

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ОДЕСЬКА ПОЛІТЕХНІКА»
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ШТУЧНОГО ІНТЕЛЕКТУ ТА
РОБОТОТЕХНІКИ
КАФЕДРА ШТУЧНОГО ІНТЕЛЕКТУ ТА АНАЛІЗУ ДАНИХ

М.А. Годовиченко, С.А. Устенко

ЗАВДАННЯ НА ЛАБОРАТОРНІ РОБОТИ
з дисципліни
«Гнучкі методології управління ІТ-проектами»

Спеціальність:
122 Комп'ютерні науки

Освітня програма:
Інтелектуальний аналіз даних

ЛАБОРАТОРНА РОБОТА 1

Тема: Створення користувацьких історій, формування беклогу продукту.

Мета роботи:

- ознайомитися з поняттям користувацьких історій (User Stories) та їхнім формуванням;
- навчитися складати беклог продукту на основі користувацьких історій;
- освоїти процес пріоритезації завдань у беклозі;
- визначити базову структуру продукту для подальшої розробки у Scrum.

ЗАВДАННЯ

Для успішного виконання лабораторної роботи необхідно виконати наступні завдання:

- ознайомитися з основами Scrum, поняттям беклогу продукту та користувацьких історій;
- вибрати один із запропонованих ІТ-продуктів або запропонувати власний (за погодженням із викладачем);
- описати призначення продукту та його цільову аудиторію (коротке пояснення, які проблеми він вирішує);
- сформулювати не менше 10 користувацьких історій у стандартному форматі: *як [тип користувача], я хочу [функціонал], щоб [користь від функціоналу]*.
- для кожної історії визначити пріоритетність (високий, середній, низький);
- оцінити кожну історію за складністю у Story Points (наприклад, від 1 до 8);
- представити результати у вигляді таблиці беклогу продукту;
- оформити звіт та зробити висновки щодо процесу створення беклогу.

ПОРЯДОК ВИКОНАННЯ РОБОТИ

Крок 1: Вивчення теоретичного матеріалу. Перед початком виконання практичної частини необхідно ознайомитися з такими поняттями:

- Scrum та беклог продукту – список завдань для розробки продукту;
- Користувацькі історії – спосіб опису функціональності з точки зору користувача;
- Story Points – умовна оцінка складності завдань у Scrum;
- Пріоритезація беклогу – процес визначення важливості завдань.

Крок 2: Вибір IT-продукту. Здобувач вибирає один із варіантів додатків або пропонує власний. Варіанти додатків наведені нижче.

Крок 3: Опис продукту та цільової аудиторії. Здобувач коротко описує, що це за продукт, для кого він створений та які проблеми вирішує.

Приклад: «Застосунок для управління фінансами допоможе користувачам контролювати свої доходи та витрати, планувати бюджет і отримувати фінансові звіти. Основна аудиторія – молодь, фрілансери та офісні працівники.»

Крок 4: Формування користувацьких історій. Здобувач створює не менше 10 історій користувачів у наступному форматі: «Як [тип користувача], я хочу [функціонал], щоб [користь від функціоналу].»

Приклад для додатку з управління фінансами:

- як користувач, я хочу зареєструватися через Google, щоб швидко отримати доступ до додатку;
- як користувач, я хочу додавати витрати вручну, щоб контролювати свої фінанси;

– як користувач, я хочу автоматично синхронізувати свої витрати з банківської карти, щоб не вводити їх вручну;

– як користувач, я хочу переглядати аналітику витрат у вигляді графіків, щоб краще розуміти свої фінансові звички;

– як користувач, я хочу отримувати нагадування про важливі платежі, щоб не забувати оплачувати рахунки.

Крок 5: Пріоритезація користувацьких історій. Кожну історію потрібно оцінити за важливістю:

- високий пріоритет – критично необхідна функція;
- середній пріоритет – важлива, але не критична функція;
- низький пріоритет – додатковий функціонал, який можна реалізувати пізніше.

Крок 6: Оцінка складності завдань у Story Points. Кожна історія оцінюється за складністю у Story Points (наприклад, 1 – дуже легко, 8 – складне завдання).

Крок 7: Формування таблиці беклогу продукту. Усі історії заносяться до беклогу у вигляді таблиці.

Крок 8: Оформлення звіту. Звіт повинен містити:

- тему та мету роботи.
- опис обраного продукту та його цільової аудиторії.
- таблицю беклогу продукту з користувацькими історіями.
- висновки щодо процесу формування беклогу.

