**Домашнє завдання #3**

**Перший рівень:**

| **Назва** | **Сильні сторони** | **Слабкі сторони** | **Для якої галузі є доцільною** |
| --- | --- | --- | --- |
| **Scrum** | 1. Контроль часових рамок процесу розробки. Тому що завдяки тайм - менеджменту ризик виходу за часові рамки проекту мінімальний.  2. Ретроспектива. Тому що це можливість обговорити як проходили процеси і що можна покращити на майбутнє.  3. Daily meeting - бо це можливість зрозуміти як проходять процес і чи не виникає ні у кого ніяких проблем.  4. Отримання робочого функціоналу частинами - корисно оскільки можна заздалегідь оцінити продукт. | 1. Можливий нерівномірний розподіл часу в спринті. Це коли наприклад: щоб тестувальник мав що тестувати розробнику спочатку потрібно це створити  2.Краще для невеликих команд(важче маштабується на великі проекти). Оскільки якщо команда велика буде потрачено багато часу на комунікацію.  3. Довжину спринту не можна змінити, а тому додавати завдання, які можуть виникнути в ході роботи під час спринту важче.  4. Затрачається час на планування спринту, а також робоче навантаження повинно бути добре сплановано | 1. Для стартапів, оскільки невелика команда краще кооперується та комунікує. 2. Для розробки ПЗ де є чітка архітектура проекту.  3. Є доцільною в галузі освіти. |
| **Kanban** | 1. Команда зосереджується на завданнях, а не часових рамках, оскільки не завжди завдання можна поділити на рівномірні частини  2. Підвищення ефективності роботи. Оскільки співробітники не чекають, поки їм дадуть роботу, а відразу по завершенню завдання беруть інше, немає простою.  3. Скорочення часу на обговорення та наради. Оскільки члени команди мають безперервний потік завдань, менше часу витрачається на планування. | 1. Якщо виникає затримка якогось процесу потрібно розбиратись чому так сталося на якому етапі і що потрібно оптимізувати  2. Оскільки немає часових обмежень можуть бути проблеми із дедлайнами проектів. | Для тривалих проектів (не дуже обмежених в часі) |

**Другий рівень:**

У зв'язку із швидким розвитком технологій для того щоб продукт до завершення його розробки був валідним виникає необхідність зміни вимог і внесення правок у проект у процесі його розробки. На відміну від традиційних підходів де все відбувається значно повільніше, а зворотний зв’язок ігнорується до повного закінчення проекту, методики Agile швидко набули популярності. За Agile усе відбувається значно швидше: представлення прототипу, тестування, доставка самого програмного продукту на ринок з адаптацією до його мінливих потреб. Така філософія управління проектом, що грунтується важливості комунікації з клієнтом є запорукою успіху проекту, адже у комунікації народжується істина і можливість краще зрозуміти потреби клієнта. А комунікація в команді дає можливість покращення і налагодження внутрішніх процесів. Коли у клієнта є можливість перегляду частинами працюючого продукту – це основний показник прогресу.

**Третій рівень:**

Оскільки це мобільний застосунок, а не якась критична область як от медицина чи космонавтика то я можу використати одну з гнучких методологій. Я б обрала Scrum методологію, оскільки вона мені подобається своїм тайм-менеджментом, тобто можливістю встановлення часових рамок для розробки певних функцій, а також можливістю перегляду розроблюваного функціоналу в процесі розробки і можливістю внесення правок на стадії розробки.