

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БУДІВНИЦТВА І
АРХІТЕКТУРИ**

Факультет автоматизації і інформаційних технологій

Кафедра управління проектами

ЗАТВЕРДЖУЮ

Завідувач кафедри

Сергій БУШУЄВ

“ ___ ” _____ 2023 року

**ПОЯСНЮВАЛЬНА ЗАПИСКА
ДО АТЕСТАЦІЙНОЇ РОБОТИ
НА ЗДОБУТТЯ ОСВІТНЬОГО СТУПЕНЯ МАГІСТРА**

**Дослідження та розробка засобів для міграції проекту на
іншу технологію**

(назва)

Виконав студент групи: Кангін Ілля Юрійович

(прізвище, ім'я та по батькові повністю)

Спеціальність: 073 “Менеджмент”

Освітня програма: Управління проектами

Керівник: Запривода Аліна Анатоліївна

(прізвище, ініціали.)

К.Т.Н., ДОЦ.

науковий ступінь, вчене звання

Рецензент: _____

(прізвище, ініціали.)

науковий ступінь, вчене звання

Київ 2023 р

ЗМІСТ

ЗМІСТ	1
ВСТУП	4
1 Теоретико-методологічні аспекти міграції проектів	5
1.1 Основні терміни та визначення	5
1.1.1 Життєвий цикл проекту міграції	5
1.1.2 Поняття аудиту проекту	6
1.1.3 С4 модель візуалізації архітектури проекту	7
1.1.4 Документація ІТ проекту	8
1.2 Методи та підходи до міграції програмних проектів	9
1.2.1 Повне переписування коду	10
1.2.3 Часткова або плавна міграція	12
1.3 Вплив міграції на функціональність та продуктивність	15
1.3.1 Покращення функціональності	15
1.3.2 Оптимізація продуктивності	15
1.3.3 Підвищення масштабованості	15
1.3.4 Збереження сумісності	16
1.3.5 Витрати на навчання та підтримку	16
1.3.6 Зміни у інтерфейсі та взаємодії з користувачем	16
1.3.7 Безпека та відновлення	17
1.3.8 Підтримка та обслуговування	17
1.4 Висновки до розділу	17
2 Виявлення вимог та ініціалізація міграційного проекту	19
2.1 Аналіз існуючого продукту	19

2.1.1 Бізнес-аудит	20
2.1.2 Дизайн аудит.....	23
2.1.3 Технічний аудит	25
2.2 Дослідження та позиціонування майбутнього продукту	27
2.2.1 Формулювання стратегії.....	28
2.2.2 Опитування користувачів.....	29
2.3 Визначення обсягу робіт	30
2.3.1 Приоритизація вимог	31
2.3.2 Визначення ресурсів	32
2.4 Створення міграційного плану проекту.....	33
2.4.1 Розробка графіка	33
2.4.2 Оцінка ризиків	34
2.4.3 Визначення комунікаційного плану	34
2.5 Ініціалізація проекту.....	35
2.5.1 Визначення усіх зацікавлених сторін.....	36
2.5.2 Кік-офф зустріч	37
2.6 Висновки до розділу	38
3 Управління проектом міграції веб-сайту на нову технологію	40
3.1 Проведення фази дослідження проекту	41
3.1.1 Формування команди для дослідження проекту	42
3.1.2 Визначення обсягу робіт	44
3.1.3 Проведення технічного аудиту	49
3.2 Створення плану реалізації проекту	52
3.2.1 Оцінка плану робіт.....	54
3.2.2 Створення ресурсного плану	54

3.3	Визначення зацікавлених сторін та їх впливу на проєкт	55
3.3.1	Ідентифікація зацікавлених осіб.....	56
3.3.2	Розставлення пріоритетів	56
3.3.3	Проведення класифікації.....	58
3.4	Ризик-менеджмент проєкту.....	58
3.4.1	Ідентифікація ризиків	59
3.4.2	Аналіз ризиків.....	60
3.4.3	Стратегія роботи з ризиками.....	60
3.4.4	Контроль та моніторинг ризиків.....	61
3.5	Операційна діяльність та гнучке планування проєкту.....	62
3.6	Повторний аналіз проєкту, звітування.....	64
3.7	Висновки до розділу	65
	Висновки	67
	Список джерел.....	Ошибка! Закладка не определена.

ВСТУП

За останні десятиліття в галузі розробки програмного забезпечення спостерігається стрімкий розвиток, що часто призводить до виникнення необхідності в міграції програмних проектів на інші технології. Більшість людей не уявляють, що насправді інформаційні продукти котрими вони користуються кожного дня постійно переписуються на нові технології, щоб зменшити час обробки інформації, оптимізувати кількість пам'яті та інше. Цей процес стає актуальним як для комерційних компаній, так і для органів державної влади та місцевого самоврядування. З ростом завдань і вимог до програмного забезпечення виникає потреба в розробці ефективних інструментів для проведення міграції проектів.

Метою даної магістерської роботи є дослідження та розробка засобів для ефективної міграції програмних проектів на інші технології. Основні завдання включають аналіз існуючих підходів до міграції, виявлення невдач та прорахунків у попередніх проектах, а також розробку рекомендацій та інструментів для полегшення цього процесу.

Об'єктом дослідження будуть реальні проекти котрі мали свою основною або вторинною задачею міграцію продукту на іншу технології, а предметом – засоби та методи, що використовуються для успішної реалізації цього процесу.

У дослідженні буде використана комбінація теоретичних методів (аналіз літературних джерел, систематизація і узагальнення інформації) та практичних підходів (розробка та тестування інструментів для міграції). Застосування інтегрованого підходу дозволить здійснити комплексний аналіз та розробку пропозицій для покращення процесу міграції. Більшість практичних підходів будуть складатися з моєї практики роботи на міграційних проектах.

1 ТЕОРЕТИКО-МЕТОДОЛОГІЧНІ АСПЕКТИ МІГРАЦІЇ ПРОЄКТІВ

1.1 Основні терміни та визначення

1.1.1 Життєвий цикл проєкту міграції

Життєвий цикл проєкту – це визначений період від його старту до завершення, який включає в себе ряд етапів та процесів, спрямованих на досягнення мети проєкту.[1] Такий цикл дозволяє організаціям систематично та контрольовано реалізовувати ідеї та досягати поставлених цілей.

1. Формулювання ідеї: Початковий етап, на якому визначається концепція та мета проєкту. Створюється загальне бачення та план виконання.

2. Старт та планування: Після узгодження ідеї, розпочинається офіційний старт проєкту. Тут визначається команда, розробляється детальний план робіт, встановлюються терміни та бюджет.

3. Виконання: На цьому етапі реалізується план, здійснюються всі необхідні роботи та завдання. Команда працює згідно з графіком та забезпечує виконання планованих результатів.

4. Моніторинг та контроль: Протягом усього процесу виконання важливо вести моніторинг прогресу, контролювати витрати та виявляти можливі відхилення від плану. Це дозволяє вчасно вживати заходів для корекції.

5. Повторний аналіз: По завершенні виконання проєкту проводиться порівняльний аналіз. Аналізуються досягнуті цілі, визначаються сильні та слабкі сторони, робляться висновки для майбутніх проєктів.

6. Завершення та передача: На останньому етапі проєкт формально завершується. Здійснюється передача результатів замовнику або власнику. Збирається документація та проводиться остаточний звіт.

Цей життєвий цикл надає структуру та організацію для керування проєктом, забезпечуючи ефективність, якість та відповідність поставленим завданням. Він дозволяє проєктній команді систематично взаємодіяти з усіма етапами розробки та контролювати процеси для досягнення успіху в умовах обмежень термінів та ресурсів.

1.1.2 Поняття аудиту проєкту

Поняття аудиту продукту – це комплексний процес оцінки та перевірки якості та ефективності продукту з метою виявлення його сильних та слабких сторін, а також можливостей для вдосконалення.[2] Аудит спрямований на те, щоб забезпечити відповідність продукту стратегії компанії, вимог користувачів та ринкових стандартів.

На початковому етапі аудиту продукту проводиться аналіз функціональності та властивостей продукту. Це включає в себе ретельне вивчення можливостей та обмежень, які надає продукт користувачам, а також виявлення потенційних проблем або несправностей. Дослідження також охоплює аспекти безпеки та стабільності продукту.

Далі, аудит продукту також включає в себе оцінку дизайну та користувацького інтерфейсу. Досвідчені дизайнери аналізують естетику, зручність використання та співвідношення елементів, спрямовуючись на покращення користувацького досвіду та відповідність до стандартів індустрії.

У процесі аудиту, великий акцент приділяється визначенню користувацьких історій. Це дозволяє зрозуміти, як користувачі взаємодіють з продуктом, що вони шукають та як можливо поліпшити їхні враження та задоволення від використання продукту.

Особлива увага приділяється технічному аспекту продукту. Проводиться аналіз архітектури, технічних рішень та інфраструктури, що використовуються. Оцінка технічних рішень допомагає визначити їхню ефективність, стабільність, актуальність а також необхідність до змін.

У контексті аудиту продукту обов'язково проводиться оцінка прибутковості. Це включає вивчення фінансових показників, пов'язаних з продуктом, аналіз затрат та прибутків. Такий аналіз допомагає визначити ефективність продукту з точки зору прибутковості.

Після збору даних та проведення всіх аспектів аудиту, формується детальний звіт, в якому виділяються ключові аспекти, виявлені проблеми та рекомендації для подальшого вдосконалення продукту. Цей звіт слугує основою для розробки стратегії вдосконалення та забезпечення високої якості продукту на ринку.

1.1.3 C4 модель візуалізації архітектури проєкту

C4 модель візуалізації архітектури проєкту - це ефективний підхід для створення зрозумілих та деталізованих зображень архітектури програмного забезпечення, які дозволяють команді та зацікавленим сторонам легше спілкуватися, розуміти та керувати аспектами системи.[3]

Модель C4 (Context, Containers, Components, Code) розроблена Саймоном Брауном та надає ієрархічну структуру для представлення архітектурних концепцій на різних рівнях деталізації.[3]

1. Контекст (Context): Цей рівень дозволяє розглядати систему як чорний ящик та визначає її зовнішнє середовище. Діаграма контексту ідентифікує основні компоненти системи, їх взаємодію та обмеження.

2. Контейнери (Containers): На цьому рівні фокус зосереджений на внутрішній структурі системи. Визначаються контейнери, такі як веб-додатки, мобільні додатки або бази даних, та показується їх взаємодія та залежності.

3. Компоненти (Components): Тут архітектурні компоненти розглядаються детальніше. Діаграми компонентів показують взаємодію між класами, модулями або сервісами.

4. Код (Code): Останній рівень, "Code", звертає увагу на реалізацію конкретних функціональних частин системи на рівні джерела. Використання діаграм коду дозволяє подробиць аналізувати реалізацію окремих компонентів.

С4 модель стала популярною завдяки своїй здатності надавати чітке та легко зрозуміле представлення архітектурних аспектів проєкту на різних рівнях деталізації. Вона сприяє комунікації між різними учасниками проєкту, робить можливим швидке розуміння та прийняття рішень, а також допомагає уникнути непорозумінь та невірних інтерпретацій у процесі розробки.

1.1.4 Документація ІТ проєкту

Документація ІТ-проєкту є важливим інструментом, який визначає та фіксує всі аспекти розробки, впровадження та експлуатації інформаційно-технічного продукту. Ця документація включає в себе різноманітні складові, що забезпечують зрозумілість, спілкування та керування процесами усього проєкту.[4]

1. Технічне завдання (ТЗ): Це вихідний документ, що містить визначення завдань, вимог та функціональності проєкту. [5]Технічне завдання є основою для всього процесу розробки та визначає обсяг робіт.

2. Архітектурна документація: Описує структуру та взаємозв'язки компонентів системи. Включає в себе діаграми, описи архітектурних рішень та принципів, що лежать в основі проєкту.

3. Кодова документація: Містить опис коду, коментарі, інструкції зі збирання та налаштування. Надає інформацію для розробників щодо внутрішньої структури коду та його функціональності.

4. Документація по тестуванню: Визначає тести, їх опис та результати. Забезпечує перевірку правильності та стабільності системи.

5. Документація з управління проектом: Містить плани проекту, графіки, розклади, звіти по виконанню, ризики та прийняття рішень.

6. Документація користувача: Забезпечує інструкції та рекомендації для кінцевих користувачів, допомагає їм розуміти та використовувати систему.

7. Документація по забезпеченню безпеки: Визначає заходи безпеки, політики та процедури, щоб гарантувати захищеність інформації та системи в цілому.

8. Документація з експлуатації: Містить інформацію про встановлення, налаштування та підтримку системи, яка допомагає управляти її ефективністю.

9. Аналітична та звітна документація: Включає в себе результати аналізу та звіти, які можуть використовуватися для оцінки ефективності проекту та прийняття рішень.

Ця різноманітна документація сприяє розумінню, комунікації та ефективному керуванню процесами в рамках ІТ-проекту, забезпечуючи відповідність вимогам та якість результату.

1.2 Методи та підходи до міграції програмних проектів

У певний момент часу, компанія котра стикається з питання міграції свого продукту на нову технологію зазвичай робить це через низку факторів:

- Незадовільні бізнес-показники;
- Зріст витрат на поточну розробку;
- Ускладнена підтримка та баг-фіксинг.

