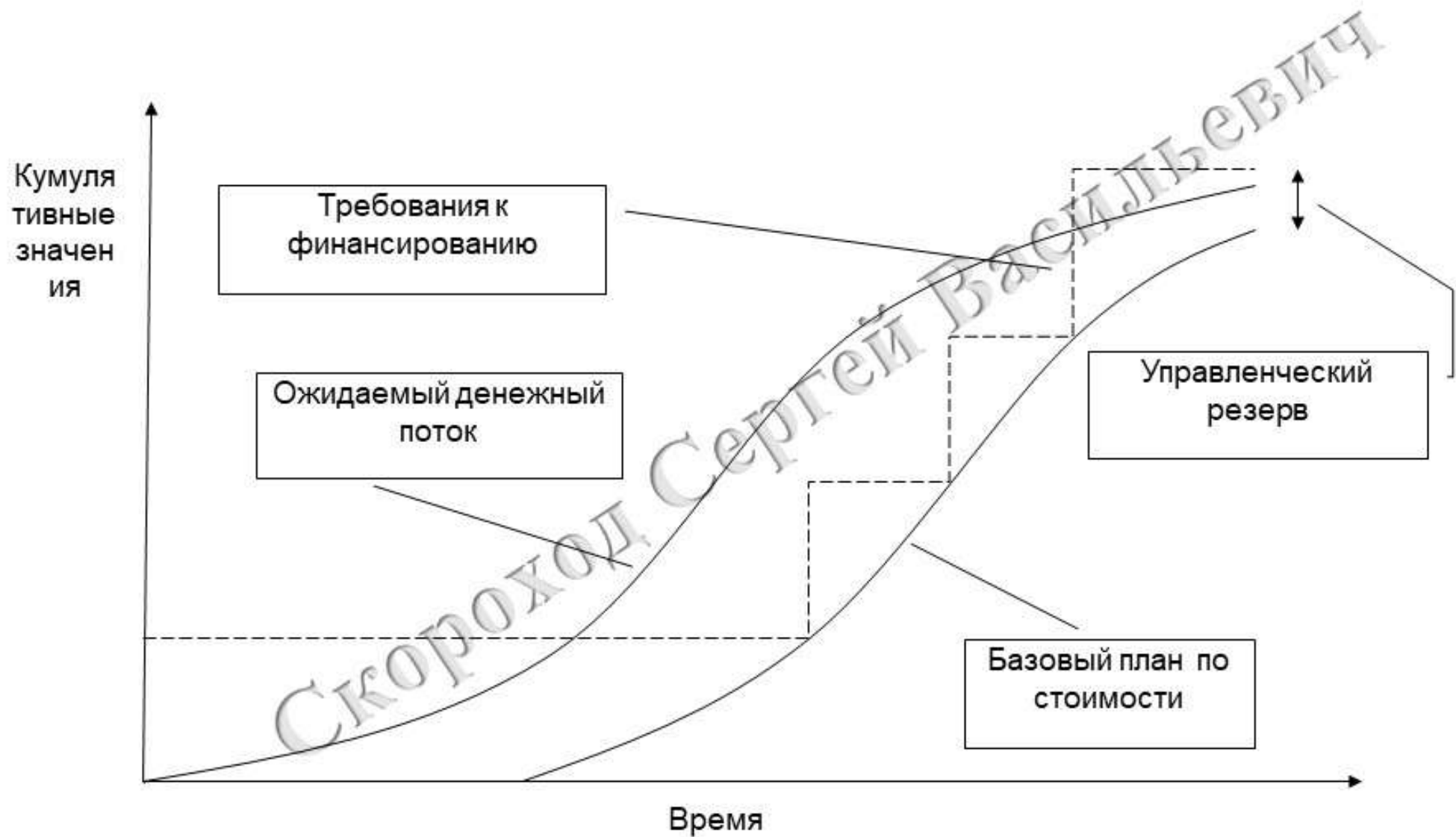


Управление стоимостью проекта (окончание)

Лектор — Скороход С. В.

Результаты разработки бюджета расходов



Основные показатели методики освоенного объема

1. **Базовая стоимость запланированных работ (БСЗР, PV).** Обозначает общую стоимость работ, которые должны быть завершены к текущему моменту (каковы должны быть затраты на проект по базовому плану).
2. **Фактическая стоимость выполненных работ (ФСВР, AC).** Обозначает общую фактическую стоимость трудозатрат на текущий момент (сколько фактически потрачено на проект к текущему моменту).
3. **Базовая стоимость выполненных работ (БСВР, EV).** Обозначает запланированную по базовому плану стоимость фактически выполненных работ (сколько планировалось потратить на трудозатраты, которые были фактически осуществлены).
4. **Бюджет по завершении (БПЗ, BAC).** Плановые затраты на проект (или задачу) согласно базовому плану

