

Google
Sheets

Summary Session 1



Преподаватель

Портрет

Имя Фамилия

Текущая должность

Количество лет опыта

Какой у Вас опыт - ключевые кейсы

Самые яркие проекты

Дополнительная информация по вашему усмотрению

Корпоративный e-mail

Социальные сети (по желанию)

Важно

- 


Камера должна быть включена на протяжении всего занятия
- 

В течение занятия вопросы задавать в чате или когда преподаватель спрашивает, есть ли у Вас вопросы
- 


Вести себя уважительно и этично по отношению к остальным участникам занятия
- 

Организационные вопросы по обучению решаются с кураторами, а не на тематических занятиях
- 

Во время занятия будут интерактивные задания, будьте готовы включить камеру или демонстрацию экрана по просьбе преподавателя



ПОВТОРЕНИЕ ИЗУЧЕННОГО



Повторение

Соотнесите название из 1 столбца с его определением из 2, ответы отправьте в чат:

1. Тестирование ПО	A. Это источник знаний о том, какое поведение системы является ожидаемым. Требования могут быть как задокументированы или незадокументированны.
2. Баг	B. Это проверка соответствия между реальным и ожидаемым поведением программы, осуществляемая на конечном наборе тестов, выбранном определенным образом.
3. Требования	C. Это отклонение фактического результата от ожидаемого.



Тестирование программного обеспечения

Это проверка соответствия между реальным и ожидаемым поведением программы, осуществляемая на конечном наборе тестов, выбранном определенным образом. Тестирование включает управление тестами, проектирование, выполнение и анализ результатов.

Полезно знать



Реальное поведение (actual behaviour, actual result)

Это то, как по факту функционирует программное обеспечение.

Ожидаемое поведение (expected behaviour, expected result)

Это то, как должно функционировать программное обеспечение



Требования (requirements)

Это источник знаний о том, какое поведение системы является ожидаемым. Требования могут быть как задокументированы или незадокументированны.

Примеры источников задокументированных требований



Технические задания



Спецификации



Документы с бизнес-требованиями



Модели и диаграммы



Майндмэпы



Контракты и соглашения и т.д.

Требования

Ожидание

Все требования были описаны в каком-либо документе, опираясь на который, можно однозначно понимать ожидаемое поведение разрабатываемой системы.

Реальность

Написание и чтение документов требует временных и ресурсных затрат → требования есть, но они не зафиксированы письменно.



Цель тестирования

Это верификация, валидация и обнаружение ошибок для их исправления.

Поиск дефектов - это одна из целей тестирования, но не его суть.

Различие между верификацией и валидацией



Верификация

Это проверка на соответствие требованиям и корректную работу функционала. Отвечает на вопрос: правильно ли создается программное обеспечение?

Валидация

Это проверка того, соответствует ли продукт ожиданиям клиента.



Баг (ошибка)

Это отклонение фактического результата от ожидаемого.

Экспресс-опрос

?

Какие принципы тестирования вы помните?

Принцип тестирования 1



Тестирование не доказывает отсутствие багов, оно лишь снижает вероятность их наличия.

Принцип тестирования 2



Исчерпывающее тестирование невозможно — необходимо использовать анализ рисков и приоритизацию.

Принцип тестирования 3



Раннее тестирование помогает снизить затраты.

Принцип тестирования 4



Кластеризация дефектов — большинство ошибок сосредоточены в небольшом количестве модулей или компонентов системы.

Принцип тестирования 5



Парадокс пестицида — повторяющиеся тесты со временем теряют свою эффективность.

Принцип тестирования 6



Тестирование зависит от контекста — подходы к тестированию различаются в зависимости от проекта.

Особенности, влияющие на выбор стратегии тестирования



Тип продукта

Цели тестирования

Жизненный цикл и
срок проекта

Принцип тестирования 7



Заблуждение об отсутствии ошибок — важнее управлять критичностью и исправлять ошибки быстро, а не пытаться исключить все баги.

Экспресс-опрос

?

Какие виды тестирования вы помните?

