Рівень вищої освіти _____

Освітня програма _____
(код та назва)

Спеціалізація _____
(код та назва)

ЗАТВЕРДЖУЮ

Завідувач кафедри _____
(підпис)

« _____ » _____ 20 _____ року

З А В Д А Н Н Я

НА КВАЛІФІКАЦІЙНУ РОБОТУ/ПРОЕКТ СТУДЕНТОВІ (СТУДЕНТЦІ)

Тонких Ірині Олександрівні

(прізвище, ім'я, по батькові)

1 Тема роботи (проекту)

Моделі управління проектами розробки web-ресурсів

керівник роботи _____ доцент.. Чеверда С. С. _____
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом ЗНУ від « _____ » _____ 20 _____ року № _____

2 Строк подання студентом роботи _____

3 Вихідні дані до роботи _____

4 Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) _____

5 Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) _____

РЕФЕРАТ

Кваліфікаційна робота магістра містить три розділи, 77 с., 20 рис., 4 табл., 1 додаток, 26 джерел.

Об'єкт дослідження – управління проектами розробки web-ресурсів.

Предмет дослідження – методи і моделі управління проектами розробки web-ресурсів.

Мета роботи – обґрунтування вибору моделі управління проектами web-розробки та її застосування для управління розробкою web-сайту.

Методи дослідження – порівняльний, логічний, математичний аналіз та синтез, класифікація.

В даній роботі активно досліджувались саме проблеми в управлінні проектами розробки web-ресурсів, так як, багато бізнес-фахівців недооцінюють важливість управління проектами. На передній план у них завжди виходять розвиток бізнесу і показники рентабельності, а управління часто взагалі не потрапляє в список пріоритетів. Більшість людей навіть не здогадуються, що відсутність налагодженої організації по управлінню проектами приносить їм збитки.

Також досліджено теоретичні засади розробки web-ресурсів, типи web-сайтів, сутність процесу управління розробки web-ресурсів. Розкрито основні проблеми, які пов'язані з управлінням проектів в українських ІТ-компаніях, та фактори, що негативно впливають на розвиток ІТ-індустрії в Україні. Проведено аналіз інформаційних систем які застосовуються для управління проектами розробки web-ресурсів, переваги та недоліки використання кожної з них. Застосовано систему ASANA та метод Kanban у процесі для підвищення гнучкості, збільшення залученості команди в процес розробки, покращення загального робочого процесу, планування ефективності управління проектами та якості кінцевого продукту – web-сайту та наведена кількісна оцінка ефективності управління проектом.

WEB-СТОРІНКА, WEB-СЕРВЕР, WEB-САЙТ, ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ, ПРОДЖЕКТ-МЕНЕДЖЕР, СИСТЕМИ УПРАВЛІННЯ ПРОЕКТАМИ, СПРИНТ, SCRUM, AGILE, KANBAN, CRM, ASANA, TRELLO, JIRA

SUMMARY

Master's qualification work consists of three sections, 77 p., 20 figures, 4 tables, 1 annex, 26 sources.

The object of the study is the project management of developing web-resources.

The subject of the study is the project management's methods and models of developing web-resources.

The purpose of the work is to substantiate the choice of the web-development project management model and its application for the management of the web-site development.

Research methods are comparative, logical, mathematical analysis and synthesis, classification.

This paper explores the problems of the project management of development web-resources, as many business professionals underestimate the importance of project management. Business development and profitability are always at the forefront, and management is often not included in the priority list at all. Most people do not even understand that the lack of a well-established project management organization is damaging to them.

Theoretical principles of web resource development, types of websites, the essence of the process of managing web resources development are also explored. The main problems related to project management in Ukrainian IT companies and factors that negatively affect the development of the IT industry in Ukraine are revealed. The information systems used to manage web development projects, the advantages and disadvantages of using each of them are analyzed. The ASANA system and the Kanban method in the process are used to increase flexibility, increase team engagement in the development process, improve overall workflow, plan project management effectiveness and end product quality, and quantify project management effectiveness.

WEB-PAGE, WEB-SERVER, WEB-SITE, PROJECT-MANAGER, PROJECT MANAGEMENT SYSTEMS, SPRINT SCRUM, AGILE, KANBAN, CRM, ASANA, TRELLO, JIRA

ПЕРЕЛІК СКОРОЧЕНЬ ТА НАУКОВИХ ПОЗНАК

ОПР – особа, що приймає рішення

ПМ – проєкт-менеджер

СУП – система управління проєктами

ІТ – інформаційні технології

ЗМІСТ

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

РЕФЕРАТ	4
SUMMARY	5
ПЕРЕЛІК СКОРОЧЕНЬ ТА НАУКОВИХ ПОЗНАК	6
ВСТУП	8
РОЗДІЛ 1 КОНЦЕПЦІЯ ТА ЗАГАЛЬНА ХАРАКТЕРИСТИКА УПРАВЛІННЯ ПРОЕКТАМИ РОЗРОБКИ WEB-РЕСУРСІВ	11
1.1 Теоретичні засади розробки web-ресурсів	11
1.2 Сутність та основні проблеми управління проектами розробки web- ресурсів в Україні.....	17
1.3 Аналіз сучасних досліджень в управлінні проектами розробки web- ресурсів.....	29
РОЗДІЛ 2_МЕТОДОЛОГІЯ УПРАВЛІННЯ ПРОЕКТАМИ РОЗРОБКИ WEB- РЕСУРСІВ	31
2.1 Класифікація та особливості моделей управління проектами розробки web-ресурсів.....	31
2.2 Сучасні системи управління проектами розробки web-ресурсів.....	42
2.3 Аналіз сучасних методів управління проектами розробки web-ресурсів .	53
РОЗДІЛ 3_ЗАСТОСУВАННЯ МОДЕЛЕЙ УПРАВЛІННЯ ПРОЕКТАМИ ПРИ РОЗРОБЦІ WEB-РЕСУРСУ	58
3.1 Постановка завдання по створенню динамічного web-сайту.....	58
3.2 Поетапна процедура створення динамічного web-сайту із застосуванням системи Asana та методології Kanban.....	62
3.3 Аналіз ефективності застосованого методу та підходу, а також якості кінцевого продукту	66
ВИСНОВКИ.....	70
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ.....	73
ДОДАТОК А.....	76
Інструменти управління проектами	76

ВСТУП

Проблема управління проектами в розробці web-ресурсів є актуальною проблемою в Україні на сьогодні, так як за досить невеликий проміжок часу сфера інформаційних технологій в Україні зі звичайної, малозначної галузі, перетворилась на один з основних рушіїв української економіки.

Автоматизація прийняття проектних рішень за допомогою відповідного математичного та програмного забезпечення дозволяє значно зменшити часові затрати на підготовку й планування проекту та кількість робочих годин працівників, залучених до його виконання. Вагомою перевагою автоматизації є також зменшення людського фактору у прийнятті рішень: зниження ризику помилки та значне зниження рівня суб'єктивності. Оскільки процес прийняття рішень за допомогою програмних засобів відбувається швидше, ніж з залученням людських ресурсів, терміни завершення проекту стають більш стислими, що є принциповим саме для web-проектів, які розробляються та розвиваються високими темпами, і для яких мінімізація часу на їх розробку є пріоритетною. Ефективне управління проектами є запорукою отримання високоякісного продукту, а високоякісний продукт в свою чергу – є престижем галузі IT-індустрії в Україні. Є багато різних підходів та методів управління проектами по розробці web-ресурсів які широко використовуються в багатьох сучасних українських компаніях, але так як дана галузь є відносно молодого, а підходи та методи управління проектами були запозичені з досвіду зарубіжних країн, то виникає питання про їх ефективність в управлінні проектами саме в українських IT-компаніях.

У напрямі дослідження та вдосконалення процесів, методів та засобів управління web-проектами активно проводяться як теоретичні дослідження, так і постійно розробляються сучасні практичні нормативи, стандарти, методики та технологічні засоби. Зокрема, серед теоретичних досліджень у цій галузі відомими є роботи таких зарубіжних фахівців як С. Беркун (Scott

Berkun), E. Верзу (Eric Verzuh), Т. Шмідт (Terry Schmidt), серед українських – С. Бушуєв, К. Кошкін, Д. Ланде та інші. Сучасні технології управління web-проектами представляють такі розробки як Сервіс TeamBridge, IPI.Manager, Redmine, Primavera Project Planner тощо. Активно займаються розробкою новітніх стандартів та методик управління web-проектами такі структури як Project Management Institute Inc., OASIS, ISO, ANSI, компанії Google, Microsoft, IBM. Ці чинники свідчать, що проблематика досліджень у галузі вдосконалення методів та засобів управління web-проектами є актуальною як з погляду теорії, так і у практичному застосуванні. с

Актуальною є потреба розроблення науково обґрунтованих методів та прикладних програмних засобів управління web-проектами, які враховують специфіку web-проектів як особливої категорії ІТ-проектів та забезпечують управління ними за умов невизначеності в проектних даних.

Метою кваліфікаційної роботи є обґрунтування вибору моделі управління проектами web-розробки та її застосування для управління розробкою web-сайту.

Досягнення поставленої мети зумовило необхідність вирішення наступних основних завдань:

проаналізувати методи та моделі управління проектами розробки web-ресурсів;

– проаналізувати основні проблеми, які пов'язані з управлінням проектами розробки web-ресурсів;

– провести аналіз інформаційних систем які застосовуються для управління проектами розробки web-ресурсів;

– розробити метод вибору стратегії управління web-проектом на основі проектних характеристик із врахуванням невизначеностей в них

– обґрунтувати економічну ефективність обраного методу управління проектами розробки web-ресурсів.

Об'єкт дослідження – управління проектами розробки web-ресурсів.

Предмет дослідження – методи і моделі управління проектами

розробки web-ресурсів.

Наукова новизна отриманих результатів полягає у вдосконаленні методу вибору стратегії управління web-проектом, який відрізняється від відомих застосуванням шаблонів стратегій, використанням динамічних пріоритетів та можливістю використання неповних та неточних проектних даних, що дало можливість формування проектних рішень з управління web-проектом за умов невизначеності та змінюваності характеристик проекту.

Методи дослідження. Для аналізу процесів прийняття рішень в управлінні комерційними web-проектами та класифікації факторів прийняття проектних рішень було використано теорію систем та системний аналіз. Для розроблення методів опрацювання невизначеностей у проектних характеристиках використано теорію множин, математичну логіку. Для вибору стратегії управління web-проектом було застосовано метод аналізу ієрархій.

РОЗДІЛ 1

КОНЦЕПЦІЯ ТА ЗАГАЛЬНА ХАРАКТЕРИСТИКА УПРАВЛІННЯ ПРОЕКТАМИ РОЗРОБКИ WEB-РЕСУРСІВ

1.1 Теоретичні засади розробки web-ресурсів

Для того, щоб краще розуміти з яким продуктом ми маємо справу, розглянемо такі поняття як: web-сторінка, web-сайт, web-сервер, проект, проджект-менеджер.

Web-сторінка (англ. web-page) – інформаційний ресурс, доступний у мережі World Wide Web, який можна переглянути у web-браузері. Зазвичай ця інформація записана у форматі HTML або XH.

Web-сайт (англ. website, місце, майданчик в Інтернеті), також сайт (англ. site, місце, майданчик) — сукупність web-сторінок, доступних в Інтернеті, які об'єднані як за змістом, так і навігаційно. Фізично сайт може розміщуватися як на одному, так і на кількох серверах.

Web-сервер – це підключений до Інтернету комп'ютер, який приймає запити на отримання певних даних, обробляє їх та видає результати, використовуючи протокол HTTP (Hyper Text Transfer Protocol – протокол передавання гіпертексту).

Сукупність web-сторінок, тематично пов'язаних між собою і розроблених як єдине ціле, називають web-сайтом [1].

Усі web-сайти прийнято розділяти на дві групи: статичні і динамічні.

Статичний сайт – це набір статичних HTML-сторінок, які пов'язані між собою посиланнями. Статичні HTML-сторінки створюються вручну, зберігаються і завантажуються на сервер, після чого при кожному зверненні до сайту представляються користувачеві в незмінному вигляді. Для того щоб внести зміни інформації на таких сторінках, необхідно вручну редагувати програмний код сторінки.

Статичні сайти мають як свої переваги, так і недоліки. До переваг статичних сайтів відносять такі:

- а) статичні сайти створюють мінімальне навантаження на сервер;
- б) статичні сайти швидко завантажуються;
- в) розробка статичних сайтів обходиться дешевше;
- г) простота перенесення на новий хостинг [2].

Серед недоліків статичних сайтів особливо виділяється складність оновлення сайту, внесення будь-яких змін. Управління сайтом неможливо без знань і умінь в області web-програмування це може спричинити за собою додаткові витрати при необхідності додавання нових матеріалів на сайт, нових розділів або категорій.

Динамічний сайт – це сайт, в якому передбачена можливість редагувати вміст сторінок сайту, без звернення до web-програмування. Відображення сторінок таких web-сайтів засноване на шаблонній структурі, в складі якої динамічне інформаційне наповнення. У більшості випадків інформаційне наповнення сторінок береться з бази даних. Під час запити сторінки користувачем відбувається вибірка з бази даних, яка вставляється в шаблон, утворюючи при цьому нову web-сторінку, і пересилається web-сервером в призначений для користувача web-оглядач (браузер), який і відображає її належним чином.

Можливість вносити зміни в усі сторінки надається тільки певній групі користувачів, таким як, адміністратори або привілейовані користувачі. В окремих випадках вносити зміни в контент web-сторінки допускаються і анонімні користувачі (наприклад, на форумах - додавання повідомлень).

На відміну від статичних, динамічні сайти набагато більш гнучкі в управлінні. Динамічні сайти можна розробляти «з нуля», вручну створюючи всі необхідні програмні коди, скрипти і т.д. Однак набагато частіше для створення динамічних сайтів використовуються спеціальні системи управління контентом – CMS. CMS дозволяють використовувати вже готові програмні модулі і компоненти, без необхідності кожного разу створювати їх

заново. На основі однієї CMS можна створити будь-яку кількість динамічних сайтів. У динамічних сайтах реалізовано поділ змісту і оформлення web-сторінок, що, в свою чергу, дозволяє оперативно змінювати інформацію на сайтах без необхідності змінювати програмні коди сторінок. Це є одним з найголовніших переваг динамічних сайтів. Крім перерахованих переваг, динамічні сайти мають і ряд недоліків. У порівнянні зі статичними сайтами, динамічні більш «великовагових», дають велике навантаження на сервер – отже, вони більш вимогливі до хостингу, ресурсів сервера. [2]

Проект – це обмежений часовими рамками процес, що має визначений початок та кінець, зазвичай обмежений датою, але також може обмежуватися фінансуванням або досягненням результатів, який здійснюється для реалізації унікальних цілей та завдань [3], зазвичай, щоб призвести до вигідних змін або створення доданої вартості. Тимчасова природа проектів контрастує з бізнесом (процесами) [4], які є повторюваною, постійною або частково постійною діяльністю з виробництва продуктів або послуг. На практиці, управління вищезазначеними двома системами часто різняться і таким чином вимагає розвитку окремих технічних навичок та використання розподіленого управління ними.

Проект завжди націлений на результат та має одну або декілька ключових цілей. Управління проектом по розробці web-ресурсів здійснюється повноважним керівництвом, менеджером, який закріплений за проектом і командою, та всіма іншими учасниками проекту, які виконують окремі специфічні види діяльності та процеси по проекту.

Головним завданням проектного управління є досягнення всіх цілей [5] та виконання завдань проекту, одночасно виконуючи зобов'язання щодо наперед визначених обмежень проекту [5]. Типовими обмеженнями є межі та зміст проекту, час, бюджет [5]. Другорядним завданням, але більш амбітним, є оптимізація, розподілення та інтеграція завдань, необхідних для досягнення наперед визначених цілей.

Проджект-менеджер (project-manager) – це фахівець, головним завданням якого є управління проектом в цілому: визначення пріоритетів, планування виконання поставлених завдань, контроль, комунікація між усіма учасниками процесу, а також оперативне вирішення проблем.

Керівник проекту в області інформаційних технологій відповідає за досягнення цілей проекту у рамках затверджених замовником бюджету і термінів. Здійснює управління проектами (планування, організацію виконання, контроль і аналіз відхилень) в області створення і/або введення в експлуатацію інформаційних систем:

- забезпечує виявлення вимог до інформаційної системи (ІС);
- здійснює оцінку термінів і вартості проекту розробки, введення в експлуатацію і післяпроектного супроводу ІС;
- організовує роботи із забезпечення необхідного рівня якості;
- управляє людськими ресурсами проекту;
- організовує та здійснює комунікації між усіма зацікавленими особами проекту, включаючи замовника ІС, користувачів ІС, членів команди проекту;
- управляє ризиками проекту;
- управляє закупівлями, включаючи управління субпідрядниками;
- націлює роботу команди проекту на досягнення цілей проекту;
- забезпечує проведення здачі результатів проекту замовникові;
- бере участь у вдосконаленні системи управління проектами.

Об'єкти та засоби професійної діяльності: команда проекту, інформаційна система, опис змісту проекту, плани і графіки виконання робіт, кошториси та бюджети, інструментальні засоби управління проектами, стандарти в області управління проектами; стандарти та методи інформаційної взаємодії систем, стандарти та методи організації управління, обліку і звітності на підприємствах [6, 7].

