

Міністерство освіти і науки України  
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «КИЄВО-МОГИЛЯНСЬКА АКАДЕМІЯ»  
Кафедра мультимедійних систем факультету інформатики

## УПРАВЛІННЯ ЦИФРОВИМИ ПРОДУКТАМИ

Текстова частина до курсової роботи  
за спеціальністю «Інженерія програмного забезпечення» 121

Керівник курсової роботи

Корнійчук Максим Анатолійович

*(прізвище та ініціали)*

*(підпис)*

“ \_\_\_\_ ” \_\_\_\_\_ 2021 р.

Виконла студентка

Немиря А.М.

*(прізвище та ініціали)*

“ \_\_\_\_ ” \_\_\_\_\_ 2021 р.

Київ 2022

## Календарний план виконання курсової роботи

**Тема: Управління цифровими продуктами**

**Календарний план виконання роботи:**

| №<br>п/п | Назва етапу курсової роботи                                                       | Термін<br>виконання<br>етапу | Примітка |
|----------|-----------------------------------------------------------------------------------|------------------------------|----------|
| 1.       | Отримання завдання на дипломну роботу.                                            | вересень 2021                |          |
| 3.       | Виконати аналіз сучасних методологій управління цифровими продуктами              | жовтень 2021                 |          |
| 3.       | Визначити методології, що будуть використовуватись у цій роботі. Створити команду | листопад 2021                |          |
| 4.       | Визначити вимоги до продукту                                                      | грудень 2021                 |          |
| 5.       | Створити дизайн сервісу                                                           | січень 2022                  |          |
| 6.       | Розробка сервісу                                                                  | березень 2022                |          |
| 7.       | Перевірка, тестування та запуск сервісу                                           | квітень 2022                 |          |
| 8.       | Написання текстової частини                                                       | квітень 2022                 |          |
| 8.       | Створення слайдів для доповіді та написання доповіді.                             | травень 2022                 |          |
| 10.      | Захист курсової роботи роботи                                                     | травень 2022                 |          |

Студентка Немир А.М.

Керівник Корнійчук М.А.

“        ”  
\_\_\_\_\_

## ЗМІСТ

|                                                                                                                   |    |
|-------------------------------------------------------------------------------------------------------------------|----|
| Перелік прийнятих скорочень .....                                                                                 | 5  |
| АНОТАЦІЯ .....                                                                                                    | 6  |
| ВСТУП.....                                                                                                        | 7  |
| РОЗДІЛ 1. Аналіз існуючих методів вирішення поставленої задачі.                                                   |    |
| Визначення ключових понять.....                                                                                   | 9  |
| 1.1. Управління продуктом.....                                                                                    | 9  |
| 1.2. Продукт.....                                                                                                 | 10 |
| 1.3. Менеджер продукту .....                                                                                      | 10 |
| 1.3. “Waterfall” управління продуктом .....                                                                       | 13 |
| 1.4. “Agile” управління продуктом .....                                                                           | 14 |
| 1.4. Scrum.....                                                                                                   | 14 |
| РОЗДІЛ 2. Постановка завдання та обґрунтування вибору методів його реалізації .....                               | 17 |
| 2.1. Постановка завдання .....                                                                                    | 17 |
| 2.2. Обґрунтування вибору методологій для вирішення поставленої задачі та визначення основних етапів роботи ..... | 17 |
| 2.3. Визначення основних етапів роботи над продуктом .....                                                        | 20 |
| РОЗДІЛ 3. Початок роботи над продуктом: створення команди, аналіз предметної області та визначення вимог.....     | 22 |
| 3.1. Створення команди .....                                                                                      | 22 |
| 3.2. Аналіз ЦА. Проведення інтерв’ю .....                                                                         | 22 |
| 3.3. Створення user stories та визначення вимог до продукту.....                                                  | 22 |
| РОЗДІЛ 4. Дизайн і розробка .....                                                                                 | 25 |
| 4.1. Організація робочого процесу.....                                                                            | 25 |

|                                     |    |
|-------------------------------------|----|
| 4.2. Результати роботи .....        | 26 |
| ВИСНОВКИ .....                      | 27 |
| СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ..... | 28 |

**Перелік прийнятих скорочень**

ФІ — Факультет Інформатики

НАКМА — Національний Університет «Києво-Могилянська Академія»

MVP — Minimum Viable Product

ЦА — цільова аудиторія

ІТ — Інформаційні Технології

## **АНОТАЦІЯ**

Метою цієї курсової роботи було описати процес управління розробкою сервісу для курсових робіт НАУКМА на основі методологій управління продуктом та скраму, визначити та уточнити практики, що найкраще підходять для реалізації подібних проектів.

## ВСТУП

У сучасному світі щодня створюються нові цифрові продукти, і часто конкурентну перевагу зумовлює не ідея продукту, а якість і швидкість її реалізації. Останні кілька десятиліть стало надзвичайно важливим розробляти цифрові продукти ефективно та гнучко, створюючи якомога більшу цінність як для клієнтів, так і для бізнесу. Ефективна розробка програмного забезпечення неможлива без якісного управління продукту. Останнім часом все більше компаній переходять від традиційних способів планування і управління до більш сучасних і гнучких, від Waterfall до Agile, Scrum та інших. Цей перехід активно відбувається і досі.

Зазначені методології створені відносно нещодавно і досі активно розвиваються. Саме тому ще немає напрацювань для усіх випадків їх використання, і часто менеджер продукту створює власні методики і підходи до роботи, коли вирішує нестандартні завдання у межах розробки. Ця робота досліджує використання методологій управління продуктом та Скраму при розробці цифрових продуктів у межах освітніх закладів або інших, схожих за специфікою організацій.

Метою цієї роботи було з'ясувати, які практики найкраще застосовувати при управлінні розробкою цифрових продуктів спеціалістами з порівняно невеликим досвідом, в умовах обмеженості ресурсів та високих показників невизначеності щодо вимог до кінцевого результату.