Першочергове визначення фактору, через який з'являється запит на міграцію проєкту допоможе менеджеру краще підібрати методи та підходи для започаткування процесу планування. Ми визначимо ці методи та підходи, а також їх основні принципи, переваги та недоліки. Для порівняння ми будемо розглядати такі фактори як планування, вартість, гнучкість, час реалізації та потенціал для майбутнього

1.2.1 Повне переписування коду

Повне переписування коду або рефакторинг є найбільш традиційним та прямолінійним методом міграції проєкту. Він включає у себе створення повністю нового коду на базі старого, зберігаючи існуючу функціональність.

У першу чергу такий підхід відрізняється тим, що їх досить легко планувати. У менеджера в доступі вже є працююче програмне забезпечення котрі виконує вимоги бізнесу та приносить прибуток. Таким чином, підрахунок витрат на нього займе малу кількість часу, особливо якщо кількість та якість документації відповідають високим стандартам. Першим кроком для оцінки такого проєкту буде створення User Story Map. Зазвичай ми можемо відрізнити три основні підходи:

- Bottom-up;
- Up-bottom;
- Big Bang;

На мою думку Bottom-up буде найбільш релевантним, бо ми будемо розпочинати визначення з найменших функціональних частинок, котрі потім можуть бути використані у більших за обсягом модулях. Це також допоможе у майбутньому плануванні та гнучкості такого проєкту. Наприклад, провівши рефакторинг компоненту форми вводу, вона може бути перевикористана в усіх модулях системи. Це дозволить працювати з таким продуктом як зі своєрідним конструктором.

Up-bottom використовує зворотній підхід. Він дозволяє швидше приступити до роботи, і виконувати процеси планування на ходу. Наприклад, першим елементом розробки буде функціонал сторінки оплати. У нього будуть входити такі компоненти як кнопка, текстове поле, форма вводу які зможуть бути використані у майбутньому функціоналі. Але, необхідно звернути увагу, що у порівнянні з bottom-up розробникам треба буде адаптувати компоненти у різному функціоналі, адже ми не будемо мати повний обсяг вимог. Наприклад, інформаційний блок на одній сторінці може складатися з 6 карток, а на іншій мати 8 карток.

Big Bang є найбільш хаотичним, і зазвичай використовується у випадках високої пріоритизації поставки певних частин функціоналу. Він є не-ефективним, і зазвичай використовується на проєктах з багатьма невідомими. У рамках даної магістерської роботи він розглядатися не буде.

Після визначення обсягу робіт, у менеджера проєкту предстає задача розробки та валідації архітектурного рішення. У першу чергу це робиться для того, щоб зрозуміти які технології будуть використовуватися на проєкті, та яких спеціалістів треба буде залучити. Зазвичай для виконання цієї задачі заучується архітектор, він буде визначати яке рішення буде достатнім для реалізації продукту, проведе аналіз існуючого рішення та проблем котрі у ньому присутні та як вони можуть бути вирішені. У будь-якому випадку, ставлення до такого проєкту має бути як до повністю нового та унікального. Чим більше буде залучатися існуючих рішень, тим більша ймовірність виникнення вже існуючих помилок та проблем.

Після готового архітектурного рішення та визначеного обсягу робіт, менеджер проєкту може розпочати процес планування. Визначити необхідну кількість спеціалістів, терміни та бюджет.

Продовжуючи аналізувати фінансову складову цього методу можна з впевненістю сказати що він буде найбільш коштовним. Основною причиною цього буде те, що замовнику буде необхідно наймати нову команду розробки,

котра буде мати необхідні знання для реалізації рішення. Але також, треба не забувати про те, що існуюча команда має залишитись для підтримки цього проекту що принесе додаткові витрати, а також розвиток цього проекту має бути заморожено. Якщо протягом рефакторингу у існуючий проект буде додаватися новий функціонал, це буде негативно впливати на терміни та бюджет реалізації проекту рефакторингу. В цьому випадку ми можемо зробити висновок, що такий проект не буде гнучким. Швидше за все, для того щоб внести зміни необхідно буде повністю його зупинити, передивитись обсяг робіт та внести зміни до плану та бюджету.

Покриваючи останні фактори оцінки цього методу, досить складно казати про час реалізації, але треба звернути увагу на те, що він стане прибутковим лише після релізу.

1.2.3 Часткова або плавна міграція

У сучасному світі, будь-які затримки не лише у роботі продукту, але й можливі перепони на його розвитку можуть бути дуже коштовні. Для цього існують більш гібридні способи міграції проектів на нові технології. Головною метою цих способів є поєднання вже існуючого додатку з частковим залученням нового. Але, такі міграційні проекти потребують більш уважного менеджменту та детального планування. У попередньому підрозділі ми вже розглянули методи переписування коду, тому у цьому ми сфокусуємось на тому як можливо ефективно поєднувати працездатність двох продуктів.

Наразі розрізняється достатня кількість стратегій паралельного розгортання продуктів котрі зазвичай використовуються для того, щоб плавно переводити користувачів на нову версію продукту. Нижче ми розглянемо декілька з них:

Під час паралельного накатування продуктів використовуються різні стратегії для гладкого переходу від однієї версії продукту до іншої,

забезпечуючи при цьому безперервну доступність для користувачів. Тут наведено кілька методів паралельного накатування продуктів:

А/В-тестування. Реалізація двох (або більше) версій продукту, які одночасно функціонують. Користувачі випадковим чином розподіляються між цими версіями, і ефективність оцінюється на основі поведінки користувачів. Така стратегія дозволяє збирати реальні дані про взаємодію користувачів з новою версією продукту, та робити порівняльний аналіз між двома версіями. Проте, в свою чергу у нас виникає великий ризик того, що користувачі версії «А» отримають нестабільну версію. Зазвичай, ця стратегія частіше використовується для маркетингових цілей.

Канарські релізи (Canary Releases). Випуск нової версії продукту для обмеженої групи користувачів (наприклад, внутрішнього персоналу або обраних клієнтів) перед повним випуском. Така стратегія дозволяє вирішити проблеми та зібрати відгуки в обмеженому середовищі перед випуском для всіх користувачів. Вимагає уважного управління доступом та контролю.

Поступовий реліз (Gradual Rollout). Випуск нової версії продукту для поступово зростаючого відсотка користувачів з плином часу. Дозволяє швидко виявляти проблеми та обмежити їх вплив на користувачів. Може вимагати уваги до деталей, оскільки розгортання поступово відбувається для різних груп.

Сині-зелені релізи (Blue-Green Deployments). Використання двох окремих середовищ (синього та зеленого) для випуску продукту. Одне середовище активне, а інше – готове для нового випуску. Забезпечує миттєву зміну на нову версію та можливість швидко повертатися до попередньої версії у випадку проблем. Вимагає подвійних ресурсів для утримання двох окремих середовищ.

Тіньове розгортання (Shadow Deployment). Створення копії нової версії продукту, яка працює паралельно з основною версією, але не призначена

для кінцевих користувачів.[6] Дозволяє збирати дані та оцінювати продуктивність нової версії без впливу на активних користувачів. Зменшує ризик впливу на користувачів, дозволяючи вивчати та вдосконалювати нову версію в умовах реального середовища. Може вимагати додаткових ресурсів для підтримки та управління тіньовим середовищем. Тіньове розгортання є важливим методом для аналізу та тестування нових версій продукту перед їхнім повним випуском. Цей підхід дозволяє зменшити вплив на користувачів, одночасно забезпечуючи можливість збирати важливі дані щодо ефективності та стабільності нового релізу.

Ці методи дозволяють розгортати нові версії продуктів паралельно із збереженням безперервності роботи та вивченням реальних відгуків користувачів. Необхідно додати, що у випадку часткової міграції, ці стратегії будуть використовуватися не для усього продукту, а до певних функціональних частин продуктів. Наприклад, для мікро-сервісних додатків, кожен з сервісів після його адаптації буде окремо розгортатися та тестуватися.

Розпочинаючи фазу планування такого проекту, ми стикаємось з необхідністю правильного визначення пріоритетів задач та залучення необхідних ресурсів. У порівнянні з повним переписуванням коду у менеджера нема необхідності у створенні повномасштабної команди. Розпочинаючи з малих елементів, можна розпочати із залучення лише двох-трьох розробників, поступово збільшуючи кількість спеціалістів на новому проекті, або залучати розробників з оригінального проекту методом навчання їх новим технологіям.

Однією з найбільших переваг цього методу є те, що розробка нового функціоналу буде невпинною. Хоч на початку, функціонал буде додаватися до старого додатку він також буде збільшувати обсяг робіт. З часом, коли більша частина продукту буде переписана, новий функціонал буде з'являтися вже безпосередньо у новій версії, залишаючи лише команду підтримки на старій версії. На мою думку, планування цього процесу дозволяє менеджеру проекту

проявити свою креативність, і кількість поєднання усіх змінних може бути безмежною. Завдяки такій гнучкості, бюджет такого проєкту буде більш адаптивним і не буде лінійно залежати від вже існуючого.

1.3 Вплив міграції на функціональність та продуктивність

Як і з будь-яким іншим проєктом, при його визначенні необхідно зрозуміти які цілі і задачі перед ним будуть стояти. Для цього проводяться аудити різного ступеня, котрі будуть визначати на які аспекти проєкту треба звернути увагу

1.3.1 Покращення функціональності

Міграція може дозволити використовувати нові функціональні можливості та інструменти, які можуть покращити функціональність додатку. Заміна застарілих компонентів на більш сучасні може дозволити використовувати нові API, бібліотеки або фреймворки для розширення функціональних можливостей.

1.3.2 Оптимізація продуктивності

Міграція може призвести до покращення продуктивності через оптимізацію коду, використання ефективніших алгоритмів, та усунення проблем з надмірним використанням ресурсів. Заміна застарілих алгоритмів або використання асинхронного програмування може покращити швидкодію та реактивність системи.

1.3.3 Підвищення масштабованості

Нові технології можуть дозволити легше масштабувати додаток в залежності від зростання навантаження. Використання контейнеризації та

оркестрації може полегшити горизонтальне масштабування, дозволяючи динамічно розподіляти навантаження між різними екземплярами додатку.

1.3.4 Збереження сумісності

Під час міграції може знадобитися врахувати сумісність з існуючими даними або іншими системами, щоб уникнути проблем при інтеграції.

1.3.5 Витрати на навчання та підтримку

Велика міграція може вимагати часу та ресурсів на навчання команди та користувачів новим технологіям. Перехід від одного фреймворку до іншого може вимагати часу для освоєння нового стеку технологій та опанування нових інструментів.

І навпаки, збереження застарілої технології займає занадто багато часу на розробку нового функціоналу та його підтримку.

1.3.6 Зміни у інтерфейсі та взаємодії з користувачем

Міграція може вплинути на інтерфейс користувача та способи взаємодії. Важливо розглядати, як ці зміни можуть вплинути на зручність використання та задоволення користувачів. Перехід від класичного веб-інтерфейсу до інтерфейсу з використанням нових технологій, таких як SPA (односторінкові застосунки), може потребувати пристосування користувальницького досвіду та вивчення нових методів взаємодії. Завдяки тому, що додаток буде піддаватися повній технологічній реконструкції, він також може бути і реконструйований з точки зору UI/UX практик.

1.3.7 Безпека та відновлення

Нові технології можуть представляти вищий рівень безпеки та забезпечувати кращі механізми відновлення. Використання сучасних методів шифрування або механізмів автоматичного відновлення може покращити захист від несанкціонованого доступу та зменшити час відновлення після виникнення проблем.

1.3.8 Підтримка та обслуговування

Міграція може вплинути на процеси підтримки та обслуговування, включаючи необхідність навчання персоналу та зміни процесів підтримки. Нові інструменти або архітектурні рішення можуть вимагати нових підходів до вирішення проблем та виконання рутинних завдань.

Вплив міграції на функціональність та продуктивність є складним процесом, який потребує детального планування, валідації та співпраці всіх зацікавлених сторін. Важливо враховувати конкретні потреби та вимоги проекту, а також найкращі практики в обраній технології для досягнення успішного результату.

1.4 Висновки до розділу

Головною задачею даного розділу було ретельно розглянути основні терміни та визначення, пов'язані з життєвим циклом проекту міграції. Такі поняття як аналіз понять аудиту проекту, S4 модель візуалізації архітектури проекту та документації IT проекту допомагає краще зрозуміти контекст у котрому знаходяться такі типи проектів. Були визначені їх вплив на проект, та необхідність їх розуміння для ефективної поставки такого проекту.

Крім того, у даному розділі були детально описані різні методи та підходи до міграції програмних проєктів. Ці методи визначають підходи для першочергового аналізу та визначення обсягу робіт. Не менш критичним є стратегії поставки продукту до фінального користувача. Без сумніву, кожна з цих стратегій має свої сильні та слабкі сторони, але вони є невід'ємною складовою міграційних проєктів. Ця інформація дозволяє визначити оптимальний шлях міграції в залежності від конкретних умов та завдань проєкту.

Особлива увага приділена впливу міграції на функціональність, продуктивність, масштабованість та інші якісні метрики або задачі котрі мають на меті загальне покращення якості продукту.

Загалом, розділ надає чіткий та комплексний огляд теоретичних та методологічних аспектів міграції проєктів, що стане корисним для фахівців у галузі IT-управління та розробки, сприяючи успішній реалізації та управлінні міграційними проєктами.

