

# Тестирование в Scrum, Kanban

Тестирование неразрывно связано с требованиями к ПО, поэтому формат требований и [процесс разработки требований](#) определяют то, каким образом на проекте лучше всего организовать тестирование.

Для каждого процесса характерны свои условия и ограничения. Например, **Kanban лучше подходит для проекта**, где объем работы заранее не известен и его сложно планировать. **Scrum подходит для проектов**, где требования достаточно просты и понятны, чтобы их можно было реализовать в течение пары недель. В обоих случаях можно обойтись меньшим объемом документации, хотя на это влияют как требования заказчика, так и сложность продукта и длительность разработки/поддержки разрабатываемого ПО.

Вот несколько процессов тестирования, которые встречаются чаще всего.

## Тестирование в Scrum или Kanban

1. Определение основных направлений и видов тестирования осуществляется на этапе анализа [пользовательских историй](#) (для Kanban) или планирования историй в спринте (для Scrum). Существенные [требования к качеству](#) историй фиксируются в критериях приемки.
2. На стадии тестирования (для [Kanban](#)) или при выполнении задачи по тестированию в спринте (для [Scrum](#)) для каждой истории разрабатываются детальные тестовые сценарии для ручного тестирования, важные сценарии включаются в тест-планы для последующих интеграционного и приемочного тестирования. Разрабатываются автоматизированные функциональные тесты, на основе которых будет осуществляться регрессионное тестирование продукта. По окончании реализации историй выполняются ручные и автоматизированные тесты, обнаруженные ошибки оперативно исправляются.
3. По окончании спринта (для Scrum) или перед выпуском сборки (для Kanban) производится интеграционное, приемочное и исследовательское тестирование на основе разработанных ранее тест-планов.
4. Обнаруженные ошибки попадают в бэклог продукта или спринта в зависимости от их критичности. Результаты ручного и [автоматического тестирования](#) сразу становятся доступны для всех участников проекта в списке сборок или тестов.

# Тестирование по требованиям

1. Для создания качественных системных требований тестировщики привлекаются для ревью на этапе разработки требований, таким образом документируются важные аспекты требований, проверяется их тестируемость. Процесс ревью может быть формализован путем настройки жизненного цикла (состояний) требований и организации процесса согласования.
2. С началом разработки тестировщики определяют направления и виды тестирования, разрабатывают соответствующие планы тестирования, основные тестовые сценарии и связывают их с исходными требованиями.
3. По окончании разработки релиза тестировщики дополняют и уточняют тестовую документацию, разрабатывают автоматизированные функциональные тесты, выполняют [функциональное](#), интеграционное, регрессионное и исследовательское тестирование. Обнаруженные ошибки попадают в бэклог продукта, планируется исправление и затем ошибки перепроверяются по ранее подготовленным тестовым сценариям.
4. Тестовая документация добавляется в [бейзлайны](#), соответствующие версиям продукта (то есть версиям требований).