Завдання цієї роботи:

1. Дослідити методології управління продуктом;
2. Провести їх порівняльний аналіз та визначити, які з них найбільш ефективні для реалізації поставленої задачі;
3. Реалізувати ці методології на практиці, керуючи розробкою цифрового продукту;
4. Зробити висновки щодо доцільності використання цих методологій та уточнень, які вони можуть потребувати.

У результаті цієї роботи було створено продукт — сервіс для курсових робіт у НАУКМА у вигляді веб-сайту, що є частиною загальної інформаційної системи університету. До функціоналу цього сервісу входить запис студентів на курсові роботи, розподіл навантаження між викладачами, завантаження курсових робіт студентами і їх оцінювання науковими керівниками, та інше.



## РОЗДІЛ 1. Аналіз існуючих методів вирішення поставленої задачі.

### Визначення ключових понять

Проаналізуємо основні поняття, методики та концепції, що використовуються при управлінні цифровими продуктами. Обґрунтуємо вибір методологій, що будуть застосовуватись для реалізації проекту.

#### 1.1. Управління продуктом

Product management (управління продуктом) – це процес, який зосереджується на виведенні нового продукту на ринок або розробці вже існуючого. Він починається з ідеї продукту, з яким буде взаємодіяти клієнт, і закінчується оцінкою успіху продукту. Управління продуктом поєднує бізнес, розробку продукту, маркетинг і продажі. Дослідження показують, що ефективне управління продуктом може збільшити прибуток на 34,2 відсотка, що доводить ефективність цієї методології. [1] Одним із важливих напрямків управління продуктом є створення та документування стратегії продукту. Управління продуктом здійснює product manager (менеджер продукту). Відповідальність менеджера продукту полягає в тому, щоб вести продукт від виникнення ідеї до запуску, зосереджуючись на характеристиках, цінності для бізнесу та клієнтів.

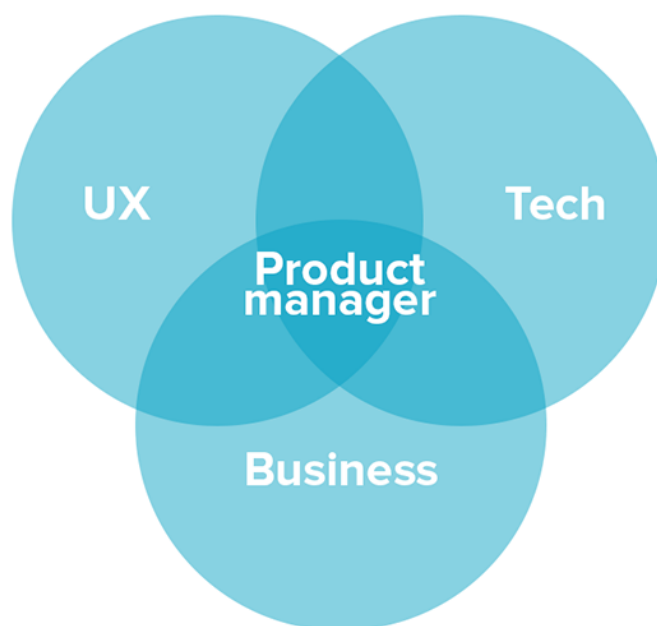


Рисунок 1.1 — зв'язок управління продуктом з іншими областями [1]

## 1.2. Продукт

Продукт — це будь-який предмет або послуга, що продається, щоб задовольнити потреби або бажання клієнта.[2] Він може бути фізичним або віртуальним. Фізичні продукти включають товари тривалого користування (наприклад, автомобілі, меблі та комп'ютери) і товари нетривалого користування (наприклад, продукти харчування та напої). Віртуальні продукти – це пропозиції послуг або досвіду (наприклад, освіта, програмне забезпечення та інші цифрові продукти). Продукт може бути гібридом і включати як фізичні, так і віртуальні елементи. Гібридні продукти стають все більш поширеними, оскільки традиційно аналогові продукти включають цифрові технології як спосіб кращого охоплення та обслуговування клієнтів.

## 1.3. Менеджер продукту

Product manager (менеджер продукту) визначає стратегію, план і необхідні функції для продукту або лінійки продуктів. Роль менеджера продукту також може включати відповідальність за маркетинг, прогнозування та прибутки та збитки. Менеджер продукту аналізує ринкові та конкурентні умови, формує бачення продукту, яке забезпечує унікальну цінність на основі клієнтських потреб. [7]

# What Does a Product Manager Do?

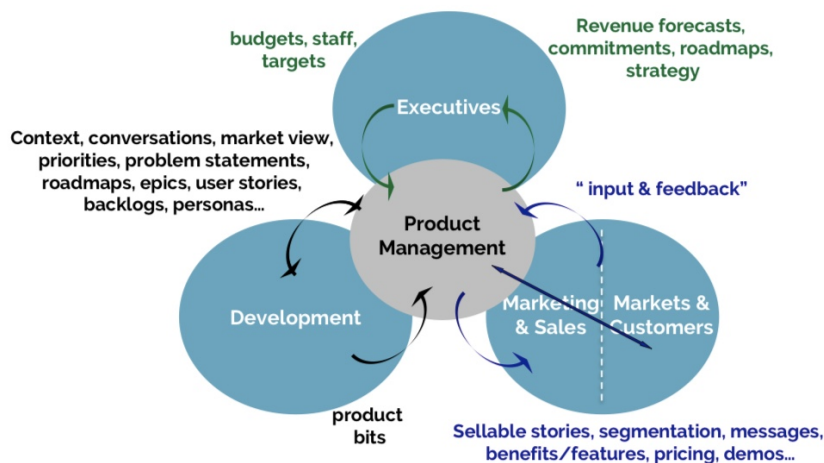


Рисунок 1.2 — роль та обов'язки менеджера продукту[8]

Основні обов'язки менеджера продукту зазвичай є такими:

### *1. Визначення стратегії*

Менеджер продукту несе відповідальність за визначення бачення та стратегічного напрямку свого продукту. Він повинен вміти чітко сформулювати бізнес-обґрунтування певної ініціативи чи функції, щоб команда розуміла, навіщо була створена і яка мета її роботи.