2 ВИЯВЛЕННЯ ВИМОГ ТА ІНІЦІАЛІЗАЦІЯ МІГРАЦІЙНОГО ПРОЕКТУ

У 2000х роках, розпочався великий бум постачання ІТ послуг компаніям різного рівня. У ті часи, використовувалися наявні технології і зазвичай були сфокусовані на постачанні продукту високої якості, та у наявний бюджет. Протягом 20 років, з'явилися багато нових мов програмування та технології, котрі на голову вище за ті котрі існували раніше. Таким чином, зараз ІТ індустрія переживає новий етап, коли вже створене програмне забезпечення потребує переосмислення як технічного так бізнесового.[7]

Загалом дану фазу ми будемо називати дослідницькою (Discovery). Вона дозволяє нам визначити першочерговий план проєкту, та надати сумніву його необхідність. Протягом цієї фази команда не буде розробляти програмне забезпечення, а лише оцінювати проєкт та відповідати на питання:

- Які задачі та цілі цього проєкту?
- Чи є необхідність у цьому проєкті?
- Яка вартість такого проєкту?
- Яку бізнес цінність зможе принести проєкт?
- Чи буде такий проєкт фінансово спроможним?

2.1 Аналіз існуючого продукту

Перед тим як розпочати будь-який міграційний проєкт, необхідно ретельно проаналізувати поточний стан існуючого продукту. Бізнес-аудит дозволить нам глибше зрозуміти функціонування, сильні та слабкі сторони, а також визначити ключові аспекти, які варто покращити чи оптимізувати. В залежності від результатів цього аудиту команда проєкту зможе визначити загальний обсяг робіт, котрий буде необхідний для того щоб його реалізувати.

У ідеальному світі, бізнес-аналітик, котрий відповідає за цей аудит, має просто скласти обсяг з вже існуючого продукту. Але, у більшості випадків, коли замовник або компанія приймає рішення щодо міграції продукту, вони не хочуть щоб він був обмежений вже існуючим функціоналом. У більшості випадків, продукти котрі потребують таких змін вже будуть мати низку задач та функціоналу котрий було неможливо реалізувати раніше через протиріччя в бізнес-логіці, обмеження технології або занадто великі витрати для реалізації. Також, це гарний час для того, щоб відмовитись від вже непотрібного функціоналу, котрий, наприклад, мав проблеми з сумісністю до іншого.

2.1.1 Бізнес-аудит

Бізнес-аудит є ключовою складовою успішної ідентифікації цілей та задач проєкту. Як було визначено раніше, за цю задачу відповідає БА (бізнес аналітик), котрий може залучитися до цього проєкту раніше за інших спеціалістів, та буде основним джерелом знань. При традиційних проєктах, процес аналізу розпочинається з проведення інтерв'ю з зацікавленими сторонами, створення опитувань та інше. Але специфіка проєктів міграції полягає у тому, що об'єкт рефакторингу вже існує. Таким чином, БА розпочинає свою роботу з приміряння на себе роль користувача програмного забезпечення. Це дозволяє йому оцінити увесь присутній функціонал, та створити першочергову версію мапи користувацьких історій.

Кращою практикою використання такої мапи є декомпозиція її елементів на два, та у певних випадках, аж до чотирьох рівнів. Вони можуть бути класифіковані наступним чином:

- Theme
- Initiative
- Epic
- Story

Кожен з цих рівнів буде критичним для того, щоб зробити групування її найменших елементів, та перейти до етапу валідації з усіма зацікавленими особами.

Наступним етапом у роботі БА на проєкті буде детальний огляд бізнес процесів. У процесі детального огляду бізнес-процесів для існуючого продукту, першим етапом є ідентифікація цих процесів, що виконується шляхом інтерв'ю з різними відділами компанії та аналізу вже наявної документації. Організація зустрічей із ключовими представниками відділів дозволяє отримати об'єктивні дані та визначити перелік основних бізнес-процесів, які детально аналізуватимуться.

Після ідентифікації бізнес-процесів команда здійснює детальний огляд кожного з них. Для кожного процесу фіксуються кроки, учасники, використовувані інструменти та ресурси. Важливим етапом є створення діаграм потоку робіт, які візуалізують послідовність етапів кожного бізнес-процесу. Це дозволяє команді отримати чітке уявлення про роботу кожного процесу та виділити можливі області для покращень.

Далі здійснюється аналіз ефективності кожного бізнес-процесу. Визначаються ключові показники ефективності, такі як час виконання, вартість та якість результатів. Під час огляду спрямовується увагу на ідентифікацію можливих проблем у функціонуванні бізнес-процесів, таких як затримки, невідповідність вимогам чи незадовільна взаємодія між модулями. Здійснюється порівняння цих показників із заданими стандартами та очікуваннями, а також порівняння з конкурентами. Остання складова є більш критичної аніж здається. Ринок ділиться між схожими продуктами і визначає як продукт замовника виглядає на їх фоні. Певний функціонал може бути кращим, а певний – гіршим. Завдяки визначенню слабких та сильних сторін, можливо визначити зони для розвитку, стабілізації або оптимізації.

Визначивши показники ефективності, аналітик може перейти до оцінки прибутковості. В процесі оцінки прибутковості для існуючого продукту

важливо ретельно проаналізувати його фінансовий стан та визначити чинники, що впливають на прибуток. Починаючи зі збору фінансових даних, команда фінансистів, маркетологів та аналітиків ретельно досліджує комерційні показники продукту, підготовлює звіт про прибуток і збитки. Це дозволяє зрозуміти структуру доходів та витрат, а також виявити потенційні точки для покращення прибутковості. Враховуючи різні аспекти, такі як витрати на виробництво, маркетингові витрати, та витрати на обслуговування клієнтів. Здійснюється розрахунок оптимальної цінової стратегії, що відображає конкурентоспроможність продукту на ринку. Оцінка включає аналіз рентабельності інвестицій, оскільки важливо враховувати витрати на розробку та підтримку продукту. Розрахунок показників, таких як ROI (Return on Investment), дозволяє зрозуміти, наскільки ефективно використовуються кошти, вкладені в продукт. Для повного зору на прибутковість, важливо враховувати долю ринку та перспективи росту. Аналізуючи частку ринку, команда може визначити, наскільки продукт задовольняє потреби клієнтів у порівнянні з аналогічними пропозиціями конкурентів.[8] Прогнозуючи перспективи росту, можна визначити можливості для збільшення прибутковості через розширення частки ринку та введення продукту на нові ринки або імплементацію нового функціоналу.

Завершальним етапом бізнес-аудиту є формування детального звіту та рекомендацій. Готується презентація, в якій висвітлюються ключові аспекти детального огляду бізнес-процесів, представляються результати аналізу та надається стратегія подальших дій для впровадження виявлених покращень. Цей процес допомагає визначити оптимальні шляхи вдосконалення бізнес-процесів та підвищення ефективності компанії. Також цей звіт включає в себе всі аспекти оцінки прибутковості, результати аналізу та конкретні стратегії для оптимізації прибутковості. Під час презентації результатів команді та зацікавленим сторонам, команда розкриває ключові аспекти, виявлені можливості та плани дій для подальшого вдосконалення. Цей комплексний

підхід дозволяє зробити інформовані рішення для підвищення прибутковості та якості продукту.

Після аналізу усіх даних, зацікавлені сторони проєкту можуть прийняти рішення щодо того, чи є необхідність у розвитку бізнес-складової цього проєкту. Може виглядати, що такий обсяг робіт буде занадто великим, але на кожному з етапів відбувається щоденний контроль виконання та результативності. У будь-якому випадку, уся робота котру робить БА на дослідницькій фазі допомагає зекономити гроші у майбутньому. Чим більше факторів та елементів буде покрито, тим легше буде оцінювати реальний обсяг робіт, та потенційно економити бюджет на команду розробки.

2.1.2 Дизайн аудит

UI/UX дизайн-аудит є додатковим етапом у розробці та вдосконаленні продуктів, спрямованих на максимальне задоволення користувачів. [9] Цей процес включає в себе глибокий аналіз інтерфейсу та взаємодії з користувачем для виявлення поточних проблем та можливостей для поліпшення.

Початковим етапом є збір ключової інформації про продукт, включаючи його цільову аудиторію, основні функціональність та мету існування. Дизайн-аудит вимагає ретельного вивчення документації, проведення інтерв'ю з командою розробників та користувачами, щоб отримати повний обсяг інформації. Після завершення збору інформації та аудиту продукту, фаза аналізу інтерфейсу визначає ефективності та зручності використання. Досвідчені дизайнери вдаються до детального розгляду кожного елемента користувацького інтерфейсу з метою визначення його естетичних якостей, зручності та процесу взаємодії з іншими складовими інтерфейсу.

У процесі аналізу особлива увага приділяється зручності використання. Дизайнери оцінюють, наскільки інтуїтивно користувач може взаємодіяти з різними елементами та функціями продукту. Це включає в себе експертну

оцінку логіки розташування елементів, навігації та загальної користувацької дружелюбності. Найбільший акцент робиться на забезпеченні ефективної навігації для кінцевого користувача. Дизайнери стежать, щоб підходи до розташування елементів допомагали спростувати процес переміщення по продукту, роблячи його зручним та легким у використанні. Аналіз інтерфейсу покликаний виявити потенційні проблеми та надати рекомендації щодо покращення. Цей етап визначає, наскільки дизайн відповідає вимогам користувачів та чи можливі поліпшення, спрямовані на створення продукту, який буде максимально привабливим, зрозумілим та зручним для своєї цільової аудиторії.

Важливо також враховувати відповідність дизайну корпоративному стилю та брендовому образу. Дизайн-аудит допомагає переконатися, що весь інтерфейс продукту виражає єдиний стиль та сприймається користувачем згідно з брендovими цінностями. Безпосереднім етапом такого аналізу є визначення новітності системи та відповідності до міжнародних стандартів. Дизайн сфера постійно розвивається, та її тренди змінюються з часом. Те що виглядало свіжим 5-10 років тому, може виглядати застарілим наразі.

Після завершення збору даних та проведення аналізу, команда дизайнерів переходить до створення детального звіту, який не лише виявляє проблеми в інтерфейсі продукту, але й пропонує конкретні рекомендації для їх вирішення та поліпшення. Одним із ключових етапів в цьому процесі є створення дизайнерського. Демонструються концепції, які враховують усі виявлені проблеми та відповідають стратегічним цілям продукту. Це може включати в себе перегляд структури інтерфейсу, оптимізацію навігації та введення нових елементів дизайну для поліпшення користувацького досвіду. Створення концепту також означає врахування останніх тенденцій у дизайні та інноваційних рішень, які можуть бути використані для вирішення існуючих проблем. Це дозволяє дизайнерам підібрати кращі практики та ідеї, які допоможуть вдосконалити продукт та зробити його більш

конкурентоспроможним на ринку. Даний звіт, підготовлений командою, є своєрідним планом дій для подальшої розробки та оптимізації інтерфейсу. Кожна рекомендація має свій концептуальний фундамент, що враховує не тільки виявлені проблеми, але й стратегічні цілі продукту та відповідь на зворотній зв'язок користувачів.

2.1.3 Технічний аудит

Аналіз архітектури, технічних рішень та інфраструктури продукту є важливим етапом в процесі розробки та вдосконалення програмного забезпечення. Команда розробників та архітекторів систем проводить глибокий огляд всіх технічних аспектів продукту з метою визначення ефективності, масштабованості та стабільності.[10] На етапі аналізу архітектурних рішень команда розробників проводить глибокий огляд внутрішньої структури та організації програмного продукту. Основна мета цього дослідження – визначити, наскільки ефективно та згідно з вимогами розроблено архітектуру системи.

Під час розгляду логічної структури системи, аналізуються різні компоненти, їхні взаємозв'язки та взаємодія. Наприклад, якщо продукт є великою монолітною системою, може виникнути питання про її масштабованість та гнучкість у внесенні змін. З іншого боку, мікро-сервісна архітектура може бути обрана для полегшення розгортання та розвитку окремих компонентів незалежно. Архітектурний аналіз допомагає виявити можливості для подальшого вдосконалення системи. Наприклад, якщо під час аналізу виявлено слабкі сторони в логічній структурі, можна розглянути реорганізацію компонентів для покращення ефективності та зниження залежностей. Прикладом може служити аналіз архітектури веб-додатку, де основний компонент відповідає за обробку запитань користувачів, а додаткові мікро-сервіси відповідають за авторизацію, зберігання даних та інші

функціональності. Шляхом аналізу можна виявити, які саме сервіси потребують оптимізації, щоб підняти загальну продуктивність системи.

В цілому, аналіз архітектурних рішень визначає напрямки подальшого вдосконалення системи та забезпечує відповідність архітектури сучасним вимогам та стратегіям розвитку продукту. Аналіз технічних рішень у продукті визначає стратегічні аспекти вибору технологій, інструментів та мов програмування, які використовуються для створення його функціональності. Цей процес необхідний для відповідності до сучасних вимог та тенденцій у технологічному світі. Один із аспектів, що піддається огляду, - це вибір технологій. Наприклад, якщо продукт пов'язаний із веб-розробкою, важливо визначити, чи використовуються найновіші стандарти веб-технологій, такі як фреймворки React або Angular. Аналіз вибору технологій допомагає зрозуміти, наскільки сучасними та конкурентоспроможними вони залишаються. Приклади вибору оптимальних технологій та компонентів грають ключову роль у формуванні успішних стратегій, спрямованих на покращення продукту та його відповідність вимогам та очікуванням бізнесу.