ФОРМУВАННЯ КОРИСТУВАЦЬКИХ ІСТОРІЙ У SCRUM

Що таке користувацька історія (User Story)?

Користувацька історія (User Story) – це коротке і зрозуміле текстове описання функціоналу або потреби з точки зору користувача. Вона пояснює, яку функціональність має реалізувати команда розробників для того, щоб задовольнити потреби кінцевого користувача.

Користувацькі історії є основним елементом беклогу продукту (Product Backlog) в методології Scrum.

Структура користувацької історії

Користувацькі історії зазвичай пишуться у наступному форматі:

Як [тип користувача], я хочу [функціонал або дію], щоб [користь від функціоналу або причина].

Цей підхід дає змогу зрозуміти:

- хто буде використовувати функцію (тип користувача);
- яку конкретну дію або функцію потрібно реалізувати;
- чому це важливо для користувача або яку проблему вирішує функція.

Критерії якісної користувацької історії (формат INVEST)

Для того, щоб користувацька історія була якісною, вона повинна відповідати стандарту INVEST:

- I (Independent) – незалежна: історія повинна бути самодостатньою, її можна виконати незалежно від інших історій;
- N (Negotiable) – обговорювана: історію можна адаптувати на основі обговорення між замовником і командою;

- V (Valuable) – цінна: історія повинна приносити реальну цінність користувачам;
- E (Estimable) – оцінювана: завдання повинно мати зрозумілий обсяг робіт;
- S (Small) – невелика: історія повинна бути невеликою за обсягом роботи (розбита на окремі завдання, якщо потрібно);
- T (Testable) – тестована: має бути можливість перевірити, чи функція працює правильно.

Приклади користувацьких історій

Приклад для мобільного додатку з управління фінансами:

- як користувач, я хочу зареєструватися через Google, щоб швидко отримати доступ до застосунку;*
- як користувач, я хочу вводити свої доходи вручну, щоб відстежувати свої фінансові потоки;*
- як користувач, я хочу переглядати аналітику витрат у вигляді графіків, щоб краще розуміти структуру моїх витрат;*
- як користувач, я хочу отримувати нагадування про важливі платежі, щоб уникнути прострочень.*

Приклад для платформи бронювання коворкінгів:

- як користувач, я хочу шукати коворкінги за містом і ціною, щоб знайти доступне робоче місце поблизу;*
- як користувач, я хочу переглядати доступні місця на певну дату, щоб планувати свій робочий день;*
- як користувач, я хочу здійснювати оплату онлайн, щоб забронювати місце заздалегідь;*
- як адміністратор, я хочу бачити всі бронювання, щоб контролювати наявність місць у коворкінгу.*

Приклад для мобільного додатку для здорового харчування:

- як користувач, я хочу вводити свої фізичні параметри (вага, зріст, вік), щоб отримати індивідуальні рекомендації з харчування;*
- як користувач, я хочу генерувати меню на день або тиждень, щоб дотримуватися дієти;*
- як користувач, я хочу отримувати сповіщення про прийом їжі, щоб не пропускати важливі прийоми;*
- як користувач, я хочу бачити список покупок для обраного меню, щоб легко купувати необхідні продукти.*

Етапи створення користувацьких історій

Крок 1: Визначення цільової аудиторії продукту. Перед формуванням користувацьких історій необхідно зрозуміти, для кого створюється продукт і які проблеми він має вирішити. Для цього потрібно визначити:

- хто буде використовувати продукт (тип користувача);
- які проблеми є у цих користувачів;
- які завдання користувачі хочуть виконати за допомогою продукту.

Крок 2: Формулювання користувацьких історій:

- визначте тип користувача (наприклад, «звичайний користувач», «адміністратор», «новий клієнт»);
- визначте функціонал, який цей користувач потребує;
- опишіть мету або причину, чому цей функціонал важливий для користувача.

Крок 3: Пріоритезація історій. Користувацькі історії повинні бути розподілені за пріоритетом:

- високий пріоритет – критично необхідні функції;
- середній пріоритет – важливі функції, але не критичні;

– низький пріоритет – додатковий функціонал, який можна реалізувати пізніше.

Історія користувача	Пріоритет
Реєстрація та авторизація	Високий
Додавання витрат	Високий
Перегляд графіків витрат	Середній
Темна тема інтерфейсу	Низький

Крок 4: Оцінка історій (Story Points)

Story Points – це умовна одиниця для оцінки складності виконання завдання. Зазвичай використовують шкалу від 1 до 8 або інші числові ряди (наприклад, числа Фібоначчі):

- 1 – дуже легке завдання;
- 2 – легке завдання;
- 3 – середнє завдання;
- 5 – складне завдання;
- 8 – дуже складне завдання.

