			Метод черного ящика	Мы не видим исходный код. Тестирование основано на требованиях и спецификациях.		
		По доступу кода	Метод белого ящика	— Мы имеем доступ к коду, Имеем возможность анализировать и тестировать код.		
			Метод серого ящика	Комбинация черного и белого ящиков. Мы можем протестировать и код и визуал.		
			Статическое	в основном тестируем требования, техническое задание, код не запускается.		
		По запуску кода на исполнение	Динамическое	анализируя документацию мы можем составлять тест кейсы когда у нас есть работающая система, есть сайт и мы можем провести тест		
				and the American during the second se		
			Модульное теститирование	логически выделенный или изолированный элемент (модуль). проводится разроботчиками и предполагает полный доступ к коду ( по методу белого ящика).еще называют Юнит тестом.		
			Интеграционное тестирование	направлен на проверку корректности взаимодействия нескольких модулей, которые объединяются в единое целое. (виджеты погоды-некая интеграция, которая встраивается на наш сайт, она работает совместно с гео позиции. можно посмотреть как они взаимодействуют, я меняю локацию и погода должна измениться).		
		По уровню детализации приложения	Системное тестирование	Это то тестирование, которое выполняется для всей системы, чтобы проверить работает ли она должным образом после интеграции всех модулей. процесс тестирования системы, на котором проводится не только функциональное тестирование, но и оценка ее характеристик, качество системы, т.е. устойчивость, надежность, безопасность, производительность.		
			Приемочное тестирование	проверяется соответствие системы потребностям, требованиям и бизнес процессам пользователя. может проводить и тестировщик, и заказчик. (например: чтобы пользователь мог зарегистрировать свой аккаунт, авторизироваться, поменять данные, указать номер тел.) нам не обязательно лесть в код, главное результат		
			Ручное (мануальное)	— Мы работаем в ручную. Мы делаем по клику мыши и по движению мыши.		
		По степени автоматизации	Автоматизированное	Здесь мы автоматически пишут функцию, которая сама ищет элементы, забирает элемент, сама заполняет.		
		По принципам работы с	Позитивное тестирование	Например тестирование вход в интернет-банк. Мы вводим валидные данные логин и пароль и заходим в кабинет. Перевод языка. По нажатию по лакаль происходит перевод текста. Соответствие ожидаемому результату.		
		приложением	Негативное тестирование	Перевод языка. Я нажимаю по лакаль и происходит переход на сторонний сайт. Какой-либо отличный от ожидаемого результат.		
Виды тестирования			Дымовое (Смоки) + Санитарное	Тестирование, которое проводится после выпуска каждой сборки. Запуск кода и проверка основных функций бизнеса. Сайт открылся-проверка прошла. Мы можем проверить работу админки.		
		По уровню функционального тестирования ( по запуску тестов)	— Тестирование критического пути (ЮАТ) -	Проверка функциональности, которая используется обычными пользователями во время их повседневной деятельности. Например: в приложении банка открыть счет, сделать перевод.		
			Расширенное тестирование	Направлено на исследование всей заявленной в требованиях функциональности. Похоже на приемочное.		
		В зависимости от исполнителей	Альфа тесты	Закрытое тестирование внутри команды		
		D Subnominoctin of nonominitorion	Бета тестирование	Когда приглашаем сторонних пользоватей		
				Модульное тестирование		
				Дымовое (Смоки) + Санитарное		
				Регрессионное тестирование- оно проводится для того, чтобы убедиться, что добавление нового кода, какое-либо улучшение или исправление какого-либо бага не нарушило существующую функциональность или не вызывает какого-либо вида нестабильности и также работает с нашими старыми требованиями. Проверка нашего старого функционала, что он работает также после внедрения нового кода. Если это новый сайт, то регресс не делается.		
			Функциональное тестирование	— Альфа/ Бета тестирование		
				Системное тестирование		
				Интеграционное тестирование Тестирование критического лути (ЮАТ)		
				— Тестирование критического пути (ЮАТ)	Выполнение тестов на реальных	
				Е2Е (ент ту ент тесты)	конфигурациях. Когда создается копия сайта с боевой версией базы и сервера и на ней проводятся тесты.	
					Определить стабильность и потребление каких-либо ресурсов (затраты памяти, скорость загрузки страниц) в условиях	
				Производительность (Перфоманс)	различных сценариях использования и нагрузок. Например: я хочу посмотреть насколько производительно работает мой	
	\	В зависимости от целей			сайт и сервер, когда я захожу 1 пользователем, когда 5 пользователей, 10.	
		тестирования		Нагрузочное тестирование	Как наша система будет работать сверх лимита и когда она даст сбой	
				— Стресс тест	Мы проверяем стабильность работы на	
					максимальной нагрузке. Тестирование направлено на проверку	
				<ul> <li>Инсталяционное тестирование</li> </ul>	успешной установки и настройки обновлений и удаления нашего приложения.  Предназначено для мобильной версии	
					тестов. (скачать приложение, войти, обновить, удалить)	
					Шрифты	7
				— Тестирование интерфейсов UI	Размеры элементов	<ul><li>визуал/проверка дизайна</li></ul>
			Нефункциональное тестирование		Цвета Кнопки	
				VIIOTOTINO MOROTI CORRENTE LIV	Удобно ли пользоваться нашим	_
				Удобство использования UX	приложением (удобно, привлекательно)	
				<b>—</b> Локализация	Проверка нашей адаптации нашего ПО для определенной аудитории в соответствии с ее культурными особенностями ( написание текста, даты, цвета)	
				Тестирование надежности	Целью тестирования является проверка нашей работоспособности приложения при длительном тестировании	
				Тестирование безопасности	Это такая стратегия тестирования, которая используется для проверки самой безопасности системы. Анализ рисков, атака хакеров, вирусов. (скрытие пароля ****, проверка файлов на вирусы)	
				Востановление	Это когда система дает сбой и нужно посмотреть как она востанавливается	
				Кроссбраузерное		