Огляд інфраструктури, що підтримує продукт, є не менш критичним етапом в аналізі, оскільки це визначає готовність системних компонентів протистояти вимогам та завданням продукту. Цей аспект охоплює аналіз різноманітних складових, таких як серверні архітектури, бази даних та системи моніторингу. Серверні архітектури стають об'єктом уваги, коли розглядається ефективність обслуговування запитань та завдань, що надходять від користувачів. Наприклад, використання розподіленої серверної архітектури може покращити масштабованість системи, дозволяючи легко додавати або зменшувати ресурси відповідно до обсягів роботи. Оцінка баз даних грає ключову роль у визначенні продуктивності та надійності системи. Наприклад, використання реляційних баз даних може бути доцільним для забезпечення структурованої та консистентної обробки даних, тоді як нереляційні бази можуть бути використані для швидкого збереження та

отримання великих обсягів неструктурованої інформації. Системи моніторингу також займають важливе місце при оцінці інфраструктури. Вони дозволяють вчасно виявляти та вирішувати проблеми, а також вдосконалювати продукт. Наприклад, встановлення системи моніторингу взаємодії з серверами дозволяє оперативно виявляти відмови та уникати втрати продуктивності. Оцінка масштабованості та надійності інфраструктурних компонентів визначає готовність системи протистояти зростанню обсягу роботи. Наприклад, якщо велика кількість користувачів одночасно взаємодіє з продуктом, оцінка масштабованості може визначити, наскільки ефективно інфраструктура розподіляє ресурси для забезпечення стабільності та продуктивності. А такі інструменти як менеджери навантаження допомагають розробляти гібридні системи поставки продукту до кінцевого користувача. Це є визначним фактором для проєктів міграції та дозволяє уникнути велику кількість конфліктів та часу непрацездійності продукту.

Підсумковий аналіз дозволяє команді визначити сильні та слабкі сторони технічного стеку продукту. На його основі формується план вдосконалення, включаючи можливі технічні оновлення, оптимізації та модернізації. Аналіз архітектури, технічних рішень та інфраструктури стає фундаментом для стратегічного розвитку продукту, забезпечуючи його готовність до майбутніх викликів та конкурентних вимог ринку.

2.2 Дослідження та позиціонування майбутнього продукту

Дослідження та позиціонування майбутнього продукту є важливим етапом в розробці стратегії, оскільки воно дозволяє зрозуміти ринкові умови, визначити конкурентні переваги та визначити унікальні особливості, які зроблять продукт привабливим для цільової аудиторії.

На початковому етапі дослідження аналізуються ринкові тенденції та поведінка споживачів. Це включає в себе збір та аналіз даних щодо попиту на схожі продукти, ідентифікацію потреб споживачів та оцінку їхніх уподобань. Наприклад, у сфері мобільних додатків дослідження може включати аналіз популярних функцій, які користувачі шукають, та їхніх переваг. Після того, як ринкове дослідження проведено, визначається позиціонування продукту. Це означає визначення того, як продукт буде сприйматися в очах споживачів порівняно з конкурентами. Наприклад, якщо виявлено, що на ринку відсутній продукт, який задовольняє певну потребу, це може стати ключовою перевагою та визначити унікальність продукту. Стратегія позиціонування включає визначення цільової аудиторії, врахування конкурентного середовища. Це включає вивчення сильних та слабких сторін конкурентів, визначення їхніх стратегій та підходів до ринку. Наприклад, якщо існуючі продукти мають обмежені можливості, новий продукт може зосередитися на розширенні функціональності.

Узагальнюючи, дослідження та позиціонування майбутнього продукту є стратегічним процесом, який базується на глибокому розумінні ринкових умов, потреб споживачів та конкурентного середовища та будується на базисі попередньо проведених досліджень. Цей етап визначає основні напрямки розвитку продукту, його унікальність та привабливість для цільової аудиторії.

2.2.1 Формулювання стратегії

Формулювання стратегії — це складний та визначальний етап в розвитку будь-якого продукту чи підприємства.[11] Це процес, який вимагає виваженості, врахування різноманітних факторів та вирішення ключових питань, спрямованих на досягнення визначених цілей.

На першому етапі формулювання стратегії проводиться глибокий аналіз внутрішнього та зовнішнього середовища. Це включає в себе оцінку внутрішніх проблем, сильних та слабких сторін продукту, а також вивчення

можливостей та загроз у зовнішньому оточенні. Далі визначаються основні цілі та завдання стратегії. Це може включати в себе досягнення конкретного рівня прибутку, розширення ринкової частки або впровадження нових продуктів на ринок. Наприклад, стратегія розвитку нового продукту може передбачати виходження на ринок з інноваційною пропозицією, що задовольняє невиконані потреби споживачів.

Сформульовані цілі деталізуються на конкретні дії та тактичні кроки. В цьому етапі розробляється план дій, який визначає, як саме будуть досягнуті визначені цілі. Це включає у себе стратегії розширення ринкової частки можуть бути розроблені плани щодо маркетингових кампаній, партнерських угод або розширення асортименту продукції. Ключовим елементом формулювання стратегії є також управління ризиками та визначення контрольних точок. Розробляються механізми виявлення та реагування на можливі зміни у внутрішньому та зовнішньому середовищі, щоб забезпечити адаптивність стратегії. Ці кроки визначають загальний курс розвитку продукту. Воно базується на глибокому аналізі, визначенні конкретних цілей та тактичних кроків для їх досягнення, а також на гнучкості та готовності реагувати на зміни в середовищі.

2.2.2 Опитування користувачів

Опитування користувачів є важливим етапом в розробці продукту чи послуги, оскільки воно надає цінні відгуки та внутрішню інформацію, що допомагають вдосконалити та адаптувати продукт до реальних потреб користувачів.

Починається опитування з ретельного розроблення плану, який визначає цілі та завдання опитування, а також вибір методології опитування, яка може бути онлайн-анкетуванням, фокус-групами, інтерв'ю чи комбінацією різних методів. Сформульовані питання повинні бути чіткими, конкретними та спрямованими на отримання корисної інформації. Наприклад, якщо ціль -

визначити, які функції додатку є найбільш корисними для користувачів, питання може бути сформульоване так, щоб з'ясувати, які саме функції вони використовують частіше та які вони вважають найважливішими. Опитування може включати не лише питання на тему конкретних функцій чи властивостей, але і щодо загального задоволення від користування продуктом, ефективності його вирішення конкретних завдань, а також сприйняття дизайну та інші аспекти.

Збір та аналіз отриманих відповідей важливий для визначення ключових патернів та тенденцій, що дозволяє виокремити пріоритетні напрямки для подальшого вдосконалення продукту. У сучасному світі існують багато інструментів для збору кількісних метрик взаємодії користувача з продуктом, після його згоди з ним. На основі цих даних будуються метрики конверсії, теплова мапа продукту та інше. Ця інформація допомагає отримати найбільшу вибірку інформації котра буде доповнювати загальний аналіз.

2.3 Визначення обсягу робіт

Визначення обсягу робіт передбачає чітке визначення завдань, ресурсів та строків для досягнення конкретних цілей. Цей процес забезпечує розуміння того, що саме потрібно виконати, а також обговорюється із зацікавленими сторонами, забезпечуючи узгодженість очікувань.

Початок визначення обсягу робіт включає в себе узгодження з клієнтом чи внутрішнім замовником щодо основних вимог та цілей проекту. Наприклад, якщо мова йде про створення програмного продукту, узгодження може включати в себе визначення ключових функцій, платформ, підтримуваних пристроїв, та інших технічних аспектів. Після узгодження основних параметрів проекту проводиться розширений аналіз конкретних завдань та дій, необхідних для виконання. Це включає визначення функціональних вимог, технічних деталей та інших ключових елементів проекту. Наприклад, у

веб-розробці це може означати визначення конкретних сторінок, функцій та інтерактивних елементів, які повинні бути реалізовані.

Задачею визначення обсягу робіт також є врахування ресурсів, які будуть задіяні у проекті. Це може включати в себе оцінку кількості розробників, дизайнерів, тестувальників та інших фахівців, а також визначення потреб у технічних засобах та інструментах. Важливим елементом є також встановлення строків для виконання робіт. Це допомагає створити графік проекту та визначити, коли очікувати виконання певних задач на проєкті. Забезпечуючи конкретні терміни, можна ефективно керувати робочим процесом та уникнути можливих затримок.

2.3.1 Приоритизація вимог

Після створення обсягу робіт, команда переходить до приоритизації вимог, дозволяє визначити, які функції та характеристики мають вищий пріоритет та мають бути реалізовані в першу чергу. Цей процес допомагає ефективно розподілити ресурси та забезпечити максимальну користь для користувачів та бізнесу.

На початковому етапі приоритизації вимог проводиться збір інформації щодо всіх вимог, які мають бути враховані в проєкті. Це включає в себе вимоги користувачів, бізнес-вимоги, технічні вимоги та інші чинники, які впливають на розробку. Ми розглянули їх на етапах проведення різних аудитів продукту. Дані вимоги оцінюються за їхньою важливістю та впливом на проєкт. Важливість може визначатися з точки зору бізнесу, користувачів чи технічних обмежень. Наприклад, функціональність, яка прямо впливає на основні цілі бізнесу або вирішує критичні потреби користувачів, може мати вищий пріоритет. Зазвичай використовуються методи приоритизації, такі як MoSCoW (Must – має бути, Should – повинно бути, Could – може бути, Would – можливо буде), або рейтингові шкали, щоб систематизувати та визначити пріоритети для кожної вимоги. Це допомагає уникнути непорозумінь та

конфліктів у подальшій розробці. Після визначення пріоритетів формується список вимог у порядку їхнього виконання. Це стає основою для розроблення плану проекту та визначення термінів для введення кожної вимоги. Наприклад, якщо певна функціональність має високий пріоритет, її розробка може бути запланована на ранні етапи проекту. Але слід також зважувати аспекти поєднання та взаємозалежності задач. У певних випадках реалізація певних модулів буде заблокована реалізацією іншого, та її виконання може призвести до незапланованих витрат та затримок.

2.3.2 Визначення ресурсів

Від визначення ресурсів залежить ефективне використання та розподіл необхідних засобів для досягнення поставлених цілей. Цей процес включає в себе ідентифікацію та врахування усіх потрібних ресурсів, таких як людські, матеріальні, фінансові та інші, щоб забезпечити успішну реалізацію проекту.

На початковому етапі визначення ресурсів роздивляються аналіз вимог проекту, та визначений обсяг робіт. Це дозволяє зрозуміти, які ресурси будуть необхідні для їх виконання. Визначається потреба у людських ресурсах. Це включає в себе визначення кількості та кваліфікації співробітників, які займуться реалізацією проекту. Також враховується можливість використання внутрішніх або зовнішніх ресурсів, таких як аутсорсинг. Окрім людських ресурсів, визначаються матеріальні та технічні ресурси. Це може включати в себе обладнання, програмне забезпечення, інструменти та інші засоби, необхідні для виконання завдань проекту. Наприклад, для розробки високотехнологічного пристрою може бути необхідне спеціалізоване обладнання та програмувальне забезпечення. Фінальним етапом є розроблення ресурсного плану, включаючи розподіл завдань та визначення термінів. Це допомагає забезпечити оптимальне використання ресурсів та уникнення можливих затримок.

2.4 Створення міграційного плану проекту

Загальний план продукту буде складатися з обсягу робіт, оцінок цих робіт, коштів, ресурсного плану, ризиків та додаткових документів.

2.4.1 Розробка графіка

Розробка графіка проекту є важливим етапом, який визначає весь хронологічний порядок та часові рамки виконання різних завдань у межах проекту. Цей процес допомагає створити чіткий та структурований графік, що дозволяє всім учасникам проекту зрозуміти, коли та які кроки повинні бути виконані.

У розробку графіка входить визначення вхідних даних, ресурсів, технічних вимог та визначення кінцевих цілей. Обов'язково встановлюються залежності між різними завданнями. Це означає визначення того, як виконання одного завдання впливає на інші та установлення порядку їх виконання. Наприклад, для розробки програми може бути важливим завершити етап написання вимог перед тим, як розпочати розробку. Після встановлення залежностей визначається приблизний час та оцінка часу виконання кожного завдання. Це може включати в себе визначення термінів, які призначаються кожному етапу роботи та забезпечують достатньо часу для його виконання.

Після цього створюється графік, який представляє послідовність та тривалість різних завдань у вигляді діаграми. Такий графік називається Гантта, і він візуально відображає, як різні частини проекту взаємодіють та доповнюють одна одну в часі. Узагальнюючи, розробка графіка проекту - це систематичний підхід до організації та визначення хронології робіт, що допомагає забезпечити ефективне та вчасне виконання проекту, а також уникнення затримок та конфліктів в часі.

2.4.2 Оцінка ризиків

Оцінка ризиків є важливим етапом у керуванні проектами та дозволяє передбачити та підготуватися до можливих негативних подій, які можуть виникнути в ході виконання проекту.

На початку процесу проводиться ідентифікація потенційних ризиків. Це включає в себе визначення всіх можливих загроз для проекту, таких як технічні проблеми, непередбачені зміни в вимогах, проблеми з ресурсами або зміни в зовнішньому середовищі. Наприклад, в проекті розробки програмного забезпечення може існувати ризик непередбаченої зміни технологічних вимог, яка може вплинути на графік та бюджет. Далі визначається ймовірність виникнення кожного ризику та його потенційні наслідки. Оцінка ймовірності враховує ймовірність виникнення події та можливі наслідки в разі її реалізації. Наприклад, якщо існує ризик зміни в законодавстві, це може мати високі наслідки для проекту, і ймовірність виникнення такої зміни може бути визначена на високому рівні. Після цього проводиться оцінка впливу ризиків на проект та визначення їхнього рівня прийнятності. Ризики класифікуються відповідно до їхнього важливості та можливого впливу на проект. Це допомагає визначити, на які ризики слід зосередити увагу та розробляти стратегії управління ними. В результаті оцінки ризиків розробляються стратегії управління ризиками. Це може включати в себе розробку планів реагування на кризові ситуації, визначення запасних варіантів або навіть створення резервного фонду для заспокоєння можливих фінансових ризиків.

