

Сертифицированный тестировщик

Тестирование мобильных приложений ISTQB®

Программа обучения Базового уровня

Версия 2019

Международный институт качества программного обеспечения (iSQI)

Международный квалификационный совет по сертификации качества тестирования
программного обеспечения (ISTQB)



Уведомление об авторских правах

Этот документ может быть скопирован целиком или частично, если указано авторство.

Уведомление об авторских правах © International Software Testing Qualifications Board (далее просто ISTQB®)

ISTQB® является зарегистрированной торговой маркой International Software Testing Qualifications Board.

Авторы Программы обучения Сертифицированного специалиста по тестированию мобильных приложений Базового уровня – Jose Diaz, Rahul Verma, Tarun Banga, Vipul Kocher и Yaron Tsubery - передали свои права ISTQB®. Эта программа обучения была использована в качестве основы при создании текущего документа.

Авторские права © 2019 авторы перевода 2019 (Маргарита Трофимова (руководитель группы), Александр Александров (редактор), Елизавета Кручинина, Ольга Овчинникова, Александр Торговкин).

Авторские права © 2019, авторы Vipul Kocher (председатель), Piotr Wicherski (заместитель председателя), José Díaz, Matthias Hamburg, Eran Kinsbruner, Björn Lemke, Samuel Ouko, Ralf Pichler, Nils Röttger, Yaron Tsubery.

Этот документ был подготовлен основной командой Рабочей группы по Тестированию мобильных приложений International Software Testing Qualifications Board.

Vipul Kocher (председатель), Piotr Wicherski (заместитель председателя), José Díaz, Matthias Hamburg, Eran Kinsbruner, Björn Lemke, Samuel Ouko, Tal Pe'er, Ralf Pichler, Lloyd Roden, Nils Röttger, Angelina Samaroo, Yaron Tsubery.

Авторы передают свои права International Software Testing Qualifications Board (ISTQB®). Авторы (владельцы авторских прав в данный момент) и ISTQB® (как будущий владелец авторских прав) договорились о следующих условиях использования:

Любое частное лицо или обучающая компания могут использовать эту программу как основу для проведения обучающих курсов, если авторы и ISTQB® упомянуты как источник и владельцы авторских прав, при этом в любой рекламе таких курсов данная программа может быть упомянута только после письменного уведомления об аккредитации материалов тренингов коллегией, признанной ISTQB®.

Любое частное лицо или группа частных лиц может использовать программу как основу для статей, книг или других производных письменных материалов если авторы и ISTQB® упомянуты как источник и владельцы авторских прав программы.

Любая коллегия, признанная ISTQB®, может перевести эту программу (или ее перевод) для других участников.

История изменений

Версия	Дата	Содержание
Альфа	11 мая 2018	Альфа версия
Бета	27 января 2019	Бета версия
GA	28 марта 2019	Общедоступная версия
V2019	10 мая 2019	ISTQB® версия
RSTQB 2019	8 февраля 2020	Перевод на русский язык

С о д е р ж а н и е

История изменений	3
Предисловие к русскому переводу	7
Благодарности	8
0. Предисловие к программе обучения	9
0.1 Цель этого документа	9
0.2 Сертифицированный тестировщик мобильных приложений Базового уровня	9
0.3 Бизнес-результаты	9
0.4 Проверяемые цели обучения	10
0.5 Уровни практических компетенций	10
0.6 Экзамен	11
0.7 Рекомендуемое время обучения	11
0.8 Требования для допуска к экзамену	11
0.9 Источники информации	11
1. Мир мобильных устройств – Ключевые элементы бизнеса и технологий - 175 мин	12
1.1 Данные мобильной аналитики	13
1.2 Бизнес-модели для мобильных приложений	13
1.3 Типы мобильных устройств	14
1.4 Типы мобильных приложений	14
1.5 Архитектура мобильных приложений	16
1.6 Стратегии тестирования мобильных приложений	17
1.7 Сложности тестирования мобильных приложений	18
1.8 Риски тестирования мобильных приложений	19
2.1 Тестирование совместимости с аппаратным обеспечением устройства	22
2.1.1 Тестирование функций устройства	22
2.1.2 Тестирование разных экранов	23
2.1.3 Тестирование температуры устройства	23
2.1.4 Тестирование датчиков ввода данных	23
2.1.5 Тестирование различных способов ввода данных	24
2.1.6 Тестирование изменения ориентации экрана	24
2.1.7 Тестирование типичных прерываний	24
2.1.8 Тестирование прав доступа к функциям устройства	25
2.1.9 Тестирование энергопотребления и состояния	25
2.2 Тестирование взаимодействия приложения с ПО устройства	25
2.2.1 Тестирование уведомлений	25
2.2.2 Тестирование ссылок быстрого доступа	26
2.2.3 Тестирование пользовательских настроек, предоставляемых операционной системой	26

2.2.4 Тестирование разных типов приложений	26
2.2.5 Тестирование возможности взаимодействия с несколькими платформами и версиями операционной системы	27
2.2.6 Тестирование возможности взаимодействия и сосуществования с другими приложениями на устройстве	27
2.3 Тестирование различных методов подключения	27
3. Распространенные типы тестирования и процесс тестирования мобильных приложений – 200 минут	29
3.1 Распространенные типы тестирования, применимые к мобильным приложениям	30
3.1.1 Установочное тестирование	30
3.1.2 Стрессовое тестирование	31
3.1.3 Тестирование защищенности	31
3.1.4 Тестирование производительности	32
3.1.5 Тестирование практичности	32
3.1.6 Тестирование баз данных	33
3.1.7 Тестирование глобализации и локализации	33
3.1.8 Тестирование доступности	33
3.2 Дополнительные уровни тестирования, применимые к мобильным приложениям	33
3.2.1 Тестирование в условиях эксплуатации	34
3.2.2 Тестирование для получения разрешения от магазина приложений и пост-релизное тестирование	34
3.3 Методики тестирования на основе опыта	34
3.3.1 Тест-персоны и мнемоники	34
3.3.2 Эвристика	35
3.3.3 Тест-туры	35
3.3.4 Управление тестированием на основе сеансов	37
3.4 Процесс тестирования и подходы к тестированию мобильных приложений	38
3.4.1 Процесс тестирования	38
3.4.2 Подходы к тестированию	39
4. Платформы, инструменты и окружение мобильных приложений – 80 минут	40
4.1 Платформы разработки мобильных приложений	40
4.2 Общие инструменты платформ разработки	41
4.3 Эмуляторы и симуляторы	41
4.3.1 Обзор эмуляторов и симуляторов	41
4.3.2 Использование эмуляторов и симуляторов	41
4.4 Создание тестовой лаборатории	42
5. Автоматизация выполнения тестов – 55 минут	44
5.1 Подходы к автоматизации	44
5.2 Методы автоматизации	45
5.3 Оценка инструментов автоматизации	46

5.4 Подходы к созданию лаборатории автоматизации тестирования	46
6. Список использованных источников	48
6.1 Документы ISTQB®	48
6.2 Литература	48
6.3 Книги и статьи для дальнейшего изучения	48
6.4 Ссылки (Веб/Интернет)	48
7. Приложение А – Цели обучения/Уровень знаний	50
7.1 Уровень 1: Запомнить (K1)	50
7.2 Уровень 2: Понять (K2)	50
7.3 Уровень 3: Применить (K3)	50
8. Приложение Б – Глоссарий специальных терминов	52

Предисловие к русскому переводу

Уважаемые коллеги!

Вашему вниманию предлагается программа обучения ISTQB «Сертифицированный тестировщик Базового уровня. Тестирование мобильных приложений».