Стратегічне планування передбачає визначення основних напрямків інвестування ресурсів, щоб була можливість визначити пріоритети того, що є найважливішим для досягнення цілей продукту. Менеджер також відповідає за дорожню карту (roadmap) продукту — часову шкалу, яка візуалізує, що і коли буде відбуватись у розробці.

### *2. Визначення релізів*

Менеджер продукту втілює стратегію продукту в заплановану роботу — визначаючи, що команда буде створювати і коли це буде запущено. Це справедливо незалежно від того, яку методологію розробки використовує команда інженерів.

Менеджер відповідає за керування процесом випуску та міжфункціональними залежностями — усі дії, необхідні для виведення нових продуктів і функцій на ринок. Це передбачає подолання розривів між різними функціями всередині компанії та комунікацію між ключовими командами, включаючи маркетинг, продажі та підтримку клієнтів.

### *3. Оцінювання ідей*

Кожна організація прагне кращих ідей для успішного продукту. Менеджер продукту відповідає за краудсорсинг, розробку та кураторство ідей, які принесуть користь клієнтам. Він володіє процесом управління ідеями в організації та визначає, які ідеї слід просувати, щоб просувати стратегію продукту вперед.

З цією метою менеджер продукту також забезпечує інтеграцію зворотнього зв'язку та запитів у процес планування та розробки продукту. Він повідомляє про статус ідей клієнтам, партнерам і внутрішнім членам команди, які їх висунули.

#### *4. Визначення пріоритетів*

Менеджер продукту визначає пріоритети функцій, оцінюючи їх згідно зі стратегічними цілями та ініціативами. Він приймає рішення виходячи з цінності, яку нова функція може надати клієнтам і бізнесу.

Він також несе відповідальність за визначення пропонованих вимог і бажаного досвіду користувача. Він тісно співпрацює з інженерами над технічними специфікаціями і гарантує, що команди мають всю необхідну інформацію, щоб вивести на ринок повний продукт.

#### *5. Створення та обмін стратегічними дорожніми картами*

Створення та оновлення дорожньої карти продукту (roadmap) є одним із найпотужніших комунікаційних інструментів, які має менеджер продукту. Дорожня карта продукту візуалізує, як продукт досягне бізнес-цілей, і допомагає продовжувати роботу.

Існує багато різних типів дорожніх карт, які можна створити залежно від того, кому вони презентуються і що мають на меті донести. Керівники, як правило, хочуть знати плани високого рівня, тоді як інженерам та дизайнерам потрібно розуміти точні терміни та послідовність виконання важливої роботи.

#### *6. Аналіз і звітність по результатах роботи*

Менеджер з продукту має бути зосереджений на результатах — для клієнтів і для бізнесу в цілому. Він повинен мати повне уявлення про прогрес у досягненні цілей, щоб розуміти, як працює продукт.

### 1.3. “Waterfall” управління продуктом

Waterfall product management (управління продуктом за методикою «водоспаду») відноситься до послідовної моделі для планування, створення та презентування нових продуктів і функцій. Вимоги до будь-якого нового клієнтського досвіду визначаються заздалегідь і реалізуються на окремих етапах. Кожна фаза має конкретні кроки, які мають бути задокументовані та затверджені, перш ніж почати наступний етап.[3]

Модель «водоспаду» часто використовується при управлінні виробництвом та доставкою апаратних виробів, наприклад, у виробництві та будівництві. У цих галузях часто важливо визначити повний обсяг перед початком роботи, оскільки зміни є дорогими і їх важко врахувати в процесі розробки або виробництва. Управління діяльністю здійснюється лінійно, щоб забезпечити виконання вимог. Готовий продукт або функція поставляється в повному вигляді.

Розглянемо детальніше, як працює модель водоспаду при розробці ІТ-продукту. Є шість етапів. Кожен етап є окремим і має бути завершеним, щоб почати наступний. У таблиці нижче наведено послідовність етапів та огляд того, що необхідно виконати у кожному з них.

| Етап         | Опис                                                                                                                                                                                                                                                    |
|--------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Аналіз вимог | Вимоги клієнта враховуються в документі з вимогами до продукту. Визначаються будь-які обмеження проекту — наприклад, час, бюджет і залежності — і припущення, і створюється план розробки. Аналіз доцільності визначає загальну життєздатність проекту. |
| Проектування | На етапі проектування деталізуються всі вимоги, такі як архітектурні, інтерфейсні; дані, процеси та фізичні вимоги, необхідні для реалізації проекту. Ці вимоги зазначаються в технічній документації проекту.                                          |
| Розробка     | Вихідний код розробляється на основі вимог. Зазвичай система будується і тестується невеликими блоками. Блоки інтегруються, а потім весь продукт переходить до фази тестування.                                                                         |
| Тестування   | Новий продукт або функція повністю перевіряється, щоб переконатися, що він відповідає зазначеним вимогам. Будь-які проблеми чи помилки документуються та                                                                                                |

|                         |                                                                                                                                                                                                                     |
|-------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                         | повідомляються команді розробників, щоб їх можна було виправити.                                                                                                                                                    |
| Запуск                  | Після того, як система повністю функціонує та відповідає критеріям прийняття, її впроваджують у виробниче середовище та надають клієнтам.                                                                           |
| Технічне обслуговування | Фаза технічного обслуговування зосереджується на безперебійній роботі системи. Це включає виправлення будь-яких дефектів, про які повідомляють клієнти, і випуск удосконалень для покращення та оновлення продукту. |

Таблиця 1.1 — етапи розробки продукту за методологією «водоспаду»

#### 1.4. “Agile” управління продуктом

Agile product management (гнучке управління продуктом) — це методологія, ідея якої полягає у тому, щоб визначати стратегію продукту та розробляти продукт у гнучкому середовищі. Agile заохочує адаптивний підхід до планування та запуску продуктів, щоб організації могли швидко реагувати на відгуки та створювати продукти, які дійсно сподобаються клієнтам.