2.4.3 Визначення комунікаційного плану

Від ефективності взаємодії та обміну інформацією між учасниками залежить успіх проекту. Комунікаційний план встановлює рамки та стратегії спілкування, спрямовані на забезпечення чіткості, відкритості та ефективності обміну інформацією.

Спочатку формується розуміння зацікавлених осіб та їхніх потреб у звітності та інформації. Зацікавлених особи можуть включати команду проекту, виконавче керівництво, клієнтів, та інші зацікавлені сторони. Аналізуючи їхні очікування та інформаційні потреби, визначається, яка інформація та в якій формі буде найбільш корисною. Далі розробляється стратегія комунікації, визначаючи, як, коли та якою мовою буде відбуватися обмін інформацією. Наприклад, для технічної команди можуть використовуватися технічні засоби спілкування, такі як спеціалізовані платформи чи інструменти для відстеження задач, тоді як для керівництва - регулярні звіти та зустрічі.

Однак важливим є не лише способи комунікації, але і канали розповсюдження інформації. Комунікаційний план визначає, як і де буде розміщена інформація, чи це електронна пошта, внутрішні платформи, зустрічі чи регулярні звіти. Важливо також передбачити можливі ризики та виробити стратегії їх управління. Це може включати в себе проблеми з доступністю ключових учасників, неочікувані зміни в проекті чи інші фактори, які можуть вплинути на потік інформації. Завершальним етапом є впровадження та моніторинг комунікаційного плану. Регулярні огляди та оновлення дозволяють вносити корективи, якщо це необхідно, забезпечуючи адаптацію плану до змін в середовищі проекту.

2.5 Ініціалізація проекту

Ініціалізація проекту є фінальним етапом, на якому визначаються основні аспекти та параметри, необхідні для успішного старту та подальшої реалізації проекту. Цей етап затверджує фундаментальні аспекти, такі як цілі, обсяг, ресурси та зацікавлені сторони.

Початок ініціалізації проекту часто пов'язаний із створенням його концепції та систематизації бізнес-цілей. Це може включати в себе

обговорення із зацікавленими сторонами, визначення проблем, які має вирішити проект, та встановлення ключових показників ефективності. Затверджується визначений обсяг робіт. Навіть у цей момент беклог може вимагати уточнення вимог та визначення технічних аспектів проекту. Після цього затверджується склад команди проекту та визначаються її ролі та відповідальності. Це важливо для забезпечення ефективного керування та спілкування всіх учасників проекту. Зацікавлені сторони також ідентифікуються та взаємодія з ними структурується для забезпечення їхньої підтримки та залучення на різних етапах проекту, а також розуміння рівнів делегування та впливу. Уточнюється бюджет, графік робіт та інших аспектів, що забезпечують ресурсну ефективність проекту. Завершальним кроком визначаються механізми контролю та звітування. Це включає в себе розробку систем моніторингу та визначення етапів звітування перед зацікавленими сторонами.

Узагальнюючи, ініціалізація проекту поєднує у собі попередні результати дослідження за створених планів. Критично необхідно отримати формальне підтвердження на вже визначену інформацію. Це допомагає провести додатковий погляд на підготовку проекту, та провалідувати його.

2.5.1 Визначення усіх зацікавлених сторін

Визначення зацікавлених сторін спрямоване на ідентифікацію та взаємодію з особами чи організаціями, які можуть впливати на або бути впливаними проектом. Цей процес сприяє розумінню очікувань, потреб та інтересів різних сторін, що допомагає забезпечити успішне виконання завдань проекту та полегшення комунікації. Починаючи з ідентифікації всіх осіб та організацій. Це може бути внутрішні та зовнішні сторони, такі як команда проекту, клієнти, фінансові партнери, регулятори, користувачі, або інші групи осіб.

Далі проводиться аналіз цих зацікавлених сторін, де визначаються їхні очікування, потреби, інтереси. Після цього визначається рівень впливу та зацікавленості кожної сторони. Це допомагає визначити, як ефективно взаємодіяти з кожною стороною та на яких аспектах проекту слід зосередити увагу. Важливо також визначити стратегії взаємодії з зацікавленими сторонами. Це може включати в себе розробку комунікаційного плану, проведення регулярних нарад чи презентацій, а також врахування їхніх думок та пропозицій у процесі прийняття рішень. Завершальним кроком є постійний моніторинг та оновлення статусу відносин з зацікавленими сторонами на протязі усього проекту. Це допомагає адаптувати стратегії взаємодії та вирішувати будь-які зміни в їхніх очікуваннях чи потребах.

2.5.2 Кік-офф зустріч

Кік-офф зустріч є фінальним етапом ініціалізації проекту, спрямованим на об'єднання команди та визначення стратегії для успішної реалізації завдань. Підготовка до цієї зустрічі важлива для створення позитивної атмосфери, встановлення спільних цілей та забезпечення розуміння робочого процесу.[12]

Перш за все, команда повинна ретельно вивчити всі матеріали, пов'язані з проектом, щоб мати чітке уявлення про його обсяг, мету та ключові етапи. Це допомагає уникнути непорозумінь та створює загальний фундамент для обговорення на кік-офф зустрічі. Під час підготовки команда починає знайомство з основними учасниками проекту, їхні ролі та відповідальності. Це важливо для забезпечення ефективності обговорень та визначення того, на яких аспектах проекту слід зосередити увагу.

Готуючись до кік-офф зустрічі, менеджер має розробити короткий план порядку денного, щоб забезпечити логічний хід зустрічі та включити всі необхідні пункти. При цьому важливо не лише представити основні аспекти проекту, але й створити можливість для взаємодії та висловлення ідей з боку всіх учасників. Сама зустріч може розпочинатися зі знайомства команди та

представлення їхніх ролей у проєкті. Далі висвітлюються ключові моменти, такі як обрана стратегія, основні етапи виконання, розподіл завдань та визначення очікуваних результатів. Головною метою є не лише подача інформації, а й створення командного духу, сприяння взаєморозумінню та мотивації учасників. Також, важливо створити простір для відкритого обговорення питань, ідей та потенційних труднощів. В результаті проведення кік-оффу, команда проєкту та зацікавлені особи мають отримати загальне розуміння того куди рухається проєкт, які в нього задачі та цілі, стратегії розробки та інше.

2.6 Висновки до розділу

У висновку, виявлення вимог та ініціалізації міграційного проєкту можна відзначити ключові етапи та підходи, що допомагають забезпечити успішний старт та подальше виконання проєкту.

Перший етап - аналіз існуючого продукту - визначає стан та функціонал поточної системи. Бізнес, дизайн та технічні аудити дозволяють здійснити повноцінний огляд, виявити сильні та слабкі сторони, а також визначити обсяг робіт, необхідний для міграції.

Дослідження та позиціонування майбутнього продукту визначають мету та стратегію майбутнього проєкту. Цей етап дозволяє чітко сформулювати цілі, метрики успіху та бізнес-цінність майбутнього продукту.

Визначення обсягу робіт впливає із проведеного аналізу та визначення мети проєкту. Чітко визначений обсяг робіт встановлює основу для подальшого планування та виконання.

Створення міграційного плану проєкту є ключовим етапом, на якому формується загальний план дій. Оцінка робіт, рейтинги пріоритетів, ресурсний

план та управління ризиками сприяють створенню ефективного плану, який враховує всі аспекти міграції.

Ініціалізація проекту, включаючи кік-офф та визначення зацікавлених сторін, встановлює зв'язки та ролі учасників. Це важливо для створення командного духу та забезпечення ефективної взаємодії впродовж усього проекту.

3 УПРАВЛІННЯ ПРОЄКТОМ МІГРАЦІЇ ВЕБ-САЙТУ НА НОВУ ТЕХНОЛОГІЮ

Даний, останній розділ магістраційної роботи буде описувати етапи та інструменти необхідні для того, щоб успішно провести проєкт міграції продукту на нову технологію.

Для пояснення цього проєкту, його замовник надав першочергові проблеми та його думки з приводу можливого рішення. Замовник має веб-сайт міжнародної компанії, котрий розвивався останні десять років, і масштабувався на сторінки котрі надавали інформації про сервіси та послуги які ця компанія надає. Так як ця компанія розпочинала свою діяльність лише з локального ринку, та мали лише декілька послуг, вони не прогнозували що протягом 10 років стануть світовим лідером з сотнею послуг, котрі вони будуть демонструвати на своєму веб-сайті.

Першою проблемною точкою був час постачання нового функціоналу. Розробники цього замовника, як і вся проєктна команда казала про те, що виконання навіть примітивних задач буде займати 2, 4 або навіть 16 годин. Таким чином планування стрімкого розвитку та відображення нових послуг постійно блокувалося на етапі розробки. Відділи продажу та маркетингу втрачали свою ефективність, і компанія втрачала потенційний прибуток.

Другою проблемою було оновлення контенту на сайті. Команда маркетингу постійно працювала над адаптацією контенту, котрий відображався на сайті та створювала відділ, котрий мав генерувати статті для збільшення воронки потенційних клієнтів. На веб-сайті мало публікуватися 3-4 статті на тиждень. Наразі, команда веб-сайту могла публікувати лише одну, під час щотижневого релізу.

Третя проблема визначалася у низькій кількості спеціалістів, та вартості їх роботи для розробки та підтримки цього веб-сайту. Технології, котрі використовувалися на сайті були HTML, CSS та JavaScript. За дослідженням

компанії, більшість спеціалістів хотіли працювати з фреймворком React або Angular.

Останньою та найбільш критичнішою проблемою, була проблема з SEO показниками, котрі мали великий вплив на маркетингову складову веб-сайту. Без вирішення цієї проблеми, компанія втрачала від 40% до 60% потенційної кількості клієнтів.

Таким чином, компанія вже прийняла рішення, що проєкт має піддатися міграції на нову технологію. Але, так як цей проєкт був обличчям компанії, вони не могли собі дозволити жодного простою, або перепон у функціоналі існуючого продукту. В рамках цього розділу ми розглянемо, як правильно підійти до оцінки, ініціації та менеджменту такого проєкту.

3.1 Проведення фази дослідження проєкту

Завдяки першочерговому дослідженню замовника і конкретному визначені проблем, фаза дослідження може здаватися полегшеною, але це не так. Зазвичай, замовник визначає лише ті моменти, котрі впливають на його бізнес-показники і лежать на поверхні. У цьому випадку не слід орієнтуватися на визначені проблеми, але й дослідити можливі аспекти котрі знаходяться «під поверхнею». Це включає вивчення технічного стеку, аналіз бізнес-логіки та вимог до продукту, а також взаємодію з користувачами.

Такий глибокий підхід до дослідження дозволяє зрозуміти не лише експлуатаційні труднощі, а й потенційні проблеми, які можуть виникнути під час міграції на нову технологію. Врахування цих аспектів у процесі дослідження сприяє розробці комплексного та ефективного плану подальших дій.

3.1.1 Формування команди для дослідження проєкту

Для формування команди дослідження менеджер проєкту розпочне довгий шлях проєкту. На даному етапі дуже важливо не лише визначити яка команда буде необхідна, а також і бюджет та терміни за який замовник хоче провести цю фазу. Для найбільш оптимального розподілення ресурсів, є необхідним поділити команду на дві основні складові.

- Команда дослідження вимог проєкту
- Команда оцінки проєкту

Завдяки такому розподіленню, буде набагато легше визначити фінансову складову. Але в першу чергу, саме досвід та об'єктивна оцінка проблем проєкту допоможе менеджеру визначити мінімально необхідну команду для проведення Discovery фази.

Оцінивши першочергові вимоги та проблеми, можемо чітко зазначити що мінімально необхідна команда буде складатися з:

- Проєктний менеджер
- Бізнес-аналітик
- Архітектор

Проєктний менеджер буде виконувати роль оркестратора проєкту дослідження, та буде відповідальний за фінальний план котрий буде наданий спеціалістами.

Бізнес-аналітик визначить обсяг робіт, та розпочне першочерговий аналіз комерційної складової та можливих точок зросту.

Архітектор проведе технічний аудит продукту, та розпише технічний стек проєкту, можливості до масштабування та ефективного розгортання продукту.

Розуміючи, що дана команда зможе надати аналіз проєкту, також можуть бути запропоновані додаткові ролі, котрі пришвидшать процес дослідження, та також принесуть свою кваліфікацію. Такими спеціалістами можуть бути:

- Системний аналітик
- Веб-розробник
- DevOps
- Маректолог

У свою чергу системний аналітик зможе зменшити витрати на дослідження функціональної складової продукту, та забрати на себе частинку обов'язків архітектора. На відміну від архітектора, системний аналітик буде більш ефективним при аналізі технічної складової, та буде надавати іншому вже оброблену для створення рішення інформацію.

Веб-розробника має бути залучено до проєкту для того, щоб він зміг більш детально та якісно оцінити обсяг робіт.

DevOps буде розглядати проєкт з зовсім іншої сторони, а саме зі сторони пришвидшення процесу CI/CD. Так як одна з проблем замовника була у тому, що поставка нового контенту на веб-сайт займала занадто багато часу, він зможе детально описати як цей процес можливо оптимізувати.

Останнім, але не менш необхідним буде маректолог, котрий визначить зони зросту для SEO показників, та скоординує команду на основні аспекти додатку, котрі мають бути реалізовані ще до розпочатку процесу розробки.