Особенностей данной программы обучения, побудивших нас создать настоящее предисловие, две:

- Области тестирования мобильных приложений как самостоятельной инженерной деятельности не более 10 лет. В ней еще не сформировались общепринятые понятия, термины, принципы и традиции. Поэтому настоящий документ отражает скорее текущее состояние, чем итоги и анализ опыта многолетнего применения упомянутой инженерной деятельности. В силу этого документ не претендует на полноту, отражая прежде всего конкретный опыт авторов оригинального документа, перевод которого мы представляем. Мы посчитали полезным подготовить этот перевод, доступный русскоязычной аудитории, будучи уверенными, что в дальнейшем документ будет радикально расширен и детализирован.
- В рабочей группе было немало споров по поводу терминологии. С одной стороны, мы считаем неприемлемым использовать калькированные термины, следуя опыту известной переводчицы и редактора Норы Галь (см., например, «Слово живое и мёртвое»). С другой стороны, не следует открещиваться от общепринятой практики использования калькированных терминов в стремительно развивающейся индустрии разработки ПО. В итоге мы решили указывать рядом с предлагаемым термином его общепринятое использование, пометив его как жаргонное.

Любые замечания и предложения, способствующие улучшению качества настоящего документа принимаются нами с благодарностью. Но при этом следует иметь в виду, что предлагаемая программа обучения должна быть максимально точным переводом оригинальной англоязычной программы обучения. Такovy правила, которые мы не можем и не хотим менять.

От лица рабочей группы по переводу
Председатель RSTQB
Андрей Конушин

Благодарности

Перевод версии документа 2018 года выполнен Рабочей группой Базового Уровня АНО «Коллегия экспертов по качеству программного обеспечения» (Russian Software Testing Qualifications Board): Маргарита Трофимова (руководитель группы), Александр Александров (редактор), Елизавета Кручинина, Ольга Овчинникова, Александр Торговкин.

Этот документ был подготовлен основной командой Рабочей группы по Тестированию мобильных приложений International Software Testing Qualifications Board.

Vipul Kocher (председатель), Piotr Wicherski (заместитель председателя), José Díaz, Matthias Hamburg, Eran Kinsbruner, Björn Lemke, Samuel Ouko, Tal Pe'er, Ralf Pichler, Lloyd Roden, Nils Röttger, Angelina Samaroo, Yaron Tsubery.

Благодарим команду редакторов за их предложения и вклад в подготовку данного документа.

В редактировании, комментировании и голосовании по этой программе участвовали:

Graham Bath, Veronica Belcher, Armin Born, Geza Bujdoso, YongKang Chen, Wim Decoutere, Frans Dijkman, Florian Fieber, David Frei, Péter Földházi Jr., Chaonian Guo, Attila Gyuri, Ma Haixia, Matthias Hamburg, Zsolt Hargitai, Hongbiao Liu, Ine Lutterman, Marton Matyas, Petr Neugebauer, Ingvar Nordström, Francisca Cano Ortiz, Nishan Portoyan, Meile Posthuma, Emilie Potin-Suau, Liang Ren, Lloyd Roden, Chaobo Shang, Mike Smith, Péter Sótér, Marco Sogliani, Michael Stahl, Chris Van Bael, Paul Weymouth, Salinda Wickramasinghe, Minghui Xu.

Этот документ официально опубликован ISTQB® 10 мая 2019 г.

0. Предисловие к программе обучения

0.1 Цель этого документа

Программа обучения представляет собой основу сертификации на квалификацию Базового уровня в области тестирования мобильных приложений. ISTQB® предоставляет эту программу:

1. Национальным коллегиям для перевода ее на национальный язык и аккредитации организаторов обучения. Национальные коллегии могут приспособить программу обучения к особенностям конкретных языков и модифицировать ссылки для адаптации к местным публикациям.
2. Экзаменационным комиссиям для формирования экзаменационных вопросов на национальном языке, адаптированных к целям обучения для каждой программы.
3. Организаторам обучения для разработки учебной программы и определения подходящих методов обучения.
4. Кандидатам на получение сертификатов для подготовки к экзамену (либо в рамках учебного курса, либо самостоятельно).
5. Международному сообществу разработки ПО и систем для повышения уровня профессионализма при тестировании ПО и систем и использования как основы для книг и статей.

ISTQB® может разрешить другим лицам использовать эту программу обучения в своих целях при условии, что они обратятся и получат письменное разрешение.

0.2 Сертифицированный тестировщик мобильных приложений Базового уровня

Квалификация Базового уровня предназначена для всех людей, связанных с тестированием ПО, которые желают расширить свои знания в области тестирования мобильных приложений и начать карьеру специалиста по тестированию мобильных приложений.

Информация о тестировании мобильных приложений, описанная в Программе обучения Сертифицированного тестировщика Базового уровня ISTQB®, была учтена при создании этой программы.

0.3 Бизнес-результаты

В этом разделе перечислены бизнес-результаты, ожидаемые от кандидатов, которые прошли сертификацию по тестированию мобильных приложений Базового уровня.

MAT-01	Понимать и рецензировать драйверы бизнеса и технологий мобильных приложений для определения стратегии тестирования.
MAT-02	Определять и понимать ключевые сложности, риски и ожидания, связанные с тестированием мобильных приложений.
MAT-03	Применять типы и уровни тестирования, специфические для мобильных приложений.
MAT-04	Применять распространенные типы тестирования, упомянутые в Программе обучения Сертифицированного тестировщика Базового уровня ISTQB®, в специфическом контексте мобильных устройств.

MAT-05	Выполнять активности, необходимые конкретно для тестирования мобильных приложений, как часть основных активностей процесса тестирования, описанного в ISTQB®.
MAT-06	Определять и использовать подходящую среду и соответствующие инструменты для тестирования мобильных приложений.
MAT-07	Понимать методы и инструменты, используемые конкретно для поддержки автоматизации тестирования мобильных приложений.

0.4 Проверяемые цели обучения

Эти цели обучения согласуются с бизнес-результатами и используются при сдаче экзамена для сертификации тестировщиков мобильных приложений Базового уровня. Цели обучения относятся к когнитивному уровню знаний (K).

K-уровень, или когнитивный уровень, используется для классификации целей обучения в соответствии с пересмотренной классификацией Блума [Anderson 2001]. ISTQB® использует эту классификацию для составления экзаменов по своим программам обучения.

Эта программа обучения использует три различных уровня K (от K1 до K3). Для получения дополнительной информации см. Главу 7.

0.5 Уровни практических компетенций

Базовый уровень тестирования мобильных приложений представляет концепцию практических целей, которые сосредоточены на практических навыках и компетенциях.

Компетенции можно развить путем выполнения практических упражнений, указанных в приведенном ниже списке, который не является исчерпывающим:

- Упражнения по целям обучения уровня K3, выполняемые на бумаге или в текстовом процессоре, как это делается в различных существующих учебных программах ISTQB®.
- Настройка и использование тестового окружения.
- Тестирование приложений на виртуальных и физических устройствах.
- Использование инструментов на ПК и/или мобильных устройствах для тестирования или выполнения связанных с тестированием задач, таких как установка ПО, выполнение запросов, логирование, мониторинг, снятие скриншотов и т. д.

Далее указаны уровни практических целей:

- N0: Живая демонстрация упражнения или записанное видео. Поскольку не выполняется обучающимся, не является строго упражнением.
- N1: Упражнение, выполняемое под руководством учителя. Обучающийся повторяет последовательность шагов, выполняемую учителем.
- N2: Упражнение с подсказками. Обучающийся получает упражнение с соответствующими подсказками, чтобы его можно было выполнить в течение указанного периода времени.
- N3: Упражнения без подсказок, выполняемые без руководства учителя.

Рекомендации:

- Цели обучения уровня K1 обычно используют уровень N0 либо уровни N1 и N2, когда того требует ситуация.
- Цели обучения уровня K2 обычно используют уровни N1 и N2 либо уровни N0 и N3, когда того требует ситуация.
- Цели обучения уровня K3 обычно используют уровни N2 и N3, хотя для таких целей обучения не всегда обязательны практические упражнения. Если подготовка упражнений будет слишком трудоемкой и займет много времени, лучше использовать уровень N0.

