**Лабораторная работа №4**

Тема: Выбор модели, метода и подхода разработки программы.

Цель работы:

1. Изучить основные стратегии разработки и модели жизненного цикла.

2. Научиться осуществлять выбор стратегии и модели жизненного цикла для разработки конкретного проекта и обосновывать свой выбор.

Контрольные вопросы:

1. Кодирование – устранение ошибок:

- Особенности: Эта стратегия заключается в том, чтобы начать с написания кода и затем постепенно исправлять ошибки до достижения желаемого уровня функциональности и качества. Это часто используется в ситуациях, когда требования не определены достаточно четко и могут меняться в процессе разработки.

- Достоинства: Быстрое начало разработки, способствует гибкости в адаптации к изменяющимся требованиям.

- Недостатки: может привести к недостаточной структурированности кода и возможным трудностям в обнаружении и исправлении ошибок на поздних этапах.

- Рекомендуется для: Проектов, где требования нестабильны или подвержены частым изменениям.

2. Каскадная стратегия разработки программного обеспечения:

- Особенности: подразумевает последовательное выполнение фаз разработки: анализ, проектирование, реализация, тестирование и сопровождение.

- Достоинства: хорошо структурированная и планируемая модель, позволяет легко оценить прогресс и стоимость проекта.

- Недостатки: мало гибкая к изменениям требований, может привести к необходимости значительной переделки при возникновении изменений.

- Рекомендуется для: Проектов с жесткими и стабильными требованиями, где изменения неожиданны исключения.

3. Инкрементная стратегия разработки программного обеспечения:

- Особенности: Разработка в несколько итераций, при каждой из которых добавляется новый функционал или улучшения.

- Достоинства: Гибкость в адаптации к изменяющимся требованиям, клиенты получают результаты быстрее.

- Недостатки: могут возникнуть проблемы совместимости и интеграции при инкрементном добавлении функционала.

- Рекомендуется для: Проектов, где требования могут изменяться в процессе разработки, но основная цель достаточно стабильна.

4. Эволюционная стратегия разработки программного обеспечения:

- Особенности: Постепенное развитие продукта с привлечением пользователей и получением обратной связи.

- Достоинства: позволяет адаптировать продукт к реальным потребностям пользователей, легче принимать изменения.

- Недостатки: может требовать больше времени и ресурсов, нежели другие стратегии разработки.

- Рекомендуется для: Проектов, где требования не определены или субъективны, а также когда важна активная обратная связь от пользователей.

Модель жизненного цикла — это абстрактная концепция, описывающая последовательность этапов разработки программного продукта от начала до конца. Каждая модель имеет свои уникальные особенности и подходит для различных типов проектов и ситуаций.

Для разработки мобильного обучающего приложения «ОС Linux» следует выбрать стратегию разработки и модель жизненного цикла. Осуществляем выбор посредством составления таблиц:

Таблица 3 – Выбор модели жизненного цикла на основе характеристик требований

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № критерия | Критерии категории  требований | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| 1. | Являются ли требования к  проекту легко определимыми и  реализуемыми? | Да | Да | Да | Нет | Нет | Нет |
| 2. | Могут ли требования быть  сформулированы в начале ЖЦ? | Да | Да | Да | Да | Нет | Нет |
| 3. | Часто ли будут изменяться  требования на протяжении ЖЦ? | Нет | Нет | Нет | Нет | Да | Нет |
| 4. | Нужно ли демонстрировать  требования с целью их  определения? | Нет | Нет | Нет | Нет | Да | Да |
| 5. | Требуется ли проверка  концепции программного  средства или системы? | Нет | Да | Нет | Нет | Да | Да |
| 6. | Будут ли требования изменяться  или уточняться с ростом  сложности системы  (программного средства) в ЖЦ? | Нет | Нет | Нет | Да | Да | Нет |
| 7. | Нужно ли реализовать основные  требования на ранних этапах  разработки? | Нет | Да | Нет | Нет | Да | Да |

Вычисления: 5 за каскадную, 5 за V- образную, 5 за RAD, 2 за инкрементную, 5 за быстрого прототипирования и 2 за эволюционную.

Итог: На основе результатов заполнения табл. 3 подходящей является V- образная и каскадная модель и модель RAD.