Проджект-менеджер – це особа, що приймає рішення (ОПР) в загальній системі управління проектом. Саме від його рішень безпосередньо залежить якість кінцевого продукту. Завдання розробника - написати якісний, чистий

код. Розробники не завжди цікавляться чого хоче бізнес, їх головна ціль – виконати поставлені задачі. Головне завдання проджект-менеджера – із усього потоку інформації, яка надходить від клієнта, визначити основну ідею та ціль проекту та на основні визначених ідеї та цілі поставити задачі розробнику, коректувати їх при необхідності (в залежності від того, який метод управління проектами розробки ресурсів використовується) при цьому мінімізувати загальні витрати бюджету та людино-годин на даний проект.

Сорок відсотків усього часу роботи проджект-менеджера займає комунікація, безперервна та всебічна:

- а) з розробниками;
- б) з клієнтами;
- в) забезпечення комунікації між усіма учасниками процесу розробки сайту та контроль якості даної комунікації.

Планування, організація виконання, контроль і аналіз відхилень для ефективного досягнення цілей проекту в рамках затверджених замовником вимог, бюджету і термінів займає 35% усього часу роботи проджект-менеджера.

Документація займає 15% часу. Дуже важливо зберігати усю інформацію, яка поступає від замовника та інших учасників проекту: первинне технічне завдання, усі необхідні для роботи доступи та ресурси (доступи до сайту, серверу, файлів та бази даних), коректування завдань у процесі розробки сайту та нові завдання.

Утримання команди в робочому стані та мотивація команди займає 10%. І хоча ця діяльність займає у відсотках менше всього часу, але її важливість, мабуть, одна з найвищих. З часом, кожен з учасників проекту морально та фізично втомлюється від монотонної роботи, тому проджект-менеджер повинен вчасно відреагувати на це та оперативно вирішити проблеми.

Більш наочно складові частини роботи проджект-менеджера показано на рис. 1.1.



Рисунок 1.1 – Складові частини роботи проєкт-менеджера

Саме менеджер проєкту бере на себе важливу посередницьку функцію між замовником та співробітниками компанії. Він може одночасно мати глибоке розуміння процесів, які відбуваються у фірмі, при цьому постійно підтримувати зовнішні комунікації з існуючими та потенційними замовниками.

Функціональна модель [8, 9] за методологією IDEF0 описує web-проєкт як деякий керований процес (WP) (рис. 1.2), на вході якого подано ресурси проєкту (Resources), на виході отримано результат виконання проєкту – web-продукт (web-product), реалізацію процесу забезпечує виконавець проєкту – (Executant), керуючу дію процесу реалізує процес управління web-проєктом, який враховує керівні дії та проєктні ризики (Control/Risk).



Рисунок 1.2 – Функціональна модель web-проєкту

Зокрема для web-проекту, як деякого керованого процесу, характерними ознаками є такі.

Унікальність – набір проектних характеристик, вимог, способів виконання та управління для кожного web-проекту є специфічним і не повторюється в інших проектах.

Складність формалізації мети web-проекту обумовлена соціально-комунікаційними особливостями продукту який створюється в ході його реалізації, розділенням осіб замовника та споживача результатів виконання проекту.

Відсутність формальних критеріїв оптимальності прийняття рішень та процесу управління web-проектом, в цілому.

Динамічність процесу виконання web-проекту та управління ним, що відображається у зміні вимог, умов, проектних характеристик, стратегії управління, безпосередньо, в ході виконання проекту.

Неповнота, неточність, недостовірність та непостійність значень, які описують web-проект та забезпечують управління ним.

Наявність свободи вибору у процесах прийняття проектних рішень та вироблення керівних дій, обумовлена значним впливом людського чинника (Peopleware) у web-проектах [10,11].

1.2 Сутність та основні проблеми управління проектами розробки web-ресурсів в Україні

Основна проблема, яку виділяють більшість вітчизняних ІТ-компаній – недостатні комунікаційні навички інженерів, нерозуміння культурних відмінностей і вміння визначити, що представляє цінність для клієнта та його бізнесу. А адже саме від цього залежить конкурентоспроможність. Необхідно розвивати подібні якості за допомогою спеціальних тренінгів. Тут знову ж таки необхідна співпраця з вузами, оскільки в більшості технічних вузів не приділяють достатньо уваги для вивчення мовних відмінностей, специфічних

національних відмінностей, для вивчення ведення переговорів та особливо бізнес аналізу. У середньому, для того щоб фахівець вийшов на мінімальний рівень ефективності роботи, необхідно від трьох до шести місяців, в деяких випадках до року. А це вимагає від компаній інвестицій, котрі гальмують можливу співпрацю з закордонним замовником. Можна стверджувати, що по оплаті праці кваліфіковані програмісти в Україні поступово наздоганяють західних (з поправкою на вартість життя), і рівень їх доходів помітно вище, ніж інженерів в інших галузях української економіки. При грамотно побудованій на фірмі роботі проблема «текучості» кадрів не є критичною, тому що тримається в рамках допустимих значень за рахунок правильного управління знаннями, наявності методик професійного та кар'єрного розвитку співробітників. Що стосується відтоку кадрів за кордон, то на даний момент тут спостерігається затишшя. Він досить невисокий, особливо в порівнянні з «результатом» у 90-х роках, але ситуація може різко змінитися. Без державної підтримки ІТ-галузі та поліпшення економічної ситуації в країні нас чекає чергова хвиля еміграції. Дефіцит кадрів у країні великий і залежить від спеціальності та регіону. Це пов'язано з тим, що світова індустрія розробки ПЗ продовжує стрімко зростати, а кількість випускників ІТ-профілю в розвинених країнах знижується. І ні Індія, ні Китай не здатні на сьогоднішній день заповнити цей розрив, тому йде активний пошук інших альтернатив. Не вистачає перш за все професіоналів, тому якщо компанія знаходить людину з розумними амбіціями, то бере його на роботу, навіть не маючи поки для нього проекту. Крім пошуку фахівців на ринку, більшість компаній започатковують власні програми навчання у сфері тестування і розробки ПЗ. Для прикладу, компанія Елекс провела в Тернополі 3 місячний курс підготовки для .Net-програмістів. Співпраця з вузами також має місце, хоча, на жаль, поки вона не дуже ефективна. Основна проблема полягає в тому, що технології розвиваються надто швидко і сучасна система освіти не встигає міняти програми згідно потреб ІТ компаній, і відповідно останні, як правило, надають перевагу власному навчанню спеціалістів. Варто

зауважити, що перепідготовка кваліфікованого фахівця зазвичай займає від трьох до шести місяців, коли оволодіти новою технологією для випускника без практичного досвіду роботи, є задачею на рік і більше. У великих містах України програмісти отримують від \$ 500 до \$ 2000, що в 4-6 разів менше, ніж на Заході. У індійців доходи в середньому приблизно такі ж, як у нас. Щоб справитися з витоком кадрів, треба зробити роботу в компанії більш привабливою - постійно поліпшувати інфраструктуру, забезпечити кар'єрне планування, соціальні програми і, зрозуміло, підвищення зарплати відповідно до професійного росту працівників. Ще одною причиною нехватки кадрів є націленість вузів на підготовку програмістів та розробників, опускаючи при цьому потреба в програмістах – тестувальниках. При чому набір інструментів та методологія тестування змінюється набагато повільніше ніж технології для розробки. Як наслідок, підготовка спеціалістів-тестувальників програмного забезпечення лягає на компанію, котра потребує такого спеціаліста. Такий підхід обмежує динамічність компаній щодо приймання запропонованих проектів. Відчувається нестача також менеджерського складу, пов'язана з швидким зростанням компаній і появою великої кількості відповідних вакансій. Для такої спеціальності необхідні лідерські та комунікативні здібності, наполегливість та хороше знання іноземної мови, а таких спеціалістів вузи випускають значно менше, ніж запит проектів, які можуть бути виконані в Україні.

Крім вище перелічених проблем можна виділити проблему витоку кадрів в інші компанії та за кордон. Це не зовсім проблема для галузі. Проблема – це зрив проектів, невиконання зобов'язань перед клієнтами за відходу фахівців. Щоб уникнути зривів проектів під час виконання, всередині професійних співтовариств практикуються певні норми бізнес-етики. Тенденція еміграції спеціалістів за кордон зменшує приплив ІТ проектів в Україну в цілому. І хоч зарплата спеціалістів в ІТ галузі близька до світового рівня, враховуючи коефіцієнт витрат на проживання, проте професійна перспектива, та умови праці, особливо з точки зору юридичного захисту

стимулюють витік спеціалістів з країни. Залежно від досвіду і переліку технологій, якими володіє програміст, його робота може коштувати від 300 до 3 тис. євро на місяць. За кордоном ІТ-фахівці заробляють до 10 тис. На сьогоднішній день основне завдання – наситити вітчизняний ринок кваліфікованими кадрами, а для цього, крім підвищення якості технічної освіти в країні, істотних змін потребує наш кодекс законів про працю. Оскільки поки що компанії фактично не мають можливості укладати з співробітником юридично значущий контракт.

Не в усіх випадках, коли закордонний замовник зацікавлений у виконанні проекту українськими спеціалістами, є можливість підхопити проект, а саме через причини з боку самої компанії. Серед основних таких проблем можна виділити наступні.

Відсутність спеціалістів в технологіях, що вимагається замовником або рішенням проекту. Для вирішення цього ІТ компанія аналізує статистику запитів по технологіях і виділяє інвестиції на вивчення найбільш потенційних та перспективних технологій. Як правило, в кожній ІТ компанії є окремий спеціалізований відділ, котрий займається вивченням нових технологій (Research and Development department). Тоді при потребі консультації, проджект-менеджер запрошує спеціаліста з «R&D» відділу, що забезпечує можливість швидкого прийняття рішення щодо технології, котру можна використати для виконання проекту. Такий підхід дозволяє компанії вільно конкурувати у швидкій реакції на запити на нові проекти, – неодноразово тендер на проект вдавалось виграти тільки за рахунок швидкої пропозиції з рішенням, що базується на оптимальній технології. Таким чином, великі компанії мають суттєву перевагу при конкуренції за новий проект порівняно з маленькими новоствореними компаніями, котрі не можуть собі дозволити утримування «R&D» відділу. Як правило новостворені ІТ компанії спеціалізуються на обмеженому спектрі технологій і виживають тільки за рахунок пропозиції нижчої вартості виконання проекту.

Відсутність вільної команди з спеціалістами, що задовільняють вимогу кваліфікації. Для вирішення такої проблеми ІТ компанія змушена утримувати на «лавці» (Resource Pool) незадіяну команду. Такий підхід дозволяє конкурувати з тими компаніями, котрі практикують підхід, що базується на пошуку команди після запиту на виконання проекту. Досить часто замовник очікує старт проекту терміново, щоб встигнути в графік, який вимагається бізнесом. Кожен замовник це свого роду також підрядчик, якому ставлять обмеження по часу інвестори, власники, користувачі ІТ продукту і т.д. Тим більше не маючи вільної команди компанія не може впевнено заявити про можливість старту проекту, що суттєво знижує ймовірність передачі проекту цій компанії. Команду на «лавці» може собі дозволити тільки велика компанія, що знову ж таки дає їй перевагу щодо конкуренції за проект з новоствореною компанією. Таку вільну команду, як правило націлюють на вивчення технологій, про котрі йшла мова вище. Також підхід з командою на «лавці» дозволяє інтенсивно набирати нових працівників. Завжди є ризик, що новий працівник не справиться з поставленим завданням, і такий ризик покривається початковим залученням нового працівника до команди на «лавці», і перевіркою його можливостей на тестових завданнях, котрі не впливають критично на здачу комерційного проекту. Ще одною стороною команди на «лавці» є можливість моніторити працівників, з котрими важко працювати, оскільки таких працівників залучають в команду для проекту в останню чергу. І якщо є запас вибору, то з великою ймовірністю, такий працівник залишається на «лавці» довший період ніж інші, що дозволяє керівництву компанії не підняти питання про продуктивність, рівень зарплати та і взагалі, доцільність працевлаштування такого працівника. Варто зауважити, що у регіонах, де сконцентрована велика кількість ІТ компаній, і відповідно є достатньо наявних спеціалістів на ринку праці, ІТ компанія може використовувати підхід пошуку команди по потребі, не витрачаючи оплати праці на працівників на «лавці», але такий підхід вимагає високозлагодженого процесу відділів продаж, та рекрутингу. При цьому

працівників наймають на термін виконання проекту, а не на постійній основі. Як правило, працівники ІТ компаній, як і більшість, надають перевагу стабільному працевлаштуванню, і у випадку короткотривалого термінового залучення вимагають вищої зарплати ніж постійні працівники. Такий факт ставить важіль в протипагу утримування компанії на «лавці».

Вартість команди не покриває зарплату і адміністративні витрати компанії. В такі витрати входять вартість оренди приміщення, закупівля чи оренда обладнання, необхідні ліцензії, зарплата адміністративного складу – бухгалтерів, фінансистів, юристів, керівників адміністративних підрозділів, прибиральниць, охоронців, секретаріату, відділу роботи з персоналом, та інші витрати на запрошених спеціалістів, наприклад, вчителів іноземної мови. Кожна компанія на старті покриває вищезгадані потреби одним-двома працівниками, проте з ростом компанії витрати на адміністративний відділ можуть рости не лінійно. Тому у великих ІТ компаніях зважують на статистику витрат і приходів, та формують інструменти, котрі дозволяють вирахувати вартість проекту, закладаючи інформацію про вартість команди згідно зарплат працівників та інших відомих витрат по проекту. Такі інструменти вже мають закладені витрати на адміністративний департамент та інші відомі витрати з статистики виконаних проектів. Такий підхід дозволяє чітко вказати мінімальну вартість проекту за котру ІТ компанія зацікавлена його виконувати.

Замовник не має довіри до запропонованої команди чи компанії в цілому. Для вирішення такої проблеми компанія використовує ряд методів, котрі дають результат в сукупності. Для забезпечення успішного старту нового проекту чи виграшу тендера, ІТ компанії створюють спеціальний відділ, працівники котрого мають навички у написанні комерційних пропозицій з урахуванням національних відмінностей замовника, вказуючи методології виконання проектів, політики компанії в цілому, переконуючи замовника у надійності компанії, тощо. Також цей відділ як правило тісно працює з вище згаданим «R&D» відділом. На довіру особливо впливають

відгуки попередніх замовників. Тому основною задачею кожного проєкту є виконати проєкт так, щоб замовник був задоволений. Тут вимагається і високо методологічний менеджмент, і комунікативні здібності проєкт-менеджера, і наявність сильного бізнес-аналітика, котрий забезпечує виконання саме тієї функціональності, що потрібна замовнику. Щоб врахувати зауваження замовника, компанія просить замовника після виконання проєкту дати відгук. Щоб отримати більш відвертий відгук, компанія пропонує взяти участь у анонімному опитуванні по всіх клієнтах компанії. Після чого враховуються як негативні так і позитивні зауваження, для того, щоб врахувати їх в роботі з наступним замовником. Таким чином компанія може розраховувати на те що попередній чи поточний замовник надасть позитивний відгук потенційною новому замовнику.

Компанія не задовольняє вимоги замовника щодо інформаційної безпеки. Це і підписання документів про нерозголошення комерційної таємниці кожним членом команди, і обмежений доступ до робочих кімнат та відповідного обладнання, це також високорівневі налаштування доступів через інтернет та локальну мережу, використання ліцензійних програм, тощо. Для вирішення таких задач ІТ-компанії мають відповідно –відділ, котрий дбає про забезпечення та дотримання вимог інформаційного захисту та департамент системних адміністраторів, котрі забезпечують необхідне обладнання та налаштування. Крім проблем пов'язаних з стартом проєкту, не менш важливими є причини неуспішної здачі проєкту. Неуспішною задачею проєктів.

Хибне визначення потреб замовника. Це сама дорога помилка з усіх, що муть трапитись впродовж всіх стадій ІТ проєкту. Ціна коливається від потрачених зусиль на вибір технології і початок розробки, до вартості всього проєкту, враховуючи зусилля потрачені на намагання спасти проєкт на останній стадії. Для вирішення такої проблеми компанія має відділ спеціалістів по бізнес-аналізу, задача яких проаналізувати потребу замовника виходячи зі слів замовника про те, що він хоче (доволі часто замовник навіть

не завжди зрозуміло розказує що він хоче), переходячи до інформації що йому насправді потрібно – його бізнес потреба, котра дасть йому можливість заробити кошти. Крім залучення БА спеціаліста на старті проекту, сучасні компанії практикують залучення його впродовж всього життєвого циклу проекту. Повноцінний бізнес-аналіз, як правило, вимагає візиту БА в компанію замовника, проте такий підхід може собі дозволити не кожна команда та і не кожна ІТ- компанія. В такому разі вимагається інтенсивна голосова комунікація з замовником як на старті проекту так і впродовж його виконання. Одним з альтернативних профілактики БА помилок є застосування «agile» методології менеджменту ІТ проектів, котра передбачає короткотривалі «спринти» (функціональність, яку можна виконати за короткий термін, наприклад, тиждень), і обговорення результатів попереднього «спринту» та планів на наступний «спринт» безпосередньо з замовником (працівником замовника). Такий підхід дозволяє мінімізувати ціну помилки зарплатою команди не більше ніж за один чи декілька тижнів.

