

Практическое применение методов «черного ящика» при разработке тестовых наборов

Многие знают что такое тест дизайн, но не все умеют его применять. Чтобы немного прояснить ситуацию, мы решили предложить Вашему вниманию последовательный подход к разработке [тестовых случаев \(тест кейсов\)](#), используя самые простейшие [техники тест дизайна](#):

- **Эквивалентное Разделение (Equivalence Partitioning)**, далее в тексте - **EP**
- **Анализ Граничных Значений (Boundary Value Analysis)**, далее в тексте - **BVA**
- **Предугадывание ошибки (Error Guessing)**, далее в тексте - **EG**
- **Причина / Следствие (Cause/Effect)**, далее в тексте - **CE**

План разработки тест кейсов предлагается следующий:

1. Анализ требований.
2. Определение набора тестовых данных на основании **EP, BVA, EG**.
3. Разработка шаблона теста на основании **CE**.
4. Написание тест кейсов на основании первоначальных требований, тестовых данных и шагов теста.

Далее на примере, рассмотрим предложенный подход.

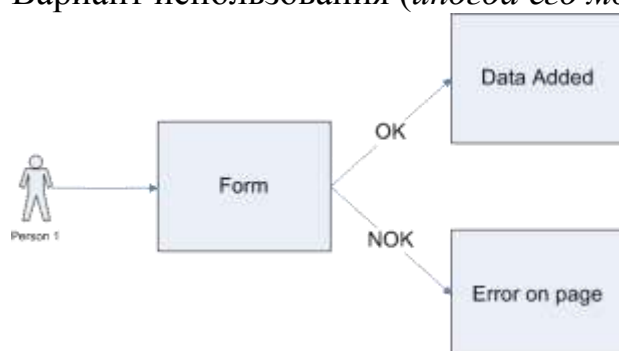
Пример:

Протестировать функциональность формы приема заявок, требования к которой предоставлены в следующей таблице:

Элемент	Тип элемента	Требования
Тип обращения	combobox	Набор данных: 1. Консультация 2. Проведение тестирования 3. Размещение рекламы 4. Ошибка на сайте * - на процесс выполнения операции приема заявок не влияет.
Контактное лицо	editbox	1. Обязательное для заполнения 2. Максимально 25 символов 3. Использование цифр и спец символов не допускается
Контактный телефон	editbox	1. Обязательное для заполнения 2. Допустимые символы "+" и цифры 3. "+" можно использовать только в начале номера 4. Допустимые форматы:

		<ul style="list-style-type: none"> ○ начинается с плюса - 11-15 цифр +31612361264 +375291438884 ○ без плюса - 5-10 цифр, например: 0613261264 2925167
Сообщение	text area	1. Обязательное для заполнения 2. Максимальная длина 1024 символа
Отправить	button	Состояние: 1. По умолчанию - не активна (Disabled) 2. После заполнения обязательных полей становится активна (Enabled) Действия после нажатия 1. Если введенные данные корректны - отправка сообщения 2. Если введенные данные НЕ корректны - валидационное сообщение

Вариант использования (*иногда его может и не быть*):



1. Анализ требований

Читаем, анализируем требования и выделяем для себя следующие нюансы:

- какие из полей обязательные для заполнения?
- имеют ли поля ограничения по длине или по размерности (границы)?
- какие из полей имеют специальные форматы?

2. Определение набора тестовых данных

Отталкиваясь от требований к полям, используя техники тест дизайна начинаем определение набора тестовых данных:

- в зависимости от того обязательное поле или нет, определим какие поля необходимо проверить на пустое значение, так как оно может вызывать ошибку (В результирующей таблице **оранжевый** цвет)

- т.к. исчерпывающее тестирование не представляется возможным *из-за огромного числа всевозможных комбинаций значений*, в первую очередь необходимо определить **минимальный набор данных**. Это можно сделать используя такие **техники**, как **EP** и **BVA**. (В результирующей таблице **голубой** цвет)
- На форме присутствует поле, имеющее составной тип (цифры используются совместно с символами), обладает специальным форматом данных и поэтому выделение тестовых данных для него - это достаточно трудоемкая задача. *В пределах данной статьи ограничимся только простой проверкой форматов и основных требований описанных в форме приема заявок.*
- По завершению генерации данных используя стандартные техники, можно добавить некоторое количество значений на основании личного опыта (**техника EG**) - это будет использование спец. символов, очень длинных строк, разных форматов данных, регистров в строках (Upper, Lower, Mixed cases), отрицательные и нулевые значения, кейворды Null - NaN - Infinity и т.д. Сюда можно включить все, что вы полагаете может вывести приложение из строя (В результирующей таблице **фиолетовый** цвет)

Примечание:

Отметим, что количество тестовых данных после окончательной генерации будет достаточно большим, даже при использовании специальных техник тест дизайна. Поэтому ограничимся лишь несколькими значениями для каждого поля, так как цель данной статьи показать именно **процесс создания тест кейсов**, а не процесс получения конкретных тестовых данных.

