

Влад Головач, 28 мая 2005

Юзабилити-тестирование по дешевке

Описание методов юзабилити-тестирования. В приложениях также разбирается экспертная оценка в ее связи с тестированием и приводятся сведения о планировании трудозатрат.

1. [Что такое юзабилити-тестирование](#)
2. [Почему по дешевке](#)
 1. [Effectiveness и efficiency](#)
 2. [Нет количеству!](#)
3. [Что именно измерять?](#)
4. [Насколько глубоко удовлетворение?](#)
 1. [Анкетирование](#)
 2. [Наблюдение за эмоциональными реакциями](#)
5. [Что нужно для тестирования](#)
 1. [Респонденты](#)
 2. [Методы тестирования](#)
 3. [Тестовые сценарии](#)
 4. [Рабочее место и способы фиксации данных](#)
 5. [Протестированный тест](#)
6. [Проведение тестирования](#)
 1. [Введение в задачу](#)
 2. [Выявление ожиданий от системы](#)
 3. [Тестирование](#)
 4. [Завершение теста](#)
 5. [Тестирование на прототипах](#)
7. [Анализ результатов](#)
 1. [Когда начинать анализ](#)
 2. [Анализ действий респондентов](#)
 3. [Анализ количественных данных](#)
 4. [Когда вносить исправления](#)
8. [Подготовка отчета](#)
 1. [Резюме](#)
 2. [Представление проблем](#)
 3. [Количественные данные](#)
 4. [Приложения](#)
9. [Приложение 1. Экспертная оценка](#)
 1. [Достоинства и недостатки экспертной оценки](#)
 2. [Виды экспертной оценки](#)
 3. [Проверка по контрольному списку](#)
 4. [Эвристическая оценка](#)
 5. [Мысленная прогонка по интерфейсу](#)
 6. [Анализ задач/характеристик](#)
 7. [Выводы](#)
10. [Приложение 2. Планирование юзабилити-тестирования](#)
 1. [Оценка объема работы](#)

2. [Формирование списков требований к респондентам и пользовательских задач](#)
 3. [Поиск респондентов](#)
 4. [Составление тестовых сценариев](#)
 5. [Пилотный тест](#)
 6. [Проведение тестирования](#)
 7. [Анализ результатов](#)
 8. [Экспертная оценка интерфейса](#)
 9. [Написание итогового отчета](#)
11. [Рекомендуемая литература](#)

Юзабилити-тестирование появилось в крайне бюрократизированных областях – в военной промышленности и сфере рискованного гражданского производства (самолеты, электростанции и т.п.). Как следствие, юзабилити-тестирование само было чрезвычайно бюрократизировано – нужно было не просто найти проблему, но, что было даже более важно, доказать, что эта проблема действительно существует. Кроме того, юзабилити-тестирование наследовало приемы и правила научных исследований, в которых не просто важно обнаружить феномен, но также нужно убедиться, что он не является следствием посторонних обстоятельств. Из-за вынужденного усложнения процесса тестированием занимались целые команды узких специалистов: один пишет тестовые задания, другой собственно тестирует, третий данные анализирует, а еще один подтаскивает и оттаскивает респондентов.

Легко понять, что такое тестирование, при всех своих достоинствах, очень уж длительно и неприемлемо дорого.

В последние годы ситуация начала меняться: вместо крайне формальных и крайне же дорогих тестов популярными становятся тесты неформальные и дешевые. Вместо небольшой толпы с кучей дорогого оборудования на первый план выходит единственный человек с минимум вещей. Квинтэссенция такого подхода описана в книге [Веб-дизайн: книга Стива Круга или «не заставляйте меня думать!»](#).

Конечно, простые тесты не универсальны. Для приборной панели самолета, они, пожалуй, не подойдут. Но для обычного ПО простые тесты гораздо лучше, хотя бы потому, что их можно провести больше.

Здесь вы научитесь проводить юзабилити-тестирование, используя как раз методики быстрого тестирования. Более сложные и более формальные методы вы сможете почерпнуть в специализированной литературе или же изобрести самостоятельно.

Что такое юзабилити-тестирование

Юзабилити-тестированием является любой эксперимент, направленный на измерение качества интерфейса или же поиск конкретных проблем в нем.

Польза юзабилити-тестирования многогранна. Тестирование позволяет:

- Понять, насколько плохо или хорошо работает интерфейс, что может либо побудить улучшить его, либо, если он уже достаточно хорош, успокоиться; в любом случае достигается польза.
- Сравнить качество старого и нового интерфейсов и тем самым дать обоснование изменениям или внедрению.

- Найти и опознать проблематичные фрагменты интерфейса, а при достаточном объеме выборки также и оценить их частотность.

В то же время юзабилити-тестирование не может сделать из плохого продукта продукт хороший; оно всего лишь делает продукт лучше.

Первое тестирование крупных систем всегда показывает, что интерфейс работает гораздо хуже, чем думает его владелец или создатель, и гораздо лучше, чем изначально уверен тестер.

Почему по дешевке

Резко снизить трудоемкость, а значит, и себестоимость юзабилити-тестирования позволяют три подхода:

1. Некоторое упрощение понятие юзабилити.
2. Отказ от сбора количественных данных.
3. Снижение стоимости оборудования и уменьшение оплаты времени респондентов.

Третий подход заключается в использовании только мобильных лабораторий, что подробно описано в разделе [Рабочее место и способы фиксации данных](#). Оставшиеся два подхода описаны ниже.

Effectiveness и efficiency

В главной и наиболее употребимой трактовке (стандарт ISO 9241-11) понятие юзабилити определяется как

The extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use.

Мой перевод этой формулировки на русский язык:

Степень эффективности*, трудоемкости** и удовлетворенности, с которыми продукт может быть использован определенными пользователями при определенном контексте использования для достижения определенных целей.

* Например, скорость работы или количество человеческих ошибок и длительность их исправления.

** Например, количество операций, которые нужно выполнить для достижения результата или объем информации, которую нужно переработать для принятия решения. Термин *efficiency* до сих пор часто переводят на русский как *продуктивность*; на мой взгляд, это грубая ошибка, поскольку в оригинальном стандарте ISO 9241-11 под *efficiency* понимается нечто близкое к понятию КПД.

Главными показателями эффективности являются скорость работы пользователя, скорость обучения и количество человеческих ошибок (более детальный список метрик см. в разделе [Что именно измерять](#)). Это нужные показатели, еще как влияющие на дизайн. Что приятно, измерять их не особо проблематично.

С показателями трудоемкости все несколько сложнее. В эту группу входят:

- Успешность, т.е. соотношение выполненных тестовых заданий к невыполненным или выполненным полностью неправильно.
- Мощность, т.е. соотношение задач из деятельности пользователя к задачам, для решения которых данный продукт предназначен.
- Нагрузка на пользователя (как ментальная нагрузка, так и, к примеру, количество действий пользователя на тестовое задание).

Проблемы здесь две. Мощность – вообще не измеряемый показатель (большая или меньшая мощность продукта есть целиком и полностью вопрос дизайна). Нагрузка же на пользователя либо тяжело измеряется, либо неактуальна, поскольку все равно проявляется в показателях эффективности – при интенсивной нагрузке снижается скорость работы. Таким образом, от всех показателей трудоемкости на практике остается только успешность.

В этом упрощении нет ничего крамольного. В реальном тестировании все равно никогда не собирается весь набор показателей – как-никак для каждой конкретной системы важны только некоторые из них. Однако принципиальное последствие данного упрощения формулировки – значительное упрощение методологии юзабилити-тестирования – не может быть достигнуто никаким другим способом.

Таким образом, здесь и далее юзабилити понимается как:

Степень эффективности, успешности и удовлетворенности, с которыми продукт может быть использован определенными пользователями при определенном контексте использования для достижения определенных целей.

Нет количеству!

Вторая возможность снизить трудоемкость юзабилити-тестирования – отказ от сбора большей части количественных данных. Делается это по двум причинам:

- Каждый конкретный тест может быть направлен либо на получение количественных данных, либо качественных данных. Качественные данные, как правило, более востребованы при дизайне, поэтому тест лучше планировать именно в расчете на них.
- Количественные данные все равно получаются ненадежными. Их можно измерить надежно, но дело это чрезвычайно трудоемкое.

Качество или количество?

Юзабилити-тестирование может быть направлено либо на получение количественных данных (нужных для измерения эргономичности интерфейса), либо на получение данных качественных (нужных для того, чтобы понять, что именно плохо и как это исправить). Как правило, в едином тесте достичь обеих целей не получается.

Предположим, мы измеряем скорость работы с системой. В этом случае тест должен быть спланирован таким образом, чтобы исключать любые несвойственные реальной работе замедления пользователя. Например, когда пользователь совершит ошибку, не получится задать ему никаких вопросов, чтобы выявить причины этой ошибки. С другой стороны, если ориентироваться на качественные данные, все количественные результаты окажутся сомнительными.

На самом деле количественные данные есть вообще роскошь юзабилити, поскольку они более приятны, нежели реально нужны. Нужнее динамика, т.е. степень изменения этих данных, что сильно облегчает жизнь. Неважно, насколько точно посчитаны ошибки – если до и после оптимизации интерфейса они подсчитывались одинаково, динамика, по-видимому, оказывается верной. Например, если при подсчете число человеческих ошибок сократилось вдвое, можно рассчитывать (только не просите меня обосновать это суждение), что их действительно стало вдвое меньше, хотя нельзя точно сказать, сколько их было и сколько их стало в действительности.

Ненадежные цифры

Кроме того, рассмотрения заслуживает вопрос, верить ли результатам юзабилити-тестирования вообще. В конце концов, тестирование это не волшебство, так что, предположив, что тестирование может быть плохим, необходимо прийти к выводу, что плохими могут быть и его результаты.

Ответ на этот вопрос прост и печален – оснований верить в результаты юзабилити-тестирования нет вовсе.

В самом деле, несмотря на то, что при юзабилити-тестировании на входе у нас совершенно реальные данные, наш неизбежный волонтаризм не позволяет в полной мере им доверять. Слишком уж много у нас потенциальных источников ошибок:

- Реальные пользователи могут отличаться от выбранных нами респондентов. В условиях маленькой выборки даже незначительная флуктуация в поведении респондентов может привести юзабилити-специалиста к ложным выводам.
- Тестовые задания могут неадекватно отражать реальную деятельность пользователей в системе.
- Юзабилити-специалист может не заметить части проблем или неправильно понять сущность замеченных проблем.

Рольф Молич (Rolf Molich) регулярно проводит [сравнительное тестирование самого юзабилити-тестирования \(en\)](#). Результаты шокирующие. Так, во втором тесте, в котором девять групп юзабилити-специалистов самого разного уровня тестировали службу HotMail, разброс результатов был очень велик, даром, что тестовые задания были идентичны. Все группы нашли в общей сложности 310 проблем с интерфейсом. Но три четверти процентов проблем были найдены только какой-либо одной группой и не найдены остальными группами (в эти проценты входят и двадцать девять действительно серьезных проблем).

В целом, юзабилити-тестирование *может быть* научным исследованием, к которому вполне применимы все требования к научным исследованиям вообще.

К примеру, сравним юзабилити-тестирование с социологическим исследованием. Социолог предпринимает специальные меры, замечу - сложные и трудоемкие – чтобы обеспечить корректность выбора респондентов. Мы – нет. Социолог пользуется проверенными, корректными со статистической точки зрения, инструментами, как при сборе данных, так и при их анализе. Мы – нет.

Так что всякий раз, когда мы пытаемся что-либо измерить точно, мы ошибаемся. Когда мы даже и не пытаемся добиться точности, мы тоже ошибаемся, разве что не так сильно, поскольку планка уже занижена почти до пола.

