## Лабораторна робота №8. Порівняння методологій розробки програмного забезпечення

**Мета роботи**: засвоєння навичок вибору методології на розробку програмного забезпечення

**Завдання:**

1. Проаналізувати різні методології розробки програмного забезпечення.
2. За результатами аналізу складіть порівняльну таблицю для двох різних за типом методологій.
3. В таблиці для кожної методології повинні бути визначені позиції: суть, дата створення, розробники, принципи застосування, плюси, мінуси, компанії-практики, за яких умов підійде вам,, з яких причин не підійде.

**КОНТРОЛЬНІ ПИТАННЯ:**

1. За якими принципами можна класифікувати методології розробки ПЗ?

2. Чому каскадна модель відноситься до прогнозованих методологій?

3. Охарактеризуйте методологію RUP?

4. Які основні ідеї і принципи "гнучких" методологій?

5. Де краще застосовувати "гнучкі", а де прогнозовані методології?

6. В чому сутність методології RAD?

7. Визначте основні принципи екстремального програмування?

8. Що таке програмування «знизу вгору»?

ТЕОРЕТИЧНА ЧАСТИНА

## Порівняльна таблиця методологій

|  |  |  |
| --- | --- | --- |
|  | **Agile** | **Waterfall** |
| **Суть** | Гнучка модель розробки, заснована на ітеративних принципах | Каскадна система розробки, заснована на жорсткій послідовності процесу розробки |
| **Дата створення** | 2001 г. | 1956 року, 1961 р 1970 г. |
| **Розробники** | Група IT-фахівців (США) | Г. Беннінгтон, Хозьер, В. Уолкер Ройс |
| **Принципи застосування** | * найвищий пріоритет в задоволенні потреб замовника * протягом всього проекту команда і замовник щодня взаємодіють між собою і один з одним * робочий продукт — головний показник прогресу * роботу можна довірити тільки самоорганізованій, мотивованій команді * оптимальні терміни випуску робочого продукту — від 2 тижнів до 2 місяців. | * жорстка послідовність етапів розробки * перехід до нового етапу — тільки після успішного завершення попереднього * фіксована вартість продукту * замовник не залучається до безпосереднього процесу розробки * зміни можуть бути внесені тільки після завершення всього процесу розробки. |
| **Плюси** | 1. високий рівень взаємодії між членами команди проекту 2. швидкий результат (робочий код) в результаті «спринтів» 3. стимулювання змін і поліпшень продукту під час його розробки 4. безпосереднє залучення замовника до робочого процесу. | 1. зрозуміла і чітка схема робочого процесу 2. можливість підрахунку точної кількості витрачених на проект ресурсів 3. не вимагає витрат по налагодженню комунікацій між усіма членами команди. |
| **Мінуси** | * ризик нескінченних змін продукту * велика залежність від рівня кваліфікації та досвіду команди * практично неможливо точно підрахувати підсумкову вартість проекту. | * пріоритет формального підходу до послідовності процесу роботи * неможливість внесення змін замовником до закінчення розробки продукту * в разі нестачі ресурсів страждає якість проекту через скорочення етапу тестування. |
| **Компанії-практики** | Unilever, ряд банків (Альфа Банк, Home Credit, Райффайзен Банк і т.д.) | Cisco Ericsson AB, Toyota (до 2010) |
| **Підійде вам, за умов** | 1. над проектом працює досвідчена, висококваліфікована команда 2. ви працюєте над стартапом 3. потрібно швидко отримати робочу версію продукту 4. замовник виступає в якості партнера, а не інвестора 5. продукт розробляється в сфері, схильній до постійних змін. | 1. більша частина або вся робота над проектом проводиться на аутсорсі 2. у вас є чітка концепція продукту, який хочете отримати 3. ви не обмежені в часі і ресурсах створення продукту 4. створення продукту або бізнесу побудовано на дотриманні суворої послідовності виконання задач. |
| **Не підійде з причин** | * ви не готові витрачати додаткові ресурси на налагодження щоденної стабільної комунікації між усіма учасниками процесу * продукт повинен бути створений до конкретного терміну * бюджет проекту строго обмежений * вам потрібна детальна документація по всім процесам розробки. | * ви хочете створити інноваційний продукт або великий проект * ви не впевнені в концепції пропонованого проекту * фінансові ресурси не є ключовим обмежувачем в вашому проекті. |

Таблиця створена за матеріалами статті "Agile чи Waterfall — який варіант відповідає вашому бізнесу?" - URL: https://worksection.com/ua/blog/waterfall-vs-agile.html