Технічні помилки. Наступною по ціні помилкою є вибір не вірного рішення в тому числі вибір не вірної технології. Щоб уникнути таких помилок, визначають чіткі критерії кінцевого результату, включаючи можливу потребу в розширенні функціональності, обговорення усіх можливих сценаріїв користувачів, середовище виконання, інтеграція зі сторонніми компонентами тощо. Типовою помилкою є спроба додати певну функціональність до вже існуючої системи без повноцінного аналізу можливих наслідків. На черговому етапі ускладнення системи, ядро може не витримати навантаження, або взаємодія компонентів стає настільки неосяжною, що не можливо проконтролювати забезпечення повноцінної якості. На старті проекту залучають висококваліфікованих спеціалістів, котрі і відповідають за вибір стратегії та використання технологій.

Проблеми з якістю. Одною з відомих причини, що призводить до втрати якості, є той факт. що етап забезпечення якості недооцінюють, а той зовсім не враховують. Програмісти- розробники здебільшого творчі люди, і в

процесі натхнення створення нового функціоналу можуть злегковажити з перевіркою всіх сценаріїв. Тому ІТ компанії залучають до процесу перевірки якості програмістів-тестувальників. Не знімаючи повністю відповідальності за якість з програміста-розробника, програмісти тестувальники фокусуються тільки на можливих причинах збою системи. Цей процес включає і аналіз поведінки системи з точки зору кінцевого користувача, і ручне тестування сценаріїв, і створення та використання інструментів для автоматизованого тестування системи, особливо перевірки поведінки системи при навантаженні (англ. Performance and Loading testing) Для забезпечення процесу відладки якості, використовують спеціальні системи, котрі дозволяють переглядати і оновлювати 34 статус декільком програмістам в паралель. Таким чином кожен член команди може переглянути, які завдання закінчені і вимагають тестування чи перетестування, які виявлені нові причини збою системи та відповідні завдання, пріоритетність завдань та котрому програмісту виставлене призначення займатись відповідним завданням.

Помилки менеджменту є ще одною типовою причиною зниження якості є тиск з боку термінів здачі проекту – як правило, у випадку затримок з розробкою певного функціоналу, потерпає саме якість. Вирішення таких проблем вимагає вмілого менеджменту при плануванні та виконанні проекту. Помилки менеджменту можуть бути критичними у випадку невиконання цілей проекту – вчасно, в рамках бюджету, та забезпечуючи необхідно функціональність. На практиці закладають певний запас в бюджеті та термінах виконання проекту зважаючи на ризики непередбачуваних обставин. Agile методи, про які більш детально ми поговоримо в 2-му розділі, дозволяють мінімізувати ризик непередбачуваних обставин, і відповідно не закладати в бюджет та терміни виконання додаткових годин. Як правило на проджект-менеджера покладається відповідальність за комунікації з замовником, і якщо рівень проджект-менеджера недостатній, то це може призвести до втрати замовника, тобто наступного проекту. Запит на новий

проект від існуючого замовника є яскравим критерієм задоволеності замовника. Проте для зменшення можливих проблем через комунікацію, сучасні ІТ компанії виділяють окремо спеціального працівника, задача котрого забезпечити спілкування між замовником і виконавцем. – Низька продуктивність команди. А якщо точніше, то в продуктивності команди , нижчій ніж очікувалось. Таке трапляється або при роботі з новою командою, або при зниженні мотивації до роботи. Тому ІТ-компанії надають перевагу працювати командам 35 відносно у стабільному складі, тобто, коли розширення і зменшення команди не відбивається на «ядрі команди». При роботі з новою командою практикується підхід закладання додаткових запасів в бюджет та терміни виконання або виділення невеликої короткотривалої ітерації з тривалості всього проекту та повідомлення замовника про можливі зміни контракту по закінченню цієї ітерації. Для забезпечення мотивації використовують різні підходи, – від підняття зарплати до залучення працівників тільки в проект, котрий їм цікавий. Дуже важливим для мотивації працівника є його відчуття приналежності і потрібності на проекті. Для цього кожного працівника команди залучають починаючи ще з стадії обговорення проекту, покладаючи на кожного працівника відповідальності за частину роботи, проводячи тренінги та збори, на котрих підкреслюється цінність всієї команди та неможливість виконати проект її частиною.

Проблеми з оплатою проекту. Не зважаючи на високо розвинуту світову юридичну систему, залишається ризик несплати або неповної оплати за виконану роботу. Для вирішення таких проблем ІТ компанії практикують запит на щомісячну оплату по факту затрачених зусиль (англ. Time and Materials Contract), або пропорційну частину у випадку домовленої загальної вартості проекту (англ. Fixed Price Contract). В будь-якому з вищевказаних випадків, залишається ризик не отримати оплату за місяць, і у випадку підозрілих замовників практикують навіть запит авансу.

Проблеми державного рівня та рівень заробітної плати розробників проджект-менеджер, на жаль, не може вирішити, але підвищити ефективність процесу розробки web-сайту, комунікаційну складову, якість поставлених завдань та чіткість цілей – це в його компетенції. Як зазначалося вище, від роботи проджект-менеджера залежить якість кінцевого продукту, а високоякісний продукт в свою чергу – є престижем галузі IT-індустрії в Україні та запорукою конкурентоспроможності та кваліфікації програмістів та розробників.

Вага та значення людського чинника є однією з характерних особливостей в управлінні web-проектами. Це обумовлено, насамперед тим, що кінцевим результатом виконання web-проекту є створення web-продукту, який реалізується через створення інформаційного web-ресурсу чи web-сервісу у мережі Інтернет. Важливим елементом web-продукту, який створюється в межах певного проекту є його візуальне подання та сприйняття користувачем. Оскільки кінцевим споживачем такого продукту завжди є людина, процес його створення супроводжується впливом низки додаткових ергономічних, психологічних, соціальних, культурних, естетичних та інших чинників, а також особливостями цільової аудиторії.

Як наслідок, соціально-комунікаційні особливості результатів виконання web-проектів, обумовлюють значення впливу людського чинника на рівні дій замовника і виконавця. проекту На цьому рівні основним є формування спільного бачення результатів виконання web-проекту та однозначного розуміння вимог до цього, що досягається шляхом взаємодії відповідальних осіб з обох сторін, погодженням позицій та формування набору узгодженого показників web-проекту, які в подальшому складають основу для вибору стратегії управління та процесів прийняття проектних рішень. Загалом, значна вага людського чинника у web-проектах ускладнює процеси управління, насамперед, через :

- складність формалізації базових показників та параметрів проекту,

- виникнення неоднозначностей, неточностей, невизначеностей у проектних даних, і, як наслідок зниження рівня їх достовірності,
- часту зміну суттєвих проектних показників безпосередньо в ході виконання web-проекту,
- ускладнення процесів прийняття проектних рішень, та зниження їх якості, можливості та ефективності застосування CASE-засобів для управління web-проектами.

Внаслідок цього правильне врахування ролі, ваги та результатів впливу людського чинника є одним з основних завдань управління web-проектом.

Особливості та роль людського чинника (peopleware) в ІТ-проектах досліджують, зокрема, в своїх працях Скотт Беркун (Scott Berkun)[12], Ларрі Константін (Larry Constantine) [13, 14], Том Демарко (Tom DeMarco) [15] та інші. Основною концепцією цих авторів є неможливість ефективної реалізації будь-якого ІТ-проекту, загалом, (а web-проектів, як специфічної категорії, зокрема,) без усестороннього аналізу та врахування впливу людського чинника і його наслідків. У [16, 17] визначено основні сфери впливу людського чинника на хід виконання ІТ-проекту і управління ним. Серед них, найбільш актуальними в управлінні web-проектами є такі:

- якість та продуктивність виконання проекту,
- взаємодія виконавця і замовника проекту,
- робота команди виконавців,
- прийняття проектних рішень,
- управління виконанням проектних дій,
- організаційні завдання,
- розроблення людино-машинних інтерфейсів,
- документування та впровадження результатів тощо.

Як наслідок впливу людського чинника у web-проектах виникають такі явища:

- неточність, неповнота та недостовірність проектних даних та показників web-проекту;

- можливість зміни проектних показників в ході виконання в web-проекту;
- виникнення неоднозначностей і суперечностей у проектних даних;
- ризику виникнення помилок в ході виконання в web-проекту;
- можливість прийняття некоректних чи неефективних рішень з управління web-проектом;
- порушення термінів, вимог, обмежень, умов фінансування тощо;
- погіршення якості кінцевого продукту;
- як наслідок, ризик зриву проектних домовленостей, досягнення кінцевої бізнес-цілі чи виконання web-проекту загалом.

Отже, вплив людського чинника у різних сферах управління web-проектами та його наслідки обумовлюють потребу і доцільність вироблення спеціальних методичних та практичних вирішень. Застосування додаткових спеціальних методів і засобів для визначення та формалізації результатів людського впливу та прийняття проектних рішень, за таких умов, створить для можливості ефективного управління web-проектом із врахуванням наслідків впливу людського чинника.

Отже, сутність управління проектами в розробці web-ресурсів полягає у мінімізації та подальшому подоланні вищезазначених проблем.

1.3 Аналіз сучасних досліджень в управлінні проектами розробки web-ресурсів

Інформаційні технології відіграють все більш важливу роль в управлінні проектами. Масштаби і складність його проектів зросли, що вимагає чудової управління проектами, щоб в повній мірі реалізувати вигоди від нього. На жаль, ІТ проекти стали горезвісними через високі показники відмов або мають значні витрати або перевитрата бюджету. Дослідження і досвід показує, що багато ІТ-проекти намагаються досягти функціональності і якості цілей. Дослідження

виявили численні причини цих проблем в проектах, таких як ескалація проектів, погане управління ризиками, нездатність управляти очікуваннями користувачів, погана розробка програмного забезпечення або процеси управління проектами, або нездатність вчитися на минулих помилках і успіхах. Професіонали з управління проектами стикаються з багатьма труднощами при успішному управлінні проектами. Крім того, виникає необхідність в тому, щоб дослідники і практики в повній мірі розуміли потенціал сучасних методів для успішного управління проектами. Дана стаття ознайомить читачів з кількома новими науковими дослідженнями за сучасними методиками для успішного його управління проектами.

У напрямі дослідження та вдосконалення процесів, методів та засобів управління web-проектами активно провадяться як теоретичні дослідження, так і постійно розробляються сучасні практичні нормативи, стандарти, методика та технологічні засоби. Зокрема, серед теоретичних досліджень у цій галузі відомими є роботи таких зарубіжних фахівців як, зокрема С. Беркун [] Ю.І. Буріменка [10], І. І. Пучкова [11], Ю.Апелло [12], J. Sutherland та М. К. Schwaber [13] та інші.

Сучасні технології управління -проектами представляють такі розробки як Сервіс TeamBridge, IPI.Manager, Redmine, Primavera Project Planner тощо. Активно займаються розробкою новітніх стандартів та методик управління web-проектами такі структури як Project Management Institute Inc., OASIS, ISO, ANSI, компанії Google, Microsoft, IBM. Ці чинники свідчать, що проблематика досліджень у галузі вдосконалення методів та засобів управління web-проектами є актуальною як з погляду теорії, так і у практичному застосуванні.

РОЗДІЛ 2

МЕТОДОЛОГІЯ УПРАВЛІННЯ ПРОЕКТАМИ РОЗРОБКИ WEB-РЕСУРСІВ

2.1 Класифікація та особливості моделей управління проектами розробки web-ресурсів

Принципова модель управління проектами, що дає вичерпне уявлення про сукупності і взаємозв'язки базових понять, показана на рис. 2.1 та рис. 2.2.

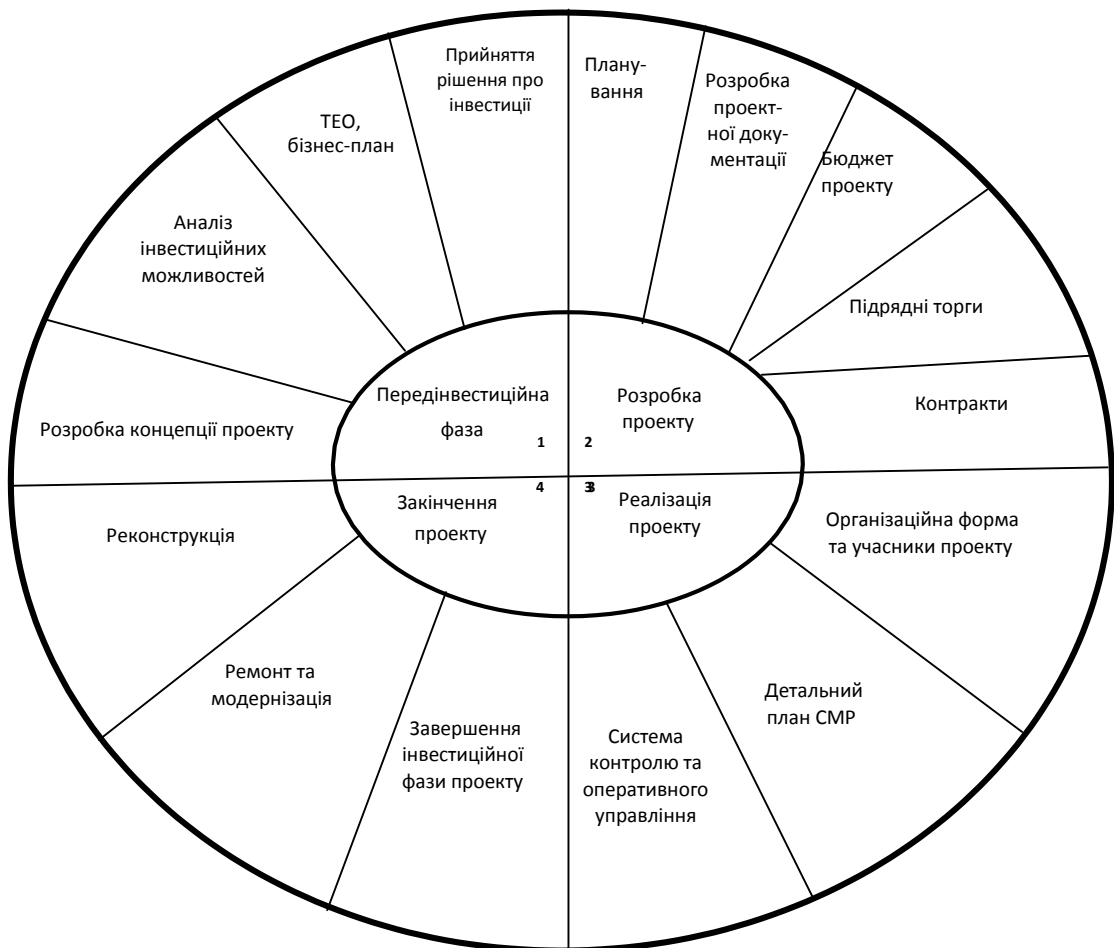


Рисунок 2.1 – Фази життєвого циклу проекту [14]

Кожна фаза, як правило, розбивається на окремі роботи, щоб забезпечити найкращу керованість. Більшість таких робіт відносяться до основного продукту фази й назва фази відповідає назві свого основного

продукту. Більшість фаз життєвого циклу проекту мають наступні схожі характеристики:

- вартість і кількість учасників на старті невеликі, але зростають наприкінці й різко зменшуються перед завершенням проекту;
- імовірність успішного завершення проекту на старті найменша, але зростає в міру реалізації проекту;
- здатність замовника вплинути на результати й вартість проекту найвища на старті й зменшується у процесі виконання проекту, оскільки наприкінці вартість внесення змін і виправлення помилок значно зростає.

Універсального підходу до поділу проекту на фази не існує. Керівники проектів розбивають життєвий цикл проекту на етапи по-різному. Найбільш традиційним є виділення у проекті чотирьох загальних фаз життєвого циклу проекту: ініціація, планування, впровадження і контроль, завершення. Частіше виділяють дві:

- фаза планування;
- фаза реалізації.

Всесвітній банк більш ґрунтовно підходить до деталізації життєвого циклу проекту і розрізняє такі етапи:

- створення ідеї, концепції.
- визначення завдань;
- розробка;
- робоче проектування;
- експертиза;
- переговори;
- ухвалення рішення про надання кредиту;
- упровадження проекту;
- контроль за реалізацією;
- і аналіз і оцінка результатів;
- завершення проекту.

Життєвий цикл проекту															
Передінвестиційна (Pre-Investment) фаза				Інвестиційна Investment, Impelementation, Execution) фаза				Експлуатаційна (Execution, Operation) фаза							
Аналіз інвестиційних можливостей	Попереднє ТЕО (Pre-feasibility study)	ТЕО (Feasibility study)	Доповідь про інвестиційні можливості (Appraisal)	Планування (Planning)	Організація фінансування	Переговори та укладання контрактів (Negotiations)	Проектування (Design)	Будівництво (Construction)	Маркетинг (Marketing)	Навчання (Training)	Приймання і запуск	Виробництво (Production)	Заміна обладнання	Розширення, модернізація, інновація (Expansion, Innovation)	Закриття проекту (Completion)

Рисунок 2.2 – Принципова модель управління проектом [14]

На рис. 2.2 зображено життєвий цикл проекту з основними етапами проектної діяльності. За класифікацією UNIDO виділяють три фази ЖЦП: передінвестиційна, інвестиційна та експлуатаційна, які, в свою чергу, розгалужуються на стадії:

1. Передінвестиційна фаза: аналіз інвестиційних можливостей, попереднє техніко-економічне обґрунтування.
2. Інвестиційна фаза включає переговори і укладання контрактів, проектування, будівництво, маркетинг, навчання.
3. Експлуатаційна фаза включає приймання і запуск, заміну обладнання, розширення та інновації.