По суті, гнучке управління продуктом є узагальненням широко використовуваних гнучких методологій розробки програмного забезпечення, таких як scrum або kanban. Ці фреймворки наголошують на еволюційному розвитку, ранньому доставленні та постійному вдосконаленні.

#### 1.4. Scrum

Scrum (скрам), згідно офіційного визначення його творців у Scrum Guide — це структура, яка допомагає людям, командам і організаціям створювати цінність за допомогою адаптивних рішень для складних проблем. [4]

Суть методології полягає у послідовності таких кроків:

1. Product owner (власник продукту) визначає, яку роботу необхідно виконати, і додає завдання у беклог продукту.
2. Команда виконує завдання, що зумовлюють приріст вартості під час спринту.

3. Команда та усі зацікавлені сторони перевіряють результати та коригують стратегію до наступного спринту.
4. Ці дії повторюються у кожному наступному спринті, аж доки продукт не буде готовим.

Спринти — це події фіксованої тривалості одного місяця або менше. Новий спринт починається відразу після завершення попереднього спринту. Вся робота, необхідна для досягнення цілей продукту, включно з планування спринту, щоденні зустрічі, огляд спринту та ретроспективу спринту, виконується в рамках спринтів.

Структура Scrum спеціально створювалась як не повна і не чітко визначена, вона визначає лише частини, необхідні для реалізації теорії Scrum. Scrum заснований на колективному розумі людей, які його використовують. Замість того, щоб надавати людям детальні інструкції, правила Scrum керують їхніми стосунками та взаємодією.

## SCRUM FRAMEWORK

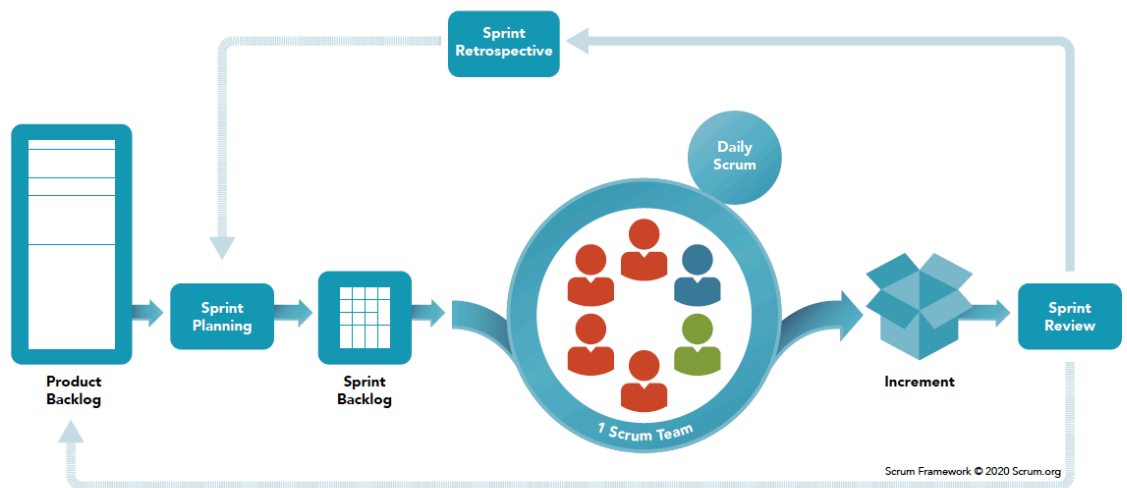


Рисунок 1.3 — етапи розробки продукту за скрамом [5]

Філософія скраму опирається на п'ять основних цінностей: відданість, цілеспрямованість, відкритість, повага та сміливість. [4] Команда, що працює за

скрамом, прагне досягати своїх цілей і підтримувати одне одного. Основна увага членів команди зосереджена на завданнях кожного спринта та на досягненні його цілей. Команда та її зацікавлені сторони відкриті до роботи та нових викликів. Члени скрам-команди поважають один одного як здібних, незалежних людей, і їх поважають як таких люди, з якими вони працюють. Члени скрам-команди мають сміливість робити правильні речі, працювати над важкими проблемами. Ці цінності дають напрямок команді щодо їх роботи, дій та поведінки. Прийняті рішення, зроблені кроки та спосіб використання скраму повинні зміцнювати ці цінності, а не зменшувати чи підривати їх. Члени скрам-команди вивчають і досліджують цінності, працюючи з подіями та артефактами скраму. Коли ці цінності втілюються командою та людьми, з якими вони працюють, емпіричні стовпи скраму — прозорість, перевірка та адаптація — оживають, створюючи довіру.



Рисунок 1.3 — цінності фреймворку Скрам[6]



## **РОЗДІЛ 2. Постановка завдання та обґрунтування вибору методів його реалізації**

У цьому розділі буде описано завдання практичної частини цієї роботи, а також розглянуто, які з вищеописаних методик застосовувались у даному проекті і чому, визначено основні етапи роботи.

### **2.1. Постановка завдання**

Завданням практичної частини цієї роботи було керування створенням сервісу курсових робіт для Києво-Могилянської Академії. Це включає в себе аналіз предметної області, проведення інтерв'ю з цільовою аудиторією та стейкхолдерами, визначення завдань у рамках проекту та послідовності їх реалізації, визначення чітких вимог та перевірку їх дотримання, координацію та підтримку команди, комунікацію з усіма зацікавленими сторонами, звітність, адаптацію у залежності від обставин, що виникають під час розробки, тобто виконання ролі продуктового менеджера, як вона описана вище.