Что это значит конкретно? Мы, как правило, не можем с уверенностью сказать, к примеру, что мы изъяли из интерфейса все причины человеческих ошибок. Просто потому, что с другими, возможно, более корректно подобранными респондентами, мы – опять возможно – нашли бы больше ошибок. Это же соображение верно и для остальных показателей качества интерфейса и уж тем более верно для других тестовых заданий. А что было бы, если бы тестирование планировал и проводил кто-то более опытный, чем мы! И представить страшно.

Таким образом, мы должны замерять эргономические показатели лишь для сравнения с новым интерфейсом, сознавая при этом, что наши измерения ни в коей мере не верны. Нужны они только для планирования ближайших оптимизаций. С уверенностью можно сказать только одно – независимо от того, сколько мы тестируем, резерв улучшения есть всегда. Как резерв улучшения самого интерфейса, так и наших методик его тестирования.

Зачем все-таки нужны цифры

Разругав количественные данные, как неинтересные и ненадежные, я не могу не указать их истинного предназначения. Количественные данные абсолютно необходимы при сравнительном юзабилити-тестировании. Если есть несколько решений, из которых нужно выбрать лучшее – количественным данным просто нет альтернативы, собирать их надо обязательно и нужно принять все возможные меры, чтобы обеспечить их надежность (тем более что при сравнительном юзабилити-тестировании качественные данные вовсе не нужны). Однако сравнительное тестирование – редкость, соответственно, здесь эта тема вовсе не рассматривается. Если вам понадобится провести сравнительное тестирование, обращайтесь к [рекомендованной литературе](#).

Что именно измерять?

Количество измеряемых показателей в конкретном тесте может быть довольно велико, но все они, как правило, сводятся к набору из пяти базовых характеристик. Ниже приведен список этих характеристик с примерами метрик, созданных на их основе.

- **Скорость работы пользователя.** Метрики: длительность выполнения операции; время, затраченное на обнаружение ошибок; время, затраченное на исправление ошибок; количество команд, исполняемых при выполнении операции (подразумевается, что чем больше команд, тем дольше их отдавать); длительность поиска сведений в документации; количество команд, более эффективных, чем использованные пользователем; снижение производительности при длительной работе.
- **Ошибки.** Метрики: процент операций, вызвавших ошибку; среднее число ошибок на операцию у опытных пользователей (именно у опытных, т.к. у неопытных могут действовать и факторы из группы скорости обучения); количество ошибок, не обнаруженных и не исправленных пользователями.
- **Обучаемость навыкам работы с системой.** Метрики: количество и частота обращений к справочной системе; длительность периода между началом использования системы и точкой, в которой скорость работы/количество ошибок пользователей перестает расти; разница в количестве ошибок/скорости работы у пользователей с опытом использования системы и без такого опыта.
- **Субъективная удовлетворенность пользователя.** Измерение этой характеристики сопряжено с определенными трудностями, заслуживающими отдельного рассмотрения. Метрики этого свойства см. ниже.

- **Сохранение навыков работы с системой.** Метрики: разница в скорости работы/количестве ошибок у пользователя после часа работы с системой и у того же пользователя в начале использования системы после длительного перерыва.

Как видно, измерение качества интерфейса может быть довольно сложным, так, например, тест сохранения навыков может занять больше месяца. Но это только тесты на вторую и третью составляющие юзабилити, а именно на эффективность и удовлетворенность. Помимо них есть еще и успешность, которая более важна и которую измерить гораздо проще – нужно всего лишь подсчитать, какой процент заданий пользователь либо выполняет полностью неправильно, либо не может выполнить вовсе. Это сильно упрощает жизнь юзабилити-специалиста.

Насколько глубоко удовлетворение?

В отличие от прочих характеристик, удовлетворенность находится не в реальном мире, а в голове у пользователя. Как следствие, ее невозможно «пощупать», а значит, и объективно измерить. Но, по крайней мере, ее можно измерить опосредованно.

Существует два возможных направления действий. Во-первых, респондента можно спросить, насколько интерфейс кажется ему удовлетворительным. Во-вторых, по поведению респондента можно определить, нравится или не нравится ему интерфейс в любой конкретный момент времени; подсчитав количество проявленных реакций, можно оценить удовлетворенность. Разумеется, эти оценки относительны; их ценность проявляется только в сравнении с новым интерфейсом или в сравнении с конкурентами.

Ниже приведены некоторые методы измерения удовлетворенности.

Анкетирование

Если пытаться определить удовлетворенность через опрос респондентов, не обойтись без формальных анкет. В самом деле, если формат опроса не фиксирован, не может быть уверенности, что респондентам задан один и тот же вопрос, а значит, и ответы становятся сомнительными.

К сожалению, анкетирование имеет в российских условиях крупный недостаток – надежных анкет просто не существует. Несмотря на то, что в странах загнивающего Запада создано множество вполне работоспособных анкет ([SUMI](#), [QUIS](#), [MUMMS](#), [IsoMetrics](#) и др.), ни одна из них не переведена на русский язык и не протестирована заново. В результате эти анкеты, будучи, кстати сказать, весьма дорогими, ничем не надежнее любых анкет, которые вы можете придумать самостоятельно.

К сожалению, разработка и тестирование надежных анкет является очень долгим и трудоемким процессом, так что рассчитывать на скорое появление качественных отечественных анкет не приходится.

См. о [сложностях разработки анкет \(en\)](#).

Ниже приводятся две работоспособные, хотя и ненадежные, анкеты.

Анкета по словам

Впервые эту анкету предложили исследователи из Microsoft Usability Laboratory как способ очень быстро, пускай и заведомо ненадежно, оценить удовлетворенность. Анкета очень проста. Респонденту предъявляется лист бумаги с набором случайно подобранных прилагательных, одна половина которых скорее позитивна, вторая – негативна. Респонденту предлагается подчеркнуть слова, которые, на его взгляд, применимы к продукту (сходство анкеты с анкетами, использующимися в методике семантического дифференциала, не значимо – это совершенно разные методы). После того, как анкета заполнена, подсчитывается разница между числом негативных и позитивных терминов.

См. в интернете: [Product Reaction Cards by Joey Benedek and Trish Miner](#).

Я использую следующий набор прилагательных:

Устаревший – Эффективный – Нечеткий – Неудобный – Замусоренный – Тусклый – Яркий – Чистый – Прямой – Ясный – Непоследовательный – Неуправляемый – Привлекательный – Стандартный – Управляемый – Хороший – Интуитивный – Веселый – Любительский – Неэффективный – Опасный – Скучный – Радостный – Безопасный – Жесткий – Раздражающий – Треугольный – Неприятный – Комфортабельный – Холодный – Умный – Беспольный – Халтурный – Теплый – Светлый – Последовательный – Загадочный – Качественный – Интересный – Ненадежный – Гибкий – Красивый – Некрасивый – Непривлекательный – Полезный – Глупый – Запутанный – Удобный – Понятный – Непредсказуемый – Четкий – Тяжелый – Современный – Легкий – Дружественный – Нестандартный – Плохой – Надежный – Сложный – Простой – Темный – Профессиональный – Медленный – Круглый – Печальный – Недружественный – Предсказуемый – Непонятный – Быстрый – Головоломный – Грустный – Приятный

Обратите внимание, что слова неслучайно даны вперемешку, именно так и нужно предъявлять их респондентам.

Формальная анкета

В отличие от анкеты по словам, эта анкета не может быть использована без адаптации под конкретный проект. Часть ее вопросов порой неактуальна, порой нуждается в смене формулировки. В любом случае, для респондентов-женщин нужно менять половую принадлежность формулировок анкеты.

Анкета представляет собой несколько вопросов, для каждого из которых респондент может выбрать один из пяти вариантов ответа. Обратите внимание, что я проектировал эту анкету только как пост-тестовую, ее применение в другом качестве сомнительно.

Вопросы анкеты:

Во время выполнения заданий я ошибался	Нет	Да
Система способна делать все, что мне нужно и даже больше	Нет	Да
Система работает достаточно быстро	Нет	Да
Мне нравится внешний вид интерфейса	Нет	Да
Я чувствую, что если я лучше изучу систему, я смогу делать в ней вещи, о которых сейчас даже и не подозреваю	Нет	Да
Систему можно легко настроить под мои нужды	Нет	Да

Начать работу было легко; я не столкнулся с существенными трудностями	Нет	Да
Всякий раз, когда я ошибался, я с легкостью замечал и исправлял свою ошибку	Нет	Да
Я доволен своей скоростью работы	Нет	Да
Во время выполнения заданий я чувствовал себя вполне уверенно	Нет	Да
В любой момент времени я понимал, что должен сделать дальше	Нет	Да
Система представляется мне полезной, я бы с удовольствием использовал бы её для решения моих задач	Нет	Да

Результаты нужно подсчитывать по следующему алгоритму: центральное значение дает ноль баллов, крайние значения дают либо –2 балла (левый вариант ответа), либо +2 балла (правый вариант), промежуточные значения либо –1 либо, +1 балл соответственно. Сумма баллов является сравниваемым значением.

Наблюдение за эмоциональными реакциями

Помимо анкетирования можно подсчитывать эмоциональные реакции респондента. К примеру, респондент улыбнулся – ставим плюс, ругнулся или поморщился – ставим минус. Количество и знак реакций и составляет искомую величину показателя.

Проблемы есть и с этим методом.

Во-первых, непонятно, как подсчитывать реакции разной силы. Сколько раз респондент должен улыбнуться, чтобы уравновесить восемь секунд отборной брани? А девять секунд брани?

Во-вторых, подсчитывать реакции всех респондентов должен один и тот же человек, поскольку нескольким людям невозможно синхронизировать свои представления о том, что, собственно, входит в понятие эмоциональной реакции. В результате сильно возрастает ресурсоемкость теста.

Не связывайтесь с этим тестом, если хоть чуть-чуть не уверены в своей способности улавливать эмоции других людей (например, если вы Эм, а не Жо).

Вторую проблему составляет неопределенность теста. Наблюдается только операциональная удовлетворенность, сиречь удовольствие, а воспринимаемая удовлетворенность, почти всегда более важная, остается за кадром.

Что нужно для тестирования

Теперь, когда принципиальные вопросы в целом разобраны, можно переходить и к практике. Начать стоит с перечня того, что нужно собрать в одном месте для проведения юзабилити-тестирования (ниже эти пункты будут описаны более подробно). Итак, что нам нужно:

- респонденты
- метод тестирования
- тестовые сценарии
- рабочее место для теста и отлаженный метод фиксации материала

- протестированный тест.

Респонденты

При выборе респондентов для тестирования сначала удобно определить общие требования к респондентам, а уж потом подбирать респондентов из целевой аудитории, пользуясь сформированными требованиями.

Учтите, что выбор респондентов, не входящих в целевую аудиторию, гораздо более опасен, чем кажется на первый взгляд. Вы либо выявите несуществующие проблемы, либо не выявите существующие. В худшем случае вы настолько упростите интерфейс, что им будет тяжело пользоваться даже среднеопытным пользователям, которых на самом деле большинство.

Общие требования к респондентам

Первый пункт – нужен ли респондентам опыт работы с системой. Общее правило: если оптимизируется интерфейс существующей системы, половина респондентов должна иметь опыт работы (на них можно определить проблемы переучивания при внедрении), а половина нет (на них определяется скорость обучения). Если существуют конкурирующие системы, лучше другая пропорция: треть с опытом работы с предыдущей версией, другая треть – с опытом использования конкурирующих систем, оставшаяся треть – без опыта работы с системой.

Второй пункт – уровень компьютерной грамотности. При прочих равных предпочтительным выбором является реальный, т.е. совпадающий с опытом целевой аудитории, уровень у трех четвертей респондентов и низкий уровень – у оставшейся четверти (на ней можно определить большее количество проблем).