Відповідно до пропозиції інституту створення і управління проектами прийнято виділяти 5 основних фаз: –

- формування концепції;
- розроблення комерційної пропозиції;
- проектування;
- виготовлення;

– задачу об'єкта і завершення проекту.

У виробничій практиці часто перші дві фази об'єднують в одну «Початкову фазу проекту». Формально фази проекту включають стадії. Стадії проекту складаються з етапів. Етапи проекту включають певні види робіт (роботи). Повна структуризація «Фаза – стадія – етап – робота» не обов'язкова. Усе визначається специфікою проекту. Головне – забезпечення найкращої керованості.

Початкова фаза або концепція. Головний зміст робіт – розроблення концепції проекту, яка включає збір початкових даних і аналіз існуючого стану, попередні дослідження. Виявлення потреб у змінах проекту, визначення проекту, яке включає, у свою чергу: цілі, завдання, результати, основні вимоги, обмежувальні умови, критерії, рівень ризику, оточення проекту, потенційних учасників, необхідний час, ресурси, кошти та ін. Визначення і порівняльна характеристика альтернатив. Подання пропозицій, їх випробування і експертиза, затвердження концепції і отримання схвалення для наступної фази розроблення.

Фаза розроблення – розроблення основних компонентів проекту і підготовка до його реалізації. Загальний зміст робіт: - призначення керівника проекту і формування команди проекту; - установлення ділових контактів, установлення вимог замовника і власника проекту, ключових учасників; - розвиток концепції та основний зміст проекту: кінцеві результати, стандарти якості, структура проекту, основні роботи, необхідні ресурси, структурне планування, у т. ч. декомпозиція проекту, календарні плани, збільшені графіки, кошторис і бюджет проекту, потреба в ресурсах, розподіл позовів. Організація проведення торгів, укладання субконтрактів. Організація виконання базових проектів і дослідно-конструкторських робіт за проектом, подання проекту, отримання ухвали на продовження робіт.

Фаза реалізації проекту – виконання основних робіт з досягнення основних цілей проекту. Основні роботи цієї фази:

а) організація проведення торгів і укладання контрактів;

- б) уведення в дію системи управління проектом;
- в) організація виконання робіт;
- г) уведення в дію засобів і способів комунікації учасників проекту;
- д) уведення в дію системи мотивації і стимулювання команди проекту;
- е) детальне проектування і технічна специфікація;
- ж) оперативне планування робіт;
- з) установа системи інформаційного контролю за ходом робіт;
- и) організація і управління матеріально-технічним забезпеченням робіт;
- к) виконання робіт, передбачених проектом;
- л) керівництво, координація робіт, узгодження темпів, моніторинг прогресу, прогноз стану, оперативний контроль, регулювання основних показників проекту;
- м) розв'язання проблем, що виникли, і задач [15].

4. Фаза завершення проекту – у цій фазі досягаються кінцеві цілі проекту, підбиття підсумків вирішення конфліктів і закриття проекту. Основний зміст робіт у цій фазі:

- планування процесу завершення;
- експлуатаційне випробування продукту;
- підготовка кадрів для експлуатації відповідного об'єкта;
- підготовка документації;
- здавання об'єкта замовнику; б) уведення в експлуатацію;
- оцінка результатів проекту і підведення підсумків;
- підготовка підсумкових документів;
- закриття робіт і проектів;
- вирішення конфліктних ситуацій;
- реалізація ресурсів, що залишилися;
- накопичення фактичних і дослідних даних для подальших проектів;

– розформування команди проекту.

Взаємозв'язок фаз, функцій і підсистем управління проектами зображено на рис. 2.3 [10].

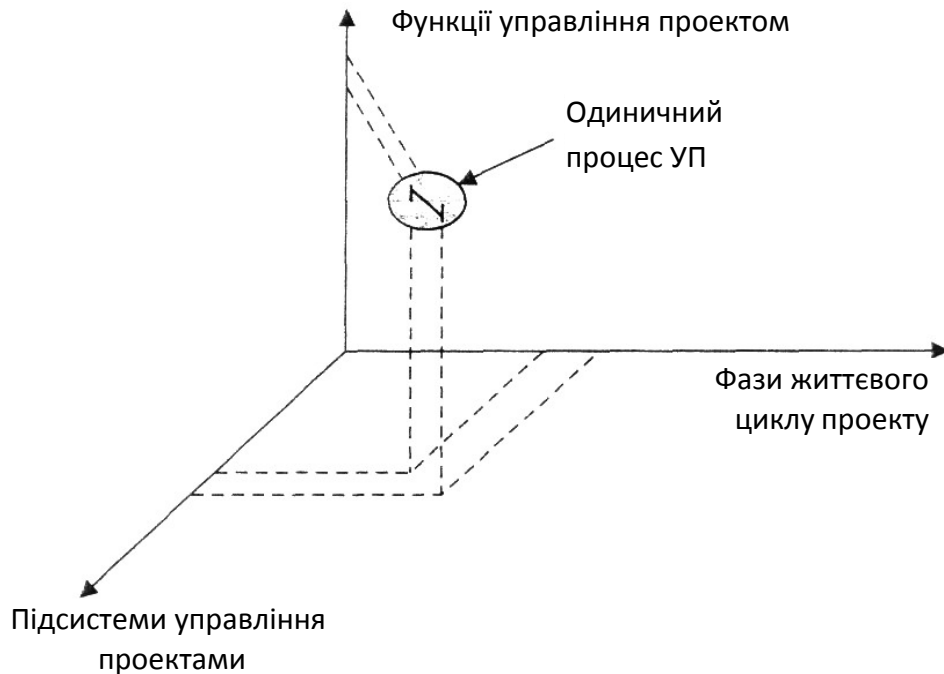


Рисунок 2.3 – Взаємозв'язок внутрішніх та зовнішніх компонентів проекту [14]

З рис. 2.3 видно, що проект функціонує в певному оточенні, що включає внутрішні і зовнішні компоненти, які враховують економічні, політичні, соціальні, технологічні, нормативні, культурні та інші фактори. Проект завжди спрямований на результат, на досягнення певних цілей, на певну предметну область. Реалізація проекту здійснюється повноважним керівництвом проекту, менеджером проекту і командою проекту, яка працює під цим керівництвом, іншими учасниками проекту, які виконують окремі специфічні види діяльності, процеси за проектом. У роботах за проектом, як правило, на умовах часткової зайнятості, можуть брати участь представники лінійних і функціональних підрозділів компаній, відповідальних за виконання покладених на них завдань, видів діяльності, функцій, включаючи

планування, керівництво, контроль, організацію, адміністрування та інші загальні функції. Керованими параметрами проекту є:

- обсяги робіт і види робіт за проектом;
- вартість, витрати, витрати за проектом;
- тимчасові параметри, що включають терміни, тривалості та резерви виконання робіт, етапів, фаз проекту, а також взаємозв'язок робіт;
- ресурси, необхідні для здійснення проекту, в тому числі: людські або трудові, фінансові, матеріально-технічні, що поділяються на будівельні матеріали, машини, обладнання, комплектуючі вироби та деталі, а також обмеження за ресурсами;
- якість проектних рішень, застосовуваних ресурсів, компонентів проекту та ін. Проект і процес його реалізації є складною системою, в якій сам проект виступає як керована підсистема, а керуючою підсистемою є управління проектом.

Управління проектом являє собою методологію організації, планування, керівництва, координації людських і матеріальних ресурсів протягом життєвого циклу проекту (кажуть також проектного циклу), спрямовану на ефективне досягнення його цілей шляхом застосування системи сучасних методів, техніки і технологій управління для досягнення визначених у проекті результатів за складом і обсягом робіт, вартістю, часом, якістю.

Для ефективного управління проектами система повинна бути добре структурована. Суть структуризації (кажуть також декомпозиції) зводиться до розбиття проекту і системи його управління на:

- фази життєвого циклу проекту, етапи, роботи, завдання, поодинокі робочі процеси;
- окремі пакети робіт, пов'язані між собою у структуру робіт за проектом;
- організаційну структуру виконавців за проектом;

- структуру розподілу відповідальності й обов'язків виконавців при виконанні робіт за проектом у вигляді матриці;
- підсистеми за областями управління проектом, що функціонують практично протягом усіх фаз управління проектом;
- загальні системні функції, що виконуються на всіх фазах реалізації проекту і в усіх підсистемах.

Основною структурною одиницею учасників проекту є команда проекту – спеціальна група, яка стає самостійним учасником проекту (або входить до складу одного з цих учасників) і здійснює управління інвестиційним процесом в рамках проекту.

Реалізація проекту відбувається в рамках організаційної форми, структура якої значною мірою впливає на сам проект. Розрізняють:

- функціональну структуру. Як її різновид може застосовуватися дивізіональна форма організації управління, сформована за регіональною, продуктовою або технологічною ознаками;
- проектну структуру;
- матричну структуру.

Можуть бути виділені три її різновиди: слабка матриця, коли координатор проекту відповідає за координацію завдань за проектом, але має обмежену владу над ресурсами; збалансована матриця, коли менеджер проекту координує всі роботи і розподіляє відповідальність за досягнення мети з керівниками функціональних підрозділів; жорстка матриця, коли менеджер проекту має максимальні повноваження, але і несе повну відповідальність за виконання завдань проекту.

Життєвий цикл проекту (проміжок часу між моментом появи, зародження проекту і моментом його ліквідації, завершення) є вихідним поняттям для дослідження проблем фінансування робіт за проектом і прийняття відповідних рішень. Укрупнено життєвий цикл проекту можна поділити на три основні смислові фази: передінвестиційну, інвестиційну та експлуатаційну.

Подальше розбиття істотно залежить від специфіки проекту. Так, життєвий цикл може поділятися на чотири фази, а саме:

- концептуальна фаза, що включає формулювання цілей, аналіз інвестиційних можливостей, обґрунтування здійсненності (техніко-економічне обґрунтування) і планування проекту;
- фаза розробки проекту, що включає визначення структури робіт і виконавців, побудову календарних графіків робіт, бюджет проекту, розробку проектно-кошторисної документації, переговори й укладення контрактів з підрядниками і постачальниками;
- фаза виконання проекту, що включає роботи з реалізації проекту, включаючи будівництво, маркетинг, навчання персоналу;
- фаза завершення проекту, що включає в загальному випадку приймальні випробування, дослідну експлуатацію і здачу проекту в експлуатацію;
- експлуатаційна фаза, що включає: приймання і запуск, заміну обладнання, розширення, модернізацію, інновацію.

Функції управління проектом включають: планування, контроль, аналіз, прийняття рішень, складання і супровід бюджету проекту, організацію здійснення, моніторинг, оцінку, звітність, експертизу, перевірку і приймання, бухгалтерський облік, адміністрування.

Підсистеми управління проектом включають: управління вмістом і обсягами робіт, управління часом, тривалістю, управління вартістю, управління якістю, управління закупівлями та поставками, управління розподілом ресурсів, управління людськими ресурсами, управління ризиками, управління запасами ресурсів, інтеграційне управління, управління інформацією та комунікаціями.

Основу управління проектом складають процеси прийняття проектних рішень. Рішення, прийняте керівником проекту, визначає дії виконавців та порядок їх виконання на всіх етапах життєвого циклу проекту – від ініціювання та планування до завершення та впровадження результатів.

Аналіз змісту та характеру дій з управління проектом дає змогу розглядати його як процес взаємодії низки активних та пасивних сутностей. У процесі управління проектом визначено такі активні сутності (актори).

Замовник web-проекту, який в даному процесі ініціює проект, формулює мету проекту, забезпечує фінансовими та іншими ресурсами, визначає вимоги до результатів виконання проекту, впливає на умови реалізації проекту та управління ним і використовує результати виконання проекту для своїх інтересів.

Керівник проекту у процесі управління реалізує такі основні функції: в результаті взаємодії із замовником, отримує від нього необхідні для управління проектом дані, визначає базові показники проекту, способи, шляхи та порядок управління проектом, обирає стратегію управління, виробляє оперативні проектні рішення, забезпечує їх виконання та контроль результатів.

Активний об'єкт типу "Споживач web-продукту" в даному процесі виконує єдину функцію – отримує web-послуги або використовує інформаційний ресурс, утворені як результат виконання проекту, згідно з їх призначенням.

Окрім активних об'єктів, модель описує низку пасивних об'єктів, між якими відбувається взаємодія в межах процесу управління web-проектом. Такі об'єкти не виконують самостійних операцій, але безпосередньо або опосередковано впливають на процедури вироблення проектних рішень та їх реалізацію через сприйняття дій активних об'єктів та зв'язки з іншими пасивними об'єктами web-проекту. Пасивними об'єктами в управлінні web-проектом є наступні.

Цілі web-проекту формуються замовником web-проекту. Вони, в свою чергу, впливають на зміст стратегії управління web-проектом та на зміст самого web-проекту.

Ресурси web-проекту позначають ті фінансові, матеріальні та інші вкладення, які замовник виділяє на виконання web-проекту. Ресурси web-

проекту формує замовник, а вони впливають на такі об'єкти як можливості виконавця, стратегія управління web-проектом та проектні потреби.

«Вимоги web-проекту» є одним з найсуттєвіших об'єктів в процесі управління. Вимоги поділяють на проектні та бізнес-вимоги. Об'єкт типу "Вимоги web-проекту" впливає на об'єкт «Можливості виконавця», а також визначає властивості таких об'єктів як «Стратегія управління проектом», «Проектні потреби» та, власне, «web-проект».

Об'єкт «Проектні потреби» позначає набір вимог та потреб виконавця, необхідних для реалізації проекту. Його формує активний об'єкт «Керівник web-проекту», при цьому об'єкт «Проектні потреби» взаємодіє з іншими об'єктами, а саме отримує обмеження від об'єкту «Ресурси web- проекту» та визначається об'єктом «Вимоги web-проекту». Об'єкт «Проектні потреби», своєю чергою, впливає на об'єкт «Проектні рішення».

Об'єкт «Можливості виконавця» характеризує різного виду ресурси (фінансові, матеріальні, людські, технічні, часові тощо), які є в розпорядженні виконавця і можуть бути задіяні для реалізації проекту. На цей об'єкт існує вплив з боку таких об'єктів як «Вимоги web-проекту» і «Ресурси web-проекту», об'єкт оцінюється виконавцем впроекті та впливає на об'єкт «Стратегія управління web-проектом» і визначає обмеження для об'єкта «Проектні рішення».

Об'єкт «Стратегія управління web-проектом» позначає набір принципів, правил та умов, які визначає порядок вироблення та реалізації проектних рішень в управлінні проектом. Вплив на стратегію управління проектом мають такі об'єкти як «Цілі web-проекту», «Ресурси web-проекту», «Можливості виконавця», додатково визначення стратегії залежить від об'єкту «Вимоги web-проекту». Об'єкт «Стратегія управління web-проектом» безпосередньо впливає на об'єкт «Проектні рішення». Керівник проекту обирає одне із можливих значень стратегії управління. Вибір стратегії

управління є одним із ключових елементів організації процесу управління проектом в цілому.

Об'єкт «Проектні рішення» позначає директивні дії керівника проекту спрямовані на реалізацію окремих дій та проекту в цілому. Керівник проекту виробляє проектні рішення на основі обраної стратегії, Вплив на Вироблення проектних рішень при цьому мають об'єкти «Стратегія управління web-проектом» та проектні потреби, а з боку об'єкта «Можливості виконавця» підтримуються обмеження. За допомогою проектних рішень керівник регулює хід виконання та властивості проекту.

2.2 Сучасні системи управління проектами розробки web-ресурсів

Розглянемо основні системи управління проектами розробки web-ресурсів, які широко використовуються в сучасних українських ІТ-компаніях різного рівня.

ASANA – це універсальний та частково безкоштовний таск-менеджер. Це не просто список завдань, а ціла система дрібних інструментів, які допомагають замінити майже будь-який додаток по управлінських задачах, взаємозв'язків з клієнтами, проектами та багатьма іншими. Існує тільки англійська версія. Є мобільний та десктопний додаток, а також web-версія.

Якщо коротко – він допомагає не загубитися у стрімкому потоці нових завдань, збирати в одному місці всі необхідні для них файли та контролювати прогрес.

В ASANA зручно розділяти великі задачі на маленькі зрозумілі таски. СУП допомагає визначити наскільки завантажений робочий день усіх учасників проекту та відсортувати завдання у зручному порядку.

Вона буде корисною командам або особам з великою кількістю різних проектів та різнопланових завдань, коли записник вже не допомагає, а звичайні таск-менеджери виглядають малофункціональними.

Основні корисні функції:

- опис може містити посилання на інші завдання або інших користувачів (розумне автозаповнення з'являється після введення знака @);
- підзадачі немає нічого, що таке, що таке завдання, щоб зробити те, що доступно для кого виконується через родительську задачу;
- одна задача може взагалі не бути пов'язана з проектом (наприклад, дочірні завдання).