Производные показатели методики освоенного объема

Название	Формула	Знач.	Трактовки
Отклонение от календарного плана (ОКП, SV)	$ОКП = БСВР - БСЗР$	<0 =0 >0	Отставание от плана Выполнение в срок Опережение плана
Отклонение по стоимости (ОПС, CV)	$ОПС = БСВР - ФСВР$	<0 =0 >0	Превышение затрат Затраты по плану Экономия средств
Относительное отклонение по стоимости (ООПС)	$ООПС = ОПС / БСВР * 100$	<0 =0 >0	Превышение затрат Затраты по плану Экономия средств
Индекс отклонения стоимости (ИОС, CPI)	$ИОС = БСВР / ФСВР$	<1 =1 >1	Превышение затрат Затраты по плану Экономия средств
Относительное отклонение от календарного плана (ООКП)	$ООКП = ОКП / БСЗР * 100$	<0 =0 >0	Отставание от плана Выполнение в срок Опережение плана

Производные показатели методики освоенного объема

Название	Формула	Знач.	Трактовки
Индекс отклонения от календарного плана (ИОКП, SPI)	$\text{ИОКП} = \text{БСВР} / \text{БСЗР}$	<1 =1 >1	Отставание от плана Выполнение в срок Опережение плана
Предварительная оценка по завершении (ПОПЗ, EAC)	$\text{ПОПЗ} = \text{ФСВР} + (\text{БПЗ} - \text{БСВР}) / \text{ИОС}$	<БПЗ =БПЗ >БПЗ	Экономия средств Затраты по плану Превышение затрат
Отклонение по завершении (ОПЗ, VAC)	$\text{ОПЗ} = \text{БПЗ} - \text{ПОПЗ}$	<0 =0 >0	Превышение затрат Затраты по плану Экономия средств
Показатель эффективности выполнения (ПЭВ)	$\text{ПЭВ} = (\text{БПЗ} - \text{БСВР}) / (\text{БПЗ} - \text{ФСВР})$	<1 =1 >1	Средства экономятся Ход работ по плану Превышение затрат
Прогноз до завершения (ПДЗ, ETC)	$\text{ПДЗ} = \text{ПОПЗ} - \text{ФСВР}$		Ожидаемая стоимость выполнения оставшейся части проекта

Пример оценки проекта по методу освоенного объема

Показатель	Интерпретация
БСЗР = 1000	На момент оценки по проекту запланировано работ на эту сумму
ФСВР = 900	На момент оценки по проекту выполнено работ на эту сумму
БСВР = 1100	Те работы, которые выполнены к моменту оценки, по базовому плану стоят столько
БПЗ = 10000	Столько планируется потратить на все работы проекта
ОКП = БСВР – БСЗР = 1100 – 1000 = 100	Отклонение от календарного плана > 0 – опережение плана
ОПС = БСВР – ФСВР = 1100 – 900 = 200	Отклонение по стоимости > 0 – экономия средств, сэкономлено 200р
ООПС = ОПС / БСВР * 100 = 200/1100*100 = 18,2	Экономия средств в процентах Сэкономлено 18,2% средств от запланированного объема
ИОС = БСВР / ФСВР = 1100 / 900 = 1,2	Индекс отклонения по стоимости. Экономия затрат на 20%

Пример оценки проекта по методу освоенного объема

Показатель	Интерпретация
$ООКП = ОКП / БСЗР * 100 = 100/1000*100=10$	Относительное отклонение от календарного плана: опережение графика работ на 10%
$ИОКП = БСВР / БСЗР = 1100/1000 = 1,1$	Индекс отклонения от календарного плана: опережение графика работ на 10%
$ПОПЗ = ФСВР + (БПЗ - БСВР) / ИОС = 900 + (10000 - 1100)/1,2 = 8316$	Столько будет фактически стоить весь проект, если мы будем работать также как работали до момента оценки
$ОПЗ = БПЗ - ПОПЗ = 10000 - 8316 = 1684$	Отклонение по завершении: К концу проекта мы сэкономим столько средств, если будем работать также как работали до момента оценки
$ПЭВ = (БПЗ - БСВР)/(БПЗ - ФСВР) = (10000 - 1100) / (10000 - 900) = 0,98$	Показатель эффективности выполнения: Проект обойдется нам в 98% запланированной ранее суммы, если будем работать также как работали до момента оценки
$ПДЗ = ПОПЗ - ФСВР = 8316 - 900 = 7416$	Прогноз до завершения: На оставшуюся часть проекта потребуются потратить эту сумму, если будем работать также как работали до момента оценки

Управление качеством проекта

Лектор – Скороход С. В.

Качество и процессы управления качеством

Качество – это степень, в которой совокупность внутренних характеристик объекта соответствует заданным требованиям.

Управление качеством проекта направлено как на управление проектом, так и на поставляемые результаты проекта.

Процессы управления качеством.

1. **Планирование качества** – определение того, какие из стандартов качества относятся к данному проекту и как их удовлетворить.
2. **Процесс обеспечения качества** – выполнение плановых систематических мероприятий по качеству, необходимых для того, чтобы проект соответствовал оговоренным требованиям.
3. **Процесс контроля качества** – мониторинг результатов с целью определения их соответствия принятым стандартами качества, и определение путей устранения причин, вызывающих неудовлетворительное исполнение.

