**Практическая работа №4**

**Выработка требований к программному обеспечению и программному модулю**

**1. ЖЦ ПП, критерии качества ПП, виды ПО, стадии разработки ПП**

**Жизненный цикл ПО (ЖЦ ПП)** — это последовательность этапов, через которые проходит программное обеспечение от момента возникновения идеи до окончания его использования.

**Критерии качества ПО:**

* Функциональность
* Надежность
* Удобство использования
* Эффективность
* Сопровождаемость
* Переносимость

**Виды ПО:**

* Системное ПО
* Прикладное ПО
* Инструментальное ПО

**Стадии разработки ПО:**

1. Анализ требований
2. Проектирование
3. Реализация
4. Тестирование
5. Внедрение
6. Сопровождение

### ****2. Разработка требований (определение, виды работ)****

**Разработка требований** — процесс выявления, документирования и поддержки требований к системе.

**Виды работ:**

* Сбор требований
* Анализ требований
* Спецификация требований
* Верификация требований
* Управление изменениями требований

### ****3. Определения****

* **Пользовательские требования** — высокоуровневое описание того, что система должна делать, с точки зрения пользователя.
* **Системные требования** — детальное описание функций, возможностей и ограничений системы.
* **Проектная системная спецификация** — технический документ, описывающий архитектуру, компоненты и интерфейсы системы.

### ****4. Виды требований к ПП****

* Функциональные требования
* Нефункциональные требования
* Бизнес-требования
* Пользовательские требования
* Системные требования

### ****5. Функциональные и нефункциональные требования****

* **Функциональные требования** — определяют, что система должна делать.
* **Нефункциональные требования** — определяют, как система должна выполнять свои функции (производительность, безопасность, надежность).

### ****6. Таблица «Сравнение моделей разработки»****

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № п/п | Модель разработки | Особенности | + | - |
| 1. | Водопадная | Последовательная разработка | Простота управления | Сложность внесения изменений |
| 2. | Итерационная | Циклическая разработка | Гибкость, ранее тестирование | Высокие затраты на управление |
| 3. | Спиральная | Учет рисков, итерации | Управление рисками | Сложность реализации |
| 4. | Agile | Гибкость, короткие итерации | Быстрая адаптация к изменениям | Требует высокой дисциплины команды |

### **Вывод:** Для проекта «Учет успеваемости студентов» я выбрал бы **Agile-модель**, так как она позволяет быстро адаптироваться к изменениям требований заказчика и обеспечивает постоянное взаимодействие с пользователями.

### ****7. SADT-модели (определение, примеры)****

**SADT (Structured Analysis and Design Technique)** — методология структурного анализа и проектирования систем.

**Пример:** Диаграммы IDEF0 для моделирования бизнес-процессов университета.

### ****8. CASE-средства (определение, примеры)****

**CASE-средства** — инструменты для автоматизации процессов разработки ПО.

**Примеры:** IBM Rational Rose, Enterprise Architect, Visual Paradigm.

### ****9. Объектно-ориентированный анализ и проектирование (OOAD)****

**OOAD** — методология, основанная на представлении системы в виде совокупности объектов.

**Инструментальные средства:** Rational Unified Process (RUP), IBM Rational Rose, StarUML.

### ****10. UML, языки UML (определение, примеры)****

**UML (Unified Modeling Language)** — стандартизированный язык визуального моделирования ПО.

**Примеры диаграмм:** Use Case, Class, Sequence, Activity.

### ****11. Группа разработчиков****

Типовая структура:

* Project Manager
* Business Analyst
* System Architect
* Backend/Frontend Developers
* QA Engineer
* Technical Writer

**Роли в проекте «Учет успеваемости студентов»:**

* Аналитик (сбор требований деканата)
* Архитектор (проектирование БД)
* Backend-разработчик (реализация логики)
* Frontend-разработчик (интерфейс для сотрудников)
* Тестировщик (проверка функциональности)

Вывод

Мы молодцы, мы все поняли, мы самые лучшие котейки.