Практическая работа 3

Тема: Выработка требований к программному обеспечению и программному модулю

Задания:

Жизненный цикл программного продукта (ПП), критерии качества ПП, виды ПО, стадии разработки ПП.

Разработка требований (определение, виды работ).

Пользовательские требования (User Requirements), системные требования (System Requirements), проектная системная спецификация (Software Design Specification) – определения.

Виды требований к ПП.

Функциональные требования, нефункциональные требования – определения.

Заполнить таблицу «Сравнение моделей разработки».

SADT-модели (определение, примеры).

CASE-средства (определение, примеры).

Объектно-ориентированный анализ и проектирование (OOAD) – определение, примеры инструментальных средств.

UML, языки UML (определение, примеры).

Группа разработчиков.

Выполнение заданий:

1. Жизненный цикл программного продукта (ПП), критерии качества ПП, виды ПО, стадии разработки ПП

Жизненный цикл программного продукта (ПП) – последовательность этапов развития программного обеспечения, начиная с идеи до завершения поддержки. Включает этапы:

Планирование и анализ требований.

Проектирование.

Реализация (кодирование).

Тестирование.

Эксплуатация и поддержка.

Завершение жизненного цикла.

Критерии качества ПП: надежность, удобство использования, производительность, безопасность, переносимость, сопровождаемость.

Виды ПО:

Прикладное программное обеспечение.

Системное программное обеспечение.

Инструментальное программное обеспечение.

Стадии разработки ПП:

Постановка задачи и анализ требований.

Проектирование.

Кодирование.

Тестирование.

Документирование.

Интеграция и развертывание.

Обслуживание и сопровождение.

2. Разработка требований (определение, виды работ)

Определение: Процесс выявления и документирования функций и характеристик, которым должна соответствовать система или продукт, чтобы удовлетворять потребности клиентов и бизнеса.

Виды работ:

Сбор требований.

Анализ и проверка требований.

Спецификация требований.

Утверждение требований.

3. Определения основных типов требований

Пользовательские требования (User Requirements): Определяют функциональные и качественные ожидания пользователей относительно будущего продукта. Обычно задаются на высоком уровне абстракции.

Системные требования (System Requirements): Подробные спецификации, определяющие конкретные функции и ограничения системы, необходимые для выполнения указанных пользовательских требований.

Проектная системная спецификация (Software Design Specification): Документация, содержащая детальные технические спецификации системы, включая структуру компонентов, интерфейсы, используемые технологии и стандарты.

4. Виды требований к ПП

Функциональные требования: Что система должна делать?

Поддерживать выполнение определенных действий.

Нефункциональные требования: Как система должна функционировать?

Производительность, безопасность, масштабируемость, доступность, удобство использования.

5. Определения ключевых видов требований

Функциональные требования: Четко обозначают, какие операции должна поддерживать система.

Нефункциональные требования: Устанавливают дополнительные условия функционирования, такие как производительность, надежность, удобства использования и другие метрики качества.

6. Таблица сравнения моделей разработки

№ п/п Модель разработки Особенности + -

1 Каскадная Последовательность стадий Простота планирования Сложность изменений

2 Итеративная Повторяемые циклы Возможность адаптации Увеличенные сроки

3 Гибкая (Agile) Постоянная обратная связь Высокая гибкость Необходимость постоянной связи

4 V-образная Строгость тестирования Раннее обнаружение ошибок Ограниченность изменений

Выбор модели: Я выбрал бы Agile-модель, поскольку она обеспечивает большую гибкость и адаптацию к изменениям требований, что особенно полезно в современных условиях неопределенности.

7. SADT-модели

SADT (Structured Analysis and Design Technique) – структурированная техника анализа и проектирования. Используется для моделирования процессов и потоков данных. Примеры инструментов: IDEF-диаграммы.

8. CASE-средства

CASE (Computer-Aided Software Engineering) – автоматизированные инструменты для помощи в разработке ПО. Примеры: Rational Rose, ERWin Data Modeler.

9. OOAD (Object-Oriented Analysis and Design)

ООА (Объектно-ориентированный анализ) – выявление объектов, классов и взаимодействий. ООД (Объектно-ориентированное проектирование) – детализированное проектирование классов и методов. Примеры инструментальных средств: StarUML, Enterprise Architect.

10. UML (Unified Modeling Language)

UML – унифицированный язык моделирования. Использует диаграммы для визуализации, спецификации и документации элементов системы. Примеры диаграмм: Use Case Diagram, Class Diagram, Sequence Diagram.

11. Группа разработчиков

Группа разработчиков состоит из ролей:

Руководитель проекта.

Архитектор.

Разработчик.

Тестировщик.

Специалист по качеству.

Менеджер по требованиям.

Заключение

Данная практика позволила углубленно изучить методики выработки требований к программному обеспечению, рассмотреть основные концепции проектирования и выбора подходящих моделей разработки. Применение полученных знаний обеспечит эффективное создание качественного программного продукта.