Дедлайном може бути як одна проста дата, так дата, яка неодноразово повторюється (при повторенні створюється така сама задача, тільки з дедлайном збільшенням на вказаний інтервал).

Задача призначається лише одній людині (з можливістю змін), але за процесом виконання завдання можна слідкувати і якщо задача буде перевизначена на іншого розробника, то він побачить усі зміни в задачі, які були до нього. Це дуже зручно як для розробника, так і для проджект-менеджера, тому що у процесі роботи досить часто первинні задачі підлягають певним змінам. Замовник вносить свої корективи про які необхідно негайно сповістити усіх учасників проекту.

Задачу можна призначити людині, яка не додана до робочого столу (проекту). Для цього важливо вказати поштову адресу людини, якій потрібно назначити задачу і після чого він отримає повідомлення з можливістю реєстрації та отримає доступ до даного робочого столу або завдання (в залежності від того, що ви вибрали при призначенні).

Якщо навіть людина не хоче зареєструватися, він просто може відповісти на дане письмо і його відповідь буде коментарем у даній задачі.

У коментарях також працює розумне автозаповнення.

Розглянемо більш детально інтерфейс програми на рис. 2.4 -2.6.

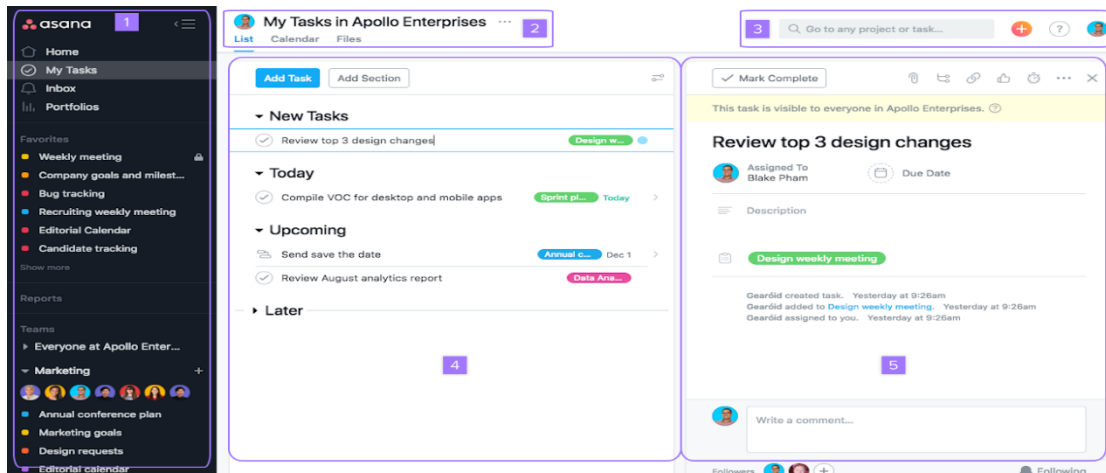


Рисунок 2.4 – Інтерфейс ASANA [15]

Інтерфейс ASANA розділений на 5 областей:

- а) бічна панель – доступ до головної сторінки, завдвнь, папки «Inbox», команди та проектів у вашому робочому просторі чи організації;
- б) заголовок (header) – містить дії та перегляди проекту, «My tasks» або «View», який зараз відкрит;
- в) верхня панель (top bar) – пошук, швидке додавання, налаштування профілю та робочої області чи налаштування організації;
- г) основна панель (main pane) – відображає список завдань, розмов, календаря, ходу чи файлів;
- д) права панель (right pane) – містить деталі завдання чи розмови.

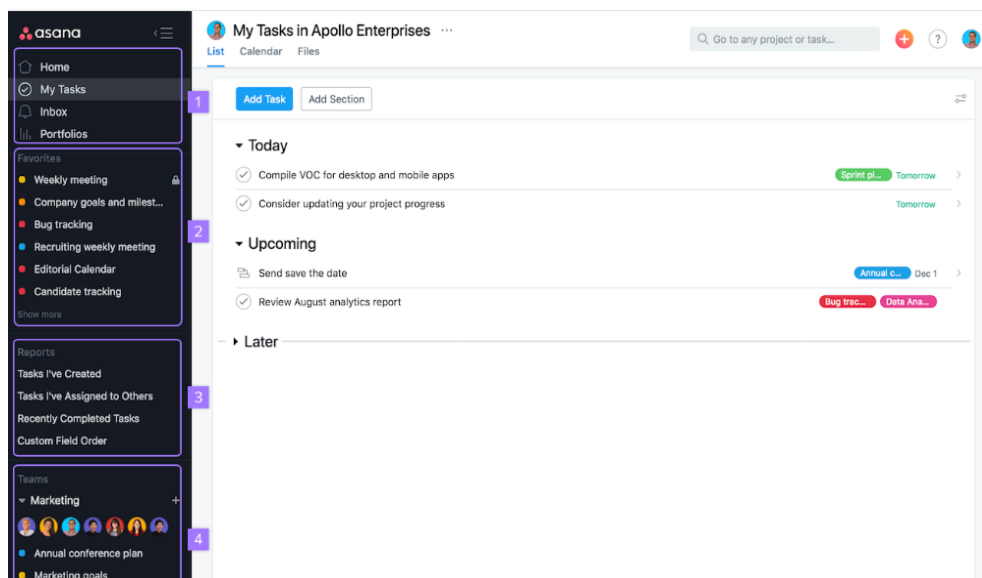


Рисунок 2.4 – Бічна панель ASANA з задачами та проектами [15]

На рис. 2.6 зображено вікно проекту зі списком усіх завдань, які в ньому є. Завдання можна редагувати, додавати нові, видаляти, залишати коментарі, призначати нових людей.

ASANA – досить нескладна програма з якою можна розібратись за день, але недоліком є те, що для великих компаній вона може бути недостатньо функціональною, а для малих компаній з великою кількістю невеликих проектів з часом програма стане загроможженою. Тому підходить вона для невеликих компаній з невеликою кількістю проектів або з великою кількістю, але при цьому необхідно буде чистити її від старих проектів для того щоб вона не стала громіздкою, що не досить добре, тому що тоді постає питання про збереження інформації про усі проекти.

Trello – сервіс для управління проектами та персональними завданнями. Простий, безкоштовний та гнучкий візуальний онлайн-інструмент для організації роботи команд чи відділу за методологією Kanban [16].

Trello – це дошка та колонки простими словами. Необмежена кількість дошок та колонок. Дошка – це сам проект, де задачі задачі систематизовані та розподілені по якимось спільним рисам(відділ, направлення, канал, робітник і т.д.). Колонки містять у собі картки, допомагають сортувати їх за прогресом або різним етапом. Їх можна використовувати для створення робочого процесу, в якому картки можна переміщати між колонками та архівувати (коли задача виконана). Архівувати необхідно завжди, інакше дошка та колонки переповняться картками.

Головні переваги, які зробили Trello популярним – це:

- а) простий інтерфейс (рис. 2.7, 2.8);
- б) майже необмежений безкоштовний доступ;
- в) зручність у роботі та можливість інтеграції з іншими популярними інструментами для онлайн-роботи.

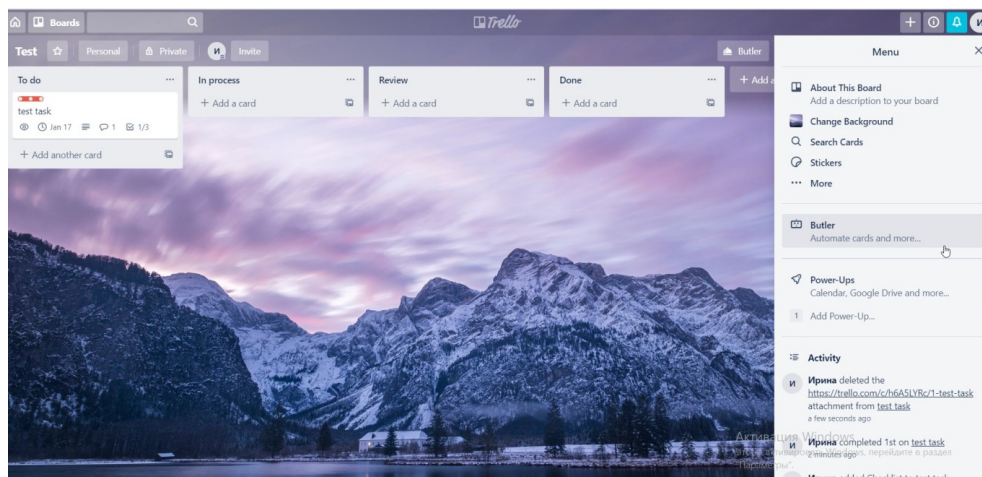


Рисунок 2.7 – Загальний вигляд Trello

Як бачимо з рис. 2.7 Trello – це також Kanban-дошка, але з дещо меншим функціоналом, ніж в Asana. Дошку не можна проглянути у вигляді списку та немає жодної статистики. Коли завдання попадає в колонку типу «Done», вона вважається зробленою. Але прослідкувати коли саме завдання було зроблено та скільки часу зайняло виконання ми, на жаль, не можемо. В Trello немає можливості поглибитися в кожне конкретне завдання при необхідності. Можна побачити стан усього проекту вцілому, але якщо проблема виникає з якоюсь конкретною задачею – це дуже важко прослідкувати та проаналізувати.

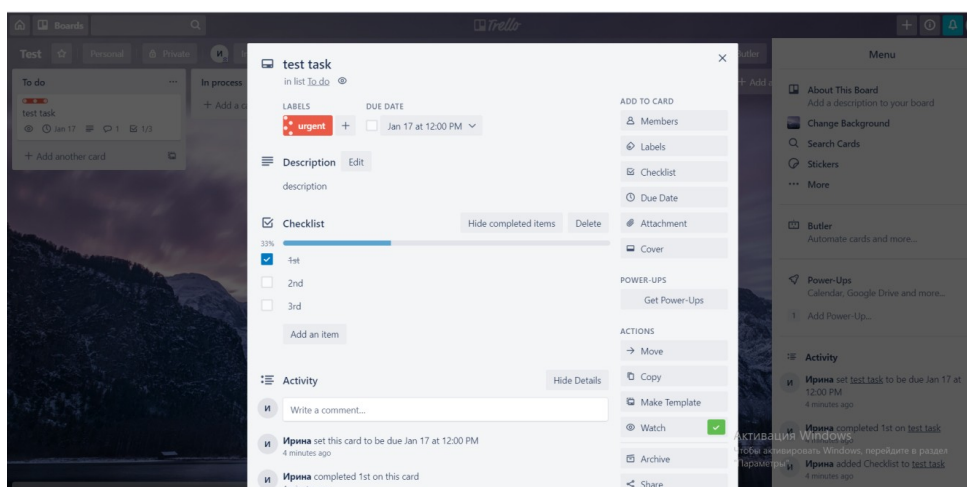


Рисунок 2.8 – Картка задачі з детальною інформацією

Не дивлячись на те, що на рис. 2.8 ми бачимо, що в Trello також можна проглянути картку з завданням, але чим більше інформації у картці знаходиться – тим загромадженіше вона буде та складніше орієнтуватися в завданні.

Тому, переваги Trello стали його основними недоліками. По перше, від безкоштовних програм не можна чекати чогось справді корисного. Це непоганий таск-менеджер, але на мою думку, він більше підійде для планування свого власного розпорядку та особистих справ, але не для того щоб ефективно управляти проектами розробки ресурсів. З власного досвіду користування даною програмою з впевненістю можу сказати, Trello не забезпечити в повній мірі основні процеси управління: планування, організацію виконання, контроль і аналіз відхилень.

JIRA – це система управління проектами (СУП), яка підтримує для усунення помилок і управління проектами в компанії любого розміру. Цей інструмент для всіх співробітників у командах та керівників проектів. Розробники даної системи з впевненістю заявили, що вона – універсальна, а отже повинна майже ідеально підійти для ІТ-компаній різного рівня. Проаналізуємо дану систему, щоб впевнитись в цьому [17].

JIRA – це інструмент, розроблений австралійською компанією Atlassian. Він використовується для відстеження помилок, проблем та управління проектами. Назва «JIRA» насправді успадкована від японського слова «Gojira», що означає «Годзілла».

Основне використання цього інструмента – відстеження проблем та помилок, пов'язаних із програмним забезпеченням та мобільними програмами. Він також використовується для управління проектами. Тому саме з цієї сторони ми і розглянемо дану систему – її цінність як системи управління проектами. Інформаційна панель JIRA складається з безлічі корисних функцій і функцій, які полегшують вирішення проблем. Деякі з основних функцій перераховані нижче:

Деякі корисні функції, які адміністратор JIRA надає користувачам:

- а) журнал аудиту. У розділі Журнал аудиту ви можете переглянути всі відомості про створену проблему та зміни, внесені до проблем;
- б) зв'язок між питаннями. Показує, чи пов'язані ваші проблеми з будь-якою іншою проблемою, яка вже присутня або створена в проекті;
- в) пошта в JIRA. Використовуючи поштову систему в адміністраторі, ви можете надсилати задачі до облікового запису на поштовому сервері POP або IMAP або повідомлень, записаних у файлову систему, створену зовнішньою поштовою службою;
- г) події. Подія описує статус, шаблон за замовчуванням, схему сповіщень та асоціації функцій пост переходу робочого процесу для події. Події класифікуються на дві події системи (події, визначені JIRA) та спеціальні події (події, визначені користувачем);
- д) список спостережень. JIRA дозволяє переглядати певну проблему, яка повідомляє про сповіщення про будь-які оновлень, пов'язаних з цим питанням;
- е) випуск колекціонерів. У формі питань JIRA колекціонер випусків дозволяє збирати відгуки на будь-якому web-сайті. Якщо ви натиснете на колекціонерів випуску, в адміністрації відкриється опція із запитом «Додати колектор випуску». Після того, як ви налаштували зовнішній вигляд колектора випусків, вставте створений JavaScript на будь-який web-сайт для зворотного зв'язку;
- ж) інструменти розробки. Ви також можете підключити свої засоби розробки програмного забезпечення до JIRA за допомогою цієї функції адміністратора. Вам потрібно ввести URL-адресу програми, щоб з'єднатися з JIRA [17].

Основні можливості JIRA:

- відстеження та робота з проектами – огляд процесів роботи розробників над проектами, виявлення помилок в використанні проекту клієнтами, та їх оперативне виправлення.

- вирішення питань – документи, пошта, публічність – все в одному місці, швидко і без зусиль.
- планування робочого процесу – побудова робочого процесу ефективніше і зручніше.
- відстеження ефективності – вирішення завдань, пріоритетів, допомога командам у виділені важливих частин в їх роботі та відстеження виконання завдань.
- спільна робота – обмін інформацією по проекту, спільне вирішення питань і звернення за підтримкою до колег.
- видимість онлайн – новини, пошта, огляд роботи в режимі реального часу в одному місці.
- інтеграція з різними розробками та доповненнями altassian та іншими розробниками.

Поглянемо на інтерфейс даної програми. Інформаційна панель JIRA відкриється, коли ви введете свій логін та пароль. Під панеллю приладів JIRA ви знайдете варіант Project, натиснувши на нього, відкриється вікно з переліком таких варіантів, як Просте відстеження випусків, Управління проектами, Agile Kanban, Jira Classic тощо, як показано на рис. 2.9.

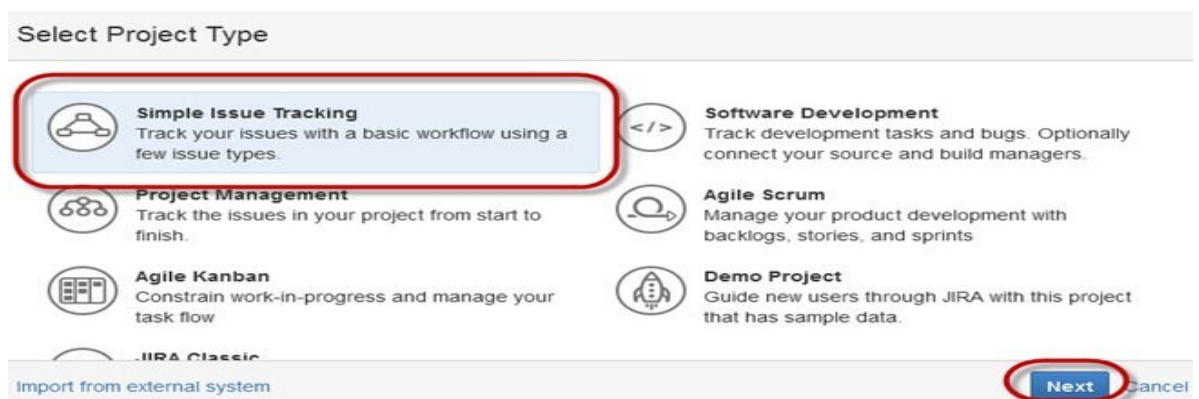


Рисунок 2.9 – Інформаційна панель JIRA [17]

Після натискання кнопки «Submit» відкриється вікно, де ви можете створити список робіт, таких як створення проблем, призначення проблем,

перевірка стану таких питань, як «Вирішені», «Прогресуючі» або «Закриті» тощо.