Переходячи до оцінки фази дослідження, ми визначаємо два основних фактори:

- Щоденна вартість команди
- Результати/Документи фази дослідження

Чим більше вартість, тим більше артефактів буде готово при завершенні. Для кращого опису даного розділу, ми зробимо припущення, що замовник орієнтується на найкращий результат, адже для нього ця інвестиція буде критичною для майбутнього його бізнесу.

3.1.2 Визначення обсягу робіт

Визначивши об'єкт аналізу, першим на проєкт запрошується бізнес та системний аналітик. Основою їх задачею буде:

- Створити список функціоналу, присутній у поточній версії;
- Проаналізувати актуальність виконання бізнес-задач продукту;
- Провести пріоритизацію задач;
- Оцінка

Під час, коли аналітики працюють над проєктом, менеджер має налаштувати щоденний зворотній зв'язок, для того щоб визначити контрольні точки для того щоб залучити інших спеціалістів для дослідження. Варто зазначити, що існують проєкти на котрі усі спеціалісти заходять із самого початку, але без декомпозиції, бюджет буде сильно підвищений.

Розпочнемо опис процесу визначення обсягу робіт примірюючи на себе роль бізнес-аналітика. В першу чергу, нам необхідно отримати усі матеріали, котрі пов'язані з проєктом. Так як ми досліджуємо продукт у фазі активної розробки ми звернемо увагу на наступні артефакти, за наявності:

- Функціонуючий веб-додаток у виробничому середовищі;
- Функціонуючий веб-додаток у тестовому середовищі;
- Уся наявна документація продукту;
- Вихідний код продукту;

Отримавши доступ до веб-додатку, та його активних середовищ, розпочнемо up-bottom аналіз продукту. Так як ми працюємо з веб-сайтом, нам необхідно

побудувати ієрархію доступних сторінок, визначити функціональні елементи такі як блоки або секції, та продовжити дослідження декомпозуючи складові цих блоків або секцій. Зазвичай, SEO оптимізовані веб-сайти мають мати у корені пакету додатку файл з назвою sitemap.xml. Цей файл використовується пошуковими системами для пришвидшення індексації сторінок, і у випадку аналізу дозволить визначити наявні сторінки веб-сайту. Додатково треба мати на увазі наявність сторінок-привидів. Такі сторінки можуть знаходитись за будь-яким URL, і можуть бути знайдені вручну лише завдяки статичному аналізу кодової бази.[13] Пошук цих сторінок буде виконувати системний аналітик.

При наявності sitemap, робота аналітика буде сильно полегшена. У разі його відсутності буде необхідно вручну визначити сторінки на котрі може перейти потенційний користувач. Наступним кроком для БА буде подубування абстрактної моделі веб-сайту завдяки UML (Unified Modelling Language) або використанню ментальної карти (Mind map). Гарною практикою також буде групування цих сторінок за напрямками. Найбільш органічним варіантом буде використання рівнів URL. Наприклад, маємо веб-сторінку з адресою <https://site.com/A/B>. Де site.com – це корінь, А – перший рівень, котрий буде містити у собі сторінку другого рівня В.

UML (Unified Modelling Language) – це UML (Unified Modeling Language) — це стандартизована мова моделювання, яка використовується для візуального представлення системних або програмних концепцій. UML є результатом спільної роботи багатьох фахівців у галузі об'єктно-орієнтованого програмування та моделювання.[14]

Наступним кроком буде визначення функціональних блоків або секцій. Так як вже було створено ментальну карту веб-сайту, наступні кроки будуть повністю ручні. Кожна сторінка має бути поділена на функціональні блоки, котрий з яких має бути незалежним один від одного, та мати бізнес цінність. Проводячи такий аналіз ми скористаємось ще одним інструментом, а саме –

матрицею простежуваності. Кожен рядок цієї матриці буде визначати унікальну сторінку, визначену у ментальній карті, кожен стовпчик буде визначати унікальний функціональний блок, та його ідентифікатор. Кожен перетин цих елементів буде визначатися умовною позначкою. Додатково ми додамо ще один вимір – цифру, котра буде визначати порядковий номер відображення цієї секції на певній сторінці. Зверніть увагу, що у великих продуктах, певні сторінки можуть мати повністю унікальний функціонал, і не буде пере-використаний для інших сторінок. Бізнес-аналітику треба буде їх позначити як унікальну. Та визначити їх реальну необхідність та бізнес-цінність котру вони приносять. Це допоможе позбутися зайвого функціоналу, та оцінити який з інших існуючих блоків може його замінити.

Продовжуючи процес декомпозиції, ми маємо пам'ятати і про мінімально функціональні складові тих блоків, котрі ми визначили на попередній стадії. Звичайно, може бути перевикористаний метод визначення як той, коли ми визначали блоки. Проте, ще один рівень вкладеності буде дуже складним для розуміння та підтримки документації. Особливо на етапі розробки. Отже, ми приходимо у невеликий глухий кут. Яким чином можуть бути визначені мінімальні компоненти цього великого конструктору, щоб описати їх обмеження та рекомендації до застосування? Найкращою практикою для цього буде створення UI-Kit. І для цього нам потрібно буде залучити дизайнера. У більшості випадків, UI-Kit є певною директивою для підходу до дизайну і містить у собі такі елементи як кольори, кнопки, картки, перемикачі і т.д.[15] У випадку цього проєкту, у замовника вже був створений UI-kit, та найнятий дизайнер у штат. Завдяки цьому, бізнес-аналітику достатньо визначити ці мінімально-функціональні елементи у єдиному списку, та передати дизайнеру на валідацію того, що дані елементи вже представлені у інтерфейсі користувача.

У випадку, якщо замовник не має дизайнера у штаті, проєктний менеджер має терміново залучити такого співробітника до фази дослідження. Насправді, дуже складно явити у сучасному світі проєкти, які працюють без UI/UX дизайнера. Це саме той спеціаліст котрий може перевести вимоги замовника у графічний формат та потенційно зменшити витрати на розробку у два рази. Один із найбільш збиткових факторів розробки програмного забезпечення є невідповідність першочерговим вимогам. Процес дизайну дозволяє зменшити, або і зовсім уникнути кількість таких ситуацій. Дизайнер спочатку створить каркас функціоналу, після підтвердження зможе розробити макет, та навіть перетворити його у функціональний прототип. Це дозволить зацікавленим особам подивитись на практиці як буде виглядати замовлений функціонал, та підтвердити свої гіпотези щодо його фінального відображення у продукті.

Визначивши ці змінні, бізнес-аналітик може приступати до наповнення першочергової версії беклогу продукту. Беручи до увагу проаналізовану вище інформацію, найкращою практикою буде скласти об'єм робіт, котрий буде складатися з двох основних категорій або груп. Перша група буде визначати задачі, котрі опишуть створення функціональних сторінок. Друга група опише функціональні блоки, котрі будуть частинами вище зазначених сторінок.

Паралельно із роботою бізнес аналітика, до нашого проєкту також був залучений системний аналітик. Хоч на перший погляд може здаватися, що вони будуть заважати один одному, або виконувати ту саму роботу, слід зазначити що це має відбуватися під контролем проєктного менеджера. Завдяки роботі у команді, кожен з цих спеціалістів буде ділитися своїми знаходження у щоденному форматі, та визначати як вони можуть допомогти один одному. Вище було зазначено, що бізнес-аналітик дивиться на продукт з точки зору зацікавлених осіб та кінцевого користувача, і на мою думку такий тип аналізу не завжди є об'єктивним, тому що об'єкт аналізу оточений навколишнім середовищем котре має тенденцію до змін, та дуже часто

визначається суб'єктивною думкою різних осіб. У свою чергу, системний аналітик буде непристрасний до об'єкту свого аналізу, і зможе чітко визначити, що з цього добре, а що – погано.

Головним джерелом інформації для аналізу буде вихідний код продукту. Ми можемо порівнювати аналіз одного й того самого продукту між системним та бізнес аналітиком як аналог тестування білою та чорною скринею. Тобто, системний аналітик розуміє які вимоги стоять перед написаним кодом, та можуть оцінити наскільки якісно та ефективно він виконується. Він буде роздивлятися такі аспекти як:

- Взаємодія компонентів коду;
- Визначення компонентів привидів, коду котрий не використовується;
- Структура зв'язку Front-end та Back-end частини;
- Data flow;
- Задіяння та використання баз даних.

Результатами такого дослідження буде отримано низку діаграм та статей, котрі будуть описувати функціональні метрики та підсвічувати проблемні місця котрі впливають на фінальний продукт. Це можна вважати «їжею для мозку», для котрої треба буде придумати технічне рішення. Зазвичай це технічне рішення буде лежати на плечах архітектора та розробника.

Як тільки системний аналітик завершує роботу над своєю частиною аналізу, він починає доповнювати наявні групи беклогу, або ж додавати нові, котрі бізнес аналітик не зміг визначити. Такого обсягу має буде достатньо для того щоб почати процес пріоритизації та визначення необхідних зон для дослідження іншими спеціалістами. На початку цієї фази ми отримали від замовника список проблемних місць, на котрі ми маємо вже надати відповідь або доповнити його своїми. Визначивши у такому списку додаткові місця,

додає нам той самий необхідний рівень абстракції, який буде адресовано співробітникам для проведення технічного аудиту, та технічного рішення.

3.1.3 Проведення технічного аудиту

Технічний аудит веб-сайту – це найкритичніший етап у визначенні коштів та якості котрі будуть додані до проєкту. Результати цього аудиту стануть базисом до архітектурного рішення, до використання мов програмування, та до забезпечення його ефективності, безпеки та відповідності стандартам.[16] Цей процес передбачає ретельний аналіз технічних аспектів сайту з метою виявлення потенційних проблем, оптимізації функціональності та надійності у майбутньому. Перед залученням технічних спеціалістів ми вже визначали поверхневий обсяг робіт та проблеми, котрі потребують технічного рішення. Завдяки цьому, ми можемо завести усіх спеціалістів одночасно та контролювати час необхідний для виконання.

Спочатку, технічний аудит включає вивчення архітектури веб-сайту. За цей процес повністю відповідає архітектор. Це включає в себе огляд структури сторінок, розташування елементів та взаємодії між ними. Окрім веб інтерфейсу, архітектор також буде роздивлятися взаємодію між клієнтською та серверною частиною, і додатково те як цей продукт має розгортатися на загальну аудиторію. Йому будуть підпорядковані розробник та DevOps інженер, котрі будуть поглиблюватись у частини архітектури визначені архітектором.

Проблеми, які має вирішити архітектурне рішення проєкту:

- Зменшити час from merge to prod. Час, котрий займає додаток від переливання нового коду до відображення його в оточенні проєкту має бути не більшим за 5хв.

- Максимально скоротити час відображення динамічного контенту (статей). Час для того, щоб відобразити текстові зміни у статті мають бути миттєвими, займати не довше 10 сек після зміни.
- Повністю прибрати дублювання коду.
- Покращити усі показники життєво важливих для веб-додатків (web vitals), щоб вони були у зеленій зоні.
- Покращити можливість до масштабування проєкту.
- Зменшити витрати на розробку та обслуговування.

Оцінивши усі вхідні дані, Архітектор буде малювати схему проєкту за стандартом С4. Це допоможе поділити систему на декілька рівнів і визначити який із рівнів буде закріплено за певним спеціалістом. Першим рівнем визначається поверхневий контекст системи.

На другому рівні буде описано контейнерну діаграму

На наступному рівні будуть описані складові кожної з додатків, та функціональні особистості котрі вони виконують. Це можна вважати більш детальним рівнем, котрий Архітектор надасть кваліфікованому розробнику (-кам). Тобто, на цьому рівні вже визначаються технології котрі мають використовуватися для проєкту, але варто зазначити що це рішення не буде фінальним, і поринувши нижче у продукт розробник може використати свій досвід щоб довести протилежне. Нажаль, для повного опису такого проєкту необхідно мати великий досвід у декомпозиції таких систем, тому у рамках цієї роботи ми не будемо занурюватись нижче контейнерного рівня.

Як можна зазначити вище, новий стек технологій на проєкті в першу чергу є більш гнучким. Та дозволяє розробляти динамічний веб-додаток у короткі терміни. Але для відповіді або підготовки рішення до проблем зазначених вище нам необхідно подивитись на продукт не лише з архітектурної, але й з інфраструктурної сторони. Тобто, з тієї сторони котра

буде відповідати за розгортання нашого продукту, та відображення його у мережі Інтернет для потенційних користувачів та клієнтів цього веб-сайту.

Для цього ми залучимо DevOps інженера, котрий зможе використати технології та послуги такої платформи як AWS для найбільш ефективного розгортання такого додатку. Наприклад, так як наш додаток буде динамічним і змінюватись після зміни контенту у CMS або Базі Даних, буде використовуватися CloudFront для його розгортання. Для збереження інформації може використовуватися база даних S3. А для контейнеризації ECS (Elastic Container Service) або EKS (Elastic Kubernetes Service).[17] Насправді, кількість комбінацій для досягнення необхідних результатів буде досить великою. У контексті даної магістерської роботи, такий рівень деталізації розглядати сенсу нема.

На мою думку, на цьому етапі також треба звернути увагу на те, як ми будемо розгортати два додатки пліч о пліч. Ми пам'ятаємо, що замовник хоче забезпечити стабільну роботу веб-сайту і частково підставляти готовий функціонал у новому додатку.

Важливо враховувати, що найкраще рішення для проєкту буде розроблено лише завдяки командній роботі. І ці етапи мають бути затверджені результатом аудиту наявного продукту. Для цього розробляється документ, котрий буде мати у наявності список проблем зазначених вище, причини їх виникнення та метрики котрі це демонструють, та також рішення котре позитивно вплине на їх покращення.