0.6 Экзамен

Экзамены для сертификации тестировщиков мобильных приложений Базового уровня основаны на данной Программе обучения. Ответы на экзаменационные вопросы могут потребовать использования материала, основанного более чем на одной главе этой Программы обучения. Все разделы Программы обучения поддаются экзаменационной проверке, за исключением Введения и Приложений. Стандарты, книги и другие учебные программы ISTQB® включены в качестве ссылок, но их содержание не рассматривается за рамками того, что обобщено в этой учебной программе.

Формат экзамена по тестированию мобильных приложений Базового уровня – тест с несколькими вариантами ответов. В нем 40 вопросов. Чтобы сдать экзамен, необходимо дать правильные ответы не менее, чем на 65% от всех вопросов (т. е. на 26 вопросов). Практические цели и упражнения экзаменационной проверке не подвергаются.

Экзамены можно сдавать как часть аккредитованного курса обучения или независимо (например, в экзаменационном центре или на открытом экзамене). Прохождение аккредитованного курса обучения не является необходимым условием для сдачи экзамена.

Люди, желающие сдать экзамен без прохождения обучения у аккредитованного организатора обучения, должны прочитать рекомендации по компетенциям, представленные в рекомендациях по аккредитации [CTFL-MAT-2019-Accreditation-and-Competence-Guidelines.pdf], и выполнить указанные в нем практические упражнения самостоятельно. Это поможет им получить компетенции, которые могли быть сформированы при помощи аккредитованного организатора обучения. Обратите внимание, что это не имеет никакого отношения к экзамену для сертификации тестировщиков мобильных приложений Базового уровня, т. к. экзамен основан только на данной Программе обучения и ее целях обучения.

0.7 Рекомендуемое время обучения

В данной Программе обучения определено минимальное время обучения для каждой цели обучения. Общее время для каждой главы указано в заголовке главы.

Организаторам обучения следует иметь в виду, что в других учебных программах ISTQB® применяется подход “стандартного времени”, в соответствии с которым выделяется фиксированное время согласно уровню К. Программа обучения по тестированию мобильных приложений не следует строго этой схеме. Как результат, организаторам обучения даются более гибкие и реалистичные показатели минимального времени обучения для каждой цели обучения.

0.8 Требования для допуска к экзамену

Для допуска к экзамену необходимо пройти сертификацию Базового уровня ISTQB®.

0.9 Источники информации

Определения терминов, используемых в данной программе обучения, содержатся в Глоссарии терминов, используемых в тестировании программного обеспечения [ISTQB_GLOSSARY].

Глава 6 содержит список рекомендуемой литературы и статей по тестированию мобильных приложений.

1. Мир мобильных устройств – Ключевые элементы бизнеса и технологий - 175 мин

Ключевые слова

анализ рисков, снижение рисков, тестирование, основанное на рисках, стратегия тестирования

Цели обучения для главы “Ключевые элементы бизнеса и технологий”

1.1 Данные мобильной аналитики

МАТ-1.1.1 (К2) Объяснить, как доступные данные мобильной аналитики могут быть использованы в качестве основы при разработке стратегии тестирования и тест-плана.

НО-1.1.1 (Н3) Выбрать типы устройств, на которых необходимо провести тестирование, и расставить приоритеты, основываясь на данных, собранных из одного или нескольких источников (географическое местоположение, платформа, версия операционной системы, популярность типов устройств).

Примечание: НО-1.1.1 и НО-1.7.1 (см. ниже) могут быть объединены.

1.2 Бизнес-модели для мобильных приложений

МАТ-1.2.1 (К2) Определить различия между бизнес-моделями для мобильных приложений.

1.3 Типы мобильных устройств

МАТ-1.3.1 (К1) Вспомнить различные типы мобильных устройств.

1.4 Типы мобильных приложений

МАТ-1.4.1 (К2) Определить различия между типами мобильных приложений.

1.5 Архитектура мобильных приложений

МАТ-1.5.1 (К2) Определить различия между основными типами архитектуры мобильных приложений.

1.6 Стратегии тестирования мобильных приложений

МАТ-1.6.1 (К3) Использовать специфику и характеристики мобильного рынка при подготовке стратегии тестирования.

1.7 Сложности тестирования мобильных приложений

МАТ-1.7.1 (К2) Привести примеры сложностей, связанных с тестированием мобильных приложений.

НО-1.7.1 (Н1) Собрать маркетинговые данные, такие как доля устройств или операционных систем в выбранном регионе. Собрать данные по распределению размеров и разрешений экрана. Составить список из пяти устройств и рассчитать для него ожидаемое покрытие рынка.

Примечание: НО-1.1.1 (см. выше) и НО-1.7.1 могут быть объединены.

1.8 Риски тестирования мобильных приложений

МАТ-1.8.1 (К2) Описать, каким образом могут быть снижены риски тестирования мобильных приложений.

1.1 Данные мобильной аналитики

В мире мобильных устройств существует множество заинтересованных лиц, включая производителей, поставщиков платформ, поставщиков операционных систем (ОС), поставщиков рыночных данных, поставщиков инструментов и, конечно, разработчиков и тестировщиков приложений.

Чтобы внести вклад в ходе обсуждения планирования тестирования и анализа тестирования, тестировщик мобильных приложений должен хорошо знать и учитывать следующие факторы:

- Влияние на бизнес распределения платформ на рынке
- Количество скачиваний приложения для платформы
- Количество и распределение версий ОС
- Распределение различных типов устройств на рынке, включая влияние географического местоположения
- Различные размеры и разрешения экрана
- Различные способы ввода данных
- Типы камер

Источников этой информации множество, как бесплатных, так и на коммерческой основе. Например, StatCounter GlobalStats [URL1], сами производители ОС и другие сторонние источники.

Данные мобильной аналитики используются при выборе набора устройств для выполнения тестирования, подходящего для целевого рынка. Тесты прогоняются для выбранного набора устройств, приложение тестируется на устройствах в соответствии с их важностью. Данные, относящиеся к устройствам и их специальным функциям, если таковые имеются, могут также использоваться при разработке тестов, специфичных для конкретного типа устройств. Например, для устройства с датчиком измерения пульса могут потребоваться специальные тестовые сценарии.

1.2 Бизнес-модели для мобильных приложений

Существует несколько моделей монетизации мобильных приложений. К ним относятся: Freemium-приложения, рекламные приложения, приложения на основе транзакций, а также платные и корпоративные приложения. Кроме того, покупки в приложении могут относиться к некоторым моделям.

У каждого из этих подходов есть определенные преимущества и недостатки, поэтому при тестировании мобильных приложений тестировщик должен иметь в виду конкретную бизнес-модель.

В модели Freemium приложения, как правило, бесплатны, однако дополнительные функции пользователи должны оплачивать. Приложение должно обладать достаточным количеством функций для привлечения пользователей и в то же время предоставлять расширенный набор функций, за которые большое количество пользователей будут готовы заплатить.

Рекламные приложения отображают рекламу на экране по мере взаимодействия пользователя с приложением. Эта стратегия получения дохода более эффективна, если приложение используется в течение относительно долгого периода времени. Проектировщики пользовательского интерфейса должны быть осторожными с отображением рекламы. Она должна быть достаточно заметной, не скрывая важных частей приложения, и то же время не должна отвлекать пользователей, делая приложение неприятным в использовании.

Приложения на основе транзакций взимают с пользователей плату за транзакции, фиксированную плату, процент от стоимости транзакции или иную сумму. Эта бизнес-модель подходит только для ограниченного числа приложений и обычно применяется для деловых и финансовых приложений, таких как мобильные кошельки.

Платные приложения требуют от пользователей оплаты за скачивание и установку приложения. Выбор такой бизнес-модели должен быть продуманным, поскольку для большинства типов приложений существует большое количество бесплатных или Freemium вариантов. Вероятность того, что пользователи купят такое приложение, выше, если оно обладает выдающимися функциями или практичностью, либо конкурирующие приложения на рынке отсутствуют.

Бесплатные корпоративные приложения не взимают плату со своих пользователей. Они разрабатываются для использования внутри организации, их интерфейс предназначен для предоставления услуг. Таких приложений множество, например, в банках или компаниях электронной коммерции. Эта модель обычно не ориентирована на монетизацию самого приложения, однако позволяет получать доход, предоставляя пользователям услуги от организации.

