**Домашня робота до уроку № 3**

**Завдання 1**

**Склади порівняльну таблицю найбільш поширених методологій:**

| **№** | **Назва методології** | **Сильні сторони** | **Слабкі сторони** | **Для якої галузі є доцільною** |
| --- | --- | --- | --- | --- |
| **1** | **Waterfall** |  |  |  |
| **2.** |  |  |  |  |

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

| **№** | **Назва методології** | **Сильні сторони** | **Слабкі**  **сторони** | **До якої галузі є доцільною** |
| --- | --- | --- | --- | --- |
| 1 | Каскадна модель | * документування кожного етапу; * для замовника всі процеси прозорі; * чітке планування термінів виконання та затрат. | * затвердження повного пакету вимог ще на початковому етапі; * досить суворі правила; * якщо потрібно буде вносити зміни ще в початкові етапи, то це стане великою проблемою, адже переробляти доведеться практично з самого початку; * при внесенні змін у вимоги - збільшення витрат часу та бюджету. | для проектів з чітко визначеними вимогами, де внесення змін не передбачено у процесі розробки  (авіаконструювання, космонавтика, медицина). |
| 2 | V-модель | * чітка сувора етапізація; * планування тестування і верифікація проходить на ранніх етапах; * проміжне тестування. | * недостатній аналіз ризиків; * не передбачає роботу з паралельними подіями; * недостатня гнучкість моделі. | застосовується у проектах де існують часові та фінансові обмеження;  для завдань, які передбачають більш широке тестове покриття , ніж у каскадній моделі; наприклад - для створення систем управління базами даних,медичних пристроїв, системи контролю доступу. |
| 3 | Ітеративна модель | * раннє створення працюючого продукту; * можливість внесення змін вимог на будь-якому етапі розробки; * на кожній ітерації - простіше застосувати тестування та аналіз ризиків, ніж для всього життєвого циклу продукту. | * окремі ітерації не накладаються одна на другу; * через відсутність всіх вимог до початку тестування - можуть виникати проблеми з реалізацією загальної архітектури системи; * потребує багато часу, тому можна не вкластися у визначений бюджет. | застосовують у великих проектах, коли відомі хоча б основні вимоги, але при цьому вимоги можуть змінюватися в процесі розробки (наприклад стартапи, мобільні додатки, для створення продукту, що задовольняє конкретні потреби). |

**Завдання 2**

**2.1 Виконай завдання попереднього рівня.**

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

* На твою думку, чому з’явився Agile-маніфест?
* Які проблеми він мав вирішити і чи це вдалося.

Agile-маніфест - це документ, що описує основні принципи, на яких базується [гнучка розробка](https://uk.wikipedia.org/wiki/%D0%93%D0%BD%D1%83%D1%87%D0%BA%D0%B0_%D1%80%D0%BE%D0%B7%D1%80%D0%BE%D0%B1%D0%BA%D0%B0).

На мою думку, Agile-маніфест з’явився по мірі того, як почала активно розвиватися галузь IT, грубо кажучи коли про неї почула основна маса людей, чим далі розвивалась ця галузь стало зрозуміло, що це супер, коли людина має прекрасну базу знань, але для активної швидкої та результативної співпраці і не тільки всередині колективу , а й із замовником потрібна досить добра гнучка комунікація. Адже, це своєрідна тактика, завдяки якій проекти будуть виконуватись швидко і водночас гнучко, при цьому на всіх етапах є гарна комунікація, своєрідний контроль моніторинг і допомога, задля отримання бажаного результату.

Ключовий принцип Agile — це розробка шляхом коротких ітерацій (циклів), в кінці кожного з яких замовник (користувач) отримує робочий код або продукт.

Наводжу цінності, так звані основні принципи, виділені в Agile-маніфесті:

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

Як на мене проблему котру має вирішити Agile-маніфест - це побудувати злагоджену роботу з чудовою комунікацією з правильно підібраним персоналом, задля якісного результату, де кожен працівник розуміє свою роль, значимість, всі зусилля націлені на результат, можливість внесення змін до проекту в процесі роботи, відходячи від початкових вимог за бажанням та потребою замовника. І як результат - задоволений клієнт та команда. І на мою думку - Agist-маніфест вирішує поставлені проблеми та цілі, які задані в в його методології.

На основі виділених цінностей, хочу викласти своє розуміння. Головна ціль безумовно гарний якісний продукт та задоволений замовник, але при цьому безмежну цінність має комунікація між людьми, нормальні взаємовідносини, дослухання до думки іншого, аналіз цих думок і відкритість та готовність до змін. Грубо кажучи можливість відійти від правил і встановлених рамок (маю на увазі перший та другий пункт: люди та співпраця важливіші за процеси та інструменти; працюючий продукт важливіший за вичерпну документацію) для того, щоб мати задоволеного клієнта готовим продуктом, і в прекрасній комунікації, що створить передумови для подальшої співпраці.

**Завдання 3**

**3.1 Виконай завдання двох попередніх рівнів.**

**3.2 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.**

Так , як я стартапер, то це значить я повна енергії і в моїй голові кишить безліч ідей, котрі хочеться реалізувати. А це означає, що в процесі розробки мої вимоги поставлені на початку можуть змінюватися та доповнюватися. Тож такі суворі по початковим вимогам методології розробки, як каскадна та й в принципі v-модель мені однозначно не підійдуть. Я б використала ітеративну та інкрементну модель, саме по принципу можливості внесення змін по вимогам та удосконалення після кожної ітерації. Ітеративна розробка дає можливість задати вектор розвитку та зменшити витрати завдяки валідації ідей та функцій в продукті. А якщо брати інкрементну модель то вона дозволяє збирати фідбек в момент розробки і паралельно вдосконалювати продукт, витрачаючи при цьому набагато менше коштів, ніж потрібно для виконання повторної розробки.