На основі рішення котре буде зазначено у документі результатів аудиту, технічна команда також доповнює обсяг робіт необхідними технічними завданнями, котрі будуть впливати на бюджет та терміни проєкту міграції. Зазвичай такі технічні задачі виділяються в окрему категорію задач, і мають бути реалізовані ще до того як почнеться фактична розробка функціоналу веб-додатку. У нашому випадку, такими задачами будуть ті котрі будуть впливати

на переробку інфраструктурної складової та підготовку контейнерів для подальшого початку розробки.

3.2 Створення плану реалізації проекту

Після завершення етапу першочергового дослідження, усі результати робіт надаються проєктному менеджеру. На мою думку, планування будь-якого міграційного проєкту є найкритичнішим кроком для його успішної реалізації. Загалом ми розглянемо три основні документи, котрі будуть впливати на процес планування проєкту.

- Документ оцінки функціональних складових
- Ресурсний план
- Бюджетний план
- Дорожня карта проєкту

Необхідно зазначити, що на даній фазі проєкту ми визначимо лише першу версію цих документів, котрі будуть орієнтиром для операційної діяльності проєкту. Правильна підготовка цих документів дозволить легко вносити зміни до кожного з них та бачити вплив зміни параметрів на трикутник проєктного менеджменту. Зрозуміло, що навіть у детально підготовлених проєктах необхідно балансувати між якістю, бюджетом, часом, та очікуваннями замовника. Це є поганою практикою надати замовнику, або вашому менеджменту сухий бюджет без будь-якої можливості до модифікації. Ресурси, котрі виділяються на проєкт, не є безмежними і це треба брати до уваги. В певній мірі, менеджер демонструє свою креативність і має надавати більш ніж один варіант для вирішення та виконання задач проєкту. За моєю практикою, найкращою альтернативою буде створення декількох фаз проєкту.

Розпочинаючи створення плану реалізації проєкту ми розпочнемо створення документу оцінки функціональних складових. У попередніх

розділах ми вже висвітлювали такі поняття як беклог та обсяг робіт. Він постійно доповнювався різними членами команди та був здебільшого хаотичним списком задач, проблем та рішень котрі мають формувати план проєкту. У поєднанні з бізнес-аналітиком, проведемо розбиття на великі складові під назвою Еріс. Ці складові будуть визначати великі функціональні частини нашого продукту, і будуть містити у собі користувацькі історії (User Story) або задачі, котрі будуть приносити пряму цінність проєкту. Ми вже визначали поверхневу структуру на рис.3.5. Нашим наступним кроком буде розширення цієї таблиці та підготовка її для оцінювання командою проєкту.

Загальний шаблон буде виглядати подібно до того, що відображений на табл.3.2. Цей документ буде використовуватися різними ролями на проєкті для проведення оцінки робіт. Ми можемо звернути увагу, що у даному списку будуть наявні задачі як ті, котрі використовуються лише певними спеціалістами, так і ті котрі будуть потребувати залучення декількох з них. Для цього найкращим способом буде відокремити кожен з функцій проєкту та визначати оцінку окремо один від одного. На даному шаблоні ми бачимо такі колонки як «Оптимістична», «Реалістична», «Песимістична» та «Фінал». Для експерта буде необхідно заповнити лише поля у стовпчиках «Оптимістична» та «Песимістична». Такий процес оцінки є досить продуктивним, адже зазвичай існує ризик переоцінити або недооцінити задачі. Щоб оцінка була оптимізована, ми будемо рахувати поле «Реалістична», котра визначається середнім між попередніми двома. І нарешті «Фінал», буде додавати коефіцієнти необхідних додаткових активностей котрі будуть присутні при роботі над кожної з цих задач.

Після проведення оцінки кожним з експертів на проєкті ми зведемо їх у єдину матрицю та підрахуємо загальні витрати часу необхідні для проєкту.

3.2.1 Оцінка плану робіт

Розпочнемо нашу оцінку зі сторони бізнес-аналізу. Після того як проєкт розпочнеться нам буде необхідно зрозуміти кількість часу для опису вимог кожної з користувацьких історій.

Як ми бачимо на таблиці, оцінка потребує лише функціональних компонентів котрі принесуть пряму цінність бізнесу. Із додаткових активностей у нас присутня лише похибка

Продовжуючи процес оцінки, ми залучимо розробників.

Як ми бачимо, процес оцінки зі сторони розробки буде подібний до оцінки бізнес-аналітиком, проте ми маємо набагато більше додаткових активностей котрі впливають фінальний підрахунок.

Роздивившись ці дві таблиці, у нас залишається EPIC-1 котрий буде повністю підпорядкований DevOps інженеру. Проведемо оцінку і з його сторони.

В результаті ми отримуємо аналог WBS, котру треба перевести у загальну оцінку проєкту. Для цього ми використаємо матрицю агрегації

Дана матриця допомагає визначити загальні зусилля на проєкт та підготувати ресурсний план.

3.2.2 Створення ресурсного плану

Найскладнішою складовою планування є визначення ресурсного плану. Це перший конверсійний момент, коли умовні години та зусилля перетворюються у гроші. Цей делікатний момент треба розглядати з точки зору максимальної оптимізації, але при тому закладання можливих ризиків. Отже, наразі ми маємо години та для конвертації нам необхідні зарплатні, а саме вартість години роботи спеціаліста. Але, давайте розпочнемо з календарного плану проєкту. Також, ми прорахуємо вартість на підтримку

такої команди помножуючи їх на рейти спеціалістів. Останнім кроком ми додамо додаткових спеціалістів, котрі потребуються для підтримки якості постачання продукту.

Визначивши усі ці зміни ми отримуємо досить компактну таблицю, котра поєднує бюджет та ресурсний план. У цьому випадку проєкт дослідження є досить малим, але його достатньо для отримання розуміння концепції підрахунку проєкту будь-якого обсягу.

3.3 Визначення зацікавлених сторін та їх впливу на проєкт

Кожен проєкт залучає значну кількість зацікавлених осіб, які впливають на нього або отримують від нього користь. Управління цими особами та аналіз їхнього впливу є ключовим для визначення оточення будь-якого проєкту. Stakeholder-аналіз є ефективним інструментом, оскільки дозволяє визначити та зрозуміти вплив, який кожен зацікавлений має на проєкт.

Результати аналізу призводять до розкриття різних точок зору тих осіб, які мають найбільший вплив. Використовуючи їхню експертність та досвід, можна значно покращити якість проєкту. Крім того, цей процес може допомогти проєктному менеджеру залучити додаткові ресурси у вигляді нових фахівців, додаткового часу та фінансових коштів.

Ефективна комунікація з зацікавленими сторонами дозволяє досягти максимальної прозорості та забезпечити зворотній зв'язок, щоб кожна сторона розуміла, яка користь проєкт приносить саме їй. Щільна комунікація дозволяє менеджеру ефективно впливати на зацікавлених осіб та передбачати їхню реакцію на результати проєкту. Цей взаємний обмін інформацією сприяє успішному управлінню впливом зацікавлених осіб та сприяє досягненню успішних результатів проєкту.

3.3.1 Ідентифікація зацікавлених осіб

Цей процес схожий на етап початкового збору вимог для запуску проекту, однак у даному випадку він повністю покладається на плечі команди проекту та залишається активним протягом усього життєвого циклу проекту. Перший крок - визначення методу роботи, і, на моєму досвіді, найефективніше використовувати техніку брейншторму.

Під час брейншторму, проектний менеджер виступає у ролі фасилітатора, підготовлюючи всі необхідні матеріали для зустрічі. Для традиційних зборів можна використовувати дошку або стікери, на які кожен учасник команди записує свої думки.[18] У випадку визначення стейкхолдерів, кожен учасник вносить тих осіб, з якими він взаємодіє під час роботи над проектом або під час фази дослідження, або тих, на кого може впливати проект. Після додавання всіх записів команда аналізує їх і видаляє ті, які не мають сенсу, обговорюючи правдивість цього факту. Важливо враховувати, що занадто велика кількість учасників може призвести до негативних наслідків та бурхливого розширення фінальних записів.

В результаті цього процесу проектний менеджер обробляє всі виводи зустрічі та оформлює їх у звіт. Цей звіт може бути представлений у формі графіку, таблиці або простого списку зацікавлених осіб. Для кращої наочності рекомендується вказувати не лише посаду чи роль особи, але і її ім'я та прізвище. Я особисто обираю використання схеми (рис. 3.6), щоб зробити інформацію більш зрозумілою та доступною.

3.3.2 Розставлення пріоритетів

Після отримання завершеної схеми зацікавлених осіб, наступним етапом для проектного менеджера є розподілення пріоритетів між ними. Це критично важливо для того, щоб забезпечити відповідний підхід до кожного з них та уникнути розпилення уваги на комунікацію котра не приносить ваги. Якщо

менеджер витратить час на вирішення питань тих, хто не має суттєвого впливу на проект, це може викликати невдоволення від інших осіб та впливати на прогрес проекту.

Для кращого управління зацікавленими сторонами, їх можна розділити на дві категорії відповідно до рівня їхнього впливу на проект і зацікавленості у його результатах. Для цього використовується матриця з двома осями: впливу та зацікавленості (рис. 3.7).

Зазначимо, що осіб із значним впливом та великою зацікавленістю слід приділяти більше уваги, оскільки їхні відгуки та погляди можуть суттєво впливати на успішність проекту. З іншого боку, стейкхолдери з меншим впливом або зацікавленістю можуть вимагати менше уваги, і їхні питання можна вирішити на пізніших етапах проекту.

Ця матриця подібна до матриці Ейзенхауера і віддзеркалює її принципи, проте використовує інші осі. Чотири квадранти, які ми отримали, можна характеризувати наступним чином:

1. Тісна взаємодія: Ці особи вимагають тісної взаємодії та повної уваги до своїх потреб. Близька співпраця з ними є ключовою, і вони можуть допомогти в масштабуванні проекту.

2. Тримати задоволеними: Для цієї групи важливо тримати їх задоволеними, але не обтяжувати постійною взаємодією. Ефективна комунікація з ними може забезпечити високу підтримку.

3. Тримати інформованими: Ці особи повинні бути інформованими про результати роботи та мати можливість висловлювати свою думку. Їхні бажання стосовно результатів проекту слід враховувати.

4. Моніторинг: Ця група знаходиться далеко від ядра проекту, проте проектному менеджеру необхідно слідкувати за ними, оскільки вони можуть змінити своє ставлення до проекту та перейти до інших груп. Забезпечення

певного рівня моніторингу є важливим для попередження можливих змін у їхньому ставленні до проекту.

3.3.3 Проведення класифікації

Настала пора доповнення підготовленої матриці, ролями визначеними у схемі. Цей процес включає в себе суб'єктивну та об'єктивну оцінку. Загалом, для досягнення ефективних результатів ми ставимо низку конкретних питань та шукаємо на них відповіді. Ефективніше всього використовувати закриті запитання, на які можна отримати відповіді, які легко ввести до матриці. Ці запитання поділяються на дві групи відповідно до вісей матриці, наприклад:

- Впливає на бюджет?
- Управляє обсягом робіт?
- Спілкується з командою?
- Має цінність від діяльності проекту?

Кількість запитань може бути значною, і на західних проектах практикується ставлення таких питань безпосередньо зацікавленим особам, використовуючи інтерв'ю чи опитувальні листки. Отримавши відповіді у вигляді булевих значень "так" або "ні", менеджер може проаналізувати і класифікувати зацікавлених осіб за допомогою матриці (рис. 3.8). Такий підхід допомагає забезпечити об'єктивність та систематизацію при оцінці впливу та інтересів учасників проекту та побудувати класифіковану схему (рис. 3.9).

3.4 Ризик-менеджмент проекту

Будь-який фактор, що впливає на проект, призводить до наслідків, які можуть або не повинні виникати. Ці наслідки вважаються ризиками. Вони можуть виникнути на будь-якому етапі роботи над проектом та мати різні

форми впливу. У цьому контексті розглядаються ризики, які виникають під час етапу дослідження та активної розробки.

Оцінка ризиків є дуже важливим та чутливим процесом, який вимагає постійного моніторингу, оскільки стан ризику та його вплив на проект може змінюватися щодня. Відповідальність за цю задачу несе як проектний менеджер, так і вся команда проекту. Незалежно від того, чи маємо статичне чи динамічне середовище, робота з ризиками базується на створенні реєстру ризиків та включає ідентифікацію, аналіз, стратегію та контроль ризиків. Ці етапи сприяють систематизації та управлінню ризиками протягом усього життєвого циклу проекту, допомагаючи забезпечити ефективний контроль над можливими негативними впливами.

3.4.1 Ідентифікація ризиків

Процес ідентифікації ризиків є ітеративним і розпочинається на етапі початкового дослідження проекту. Протягом життєвого циклу проекту ризики можуть з'являтися, закриватися або активуватися. Питання, яке ставиться до ризиків, зазвичай починається з "чи?". Ми не можемо бути впевнені, коли мова йде про ідентифікацію ризиків. В тому випадку, коли ми щось стверджуємо в рамках цього процесу, ризик визначається як той, який вже спрацював, і ми можемо лише спостерігати за його впливом на проект. Немає необхідності вважати ризиком лише негативні чинники, оскільки ризиком можуть бути і події, які мають позитивний вплив та на які ми не можемо впливати.

Першим етапом у ідентифікації ризиків проектний менеджер опирається на їхні зони впливу[19]:

- Технічні ризики;
- Зовнішні ризики;
- Організаційні ризики;

- Ризики управління.