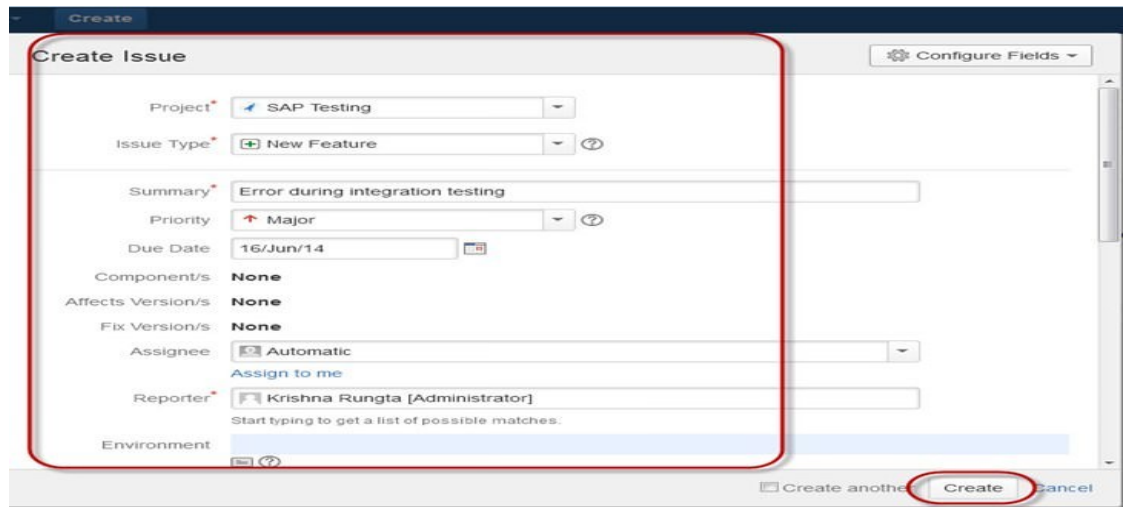


Рисунок 2.10 – Вікно створення задачі в JIRA [17]

Якщо ви хочете відредагувати проблему або хочете експортувати її до документа XML або Word, ви можете навести курсор миші на головній панелі та натиснути на «Issues». У розділі «Issues option» натисніть на пошук проблем, що відкриє вікно, звідки ви зможете знайти свої проблеми та виконати кілька функцій. На рис. 2.12 зображено основні функції та , які можна застосувати до поточних завдань в Jira.

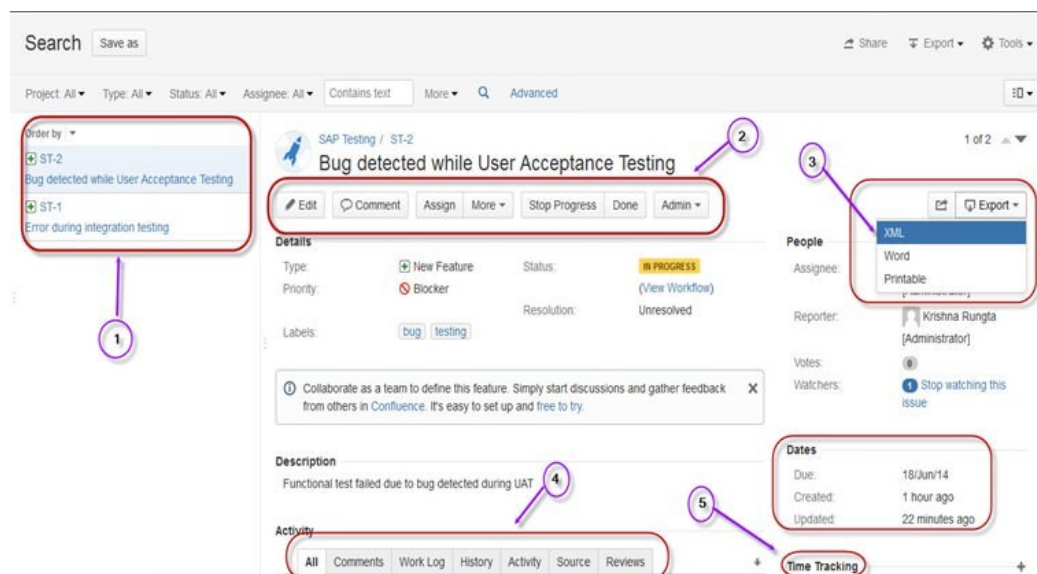


Рисунок 2.12 – Основні дії, які можна застосувати до задачі в JIRA [11]

Параметр «Search Issues» відкриє вікно, де ви зможете побачити створені вами проблеми. Тут у нас є проблеми ST1 та ST2.

На рис. 2.12 бачимо проблему «Виявлено помилку під час тестування прийняття користувача» та всі деталі, пов'язані з нею. Звідси можна виконати декілька задач, таких як:

- а) зупинити прогрес у задачі;
- б) редагувати задачі;
- в) коментувати задачі;
- г) призначати задачі.

Можна навіть експортувати деталі випуску в документ XML або Word.

Також є можливість переглядати активність, яка робиться по даному питанню, робити огляди цього питання, журнал роботи, історію випуску тощо.

У параметрі відстеження часу можна побачити час оцінки для вирішення проблем.

Це тільки невелика частина усього функціоналу JIRA. Але навіть з даного опису видно, що програма враховує усі основні задачі управління проектами розробки web-ресурсів. Вирішується питання про збереження документації, що дуже важливо.

Але, не зважаючи, на багатofункціональність та практично універсальність, для моєї команди на даному етапі розвитку компанії дана СУП не підійшла. Основні причини:

- програма достатньо складна. Для її освоєння потрібен час, якого наразі в нас недостатньо;
- інтерфейс перевантажений, є багато непотрібних функцій для нашої команди на даному етапі;
- не зважаючи на те, що програма універсальна, деякі з шаблонів потрібно налаштувати під специфіку роботи кожної окремої компанії;
- не усі робітники наразі повністю готові перейти до використання даної програми.

Було проведено опитування серед членів команди щодо готовності перейти на JIRA. Лише 10% готові перейти до використання даної програми. 90% тих, хто не готовий користуватися даною програмою основними причинами назвали:

- незрозумілий інтерфейс;
- через навантаження немає достатньо часу для освоєння даної програми;
- надаю перевагу іншій програмі.

Для проджект-менеджера основною проблемою є час. На те, щоб розібратись в ній, підготувати необхідні шаблони, або навіть внести певні налаштування та скоріш за все провести певні майстер-класи для усієї команди для того, щоб кожен з учасників користувався програмою з комфортом. Тому потрібно все добре зважити і спланувати, якщо ви чітко вирішили обрати JIRA для своєї роботи, тому що переваг більше ніж достатньо.

У додатку А в таблиці А.1 наведена порівняльна характеристика вище описаних систем управління проектами web-ресурсів [15].

2.3 Аналіз сучасних методів управління проектами розробки web-ресурсів

Найбільш розповсюдженими підходами до управління сучасними web-проектами, є такі як традиційний підхід, Agile методи, метод критичного шляху та процесно-орієнтований метод [16,17].

Традиційний підхід до управління проектами визначає певну послідовність дій, які необхідно здійснити. У традиційному підході виділяють п'ять основних етапів:

- а) ініціація;
- б) планування та дизайн;
- в) виконання;

- г) контроль (моніторинг);
- д) завершення.

Не у всіх проектах є всі вищезазначені етапи, так як проекти можуть бути завершені раніше, ніж будуть виконані повністю. Деякі проекти не проходять етап структурного планування або моніторингу, а деякі можуть пройти через 2, 3 та 4 етап багаторазово. У різних галузях використовують різні варіації цих етапів проекту.

В ІТ-галузі традиційний підхід є більш відомий як водоспадна (каскадна) модель [15], тобто, виконання ряду задач одну за другою по порядку.

Водоспадна модель добре працює на малих, чітко визначених проектах, і часто є неефективною для великих проектів, які не повністю визначені на початковому етапі. Особливо важливо мати це на увазі при розробленні нових або інноваційних продуктів.

Варто зазначити, що для розробки програмного забезпечення водоспадна модель не дуже підходить з наступних причин. По-перше, через те, що вимоги, які висловлює замовник до продукту, є не формалізовані, можуть бути неодноразово уточнені протягом процесу розроблення, а інколи навіть після завершення процесу розробки та тестування. По-друге, змінними є обставини та умови, у яких використовується СУП, і з часом вона система потребує удосконалення або перестає використовуватися.

Зважаючи на необхідність удосконалення, можна зробити висновок, що каскадна модель виправдає себе лише як модель життєвого циклу для першої версії продукту [18].

Гнучкий підхід до управління проектами включає в себе деякі ітеративні підходи, що засновані на принципах управління взаємодією людей та базується на процесному погляді на людську співпрацю. Методи, що базуються на Agile, найбільш часто використовуються у розробці програмного забезпечення, web-сайтів, творчої та маркетингової індустрії.

Цей підхід контрастує із традиційними підходами, такими як водоспадний метод. У гнучкому розробленні програмного забезпечення проект позиціонується як послідовності відносно невеликих задач, які виконуються адаптивним шляхом, як того вимагає ситуація, а не так, як було заплановано.

Agile – це збірний термін для багатьох методологій управління проектами, зокрема:

Scrum. Цілісний підхід до розробки продукту, який фокусується на ітеративних цілях, які ставить Власник продукту (Product Owner), та які сформульовані у перелік вимог до функціональності проекту. Вони втілюються Командою проекту за сприянням Scrum Master'а.

Extreme Programming (Екстремальне програмування, XP) – набір практик, що базуються на наборі принципів та цінностей, ціллю яких є впровадження тісного співробітництва та зворотних відгуків на всіх рівнях розробки продукту. Екстремальне програмування використовують для зниження вартості змін до проекту. XP популяризувало TDD (розробка через тестування) та парне програмування [19, 20].

eXtreme Manufacturing (XM), що базується на методах Scrum, Kanban та Kaizen і сприяє швидкому проектуванню та прототипуванню.

Crystal Clear, що являє собою гнучну методологію, що зосереджена на спільному розташуванні команди та тісній взаємодії. CrystalClear фокусується на людях, а не на процесах чи артефактах.

Kanban – основа для вдосконалення процесів, що використовуються для управління поточною роботою у проектах, які управляються за гнучкими підходами. Kanban є специфічним підходом для розробки програмного забезпечення [21].

Scrumban – змішаний підхід до управління проектами, що базується на Scrum та Kanban. Він фокусується на тому, щоб використати гнучкість Kanban та додавання до нього структури Scrum.

Метод Kanban означає систему візуального управління процесом (для цього використовується дошка з завданнями на різних фазах), яка вказує що виробляти, коли виробляти, і як багато виробляти.

Основний наголос в цьому підході робиться на «зробити все вчасно» та униканні перевантаження членів команди. При цьому підході, процес від опису завдання до доставки результатів її виконання користувачу наочно показується учасникам процесу, і члени команди можуть брати завдання з черги.

Kanban буде більш відповідним інструментом для компаній, в яких вже є робочі процеси, які хочеться з часом поліпшити, не перетрушуючи всю систему.

Якщо організація дійсно застрягла і потребує докорінних змін і зрушення в бік більш ефективного процесу, більш відповідним буде Scrum.

Scrum – це протилежність великих колекцій переплетених обов'язкових компонентів. Scrum – це більше не методологія. Scrum реалізовує науковий метод емпіризму. Scrum означає запрограмований алгоритмічний підхід на евристичний, з повагою до людям та самоорганізацією, що має зробити з непередбачуваності та зменшити складні проблеми.

Принципи Scrum дозволяють в обмежених строках і з невеликим інтервалом ітерацій, які називають спринтами (спринти), надати готовий результат. Порядок виконання задач визначає пріоритет.

В середньому, один спринт триває один тиждень. Задачі розповсюджують команду на scrum-мітингу. Команда обирає проджект-менеджера, який є власником продукту – власник продукту – і відповідає перед клієнтами [22].

Першочергові задачі за всіма проектами спливають в «беклог» – так називають «список проблем». Вони мають пріоритет від звичайних до строкових. Команда берет задачі з беклогів і виконує їх за одну ітерацію, або розбиває на підзадачі.

Більш пріоритетні показники отримують статус коротких, «красних». Таким завданням відводиться від 30 до 50% часу в залежності від відділу та команди.

Степінь завантаженості команд визначає фокус-фактор – це результат часу, який планується витратити на виконання завдань. Зазвичай він становить 60–70%, а час, який залишився закладається на термінові та непередбачуванні завдання.

Але, наприклад, у розробників початковий фокус-фактор становить 50%, після чого багато задач просто «прилітає» у процесі виконання тижневого спринта. Таке гнучке планування не тільки не зміщувати терміни виконання запланованих задач і при цьому успішно виконувати ті непередбачувані задачі, які «прилетіли» [23, 24].

РОЗДІЛ 3

ЗАСТОСУВАННЯ МОДЕЛЕЙ УПРАВЛІННЯ ПРОЕКТАМИ ПРИ РОЗРОБЦІ WEB-РЕСУРСУ

3.1 Постановка завдання по створенню динамічного web-сайту

До нас звернувся клієнт із проханням створити для нього динамічний web-сайт для його бізнесу з використанням конкретної CMS – Wordpress.

Основні вимоги клієнта щодо сайту:

- а) WordPress, можливо розміщений на хостингу Siteground;
- б) повна підтримка редактора Gutenberg;
- в) адаптивна тема (кросбраузерність);
- г) повна підтримка базових плагінів:
 - ActiveCampaign;
 - SEO Framework;
 - плагін перенаправлення;
 - плагін перевірки теми;
 - W3 Total Cache (або кешування може бути зроблено хостинговою компанією);
 - плагін для можливості вставляти код в хедер та футер.

Основна ціль клієнта – створити швидкий, SEO-оптимізований динамічний сайт з можливістю зміни існуючих та додавання нових web-сторінок без звернення до коду.

1-й етап. На першому етапі перегорів, стадії інтерв'ю, клієнт має:

- а) куплений хостинг та доменне ім'я для розміщення майбутнього сайту;
- б) готові на 90% макети усіх необхідних web-сторінок.

На першому етапі від нас необхідна:

- а) попередня оцінка часу виконання даного проекту;

б) рекомендації щодо технології роботи виходячи з основних цілей клієнта.

Так, як однією з вимог клієнта є використання редактору Gutenberg для створення сайту, було запропоновано 2 способи створення web-сайту. Але для початку, коротко про Gutenberg редактор. Цей редактор є стандартний редактором, який «вшит» у CMS Wordpress для створення і редагування записів і сторінок. Проект названий ім'ям Йогана Гутенберга, який презентував Європі друкарський верстат і почав друковану революцію.

Обидва способи створення динамічного web-сайту передбачають його створення з нуля, без використання будь-яких платних шаблонів - тобто створення власної кастомізованої теми. Переваги даного способу в тому, що сайт буде «чистим», тобто без зайвого коду та непотрібних плагінів, що позитивно вплине на швидкість завантаження сайту після закінчення всіх робіт.

1-й спосіб. Створення кастомізованої теми на базі використання тільки Gutenberg-редактора. Перевагою є 100%-ва відповідність вимозі клієнта щодо Gutenberg-редактора. Недоліком є те, що час час виконання проекту збільшується майже в два рази у даному випадку. Попередня оцінка проекту – 200 годин.

2-й спосіб. Створення кастомізованої теми на базі використання Gutenberg-редактора та додаткового плагіну Advanced Custom Fields (ACF Pro). Головна перевага – зменшення часу виконання проекту майже в 2 рази у порівнянні з 1-м способом. Недоліком є те, що вимога клієнта щодо використання Gutenberg виконується частково. Попередня оцінка проекту: 100 годин.

Проджект-менеджером було запропоновано 2 варіанти створення web-сайту на вибір, приведені аргументи щодо кожного з них та надіслана додаткова інформація, яка допомогла клієнту зробити вибір – був обраний 2-й варіант.

1-й етап проекту пройдена. Клієнта влаштовує попередня оцінка та обрана технологія.

2-й етап. На цьому етапі йде активна комунікація між проджект-менеджером та клієнтом та усіма учасниками проекту щодо виявлення усіх деталей проекту:

- а) структура web-сайту;
- б) функціонал, який буде використовуватись на web-сайті;
- в) інтеграція з іншими програмами, сайтами і сервісами.

Визначено, що на даному етапі планується розробка 12 сторінок, з функціоналу буде присутня інтеграція контактних форм с сервісом ActiveCampaign, налаштування кешування та оптимізація швидкості завантаження сайту. З учасниками проекту, зокрема front-end та back-end програмістом та SEO-спеціалістом були узгоджені усі задачі та технології роботи.

3-й етап. Після обговорення даних пунктів, проджект-менеджер займається плануванням та постановкою задач. Планування включає в себе:

- а) розробку проектної документації;
- б) складання плану проекту;
- в) узгодження термінів (дедлайнів);
- г) аналіз можливих ризиків.

Це один з найважливіших етапів. На етапі 2 клієнт надіслав усі необхідні макети для роботи в потрібному для розробника форматі (рис. 3.1).