Виды тестирования

Тестирование

Нефункциональное тестирование

Функциональное тестирование

Виды тестирования

Нефункциональное тестирование

Тестирование
производительности

Инсталляционное
тестирование

Usability тестирование

Тестирование
локализации и
интернационализации

Конфигурационное
тестирование

Нагрузочное тестирование
(Load Testing)

Стресс-тестирование
(Stress Testing)

Тестирование стабильности
(Stability Testing)

Тестирование
безопасности

Тестирование на отказ
и восстановление

Тестирование
доступности

Тестирование
документации

Виды тестирования

Функциональное тестирование

По уровню тестирования				Связанное с изменениями			По знанию системы			По критерию запуска программы		Относительно позитивности сценариев		По типу интерфейса		По степени автоматизации		По исполнению сценария			
Модульное	Интеграционное	Системное	Приемочное	Smoke-тестирование	Регрессионное тестирование	Sanity-тестирование	Black box	White box	Grey box	Статическое тестирование	Динамическое тестирование	Позитивные тесты	Негативные тесты	API-тестирование	CLI-тестирование	GUI-тестирование	Ручное тестирование	Автоматизированное тестирование	Сценарное тестирование	Исследовательское тестирование	Ad-hoc тестирование



ВОПРОСЫ



Экспресс-опрос

?

Что такое тестовые артефакты?



Тестовые артефакты

Это документы и материалы, которые создаются и используются в процессе тестирования, чтобы планировать, контролировать и отслеживать выполнение тестов.

Основная цель тестовых артефактов



Обеспечить систематичность, полноту и прозрачность процесса тестирования, а также гарантировать, что проверка функциональности продукта проводится качественно и последовательно.

Зачем нужны тестовые артефакты



Для полноты покрытия



Для исключения избыточности



Для контроля и анализа



Для упрощения процесса тестирования

Экспресс-опрос

?

Что такое тест-кейсы и чек-листы?



Тест-кейсы

Это пошаговые инструкции, описывающие, как проверять конкретные функции продукта. Каждый тест-кейс содержит ожидаемый результат, чтобы можно было легко определить, прошел ли тест.




Чек-листы

Это списки проверок, которые содержат ключевые элементы и функции, которые нужно протестировать. Чек-листы обычно короче и менее детализированы, чем тест-кейсы, и позволяют быстро охватить основные области приложения.



ISTQB (International Software Testing Qualifications Board)

Это международная организация, которая занимается сертификацией профессионалов в области тестирования программного обеспечения. Ею установлен набор условий и переменных, при которых тестировщик определяет, выполнено ли требование к приложению.

**Поля, заполняемые
до момента 
тестирования**

ID: Уникальный идентификатор тест-кейса



Цель

Быстрое и однозначное нахождение и отслеживание тест-кейса.

Заполнение

Присваивается каждому тест-кейсу заранее и используется для его идентификации в тестовой документации.

Title/Purpose/Description: Краткое название или описание тест-кейса

Цель

Дать общее представление о том, что тестируется, и почему этот тест-кейс важен.

Заполнение

Заранее указывается, какая функция или аспект продукта тестируется. Например, "Проверка успешного входа в систему при корректных данных".

Instruction: Пошаговые инструкции для выполнения теста

Цель

Указать точные действия, которые тестировщик должен выполнить, чтобы воспроизвести сценарий

Заполнение

Заранее прописываются все шаги теста в порядке их выполнения. Пример:

- Открыть страницу входа.
- Ввести корректный логин.
- Ввести корректный пароль.
- Нажать кнопку "Войти".

Expected Result: Ожидаемый результат выполнения теста

Цель

Четко описать, каким должен быть результат, если функция работает правильно.

Заполнение

Заранее указывается, что должно произойти при правильной работе системы. Например, "Пользователь успешно заходит в систему и видит страницу профиля".

**Поля, заполняемые в
рамках тестирования**

Actual Result: Фактический результат выполнения теста

Цель

Зафиксировать, что произошло во время тестирования, и сравнить с ожидаемым результатом.

Заполнение

Заполняется во время тестирования. Тестировщик указывает, что произошло на самом деле после выполнения всех шагов. Например, "Пользователь видит сообщение об ошибке при входе".

Status (Pass/Fail/etc): Статус тест-кейса

Цель

Определить, прошел ли тест-кейс успешно или выявил проблему.