1.3 Типы мобильных устройств

Существует множество типов мобильных устройств, поддерживающих различные приложения.

К типичным устройствам относятся:

- Обычные (базовые) телефоны
- Коммуникаторы
- Смартфоны
- Планшеты
- Устройства-компаньоны, включая носимые устройства (“умная одежда”) и некоторые устройства Интернета вещей

При тестировании следует иметь в виду, что каждый тип устройств имеет свои особенности для конкретных нужд.

Обычные телефоны используются только для звонков и SMS и имеют очень мало встроенных приложений и игр. Установка приложений или использование Интернета невозможны.

Коммуникаторы обладают ограниченной поддержкой приложений и их установки. Они предоставляют доступ в Интернет через встроенный браузер и могут иметь дополнительные аппаратные средства, например, камеру.

Смартфоны имеют несколько датчиков. Операционная система поддерживает такие функции, как установка приложений, поддержка мультимедиа и просмотр страниц в Интернете.

Планшеты схожи со смартфонами, но больше по размеру. Обычно они используются, когда необходим или желателен большой экран, а также могут предоставлять более длительное время автономной работы.

Устройства-компаньоны и некоторые устройства Интернета вещей - это компьютерные устройства, которые обычно используются вместе со смартфоном или планшетом для расширения доступного функционала или для более удобного доступа к данным на телефоне или планшете.

“Умная” одежда - это устройства, которые могут носить потребители. Они могут выступать в качестве дополнения к существующим устройствам или функционировать независимо. Примеры популярных носимых устройств - часы и фитнес-браслеты.

1.4 Типы мобильных приложений

Существует три основных типа мобильных приложений:

- Приложения для конкретной платформы
- Браузерные (веб-) приложения;

- Гибридные приложения

У каждого типа приложений есть свои преимущества и недостатки, которые необходимо иметь в виду перед началом разработки приложения.

Приложения для конкретной платформы разрабатываются с помощью SDK, инструментов разработки, датчиков и функций конкретных платформ. Они загружаются, устанавливаются и обновляются из магазинов поставщиков. Таким приложениям может потребоваться тестирование на всех поддерживаемых устройствах.

Приложения для конкретной платформы характеризуются более высокой производительностью, они могут в полной мере использовать функции платформы и соответствовать ожиданиям в отношении платформы, для которой они разработаны. Стоимость разработки, как правило, высокая. Могут возникнуть дополнительные проблемы, такие как использование нескольких платформ, а также установка и тестирование на большом количестве устройств.

Доступ к браузерным приложениям осуществляется через мобильный браузер. Поскольку в них используются типичные технологии веб-разработки и браузеры, мультиплатформенность легко поддерживается и стоимость разработки обычно ниже.

Существует четыре основных метода разработки мобильных веб-приложений:

- Специальные мобильные версии сайтов и приложений (также известные как m(dot)-сайты). Обычно это означает, что при обращении мобильного браузера к приложению, предоставляется мобильная версия приложения. Например, facebook.com перенаправляет на m.facebook.com при доступе с мобильного устройства.
- Отзывчивые веб-приложения адаптируют дизайн к форме и размеру экрана, т. е. к области просмотра.
- Адаптивные веб-приложения выбирают дизайн в соответствии с некоторыми предопределенными размерами. Для этих размеров существуют различные варианты дизайна, и согласно им регулируются функции, доступные пользователю.
- Прогрессивные веб-приложения позволяют создавать ярлыки определенных веб-страниц на рабочем столе мобильного устройства. Они выглядят как приложения для конкретной платформы и иногда могут работать в офлайн режиме.

При создании мобильных веб-приложений используются различные веб-технологии, облегчающие их разработку и управление по сравнению с приложениями для конкретной платформы и гибридными приложениями. В то же время, они обладают не таким богатым функционалом, как приложения для конкретной платформы или гибридные приложения, и имеют ограниченный доступ к API платформы. Доступ к мобильным датчикам также ограничен. Установочное тестирование на устройствах проводить не нужно, однако требуется тестирование совместимости с браузерами.

Гибридные приложения представляют собой сочетание приложений для конкретной платформы и веб-приложений. Они используют встроенную оболочку приложения, которая содержит веб-представление для запуска веб-приложения внутри приложения для конкретной платформы. Эти приложения загружаются из магазинов поставщиков и могут требовать доступ ко всем функциям устройства. Их относительно легко разрабатывать, обновлять и поддерживать без обновления приложения, установленного на устройстве. Навыки, необходимые для разработки этих приложений, почти такие же, как и для веб-разработки. Возможные слабые места таких приложений - проблемы с производительностью из-за использования оболочки и возможные отклонения от ожидаемого внешнего вида и поведения приложения из-за особенностей платформы.

Приложения для конкретной платформы и гибридные приложения физически устанавливаются на устройство и поэтому всегда доступны пользователю, даже если устройство не подключено к Интернету. Для сравнения, браузерные приложения требуют доступа в Интернет.

Некоторые приложения предварительно установлены на мобильное устройство, другие могут быть установлены через различные каналы распространения, такие как App Store от Apple, Google Play

Store, магазины корпоративных приложений (доступны только внутри сети предприятия) и сторонние магазины приложений.

При тестировании каждого из этих типов приложений могут потребоваться разные подходы. К параметрам, влияющим на выбор подхода, относятся:

- Различные типы поддерживаемых устройств
- Используемые функции датчиков и устройств
- Доступность при различных состояниях сети
- Установка, совместимость, эффективность и практичность

1.5 Архитектура мобильных приложений

Существует множество решений для разработки мобильных приложений.

К факторам, влияющим на выбор конкретной архитектуры или проектного решения, относятся:

- Целевая аудитория
- Тип приложения
- Поддержка различных мобильных и других платформ
- Требования к подключаемости
- Требования к памяти
- Связь с другими устройствами, включая устройства Интернета вещей

Архитектурные решения включают:

- Архитектура клиента - тонкий или толстый клиент
- Архитектура сервера - одноуровневая или многоуровневая
- Тип подключения - Wi-Fi, мобильные данные, ближняя бесконтактная связь (NFC), Bluetooth
- Методы синхронизации данных - промежуточное хранение, помещение и получение данных (push and pull), синхронная и асинхронная передача данных

Приложения с тонким клиентом не содержат код приложения, настроенный для устройства, и используют функции мобильной операционной системы по минимуму. Такие приложения обычно используют веб-браузер в качестве внешнего интерфейса и JavaScript в качестве языка реализации клиентской логики.

Приложения с толстым клиентом могут иметь несколько уровней кода приложения и используют функции мобильной операционной системы. Обычно это приложения для конкретной платформы или гибридные приложения.

Серверная архитектура включает в себя следующие возможности:

- Одноуровневая архитектура является монолитной, все серверы находятся на одной машине. Она слабо масштабируема, и ее сложно защитить.
- В многоуровневой архитектуре компоненты распределяются на стороне сервера по различным модулям. Двухуровневая архитектура включает в себя отдельные веб-сервер и сервер баз данных, тогда как трехуровневая архитектура помимо этого также включает сервер приложения. Многоуровневая архитектура позволяет разделить выполняемые функции, обеспечить специализацию базы данных и предоставить лучшую гибкость, масштабируемость и безопасность. Тем не менее, она может быть значительно дороже в разработке, управлении и размещении по сравнению с одноуровневой архитектурой.

Существуют различные типы подключения мобильных устройств к серверу, такие как Wi-Fi или мобильные данные, например, 2G, 3G, 4G и 5G. Мобильные приложения обычно работают в одном из следующих трех режимов:

- Никогда не подключаемые к сети приложения работают в офлайн режиме и не требуют

подключения. Пример такого приложения - простой калькулятор.

- Постоянно подключенные к сети приложения требуют постоянного подключения к сети во время работы. Все мобильные веб-приложения попадают в эту категорию, хотя некоторые из них могут работать в ограниченном режиме при частичном подключении.
- Частично подключенные к сети приложения требуют подключения для таких задач, как передача данных, однако могут работать в течение длительного времени без подключения к сети.