Уровень компьютерной грамотности удобно определять по следующей шкале:

1. **Высокий.** Респондент имеет компьютер на работе и дома, большая часть трудовой деятельности выполняется на компьютере, респондент самостоятельно использует компьютер как средство саморазвития, активно пользуется сервисами в интернете (например, регулярно покупает товары и услуги в онлайн-магазинах).
2. **Выше среднего.** Респондент имеет компьютер на работе и дома, большая часть трудовой деятельности выполняется на компьютере, но респондент не использует компьютер для решения задач, выходящих за пределы его основной деятельности (работает на компьютере «от звонка до звонка» и не больше).
3. **Средний.** Работа с компьютером является частью обычной (трудовой или личной) деятельности в течение двух лет или больше.
4. **Низкий.** Либо на работе, либо дома есть компьютер, но опыт работы с компьютером не превышает двух лет и компьютер не является значимым инструментом в работе.
5. **Очень низкий.** Опыт использования компьютера спорадический, по длительности не превышает трех лет. Компьютер не используется ни на работе, ни дома.

На третьем месте стоит возраст. Оптимальная пропорция: три четверти респондентов имеет возраст целевой аудитории системы, оставшаяся четверть старше (на ней можно определить большее количество проблем).

Меньшее влияние на результаты имеет пол респондентов – но это не значит, что подбирать респондентов корректного пола необязательно. Стоит увеличить количество женщин среди респондентов по сравнению с пропорцией в целевой аудитории, поскольку на женщинах легче определять проблемы при внедрении (женщины, в целом, медленнее обучаются, но, обучившись, лучше работают).

Последняя значимая характеристика – уровень эмоциональной открытости респондентов. Чем более скован респондент, тем меньше он способен сказать вам ценного. Даже определив наличие проблемы, вы не сможете добиться от него никакой информации о том, что эту проблему вызвало. Существует прекрасный способ решить проблему недостаточной эмоциональной открытости – стоит иметь базу респондентов и использовать их повторно. Респондент, уже знающий на опыте, что в юзабилити-тестировании нет ничего страшного, значительно охотнее идет на контакт и вообще более разговорчив.

Наконец, когда желаемые для теста свойства пользователей уже определены, пришло время подобрать таких респондентов, которые не только удовлетворяют вышеперечисленным требованиям, но и входят в целевую аудиторию системы.

Сколько нужно респондентов

В 1992 году Роберт Вирзи (Robert Virzi) в статье Refining the test phase of usability evaluation: how many subjects is enough? предположил, что для теста достаточно пяти респондентов. Через год эстафету приняли Якоб Нильсен и Томас Ландауэр (Jakob Nielsen и Thomas K. Landauer) со статьей A mathematical model of the finding of usability problems, в которой утверждали, что пяти респондентов достаточно для того, чтобы выловить 70% проблем и еще три респондента нужны для того, чтобы довести результативность до 85%.

Юзабилити-сообщество полюбило эти цифры всей душой. С тех пор фраза «5-8 респондентов» стала чуть ли не мантрой. Увы, эта мантра ложна.

Во-первых, все три автора писали только о тестировании маленьких систем. Что делать, если система слишком велика, чтобы можно было уместить тест на каждом респонденте в полтора часа (это максимум, который может выдержать человек, как респондент, так и экспериментатор; гораздо лучше тесты по 40 минут). В этом случае придется выполнять несколько разных тестов на разных респондентах; без этого охватить весь интерфейс системы окажется попросту невозможно. Сколько респондентов при этом понадобится – зависит от системы, четких градаций тут быть не может. Так, для тестирования крупного корпоративного сайта, по-хорошему, нужно человек двадцать несколькими сериями по 5 человек.

Во-вторых, восемь человек – это слишком мало, чтобы говорить хотя бы о какой-нибудь точности измерения эргономических характеристик. Для измерения нужно хотя бы человек двенадцать.

В-третьих, в восемь человек не вместить ни гендерное, ни возрастное, ни какое-либо другое разнообразие респондентов. Если нужно протестировать интерфейс, рассчитанный на несколько четких групп аудитории, каждая группа должна получить свои пять-восемь респондентов.

С другой стороны, всё-таки первые несколько респондентов позволяют выявить львиную долю проблем. Посему единственным реально возможным выходом является

тестирование сериями: первая серия проходит, выявленные проблемы решаются, затем вторая серия, проблемы решаются опять и так далее. Если в первой серии использовать респондентов всех типов, остальные серии можно будет безбоязненно прервать, когда количество выявляемых данных сильно упадет. Первая серия при этом должна быть побольше, остальные поменьше.

Организационные вопросы

Помимо собственно требований к респондентам, открытым остается вопрос: как убедить потенциального респондента участвовать в тестировании?

Если вы проектируете интерфейс на заказ, старайтесь переложить поиск респондентов на заказчика. Почти всегда у системы есть реальные или потенциальные пользователи, с которым у заказчика сложились особые отношения и которые – что приятно – особо заинтересованы в новом интерфейсе, отчего весьма коммуникабельны.

Если вы проектируете интерфейс для системы с широкой (обычной) целевой аудитории, не пренебрегайте своими близкими. Они и коммуникабельны и легко доступны.

Многие компании по подбору персонала занимаются подбором респондентов для фокус-групп, соответственно, они могут подобрать респондентов и для юзабилити-тестирования. К сожалению, фокус-группа является разовым, сравнительно непродолжительным мероприятием. Юзабилити-тестирование потребует назначать встречи с респондентами поочередно, по одному человеку за раз, что сильно утяжеляет процесс.

Ведите базу людей, которых вы уже использовали для тестирования. Как правило, с ними легче договориться, чем с теми, кто не имеет опыта участия в тестировании.

Договариваясь с респондентом о встрече, проявляйте максимальную гибкость и податливость. Респондент, даже если его время оплачивается, делает вам одолжение, соглашаясь участвовать в тестировании.

Если респондента нашли вы сами, даже если это ваш друг или родственник, он должен быть вознагражден за затраченное им время (можно обойтись без вознаграждения, если респонденты предоставлены заказчиком интерфейса, если же вы воспользовались услугами служб подбора персонала, вознаграждение нужно обсуждать с представителем службы). Для неспецифической аудитории лучшим стимулом являются деньги – оптимальной суммой оплаты является удвоенная часовая ставка конкретного респондента с учетом времени, которое респондент затратил на дорогу. Для специфической аудитории часто нужны очень большие суммы вознаграждений, в таких условиях респондентов разумно вознаграждать ценными подарками, на которые можно получить крупную оптовую скидку (лично я предпочитаю использовать дорогой алкоголь).

Методы тестирования

Основных методов юзабилити-тестирования всего три: пассивное наблюдение за выполнением тестовых заданий, поток сознания и активное вмешательство; первый предназначен для сбора количественных данных, последний – качественных:

- **Пассивное наблюдение за выполнением тестовых заданий.** Сущность метода очень проста: респондент выполняет тестовые задания, его действия анализируются (во время теста или после, по протоколам), что позволяет как найти

проблематичные фрагменты, так и замерить эргономические характеристики интерфейса.

- **Поток сознания (think aloud).** Соответствует проверке посредством пассивного наблюдения, но респондента при этом просят также устно комментировать свои действия. Затем комментарии анализируются. Метод довольно нестабильный, но порой дающий интересные результаты (очень зависит от разговорчивости респондента). Крупный минус потока сознания – измерения эргономических характеристик интерфейса весьма сомнительны.
- **Активное вмешательство.** В этом методе юзабилити-специалист не ждет милостей от природы в лице респондента, а старается взять их сам. После каждого действия респондента экспериментатор спрашивает его, почему респондент действует именно так; на каждом экране экспериментатор спрашивает, как именно респондент понимает назначение и функции этого экрана. Этот метод ближе к фокусированному интервью, чем к собственно тестированию – например, метод можно использовать даже без тестовых заданий, лишь бы был интерфейс для обсуждения. Понятно, что при активном вмешательстве никакие измерения попросту невозможны, но зато объем получаемых качественных данных наиболее велик.

Тестовые сценарии

Тестовый сценарий – это тестируемый аспект системы. На мой взгляд, адекватно подобранные тестовые сценарии являются важнейшей предпосылкой качества тестирования.

Сценарии состоят из пользовательской задачи и сопутствующих ей:

- значимых эргономических метрик
- тестовых заданий респондентам (заданий может быть несколько)
- признаков успешности выполнения задачи.

Разберем их подробно.

Пользовательская задача

Первым шагом определения сценариев является определение значимых пользовательских задач. Эти задачи – исходный материал для составления сценариев.

Что такое пользовательская задача? Это задача, которую ставит перед пользователями их деятельность, и которая имеет самостоятельную ценность для пользователей.

Пользовательская задача выполняется как одна или несколько операций (пользовательская операция не имеет самостоятельной ценности). Например, для программы-почтового клиента задачами являются:

- написание и отправка письма
- прием сообщений с сервера
- настройка программы под свои нужды (например, настройка автоматического приема почты через заданные интервалы).

А вот выбор адресата из адресной книжки при написании нового письма уже не является задачей, потому что это действие не самоценно. Это операция, состоящая из многих действий (нажать на кнопку Кому... > выбрать контакт > подтвердить выбор).

При выборе задач для тестирования следует руководствоваться двумя соображениями:

- Все задачи должны быть реальными, т.е. выявленными из актуальной деятельности пользователей: желание сделать задания посложнее, чтобы найти разом много проблем, несостоятельно – задания должны быть обычными, поскольку нет смысла искать проблемы, с которыми никто не сталкивается.
- Поскольку протестировать весь интерфейс на всех пользовательских задачах возможно только в идеале, приходится ограничивать себя и выбирать только важные задачи. Важными задачами являются, во-первых, частотные задачи, т.е. которые выполняются всеми пользователями и/или выполняются часто, во-вторых, все остальные задачи, которые, как вы подозреваете, выполняются в системе плохо и, наконец, задачи, неправильное выполнение которых приводит к крупным проблемам.

Значимые эргономические метрики задачи

Для каждой задачи нужно выбрать значимые для нее характеристики интерфейса. Конечно, в нашем распоряжении есть метрики из раздела «Что именно измерять». Однако эти метрики неудобны: их трудно измерять и трудно понять (правда, их легче сравнивать). С практической точки зрения гораздо удобнее более приземленные характеристики.

Например, можно посчитать число человеческих ошибок. Но влияние этой метрики на интегральный результат настолько сложно, что все равно при анализе не обойтись без элементов волюнтаризма и субъективизма. Гораздо проще определить влияние тех же ошибок сразу же, еще до теста, например, установив значимой интерфейсной характеристикой «Пользователь способен установить программу менее чем за пять минут, совершив при этом не более двух незначительных ошибок» (где под незначительной ошибкой подразумевается ошибка, которую сам пользователь заметил и исправил). Разумеется, произвола в таком требовании много. Но его все равно меньше, чем в сомнительных спекуляциях о числе выявленных человеческих ошибок. Таким образом, «приземленные» метрики удобнее метрик универсальных.

Кроме того, простые декларативные метрики удобны еще и тем, что позволяют понять, когда нужно продолжать оптимизацию интерфейса, а когда можно и остановиться. Возвращаясь к примеру из предыдущего абзаца, можно сказать, что оптимизировать интерфейс нужно до тех пор, пока установка не станет занимать меньше пяти минут. Если после оптимизации интерфейса программа устанавливается за шесть минут, это повод еще раз модифицировать интерфейс и протестировать его заново.