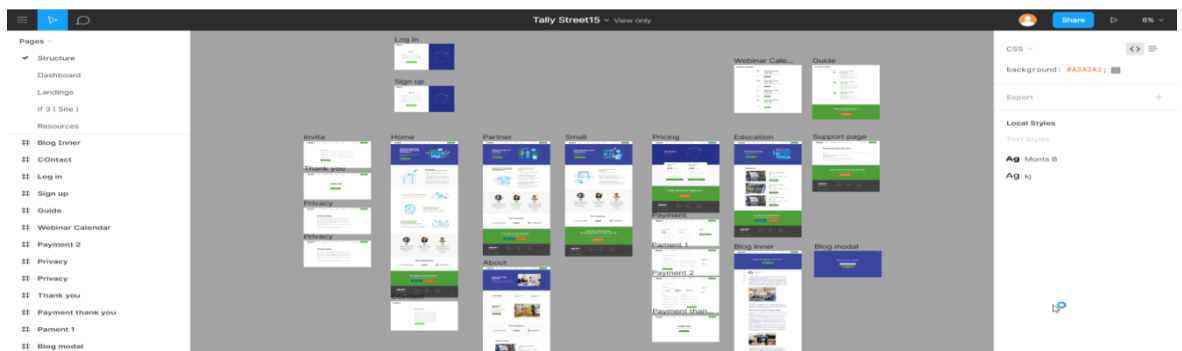


Рисунок. 3.1 – Макети для майбутнього сайту

Основні завдання:

- а) підготовка сайту для роботи:
 - налаштування CMS та адмін-панелі;
 - створення необхідного базового шаблону на основі ACF+Gutenberg як фундаменту для усіх інших web-сторінок;
- б) безпосередньо верстання усіх сторінок з урахуванням SEO-вимог;
- в) наповнення контентом;
- г) налаштування контактних форм та інтеграція з сервісом ActiveCampaign;
- д) налаштування кешування та оптимізація швидкості завантаження сайту;
- е) тестування. Паралельно після наповнення web-сторінок контентом необхідно запустити тестування готових сторінок для виявлення відхилень від макету та їх перевірка на кросбраузерність. Тому даному етапі у систему «Проект» підключається тестувальник. Проджект-менеджер видає задачі тестувальнику та чекає від нього фідбеку. Усі зауваження проджект-менеджер передає розробнику проекту. Після виправлень – сторінки знову відправляються на тестування допоки останній не дасть фідбек, що зауважень немає. Також є такий нюанс в даному пункті – часом, не усі зауваження тестувальника потрібно виправляти, тому що іноді в отриманих від клієнтів макетах не враховані головні правила верстання і розробник просто не може «зробити так як на макеті». Такі моменти потрібно обговорювати на етапі первинної чи хоча б фінальної оцінки проекту, але іноді, розробник не відразу звертає увагу на такі деталі, тому з'являється необхідність швидкої реакції проджект-менеджера на стадії верстання – необхідно сповістити про це клієнта та врегулювати дане питання;
- е) здача готового проекту клієнту;
- ж) післяпроектна підтримка (завжди пропонується клієнту).

3.2 Поетапна процедура створення динамічного web-сайту із застосуванням системи Asana та методології Kanban

Основна інформація щодо проекту наведена у табл. 3.1.

Таблиця 3.1– Основна інформація щодо проекту

Параметр	Результат
Початок проекту	14.11.19
Орієнтований кінець проекту (узгоджено з клієнтом)	13.12.19
Технічне завдання	Є
Середовище для роботи	Є
Необхідні матеріали для роботи	Є
Назначені учасники проекту	front-end розробник, один back-end розробник, SEO-спеціаліст, тестувальник.
Відповідальний за проект проджект-менеджер	Є
Оцінка проекту	Є

Визначенні початок та кінець проекту, основне технічне завдання, середовище та матеріали для початку роботи. Проджект-менеджером були назначені усі учасники проекту (розробники, seo-спеціаліст для консультацій та тестувальник). Роботу усіх учасників проекту контролює проджект-менеджер.

Проаналізувавши системи та методи управління проектами по розробці web-ресурсів у розділі 2, для визначення пріоритетів, планування виконання поставлених завдань та їх контролем було вирішено використовувати систему ASANA та KANBAN. Постановка завдань у системі ASANA виглядає наступним чином (рис. 3.1 - 3.3).

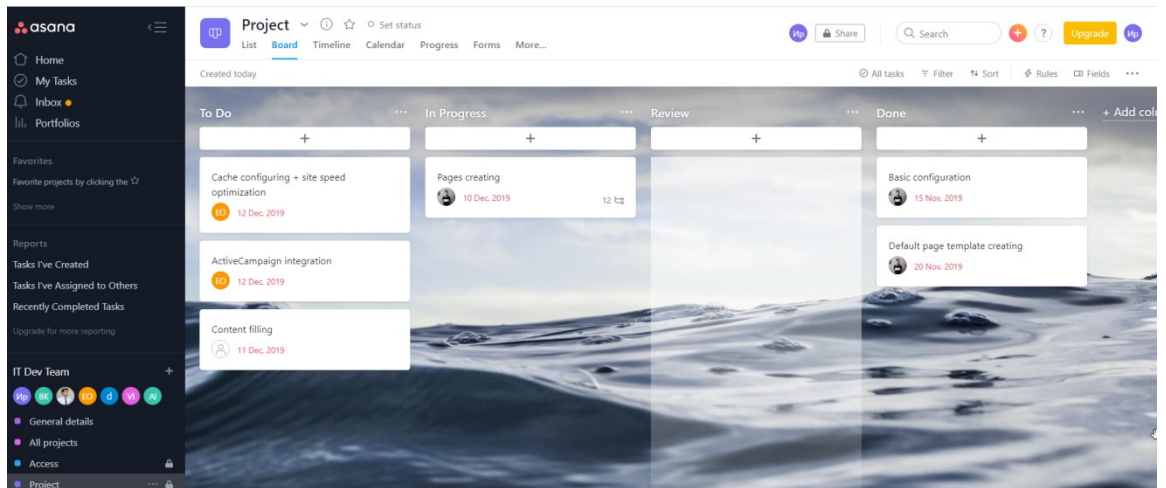


Рисунок 3.1 – Канбан-дошка з проектом в ASANA

На рис.3.1 зображено вид Канбан-дошки з завданнями у вигляді карток. Картки організовані за допомогою горизонтальних стовпчиків (Swimlanes), обмежень (WIP межі), субстовпчиків, а також стандартного набору стовпчиків («To-do», «In Progress» та «Done»).

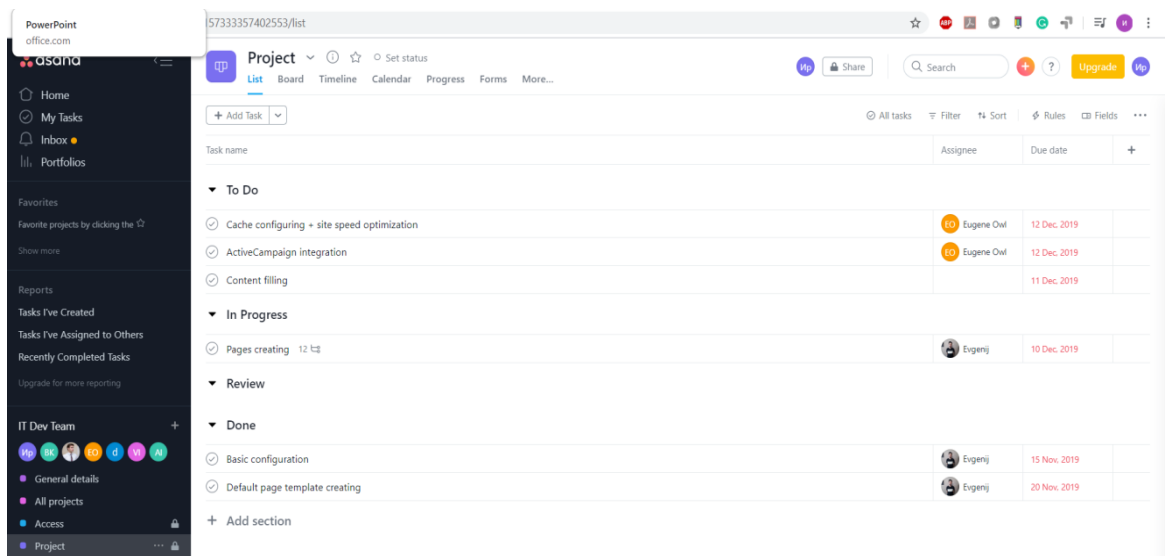


Рисунок 3.2 – Канбан-дошка у вигляді списку з проектом в ASANA

На рис. 3.2 зображено дошку у вигляді списку. За допомогою списку легше орієнтуватися у термінах виконання та кількості завдань, а також з легкістю можна побачити відповідальних розробників за кожен конкретну задачу.

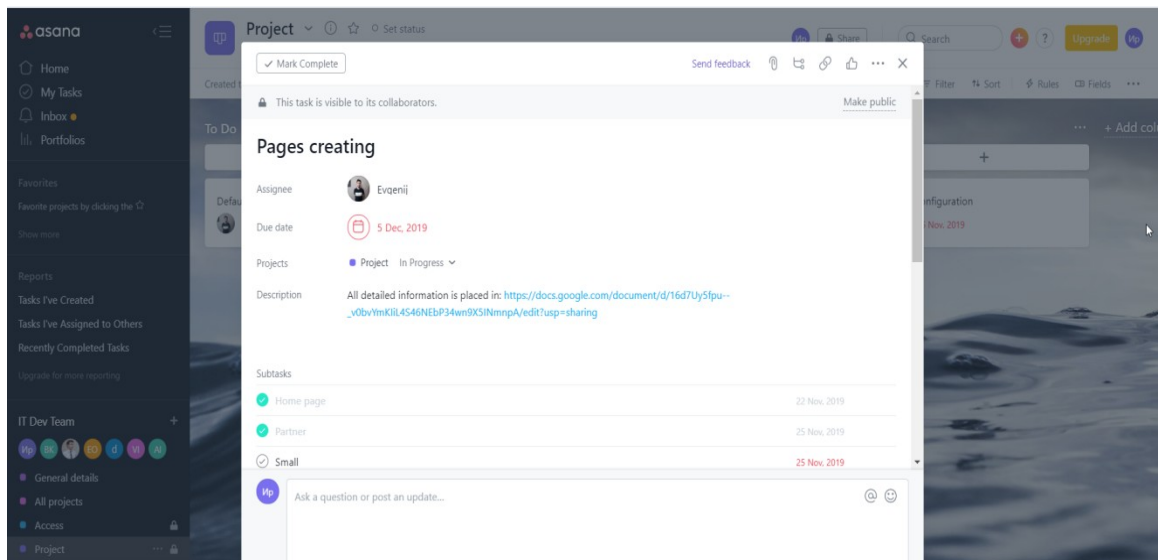


Рисунок 3.3 – Картка завдання в відкритому вигляді

На рис.3.3 зображено картку з завданням у відкритому вигляді . Тут можна проглянути коментарі, які були залишені безпосередньо проджект-менеджером, або іншими учасниками проекту щодо конкретного завдання. Також можна проглянути підзадачі, якщо вони є та помітити ті, що вже виконані.

На кожному з етапів, починаючи зі старту проекту проджект-менеджер контролює дедлайни, спілкується з клієнтом, формує правки та нові завдання, підготує документи та утримує команду в робочому стані – і так по колу.

У результаті створення кастомізованої теми на базі використання Gutenberg-редактора та додаткового плагіну Advanced Custom Fields (ACF Pro). На рис. 3.4 та 3.5 зображено готовий web-сайт. На рис. 3.4 зображено саме редактор головної сторінки (Home page). Справа відображаються ACF-поля, а в центрі – Gutenberg редактор. Редагувати сторінку можна у самому Gutenberg-редакторі або за допомогою ACF-полів. Зміни з'являються миттєво на сторінці після її оновлення. Якщо редагувати сторінку за допомогою ACF, то, навіть, не потрібно оновлювати сторінку та переходити на неї у браузері, щоб побачити зміни.

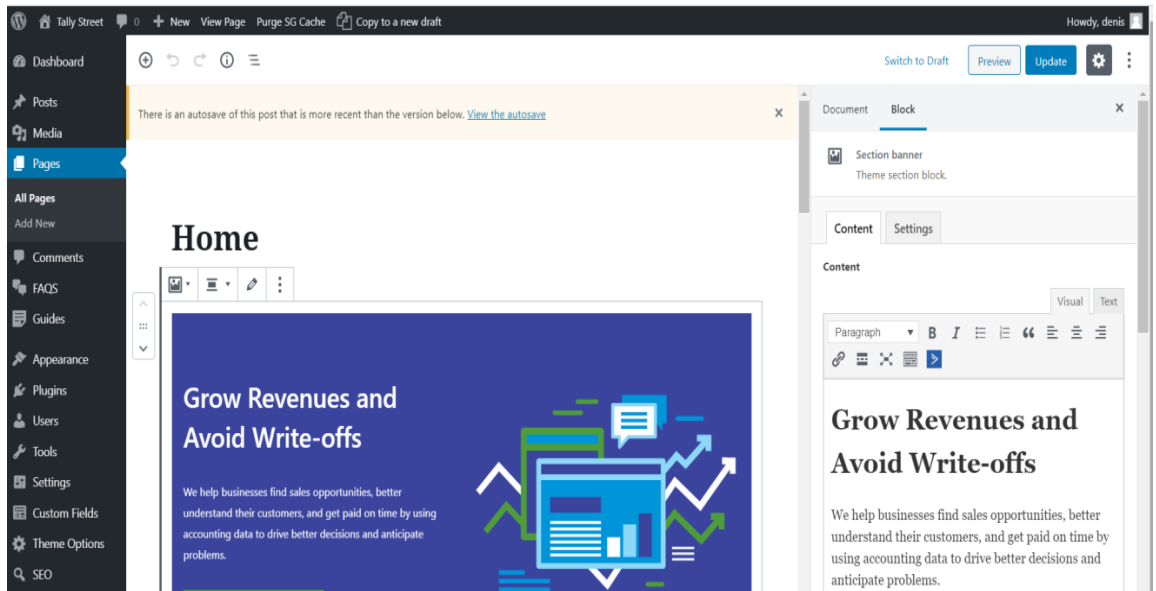


Рисунок 3.4 – Адмін-панель сайту.

Нижче, на рис. 3.5 зображено вигляд головної сторінки готового веб-сайту у браузері. Так бачить сайт кожен відвідувач сайту.

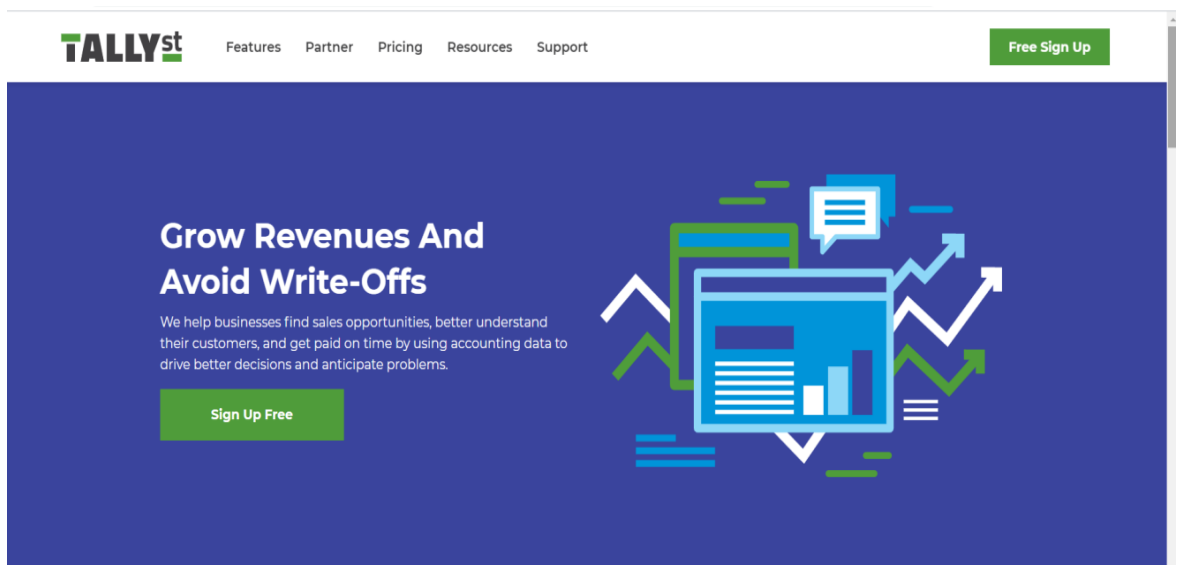


Рисунок 3.5 – Зовнішній вигляд сайту

Зовнішній вигляд сайту повністю відповідає макетам, які були отримані від замовника на етапі 2. Усі вимоги були виконані, сайт повністю адаптивний (на мобільних пристроях зовнішній вигляд сайту був зроблений на розсуд розробника на seo-спеціаліста).

3.3 Аналіз ефективності застосованого методу та підходу, а також якості кінцевого продукту

Ефективність управління розробкою web-ресурсів за Kanban методом та з використанням ASANA полягає у наступному:

- а) використанні максимуму ресурсів і підвищенні ефективності управління;
- б) зменшенні часу на роботу, її трудомісткості;
- в) поліпшенні якості продукції та послуг;
- г) збільшенні швидкості аналізу і оцінки стану проекту внаслідок контролю виконання робіт;
- д) збільшенні гнучкості команди;
- е) чіткому контролю термінів, а також узгодженості бюджету на даний проект;
- є) появі можливості кращого управління ризиками;
- ж) централізованості інформації, поліпшення оперативності отримання необхідних даних.