Положения управления качеством

1. Удовлетворение потребностей заказчика.

- Понимание, оценка, выявление и управление ожиданиями заказчика так, чтобы его требования оказались выполненными.
- Для этого нужно обеспечить сочетание соответствия требованиям (проект должен создать то, что было заявлено) и пригодности к использованию (результат проекта должен удовлетворять реальным потребностям).

2. Предотвращение важнее инспектирования.

- Затраты на предупредительные меры по предотвращению ошибок всегда гораздо ниже, чем затраты на их исправления после обнаружения по результатам инспектирования.

3. Ответственность руководства.

- Для достижения успеха требуется участие всей команды, но обеспечение ресурсами, которые необходимы для достижения успеха – это обязанность руководства.

4. Постоянное совершенствование.

- Непрерывно повторяющийся цикл "планирование – исполнение – проверка – воздействие" – это основа повышения качества

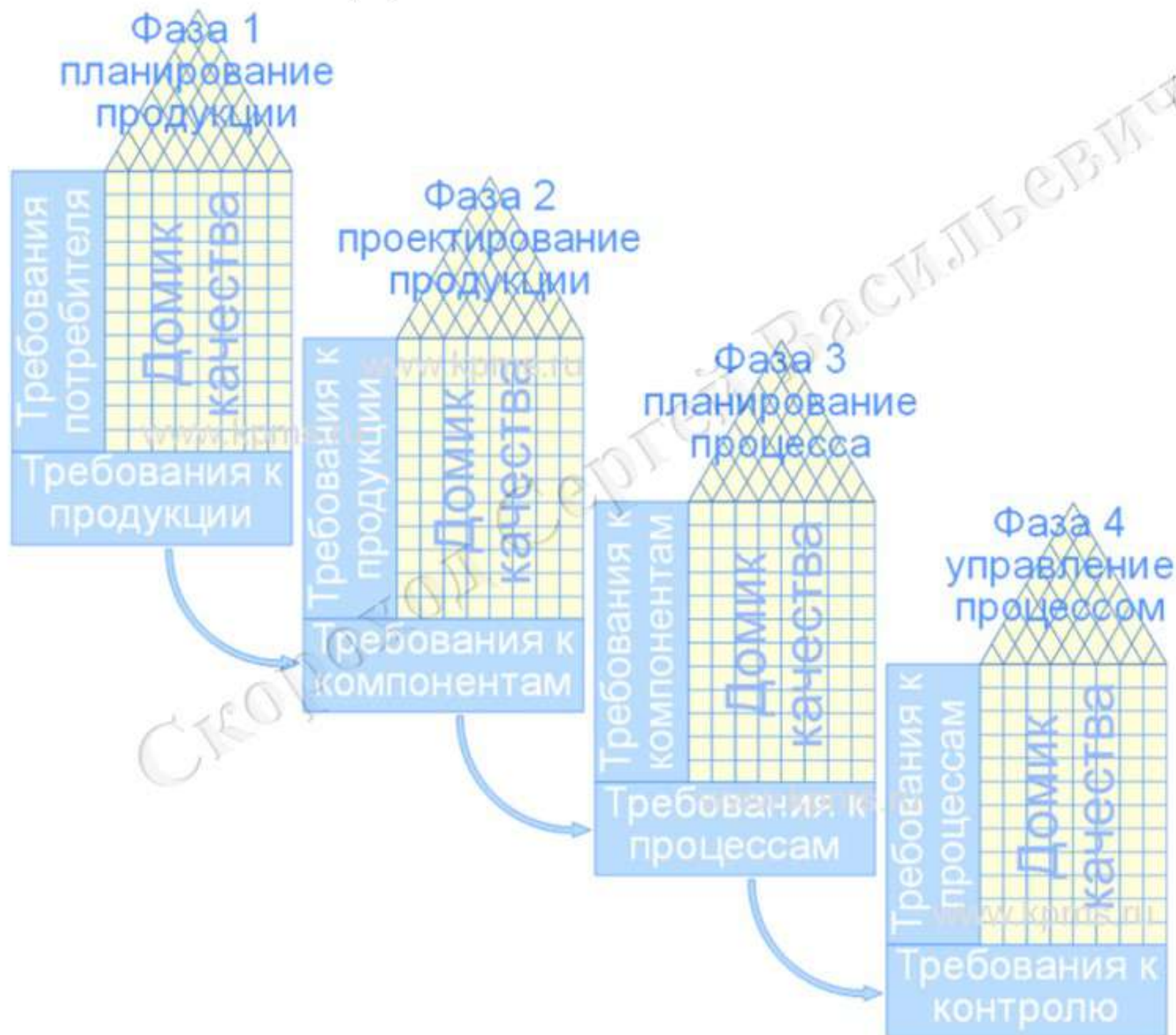
Технология развертывания функций качества QFD

- Развертывание функций качества - системный подход к проектированию, основанный на четком понимании желаний потребителей.
- Применение этой технологии позволяет перевести пожелания потребителя в технические характеристики изделия.
- Развертывание функций качества позволяет решить ряд важных задач при создании продукта проекта.
 1. Определить приоритетность пожеланий и ожиданий потребителя, как высказанных в явной форме, так и предполагаемых.
 2. Перевести эти пожелания и ожидания в технические характеристики и спецификации.
 3. Создать и предоставить качественный продукт или услугу с характеристиками, направленными на удовлетворение всех важных и существенных запросов потребителя.