Вот примеры таких метрик:

- Успешность – респонденты правильно выполняют 90% заданий.
- Эффективность – скорость работы пользователя: регистрация на сайте выполняется меньше чем за 7 минут.
- Эффективность – ошибки: при вводе 10 форм количество ошибок ввода не превышает двух.
- Эффективность – обучаемость навыкам работы с системой: при выполнении задания 9, отличающегося от задания 2 только вводимыми данными, респонденты не совершают ни одной ошибки (не считая опечаток).
- Удовлетворенность – по результатам анкетирования число баллов выросло на 20% по сравнению с прежними результатами.

Тестовые задания

Тестовое задание – это то, что получает от вас респондент, задание, позволяющее провести респондента через фрагмент интерфейса системы и определить характеристики этого фрагмента.

Тестовые задания, помимо того, что должны соответствовать пользовательским задачам, должны обладать еще и следующими свойствами:

- **Однозначность.** Задания должны быть сформулированы так, чтобы исключить их неправильное толкование респондентом. Если респондент поймет задание неправильно, вам почти наверняка не удастся походу теста направить его на правильный путь, не подсказав ему одновременно последовательности выполнения задания.
- **Полнота.** В тексте задания должна присутствовать вся информация, необходимая для выполнения этого задания.
- **Краткость.** Если вы измеряете скорость выполнения заданий, задания должны быть достаточно краткими, чтобы длительность чтения заданий респондентами не влияла на продолжительность выполнения самих заданий (люди читают с разной скоростью). Если текст задания будет велик по объему, вам придется вручную отсекал длительность чтения для каждого задания, что очень трудоемко.
- **Отсутствие подсказок.** По тексту задания не должно быть понятно, как это задание нужно выполнять. Например, недопустимо использовать терминологию системы – вместо каждого термина нужно расписывать его значение, иначе респонденты просто нажмут кнопки с теми же словами и вы не выявите никаких проблем.
- В задании должна присутствовать **точка начала выполнения задания**, т.е. должно быть прописано то окно или экран, на котором респондент должен находиться в начале. Если такой информации представлено не будет, респонденты неизбежно будут переходить к другим фрагментам интерфейса, а значит, задание разными респондентами будет выполняться по-разному, что делает бессмысленным все статистические расчеты. Фиксировать начальную точку задания нужно еще в конце предыдущего задания. Если задание начинается с чистого листа, в конце предыдущего задания должно быть написано «вернитесь на главный экран». Если задание должно начинаться с места, на котором закончилось предыдущее задание, предыдущее задание должно заканчиваться словами «закончив, не закрывайте текущее окно/останьтесь на этом экране».

Помимо этих общих требований нужно учитывать еще и следующее:

- Возможно, что на одну пользовательскую задачу нужно будет написать несколько тестовых заданий. Типичный случай – задача слишком велика, чтобы ее можно было вместить в одно задание. Кроме того, если пользовательская задача является частотной, вас не должно особо интересовать, как она выполняется в первый раз – гораздо интереснее узнать, как пользователи будут выполнять ее во второй, третий, четвертый (и так далее) разы. В этом случае в пределах теста на одном респонденте прогонять эту задачу потребуется несколько раз, каждый раз варьируя задания.
- Помимо заданий, в которых респондент должен выполнить какое-либо действие, допустимы и желательны двойные задания, в которых **респондент должен сначала решить, нужно ли ему в данное время это действие выполнять.** Например, если мы тестируем программу дефрагментации диска, вместо задания «Дефрагментируйте диск компьютера» лучше использовать задание «Проверьте степень фрагментации диска и, если вы сочтете это необходимым, дефрагментируйте диск компьютера». Такие задания должны быть составлены

таким образом, чтобы респондент не мог отказаться от принятия решения, не глядя сказав, что, дескать, все хорошо и дефрагментация не нужна. Кроме того, перед таким тестом разумно намеренно фрагментировать диск, чтобы респондент не смог избежать задания.

- Порой по ходу задания нужно насильственно изменить состояние системы. К примеру, если вы хотите узнать, как именно пользователи решают определенную проблему, вам придется эту проблему создать. Прерывать для этого выполнение теста недопустимо, поскольку это отвлечет респондента. В таких случаях перед соответствующим заданием можно вставить другое задание, в котором респондент должен создать проблему самостоятельно. Разумеется, никаких сведений об интерфейсе такое задание не предоставит.
- Анализ результатов и подведение статистики сильно упрощаются, если делать не малое число длительных заданий, а большое число заданий кратких, требующих перемещения всего на пару экранов или заполнения одной–двух форм.
- Первое задание теста должно быть вводным, предназначенным исключительно для введения респондента в процесс. Соответственно, оно должно быть простым, а его результаты можно не учитывать.

Не забудьте проверить, что ваши сценарии могут быть выполнены респондентами за ожидаемое время теста. Вероятно, список сценариев придется сокращать.

Признаки успешности выполнения задачи

Последней составляющей сценария являются признаки успешности выполнения задач. Дело вот в чем: не всегда одно и то же задание можно выполнить единственным способом. Запускать же тест, не зная всех этих способов, некорректно, поскольку дальнейший анализ получится сомнительными. Предположим, респондент А выполнил задание способом А, а респондент Б способом Б. Оба респондента справились с заданием, но один все таки лучше другого. Как-никак разные способы, по-видимому, имеют разную эффективность, например, число действий, входящих в способ Б в полтора раза выше числа действий способа А. Способ А в такой ситуации предпочтителен, в идеальной системе (к которой нужно стремиться) все пользователи должны использовать только его.

Кроме того, порой правильный результат теста с точки зрения экспериментатора на самом деле не является правильным, особенно, если предметная область сложна, а юзабилити-специалист знает ее недостаточно. Чтобы убедиться, что правильный результат именно таков, как считает юзабилити, ему необходимо найти специалиста по системе и предметной области, и спросить. Не зная твердо всех способов выполнения задания, вы просто не сможете опознать ошибки.

Рабочее место и способы фиксации данных

Существует два подхода к организации рабочего места для юзабилити-тестирования: стационарное рабочее место и мобильное. Здесь приводятся только рекомендации, относящиеся к мобильным рабочим местам, поскольку мобильные лаборатории и сами дешевле и позволяют уменьшить вознаграждения респондентам (правда, за счет стоимости времени юзабилити-специалиста, которому приходится ездить к респондентам самому).

Итак, что нужно иметь для полноценного тестирования:

1. **Ноутбук.** Требования к ноутбуку просты. Во-первых, как можно более мощный процессор для одновременной записи содержимого экрана (хотя даже самые слабые процессоры Intel Centrino позволяют записывать видео-поток в фоновом режиме, на более мощных процессорах будет быстрее работать сам тестируемый интерфейс, да и качество видео будет выше). Во-вторых, если вы собираетесь записывать видео с самим респондентом (см. ниже), потребуется экран побольше, чтобы на него уместились и тестируемый интерфейс и окно с физиономией респондента.

2. **Веб-камера,** если вы собираетесь фиксировать мимику и направление взгляда респондента. Как правило, чем дороже камера, тем лучшего качества изображение она дает. Желательна камера с креплением на экран ноутбука, поскольку ей удобней пользоваться.

3. **Микрофон.** В принципе, подойдет любой. Лично я пользуюсь обычным микрофоном Genius стоимостью в семьдесят рублей. Если в веб-камеру встроен микрофон, вполне подойдет и он. С другой стороны, более качественный микрофон дает лучшее качество записи, так что будет меньше шипения (но оно ничему не мешает).

4. **Программа записи содержимого экрана.** Фактическим стандартом является TechSmith Camtasia, если же позволяют средства, инвестируйте в TechSmith Morae, специально предназначенную для юзабилити-тестирования (она записывает не только содержимое экрана, но и регистрирует действия пользователя, что позволяет сильно ускорить последующий анализ – с другой стороны, Morae в четыре раза дороже, чем Camtasia, которая и так штука недешевая).

Перед первым тестированием как можно лучше изучите ваше оборудование. Исследуйте, как эффективнее всего располагать камеру и микрофон для получения наилучших результатов. Заучите горячие клавиши программы записи содержимого экрана, научитесь быстро запускать ее в любом режиме. Ничто так не подрывает доверия респондента к тестированию, как зрелище суетящегося экспериментатора, пытающегося правильно включить запись в последний момент перед тестом.

5. Если вы собираетесь фиксировать длительность выполнения заданий, полезно обзавестись **спортивным секундомером** с записью кругов (Laps), позволяющим запоминать серию интервалов. В противном случае вам придется лишний раз просматривать видеозаписи, чтобы вычислить продолжительность заданий, а это очень утомительно.

6. **Тестовые задания** для предъявления респондентам. Как правило, оптимальный вариант – распечатка, каждое задание на отдельном листе, чтобы респондент не мог забежать вперед и прочесть задания, которые он еще не выполнил. На первом листе нужно выводить **вводную форму**. Пример такой формы (в квадратных скобках – переменные данные):

Уважаемый [Имя респондента]!

Предлагаем Вам выполнить ряд заданий, предназначенных для оценки простоты и удобства использования [Наименование системы]. При выполнении заданий чувствуйте себя свободно. Целью исследования является оценка качеств изучаемого интерфейса, а не Вас лично. Если Вы что-то сделаете неправильно, это будет значить, что интерфейс и только интерфейс нуждается в улучшении.

При выполнении заданий Вы должны действовать так, как считаете нужным. Например, если Вы решите воспользоваться Справкой, вы можете это сделать, не спрашивая разрешения экспериментатора.

Обратите внимание, что Ваши действия и слова записываются для дальнейшего изучения, но все собранные данные останутся строго конфиденциальными и будут доступны только исследователям.

Внимательно прочитайте задание и точно следуйте изложенным в нем инструкциям.

Старайтесь довести выполнение каждого задания до конца, но если во время выполнения задания Вы поймете, что не можете или не хотите его заканчивать, сообщите об этом экспериментатору и перейдите к следующему заданию.

Пожалуйста, переворачивайте страницу с заданием только тогда, когда выполните задание на открытой странице.

Если Вы не понимаете какое-либо задание, не стесняйтесь, переспросите проводящего тестирование специалиста.

С другой стороны, в некоторых случаях гораздо эффективнее выдавать респондентам задания не на бумаге, а способом, более приближенным к реальности. Например, тестируя интерфейс кассового терминала, лучше притвориться клиентом, а клиенты редко излагают свои потребности в письменной форме.

7. Если вы собираетесь анкетировать пользователей, понадобятся **распечатанные анкеты**.

8. Если вы собираетесь находиться рядом с респондентом и сразу фиксировать какой-либо параметр, вам понадобятся **планшет с бумагой и ручка**. Удобно заранее напечатать несколько листов с именем респондента и номерами страниц – если вы проводите несколько тестирований подряд, это гарантированно позволит избежать мучений с перепутанными бумагами.

Как видим, нужно не так уж много. Стоимость необходимого оборудования и ПО (не считая ноутбука, в 21-ом веке он не роскошь) составляет в экономичном варианте не более 450 долларов. Достоинства такого решения – надежность и простота работы; кроме того, мобильность позволяет проводить тестирование у самих респондентов, что в разы повышает их число (многие потенциальные респонденты ни при каких условиях не поедут в офис к юзабилити-специалисту).

Запись мимики респондентов

Если вы собираетесь анализировать результаты уже после теста (а не во время него), крайне полезно будет сделать видеозапись с мимикой и жестикуляцией респондентов. Без видео с респондентом придется анализировать движения курсора (сопровождающиеся, поскольку записан и звук, шипением и вскриками). С записью получится анализировать взаимодействие человека с интерфейсом, так как появятся предпосылки для гештальта. Объективно незначительная разница в методе аукнется объективно значительным улучшением результатов.

Проблема в том, что запись видео с самим респондентом сопряжена с определенной трудностью – нужно автоматически синхронизировать лицо респондента и запись его действий.