Таблица 4 – Выбор модели жизненного цикла на основе характеристик команды разработчиков

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № критерия | Критерии категории команды  разработчиков  проекта | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| 1. | Являются ли проблемы  предметной области проекта  новыми для большинства  разработчиков? | Нет | Нет | Нет | Да | Да | Нет |
| 2. | Являются ли инструментальные  средства, используемые в  проекте, новыми для  большинства разработчиков? | Да | Нет | Нет | Нет | Да | Да |
| 3. | Изменяются ли роли участников  проекта на протяжении ЖЦ? | Нет | Нет | Нет | Да | Да | Да |
| 4. | Является ли структура процесса  разработки более значимой для  разработчиков, чем гибкость? | Нет | Да | Нет | Да | Нет | Нет |
| 5. | Важна ли легкость распределения  человеческих ресурсов проекта? | Да | Да | Да | Нет | Нет | Да |
| 6. | Приемлет ли команда  разработчиков оценки, проверки,  стадии разработки? | Да | Да | Нет | Да | Да | Нет |

Вычисления: 3 за каскадную, 3 за V- образную, 5 за RAD, 4 за инкрементную, 2 за быстрого прототипирования и 3 за эволюционную.

Итог: На основе результатов заполнения табл. 4 подходящей является RAD модель

Таблица 5 – Выбор модели жизненного цикла на основе характеристик

коллектива пользователей

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № критерия | Критерии категории коллектива  пользователей | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| 1. | Будет ли присутствие  пользователей ограничено в ЖЦ  разработки? | Да | Нет | Нет | Да | Нет | Нет |
| 2. | Будут ли пользователи оценивать  текущее состояние программного  продукта (системы) в процессе  разработки? | Нет | Да | Нет | Да | Да | Да |
| 3. | Будут ли пользователи вовлечены  во все фазы ЖЦ разработки? | Нет | Нет | Нет | Да | Нет | Нет |
| 4. | Будет ли заказчик отслеживать  ход выполнения проекта? | Нет | Да | Нет | Да | Нет | Нет |

Вычисления: 4 за каскадную, 1 за V- образную, 3 за RAD, 1 за инкрементную, 1 за быстрого прототипирования и 2 за эволюционную.

Итог: На основе результатов заполнения табл. 5 подходящей является каскадная

Таблица 6 – Выбор модели жизненного цикла на основе характеристик типа проектов и рисков

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № критерия | Критерии категории типов  проекта и рисков | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| 1. | Разрабатывается ли в проекте  продукт нового для организации  направления? | Нет | Нет | Да | Да | Нет | Нет |
| 2. | Будет ли проект являться  расширением существующей  системы? | Нет | Да | Нет | Да | Да | Нет |
| 3. | Будет ли проект крупно- или  среднемасштабным? | Нет | Да | Нет | Да | Да | Да |
| 4. | Ожидается ли длительная  эксплуатация продукта? | Нет | Да | Да | Да | Нет | Да |
| 5. | Необходим ли высокий уровень  надежности продукта проекта? | Нет | Да | Да | Да | Нет | Нет |
| 6. | Предполагается ли эволюция  продукта проекта в течение ЖЦ? | Нет | Нет | Да | Да | Да | Нет |
| 7. | Велика ли вероятность изменения  системы (продукта) на этапе  сопровождения? | Нет | Да | Нет | Нет | Нет | Да |
| 8. | Является ли график сжатым? | Да | Нет | Нет | Нет | Да | Нет |
| 9. | Предполагается ли повторное  использование компонентов? | Нет | Нет | Нет | Да | Нет | Нет |
| 10. | Являются ли достаточными ресурсы (время, деньги,  инструменты, персонал)? | Да | Нет | Нет | Да | Да | Да |

Вычисления: 9 за каскадную, 4 за V- образную, 5 за RAD, 1 за инкрементную, 5 за быстрого прототипирования и 5 за эволюционную.

Итог: На основе результатов заполнения табл. 6 подходящей является каскадная.

Общий итог: 21 за каскадную, 8 за V- образную, 18 за RAD, 8 за инкрементную, 13 за быстрого прототипирования и 12 за эволюционную.

В итоге заполнения табл. 3 – 6 наиболее подходящей является инкрементная модель.