Домик качества

- Технология QFD включает в себя 4 фазы, на каждой из которых применяется матричная диаграмма особого вида – **домик качества**.
- Каждая фаза представляет определенные аспекты требований к продукции или услуге.
- На каждой фазе производится оценка взаимосвязи между элементами домика качества.
- Только наиболее важные аспекты переходят на следующую фазу, в следующий домик качества.

Домик качества



Фазы QFD

Фаза 1. Планирование продукции

- Домик качества помогает перевести пожелания и требования потребителя в технические характеристики изделия.
- В работе принимают участие специалисты, непосредственно взаимодействующие с потребителями. Эффективность QFD целиком зависит от качества данных, полученных на этой фазе.

Фаза 2. Проектирование продукции

- Данная фаза выполняется специалистами инженерно-технических служб (проектировщиками, архитекторами ПО, дизайнерами).
- Проектирование продукта требует творческих и инновационных идей.
- На фазе 2 разрабатывается концепция будущего продукта, и документируются спецификации (схемы, технические требования) всех компонентов.
- Второй домик качества помогает перевести технические характеристики продукта в целом в технические спецификации компонентов.

Фазы QFD

Фаза 3. Планирование процесса

- Фаза планирования процесса выполняется производственными службами (разработчиками, специалистами отдела закупок).
- В ходе работы осуществляются мероприятия по планированию, организации и подготовке процесса разработки.
- В результате этой фазы документируется порядок выполнения и параметры (ключевые характеристики) процесса.
- Третий домик качества дает возможность связать характеристики компонентов изделия с параметрами и характеристиками процесса разработки.

Фаза 4. Управление процессом

- В реализации этой фазы принимают участие сотрудники службы качества.
- На выходе создаются документы, включающие в себя контролируемые показатели процесса разработки.

печатающего устройства

Шаг 1. Определение требований потребителя

- Требования вносятся в матрицу - домик качества в раздел требования потребителя.
- Требования могут быть структурированы по видам, например, эргономичность, исполнение и т.п.

[illegible]

печатающего устройства

Шаг 2. Определение важности требований для потребителя

- Для определения рейтинга важности может применяться шкала от 1 до 5, где 5 означает максимальную важность, а 1 минимальную важность.
- Чтобы ранжировать требования потребителей по степени важности применяют матрицу приоритетов или метод консенсуса.
- Результат заносят в домик качества.

[illegible]

Построение домика качества на примере

Шаг 3. Определение конкурентного рейтинга потребителя

- Конкурентный рейтинг дает возможность установить конкурентные преимущества разрабатываемого продукта или услуги в сравнении с аналогичными.
- Для сравнения выбираются продукты нескольких компаний и проводится их сравнительная оценка.

		Рейтинг потребителя				
		<div>Наша компания</div> <div>Компания А</div> <div>Компания Б</div>				
		1	2	3	4	5
Эргономичность	Легко вставить картридж	3				
	Легко активировать	3				
	Легко подключить	5				
	Не требует настроек	4				
Исполнение	Малый вес	2				
	Разные цвета корпуса	1				
	Безотказность в работе	4				
	Не ломается при падении	3				

Построение домика качества на примере

Шаг 4. Определение технических требований.

На данном шаге домик качества содержит только названия требований.

Технические требования		Требования потребителей		Важность для потребителя	Программные средства		Размеры		Технические особенности		Рейтинг потребителя						
					Соответствие стандартов	Количество интерфейсов	Количество систем	Количество цветов	Количество типоразмеров	Диапазон рабочих температур	Удельный вес материалов	Усилие на включение	Рейтинг потребителя				
													1	2	3	4	5
Эргономичность	Легко вставить картридж	3									●	■	▲				
	Легко активировать	3												■	●		
	Легко подключить	5										■			▲		
	Не требует настроек	4										●	■		▲		
Исполнение	Малый вес	2									●		▲	■			
	Разные цвета корпуса	1										●	▲	■			
	Безотказность в работе	4										■	●		▲		
	Не ломается при падении	3										▲		●	■		

Построение домика качества на примере

Шаг 5. Построение матрицы взаимосвязи.

- Для выявления силы взаимосвязи применяется шкала значений 9,3,1, где 9 означает сильную взаимосвязь, 3- среднюю, 1 – слабую.