Синхронизация данных между клиентом и сервером может проводиться в следующих режимах:

- В непрерывном режиме данные передаются сразу после их отправки.
- В режиме с промежуточным хранением данные могут храниться локально перед их передачей, особенно когда подключаемость отсутствует.

Существует два подхода к передаче данных:

- Синхронная передача данных используется, когда вызывающая функция ожидает завершения вызываемой функции перед возвратом.
- Асинхронная передача данных используется, когда вызываемая серверная функция немедленно возвращает данные, обрабатывает данные в фоновом режиме и вызывает обратно вызывающую клиентскую функцию после завершения задачи. Это предоставляет пользователям больше возможностей контроля. Однако реализация механизма рукопожатия повышает сложность в отношении доступности клиента или сети, когда сервер инициирует обратный вызов.

1.6 Стратегии тестирования мобильных приложений

Разработка стратегии тестирования мобильных устройств требует от тестировщика принятия во внимание всех параметров, перечисленных в этой главе. Кроме того, следует также учитывать риски, обсуждаемые в этом разделе, и сложности, описанные в разделе 1.7.

Например, типичные риски могут быть следующие:

- Не обладая данными о распределении устройств в определенном географическом местоположении, невозможно выбрать устройства, на которых рационально проводить тестирование приложения.
- Не зная тип бизнес-модели, невозможно проверить, подходит ли поведение приложения этой бизнес-модели.

При подготовке стратегии тестирования мобильного приложения дополнительно нужно учитывать следующие риски и сложности:

- Специфические дефекты на конкретных устройствах из всего разнообразия мобильных устройств.
- Наличие устройств внутри компании или использование внешних тестовых лабораторий.
- Внедрение новых технологий, устройств и/или платформ в течение жизненного цикла приложения.
- Установка и обновление самого приложения через различные каналы, включая сохранение данных и настроек приложения.
- Слабые места платформы, которые могут повлиять на приложение.
- Покрытие сети и его влияние на приложение в глобальном контексте.
- Возможность тестирования с использованием сетей от различных поставщиков услуг.
- Использование мобильных эмуляторов, симуляторов и/или реальных устройств для конкретных уровней тестирования и типов тестирования.

Эти сложности более подробно описаны в разделе 1.7.

Стратегия тестирования учитывает риски и сложности. Например:

- Стратегия тестирования может указывать на использование мобильных эмуляторов/симуляторов на ранних этапах разработки и использование реальных устройств на более поздних этапах. Некоторые типы тестирования могут проводиться на мобильных эмуляторах/симуляторах. Подробнее об этом написано в разделе 4.3.
- Стратегия тестирования может учитывать сложности, создаваемые большим количеством различных устройств, используя один из следующих подходов:
 - Одноплатформенный подход: Сократить объем работ до одного типа устройства, одной версии ОС, одного оператора связи и одного типа сети.
 - Мультиплатформенный подход: Сократить объем работ до репрезентативной выборки устройств и ОС, используемых большинством пользователей на целевом рынке, созданной на основе мобильного трафика или других аналитических данных.
 - Подход с максимальным покрытием: покрытие всех версий ОС, устройств, производителей, операторов связи и типов сетей. По сути, это исчерпывающее тестирование, которое, как правило, экономически нецелесообразно, особенно при рассмотрении множества устройств и версий ОС, представленных на рынке.
- Стратегия тестирования может учитывать сложности, связанные с недоступностью устройств, сетей или реальных условий, используя внешние ресурсы, такие как:
 - Сервисы удаленного доступа к устройствам - способ получения доступа к устройствам через Интернет.
 - Сервисы многопользовательского тестирования (жарг. «краудтестинг») - способ получения доступа к устройствам широкого круга лиц на добровольной основе.
 - Тестирование знакомыми и друзьями (жарг. «нетворкинг») - привлечение друзей и коллег, т. е. использование так называемой собственной социальной сети.
 - «Охота на ошибки» (жарг. «багхантинг») - тестирование в игровой форме с участием добровольцев из компании или широкой аудитории.

В дополнение к уровням тестирования, описанным в [ISTQB_CTFL_2018], стратегия тестирования также учитывает распространенные типы тестирования мобильных приложений (см. раздел 3.1) и любые необходимые дополнительные уровни тестирования (см. раздел 3.2).

1.7 Сложности тестирования мобильных приложений

В мире мобильных устройств существует множество дополнительных сложностей, нетипичных или некритичных для настольного и серверного программного обеспечения. Тестировщики должны помнить об этих проблемах и о том, как они могут повлиять на успех приложения.

К типичным сложностям относятся:

- Фрагментация устройств: разные платформы, типы и версии ОС, размеры экрана и их качество.
- Аппаратные средства устройств: разные типы датчиков и трудности моделирования тестовых условий с ограниченными ресурсами ЦП и ОЗУ.
- Разнообразие инструментов разработки для разных платформ.
- Отличия между дизайном пользовательского интерфейса и ожиданий от использования платформы.
- Множество типов сетей и провайдеров.
- Устройства с недостатком ресурсов.
- Различные каналы распространения приложений.
- Разнообразие пользователей и групп пользователей.
- Различные типы приложений с различными типами подключения.
- Доступность обратной связи о серьезных дефектах приложения, что может привести к публикации негативных отзывов других пользователей на торговых интернет-площадках.

- Публикация на рынке, требующая дополнительных циклов одобрения со стороны владельцев рынка, таких как Google Play Store или App Store от Apple.
- Недоступность недавно выпущенных устройств, что требует использования мобильных эмуляторов/симуляторов.

Влияние этих сложностей заключается в следующем:

- Большое количество комбинаций для тестирования.
- Необходимость большого количества устройств для тестирования, что повышает его стоимость.
- Необходимость обратной совместимости для запуска приложения на старых версиях платформы.
- Новые функции, выпускаемые в каждой версии базовой операционной системы.
- Необходимость рассмотрения рекомендаций для различных платформ.
- Нехватка ресурсов ЦП и ограниченный объем оперативной и внешней памяти.
- Различная пропускная способность и фазовое дрожание цифрового сигнала данных различных сетей.
- Изменение доступной скорости загрузки и отдачи в зависимости от тарифного плана.

Следующие два примера иллюстрируют типичные сложности и их потенциальное влияние:

- Разные устройства имеют разные типы датчиков, и тесты должны это учитывать. Для каждого нового датчика, добавляемого к аппаратным средствам, может потребоваться дополнительное тестирование обратной совместимости.
- Некоторые из трудностей, связанных с сетью, могут быть преодолены даже в различных сетевых условиях путем использования подходящей стратегии кэширования или иной упреждающей стратегии. Однако это обходится дорого; большое количество открытых подключений может повлиять на производительность сервера, так как в большинстве приложений авторизация пользователя происходит именно на сервере.

1.8 Риски тестирования мобильных приложений

Сложности, упомянутые в разделе 1.7, могут возникать изолированно друг от друга или в совокупности. Это может привести к дополнительным рискам мобильного приложения.

Тестирующий должен иметь возможность принимать участие в анализе рисков продукта. Основные методы анализа рисков и методы снижения рисков, как обсуждалось в [ISTQB_CTFL_2018], глава 5.5, также могут применяться в мобильном контексте. Кроме того, существуют следующие риски и методы их снижения, специфичные для мобильных устройств:

Риск	Возможный метод снижения
Фрагментация рынка	Выбрать подходящий набор устройств для выполнения тестирования, например, наиболее часто используемые устройства.
Стоимость поддержки нескольких платформ	Провести анализ платформ, чтобы выявить наиболее распространенные, избегая тем самым тестирования тех платформ, которые больше не используются.
Внедрение новых технологий, платформ и устройств	Использовать предпроизводственные версии этих технологий.

Нехватка доступных устройств для выполнения тестирования	Использовать сервисы удаленного доступа к устройствам или сервисы многопользовательского тестирования.
Риски ожидаемых моделей использования мобильных приложений	Применять соответствующие подходы к тестированию, например, тестирование в условиях эксплуатации.

