## SDLC та STLC. Методології розробки ПЗ.

## Beet Seed — відпрацюй навички на базовому рівні.

Склади порівняльну таблицю найбільш поширених методологій

а. Надаючи відповіді, обґрунтуй свою думку: чому це саме сильна/слабка сторона/доцільна галузь застосування.

б. Відповіді запиши в той самий файл Google Docs. Додай посилання на нього в LMS.

| № | Назва методології | Сильні сторони | Слабкі сторони | Для якої галузі є доцільною |
| --- | --- | --- | --- | --- |
| 1 | Waterfall | Є чітки вимоги/чітке бачення того, як має виглядати готовий програмний продукт ще до початку створення - тож можна легко визначити на якому етапі розробки ми знаходимось зараз та чи не було відхилення від початкової цілі | Немає гнучкості в процесі розробки - продукт не має змінюватись під впливом зовнішніх чинників - немає адаптації під актуальні на сьогодні потреби бізнеса. Як результат - велики фінансові ризики | Сфери, що не вимагають частих оновлень: держоргани, машинобудівництво, медицина. Проекти із часовими та фінансовими обмеженнями. |
| 2 | Scrum | Велика задача ділиться на дрібні, тож внесення змін набагато простіше, ніж при “каскаді”. Постійний контакт з клієнтом, відбуваються уточнення щодо актуальності.  Як результат - фінансові ризики мінімізуються | Вимагає окремого спеціаліста - скрам-майстра.  Постійний контакт з замовником може тормозити процес розробки. | Сфери що вимагають частих оновлень: telecom, e-commerce |
| 3 | Kanban | Чітке визначення статусів задач (TODO, In Progress, Completed etc.)  Візуалізація задач, їх статусів, термінів  Добре підходить для опису короткострокових цілей | Не підходить для великих команд через громіздкий “візуал” Не підходить для планування задач на довгий строк або майбутніх задач | Підходить для усіх сфер, задачі яких можна розділяти за статусами виконання цих задач |
| 4 | V-модель (V-model) | Чітке визначення етапів розробки продукта.  Планування тестування та верифікація системи проводяться на ранніх етапах.  Кілька етапів тестування в процесі розробки продукта. | Немає гнучкості в процесі розробки, як і в Waterfall | В тих самих, що і в Waterfall, але коли необхідне більш ширше тестове покриття |
| 5 | Extreme programming (XP) | Висока швидкість розробки.  Висока якість продукта.  Мінімальні витрати ресурсів на доопрацювання системи завдяки частим релізам, повноцінному тестуванню. | Постійний контакт з замовником може тормозити процес розробки.  Немає чіткості щодо витрат та часу на виконання задач, оскільки спочатку немає чіткого бачення продукту. | Тільки в галузі розробки програмного забезпечення. |

## Beet Sprout — детальніше заглибся в практику.

1. Виконай завдання попереднього рівня.

2. Напиши розгорнуті відповіді (0,5 - 1 сторінки тексту) на такі два питання:

**Чому з’явився Agile-маніфест?**

Аби привертати увагу клієнтів до себе, бізнес (його послуги/продукти) повинен бути актуальним для клієнта - а отже, вміти швидко адаптуватися під різні ситуації, що виникають через зовнішні фактори.

Agile-маніфест з’явився через необхідність знайти альтернативу негнучким методологіям розробки, що передбачали велику кількість документації, строгі строки та плани щодо готового продукту. Внести зміни в готовий продукт або в продукт на етапі розробки було або неможливо, або дуже затратно та складно - тож, проблеми виникали як у замовника, так і у розробника продукта. Також, ці методології не передбачали часті перевірки/відгуки від замовника.

Таким чином, для команди розробки існує ризик, що проект не буде вирішувати проблеми клієнта. Або створенний продукт міг виявитися зовсім не тим, чого очікував замовник.

**Які проблеми він мав вирішити і чи це вдалося?**

Agile мав вирішити проблеми як розробників, так і бізнеса (замовника). Головною метою стала коммунікація між розробником та клієнтом, від якого очікується включення в процес розробки - затвердження та оцінка того, що було зроблено за певний час, постановка нових цілей і тд.

Як вказали самі розробники маніфесту:

* ***Люди та співпраця*** *важливіші за процеси та інструменти*
* ***Працюючий продукт*** *важливіший за вичерпну документацію*
* ***Співпраця із замовником*** *важливіша за обговорення умов контракту*
* ***Готовність до змін*** *важливіша за дотримання плану*

*“Тобто, хоча цінності, що справа, важливі, ми все ж цінуємо більше те, що зліва.”*

## Mighty Beet — різнобічно опануй тематику уроку.

1. Виконай завдання двох попередніх рівнів.

2. Ти – засновник/ця стартапу і плануєш випустити на ринок мобільний застосунок для обміну світлинами котиків.

Яку методологію ти обереш для процесу розробки і чому? Відповідь текстово обґрунтуй.

Відповіді запиши в той самий файл Google Docs.

Цільова аудиторія - люди різного віку та різного досвіду користування мобільними застосунками.

Я би обрала гнучку методологію розробки - на даний момент у мене немає чіткого бачення того, як продукт повинен виглядати в кінці розробки - тож, постійно будуть вноситися різні оновлення.

Наприклад, у 2024 відбудеться Чемпіонат Європи з футболу - до цієї події можна підготувати тематичні оновлення для застосунку. Або додавання нового функціоналу для пошуку/фільтрації у разі додавання в систему нової породи котиків.

Scrum:

Розробка буде вестись поетапно ітеративно невеликими спрінтами по 3 тижні.

Також можливо використання канбан-дошки для фіксації статусів різних задач команди.