### **2.2. Обґрунтування вибору методологій для вирішення поставленої задачі та визначення основних етапів роботи**

Розглянемо, чому для реалізації поставленої задачі було обрано саме методологію продакт менеджменту. Сервіс для роботи з курсовими роботами НУКМА повністю відповідає визначенню продукту, адже це є система, створена що задовольняє потреби клієнтів (студентів, викладачів та методистів університету), створює для них цінність, полегшуючи виконання багатьох завдань і роблячи їх реалізацію більш ефективною, а також створює цінність для стейкхолдерів (адміністрації університету). До того ж, сервіс матиме активну фазу розробки, а потім довготривалу фазу підтримки та розвитку. Звідси було зроблено висновок, що практики управління продуктом будуть позитивно впливати на розробку даного сервісу.

Після того, як було обрано основну методику управління проектом, постало питання вибору фреймворку, за яким відбуватиметься робота команди. Було проведено порівняльний аналіз фреймворків Waterfall та Agile, частиною якого є Scrum, стосовно розробки цифрових продуктів.

| <b>Agile</b>                                                                                                                                                                             | <b>Waterfall</b>                                                                                                                      |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------|
| Поділяє життєвий цикл розробки проекту на спринти.                                                                                                                                       | Процес розробки програмного забезпечення поділяється на окремі, чітко визначені етапи.                                                |
| Дотримується поступового підходу.                                                                                                                                                        | Послідовний процес проектування, де кожний блок розробляється одразу повністю, і до нього вже не повертаються.                        |
| Методологія Agile відома своєю гнучкістю.                                                                                                                                                | Waterfall — це структурована методологія розробки програмного забезпечення, тому в більшості випадків вона може бути досить жорсткою. |
| Agile можна розглядати як сукупність багатьох різних проектів.                                                                                                                           | Розробка програмного забезпечення буде завершена як один проект.                                                                      |
| Agile є досить гнучким методом, який дозволяє вносити зміни в вимоги до розробки проекту, навіть якщо початкове планування було завершено.                                               | Немає можливості змінювати вимоги після початку розробки проекту.                                                                     |
| Дотримується ітераційного підходу до розробки, оскільки планування, розробка, створення прототипів та інші етапи розробки програмного забезпечення можуть з'являтися більше одного разу. | Усі етапи розробки проекту, такі як проектування, розробка, тестування тощо, завершуються один раз.                                   |
| План тестування переглядається після кожного спринту.                                                                                                                                    | План тестування рідко обговорюється і змінюється після початку тестування.                                                            |
| Процес, у якому очікується, що вимоги будуть змінюватися та розвиватися.                                                                                                                 | Метод ідеально підходить для проектів, які мають певні вимоги, щодо яких зміни зовсім не очікуються.                                  |
| У методології Agile тестування виконується одночасно з розробкою програмного забезпечення.                                                                                               | У цій методології фаза тестування настає чітко після етапу розробки.                                                                  |
| Філософія Agile фокусується на тому, щоб програмний продукт задовольняв потреби своїх кінцевих клієнтів і                                                                                | Ця модель повністю зосереджена на виконанні проекту у такій формі, в якій                                                             |

|                                                                                                                                                       |                                                                                                              |
|-------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------|
| змінювався відповідно до вимог замовника.                                                                                                             | його було запланованого з самого початку.                                                                    |
| Agile методологія надзвичайно добре працює з нефіксованим фінансуванням. Це може посилити стрес у сценаріях з фіксованою ціною.                       | Зменшує ризик у контрактах фірми з фіксованою ціною шляхом отримання угоди про ризик на початку процесу.     |
| Віддає перевагу невеликим, але цілеспрямованим командам з високим ступенем координації та синхронізації.                                              | Координація/синхронізація команди дуже обмежена.                                                             |
| Власник продукту разом із командою готує вимоги майже щодня під час проекту.                                                                          | Бізнес-аналітик готує вимоги до початку проекту.                                                             |
| Команда тестування може без проблем брати участь у зміні вимог.                                                                                       | Тестувальникам важко ініціювати будь-яку зміну вимог.                                                        |
| Опис деталей проекту можна змінити в будь-який час під час процесу SDLC.                                                                              | Детальний опис потребує реалізації Waterfall підходу до розробки програмного забезпечення.                   |
| Члени Agile команди взаємозамінні, в результаті вони працюють швидше. Також немає потреби в менеджерах проектів, оскільки проектами керує вся команда | У методі водоспаду процес завжди простий, тому менеджер проекту відіграє важливу роль на кожному етапі SDLC. |

Таблиця 2.1 — порівняльний аналіз методологій Agile і Waterfall [9]

## Waterfall vs. Agile

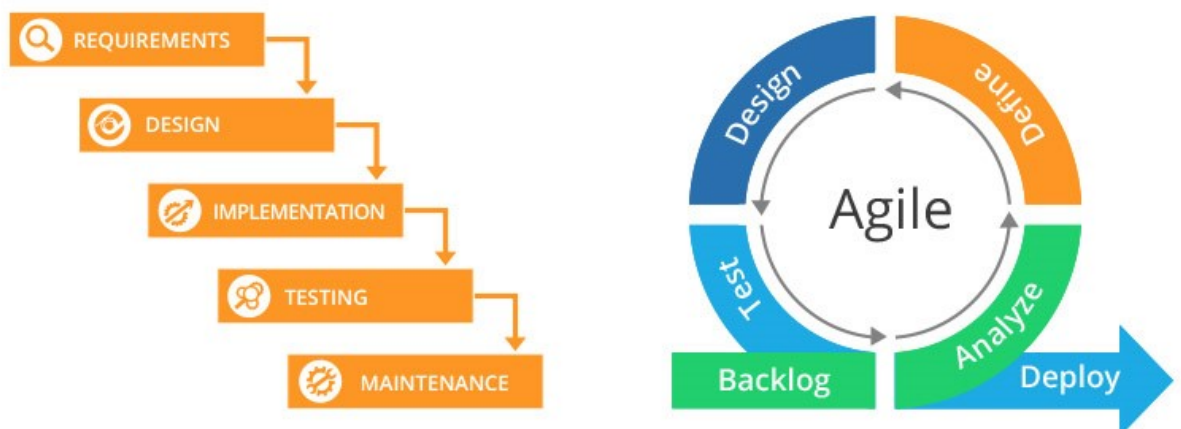


Рисунок 2.1 — порівняння послідовності роботи у Waterfall і Agile [10]