Крок 5: Приклад повної таблиці беклогу продукту

Приклад таблиці

№	Історія користувача	Пріоритет	Story Points
1	Як користувач, я хочу зареєструватися через Google, щоб швидко отримати доступ до додатку	Високий	2
2	Як користувач, я хочу додавати витрати вручну, щоб контролювати свої фінанси	Високий	3

3	<i>Як користувач, я хочу автоматично синхронізувати витрати з банківської карти, щоб не вводити їх вручну</i>	<i>Високий</i>	<i>5</i>
4	<i>Як користувач, я хочу переглядати аналітику витрат у вигляді графіків, щоб краще розуміти свої фінансові звички</i>	<i>Середній</i>	<i>5</i>
5	<i>Як користувач, я хочу отримувати нагадування про важливі платежі, щоб не забувати оплачувати рахунки</i>	<i>Середній</i>	<i>3</i>

ЛАБОРАТОРНА РОБОТА 2

Тема: Розбиття беклогу на спринти, формування спринтів.

Мета роботи:

- навчитися розподіляти завдання з беклогу продукту на спринти;
- освоїти процес формування беклогу спринту та оцінювання завдань⁴
- ознайомитися з принципами планування спринту у Scrum;
- закріпити навички пріоритезації завдань для ефективного управління розробкою.

ЗАВДАННЯ

Для успішного виконання лабораторної роботи необхідно виконати наступні завдання:

- ознайомитися з основами планування спринтів у Scrum;
- взяти сформований беклог продукту (мінімум 10 користувацьких історій) або створити новий;
- розподілити історії користувачів за пріоритетами (високий, середній, низький);
- оцінити кожне завдання за складністю у Story Points;
- визначити тривалість спринту (наприклад, 1 або 2 тижні);
- розподілити завдання між спринтами так, щоб кожен спринт мав реалістичний обсяг роботи;
- визначити очікувані результати (інкременти) після кожного спринту;
- описати, як проходитиме планування спринту та Sprint Review;
- оформити результати роботи у вигляді звіту.

ПОРЯДОК ВИКОНАННЯ РОБОТИ

Крок 1: Вивчення теоретичного матеріалу. Перед початком роботи необхідно ознайомитися з такими поняттями:

- спринт (Sprint) – основний робочий цикл у Scrum, протягом якого команда реалізує частину функціоналу продукту.
- беклог продукту (Product Backlog) – список усіх завдань, які потрібно виконати для створення продукту.
- беклог спринту (Sprint Backlog) – підмножина беклогу продукту, яку команда реалізує у конкретному спринті.
- story Points – оцінка складності виконання користувацької історії.
- планування спринту (Sprint Planning) – зустріч команди для вибору завдань на спринт.
- огляд спринту (Sprint Review) – демонстрація результатів після завершення спринту.

Крок 2: Вибір продукту та беклогу. Здобувач використовує беклог продукту, створений в рамках виконання першої лабораторної роботи.

Крок 3: Визначення тривалості спринту. Зазвичай у Scrum використовуються 1-2 тижневі спринти. Здобувач вибирає тривалість (наприклад, 2 тижні).

Крок 4: Формування спринтів. Завдання з беклогу продукту розподіляються між спринтами так, щоб їх можна було реалізувати за відведений час.

Приклад розподілу беклогу по 3 спринтах:

<i>Спринт</i>	<i>Завдання (історії користувачів)</i>	<i>Story Points</i>
<i>Спринт 1</i>	<i>Реєстрація через Google, введення витрат вручну, автоматична синхронізація витрат</i>	<i>10</i>

<i>Спринт 2</i>	<i>Аналітика витрат, нагадування про платежі</i>	<i>8</i>
<i>Спринт 3</i>	<i>Темна тема, додаткові налаштування користувача</i>	<i>5</i>

Крок 5: Планування спринту (Sprint Planning). Для кожного спринту здобувач:

- обирає пріоритетні завдання;
- визначає відповідальних за виконання завдань (якщо працює індивідуально
- умовно призначає ролі);
- оцінює потенційні ризики та способи їх уникнення.

Приклад:

- *ризик: складність інтеграції з Google-акаунтом.*
- *рішення: використовувати Firebase Authentication, яка спрощує інтеграцію.*

Крок 6: Огляд спринту (Sprint Review). Здобувач описує, як відбуватиметься демонстрація результатів після завершення кожного спринту.