Технические требования		Требования потребителей		Важность для потребителя		Программные средства		Размеры		Технические особенности		Рейтинг потребителя				
						Соответствие стандартов	Количество интерфейсов	Количество систем	Количество цветов	Количество типоразмеров	Диапазон рабочих температур	Удельный вес материалов	Усилие на включение	■ Наша компания	▲ Компания А	● Компания Б
Эргономичность	Легко вставить картридж	3					▲					●	■	▲		
	Легко активировать	3	○		▲					●			▲	■	●	
	Легко подключить	5	●	○	○							■		●	▲	
	Не требует настроек	4	●									●	■		▲	
Исполнение	Малый вес	2					▲		▲			●		▲	■	
	Разные цвета корпуса	1			●							●		▲	■	
	Безотказность в работе	4	○				○	○				■		●	▲	
	Не ломается при падении	3							▲			▲		●	■	

● Сильная взаимосвязь - 9

○ Средняя взаимосвязь - 3

▲ Слабая взаимосвязь - 1

Методы процесса обеспечения качества

- 1. Аудит качества.** Независимая экспертная оценка, определяющая, насколько операции проекта соответствуют установленным в рамках проекта или организации правилам, процессам и процедурам. Выполняется внешней организацией.
- 2. Анализ процесса.** Предусматривает выполнение действий, направленных на выявление нуждающихся в улучшении моментов с технической и организационной точек зрения. Выполняется командой проекта.

Методы процесса контроля качества

1. Диаграммы причинно-следственных связей.

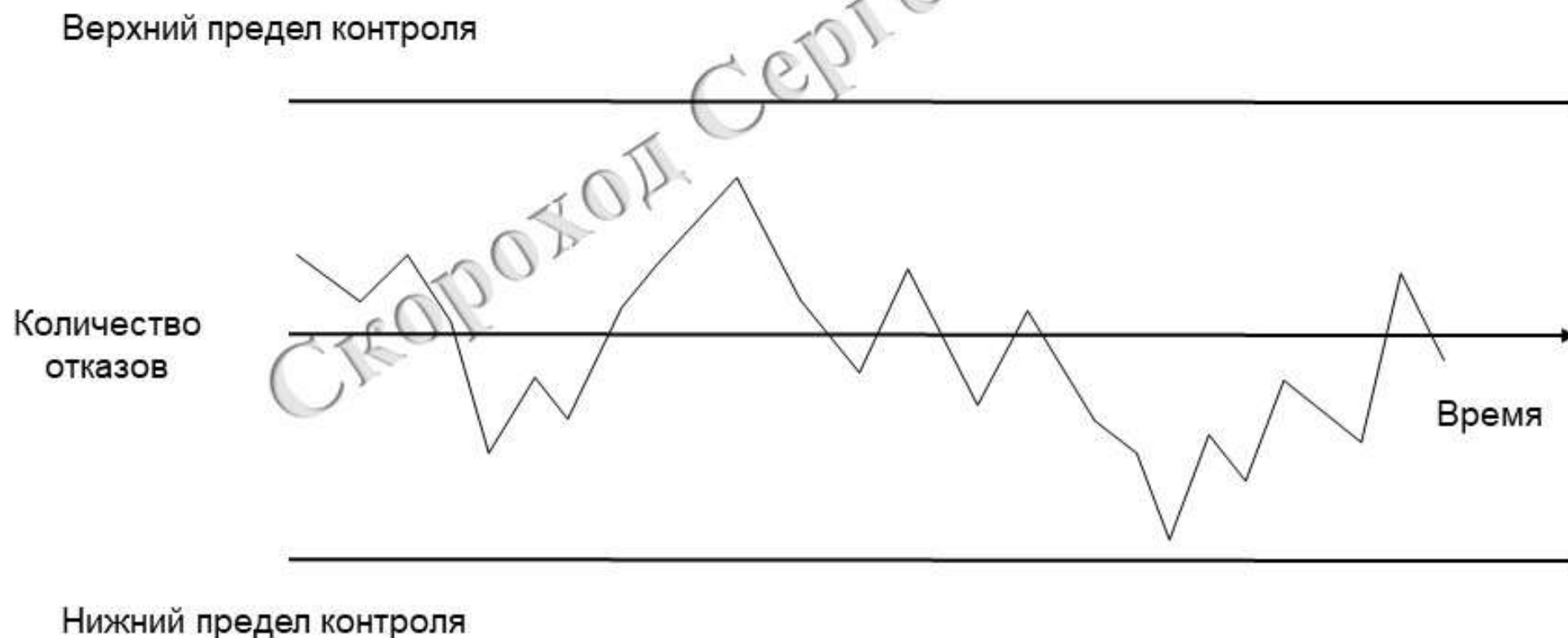
- Изображает связь разнообразных ранее установленных факторов с возможными проблемами или эффектами.



Методы процесса контроля качества

2. Контрольные диаграммы.

- Отображение случаев, когда в процессе возникают разнообразные изменения.
- Визуальное представление о развитии процесса в течение определенного времени.
- Выяснение, как произведенные изменения повлияли на улучшение или ухудшение процесса.



