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.

**2 Проектирование**

**2.1 Выбор стратегии разработки и модели жизненного цикла**

Для разработки сайта для печати фото ‘CustomPrintCo’ 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 | Да |
| 4. | Нужно ли демонстрировать требования с целью их определения? | Нет | Нет | Да | Нет | Да | Да |
| 5. | Требуется ли проверка концепции программного средства или системы? | Нет | Нет | Да | Нет | Да | Да |
| 6. | Будут ли требования изменяться или уточняться с ростом сложности системы (программного средства) в ЖЦ? | Нет | Нет | Нет | Да | Да | Да |
| 7. | Нужно ли реализовать основные требования на ранних этапах разработки? | Нет | Нет | Да | Да | Да | Да |

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

Итог: На основе результатов заполнения табл. 3 подходящей является RAD модель и инкрементная модель.

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 | Нет | Да | Да |
| 2. | Являются ли инструментальные средства, используемые в проекте, новыми для большинства разработчиков? | Да | Да | Нет | Нет | Нет | 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? | Да | Да | Нет | Да | Нет | Нет |
| 5. | Важна ли легкость распределения человеческих ресурсов проекта? | Да | Да | Да | Да | Нет | Нет |
| 6. | Приемлет ли команда разработчиков оценки, проверки, стадии разработки? | Да | Да | Нет | Да | Да | Да |

Вычисления: 5 за каскадную, 5 за V-образную, 4 за RAD, 5 за инкрементную, 2 за быстрого прототипирования и 1 за эволюционную.

Итог: На основе результатов заполнения табл. 4 подходящими являются каскадная, V-образная и инкрементная модели.

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 | Да | Нет | Да | Нет | Да |
| 2. | Будут ли пользователи оценивать текущее состояние программного продукта (системы) в процессе разработки? | Нет | Нет | Нет | Да | Да | Да |
| 3. | Будут ли пользователи вовлечены во все фазы ЖЦ разработки? | Нет | Нет | Да | Нет | Да | Нет |
| 4. | Будет ли заказчик отслеживать ход выполнения проекта? | Нет | Нет | Нет | Нет | Да | Да |

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

Итог: На основе результатов заполнения табл. 5 подходящей является модель быстрого проектирования и Эволюционная.

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 | Нет | Да | Да | Да |
| 2. | Будет ли проект являться расширением существующей системы? | Да | Да | Да | Да | Нет | Нет |
| 3. | Будет ли проект крупно- или среднемасштабным? | Нет | Нет | Нет | Да | Да | Да |
| 4. | Ожидается ли длительная эксплуатация продукта? | Да | Да | Нет | Да | Нет | Да |
| 5. | Необходим ли высокий уровень надежности продукта проекта? | Нет | Да | Нет | Да | Нет | Да |
| 6. | Предполагается ли эволюция продукта проекта в течение ЖЦ? | Нет | Нет | Нет | Да | Да | Да |
| 7. | Велика ли вероятность изменения системы (продукта) на этапе сопровождения? | Нет | Нет | Нет | Да | Да | Да |
| 8. | Является ли график сжатым? | Нет | Нет | Да | Да | Да | Да |
| 9. | Предполагается ли повторное использование компонентов? | Нет | Нет | Да | Да | Да | Да |
| 10. | Являются ли достаточными ресурсы (время, деньги, инструменты, персонал)? | Нет | Нет | Нет | Нет | Да | Да |

Вычисления: 2 за каскадную, 3 за V-образную, 3 за RAD, 7 за инкрементную, 7 за быстрого прототипирования и 9 за эволюционную.

Итог: На основе результатов заполнения табл. 6 подходящей является эволюционная модели.

Общий итог: в итоге заполнения табл. 3 – 6 , 12 – за каскадную, 13 – за V-образную, 13 – за RAD, 19 – за инкрементную, 15 – за быстрого прототипирования, 16 – за эволюционную. Наиболее подходящей является инкрементная модель.

Контрольные вопросы:

1.1. «кодирование – устранение ошибок»:

- Основная идея этой стратегии заключается в том, что разработка программного обеспечения начинается с быстрого написания кода, а затем идет процесс выявления и устранения ошибок.

- Достоинства: быстрый старт разработки, возможность быстро исправлять ошибки.

- Недостатки: недостаточное планирование, возможные проблемы с качеством исходного кода.

- Рекомендуется использовать для небольших проектов с краткими сроками выполнения.

1.2. Каскадная стратегия разработки программного обеспечения:

- Последовательный процесс разработки, где каждая фаза зависит от успешного завершения предыдущей.

- Достоинства: простота управления проектом, хорошее понимание требований заказчика.

- Недостатки: отсутствие возможности вносить изменения на поздних этапах, риски связанные с недостатком адаптивности.

- Рекомендуется использовать для проектов с четко определенными требованиями и стабильной спецификацией.

1.3. Инкрементная стратегия разработки программного обеспечения:

- Разработка происходит последовательно итеративно, с постепенным добавлением новых функций и улучшений.

- Достоинства: возможность быстро реагировать на изменения требований, улучшение качества продукта на ранних этапах.

- Недостатки: возможные проблемы с масштабируемостью проекта, дополнительные затраты на управление процессом разработки.

- Рекомендуется использовать для проектов, требующих гибкости и поддержки изменений в требованиях.

1.4. Эволюционная стратегия разработки программного обеспечения:

- Процесс разработки происходит через итерации с постепенным улучшением продукта на основе обратной связи.

- Достоинства: способствует развитию продукта в соответствии с потребностями пользователей, возможность итеративного улучшения.

- Недостатки: возможны сложности с управлением проектом, необходимость постоянного сбора обратной связи.

- Рекомендуется использовать для проектов, требующих пошагового улучшения и развития с учетом изменяющихся требований пользователей.

2. Модель жизненного цикла - это концепция, описывающая все этапы разработки программного продукта, начиная от определения требований и заканчивая его снятием с использования. В зависимости от специфики проекта, могут быть использованы различные модели жизненного цикла.

Из каскадной стратегии:

- Каскадная модель (водопад): последовательное выполнение этапов разработки без возможности возврата к предыдущему.

- Каскадная модель с возвратом: после завершения каждого этапа предусмотрены корректировки и изменения.

- V-образная модель: включает в себя структурирование тестирования согласно этапам разработки.

Из инкрементной стратегии:

- Инкрементная модель: пошаговое улучшение продукта через добавление новых функций.

- Модель быстрой разработки приложений (RAD-модель): быстрое создание прототипов для быстрой обратной связи.

Из эволюционной стратегии:

- Быстрое прототипирование: создание быстрых прототипов для оценки концепций и требований.

- Эволюционная модель: разработка продукта через итеративные циклы с улучшением и развитием.