Враховуючи ці групи, менеджер працює з кожною з них окремо, може залучати аналітиків або експертів для консультацій. У випадку відсутності такої можливості, можна використовувати техніки якісного та кількісного аналізу, наприклад, проведення інтерв'ю з зацікавленими особами або брейншторму. Якщо проект вже у активній фазі, можна використовувати метрики та їхні виведення як кількісні дані для аналізу. Цей підхід є найбільш продуктивним, але вимагає додаткового налаштування для старту і стає неможливим до етапу активної розробки.

3.4.2 Аналіз ризиків

У динамічному середовищі, аналіз ризиків вимагає ретельності як для зовнішнього, так і для внутрішнього середовища. З метою оптимізації часу і витрат на подвійний аналіз для кожного з середовищ можна використати SWOT-аналіз. Суть його полягає в тому, щоб одночасно розглядати проект як об'єкт аналізу з внутрішнього середовища (SW) та зовнішнього (OT).[20]

Як результат, ми миттєво можемо класифікувати ризики як ті, що можуть призвести до користі для проекту, так і ті, які можуть йому завадити. Крім того, важливо звернути увагу на ймовірність виникнення таких ризиків та оцінити ступінь їх впливу на проект. Інформація про вплив та ймовірність буде додана до реєстру ризиків.

3.4.3 Стратегія роботи з ризиками

В контексті управління ризиками можна виділити чотири основні стратегії:

1. **Прийняття ризику:** Ця стратегія використовується у випадках, коли ризик вважається прийнятним, і вартість або час, потрібний для його усунення,

перевищує його можливі негативні наслідки. Прийняття ризику дозволяє економити ресурси та уникати непотрібних витрат.

2. Передача ризику: Ця стратегія передбачає делегування відповідальності за управління ризиком сторонній стороні, такій як страхова компанія або підрядник. Це може допомогти зменшити фінансові втрати та відповідальність.

3. Уникання ризику: Ця стратегія передбачає вжиття заходів для усунення або зменшення ймовірності виникнення ризику. Це може включати в себе зміну стратегії проекту або введення додаткових контрольних заходів.

4. Зниження впливу ризику: Ця стратегія спрямована на зменшення наслідків ризику в разі його реалізації. Це може включати попередні заходи, щоб зменшити вплив або підготовку планів невідкладних заходів для швидкого реагування.

Обираючи конкретну стратегію для кожного ризику, необхідно враховувати індивідуальні характеристики та умови проекту. Результати цього вибору будуть внесені до реєстру ризиків, що допоможе ефективно управляти їх впливом на проект.[21]

3.4.4 Контроль та моніторинг ризиків

Після завершення попередніх етапів, проектний менеджер формує реєстр ризиків, який стає ключовим інструментом управління проектом. Забезпечення актуальності та розширення реєстру вимагає систематичних переглядів, і найкраще проводити повторні оцінки ризиків на завершення кожної ітерації проекту.

Це забезпечить постійну актуальність реєстру, дозволяючи кожному члену команди відстежувати статус конкретних ризиків. Команді проекту стає можливим вчасно реагувати на ризики та використовувати встановлені плани

для їх вирішення. Результатом заповнення реєстру ризиків стає докладний план дій для кожного ризику.

Заповнений реєстр ризиків (див. Таблицю 1) стає невід'ємним елементом операційного керівництва проектним менеджером. Важливо зауважити, що цей інструмент також вимагає взаємодії та зворотного зв'язку від команди проекту.[22]

3.5 Операційна діяльність та гнучке планування проєкту

Підхід до гнучкого планування зазвичай відрізняється від школи менеджменту. Одні мають думку, що першочерговий план має складатися першим, після чого він допоможе оцінити вплив зацікавлених сторін та можливі ризики. Але на мою думку, такий підхід є передвзятим та зазвичай не дає оцінити усі зони перед розпочатком планування. Тому, у рамках цієї роботи ми будемо користуватися вихідними даними з обсягу робіт, ресурсного плану, бюджету та зацікавлених сторін і їх інтересів для того щоб побудувати максимально ефективний календарно-мережевий графік проєкту.

Як ми визначили раніше, не усі типи активностей на проєкті будуть потребувати залучення усіх спеціалістів. Тобто ми можемо побудувати план таким чином, щоб зменшити можливий вплив залежностей між активностями та результатами цих активностей. Зазвичай я притримуюсь думки, що на проєктах котрі мають чітке розподілення ролей планування активностей на одному рівні не є ефективним. Наприклад, команда бізнес-аналізу буде працювати за своїм планом та мати власні дедлайни, котрі мають забезпечувати готовність роботи для команди розробки. Таким чином ми можемо отримати правильний гібрид застосування різних методологій управління проєктами. БА команда будуть використовувати Канбан підхід та фокусуватися на пріоритетах для поставки своїх задач. Таким чином вони сформуують скрам-беклог робіт, котрі будуть готові для переоцінки командою

розробки та планування їх у спринти. Ми також будемо бачити лінійну залежність обсягу готових вимог, та можемо поєднати її з місткістю команди розробки.

Отже, розпочнемо формулювання дорожньої карти проєкту. Для цього найкраще використовувати таке програмне забезпечення як MS Project, або більш сучасні Jira Advanced Roadmaps, або надбудову Structure by Tempo. У рамках цієї роботи я буду вимушений спросити дорожню карту та розробити її за допомогою інструментів Google Sheets.

Нажаль, даний інструмент не дає нам адаптивності, і потребує ручного аналізу та відтворення подій. Цього може бути достатньо на фазі першочергового планування, але під час дії проєкту, менеджеру необхідно зробити запит на закупівлю програмного забезпечення для управління проєктом. Але, навіть у ситуації як в мене, коли наявного інструменту нема під рукою, але план треба скласти, необхідно проявити креативність.

Трошки проаналізувавши цей план, ми можемо поділити його на декілька певних фаз. Перша фаза – підготовка аналізо-технічних даних для розробки проєкту. Бізнес-аналітик буде описувати першочергові вимоги, і в цей час ми утилізуємо години розробки для того щоб зробити першочергове налаштування проєкту. Як ми бачимо, а даному випадку у нас навіть є два тижні запасу на роботу аналітика, адже раніше розробник не приступить до виконання цих задач.

У свою чергу DevOps інженер буде працювати з своїм окремим планом, та також встигне виконати свої задачі, щоб не блокувати розробників. Ми бачимо, що розробка функціоналу починається коли продовжується конфігурація CD пайплайну, але це не буде необхідним, адже розробнику буде достатньо наповнювати репозиторій. Але цей ризик треба прийняти до уваги.

В операційні діяльності також слід зазначити, що першою визначною віхою буде надходження першої сторінки. Як тільки це відбудеться, проєктний

менеджер може розпочати планування її розгортання у паралель з активним продуктом. Реалізація цього процесу розпочнеться вже в Березні і одразу надасть можливість отримувати якісні показники для оцінки ефективності нової системи у порівнянні зі старою. В результаті, проєктний менеджер має розпочати підготовку до наступних фаз проєкту, та визначати які ще модулі можуть бути додані та підготовлені.

3.6 Повторний аналіз проєкту, звітування

Протягом життя проєкту нам потрібно буде проводити постійне звітування, та контроль основних показників проєкту. Такі показники ми можемо поділити на проєктні та продуктові.

Перші будуть стосуватися життєвого циклу проєкту та ефективності його постачання. Тут ми можемо розглядати такі звіти метрик як:

- SWOT аналіз
- Здоров'я беклогу
- Розподілення витраченого часу командою розробки
- SCRUM метрики
- Та інші.

Такий звіт має відповідати на питання, чи йдемо ми за графіком, чи існують у нас проблеми? Чим швидше ми будемо їх ідентифікувати, або бачити будь-які девіації, тим швидше ми зможемо застосувати дії для зменшення їх впливу.

Продуктові показники у свою чергу є важливішими для проєкту міграції. Головною задачею менеджера у даному звіті буде якнайшвидше показати що ті проблеми котрі були визначені на етапі дослідження мають позитивну динаміку. Наприклад, ми можемо постійно оцінювати час необхідний для того щоб код з'явився на активному середовищі. Або, кількість позитивний

конверсій із нових сторінок. Головне на цьому етапі визначити разом із замовником, які саме метрики він хоче бачити, та які з них дійсно приносять цінність його продукту. Нікому не потрібні результати звітів, котрі не надають цієї ваги для оцінювання результативності проєкту.

Завершуючи такий проєкт, ми зможемо зібрати загальні результати наших звітів та продемонструвати їх замовнику у фінальному репорті. Без сумніву, чим раніше ми розпочнемо збір усіх цих метрик тим краще ми зможемо показати у динаміці як саме зміни впливали на проєкт та продукт. Можливо якісь зміни принесли свою цінність одразу, а можливо необхідно було провести додаткові роботи для того щоб їх адаптувати.

Для подальших можливостей проєкту край необхідно провести роботу над помилками. Це допоможе побудувати міцний контакт із замовником, та знайти точки впливу для продовження проєкту та розширення можливостей продукту.

3.7 Висновки до розділу

Даний розділ був націлений на визначення елементів та факторів успішним управлінням проєктом міграції веб-сайту на нову технології. Було розглянуто основні фази, інструменти та підходи його менеджменти для досягнення очікуваного результату.

Визначення фази дослідження проєкту є найкритичнішою складовою планування проєкту. На даному етапі проводиться детальний розбір існуючого додатку, проведення аудитів різного типу та складності, котрі визначають зони для покращення продукту. Сформована команда такої фази може забезпечити замовника або власника проєкту повним обсягом інформації, котрий допоможе направити розвиток цього проєкту у правильному напрямку, побудувати стратегію його розвитку та реалізації. Ця фаза є найбільш

навантаженою та вимагає високого рівня знань та досвіду від усіх членів команди, котрі будуть вивчати об'єкт цього дослідження.

На основі визначеного обсягу робіт, проєктний менеджер зможе детально прорахувати критичні для проєкту фактори завдяки таким документам як:

- Документ оцінки функціональних складових
- Ресурсний план
- Бюджетний план
- Дорожня карта проєкту

Завдяки підготовці цих документів буде можливо провести повноцінну пріоритизацію робіт та розкласти їх на календар. Така підготовка буде критично впливати на успіх проєкту.

Було розглянуто у якому середовищі буде існувати проєкт під час та після його ініціалізації. Які зацікавлені особи існують, як їх визначити, та як задовольняти їх потреби. Був визначений вплив ризиків на проєкт, та описана потреба у їх ідентифікації та аналізу. Завдяки стратегіям роботи з ризиками, було затверджено як їх можливо контролювати та моніторити.

Операційна діяльність проєкту поєднує у собі вище зазначені фактори для того щоб оптимально та гнучко займатися процесами управління та контролю. У висновку, саме постійний та ітераційний підхід до операційної діяльності допомагає на щоденній або щотижневій основі проводити повторні аналізи стану проєкту та продукту. Оцінка отриманих метрик допомагає визначати напрям проєкту, досягнення його мети та адаптивно змінювати складові проєкту для покращення його ефективності.

ВИСНОВКИ

Розвиток технологій за останні 20 років відкрив низку можливостей щодо оптимізації та покращення ефективності будь-яких продуктів. Наразі ці технології використовуються на більшості проєктів котрі створюються у сьогоденні, але старі продукти залишаються на старому технічному стеку. Потреба у міграції таких Legacy проєктів на нові технології все зростає по експоненті. Чим довше власники продуктів будуть відкладати міграцію, тим меншими для них будуть шанси конкурування на ринку.

У ході дослідження було проаналізовано методи та підходи до міграції проєктів, а також визначені їх сильні та слабкі сторони з точки зору бюджету, часу виконання та гнучкості. Частковий метод міграції виграє майже за усіма показниками, та дозволяє змінювати середовище проєкту в залежності від динамічних змін зовнішнього середовища. Було інструменти та терміни котрі використовуватись у рамках міграції проєкту, а також вплив на проєкт при їх наявності. Також, в рамках дослідження були визначені зони впливу міграції на продукт. У певних випадках вони покращують технічні та інтерактивні складові, хоча також можуть негативно впливати на адаптацію до вже існуючих складових та їх вплив на функціональність, продуктивність, масштабованість.

В рамках даної роботи були визначені та описані етапи проведення дослідження продукту. Такі процедури як проведення аудитів продукту різного рівня допомагають визначити слабкі місця, та розробити технологічне та бізнес рішення завдяки експертизі спеціалістів різної кваліфікації. За результатами дослідження проводиться опис обсягу робіт, та підготовка плану роботи проєкту, розробка графіку робіт, оцінка ризиків та визначення задач та цілей проєкту.

За допомогою інформації даного дослідження були розроблені засоби для управління проєктом міграції веб-сайту на іншу технологію. Застарілий

веб-сайт мав низку проблем, котрі впливали на прибутковість компанії та її майбутні можливості для розвитку.

Була проведена фаза дослідження стану наявного продукту зі сторін:

- Бізнес показників та прибутковості
- Технічних проблем, та розробки нової архітектури
- Подальшого розвитку

Для планування робіт такого проєкту була створена дорожня мапа та бюджет, котрі складалася з результатів оцінювання задач різними експертами. До уваги були взяті результати визначення зацікавлених осіб проєкту, та проведення аналізу ризиків, котрі сформували внутрішнє середовище проєкту. В результаті були розглянуті інструменти ітеративного контролю результатів проєкту та продукту. Ці інструменти визначають вплив на показники, котрі приносять цінність замовнику або власнику продукту.