Методы процесса контроля качества

3. Диаграммы зависимостей.

Графическое изображение процесса, в котором используются операции, точки принятия решений и последовательность обработки данных. Дает представление о взаимодействии элементов системы между собой.

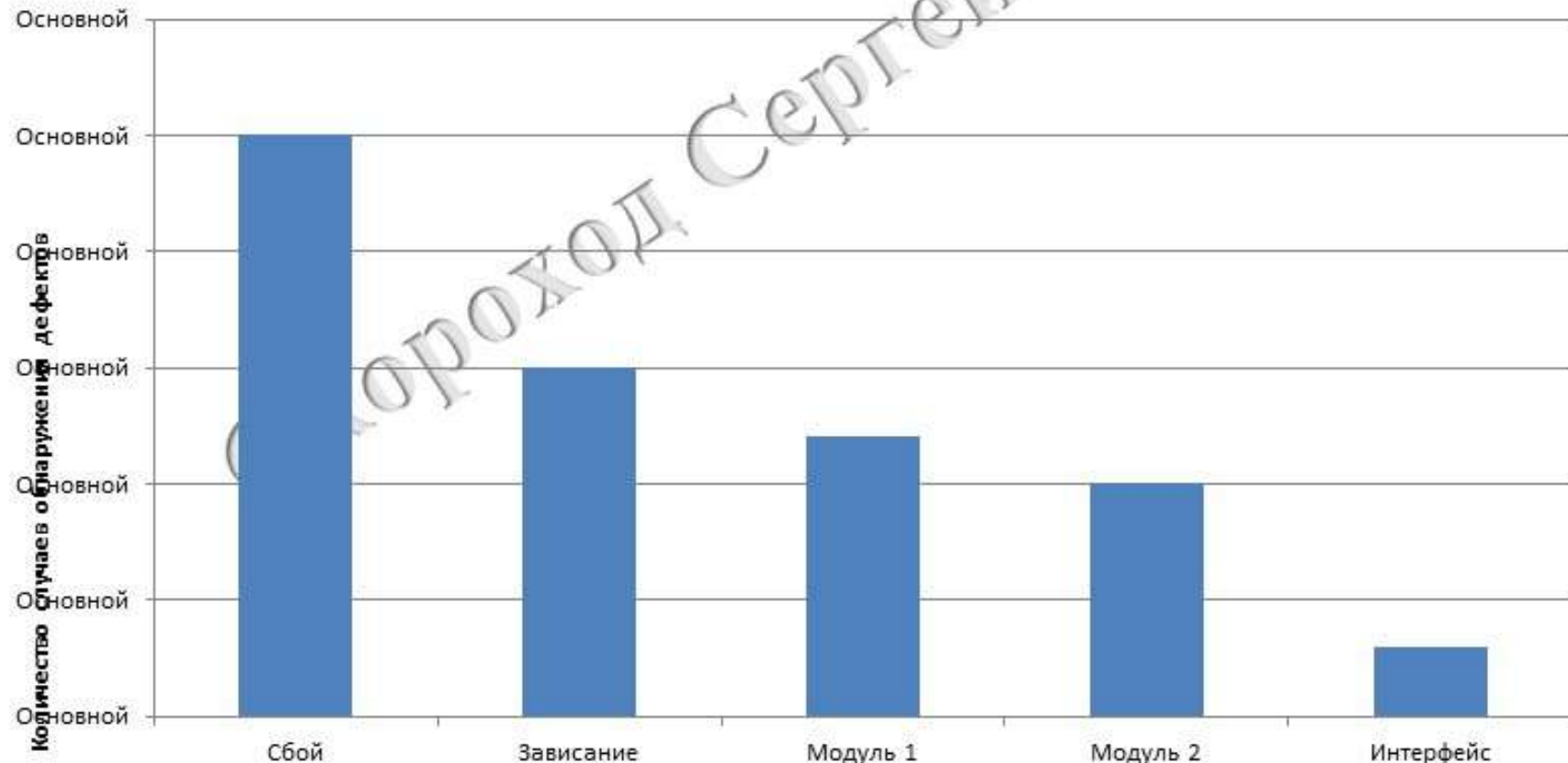


Методы процесса контроля качества

4. Гистограмма - столбиковая диаграмма, отображающая распределение некоторых переменных.

5. Диаграмма Парето - гистограмма, упорядоченная по частоте возникновения. Связана с законом Парето (принцип 80/20): 80 процентов проблем создается 20 процентами причин.