2.1 Выбор тестовых данных для каждого отдельно взятого поля

- Поле **Тип обращения**. Так как все данные входят в 1 класс эквивалентности, то есть не изменяют сам процесс выполнения приема заявки, берем любую (1-ю) позицию в листе с ожидаемым результатом ОК. Но т.к. реализовано поле как лист, имеет также смысл рассмотреть и граничные условия (техника BVA), т.е. берем первый и последний элементы. Итого: 1-я и последняя позиции в листе. Ожидаемый результат при использовании - ОК.
- Поле **Контактное лицо**. Это обязательное поле размером от 1 до 25 символов (включая границы). Проверка на обязательность добавляет к тестовым данным пустое значение. Проведем анализ граничных условий (BVA), получим набор: 0, 1, 2, 24, 25 и 26 символов. Пустое значение (0 символов) уже было добавлено при анализе обязательности поля для ввода, поэтому при BVA мы не будем добавлять его еще раз. (если его добавить второй раз, произойдет дублирование тестовых данных, которое не приведет к нахождению новых дефектов, а значит повторное добавление в домен не имеет смысла). В связи с тем, что значения 2 и 24 символа являются, с нашей точки зрения, не критичными, их можно не добавлять. В итоге получаем, что

минимальный набор данных для тестирования поля - это строки 1 и 25 - ОК, и 0 (пустое значение), 26 символов - NOK.

- поле **Контактный телефон** состоит из нескольких частей: код страны, код оператора, номер телефон (который может быть составной и разделенный дефисами). Для определения правильного набора тестовых данных необходимо рассматривать каждую составную часть по-отдельности. Применяя BVA и EP, получим:
 - для **номеров с плюсом**
По BVA получим номера с 10, 11, 12 и 14, 15, 16 цифрами, где 10 и 16 - NOK, а 11, 12, 14, 15 - ОК
Рассматривая полученные данные с позиции EP выделим, что 11, 12, 14, 15 входят в один класс эквивалентности. Поэтому при тестировании мы можем использовать любое из них, но так как 11 и 15 - это границы интервала, то на наш взгляд их пропускать нельзя. Следовательно мы можем уменьшить набор значений до двух, исключив 12 и 14, а оставив 11 и 15 для проверки граничных условий.
Итого имеем:
11 и 15 цифр - ОК, (+12345678901, +123456789012345)
10 и 16 цифр - NOK; (+1234567890, +1234567890123456)
 - для **номеров без плюса:**
По BVA получим номера с 4, 5, 6 и 9, 10, 11 цифрами. Действуя аналогично примеру для номеров телефонов с плюсом, исключим значения 6 и 9, оставив 5 и 10.
Итого имеем:
5 и 10 цифр - ОК, (12345, 1234567890)
4 и 11 цифр - NOK; (1234, 12345678901)
- поле **Сообщение**. подбор данных проводим по аналогии с полем Контактное лицо. На выходе получаем значения: строки 1 и 1024 - ОК, и 1025 символов - NOK.

Результирующая таблица данных, для использования при последующем составлении тест кейсов

е	Пол	О К/NOK	Значение	Комментарий
Тип обращени я	К	О	Консультация	первый в списке
		О	Ошибка на сайте	последний в списке
		N ОК		
Кон тактное	К	О	йцукенгшщзйцуке ншщзйцу	25 символов нижний

лицо			регистр
		a	1 символ
		ЙЦУКЕНГШЩЗФЫВАПРО ЛДЖЯЧСМИ	25 символов ВЕРХНИ Й регистр
		ЙЦУКЕНГШЩЗфывапролд жЯЧСМИ	25 символов СМеШаН Ный регистр
	OK N		пустое значен ие
		йцукенгшщзйцукенгшщзйцу кей	длина больше максимальной(26 символов
		@#\$\$%^&;.?,> \№"!()_{}<~	спец. символы (ASCII)
		1234567890123456789012345	только цифры
		adsadasdasdas dasdasd asasdsads(...)sas	очень длинная строка (~1Mb)
Кон тактный телефон	K O	+12345678901	с плюсом - минимальная длина
		+123456789012345	с плюсом - максимальная длина
		12345	без плюса - минимальная длина
		1234567890	без плюса - максимальная длина
	OK N		пустое значение
		+1234567890	с плюсом - < минимальной длины
		+1234567890123456	с плюсом - > максимальной длины
		1234	без плюса - < минимальной длины

		12345678901	без плюса - > максимальной длины
		+YYYYXXxyuuxxzz	с плюсом - буквы вместо цифр
		уууxxxxzz	без плюса - буквы вместо цифр
		+###-\$\$\$-%^-&^-&!	спец. символы (ASCII)
		1232312323123213231232(...) 99	очень длинная строка (~1Mb)
Сообщение	К О	йцуйцуйц(...)йцу	максимальная длина (1024 символа)
			пустое значение
	ОК N	йцуйцуйц(...)йцуц	длина больше максимальной (1025 символов)
		adsadasdasdasdasdasd asasdsads(...)sas	очень длинная строка (~1Mb)
		@###\$\$\$%^&^&	только спец. символы (ASCII)

3. Разрабатываем шаблон теста

На основании **техники СЕ** и, по возможности, имеющихся вариантов использования (Use case) создадим шаблон планируемого теста. Данный документ будет представлять собой шаги и ожидаемые результаты теста, но без конкретных данных, которые подставляются на следующем этапе разработки тест кейсов.