Розробка сервісу для роботи з курсовими планувалася у невеликій команді, адже цей продукт є некомерційним і команда мала формуватись зі студентів ФІ, до того ж, співвідношення необхідного функціоналу та часу, виділеного на розробку дозволяло зробити висновок, що команду можна буде зформувати з 5-7 людей. Чітку послідовність дій та довгострокову стратегію на початку роботи було визначити дуже складно, враховуючи те, що сервіс, про який іде мова, мав стати частиною загальної інформаційної системи університету, яка також перебувала у стадії розробки. Допускалося, що члени команди будуть займатись розробкою функціоналу, що стосуватиметься не тільки курсових робіт, але й зберігання та оновлення інформації, документообігу тощо. Конкретні довгострокові часові обмеження по розробці також було б даремно встановлювати, адже розробка даного продукту не була основною зайнятістю для жодного з членів команди, і їх внесок коливався в залежності від їх навантаження. До того ж, студенти могли зустрітись у розробці з викликами, яких не могли передбачити і оцінити час на їх подолання, адже є спеціалістами з порівняно невеликим досвідом. Усі вищенаведені критерії зумовили вибір в бік гнучкого управління на противагу фреймворку Waterfall. А точніше, було обрано фреймворк Scrum як найбільш придатний для реалізації усіх задач проекту.

### **2.3. Визначення основних етапів роботи над продуктом**

Робота за методологією Скрам відбувається у спринтах, що йдуть один за одним до завершення роботи над продуктом. Це звільняє менеджера продукту від необхідності детального і послідовного планування усіх етапів розробки перед початком роботи. Однак основні етапи все-таки потрібно визначати одразу, особливо у проектах, які мають хоча б приблизні часові обмеження (як той, про який іде мова у цій роботі). Це полегшує комунікацію з усіма учасниками процесу, дозволяє ефективніше визначати пріоритети під час планування спринтів, планувати залучення ресурсів і так далі.

Для роботи над сервісом роботи з курсовими було визначено такі основні етапи:

1. Створення команди;
2. Аналіз наявної інформаційної системи, на основі якої буде здійснюватись розробка;
3. Інтерв'ю зі стейкхолдерами та ЦА;
4. Оформлення User Stories та вимог до продукту;
5. Візуальний дизайн нового сервісу;
6. Розробка сервісу;
7. Тестування;
8. Запуск продукту, аналіз фідбеку від користувачів;
9. Моніторинг та поступове покращення сервісу у разі потреби.

Зазначимо, що при використанні фреймворку Скрам допускається непослідовна або паралельна реалізація етапів, і ця практика широко використовувалась при розробці даного продукту.

## **РОЗДІЛ 3. Початок роботи над продуктом: створення команди, аналіз предметної області та визначення вимог**

### **3.1. Створення команди**

Було створено скрам-команду такого складу:

- власник продукту;
- дизайнер;
- два фронтенд-розробники;
- два бекенд-розробники.

Усі члени команди були студентами ФІ НАУКМА. Стейкхолдером даного продукту виступав декан факультету Андрій Миколайович Глибовець.

### **3.2. Аналіз ЦА. Проведення інтерв'ю**

Було визначено цільову аудиторію (клієнтів) сервісу:

- студенти НАУКМА;
- методисти НАУКМА;
- викладачі НАУКМА.

З кожною групою ЦА було проведено по кілька розмовних інтерв'ю, під час яких було зафіксовано потреби та побажання майбутніх користувачів сервісу.

### **3.3. Створення user stories та визначення вимог до продукту**

На основі проведених інтерв'ю було сформовано історії користувачів, що описують необхідний функціонал сервісу.

|                 |                                                                                |                                                      |                                                                     |                                                         |                                  |                                                       |                                                         |                                    |
|-----------------|--------------------------------------------------------------------------------|------------------------------------------------------|---------------------------------------------------------------------|---------------------------------------------------------|----------------------------------|-------------------------------------------------------|---------------------------------------------------------|------------------------------------|
| Студент         |                                                                                |                                                      |                                                                     |                                                         |                                  |                                                       |                                                         |                                    |
| User activities | завантажує курсову                                                             |                                                      | переглядає оцінку за курсову                                        |                                                         | переглядає відсоток плагіату     |                                                       | захищає перед комісією                                  |                                    |
| User tasks      |                                                                                |                                                      |                                                                     |                                                         |                                  |                                                       |                                                         |                                    |
| User stories    | Завантажує файл з курсовою                                                     | Хоче мати можливість видалити/оновити файл           | Переглядає оцінку там, де завантажив файл                           | Хоче отримати повідомлення, коли рцінка буде виставлена | Бачить статус роботи: перевірена | Отримує повідомлення, що робота перевірена на плагіат | Бачить список викладачів, перед якими буде захищати     | Бачить розклад захисту             |
| Release 1       |                                                                                |                                                      |                                                                     |                                                         |                                  |                                                       |                                                         |                                    |
|                 | Хоче бачити інформацію про дедлайн                                             | Хоче отримувати попередження про наближення дедлайну | Хоче бачити складові оцінки: від наукового керівника та від комісії |                                                         |                                  |                                                       | Отримує повідомлення, коли списки і розклад з'являються | Отримує комунікацію, де захищатиме |
| Release 2       | При завантаженні курсової бачить попередження про допустимий відсоток плагіату |                                                      |                                                                     |                                                         |                                  |                                                       |                                                         |                                    |
|                 |                                                                                |                                                      |                                                                     |                                                         |                                  |                                                       |                                                         |                                    |

