



1. Порівняльна таблиця поширених методологій.

№	Назва методології	Сильні сторони	Слабкі сторони	До якої галузі є доцільною
1.	Waterfall	<ul style="list-style-type: none">- чітко сплановані терміни і витрати;- усі процеси задокументовані;- замовник не вносить змін у процесі.	<ul style="list-style-type: none">- великий обсяг роботи на першому етапі (затвердженн я усіх вимог);- при внесенні змін у процесі треба заново перероблюват и кожен етап;- збільшення витрат при необхідності змін.	<ul style="list-style-type: none">- для невеликих компаній, бо вартість обслуговува ння невисока;- для військових чи медичних потреб, бо наявні фіксовані вимоги, якість та надійність на кожному етапі.
2.	V-модель	<ul style="list-style-type: none">- чітке планування;- верифікація на ранніх етапах розробки;- кожні отримані дані тестуються.	<ul style="list-style-type: none">- не працює з паралельним и подіями;- тестування відбувається пізно, що впливає на графіки;- недостатній аналіз ризиків.	<ul style="list-style-type: none">- проекти, у яких є часові і фінансові обмеження;
3.	Ітераційна	<ul style="list-style-type: none">- раннє створення працюючого ПЗ;- гнучкість;- простіше забезпечення тестування та аналізу ризиків.	<ul style="list-style-type: none">- кожен етап - відокремлени й один від одного;- можуть бути проблеми з загальною архітектурою системи, бо не всі вимоги відомі завчасно.	<ul style="list-style-type: none">- проекти, що постійно розвивають ся, не мають остаточного набору вимог і термінів.

4.	Спіральна	<ul style="list-style-type: none"> - чіткий аналіз ризиків; - хороша документація процесу розробки; - гнучкість; - раннє створення робочих прототипів. 	<ul style="list-style-type: none"> - дорога у використанні; - не підходить для невеликих проєктів. 	<ul style="list-style-type: none"> - великі довгострокові проєкти, з відсутністю чітких вимог; - при розробці нової лінійки вже існуючих продуктів.
5.	Scrum	<ul style="list-style-type: none"> - прозорий процес роботи; - гнучкість; - легко контролювати. 	<ul style="list-style-type: none"> - немає довгострокового плану дій; - часто недостатньо опрацьована документація ; - постійне залучення замовника. 	<ul style="list-style-type: none"> - проєкти, що ростуть, масштабуються; - мобільні додатки, сайти тощо.
6.	Kanban	<ul style="list-style-type: none"> - гнучкість; - візуалізація процесів; - зворотний зв'язок; - прозорі правила роботи. 	<ul style="list-style-type: none"> - не підходить для великих робочих груп; - націлена на короткострокові проєкти. 	<ul style="list-style-type: none"> - проєкти, що не мають чітких вимог, часових обмежень; сайти, веб-застосунки.



2. Чому з'явився Agile-маніфест?

Agile - це і методологія, і набір практик, і зведення правил. Але на вищому абстрактному рівні Agile - це підхід, який має підвищити ефективність певних робочих процесів.

Agile методологія базується на таких принципах:

- Працюючий стан продукту необхідно забезпечувати, починаючи з ранніх стадій;
- Вимоги можуть активно змінюватись на будь-якому етапі;
- Проєкт має будуватися навколо зацікавлених людей;
- Продукт має бути максимально актуальним та ефективно вирішувати поставлені завдання;

- Необхідно дотримуватися балансу між простими рішеннями та технічною досконалістю;
- Взаємодія із замовником або його представником має відбуватися на регулярній основі.

Виконання рекомендацій, що описані у Agile-маніфесті, має забезпечити проєкту можливість постійного тестування та трансформації.

Таким чином гнучкий підхід до розробки став популярним на ринку ІТ. Висока конкуренція та необхідність швидко видавати працюючий продукт поставили під сумнів методології, які були актуальними раніше. За своєю сутністю підхід Agile є протилежністю до моделі Waterfall. Адже коли бізнесу стали важливими такі аспекти, як швидкість одержання продукту та можливість його швидкої адаптації під поточні потреби ринку, негнучка методика Waterfall стала неактуальною.

Підхід до роботи, заснований на філософії Agile, сьогодні є базовим для ІТ-галузі і не тільки. Розуміння гнучкої методології розробки продукту є важливим для ефективної діяльності в ІТ-сфері та суміжних галузях.



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.

Для розробки такого мобільного застосунку я б обрала методологію Scrum.

На мою думку Scrum є найзручнішим для розробки мобільних застосунків, адже процес відбувається прозоро, можна контролювати кожен етап, додавати певні нові функції. До того ж, можливо, у майбутньому ми захочемо масштабуватись і додати до застосунку ще й песиків, у чому допоможе вже готова невелика Development team. Теж великим плюсом будуть постійні мітинги, на яких будемо продуктивно і досить весело проводити час.