2. Типы тестирования мобильных приложений – 265 мин

Ключевые слова

сосуществование, совместимость, возможность подключения, кроссбраузерная совместимость, возможность взаимодействия, тестируемая система (SUT - system under test), тип тестирования, практичность

Цели обучения для главы “Типы тестирования мобильных приложений”

2.1 Тестирование совместимости с аппаратным обеспечением устройства

MAT-2.1.1	(K2) Описать специфические для устройства функции и аппаратные средства, которые нужно учитывать при тестировании.
HO-2.1.1	(H1) Протестировать функциональность нескольких мобильных устройств во время использования тестируемой системы (SUT), чтобы определить правильность функционирования тестируемой системы.
MAT-2.1.2	(K3) Подготовить тесты на совместимость приложения с размерами экрана, соотношением сторон и разрешением экрана.
HO-2.1.2	(H3) Протестировать приложение на нескольких мобильных устройствах (виртуальных или физических), чтобы показать влияние разрешения и размера экрана на пользовательский интерфейс приложения.
MAT-2.1.3	(K2) Описать, как тесты могут показать потенциальное влияние перегрева устройства на тестируемую систему.
MAT-2.1.4	(K1) Вспомнить типы тестирования для различных датчиков мобильных устройств.
MAT-2.1.5	(K1) Вспомнить тесты для различных способов ввода данных.
HO-2.1.5	(H0) Протестировать приложение, используя различные способы ввода данных, включая тесты, связанные с клавиатурой, с несколькими клавиатурами, с жестами, и (опционально) с камерой.
MAT-2.1.6	(K2) Описать, как тесты могут выявить слабые места пользовательского интерфейса при изменении ориентации экрана.
HO-2.1.6	(H3) Протестировать приложение, чтобы проверить влияние изменения ориентации экрана на функциональность приложения, включая сохранение данных и корректность пользовательского интерфейса.
MAT-2.1.7	(K3) Подготовить тесты на типичные прерывания мобильных устройств.
HO-2.1.7	(H3) Протестировать приложение на несколько прерываний мобильных устройств во время использования приложения.
MAT-2.1.8	(K3) Подготовить тесты на изменение прав доступа к функциям устройства, запрашиваемым приложением.
HO-2.1.8	(H3) Протестировать управление разрешениями приложения путем предоставления и отклонения запрошенных разрешений и наблюдения за поведением приложения при отключенных во время установки разрешений доступа к папкам и датчикам или изменении этих разрешений.
MAT-2.1.9	(K3) Подготовить тесты на влияние приложения на энергопотребление устройства и влияние состояния питания устройств на приложение.
HO-2.1.9	(H3) Протестировать приложение при различных уровнях заряда батареи, чтобы собрать данные о потреблении батареи и установить производительность при низком уровне заряда и разряженном состоянии батареи.

2.2 Тестирование взаимодействия приложения с ПО устройства

MAT-2.2.1	(K3) Подготовить тесты на реакцию тестируемой системы на уведомления.
HO-2.2.1	(H2) Протестировать реакцию приложения на получение уведомлений, когда оно находится на переднем плане или работает в фоновом режиме. Протестировать влияние изменения настроек уведомлений на функциональность приложения.
MAT-2.2.2	(K2) Описать, как тесты могут определить правильность работы ссылок

	быстрого доступа.
НО-2.2.2	(Н3) Протестировать функциональность ярлыков/ссылок быстрого доступа приложения.
МАТ-2.2.3	(К3) Подготовить тесты на влияние пользовательских настроек, предоставленных операционной системой, на приложение.
НО-2.2.3	(Н3) Протестировать работу приложения при изменении заданных по умолчанию настроек, предоставленных операционной системой.
МАТ-2.2.4	(К2) Описать различия между тестированием приложений для конкретной платформы, браузерных и гибридных приложений.
НО-2.2.4	(Н0) (Необязательно) Определить необходимые тесты в зависимости от типа приложения.
МАТ-2.2.5	(К1) Вспомнить тесты, необходимые для приложений, доступных на нескольких платформах или версиях операционной системы.
МАТ-2.2.6	(К1) Вспомнить тесты на сосуществование и возможность взаимодействия с другими приложениями.

2.3 Тестирование различных методов подключения

МАТ-2.3.1	(К2) Кратко охарактеризовать тесты на возможность подключения, включая тесты через сети, например, при использовании Bluetooth или включении режима полета.
НО-2.3.1	(Н0) (Опционально) Провести тестирование приложения, передающего данные на сервер, когда телефон переключается между Wi-Fi и мобильными данными в зависимости от их доступного уровня сигнала.

2.1 Тестирование совместимости с аппаратным обеспечением устройства

2.1.1 Тестирование функций устройства

Существование различных типов устройств с разными возможностями означает, что тестирование совместимости должно проводиться на большом количестве устройств. Для этого требуется приоритезация целевых устройств для тестирования. Как обсуждалось в разделе 1.1, для приоритезации используются рыночные данные, на основе которых происходит выбор портфолио устройств, наиболее подходящих для целевого рынка. Выбор портфолио обычно является компромиссом между покрытием рынка, стоимостью и рисками.

Приложения могут быть установлены на различные типы устройств со следующими функциями:

- Разные способы выключения
- Разные способы навигации
- Использование аппаратной (внешней) и программной (экранной) клавиатуры
- Различные аппаратные средства, такие как:
 - Радио
 - USB
 - Bluetooth
 - Камера
 - Динамик
 - Микрофон
 - Гнездо для наушников

Ни одна из этих функций не должна отрицательно влиять на работу приложения.

Функции устройств могут сильно отличаться друг от друга, даже у разных моделей устройств одного и того же производителя. Это влияет на разграничение между сегментами рынка и может быстро меняться со временем. Например, в настоящее время довольно часто устройства высокого и среднего класса имеют сканер отпечатков пальцев, в то время как на бюджетных устройствах его нет. Это меняется со временем: несколько лет назад сканера отпечатков пальцев не было ни на одном мобильном устройстве. В силу этой изменчивости тестировщик должен иметь ясное представление об устройствах и их функциях, необходимых пользователям. Тестировщик должен

создать портфолио устройств и разработать соответствующие тесты.

Как правило, недостаточно протестировать приложение на правильность работы при наличии необходимых функций. Кроме этого, необходимо проверить, что приложение работает должным образом, если определенная функция отсутствует. Например, приложение, которое поддерживает использование фронтальной и основной камеры, не должно аварийно завершать работу, если оно установлено и выполняется на устройстве с несколькими камерами, одной камерой или без камеры вовсе.

2.1.2 Тестирование разных экранов

Экраны устройств могут отличаться по своему размеру, размеру области просмотра, соотношению сторон и разрешению экрана, которое измеряется в количестве пикселей на дюйм (ppi) и в количестве точек на дюйм (dpi). Из-за фрагментации устройств необходимо расставлять приоритеты. Должны быть созданы тесты для проверки пользовательского интерфейса на различных устройствах с различными размерами экрана, разрешениями и соотношением сторон, наиболее распространенными на целевом рынке.

Необходимо провести тестирование разных экранов, чтобы проверить следующее:

- Приложение масштабирует все элементы пользовательского интерфейса в соответствии с размером и разрешением экрана.
- Элементы пользовательского интерфейса не перекрывают друг друга.
- Не возникают проблемы, связанные с практичностью или датчиками.
- Не возникают проблемы, связанные с масштабированием изображений из-за высокого dpi/ppi.

2.1.3 Тестирование температуры устройства

В отличие от настольных компьютеров, мобильные устройства по-разному реагируют на повышение температуры устройства.

Мобильные устройства могут перегреваться по разным причинам, таким как зарядка батареи, большая нагрузка, приложения, работающие в фоновом режиме, постоянное использование мобильных данных, Wi-Fi или GPS.

Перегрев может повлиять на устройство, поскольку оно пытается уменьшить нагрев и сохранить уровень заряда батареи. В результате это может вызвать снижение частоты процессора, освобождение памяти и отключение частей системы.