Більш детально по кожному пункту.

Загальні критерії оцінки ефективності по проектах поділяються на три основні групи:

- збільшення ефективності роботи за затвердженими параметрами (поліпшення якості виробленого продукту, досягнення завдань проекту);
- кількісні показники, які дозволяють побічно дати оцінку роботі (наявність помилок в роботі співробітників, кількість оброблених ними запитів і т.д.);
- виконання певних цілей (інтерпретація звітної інформації, консультативна підтримка, організація аналізу зібраної інформації та інше).

Кількісна оцінка ефективності управління проектом може бути здійснена за допомогою порівняльного аналізу зміни будь-яких характеристик:

- а) відхилення в бюджеті, затвердженому на проєкті;
- б) відхилення у виконанні робіт або в розкладі;
- в) число нерозв'язаних проблем;
- г) позбавлення від недоліків, які були виявлені під час перевірки якості проєкту;
- д) повна укомплектованість проєктної групи [25,26].

У результаті використання системи управління ASANA та методу Kanban, ефективність даного проєкту зросла у 1,5 рази. Кількісна оцінка ефективності управління виглядає наведена у табл.3.1.

Таблиця 3.1– Кількісна оцінка ефективності управління проєктом

Критерій	Оцінка
Відхилення в бюджеті, затвердженому на проєкті	відхилення на 10%
Відхилення у виконанні робіт або в розкладі	усі роботи були виконані за розкладом
Кількість проблем у процесі управління проєктом	3 проблеми у процесі роботи
Позбавлення від недоліків, які були виявлені під час перевірки якості проєкту	100% (3 проблеми були оперативно вирішені)
Повна укомплектованість проєктної групи	100%

Виконання основних вимог клієнта наведено у таблиці 3.2.

Таблиця 3.2 – Виконання основних вимог клієнта щодо створення динамічного web-сайту

Вимога	Статус виконання
WordPress, можливо розміщений на хостингу Siteground	виконано
Повна підтримка редактора Gutenberg	виконано частково, запропонована інша технологія
Адаптивна тема (кросбраузерність)	виконано
Повна підтримка базових плагінів:	
– ActiveCampaign;	виконано
– SEO Framework;	виконано
– плагін перенаправлення;	виконано
– плагін перевірки теми;	виконано

Продовження таблиці 3.2

Вимога	Статус виконання
– W3 Total Cache (або кешування може бути зроблено хостинговою компанією);	виконано
– плагін для можливості вставляти код в хедер та футер.	виконано

Отже, з таблиці 3.2 бачимо, що основні вимоги були виконані. Вимога щодо редактора була виконана частково, але запропоноване рішення повністю задовольнило клієнта та зменшило час розробки удвічі. У таблиці 3.3 наведені досягнення основних цілей клієнта.

Таблиця 3.3 – Досягнення основних цілей клієнта щодо створення динамічного web-сайту

Ціль	Статус виконання
Швидкий сайт	виконано
SEO-оптимізований	виконано
Можливість редагувати та додавати нові сторінки без звернення до коду	виконано

Головні цілі проекту досягнуті. У результаті ми отримали якісний продукт у вигляді динамічного web-сайту, виконали усі задачі у визначені терміни та оперативно вирішили усі проблеми, які виникали у процесі роботи.

Завдяки Kanban-методу:

- а) команда концентрувалася тільки на поточній роботі, пріоритет завдання виставлявся проджект-менеджером;
- б) процес управління проектом розробки web-ресурсів став більш «прозорим», завдання стали більш чіткими;
- в) усі правки та нові задачі одразу з'являлись у системі ASANA, тому всі учасники одразу бачили нові завдання та пріоритетність;

г) контроль над виконання завдань у визначені терміни покращився, так як було чітко видно термін виконання кожної задачі та кількість виконаних завдань на певний момент.

Проджект-менеджер міг швидко знаходити вузькі і проблемні місця, які з'явилися через дефіцит уваги, людей або навичок та оперативно їх вирішувати.

Варто зазначити, що порівняти ефективність декількох проєктів є достатньо складною темою та потребує подальшого аналізу та створення напів-універсального механізму оцінки ефективності, тому що усі проєкти – різні, в кожному є свої нюанси, які досить важко оцінити.

Для кожної ІТ-компанії підійде своя система управління проєктами та методологія. Для нашої компанії, на даному етапі розвитку, було прийнято рішення використовувати систему ASANA та метод Kanban з подальшим переходом до KanbanScrum-методу + ASANA. Використання даного методу дало змогу підвищити ефективність управління проєктами розробки web-ресурсів та зменшити час на обговорення завдань. Не менш важливим є те, що система управління Asana дала змогу зберігати основну інформацію та документацію по проєкту в одному місці. Усі учасники проєкту мали швидкий доступ до усіх необхідних матеріалів.

ВИСНОВКИ

Інформаційні технології міцно інтегрувалися в наше життя і охоплюють практично всі сфери діяльності. Ринок «ІТ» – один з найбільших і швидкозростаючих в Україні, і його перспективи хвилюють не тільки сотні тисяч працюючих на ньому українців, а й зарубіжних замовників, у яких українські ІТ-розробки користуються високим попитом.

Сучасний етап розвитку проектів, результатом виконання яких є продукт, що функціонує у World Wide Web, характеризується обов'язковою потребою максимально швидкого прийняття оптимальних рішень та необхідністю підвищення ефективності функціонування у жорстких конкурентних умовах. На відміну від попереднього етапу первинних спроб ведення комерційної діяльності у всевітній мережі Інтернет, коли принциповою задачею був власне перехід бізнесу від традиційних вербальних до інноваційних інтерактивних методів роботи, зараз більш актуальними є задачі максимально швидкого зростання та захоплення своєї частки ринку у Інтернеті. Забезпечити необхідну швидкість та якість рішень, що приймаються, можна шляхом розроблення відповідного математичного та програмного забезпечення, за допомогою якого можна буде оптимізувати та частково або повністю замінити дії менеджера проекту.

Оскільки управління проектами в розробці web-ресурсів в Україні є достатньо молодого дисципліною, то постає багато невирішених питань та проблем.

Сучасні системи управління проектами забезпечують широкі можливості безпосередньо у процесі управління проектом, але не дають можливості обробки вхідної інформації про проект та його зацікавлені сторони, яка може бути використана для прийняття проектних рішень.

Перспективи розвитку управління проектами необхідно розглядати з урахуванням таких обставин сучасних умов господарювання, як: скорочення життєвого циклу продукту, глобальна конкуренція, зростання обсягу знань,

збільшення розміру корпорацій, поглиблення товарної диференціації, геополітичні зміни у світі, зростання складності управління невеликими проектами, що виконуються одночасно, за рахунок збільшення їх кількості. Вплив цих чинників обумовлює зростання важливості управління проектами. Проявом зростаючого інтересу до управління проектами є стрімкий розвиток професійних організацій, які об'єднують фахівців із управління проектами різних континентів і країн, напрямів і сфер діяльності, національностей та культур.

Незважаючи на оптимістичні прогнози, залишаються фактори, що негативно впливають на розвиток ІТ-індустрії в Україні:

- а) нестача кваліфікованих кадрів;
- б) застарівші освітні програми;
- в) проблеми в управлінні проектами розробки web-ресурсів;
- г) проблеми в питаннях захисту інтелектуального права;
- д) нестабільна репутація держави як надійного бізнес-партнера.

У даній роботі активно досліджувались саме проблеми в управлінні проектами розробки web-ресурсів, так як, багато бізнес-фахівців недооцінюють важливість управління проектами. На передній план у них завжди виходять розвиток бізнесу і показники рентабельності, а управління часто взагалі не потрапляє в список пріоритетів. Більшість людей навіть не здогадуються, що відсутність налагодженої організації по управлінню проектами приносить їм збитки.

Результатами роботи є отримання якісного продукту у вигляді динамічного web-сайту, виконання усіх задач у визначені терміни та оперативне вирішення проблем, які виникали у процесі роботи.

Завдяки застосованому методу та системи управління проектами:

- а) команда концентрувалася тільки на поточній роботі, пріоритет завдання виставлявся проджект-менеджером;
- б) процес управління проектом розробки web-ресурсів став більш «прозорим», завдання стали більш чіткими;

в) усі правки та нові задачі одразу з'являлись у системі ASANA, тому всі учасники одразу бачили нові завдання та пріоритетність;

г) контроль над виконання завдань у визначені терміни покращився, так як було чітко видно термін виконання кожної задачі та кількість виконаних завдань на певний момент;

Нові технології дають менеджерам проектів прекрасні інструменти для поліпшення спільної роботи, оптимізації планування і підвищення ефективності роботи над проектами та якості кінцевого продукту.

ІТ-індустрія за досить короткий проміжок часу стала одним з ключових компонентів економіки України. При успішному розвитку вона здатна надати і вже робить позитивний вплив на економічні показники. А це означає, що попит на ІТ-фахівців і їх мотивація та удосконалення управління проектами будуть неухильно зростати.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Д. Кестел, В. Даве Руководство к своду знаний по управлению проектами: Руководство РМВОК. Москва : Олимп-бизнес, 2014. 792 с.
2. Статичні та динамічні web-сайти (2018). URL: <https://www.armedsoft.com/ua/blog/statychni-ta-dynamichni-web-sayty> (дата звернення: 15.09.19).
3. Berkun S. Making Things Happen: Mastering Project Management. Seattle: O' Reilly Media, 2008. 410 p.
4. Пучков И. И. Управление IT-проектами. *Молодой ученый*, Казань, 2017. №49 (183). С. 78 – 81. URL: <https://moluch.ru/archive/183/46925/> (дата звернення: 14.01.2020).
5. Томсон Л., Веллинг Л. Разработка Web-приложений на PHP и MySQL. Индианополис : ДиаСофтЮП, 2013. 672с.
6. Chatfield, Carl A short course in project management. Washington: Microsoft, 2007. 350 p.
7. Саати Т. Принятие решений. Метод анализа иерархий. Москва : Радио и связь, 1993. 278 с.
8. Саати Т. Аналитическое планирование. Организация систем. Москва : Радио и связь, 1991. 224 с.
9. Сервисы для совместной работы и управления проектами (2015).URL: <http://www.onlineprojects.ru/tools/pm/> (дата звернення 28.09.19).
10. Sebastian N. The Definitive Guide to Project Management: Every executives fast-track to delivering on time and on budget. London : FT Press, 2007. 256 p.
11. Paul C. The right projects done right. San Francisco: Jossey Bass Publishers, 2005. 336 p.
12. Lewis R. Ireland (2006) Project Management. McGraw-Hill Professional, 2006. 110 p.

13. Буріменко Ю. І. Управління проектами: навч. посібн. Одеса : Редакційно-видавничий центр ОНАЗ ім. О.С. Попова, 2017. 207 с.
14. Asana Guide. Navigating Asana. URL: <https://asana.com/guide/help/fundamentals/navigating-asana> (дата звернення: 11.10.2019).
15. J. Appelo. Management 3.0: Leading Agile Developers, Developing Agile Leaders. Boston: Addison-Wesley Signature Series Cohn, 2017. 418 p.
16. J. Sutherland, M. Schwaber Scrumguides. URL: <https://www.scrumguides.org/> (дата звернення: 18.10.19).
17. J. Phillips PMP Project Management Professional Study Guide. New York: McGraw-Hill Professional, 2003. 845 p.
18. Документація Trello. URL: <https://trello-gid.ru/> (дата звернення: 11.12.2019).
19. JIRA Tutorial: A Complete Guide for Beginners. URL: <https://www.guru99.com/jira-tutorial-a-complete-guide-for-beginners.html>
20. Архипенков С. В. Лекции по управлению программными проектами. Москва : Наука, 2009. 128 с. URL: http://www.arkhipenkov.ru/resources/sw_project_management.pdf (дата звернення: 02.12.19).
21. Бушуев С.Д. Развитие систем знаний и технологий управления проектами. Управление проектами и программами. Луганськ : СНУ ім. В. Даля, 2005. 205 с.
22. Професійний стандарт. Керівник проектів в області інформаційних технологій. URL: <https://mon.gov.ua/storage/app/media/vyshcha/IT-prof-standarty/3-ps-project-manager-13.12.2014.pdf> (дата звернення: 15.12.2019).
23. Tripathi B. Project Management Trends to Watch for in 2015 (2015). URL: <http://www.synquis.com/blog/8-project-management-trends-to-watch-for-in-2015> (дата звернення: 15.12.19).
24. Burger R. The 5 Biggest Project Management Trends for 2015. URL: <http://blog.capterra.com/biggest-project-management-trends-for-2015/> (дата звернення: 17.12.19).

25. Orton-Jones C. 7 global project management trends (2015). URL: <http://raconteur.net/business/seven-global-project-management-trends/> (дата звернення: 17.12.19).
26. Asana vs JIRA vs Trello URL: <https://www.capterra.com/task-management-software/compare/120550-19319-72069/Asana-vs-JIRA-vs-Trello> (дата звернення: 20.12.19).
27. Justin E., C. Steins Pro Web project management. New York : Apress, 2011. 248 p.

ДОДАТКИ

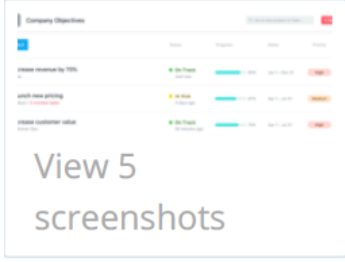
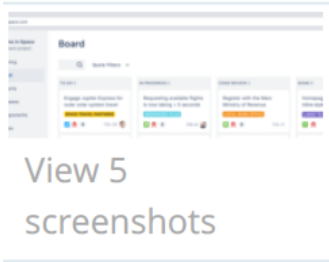
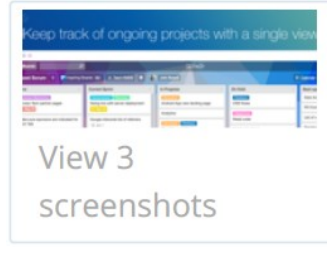















ДОДАТОК А

Інструменти управління проектами

Таблиця А.1 – Порівняльна характеристика систем управління проектами web-ресурсів [26].

	ASANA	JIRA	TRELLO
By	Asana	Atlassian	Atlassian
Price	\$10.99/month/user	\$10.00/month	\$9.99/month/user
Best for	Why do nearly 6000 customers on Capterra give Asana 4.5 out 5 stars? Because Asana is the easiest way to manage team projects, processes, and tasks, stay productive, and deliver better work faster.	Jira Software is trusted by agile teams looking to capture & organize issues, assign work&track team activity. The software is designed so Scrum, Kanban, & hybrid models are all successful.	Trello is for teams of all sizes, for the creators and doers, and for those who take dreams to done.
Rating	4.4 / 5 (8324)	4.3 / 5 (8168)	4.5 / 5 (15054)
Ideal number for	2 - 1000+	1 - 1000+	1 - 1000+
Easy to use	4.3 / 5	3.9 / 5	4.5 / 5
Customer Support	4.3 / 5	4.1 / 5	4.3 / 5
Functionality	4.3 / 5	4.4 / 5	4.3 / 5
Value for Money	4.4 / 5	4.2 / 5	4.6 / 5
Product Features	Collaboration Tools Create Subtasks Gantt/Timeline View, Mobile Access, Percent-Complete Tracking, Recurring Task Management, Task Board View, Time Tracking, To-Do List View	Collaboration Tools Create Subtasks Gantt/Timeline View, Percent-Complete Tracking, Recurring Task Management, Task Board View, Time Tracking	Collaboration Tools Create Subtasks Gantt/Timeline View, Mobile Access, Percent-Complete Tracking, Recurring Task Management, Task Board View, To-Do List View

Таблиця А.2 – Платформи та підтримка систем управління проектами web-ресурсів [26].

<p>Screenshot</p>	 <p>View 5 screenshots</p>	 <p>View 5 screenshots</p>	 <p>View 3 screenshots</p>
<p>Platform</p> <p>Web/</p>	<p>Web/Installed</p>  <p>Mobile</p> 	<p>Web/Installed</p>  <p>Mobile</p> 	<p>Web/Installed</p>  <p>Mobile</p> 
<p>Support + Training</p>	<p> 24/7 (Live Rep)</p> <hr/> <p> Business Hours</p> <hr/> <p> Online</p> <hr/> <p><input checked="" type="checkbox"/> In Person</p> <hr/> <p><input checked="" type="checkbox"/> Live Online</p> <hr/> <p><input checked="" type="checkbox"/> Webinars</p>	<p> 24/7 (Live Rep)</p> <hr/> <p> Business Hours</p> <hr/> <p> Online</p> <hr/> <p><input type="checkbox"/> In Person</p> <hr/> <p><input checked="" type="checkbox"/> Live Online</p> <hr/> <p><input checked="" type="checkbox"/> Webinars</p>	<p> 24/7 (Live Rep)</p> <hr/> <p> Business Hours</p> <hr/> <p> Online</p> <hr/> <p><input type="checkbox"/> In Person</p> <hr/> <p><input checked="" type="checkbox"/> Live Online</p> <hr/> <p><input checked="" type="checkbox"/> Webinars</p>