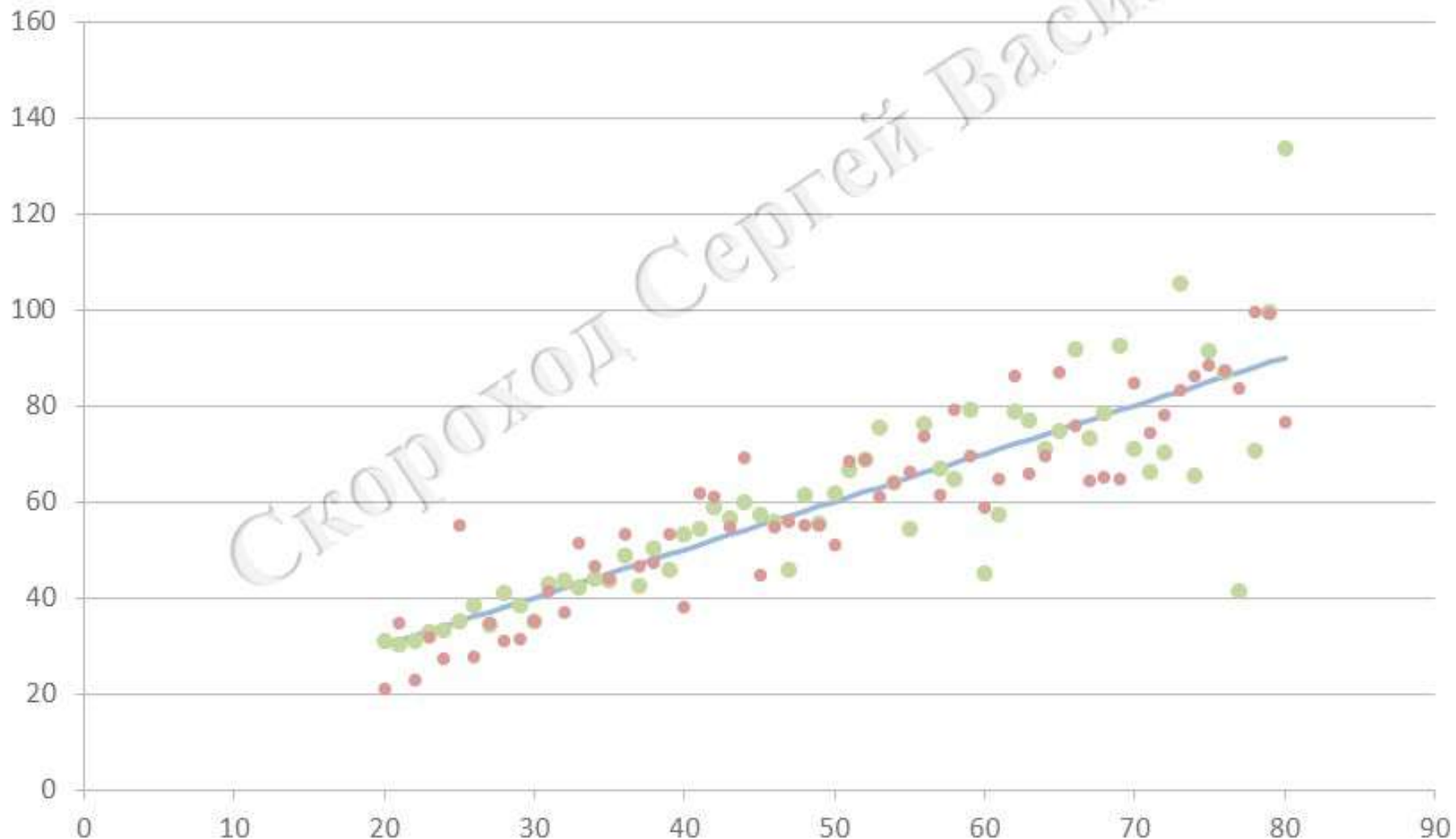
Распределение частоты по причинам



Методы процесса контроля качества

6. Диаграмма разброса. Отображает модель взаимодействия между двумя величинами. При помощи этого инструмента можно проводить изучение и определять возможные зависимости между значениями двух переменных.

Линейный тренд



Методы процесса контроля качества

7. Выборочные оценки.

Предполагают выбор части интересующей совокупности для проверки (например, случайный выбор десяти тестов из списка в сто тестовых примеров для анализа работоспособности программной системы). Представительная выборка обычно может существенно сократить стоимость управления качеством.

8. Инспекция.

Изучение работы продукта с целью определить соответствует ли он стандартам.

9. Проверка исправления дефектов.

Предпринимается с целью убедиться, что дефекты продукта были исправлены

Виды тестирования ПО

Все **виды тестирования программного обеспечения** можно условно разделить на следующие группы:

1. Функциональные
2. Нефункциональные
3. Связанные с изменениями

Функциональные тесты базируются на функциях и особенностях ПО, а также взаимодействии с другими системами. Функциональные виды тестирования рассматривают внешнее поведение системы:

- Функциональное тестирование (Functional testing)
- Тестирование безопасности (Security and Access Control Testing)
- Тестирование взаимодействия (Interoperability Testing)

Нефункциональное тестирование описывает тесты, необходимые для определения характеристик программного обеспечения, которые могут быть измерены различными величинами. В целом, это тестирование того, "Как" система работает:

- Все виды тестирования производительности:
- Тестирование установки (Installation testing)
- Тестирование удобства пользования (Usability Testing)
- Тестирование на отказ и восстановление (Failover and Recovery Testing)
- Конфигурационное тестирование (Configuration Testing)

