ПРАКТИЧЕСКАЯ РАБОТА №2

Инструкционно-технологическая карта

Тема: «Планирование тестирования»

Цель работы: Знать основные задачи планирования тестовых испытаний, знать основные раздела тест плана, научиться составлять тестовый план.

Время выполнения: 2 часа

1. Теоретические сведения для выполнения работы

 Планирование тестирования

Планирование начинается с формирования общего видения продукта.

На этапе проектирования группа соответствующих специалистов решает, как будут реализованы запланированные возможности продукта. Они разрабатывают внешний дизайн программного продукта (то, каким будет продукт с точки зрения пользователя) и его внутреннюю структуру. Обе эти составляющие тесно связаны между собой и проектируются одновременно.

Процесс тестирования тесно связан с процессом разработки. Соответственно планирование тестирования тоже зависит от выбранной модели разработки. Однако вне зависимости от модели разработки при планировании тестирования необходимо ответить на несколько вопросов, определяющих этот процесс:

**Кто будет тестировать и на каких этапах?**

Разработчики продукта, независимая группа тестировщиков или совместно?

**Какие компоненты надо тестировать?**

Будут ли подвергнуты тестированию все компоненты программного продукта или только компоненты, которые угрожают наибольшими потерями для всего проекта?

**Когда надо тестировать?**

Будет ли это непрерывный процесс, вид деятельности, выполняемый в специальных контрольных точках, или вид деятельности, выполняемый на завершающей стадии разработки?

**Как надо тестировать?**

Будет ли тестирование сосредоточено только на проверке того, что данный продукт должен выполнять, или также на том, как это реализовано?

**В каком объеме тестировать?**

Как определить, в достаточном ли объеме выполнено тестирование, или как распределить ограниченные ресурсы, выделенные под тестирование?

**Кто будет тестировать?**

Разработчик- это роль, для которой характерны виды деятельности, ориентированные на создание программного продукта (ПП).

Тестировщик - это роль, для которой характерны виды деятельности, ориентированные на обнаружение ошибок в ПП. Эта роль предусматривает выбор тестов, необходимых для конкретных целей, построение тестов, выполнение тестов и оценку результатов. Конкретный исполнитель проекта может выступать как в роли разработчика, так и в роли тестировщика.

**Какие компоненты надо тестировать?**

Могут быть варианты, когда ничего не надо тестировать (поскольку все компоненты были протестированы ранее), а может потребоваться тестировать каждый компонент с точностью до строки кода.

**Когда надо тестировать?**

Компоненты можно тестировать на завершающем этапе, когда они будут интегрированы в единый выполняемый модуль. Частота тестирования определяется различными соображениями. Можно проводить тестирование каждый день, учитывая тот факт, что чем раньше выявлена проблема, тем легче и дешевле ее решение. Можно тестировать программный компонент по мере завершения его разработки. Частое тестирование компонентов несколько замедляет ранние этапы разработки, однако сопряженные с этим потери с лихвой восполняются за счет меньшего числа проблем на более поздних этапах разработки проекта, когда отдельные модули объединяются в более крупные компоненты системы.

**Как надо тестировать?**

Основные подходы к тестированию ПО основаны на **спецификации** и **реализации.**

Спецификациямодуля (или класса) ПП определяет, что этот модуль должен делать, т.е. она описывает допустимые наборы входных данных, подаваемых на вход модуля, включая ограничения на то, как многократные вводы данных должны соотноситься друг с другом, и какие выходные данные соответствуют различным наборам входных данных.

Реализациямодуля ПП есть выражение алгоритма, порождающего выходные результаты для различных наборов входных данных с соблюдением требований спецификации. Спецификацияуказывает, что делает модуль ПП, а реализацияпоказывает, какмодуль ПП это делает.

Полный учет требований спецификации дает гарантию того, что ПП выполняет все, что от него требуется. Полный учет требований к реализации дает гарантию того, что ПП не будет делать того, что от него не требуется.

Спецификация играет важную роль в тестировании. Обычно для множества компонентов ПП создаются спецификации, обеспечивающие разработку и тестирование, включая спецификации систем, подсистем и классов.

Тестирование следует осуществлять в достаточных объемах, чтобы быть более-менее уверенным в том, что ПП функционирует в соответствии с предъявленными к нему требованиями, т.е. выполняется принцип **адекватности** тестирования ПП. Набор тестов можно считать адекватным, если определенная часть строк исходного кода или исполняемых ветвей исходного кода была выполнена, по крайней мере, один раз во время прогона тестового набора. Эти два способа измерения отражают два базовых подхода к тестированию:

1. при использовании первого подхода проверяется, что должен выполнять ПП;
2. при использовании второго подхода проверяется, как фактически работает ПП.

**В каком объеме тестировать?**

Объем необходимого тестирования следует определять исходя из краткосрочных и долгосрочных целей проекта и в соответствии с особенностями разрабатываемого ПП. **Покрытие** - это мера полноты использования возможностей программного компонента тестовым набором. Анализ рисков в процессе тестирования применяется для определения уровня детализации и времени, затрачиваемого на тестирование конкретного компонента.

Тестовый план

Ответы на поставленные вопросы и, возможно, на многие другие, оформляются в виде набора документов, принятого в компании.

**Тестовый план** может содержать следующую информацию:

1. Перечень тестовых ресурсов.
2. Перечень функций и подсистем, подлежащих тестированию.
3. Тестовую стратегию:

Анализ функций и подсистем с целью определения слабых мест, требующих исчерпывающего тестирования, то есть участков функциональности, где появление дефектов наиболее вероятно.

Определение стратегии выбора входных данных для тестирования.

Поскольку в реальных применениях множество входных данных программного продукта практически бесконечно, выбор конечного подмножества для проведения тестирования является сложной задачей. Для ее решения могут быть применены методы покрытия классов входных и выходных данных, анализ крайних значений, покрытие случаев использования и тому подобное. Выбранная стратегия должна быть обоснована и задокументирована.

Определение потребности автоматизации процесса тестирования. При этом решение об использовании существующей, либо о создании новой автоматизированной системы тестирования должно быть обосновано, а также продемонстрирована оценка затрат на создание новой системы или на внедрение уже существующей.

1. График (расписание) тестовых циклов.
2. Указание конкретных параметров аппаратуры и программного окружения.
3. Определение тестовых метрик, которые необходимо собирать и анализировать, таких как покрытие набора требований, покрытие кода, количество и уровень серьезности дефектов, объем тестового кода и т.п.
4. Порядок выполнения работы:
5. Изучить предлагаемый теоретический материал.
6. Ознакомиться с шаблонами тестирования.
7. Разработать тестовый план для программы, разработанной ранее в рамках курсовой работы по дисциплине «Основы алгоритмизации и программирование».
8. Знать ответы на контрольные вопросы.
9. Подготовить отчет по практической работе.
10. Контрольные вопросы:
11. Перечислить основные задачи планирования тестовых испытаний.
12. Назовите основные секции тестового плана. Что описывается в каждой из них?
13. Дайте определение понятию «Риски».
14. Каковы критерии хорошего тестового плана?