Белорусский государственный технологический университет

Факультет информационных технологий

Кафедра программной инженерии

 Лабораторная работа 5

По дисциплине «Разработка и анализ требований»

На тему «Критерии приемки»

Выполнила:

Студентка 2 курса 9 группы

Бондарик Никита Дмитриевич

Преподаватель: Ромыш А.С.

2025, Минск

## 1. Пользовательская история

Как пользователь. Я хочу иметь возможность добавлять комплектующие в конфигурацию ПК, Чтобы я мог собрать компьютер с нужными характеристиками и увидеть итоговую стоимость.

## 2. Критерии приемки в виде свода правил

2.1 Уточняющие вопросы заказчику:

* Какие категории комплектующих должны быть доступны для выбора (процессор, видеокарта, ОЗУ и т.д.)?
* Должен ли пользователь видеть совместимость выбранных компонентов?
* Как будет отображаться итоговая стоимость конфигурации?
* Какие действия пользователя должны приводить к ошибкам (например, выбор несовместимых компонентов)?
* Нужна ли возможность сохранить конфигурацию для дальнейшего использования?

2.2 Ответы заказчика (предполагаемые):

* Доступные категории: процессор, видеокарта, материнская плата, ОЗУ, SSD/HDD, блок питания, корпус.
* Да, система должна проверять совместимость и уведомлять пользователя о конфликтах.
* Итоговая стоимость отображается в реальном времени при добавлении/удалении компонентов.
* Ошибки: несовместимость компонентов, превышение бюджета (если задан).
* Да, пользователь может сохранить конфигурацию под уникальным именем.

2.3 Критерии приемки (позитивные и негативные):

* Пользователь может добавлять комплектующие из списка доступных категорий.
* Система проверяет совместимость компонентов и выводит предупреждение при конфликте.
* Итоговая стоимость отображается автоматически при изменении конфигурации.
* При выборе несовместимых компонентов система блокирует добавление и показывает сообщение об ошибке.
* Пользователь может сохранить конфигурацию, указав имя, и вернуться к ней позже.
* При попытке сохранить конфигурацию без названия система выводит сообщение об ошибке.
* Если пользователь превышает заданный бюджет, система уведомляет его.

2.4 Нефункциональные требования:

Время отклика системы при добавлении компонента не должно превышать 2 секунд.

Система должна поддерживать одновременную работу 1000 пользователей без потери производительности.

Все данные конфигураций должны храниться в зашифрованном виде для защиты пользовательской информации.

## 3. Критерии приемки в формате Given-When-Then

Сценарий 1: Добавление совместимого компонента

Given Пользователь открыл страницу конфигуратора ПК When Пользователь выбирает процессор "Intel Core i7" и совместимую материнскую плату  
Then Система добавляет компоненты в конфигурацию и обновляет итоговую стоимость

Сценарий 2: Попытка добавить несовместимый компонент

Given Пользователь добавил процессор "AMD Ryzen 7" When Пользователь пытается выбрать материнскую плату с сокетом LGA 1200  
Then Система выводит сообщение: "Несовместимые компоненты. Выберите другой вариант."

Сценарий 3: Сохранение конфигурации

Given Пользователь собрал конфигурацию ПК When Пользователь нажимает "Сохранить" и вводит имя "Игровой ПК"  
Then Система сохраняет конфигурацию и выводит сообщение: "Конфигурация сохранена."

## 4. Ответы на вопросы

Определение терминов:

Definition of Ready (DoR): Критерии, определяющие готовность задачи (например, User Story) к началу разработки.

Definition of Done (DoD): Критерии, определяющие завершенность задачи и её готовность к передаче пользователю.

Acceptance Criteria (AC): Условия, которые должны быть выполнены, чтобы задача считалась успешно реализованной.

Этапы применения критериев:

DoR: Применяется на этапе планирования спринта, чтобы убедиться, что задача готова к разработке.

AC: Используются во время разработки и тестирования для проверки соответствия требованиям.

DoD: Применяются на этапе завершения задачи, чтобы убедиться, что она готова к релизу.

Уникальность критериев:

DoR и DoD одинаковы для всех User Stories в проекте.

Acceptance Criteria уникальны для каждой User Story.

## 5. Выбор формата записи критериев приемки

Для данного случая лучше подходит сценарно-ориентированный формат (Given-When-Then), так как он:

Позволяет четко описать поведение системы с точки зрения пользователя.

Упрощает взаимодействие между разработчиками, тестировщиками и заказчиком.

Позволяет легко преобразовать критерии в автоматизированные тесты.

Однако свод правил также полезен для нефункциональных требований и дополнительных условий, которые сложно описать в формате сценариев.

Итог: Оба формата можно комбинировать для полноты описания критериев приемки.