Заполнение

Заполняется после тестирования.

- "Pass" (если фактический результат совпадает с ожидаемым)
- "Fail" (если результаты не совпадают)
- "Blocked" (если выполнение теста невозможно из-за внешних факторов)
- "In Progress" (если тест не завершен)

Пример использования полей

ID	Title/Purpose/Description	Instruction	Expected Result	Actual Result	Status
TC001	Проверка успешного входа в систему	Открыть страницу входа. Ввести корректные логин и пароль. Нажать "Войти".	Пользователь видит страницу профиля	Пользователь видит страницу профиля	Pass

Повторение

1. Какие из перечисленных атрибутов имеются у тест-кейса?

- a. Заголовок
- b. Шаги
- c. Фактический результат
- d. Ожидаемый результат

2. Что означает статус `skipped` при прохождении тест-кейса?

- a. Тест-кейс успешно пройден.
- b. Не все шаги тест-кейса можно выполнить из-за имеющегося ранее обнаруженного дефекта.
- c. Тест-кейс невозможно выполнить из-за фактического результата, отличающегося от ожидаемого.
- d. Проверка пропускается из-за нехватки времени.

Правила составления тест-кейсов 1



Test case is 100% derived from requirement.

Тест-кейс на 100% определяется требованием (функция требования).

Правила составления тест-кейсов 2



Might fail or pass in just one place.
Тест-кейс может провалиться или пройти успешно.

Правила составления тест-кейсов 3



he smallest possible action in software testing.

Тест-кейс — минимальное действие в тестировании (нельзя разделить на 2 части).

Правила составления тест-кейсов 4



The lowest possible level document in software testing.
Самый низкий уровень документа в тестировании.

Статусы тест-кейса

Passed	Failed	Blocked
Проверка пройдена успешно, багов не найдено	Найден один или более багов	Невозможно проверить, т.к. один из багов блокирует текущую проверку
In Progress	Not Run	Skipped
В работе	Еще не проверено	Проверяться не будет по какой-то причине

Экспресс-опрос

?

Что такое Happy Path?



Happy Path

Это тест, рассчитанный на успешное прохождение. Это сценарий тестирования, который проверяет, как система выполняет основную функцию при корректных вводных данных и идеальных условиях.

Важно!

Если HAPPY PATH не прошел, то нет смысла выполнять остальные тесты.



Порядок выполнения тестов



Важно для тест-кейсов



Подробное описание действий и результатов, чтобы выполнение не зависело от исполнителя (автоматическое выполнение)



Каждому требованию должны соответствовать его тест-кейсы (один или больше).



Тест-кейс не может существовать без соответствующего ему требования.



Test Data: минимальные отклонения от правильного значения
(password: qwerty12345 invalid password: qwerty02345).

Чего не должно быть в тест-кейсе

- ✗ Зависимостей от других тест-кейсов
- ✗ Нечеткой формулировки шагов или ожидаемого результата
- ✗ Отсутствия необходимой для прохождения тест-кейса информации
- ✗ Излишней детализации

Пример описания тест-кейсов

Test Case ID	Test Case Description	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail
TU01	Check Customer Login with valid Data	<ol style="list-style-type: none"> Go to site http://demo.guru99.com Enter UserId Enter Password Click Submit 	Userid = guru99 Password = pass99	User should Login into an application	As Expected	Pass
TU02	Check Customer Login with invalid Data	<ol style="list-style-type: none"> Go to site http://demo.guru99.com Enter UserId Enter Password Click Submit 	Userid = guru99 Password = glass99	User should not Login into an application	As Expected	Pass

Пример описания тест-кейсов

Test Scenario ID	Login-1	Test Case ID	Login-1A				
Test Case Description	Login – Positive test case	Test Priority	High				
Pre-Requisite	A valid user account	Post-Requisite	NA				
Test Execution Steps:							
S.No	Action	Inputs	Expected Output	Actual Output	Test Browser	Test Result	Test Comments
1	Launch application	https://www.facebook.com/	Facebook home	Facebook home	IE -11	Pass	[Priya 10/17/2017 11:44 AM]: Launch successful
2	Enter correct Email & Password and hit login button	Email id : test@xyz.com Password: *****	Login success	Login success	IE-11	Pass	[Priya 10/17/2017 11:45 AM]: Login successful

Возможные ошибки при написании тест кейсов



Абстрактное название тест-кейса



Некликабельные ссылки



Лишние детали в тест-кейсе



Недостаток деталей для проведения тест-кейса

Экспресс-опрос



Что такое тест-сьют?