Не располагайте микрофон возле распечаток тестовых заданий. Оглухнете при просмотре видео.

В стационарной юзабилити-лаборатории это достигается тем, что видео с камеры аппаратно микшируется с видео-потокотом от видео-карты компьютера. Крупные недостатки этого решения – высокая стоимость и низкое качество записанной картинки экрана (количество пикселей на экране в разы превышает то, что можно записать на магнитофон). Кроме того, с такими записями неудобно работать.

Можно синхронизировать запись и вручную, записывая на компьютере одновременно потоки с камеры и с экрана. Но в этом случае после каждого теста придется потратить некоторое время на скучную работу по микшированию двух разных видео-файлов.

Раньше было доступно только тупое, хотя и работоспособное решение (новый, лучший способ описывается ниже). Перед тестированием:

1. Выключите ускорение графики в Windows (в *Панели управления* выберите *Экран*, на вкладке *Параметры* нажмите кнопку *Дополнительно*, в открывшемся окне на вкладке *Диагностика* переведите ползунок *Аппаратное ускорение* в левую часть). После теста ускорение можно включить снова.
2. Запустите любую программу, способную выводить приходящее с камеры видео. Такие программы придаются к веб-камерам, кроме того, подходят программы для видеочата.
3. Установите в правый нижний край экрана окно с приходящей с камеры картинкой (там оно менее всего отвлекает респондента), а окно самой тестируемой системы расположите так, чтобы оно не заслоняло картинки. Окно с видео стоит заклеить бумажкой (благодарю Дмитрия Сатина за идею).
4. Попросите респондента не изменять размер окна тестируемой системы.
5. Включите запись всего содержимого экрана.

Благодаря этому во время выполнения теста записывается все содержимое экрана, включая как действия респондента, так и его изображение с камеры.

Вид экрана при записи мимики таким способом.

Дополнение: В третьей версии TechSmith Camtasia Studio появился режим «картинка-в-картинке» (в угол видеозаписи с содержимым экрана вставляется поток от видеокamеры), так что теперь все стало гораздо проще.

Единственная проблема записи мимики респондента заключается в том, что если вы ездите к респондентам сами, никто не даст гарантии того, что на заднем плане видео не окажется какой-нибудь гадости – респонденты часто любят встречаться с юзабилити-специалистами просто-таки в клоповниках.

Протестированный тест

Наконец, нам нужно проверить сам тест. Нужно убедиться, что:

1. аппаратура работоспособна
2. вы умеете с ней обращаться как молодой полубог
3. все настройки по умолчанию правильны
4. у вас достаточно чистых кассет или места на диске
5. все необходимые бумаги распечатаны и проверены на актуальность и ошибки
6. тестовые задания содержат все необходимые сведения и не требуют дополнительных пояснений
7. в тестовых заданиях нет скрытых подсказок
8. вы умеете быстро приводить тестируемую систему в изначальное состояние, чтобы следующие респонденты не видели изменений, вносимых предыдущими участниками
9. ваше представление о том, что является правильным выполнением заданий, истинно
10. тест на одном респонденте может быть проведен за разумное время (не более полутора часов).

Как видим, пунктов, в которых можно допустить ляпы, настолько много, что сама по себе их проверка станет источником ошибок (сложные задачи трудно выполнить вообще без ошибок). Соответственно, нужен надежный метод проверки. Таким методом является тестирование самого теста, т.е. прогон теста на ком-то, кого не жалко и кого нетрудно отловить (например, на сослуживце). Если тестовый прогон показал хотя бы одну ошибку в подготовке теста, исправьте ее и повторите прогон снова.

Учтите, однако, что тестовый прогон не отменяет необходимости проверять тест и самому, поскольку некоторые аспекты протестировать прогоном невозможно.

Проведение тестирования

Итак, тест подготовлен и можно приступать. Процедура проста. Включив запись и усадив респондента за компьютер:

1. введите респондента в задачу
2. выясните у него его ожидания от системы
3. протестируйте интерфейс
4. выясните, насколько оправдались ожидания респондента
5. завершите тест.

Ниже эти шаги описаны подробно.

Введение в задачу

Введение респондента в задачу состоит в том, что вы последовательно объясняете ему правила тестирования. Все эти объяснения крайне важны, если вы пропустите хотя бы один пункт, результаты будут искажены.

- Объясните респонденту, что такое юзабилити-тестирование и зачем оно нужно.
- Объясните респонденту (здесь допустимо приврать), что именно он и только он нужен для проведения тестирования – почувствовав свою нужность, респондент приободрится.

- Упомяните, что интерфейс разрабатывали не вы (можно и нужно врать), так что вы не обидитесь, если респондент будет ругать интерфейс.
- Перед тестированием не забудьте выключить свой сотовый телефон и попросите сделать то же самое респондента.
- Объясните респонденту, что вы тестируете не его, а систему. Предупредите о том, что все его проблемы – на самом деле проблемы системы, и что если он совершит ошибку, его никто не будет винить, напротив, вы будете знать, что проблема не в нем, а в системе.
- Извинитесь, что вынуждены записывать его действия. Заверьте респондента, что собранные данные останутся при вас и вы передадите заказчику результаты тестирования, перед этим обезличив их. Если вы записываете содержимое экрана, дополнительно попросите респондента не вводить свою фамилию в экранных формах (чтобы ее не увидел ваш клиент, которому вы отдадите видеозапись).
- Объясните респонденту, что он в любой момент может отказаться от продолжения теста и что в этом случае ему все равно будет выплачено вознаграждение. Объясните, что респондент в любой момент может попросить прервать тест, чтобы отдохнуть.
- Наконец, объясните респонденту, что вам бесполезно задавать вопросы об интерфейсе, но зато вас можно и нужно спрашивать, если какое-либо задание респонденту непонятно.

Заучите список того, что нужно говорить перед тестированием. Это еще как влияет на результаты.

Выявление ожиданий от системы

Независимо от типа и назначения теста при тестировании нового интерфейса полезно определить, насколько он соответствует ожиданиям пользователей. Если ожидания оправдались, внедрение и начальная поддержка будут существенно облегчены; если же ожидания оказались обманутыми, система сразу же вызовет отторжение.

Выявление ожиданий, вообще говоря, следует проводить до проектирования интерфейса – но, к сожалению, на ранних этапах работы дело это исключительно трудное и требующее вдобавок незаурядного таланта слушать и задавать правильные вопросы. Однако на стадиях, когда уже есть что тестировать, например при тестировании прототипов, выявлять ожидания пользователей легче легкого, так что глупо не воспользоваться такой возможностью.

Процедура выявления ожиданий состоит из двух шагов:

1. Перед проведением теста у респондента нужно выспросить, что он ожидает от системы. Слушать респондента нужно с внимательным видом, а все, что он скажет, можно благополучно забыть, поскольку все его слова – не более чем фантазии. Спрашивать нужно не для того, чтобы узнать что-то, а для того, чтобы подготовить респондента ко второму этапу.
2. После теста у респондента нужно выспросить, насколько показанный интерфейс соответствует его ожиданиям. Здесь респонденту уже вполне можно верить: во-первых, он подготовлен своими предыдущими ответами, а во-вторых, показанный ему интерфейс может побудить его сформулировать требования, которых он раньше не осознавал.

Тестирование

При тестировании нужно следовать следующим шести «никогда»:

- Никогда не извиняйтесь за несовершенство тестируемой системы.
- Никогда не говорите «Мы потом это исправим».
- Никого не обвиняйте в том, что интерфейс плох («Разработчики, конечно, придурки, и создали нечто несуразное, но мы с вами это шас исправим»).
- Никогда не называйте процесс тестирования «пользовательским тестированием» – респондент решит, что тестируют его, и будет бояться. Идеально, если вы всегда будете называть процедуру «юзабилити-тестирование интерфейса» или просто «тестирование интерфейса».
- Никогда не прерывайте респондента. Даже если он говорит нечто неактуальное, дайте ему полностью выговориться и только тогда задавайте свои вопросы.
- Никогда не формируйте поведение респондента. Некоторые люди подстраиваются под ожидания экспериментатора, например, почувствовав, что вам хочется найти в интерфейсе побольше ошибок, они будут постоянно ошибаться сами, даже если в интерфейсе нет предпосылок для этого. Чтобы избежать такого результата, все ваши слова должны быть подчеркнута нейтральными. Достигнуть нейтральности помогают два простых метода. Во-первых, не стоит задавать вопросы с единственным вариантом ответа. Вместо того, чтобы спрашивать респондента, насколько простой показалась ему система (это явно наводящий вопрос, поскольку его можно задать иначе с другим отношением к теме – «насколько сложной показалась вам система?»), лучше спросить, простой или сложный у системы интерфейс. Во-вторых, часто респонденты сами задают вам вопросы, стараясь избежать необходимости принимать решение самому. Ответить на такие вопросы легко, вот только ответы, из-за их спонтанности, будут наводящими. В таких случаях лучший ответ – встречный вопрос. Я справляюсь? – А как вы сами думаете? Я выполнил задание правильно? – А как вы сами считаете? И так далее, пока респондент не утомится. Невежливо, но действительно.

Кроме того, есть несколько не столь категоричных правил:

- Если во время теста вы следите за каким-либо свойством интерфейса, например, считаете ошибки респондента, не следует следить больше чем за одним показателем. К примеру, если вы считаете ошибки, время выполнения операций считать не стоит – вероятность вашей собственной ошибки слишком уж возрастает. На мой взгляд, во время теста можно только записывать свои гипотезы о потенциальных улучшениях интерфейса – т.е. то, что вы видите сразу же. Высчитывать показатели интерфейса лучше по видеозаписям.
- Даже при активном вмешательстве старайтесь не задавать респонденту вопросов, не относящихся напрямую к его текущей операции. Лучше задать их после теста.
- По возможности сидите справа сзади от респондента – так, чтобы он мог увидеть ваше лицо, слегка повернув голову. Ваше присутствие для респондента обременительно, но в таком положении он хотя бы будет менее напряжен.
- Во время теста вы часто не можете увидеть проблемы с интерфейсом в целом. Например, вы заметили ошибку пользователя. Но чем она объясняется? Аномалия ли это, вызванная тем, что пользователь менее подготовлен, чем остальные? Уверены ли вы в том, что эту ошибку повторяют все? Из-за этого фиксировать нужно максимум наблюдений. Некоторые вы отбросите впоследствии, но это лучше, чем упустить проблему.

Если вы параллельно ведете заметки, рекомендуется пользоваться следующими тактиками:

- **Скорость работы.** Между заданиями переводите секундомер на новый круг. Если респондент по какой-либо причине отвлекается, ставьте секундомер на паузу.
- **Ошибки.** На листе бумаги ставьте черточку при каждой человеческой ошибке. Удобно ставить мелкие черточки при мелких ошибках и длинные – при ошибках крупных. После теста достаточно посчитать количество черточек. Если вы раздельно считаете ошибки разных типов (например, просто ошибки и отдельно неправильно выбранные элементы меню) лучше пользоваться разными кодами, например, теми же черточками при простых ошибках и буквами М при ошибках, связанных с меню.
- **Проблемы, которые вы видите сразу же.** Кратко записывайте на бумажке сущность проблемы и текущее время (время первым!). Если вы будете точно знать, когда произошла проблема, вам будет легче найти соответствующий фрагмент видеозаписи.
- **Эмоциональные реакции респондента.** Ставьте знак плюса при положительных реакциях и знак минуса – при отрицательных. Реакции, происходящие в момент завершения тестовых заданий, не считаются.