Рисунок 3.1 — user stories студента у весінньому триместрі

|                 |                                                                                |                                                                                           |                                                                                                    |                                          |                                              |                                                                                                    |                                                                                            |  |
|-----------------|--------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------|------------------------------------------|----------------------------------------------|----------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------|--|
| User persona    | Викладач                                                                       |                                                                                           |                                                                                                    |                                          |                                              |                                                                                                    |                                                                                            |  |
| User activities | Заходить на сайт                                                               |                                                                                           | Створює/редагує теми                                                                               |                                          | Переглядає і підтверджує заявки студентів    |                                                                                                    | Переглядає наявні теми курсових                                                            |  |
| User stories    |                                                                                |                                                                                           |                                                                                                    |                                          |                                              |                                                                                                    |                                                                                            |  |
| Release 1       | Хоче бачити повідомлення, коли сесія авторизації закінчилась і треба перезайти |                                                                                           | Хоче бачити, що тему успішно створено. Користувач після створення може потрапляти на "мої курсові" | Має працювати обмеження на кількість тем | Має бути зрозуміло, де заявки                | Система має контролювати, що заявка підтверджується тільки одному студенту. Іншим відхиляється (?) | Має бути видно, що працює фільтр (після застосування фільтру з'являється нагис, наприклад) |  |
|                 | Хоче чітко бачити своє навантаження на цей рік (без навантаження інших)        | Хоче бачити скільки тем вже створено, скільки ще потрібно, на скільки записалися студенти | Має бути кнопка "подати заявку на збільшення навантаження"                                         |                                          | Має бути зрозуміло, як підтвердити заявку    | Треба мати можливість відмінити підтвердження заявки                                               |                                                                                            |  |
| Release 2       | Хоче подивитись свою роль у сервісі                                            |                                                                                           |                                                                                                    |                                          | Нотифікація після створення заявки студентом | Аргументація студента має бути помітною                                                            |                                                                                            |  |
|                 |                                                                                |                                                                                           |                                                                                                    |                                          |                                              |                                                                                                    |                                                                                            |  |

Рисунок 3.2 — user stories викладача в осінньому триместрі

На основі історій користувачів було визначено вимоги до сервісу.

Вимоги до функціоналу, що стосується осіннього триместру:

- Розподіл методистами навантаження викладачів;
- Публікація викладачами тем курсових, які вони пропонують;
- Подання студентами заявок на курсові, що їх цікавлять;
- Прийом заявок викладачами або відмова;
- Автоматичне припинення запису до викладача за умови, що його навантаження повністю заповнене.

Вимоги до функціоналу, що стосується весняного триместру:

- Завантаження курсових робіт студентами;
- Перегляд та оцінка курсових науковими керівниками;
- Перегляд та оцінка курсових комісією.



## РОЗДІЛ 4. Дизайн і розробка

### 4.1. Організація робочого процесу

Робочий процес було організовано за допомогою спринтів, кожен з яких тривав один тиждень. Щотижня проводилась командна зустріч, на якій проходила ретроспектива попереднього спринта та визначення задач на наступний. Комунікація щодо беклогу, завдань на спринт, виконаних завдань, проблем і так далі відбувалася на дошці Trello.