Тест-сьют (Test Suite)

Это набор тестовых сценариев или тестовых процедур, объединенных для выполнения в одном тестовом прогоне. Он создается, чтобы организовать и структурировать тестирование, сгруппировав тест-кейсы, которые проверяют определённый функционал, модуль, компонент или выполняют конкретную задачу.



Чек-лист

Это список проверок или элементов, которые нужно протестировать для конкретной функции или части системы. Чек-лист используется для быстрой проверки, что все основные функции работают, и охватывает ключевые элементы тестирования, но без подробных шагов и ожидаемых результатов, как в тест-кейсе.

Основные отличия чек-листа от тест-кейса

	Тест-кейс	Чек-лист
Уровень детализации	Включает подробные шаги выполнения, ожидаемые результаты и критерии прохождения	Содержит только пункты или краткие описания проверок
Скорость выполнения	Предназначены для более детальной проверки и занимают больше времени на выполнение.	Позволяет быстрее пройти по основным аспектам тестирования, так как он не требует детального выполнения каждого шага
Цель использования	Используются для детального тестирования, когда важно фиксировать каждый шаг и проверять точное соответствие требованиям.	Используются для простых проверок или в ситуациях, когда не требуется глубокое тестирование.

Пример оформления чек- листа

Field Testing Checklist - (Sample Checklist)

Sr.	Check Point / Defect Statement	Check Mark (✓) the Appropriate Column	
		Yes	N/A
01	Were all codes validated?		
02	Can fields be updated properly?		
03	Is the field large enough for collecting the totals?		
04	Is the field adequately described in the program?		
05	Can the field be initialized properly?		
06	Do all references to the field use the proper field name?		
07	If the field's contents are restricted, are those restrictions validated?		
08	Were rules established for identifying and processing invalid field data? (If not, this data must be developed for the error-handling transaction type. If so, test conditions must be prepared to validate the specification processing for invalid field data.)		
09	Is a wide range of typical valid processing values included in the test conditions?		
10	For numerical fields, have the upper and lower values been tested?		
11	For numerical fields, has a zero value been tested?		
12	For numerical fields, has a negative test condition been prepared?		
13	For alphabetical fields, has a blank condition been prepared?		
14	For an alphabetic or alphanumeric field, has a test condition longer than the field length been prepared to check truncation processing?		
15	Were all valid conditions tested on the basis of the data dictionary printout?		
16	Were systems specifications reviewed to determine whether all valid conditions are tested?		
17	Do owners of data elements know whether all valid conditions are tested?		
18	Have owners of data elements reported their results?		

Пример оформления чек-листа

Tester's Name:					Comments
	XP	Vista	7	8	
Installing the Application					
Install the application	Pass				
Running the Application					
Run the application from the installer	Fail				Bug #s
Run the application from the shortcut	Warning				
Restart and verify application launches at login with "Launch on Startup" enabled	Blocked				
Restart and verify application does not launch at login with "Launch on Startup" disabled	Skipped				

Экспресс-опрос



Что такое тест-ран?



Тест-пан (test run)

Это процесс выполнения набора тест-кейсов или тест-сюэтов в рамках одного цикла тестирования. Используется для проверки определенной функциональности или версии продукта и для фиксации результатов выполнения тестов.

Основные аспекты тест-рана



Цель тест-рана



Выбор тест-кейсов



Запись результатов



Отчетность и анализ

Важность тест-плана

- Помогает структурировать и организовать процесс тестирования, координировать команду и контролировать выполнение тестовых задач.
- Служит основой для управления тестированием и позволяет достигнуть согласованности в целях и подходах к тестированию.
- Помогает заинтересованным сторонам понять объем, сроки и подходы к тестированию, а также способствует своевременному выявлению и решению возможных проблем.

Экспресс-опрос










Что такое тест-план?