Завершение теста

Закончив тест:

- Задайте респонденту накопившиеся вопросы.
- Дайте респонденту заполнить анкеты, если вы проводите анкетирование.
- Спросите у респондента, понравился ли ему интерфейс; независимо от ответа попросите уточнить, что именно понравилось и что не понравилось.
- Расплатитесь с респондентом.
- Поблагодарите его за участие в тесте. Заверьте респондента, что он справился прекрасно и что благодаря ему вы смогли выявить много интерфейсных проблем (сделайте это даже если респондент оказался замкнутым, неприятным в общении типом, тест на котором не выявил ничего нового).
- Если респондент особо хорош, спросите его, можно ли обращаться к нему в будущем для новых задач тестирования. Респондент с опытом тестирования всегда лучше такого же, но без опыта.

Тестирование на прототипах

Особо следует остановиться на тестировании на прототипах. При тестировании на прототипах у вас есть две возможности:

- Можно ограничиться тестом по методу активного вмешательства, лишаясь возможности получить какие-либо количественные данные. Для этого вам не нужен никакой специальный прототип, поскольку вы всегда имеете возможность объяснить респонденту значение служебной информации и причины лагун в прототипе.
- Можно вдобавок к обычному создать тестовый прототип и получить немного количественных данных, потеряв прорву ресурсов, затраченных на тестовый прототип.

Под тестовым прототипом подразумевается вид прототипа, в котором респондент может выполнить тестовые задания. Например, если прототип состоит из последовательности изображений экранов:

- В обычном прототипе на каждом изображении экрана будут представлены все возможные при разных обстоятельствах фрагменты интерфейса; вместо реальных данных будут выводиться обобщенные данные с пометками о том, какой у них возможен максимальный и минимальный объем.
- В тестовом прототипе тот же интерфейс будет представлен так, как он отображается в любой момент выполнения тестовых заданий: если на каком-либо экране по тестовому заданию производится редактирование данных, в прототипе придется нарисовать оба варианта этого экрана, а возможно и больше (состояние до, во время и после редактирования).

Создание тестового прототипа всегда трудоемко. Ладно еще, что экраны в таком прототипе приходится рисовать по несколько раз. Хуже другое – ограничиться созданием только тестового прототипа не получится, так что придется вам делать и обычный и тестовый прототипы. А если вы итерационно тестируете, придется еще вносить исправления в оба прототипа сразу.

Справедливости ради надо сказать, что создавать тестовые прототипы иногда все-таки приходится. Например, если на каком-то экране взаимодействие слишком сложное и вариативное, заказчик не сможет в нем разобраться (вам самому будет трудно это сделать). Единственным выходом в такой ситуации будет прорисовать этот экран во всех возможных состояниях со всеми возможными вариантами взаимодействия, т.е. создать тот же тестовый прототип.

Таким образом, использование нефункциональных тестовых прототипов есть контролируемый, но все-таки кошмар (собственно говоря, это и есть главный довод в пользу создания функциональных прототипов, которые изначально являются тестовыми). Нужен ли вам этот кошмар – решайте сами.

Анализ результатов

Наконец, пришло время для анализа результатов тестирования. Здесь важны три вещи:

- когда начинать анализ
- как анализировать действия респондента
- что можно почерпнуть из количественных данных.

Особняком стоит вопрос, когда начинать оптимизировать интерфейс.

Когда начинать анализ

Анализировать результаты можно как во время, так и после теста. Анализ во время тестирования имеет достоинства и недостатки. К достоинствам относится то, что он:

- Позволяет сэкономить время на этапе анализа, т.к. часть анализа производится во время более раннего этапа.
- Дает наиболее непосредственное впечатление от теста (гештальт), что позволяет увидеть проблемы, не замечаемые никаким другим способом.

Есть и недостатки:

- Не позволяет фиксировать более одного эргономического показателя за раз, да и то на практике можно замерять только скорость работы пользователя и число человеческих ошибок (правда, это самые востребованные показатели).
- Возможен только при значительном опыте юзабилити-специалиста.
- Невозможен, если тест проводит один специалист, а анализ делает другой (с другой стороны, этому другому человеку обязательно пригодятся наблюдения того, кто проводил тест).

Анализ после тестирования лишен этих недостатков и достоинств. Он позволяет внимательно и вдумчиво проанализировать материал, независимо от количества и сущности измеряемых показателей. Кроме того, он легко масштабируется на сколько угодно исполнителей.

Таким образом, оптимальной стратегией, по-видимому, является начало анализа во время теста. В некоторых случаях этим анализом можно и ограничиться. Если же это окажется невозможным, всегда можно проанализировать еще и видеопротоколы тестирования.

Анализ действий респондентов

Почти любое юзабилити-тестирование направлено на поиск и выявление проблем. Но как увидеть проблему в действиях респондентов?

Ошибки

Не любая ошибка респондента объясняется проблемами интерфейса, например, респондент мог проявить элементарную невнимательность. Тем не менее, любая ошибка требует рассмотрения:

- Если ошибка критическая, т.е. респондент ошибся из-за непонимания структуры интерфейса и ошибка привела к другим ошибкам (например, посетитель сайта перешел в ненужный ему раздел и там заблудился), соответствующий фрагмент должен быть переделан: нужно предпринять шаги по устранению неоднозначности, добавить подсказок и т.п.
- Если ошибка не критическая, т.е. респондент сразу ее заметил сам и сам же ее исправил, вам нужно решить, исправлять ли ее или оставить ее без внимания. Исправлять проблему стоит, если вы чувствуете, что понимаете, почему ошибка произошла (здесь вам поможет только опыт). Если не чувствуете, оставьте интерфейс как есть. Разумеется, если проблема повторится, ошибку нужно исправлять – но зато у вас будет больше информации о ней, так что исправлять будет легче.
- Возможно, ошибка объясняется несовершенством тестового задания. Обязательно убедитесь, что это не так – попросите респондента своими словами пересказать задание и если он ошибся, значит он понял задание неправильно и задание нужно срочно переделывать, ошибку же можно оставить без внимания.

Замедления выполнения задания

Если респондент без видимых причин приостановился, это значит, что он пытается понять, что ему нужно делать дальше. Вероятно, интерфейс недостаточно самоочевиден или однозначен. Проблему необходимо исправлять.

Увидеть замедление можно не только по собственно замедлению, оно как раз не всегда явно заметно, сколько по сопутствующим ему случайным движениям курсора мыши (многие люди, потеряв нить действия, машинально перемещают курсор).

Комментарии респондентов

Как правило, респонденты считают необходимым дать вам несколько советов и пожеланий («...а вот здесь было бы очень удобно...»). Принимать эти советы необязательно, поскольку респонденты:

- не обладают всей необходимой информацией о системе
- ничего не смыслят в проектировании интерфейсов
- их мотивация существенно отличается от необходимой – вместо того, чтобы стремиться сделать хороший интерфейс, они стремятся оставить в этом интерфейсе свой след («Вася здесь был»).

В то же время, пожелания являются признаком потребности в чем-то. Делать то, что предлагают респонденты, не стоит, но нужно задуматься, зачем нужно то, что они предлагают. Предлагаемые респондентами конкретные методы интерфейса, вероятно, ужасны (хотя бывают и исключения), но зато потребности, стоящие за этими методами, как правило, совершенно реальны, и их нужно удовлетворять, хотя и не так, как предлагают респонденты.

Анализ количественных данных

Сами по себе собранные количественные данные практически не способны показать *сущность* проблемы, они показывают только количество проблем.

С другой стороны, они могут быть использованы для сравнения старого и нового интерфейсов. Кроме того, сведенные вместе, они способны показать, где именно выявлены самые значительные проблемы – что полезно, т.к. начинать оптимизацию приятнее всего с областей, обещающих максимальную отдачу.

Когда вносить исправления

Если при тесте собираются количественные данные, вам необходимо провести тест на всех респондентах, прежде чем вы сможете начать править интерфейс. Если же сбор количественных данных не ведется, вы вполне можете вносить в интерфейс исправления после каждого респондента, не дожидаясь конца теста.

На мой взгляд, немедленное внесение исправлений имеет ряд преимуществ:

- Если исправление не сработало, т.е. проблема на его месте осталась (или появилась другая), у вас есть шансы исправить ее до конца теста.
- Ваши впечатления о тесте будут еще остры; а чем острее впечатления, тем глубже будут исправления.

Разумеется, чтобы оперативно вносить исправления, вам потребуется некоторое время между тестовыми сессиями и отлаженная технология внесения исправлений.

Если вы модифицируете интерфейс между тестами, не забывайте все отражать в журнале истории изменений. Начиная с некоторого момента вы будете возвращаться к старым решениям, от которых вы ранее отказались.

Подготовка отчета

После проведения тестирования необходимо передать обнаруженные сведения заказчику. Как правило, лучшим средством для этого является более или менее формальный отчет. Чтобы отчет получился эффективным, помимо, собственно говоря, качественно проведенного тестирования, нужно учитывать несколько вещей, главной из которых является оптимальная структура отчета.

В общем виде оптимальной структурой отчета является:

1. Резюме
2. Основные проблемы (интерфейсные проблемы, проявляющиеся по всему интерфейсу)
3. Частные проблемы (проблемы, проявляющиеся на отдельных экранах)
4. Количественные данные (если они собирались)
5. Приложение 1. Методика эксперимента и условия теста
6. Приложение 2. Описание тестовых сценариев
7. Приложение 3. Описание респондентов.

Ниже эти пункты расписаны более подробно.

Резюме

Юзабилити-специалиста, по сути, нанимают для того, чтобы он объективно и методично ругал интерфейс. Однако перед руганью протестированный интерфейс стоит вежливо похвалить – зачем заранее портить впечатление заказчика от отчета? Тем более, что в любом интерфейсе есть хоть что-то хорошее. Конкретные формулировки благостных слов привести трудно, но общее впечатление, которое должно остаться при их прочтении, таково – интерфейс замечательный, но многое стоит улучшить, чтобы он стал просто великолепным.

Кроме того, резюме должно содержать:

- Номер версии продукта
- Кто и когда проводил тест.

Представление проблем

Все выявленные проблемы нужно описывать в порядке убывания их важности. По возможности стоит также дать для каждой проблемы оценку ее деструктивности (в баллах). Без этого заказчику будет тяжело оценить, что требует исправления в первую очередь, а что можно оставить на потом.

Если выявленная проблема подтверждается собранными количественными данными, обязательно упоминайте это в описании проблемы. В любом случае, обязательно пишите, у каких респондентов она проявилась.

Если вы параллельно с тестированием проводите экспертную оценку (см. далее), обязательно отмечайте все рекомендации, основанные на ней – как-никак, все они могут оказаться ложными.

Вместо того, чтобы писать словами, на каких именно экранах обнаружены проблемы, используйте скриншоты, отмечая проблемные фрагменты именно на них. Если с вашей точки зрения одного скриншота достаточно, чтобы охарактеризовать проблему, ограничьтесь только им.

При написании этой части отчета соблюдайте следующие правила:

- Независимо от того, насколько формально пишется отчет, он должен быть написан возможно более официальным языком. Не должно быть жаргонизмов, фамильярности и вообще разговорного стиля.
- Недопустимы отступления от стандартного глоссария (как правило, это глоссарий Microsoft).
- Интерфейсные объекты нужно именовать так же, как они именуются в системе, недопустимы ни сокращения, ни синонимы.
- Когда дело касается веб-сайтов, следует писать «посетитель», когда речь идет о традиционном ПО или о веб-интерфейсе, нужно писать «пользователь».
- Английские наименования, которых, увы, слишком много в отчетах, должны оставаться английскими, их недопустимо ни транскрибировать, ни, тем более, склонять (не «поиск на Яндексe», а «поиск на Yandex.ru»).
- Не забывайте проверять орфографию! Полезно также попросить кого-нибудь прочесть отчет и отметить в нем непонятные места – заказчику они тоже будут непонятны.