Впоследствии перегрев также может повлиять на функциональность приложения и, следовательно, должен учитываться при планировании тестирования. Тесты должны быть рассчитаны на потребление большого количества энергии, которое приводит к нагреву устройств в течение длительного периода времени. Тестируемое программное обеспечение не должно вести себя неожиданным образом.

2.1.4 Тестирование датчиков ввода данных

Мобильные устройства получают множество входных данных с помощью различных датчиков, например, GPS, акселерометра, гироскопа, трехосевого магнитометра или датчиков, реагирующих на давление, температуру, влажность, сердцебиение, свет или бесконтактный ввод данных.

Тестирование датчиков ввода данных проверяет следующее:

- Приложение работает должным образом по отношению к каждому из доступных датчиков. Например, необходимо протестировать приложение, используя различные виды движения, такие как движение по кругу и движение вперед и назад (как при ходьбе).
- Функции, реагирующие на внешнее освещение, ведут себя должным образом при различных условиях освещения.
- Датчики ввода и вывода звуковых данных реагируют правильно в совокупности с виртуальными и физическими кнопками регулировки громкости, микрофоном, проводными и беспроводными динамиками при различных условиях внешнего шума.
- Местоположение определяется точно при следующих условиях:
 - GPS включен или выключен.

- Различный уровень сигнала GPS.
- Ситуации, когда приложение прибегает к использованию других методов определения местоположения, включая Wi-Fi, местоположение вышки сотовой связи или ручной ввод местоположения.

2.1.5 Тестирование различных способов ввода данных

Тестирование различных способов ввода данных проверяет следующее:

- Учитывая, что мобильные телефоны позволяют устанавливать различные программные клавиатуры, приложение может работать, по крайней мере, с теми из них, которые предоставляются крупными производителями устройств, и теми, которые широко используются.
- Приложение гарантирует, что при необходимости клавиатура по умолчанию всплывает с соответствующей раскладкой.
- Когда пользователь касается одним или несколькими пальцами сенсорного экрана, приложение интерпретирует это как определенный жест или команду. К типичным жестам относятся нажатие/касание, двойное нажатие, мультикасание, смахивание, постукивание, двойное постукивание, перетаскивание и сведение/разведение пальцев.
- Каждый экран приложения правильно реагирует на жесты или другие способы ввода, соответствующие этому экрану, и игнорирует все неподдерживаемые жесты или способы ввода.
- Камеры, используемые приложением, могут захватывать изображения и видео, сканировать штрих-коды, QR-коды, документы и измерять расстояние.
- Там, где доступны фронтальная и основная камеры, соответствующая камера включена по умолчанию. Например, если для видеочата требуется, чтобы фронтальная камера была включена по умолчанию, приложение должно быть протестировано и в тех случаях, когда оно использует ввод данных с помощью камеры, и когда нет. Кроме того, тесты должны проверить, что тестируемое программное обеспечение работает правильно, если присутствует только одна камера (фронтальная или основная) вместо двух. Эта проверка особенно важна, когда в тестируемом программном обеспечении используется одна конкретная камера, и именно она отсутствует.

2.1.6 Тестирование изменения ориентации экрана

Для обнаружения изменения ориентации экрана используются датчики движения, которые вызывают переключение между портретным и альбомным режимами (и наоборот) с изменениями в макете, вносимыми в пользовательский интерфейс по мере необходимости.

Тестирование, выполняемое после изменения ориентации экрана, проверяет следующее:

- Практичность и правильное функционирование приложения при переключении в портретный или альбомный режим.
- Приложение остается в том же состоянии.
- Поля ввода данных сохраняют собранные данные.
- Поля вывода данных отображают те же данные, сохраняя текущий сеанс.

Тестирование изменения ориентации экрана не должно фокусироваться только на одном переключении, поскольку проблемы с рендерингом или состоянием приложения не всегда могут появиться после одного изменения. Именно поэтому необходимо выполнять несколько непрерывных переключений между портретным и альбомным режимами.

Следует разрабатывать тесты, в которых ориентация экрана меняется несколько раз при различных состояниях пользовательского интерфейса, с данными и без данных. Приложение должно работать, как ожидалось, сохраняя свое состояние без потери или изменения данных.

2.1.7 Тестирование типичных прерываний

К распространенным типам прерываний относятся звонки, сообщения, подключение зарядного устройства, нехватка памяти и различные уведомления. Прерывания, инициируемые пользователем, возникают в результате таких действий, как переключение между приложениями или перевод устройства в режим ожидания во время работы приложения.

Тестирование прерываний проверяет следующее:

- Приложение правильно обрабатывает все прерывания, упомянутые выше, без негативного влияния на поведение приложения.
- Приложение продолжает работать правильно, сохраняя свое состояние, данные и сеанс независимо от того, какое прерывание происходит.
- Если у устройства есть режим «Не беспокоить», который блокирует уведомления, приложение должно гарантировать свою правильную работу в различных условиях. Тесты также должны выполняться при отключении режима «Не беспокоить» после того, как он был включен в течение длительного периода времени. При этом множество уведомлений будет получено одновременно после отключения режима.
- Тесты должны быть разработаны таким образом, чтобы прерывания происходили во время использования приложения. Это нужно для проверки того, что прерывания не оказывают отрицательного воздействия на приложение. Например, после ответа на телефонный звонок во время использования приложения и завершения разговора, пользователь возвращается в состояние, в котором он находился до прерывания.

2.1.8 Тестирование прав доступа к функциям устройства

Приложению необходим доступ к различным папкам, таким как контакты и изображения, а также к датчикам, таким как камера и микрофон. Отклонение разрешений при установке или их изменение после установки может повлиять на работу приложения.

Тестирование прав доступа проверяет следующее:

- Приложение может работать с ограниченными разрешениями; в этом случае приложение просит пользователя предоставить разрешения, отказы приложения отсутствуют.
- Приложение запрашивает разрешения только для ресурсов, имеющих отношение к функциональности приложения; расширенные разрешения для не связанных с приложением ресурсов не допускаются.
- Приложение функционирует правильно, если разрешение отозвано или отклонено во время установки.
- Все запросы на разрешения от приложения являются правильными и обоснованными.

Для тестирования прав доступа тестировщик должен знать, зачем приложению необходимо каждое разрешение и как отзыв разрешения или отклонение разрешения во время установки может повлиять на функциональность приложения. Тесты должны быть разработаны для проверки отклонения разрешений во время установки и предоставления разрешений после установки.

2.1.9 Тестирование энергопотребления и состояния

Тестирование энергопотребления и состояния проверяет следующее:

- Уровень заряда батареи и дефекты, связанные с разрядкой.
- Сохранность данных при низком уровне заряда и разряженном состоянии батареи.
- Потребление энергии во время работы приложения, как при активном, так и слабом использовании.
- Потребление энергии во время работы приложения в фоновом режиме.

Эти тесты необходимо тщательно планировать, так как они должны выполняться непрерывно в течение длительного периода времени. Например, устройство с приложением, запущенным в фоновом режиме или на переднем плане, не должно использоваться. Такие инструменты, как анализаторы логов, необходимы для получения информации об особенностях разрядки батареи.

2.2 Тестирование взаимодействия приложения с ПО устройства

2.2.1 Тестирование уведомлений

Существуют различные механизмы, используемые операционной системой для отображения уведомлений. Иногда операционная система либо задерживает отображение уведомлений, либо вообще не отображает уведомления для оптимизации энергопотребления. Необходимо

рассматривать следующие тестовые условия:

- Правильная обработка уведомлений, полученных во время работы приложения на переднем плане или в фоновом режиме, особенно при низком уровне заряда батареи.
- Если уведомления позволяют напрямую взаимодействовать с контентом приложения (т. е. без открытия самого приложения), позже должно быть обеспечено взаимодействие пользователя с приложением. Например, если пользователь отвечает на уведомление, позже должна быть возможность доступа к этому ответу внутри приложения.
- Если уведомления предоставляют доступ к приложению, то вместо главного экрана приложения должна открываться соответствующая страница приложения в том случае, когда уведомление содержит глубокую ссылку на эту страницу.