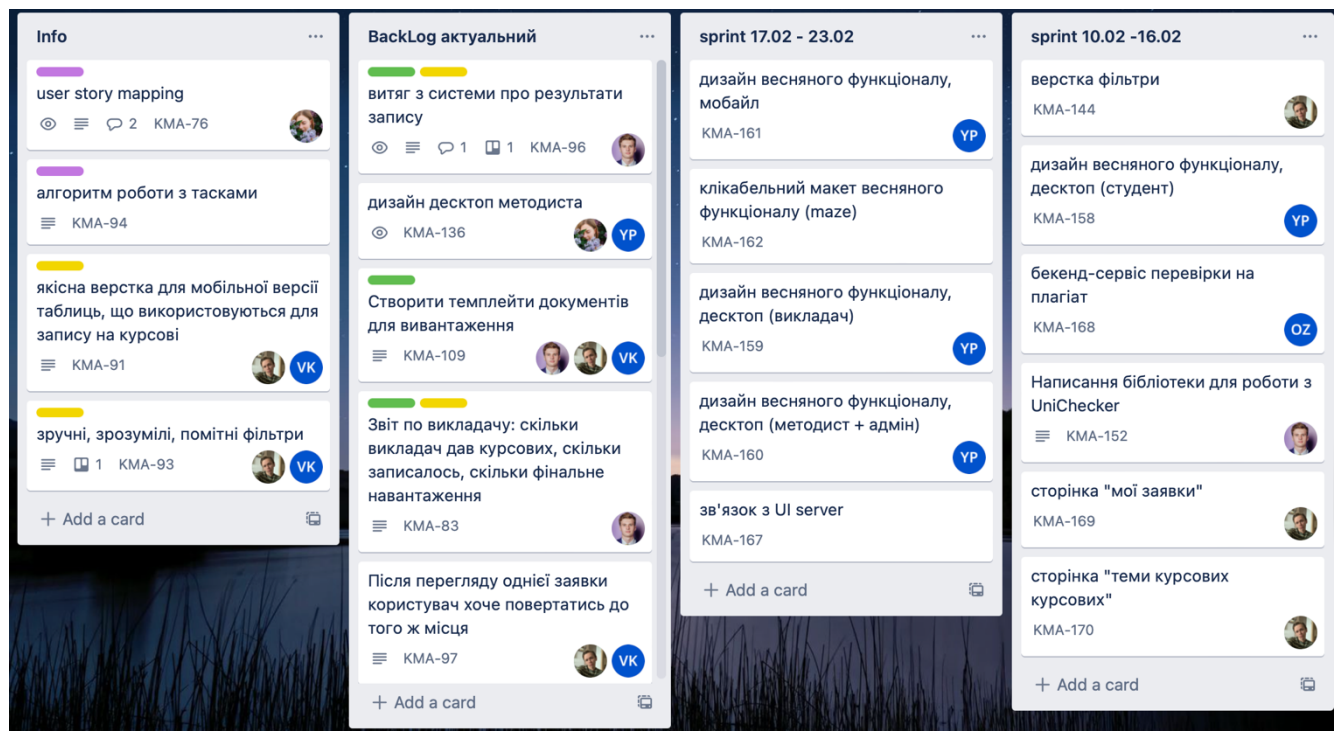


Рисунок 4.1 — дошка Trello в процесі роботи

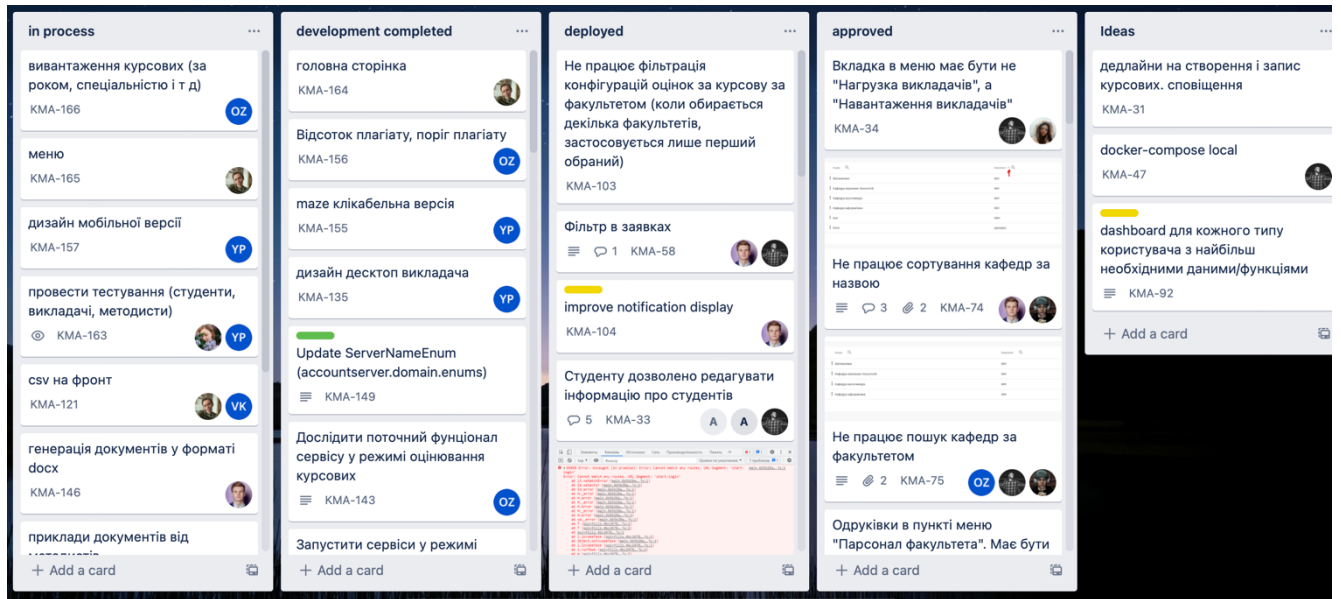


Рисунок 4.2 — дошка Trello в процесі роботи

## 4.2. Результати роботи

У результаті роботи команди було розроблено сервіс для курсових НАУКМА, що відповідає більшості визначених вимог і готовий до запуску. Через непередбачувані обставини у країні навесні 2022 року проект не вдалося завершити таким чином, як планувалось, однак основні поставлені задачі було виконано.

Окрім цього сервісу розробники реалізували сервіс генерації документів та покращили загальну інформаційну систему університету, у яку повинен бути інтегрований продукт, що розглядається в цій роботі.

## ВИСНОВКИ

Проаналізовано сучасні методології управління розробкою цифрових продуктів та застосовано їх на практиці. Реалізовано розробку цифрового продукту — сервісу для роботи з курсовими НАУКМА. Ми переконались на практиці, що методології управління продуктом та Scrum якнайкраще підходять для реалізації подібних проектів у схожих умовах. Однак варто брати до уваги необхідність за можливості заздалегідь визначати об'єми роботи, послідовність виконання завдань та розподіл часу між частинами проекту, особливо в умовах обмежених ресурсів.

Реалізований продукт має великий потенціал щодо його покращення та більш широкого використання. Рекомендовано аналізувати відгуки користувачів, вимоги до нового функціоналу від стейкхолерів та покращувати сервіс відповідно. Також планується краще інтегрувати реалізований функціонал у загальну інформаційну систему університету та надалі розвивати її комплексно.

## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Product management: main stages and product manager role:  
<https://www.altexsoft.com/blog/business/product-management-main-stages-and-product-manager-role/>  
Дата звернення: 18.05.2022
2. What is a product? : <https://www.aha.io/roadmapping/guide/product-management/what-is-a-product>  
Дата звернення: 17.05.2022
3. What is waterfall product management?:  
<https://www.aha.io/roadmapping/guide/product-development-methodologies/what-is-waterfall>
4. The 2020 Scrum Guide: <https://scrumguides.org/scrum-guide.html>  
Дата звернення: 20.05.2022
5. The Scrum framework poster: <https://www.scrum.org/resources/scrum-framework-poster>  
Дата звернення: 20.05.2022
6. What are the 5 Scrum values and why are they important?  
<https://backlog.com/blog/what-are-the-5-scrum-values-and-why-are-they-important/>  
Дата звернення: 20.05.2022
7. What is the role of a product manager?:  
<https://www.aha.io/roadmapping/guide/product-management/what-is-the-role-of-a-product-manager>  
Дата звернення: 20.05.2022
8. Why we need product management, with Rich Mironov:  
<https://brainmates.com.au/product-management/why-we-need-product-management/>  
Дата звернення: 20.05.2022

9. Agile vs Waterfall: Know the Difference Between Methodologies :

<https://www.guru99.com/waterfall-vs-agile.html>

Дата звернення: 20.05.2022

10. Waterfall Vs Agile Project Management: Which One is Better for Your Business?

: <https://www.digite.com/es/blog/waterfall-vs-agile-project-management/>

Дата звернення: 17.05.2021