Обратите также внимание на следующие соображения:

- Отчет нужен не для того чтобы обеспечить «солидность» тестирования, а для коммуникации с заказчиком. Бесполезно специально увеличивать объем отчета в надежде, что пухлая пачка бумаги более угодна заказчику. Не содержащий воды отчет всегда лучше коммуницирует себя, чем отчет разбухший; а качество коммуникации вашего отчета – первая предпосылка доверия к результатам юзабилити-тестирования.
- При утверждении отчета вы обязаны уклоняться от всех попыток заказчика дезавуировать результаты тестирования. Если заказчику что-то не нравится, он должен был ругаться раньше, когда утверждал тестовые сценарии и набор респондентов – результаты-то объективные. Напротив, вы должны проявлять максимальное уважение к мнению заказчика относительно проблем, выявленных при экспертной оценке, т.к. ее результаты ничем не подтверждены.

Количественные данные

Четких рекомендаций по представлению количественных данных дать невозможно, поскольку для каждой метрики нужно свое представление. Ограничусь тремя рекомендациями:

- Для каждого типа данных создавайте отдельную таблицу, при этом в первую таблицу раздела складировать данные по всем метрикам.
- Если можно нарисовать график, рисуйте график. Для заказчика график бесполезен (заказчики, как показывает опыт, вообще не интересуются

неинтерпретированными данными – и правильно делают), но зато он полезен вам, так как, рисуя его, вы сможете лучше понять, почему результаты именно таковы.

- Почти в любом тесте среди прочего измеряются успешность и количество ошибок. Для этих метрик удобна таблица следующего вида:

	1	2	3	4	5	Среднее
Виктор П.	3/4	1/13	1/4	0/4	1/2	1.2/5.4
Наталья Б.	2/5	0/4	0	2/12	1/3	1/4.8
Сергей К.	3/4	3/9	0/4	1/8	1/2	1.6/5.4
Елена Г.	3/5	2/9	2/1	4/19	2/4	2.6/7.6
Среднее	2.8/4.5	1.5/8.8	0.8/2.3	1.8/10.8	1.3/2.8	

где по вертикали отложены тестовые задания, а по горизонтали – респонденты.

В каждой ячейке приведено количество человеческих ошибок (крупные и незначительные отдельно), цветами удобно отмечать проблематичные задания:

- одним цветом задания, от выполнения которых респонденты отказались
- другим – задания, которые респонденты не смогли выполнить полностью
- третьим – выполненные неправильно
- четвертым цветом – задания, выполнить которые не удалось по техническим причинам (читай – «сервер лежал»).

Приложения

Кратко опишите используемую методику эксперимента (лучше даже ничего не писать своего, а сослаться на какую-либо публикацию). Не забудьте описать условия, в которых проходило тестирование (место, оборудование и т.п.) Тестовые сценарии достаточно привести как есть.

При описании респондентов нужно ограничиться их именами (без фамилий), описанием возраста, опыта использования компьютера и частными их свойствами, которые, на ваш взгляд, повлияли на выполнение ими тестовых заданий (четких рекомендаций тут дать нельзя, но, например, если один из респондентов допустил заметно меньше ошибок, чем остальные, стоит попытаться объяснить этот феномен).

Приложение 1. Экспертная оценка

Крупный недостаток юзабилити-тестирования – высокая стоимость (которая даже и при снижении качества работы категорически не желает падать). Предпринимались многочисленные попытки если уж не снизить стоимость, то по крайней мере придумать альтернативные методы, пусть похуже, но подешевле.

Так появились такие методы, как эвристическая оценка, мысленный прогон и т.д. (см. ниже), в общем виде называемые экспертной оценкой. К сожалению, оказалось, что экспертная оценка не намного дешевле тестирования, так что поставленных перед ней

целей она не выполняет. Обнаружилось однако, что забыть об экспертной оценке не получается – в некоторых случаях полноценное юзабилити-тестирование оказывается невозможным и приходится пользоваться тем, что есть.

Итак, экспертной оценкой (далее – ЭО) называется определение проблем интерфейса через его оценку профессиональным дизайнером интерфейса или юзабилити-специалистом. Поскольку речь о каком-либо эксперименте на реальности не идет, ЭО не может считаться юзабилити-тестированием. Рассматривать ЭО в этом разделе получается некорректным, однако, поскольку часть подготовительной работы у тестирования и у ЭО одинакова, экспертная оценка разбирается именно здесь. Кроме того, практика показывает, что ЭО дает результаты, в некоторых отношениях более интересные, нежели тестирование – и уж в любом случае дополняющие результаты теста.

Достоинства и недостатки экспертной оценки

Ярче всего достоинства экспертной оценки проявляются в сравнении ее с юзабилити-тестированием.

Ключевой вопрос – что находит тестирование? Ответ – крупные проблемы интерфейса. Если используется огромная выборка, удастся выявить и часть мелких. Но если тестирование показывает только плохие фрагменты интерфейса, как можно выявить фрагменты недостаточно хорошие? Как тестирование может показать, к примеру, некорректные фрагменты интерфейса, недостаточно серьезные, чтобы вызвать человеческие ошибки, но снижающие скорость работы?

Такие проблемы очень напоминают смазку в двигателе: едет, положим, и без смазки, но со смазкой едет сильно лучше.

Опыт же эксперта позволяет выявить их без труда:

- Эксперт хорошо знает стандарты на интерфейс, как писанные, так и неписанные. Сами по себе нарушения стандартов не плохи и не хороши, но если они не оправданы, т.е. появились случайно, их лучше устранить.
- Эксперт заведомо внимателен и, что гораздо важнее, смотрит на интерфейс свежим взглядом. Это позволяет ему находить множество мелочей, балансирующих между багом и проблемой интерфейса (например, несогласованность терминологии), которые самим разработчикам уже не видны.
- Эксперт располагает опытом, позволяющим ему безошибочно определять проблемы, которые проявлялись в его профессиональной деятельности раньше. Если какое-то решение не сработало в прошлом, оно, по видимости, не сработает и в будущем. Удивительное свойство таких решений – они повторяются снова и снова. Эксперт знает множество таких решений, хотя и не всегда понимает, почему они неработоспособны.
- Эксперт владеет формальными методами оценки интерфейса, способными даже без тестирования выявить неэффективные решения (например, GOMS). Применение этих методов при ЭО позволяет найти значительно больше проблем, чем можно увидеть, просто изучая интерфейс.
- Качество ЭО может быть сильно улучшено, если увеличить количество экспертов. Поскольку подготовительную работу может провести единственный эксперт, стоимость ЭО с несколькими экспертами ниже суммы гонораров экспертам, выплаченных по отдельности.

В то же время экспертная оценка обладает и рядом недостатков:

- Часть проблем, выявленных при ЭО, проблемами, собственно, не является – синдром «мне так показалось». Если плохо проведенное тестирование не показывает всех проблем, то плохо проведенная оценка не только не показывает часть проблем, но, что еще хуже, показывает проблемы несуществующие. Исправление этих квази-проблем стоит денег, но избежать появления их в отчете по ЭО, к сожалению, невозможно. Хорошо еще, что чем опытнее эксперт, тем меньше квази-проблем он «находит».
- ЭО неспособна выявить уникальные проблемы конкретного интерфейса (они уникальны, а значит, эксперт с ними еще не сталкивался).
- Применение формальных методов анализа интерфейса улучшает результат, но также значительно увеличивает стоимость ЭО.
- Экспертная оценка обладает обманчивой легкостью, делающей ее особенно опасной в неопытных руках. Кажется, что она доступна любому. Печальная правда в том, что она потому и экспертная оценка, что выполнять ее может только эксперт и никто другой.
- Количество выявленных проблем при ЭО зависит не только от опыта экспертов, но и от их количества, из-за чего заказывать работу приходится у нескольких разных исполнителей. В результате для заказчика ЭО организационно сложнее юзабилити-тестирования.

Виды экспертной оценки

Теоретически, все просто. Эксперт разглядывает интерфейс и записывает найденные проблемы. Проблема в том, что так можно увидеть отступления от стандарта и только. Чтобы сделать поиск проблем результативным, нужен хоть сколько-нибудь формальный подход.

Наиболее распространенные формальные подходы:

- проверка по контрольному списку
- эвристическая оценка
- мысленная прогонка по интерфейсу.

Эти методы довольно хороши. В то же время я полагаю, что существует более эффективный метод. Я называю его анализом задач/характеристик.

Проверка по контрольному списку

Проверка по контрольному списку ближе всего к формальному тестированию качества. Составляется список произвольных требований, после чего интерфейс проверяется на соответствие этим требованиям. В контрольном списке может быть все что угодно, как правило, туда прямым ходом отправляются требования из стандартов и руководств по проектированию интерфейсов.

Метод, вообще говоря, чрезвычайно надежен при поиске дурацких проблем, вроде грамматических ошибок в интерфейсных тестах. К сожалению, он имеет два принципиальных недостатка:

- Чем более полон и детален контрольный список, тем дольше производится проверка. В хорошем списке может быть больше сотни пунктов (в моем

собственном [контрольном списке для интерфейсов](#) одно время было больше ста пятидесяти требований). Проверки должны выполняться последовательно, больше чем один пункт за раз проверять нельзя. Предположив, что одна проверка занимает пятнадцать минут, можно посчитать длительность полного исследования.

- Контрольный список, чтобы быть совершенно надежным, не имеет права содержать сколько-нибудь размытых требований. Например, требование «На растягивающихся окнах есть индикатор растягиваемости» совершенно четкое. А вот требование «Таблицы легко читаются» допускает изрядный простор для мухляжа, поскольку невозможно определить, что такое легко читающаяся таблица. К сожалению, достаточно полный контрольный список, содержащий только строго определенные требования, есть явление недостижимое.

Кроме того, качество проверки по контрольному списку напрямую зависит от качества самого списка. Чтобы создать качественный список требований, нужно затратить годы труда – и самое обидное, что этот список никогда не будет завершен, поскольку все время проявляются новые требования.

С другой стороны, четкий контрольный список может использоваться кем угодно, даже и совершеннейшими профанами в проектировании интерфейсов – как-никак вся необходимая информация уже содержится в списке. В крупных организациях это дает возможность вынести проверку интерфейса из деятельности юзабилити-специалиста, передав ее отделу контроля качества.

Эвристическая оценка

Эвристическая оценка была разработана Якобом Нильсеном и Рольфом Моличем, которые надеялись с ее помощью сократить продолжительность проведения проверки по контрольному списку. При эвристической оценке вместо десятков и сотен конкретных требований интерфейс проверяется на соответствие всего нескольким общим принципам. Сам Нильсен (Молич впоследствии самоустранился) рекомендовал следующие принципы:

- В любой момент времени система показывает, что с ней происходит.
- Система использует термины, понятия и метафоры, присутствующие в реальном мире, а не обусловленные компьютером.
- В любой момент пользователь контролирует систему, а не наоборот. Любую команду можно отменить или повторить.
- В любой момент времени система выглядит и функционирует единообразным и стандартным способом.
- Интерфейс системы препятствует появлению человеческих ошибок.
- В любой момент времени интерфейс показывает объекты и команды сам, не требуя от пользователя вспоминать их.
- В интерфейсе есть методы ускорения работы, предназначенные для опытных пользователей и не мешающие пользователям неопытным; благодаря таким ускорителям опытные пользователи получают резерв для повышения собственной производительности.
- Интерфейс эстетичен и в любой момент времени не содержит ненужной сейчас информации.
- Интерфейс помогает пользователям обнаруживать и исправлять проблемы, включая человеческие ошибки.
- Справка доступна в любой момент времени. Она достаточна, но не избыточна; к ней легко обращаться; она не абстрактна, а нацелена на решение конкретных задач пользователя; в ней описываются конкретные шаги по решению проблем.