Тест-план (test plan)

Это документ, который определяет стратегию, цели, ресурсы и сроки, необходимые для проведения тестирования. Важный элемент в процессе тестирования, так как служит руководством для команды и устанавливает структуру и порядок выполнения тестов.

Основные элементы тест-плана

-  Цели и задачи тестирования
-  Объем и границы тестирования
-  Методология и подход к тестированию
-  Ресурсы и роли
-  Критерии завершения тестирования
-  График и сроки тестирования
-  Риски и способы их минимизации

Пример использования тест-рана

Команда выпускает обновление приложения и выполняет тест-ран для регрессионного тестирования, чтобы убедиться, что старый функционал работает корректно после внесённых изменений.


В тест-ран включены тест-кейсы для проверки ключевых функций, таких как вход в систему, оформление заказа и работа с профилем пользователя.

После выполнения тест-рана команда анализирует результаты: например, из 50 тестов 45 прошли успешно, 3 показали сбои, и 2 теста были заблокированы. На основании этих данных принимается решение об исправлении ошибок или дополнительном тестировании.




ВОПРОСЫ





РАЗБОР ДОМАШНЕГО ЗАДАНИЯ



Домашнее задание 1

Составить набор проверок:

1. Познакомьтесь с [компонентом чекбоксов](#).
2. Фиксированных требований к компоненту нет. Изучите его работу и опирайтесь на предполагаемое поведение компонента. Например, допускается выбор нескольких значений при использовании чекбоксов.
3. Составьте набор проверок для компонента чекбоксов в Excel-таблице (или таблице Google Sheets)
4. Проведите тестирование по чит-листу и запишите результаты в Excel-таблице (или таблице Google Sheets).

Домашнее задание 2

Составить тест-кейс:

1. Для <https://www.saucedemo.com/> необходимо составить тест-кейс в [Qase](#) для реализации успешной покупки одного товара.
2. Выгрузить составленный тест-кейс в формате pdf и прикрепить в LMS.



ВОПРОСЫ





ЗАДАНИЕ



Задание 1

Составление чек-листа для поля «Дата Рождения»

Требования:

- Формат ввода даты — **ДД.ММ.ГГГГ**, но также является валидным формат **ДД/ММ/ГГГГ**.
- Не во всех месяцах есть 31 день (и даже 30).
- Не забудьте учесть особенные даты високосных годов.
- Максимальная валидная дата равна сегодняшнему дню.
- Самая минимальная дата — 01.01.1900.



Задание



1. Составьте чит-лист для проверки поля «Дата Рождения»:
2. Включите в чит-лист ровно 8 позитивных (валидных) проверок. Например, «Первый день года. Пример значения 01.01.1978».
3. Включите в чит-лист ровно 8 негативных(невалидных) проверок. Например, «Дата в формате ДД.ММ.ГГ».
4. Опирайтесь на имеющиеся требования и привлеките свою фантазию, личный опыт взаимодействия с полями для ввода данных и здравый смысл.

Пример составленного чек-листа

Валидные проверки поля Дата Рождения:

1. Своя дата рождения 04.08.1985 пройдена
2. Минимальная дата 01.01.1900 пройдена
3. Максимальная дата 24.11.2024 пройдена
4. Своя дата рождения 04/08/1985 не пройдена (выдает ошибку)
5. Последний день сентября 30.09.2024 пройдена
6. Високосный год 29.02.2024 пройдена
7. Первый день года 01.01.2024 пройдена
8. Любой ввод в таком формате ДД/ММ/ГГГГ не пройдена(выдает ошибку)

Невалидные проверки поля Дата Рождения:

1. Дата в формате 04.08.85 пройдена
2. Дата в формате 04-08-85 пройдена
3. Дата меньше минимальной даты 00.08.85 пройдена
4. Дата больше максимальной даты 32.08.85 пройдена
5. Ввод нулей в месяц рождения 04.00.85 пройдена
6. Нулевой год рождения 04.08.00 пройдена
7. Формат ввода 04/08/85 пройдена
8. Ввод нулей 00.00.00 пройдена

Задание 2

Составление тест-кейсов для тестирования
функциональности сортировки товаров.





ВОПРОСЫ



Заключение