- показ готових функцій;
- отримання зворотного зв'язку;
- коригування беклогу продукту (додавання або зміна пріоритетів завдань).

Крок 7: Оформлення звіту. Звіт повинен містити:

- тему та мету роботи.
- опис сформованих беклогів спринтів;
- опис планування спринтів;
- опис процесу огляду спринту;
- висновки щодо процесу формування спринтів.

ОГЛЯД СПРИНТУ (SPRINT REVIEW)

Що таке огляд спринту (Sprint Review)?

Огляд спринту – це подія в Scrum, яка проводиться наприкінці кожного спринту. Її основна мета – продемонструвати команді та зацікавленим сторонам (Product Owner, користувачам, менеджерам) результати виконаної роботи, обговорити отриманий зворотний зв'язок і скоригувати подальший напрямок розробки.

Основні цілі огляду спринту

Основні цілі даного заходу включають:

- оцінка виконаної роботи – перевірка, чи всі заплановані завдання були завершені;
- демонстрація готового функціоналу – показ інкременту (нових можливостей продукту, що були реалізовані у спринті);
- обговорення труднощів та отриманого досвіду – виявлення проблем, які виникли під час роботи;
- отримання зворотного зв'язку від Product Owner та зацікавлених сторін – аналіз відповідності очікуванням;
- оновлення беклогу продукту – внесення змін або додавання нових завдань на основі отриманого фідбеку.

Хто бере участь в огляді спринту?

Перелік учасників спринту:

- Scrum-майстер – модерує зустріч, забезпечує ефективний процес обговорення;

- Product Owner – переглядає виконані завдання, надає зворотний зв'язок, вирішує, чи відповідають зміни очікуванням;
- команда розробників – демонструє функціонал, розповідає про труднощі, отримує фідбек;
- користувачі або представники зацікавлених сторін (опціонально) – оцінюють реалізовані функції, надають побажання.

Порядок проведення огляду спринту

Scrum-майстер відкриває зустріч і коротко підсумовує спринт:

- тривалість спринту (наприклад, 2 тижні);
- основні завдання, які планували виконати;
- загальний результат роботи.

Команда демонструє готовий функціонал (інкремент):

- розробники демонструють нові можливості продукту, що були реалізовані;
- кожен учасник команди може розповісти про свою частину роботи;
- використовується живий показ продукту або екранна демонстрація.

Приклад:

- якщо розробляється мобільний додаток для управління фінансами, демонструється, як працює функція автоматичної категоризації витрат;
- якщо це платформа для бронювання місць у коворкінгах, показується, як працює пошук і бронювання місця.

Обговорення труднощів, які виникли під час спринту

Перелік можливих труднощів:

- чи були якісь технічні складності?

- що зайняло більше часу, ніж очікувалося?
- які уроки команда винесла з цього спринту?

Отримання зворотного зв'язку від Product Owner та зацікавлених сторін

Зворотній зв'язок може включати:

- оцінку реалізованих функцій.
- чи відповідає інкремент бізнес-очікуванням?
- чи потрібно щось покращити або змінити в наступних спринтах?

Оновлення беклогу продукту

Оновлення беклогу може включати:

- якщо Product Owner або користувачі запропонували нові функції, вони додаються до беклогу;
- якщо щось не було реалізовано в поточному спринті, це переноситься у наступний спринт або коригується;
- коригуються пріоритети завдань.

Підведення підсумків зустрічі

Перелік дій із підведення підсумків:

- Scrum-майстер підбиває підсумки обговорення;
- визначаються ключові напрямки покращень для наступного спринту;
- встановлюється дата наступного спринту.

ПРИКЛАД ПРОВЕДЕННЯ ОГЛЯДУ СПРИНТУ

Ситуація: Завершено перший спринт у розробці мобільного застосунку для управління фінансами.

Вхідні дані спринту

Тривалість: 2 тижні.

Завдання:

- реєстрація користувача через Google (3 Story Points);
- введення доходів/витрат вручну (5 Story Points);
- автоматична категоризація витрат (8 Story Points).

Проведення Sprint Review

Scrum-майстер відкриває зустріч: «Ми завершили перший спринт, у якому реалізували три основні функції: реєстрацію, введення витрат і автоматичну категоризацію».

Команда демонструє інкремент:

- розробник 1: «Ось як працює реєстрація через Google – користувач може швидко створити акаунт»;
- розробник 2: «Я реалізував введення витрат вручну, користувач може вказати категорію та суму»;
- розробник 3: «Автоматична категоризація працює на основі ключових слів у витратах».