Подробнее об эвристической оценке см. [сайт Якоба Нильсена \(en\)](#).

Эти эвристики были протестированы самим Нильсеном, тест показал, что они достаточно результативны. В то же время я вижу две проблемы этого метода.

Во-первых, я полагаю, что ничего не принесло столько вреда HCI, как эвристическая оценка. Сам Нильсен в явной форме утверждает, что оценку должны проводить от трех до пяти экспертов, иначе она не будет сколько-нибудь результативной. Проблема в том, что из-за кажущейся простоты и доступности метода, требование к числу экспертов постоянно игнорируется, кстати, как и то, что проводить оценку должны именно эксперты. В результате для многих заказчиков, испробовавших этот метод, вся дисциплина юзабилити оказывается полностью дискредитированной.

Во-вторых, эвристическая оценка никак не учитывает деятельности пользователей, в лучшем случае она покрывает выполняемые пользователями операции. В результате эвристическая оценка не выявляет структурные проблемы интерфейса, которые, как правило, заметно важнее локальных проблем.

В целом, я считаю, что эвристическая оценка устарела. Это вполне простительно, если учесть, что метод появился уже пятнадцать лет назад. С тех пор на первый план вышла проблематика деятельности, а вот внимание к частным интерфейсным механизмам, напротив, снизилось. Кроме того, с тех пор юзабилити-тестирование сильно подешевело, что делает эвристическую оценку и вовсе ненужной.

Мысленная прогонка по интерфейсу

Если предыдущие методы формализовали правила, которым должен следовать интерфейс, то мысленная прогонка формализует метод, согласно которому интерфейс оценивается.

Если исходить из того, что интерфейс предназначен для использования функций, можно проверить, как эти функции вызываются и используются. Если просто проговорить словами, как работают интерфейсы всех функций, становится понятно, какие из них неоправданно подавлены, а какие работают недостаточно хорошо. Разумеется, для этого тоже необходим опыт эксперта.

Проблема есть и здесь. Хотя метод неплохо работает при простом проговаривании работы функций, наилучшие результаты он дает, если функции не просто проговариваются, а записываются («мысль изреченная есть ложь»). Сам факт записи позволяет увидеть множество противоречий и недостатков в интерфейсе. Увы, запись требует очень много времени, что не позволяет использовать этот метод всегда и всюду.

Анализ задач/характеристик

Крупной, на мой взгляд, проблемой вышеперечисленных методов оценки является игнорирование самого интересного – пользовательских задач. Без знания этих задач можно оптимизировать разве что «интерфейс-в-себе», которого, согласно определению юзабилити из ISO 9241-11, быть не может. Кроме того, хотелось бы учитывать еще и характеристики целевых пользователей, равно как и контекст использования.

Кроме того, я полагаю, что искать проблемы более продуктивно, если концентрироваться не на самих проблемах, а сразу на способах их решения. Можно просто рассматривать

интерфейс, стараясь увидеть проблемы, а можно рассматривать его, задавая себе вопрос «как это улучшить?». Второй метод, на мой взгляд, более производителен.

Соответственно, анализ задач/характеристик метод заключается в следующем. Сначала составляются список пользовательских задач, описание контекста использования и краткая характеристика пользователей (например, в формате персонажей). Затем все окна, экраны и пользовательские задачи (т.е. маршруты пользователей по интерфейсу) последовательно рассматриваются под углом следующих вопросов:

- Как здесь ускорить работу данных пользователей в данном контексте?
- Как здесь снизить число ошибок данных пользователей в данном контексте?
- Как здесь повысить удовлетворенность данных пользователей в данном контексте?
- Как здесь сделать интерфейс самопонятнее данным пользователям, чтобы повысить скорость обучения в данном контексте?

Все предлагаемые изменения, как следствие, находятся на местах каких-либо проблем, каковые и следовало выявить. Что особенно хорошо, сразу же становится ясно, как эти проблемы исправлять.

Выводы

Суммируя, можно сказать, что экспертная оценка – метод, конечно, неполноценный, но достаточно эффективный и замечательно дополняющий юзабилити-тестирование. Использовать ее без тестирования разумно только в ситуациях, когда подобрать адекватных респондентов для тестирования по каким-либо причинам невозможно.

С другой стороны, я полагаю неэффективным «чистое» юзабилити-тестирование, лишенное параллельной экспертной оценки. Если проводить и тестирование, и оценку, результаты становятся наиболее полными. Тем более, что составление списка пользовательских задач, формализация контекста использования и целевой аудитории продукта делаются и при тестировании, и при оценке. Проведенные вместе, они обходятся значительно дешевле оценки и тестирования, проведенных по отдельности.

Кроме того, в запущенных случаях тестировать интерфейс неэффективно: без всякого тестирования в нем видно огромное количество проблем. Эти проблемы, конечно, будут подтверждены тестированием, но при этом они заслонят собой множество других проблем. После внесения изменений придется тестировать заново. Если же сначала провести не особо формальную экспертную оценку, исправить ошибки и протестировать уже обновленный интерфейс, тестирование будет наиболее эффективным.

Приложение 2. Планирование юзабилити-тестирования

Несмотря на то, юзабилити-тестирование является только частью проектного цикла, оно и само является отдельным проектом, обладающим собственной стоимостью и длительностью. А значит, ему требуется планирование, иначе качественного результата не достичь.

Юзабилити-тестирование состоит из следующих этапов:

- оценка объема работы
- формирование списков требований к респондентам и задач пользователей
- поиск респондентов

- составление тестовых сценариев
- тестирование теста
- проведение тестирования
- анализ результатов
- экспертная оценка интерфейса
- написание итогового отчета.

Ниже эти этапы описаны подробно.

Оценка объема работы

Объем работы при юзабилити-тестировании зависит от четырех факторов: объема системы, количества пользовательских задач, желаемого числа респондентов и количества измеряемых метрик. Соответственно, нужно заранее определить границы проекта. Для этого:

- Посчитайте число экранов и окон системы и полученную сумму увеличьте на 20% (часть экранов вы неизбежно упустили при подсчете).
- Примерно посчитайте число пользовательских задач.
- Определитесь, на скольких респондентах вы хотите проводить тест. Учтите, что если вы выявили более пятнадцати пользовательских задач, протестировать их все на одном респонденте не получится, так что придется делить задачи между респондентами, а значит их понадобится больше. Если число респондентов слишком уж выросло, лучше сокращать не число задач, а количество респондентов на малозначимых задачах (например, задачи 1 и 2 тестировать на всех респондентах, задачу 8 – только на трех респондентах, а остальные задачи – на случайно выбранных респондентах, но чтобы одна задача предъявлялась минимум пяти респондентам).
- Определитесь, какие эргономические показатели вы собираетесь измерять. Поскольку сбор количественных данных очень трудоемок, в случае проведения теста на большом количестве респондентов допустимо собирать данные только на некоторых из них.

Кроме того, важно определиться, нужно ли проводить формальную экспертную оценку. Даже без нее вы заметите много проблем, не экстраполируемых напрямую из наблюдаемого поведения респондентов. Их надо описывать в финальном отчете в любом случае. Однако впридачу вы можете и намеренно проанализировать интерфейс – но это займет некоторое время и должно быть запланировано заранее.

В целом, отведите не менее одного рабочего дня только на оценку объема работы и планирование.

Формирование списков требований к респондентам и пользовательских задач

Если количество экранов и окон системы не превышает пятнадцати, выделите на эту работу один рабочий день. Если экранов больше, смело выделяйте пропорционально больше времени. Помните, что неправильно выбранные задачи или респонденты делают тестирование бессмысленным – даже избыточная тщательность здесь не роскошь.

Поиск респондентов

Предсказать длительность подбора респондентов можно только накопив внушительную базу с телефонами готовых к сотрудничеству людей. Если же аудитория продукта специфическая, не поможет даже и это.

Длительность поиска респондентов в часах примерно показывает формула $X + Y * 4$, где X – число респондентов, к которым у вас есть прямой доступ (ваши друзья и близкие, контакты из базы респондентов), а Y – недостающие респонденты, которых нужно найти. Понятно, что при специфической аудитории тестируемой системы первой частью формулы воспользоваться не получится.

Но это чистое время работы, реальная продолжительность этапа гораздо больше. При специфической аудитории добавьте к ожидаемой длительности поиска неделю, при очень специфической – две.

Составление тестовых сценариев

На преобразование в сценарий одной пользовательской задачи уходит около часа работы.

Пилотный тест

Разумно запланировать два цикла тестирования самого теста, т.е. как минимум на половину рабочего дня.

Проведение тестирования

Собственно продолжительность тестирования в часах вычисляется по формуле $X * (Y + 1) + (X / 4)$, где X – число респондентов, Y – продолжительность теста. Дополнительный час на респондента нужен в расчете на опоздания респондентов и на проведение кратких интервью с ними. Кроме того, примерно четверть респондентов будет выполнять тестовые задания очень медленно и не уложатся в график.

Учтите, что при планировании сеансов тестирования между ними нужно планировать двухчасовые перерывы – во-первых, это позволит юзабилити-специалисту отдохнуть, во-вторых, снизит вероятность нарушения хода теста из-за непредвиденных ситуаций.

Кроме того, если вы собираетесь править интерфейс после каждого респондента, вам понадобится дополнительное время между сессиями. Продолжительность внесения исправлений зависит от вашей технологии программирования/прототипирования, так что выбирайте ее сами.

Анализ результатов

Продолжительность анализа результатов напрямую зависит от числа респондентов, длительности одного теста и, сильнее всего, от количества замеряемых показателей. Как следствие, ее длительность варьируется очень сильно.

Если вы проводите тест без замера количественных показателей и начинаете анализ еще во время тестирования, продолжительность этого этапа может быть нулевой – все важное вы уже успеете увидеть и раньше.

Если же нужно измерять какие-либо количественные показатели, продолжительность анализа в часах вычисляется по формуле $X * Y * (Z - 1)$, где X – число респондентов, Y –

продолжительность теста, Z – количество показателей. Если вы начинаете анализ уже после завершения тестирования, вычитать единицу из Z не надо.

Экспертная оценка интерфейса

Начинать формальный анализ интерфейса нужно уже после того, как вы проведете тестирование и проанализируете его результаты – в этом случае вы не запутаетесь, где объективные данные, а где данные субъективные (как показывает опыт, перепутать их очень легко). К тому же в этом случае глубоко анализировать можно будет не весь интерфейс, а только выявленные тестом проблематичные фрагменты.

Длительность экспертной оценки в часах можно примерно предсказать следующей формулой $X * 0.3 + X + Y * 0.6$, где X – число экранов и окон системы (на большую часть экранов системы потребуется минут десять, но примерно пятая часть потребует больше времени), а Y – количество пользовательских задач (они требуют отдельного прохода по интерфейсу).

Написание итогового отчета

Несмотря на то, что уже при анализе результатов образуется большое количество текста, необходимо заложить как минимум один рабочий день на приведение его в порядок и вычитку.

Рекомендуемая литература

- [A Practical Guide to Usability Testing](#). Joseph S. Dumas, Janice C. Redish. Intellect, Ltd (UK), 1999. ISBN 1841500208.
- [Handbook of Usability Testing: How to Plan, Design, and Conduct Effective Tests](#). Jeffrey Rubin. Wiley, 1994. ISBN 0471594032.
- На русском языке существуют две переводные публикации, посвященные беглому, но очень широкому, описанию юзабилити-методов: [Методы веб-дизайна и юзабилити](#) и [The Usability Methods Toolbox](#).

Благодарности

Без помощи многих людей, прочитавших эту статью, она была бы ощутимо хуже. Приношу свою благодарность своим коллегам по Usethics, моему отцу, Кириллу Вятчину, Ярославу Перевалову и Дмитрию Сатину.