Функциональное тестирование Functional Testing

- **Функциональные тесты** основываются на функциях, выполняемых системой. Как правило, эти функции описываются в требованиях, функциональных спецификациях или в виде случаев использования системы (**use cases**).
- Тестирование функциональности может проводиться в двух аспектах:
 - Требования
 - бизнес-процессы
- Тестирование в аспекте «**требования**» использует спецификацию функциональных требований к системе как основу для дизайна тестовых случаев (**Test Cases**).
- В этом случае необходимо сделать список того, что будет тестироваться, а что нет, приоритезировать требования на основе рисков, а на основе этого приоритезировать тестовые сценарии (test cases).
- Это позволит не упустить при тестировании наиболее важный функционал.

Функциональное тестирование Functional Testing

- Тестирование в аспекте «**бизнес-процессы**» использует знание самих бизнес-процессов, которые описывают сценарии ежедневного использования системы.
- Здесь тестовые сценарии (**test scripts**) основываются на случаях использования системы.

Преимущества функционального тестирования:

- имитирует фактическое использование системы;

Недостатки функционального тестирования:

- возможность упущения логических ошибок в программном обеспечении;
- вероятность избыточного тестирования.

Тестирование безопасности

Тестирование безопасности - это стратегия тестирования, используемая для проверки безопасности системы, а также для анализа рисков, связанных с обеспечением целостного подхода к защите приложения, атак хакеров, вирусов, несанкционированного доступа к конфиденциальным данным.

Общая стратегия безопасности основывается на трех основных **принципах**:

- конфиденциальность
- целостность
- доступность

Конфиденциальность - это сокрытие определенных ресурсов или информации, т.е. ограничение доступа к ресурсу некоторой категории пользователей.

Существует два основных критерия при определении понятия **целостности**:

Доверие. Ожидается, что ресурс будет изменен только соответствующим способом определенной группой пользователей.

Повреждение и восстановление. В случае когда данные повреждаются или неправильно меняются авторизованным или не авторизованным пользователем, мы должны определить на сколько важной является процедура восстановления данных.

Доступность представляет собой требования о том, что ресурсы должны быть доступны авторизованному пользователю, внутреннему объекту или устройству. Как правило, чем более критичен ресурс тем выше должен быть уровень доступности.

Тестирование взаимодействия

С развитием сетевых технологий и интернета взаимодействие разных систем, сервисов и приложений друг с другом приобрело значительную актуальность, так как любые связанные с этим проблемы могут привести к падению авторитета компании, что как следствие повлечет за собой финансовые потери. Поэтому к тестированию взаимодействия стоит подходить со всей серьезностью.

Тестирование взаимодействия (Interoperability Testing) – это функциональное тестирование, проверяющее способность приложения взаимодействовать с одним и более компонентами или системами и включающее в себя тестирование совместимости и интеграционное тестирование.

Программное обеспечение с хорошими характеристиками взаимодействия может быть легко интегрировано с другими системами, не требуя каких-либо серьезных модификаций. В этом случае, количество изменений и время, требуемое на их выполнение, могут быть использованы для измерения возможности взаимодействия.

Нефункциональное тестирование

Нагрузочное тестирование или **тестирование производительности** - это автоматизированное тестирование, имитирующее работу определенного количества бизнес пользователей на каком-либо общем (разделяемом ими) ресурсе.

Задачей тестирования производительности является определение масштабируемости приложения под нагрузкой, при этом происходит:

- измерение времени выполнения выбранных операций при определенных интенсивностях выполнения этих операций
- определение количества пользователей, одновременно работающих с приложением
- определение границ приемлемой производительности при увеличении нагрузки (при увеличении интенсивности выполнения этих операций)
- исследование производительности на высоких, предельных, стрессовых нагрузках

Нефункциональное тестирование

Стрессовое тестирование позволяет проверить насколько приложение и система в целом работоспособны в условиях стресса и также оценить способность системы к регенерации, т.е. к возвращению к нормальному состоянию после прекращения воздействия стресса. Стрессом может быть повышение интенсивности выполнения операций до очень высоких значений или аварийное изменение конфигурации сервера.

Задачей **объемного тестирования** является получение оценки производительности при увеличении объемов данных в базе данных приложения.

Задачей **тестирования стабильности (надежности)** является проверка работоспособности приложения при длительном (многочасовом) тестировании со средним уровнем нагрузки.

Нефункциональное тестирование

Тестирование установки направленно на проверку успешной инсталляции и настройки, а также обновления или удаления программного обеспечения.

В настоящий момент наиболее распространена установка ПО при помощи инсталляторов (специальных программ, которые сами по себе так же требуют надлежащего тестирования).

В реальных условиях инсталляторов может не быть. В этом случае придется самостоятельно выполнять установку программного обеспечения, используя документацию в виде инструкций или readme файлов, шаг за шагом описывающих все необходимые действия и проверки.