Пример шаблона тест кейса

Действие	Ожидаемый результат
1. Открываем форму отправки сообщения	<ul style="list-style-type: none"> Форма открыта Все поля по умолчанию пусты Обязательные поля помечены - * Кнопка "Отправить" не активна
2. Заполняем поля формы:	<ul style="list-style-type: none"> Поля заполнены Кнопка "Отправить" - активна (Enabled)

<ul style="list-style-type: none"> • Тип обращения • Контактное лицо • Контактный телефон • Сообщение 	
<p>3. Нажимаем кнопку "Отправить"</p>	<ul style="list-style-type: none"> • Если введенные данные корректны - <ul style="list-style-type: none"> ◦ Сообщение "Заявка отправлена" выведено на экран. ◦ Новая заявка появилась в списке на странице "Заявки". • Если введенные данные НЕ корректны -; <ul style="list-style-type: none"> ◦ Валидационное сообщение со всеми ошибками выведено на экран. ◦ Заявка НЕ появилась в списке на странице "Заявки".

4. Написание тест кейсов на основании первоначальных требований, тестовых данных и шаблона теста

После того, как тестовые данные и шаги теста готовы приступаем непосредственно к разработке тест кейсов. Здесь нам помогут такие методы комбинирования как:

- **Последовательный перебор.** Представляет собой перебор всех возможных комбинаций имеющихся значений. Таким образом получается, что количество тест кейсов будет равно произведению количества вариантов тестовых данных для каждого поля. Для нашего конкретного примера мы получим **1170 тест кейсов**.
- **Попарный перебор ([Pairwise Testing](#)).** Зачастую, сбой вызывают не сложное сочетание всех параметров, а сочетание лишь пары параметров. Техника **попарного перебора**, позволяет создать тестовые наборы, комбинирующие данные из двух полей. Благодаря этому, количество полученных на выходе тест кейсов в разы меньше, чем при комбинировании того же набора данных при **последовательном переборе**. Отметим также, что в данный момент существует несколько алгоритмов генерации комбинаций для попарного тестирования: [Orthogonal Arrays Testing](#), [All pairs](#), [IPO \(In-Parameter Order\)](#). Так например, при использовании техники **All Pairs** в нашем конкретном случае мы получим всего **118 тест кейса**. (примеры сравнения эффективности разных алгоритмов генерации можно найти [здесь](#))

По завершению подготовки комбинаций данных, подставляем их в шаблон тест кейса, и в результате имеем набор тестовых случаев, покрывающий тестируемые нами требования к форме приема заявок.

Примечание:

Напоминаем, что тест кейсы разделяются по ожидаемому результату на **позитивные и негативные тест кейсы**.

Пример **позитивного** тест кейса (все поля **ОК**):

Действие	Ожидаемый результат
1. Открываем форму отправки сообщения	<ul style="list-style-type: none"> • Форма открыта • Все поля по умолчанию пусты • Обязательные поля помечены - * • Кнопка "Отправить" не активна
2. Заполняем поля формы: <ul style="list-style-type: none"> • Тип обращения = Консультация • Контактное лицо = йцукенгшщзйцукенгшщзйцуке • Контактный телефон = +7-916-111-11-11 • Сообщение 	<ul style="list-style-type: none"> • Поля заполнены • Кнопка "Отправить" - активна (Enabled)
3. Нажимаем кнопку "Отправить"	<ul style="list-style-type: none"> • Сообщение "Заявка отправлена" выведено на экран. • Новая заявка появилась в списке на странице "Заявки".

Пример **негативного** тест кейса (поле Контактное лицо - **NOK**):

Действие	Ожидаемый результат
1. Открываем форму отправки сообщения	<ul style="list-style-type: none"> • Форма открыта • Все поля по умолчанию пусты • Обязательные поля помечены - * • Кнопка "Отправить" не активна
2. Заполняем поля формы: <ul style="list-style-type: none"> • Тип обращения = Консультация • Контактное лицо = @#\$%^&.;?,> \№"!()_{}[<~ • Контактный телефон = (916)333-33-33 • Сообщение = йццуйцуйц(...)йцу 	<ul style="list-style-type: none"> • Поля заполнены • Кнопка "Отправить" - активна (Enabled)

- 1024 символа	
3. Нажимаем кнопку "Отправить"	<ul style="list-style-type: none"> Валидационное сообщение со всеми ошибками выведено на экран: "В поле "Контактное лицо" запрещено использование цифр и спец. символов." Заявка НЕ появилась в списке на странице "Заявки".

Авторы: Алексей Булат, Владимир Антонов