Обговорення труднощів:

- виникла проблема із підключенням до Google API, що зайняло більше часу;

- автоматична категоризація потребує покращення алгоритму.

Зворотний зв'язок від Product Owner:

- реєстрація працює добре;
- введення витрат зручне, але потрібно додати можливість редагування;
- автоматична категоризація працює, але потребує доповнення.

Оновлення беклогу:

- додається завдання «Редагування введених витрат»;
- додається завдання «Поліпшення алгоритму автоматичної категоризації»;
- завдання «Додавання графіків витрат» переноситься у наступний спринт.

Підведення підсумків:

- реєстрація та введення витрат готові до використання;
- наступний спринт буде присвячений покращенню автоматичної категоризації та реалізації візуалізації витрат.

ВАРІАНТИ ДОДАТКІВ

Варіант – остання цифра дня народження студента (цифра 0 – 10 варіант).

1. Мобільний застосунок для управління особистими фінансами

Опис: Додаток для відстеження доходів і витрат, аналізу фінансових звичок і планування бюджету.

Основні функції:

- реєстрація та авторизація;
- введення доходів/витрат вручну;
- автоматична категоризація витрат;
- побудова графіків витрат;
- інтеграція з банківськими картами;
- створення фінансових цілей.

2. Платформа для бронювання робочих місць у коворкінгах.

Опис: Система для пошуку та бронювання робочих місць у коворкінгах з можливістю онлайн-оплати.

Основні функції:

- пошук коворкінгів за містом і ціною;
- бронювання місця на певний час;
- оплата онлайн;
- відгуки та рейтинг коворкінгів;
- управління бронюваннями.

3. Додаток для планування здорового харчування.

Опис: Програма для підбору здорового харчування на основі уподобань користувача та його фізичних показників.

Основні функції:

- заповнення профілю (вага, ріст, алергії, цілі);
- генерація меню на день/тиждень;
- розрахунок калорійності та БЖУ;
- список покупок для вибраного меню;
- нагадування про прийом їжі.

4. Застосунок для онлайн-навчання та саморозвитку.

Опис: Платформа для проходження курсів, перегляду відеоуроків та проходження тестів.

Основні функції:

- каталог курсів;
- відео- та текстові матеріали;
- тестування після кожного модуля;
- сертифікація після завершення курсу;
- відстеження прогресу.

5. Чат-бот для організації завдань у команді.

Опис: Чат-бот для Telegram або Slack, який допомагає організувати завдання у команді та нагадує про дедлайни.

Основні функції:

- створення завдань через текстові команди;
- встановлення дедлайнів та нагадувань;
- призначення відповідальних;
- інтеграція з календарем;
- автоматичні нагадування.

6. Мобільний додаток для швидкого пошуку репетиторів.

Опис: Сервіс для пошуку репетиторів за предметами, рівнем підготовки та ціною.

Основні функції:

- фільтр репетиторів за містом, ціною, рейтингом;
- запис на заняття онлайн;
- чат із викладачем;
- відеозв'язок для онлайн-уроків;
- відгуки про репетиторів.

7. Програма для контролю розкладу занять студентів.

Опис: Мобільний застосунок, що автоматично формує розклад занять та нагадує про них.

Основні функції:

- введення розкладу вручну або завантаження з сервера;
- нагадування про початок занять;
- інтеграція з Google Calendar;
- оцінки та домашні завдання;
- чат із викладачами.

8. Застосунок для аналізу продуктивності спортсменів.

Опис: Додаток для ведення щоденника тренувань із можливістю відстеження прогресу.

Основні функції:

- створення персонального графіка тренувань;
- відстеження фізичних показників;
- аналіз продуктивності та прогресу;
- рекомендації щодо навантажень;
- синхронізація з фітнес-браслетами.
-

9. Платформа для спільних покупок у супермаркетах

Опис: Сервіс для об'єднання людей у групи для спільних покупок та зменшення витрат.

Основні функції:

- створення групових замовлень;
- спільне формування списку покупок;
- розподіл витрат між учасниками;
- відстеження статусу замовлення;
- онлайн-оплата.

10. Додаток для пошуку попутників для подорожей

Опис: Платформа для пошуку попутників для спільних поїздок, щоб зменшити витрати.

Основні функції:

- введення маршруту та дат поїздки;
- пошук інших користувачів, що їдуть у тому ж напрямку;
- чат між попутниками;
- розподіл витрат на поїздку;
- відгуки після поїздки.