2.2.2 Тестирование ссылок быстрого доступа

Тестируемое программное обеспечение может предоставлять ссылки быстрого доступа, такие как ярлыки приложений в Android и Force-touch/3D Touch в iOS. Эти ссылки выполняют набор функций приложения с рабочего стола устройства, фактически не запуская приложение целиком.

Необходимо рассматривать следующие тестовые условия:

- Если некоторые функции доступны только в конкретной версии операционной системы, тестируемая система должна функционировать правильно независимо от того, установлена ли она на версии операционной системы, в которой эти функции есть, либо на версии операционной системы, в которой их нет.
- Действия, выполняемые с помощью ссылок быстрого доступа, правильно отображаются в приложении при его открытии.

2.2.3 Тестирование пользовательских настроек, предоставляемых операционной системой

Необходимо тестировать все настройки, предоставляемые пользователю операционной системой. Если приложение не реагирует на какие-то настройки, это произведет негативное впечатление на пользователей. Например, если устройство находится в беззвучном режиме, приложение не должно воспроизводить звуки.

Необходимо рассматривать следующие тестовые условия:

- Пользователи могут менять типичные настройки, такие как звук, яркость экрана, сеть, режим энергосбережения, дата и время, часовой пояс, язык, тип доступа и уведомления.
- Приложение работает в соответствии с выбранными настройками.

2.2.4 Тестирование разных типов приложений

В зависимости от типа мобильного приложения могут выполняться специфические тесты (см. раздел 1.4). Необходимо рассматривать следующие тестовые условия:

- Для приложений для конкретной платформы:
 - Совместимость с устройствами
 - Использование функций устройства
- Для гибридных приложений:
 - Взаимодействие приложения с собственными функциями устройства
 - Потенциальные проблемы с производительностью, связанные с уровнем абстракции
 - Практичность (внешний вид и поведение приложения) на рассматриваемой платформе в сравнении с приложениями для конкретной платформы
- Для браузерных приложений:
 - Кроссбраузерная совместимость приложения с распространенными мобильными браузерами
 - Отсутствие влияния различных движков JavaScript на функциональность приложения
 - Использование функций ОС (например, выбор даты и открытие подходящей клавиатуры)
 - Практичность (внешний вид и поведение приложения) на рассматриваемой платформе в сравнении с приложениями для конкретной платформы

2.2.5 Тестирование возможности взаимодействия с несколькими платформами и версиями операционной системы

Компании, выпускающие программное обеспечение, часто поддерживают приложения на нескольких операционных системах. Каждая мобильная операционная система имеет свои ограничения, которые необходимо учитывать при тестировании приложений. Тестировщики должны знать особенности каждой тестируемой платформы, чтобы проверить, что приложение работает, как ожидалось, и в то же время согласуется с внешним видом и поведением платформы.

Необходимо рассматривать следующие тестовые условия:

- Обработка прерываний, уведомлений и оптимизация (например, для энергосбережения).
- Правильная работа приложения в случае, когда мультиплатформенные приложения совместно используют определенные части кода или были созданы в кроссплатформенной среде разработки. Обратите внимание, что если приложения не используют код совместно, необходимо протестировать все функции в каждом приложении по отдельности.
- Тестирование обратной совместимости, если платформа использует разные версии операционной системы.
- Тестирование новых или измененных возможностей платформы. Например, для внедрения режима Doze в Android требовалось тестирование на различных версиях операционной системы, как на тех версиях, которые поддерживают этот режим, так и на тех версиях, которые этот режим не поддерживают.

2.2.6 Тестирование возможности взаимодействия и сосуществования с другими приложениями на устройстве

Довольно часто приложения взаимодействуют друг с другом при установке на устройство. Типичным примером являются приложения, связанные с контактами и электронной почтой.

Необходимо рассматривать следующие тестовые условия:

- Корректная передача данных между тестируемой системой и используемым приложением.
- Пользовательские данные, хранящиеся в используемом приложении, не были повреждены.
- Конфликт приложений. Например, одно приложение может отключить GPS для экономии заряда батареи, в то время как другое приложение автоматически включает GPS.

Наличие миллионов приложений на рынке делает полное тестирование сосуществования невозможным. Тем не менее, необходимо рассматривать такие потенциальные проблемы и проводить тестирование в соответствии с рисками.

2.3 Тестирование различных методов подключения

Мобильные устройства могут использовать различные методы подключения к сети (см. раздел 1.5). К ним относятся сотовые сети, такие как 2G, 3G, 4G и 5G, а также Wi-Fi и другие беспроводные технологии, такие как NFC и Bluetooth.

Необходимо рассматривать следующие варианты при проведении тестирования возможности подключения:

- Эмуляторы/симуляторы устройств могут имитировать различные типы подключения к сети. Некоторые сервисы удаленного доступа к устройствам также обладают этой функцией. Тем не менее, использование эмуляторов/симуляторов носит ограниченный характер.
- Создание собственной мобильной сети, поддерживающей различные типы соединения, и проведение манипуляций с ее пропускной способностью. Это очень дорогостоящий вариант.
- Тестирование в условиях эксплуатации потенциально является менее затратным вариантом, однако имеет ограничения, связанные с воспроизведением тестов.

В реальных условиях методы подключения различаются. Пользователь может быть постоянно подключен к сети, используя один конкретный метод, или же переключаться между ними, например, с Wi-Fi на сотовую связь (при выходе из дома во время использования приложения). Пользователь может переключаться между различными Wi-Fi/сотовыми сетями и их версиями, а также между

GSM-сотами. Находясь в пути, пользователь может даже попасть в мертвую зону сети. Кроме того, пользователь может намеренно отключиться от сети, например, путем включения режима полета.

При тестировании возможности подключения необходимо рассматривать следующие тестовые условия:

- Правильность функционирования приложения при различных типах подключения.
- При переключении между типами подключения приложение не начинает вести себя неожиданным образом и не происходит потеря данных.
- Приложение дает четко понять пользователю, что функциональность ограничена из-за проблем с подключением к сети, из-за отсутствия подключения или низкой пропускной способности сети. В сообщении должны быть указаны ограничения и причины их возникновения.

3. Распространенные типы тестирования и процесс тестирования мобильных приложений – 200 минут

Ключевые слова

аварийное завершение, доступность, исследовательское тестирование, тестирование в условиях эксплуатации, эвристика, устанавливаемость, эффективность производительности, тестирование производительности, пост-релизное тестирование, тестирование защищенности, управление тестированием на основе сеансов, стрессовое тестирование, уровень тестирования, процесс тестирования, пирамида тестирования, тест-тур, лаборатория эргономики (практичности), тестирование практичности, внедрение кода, тестирование после выпуска программного обеспечения.

Цели обучения для главы “Распространенные типы тестирования и процесс тестирования мобильных приложений”

3.1 Распространенные типы тестирования, применимые к мобильным приложениям

- МАТ-3.1.1 (К3) Составить установочные тесты для мобильных приложений.
- МАТ-3.1.2 (К3) Составить стресс-тесты для мобильных приложений.
- МАТ-3.1.3 (К2) Привести примеры проблем безопасности, связанных с мобильными приложениями.
- МАТ-3.1.4 (К1) Вспомнить особенности использования времени и ресурсов при работе мобильных приложений .
- МАТ-3.1.5 (К3) Составить тесты практичности для мобильных приложений.
- НО-3.1.5 (Н2) Выбрать тест-тур, мнемонику или эвристику для тестирования практичности приложения, используя управление тестированием на основе сеансов.
- Примечание: НО-3.1.5 и НО-3.3.1, НО-3.3.2 и НО-3.3.3 могут быть объединены.
- МАТ-3.1.6 (К1) Выявить типы тестов, необходимые для тестирования баз данных мобильных приложений.
- МАТ-3.1.7 (К2) Кратко охарактеризовать тесты, необходимые для тестирования интернационализации (глобализации) и локализации мобильных приложений
- МАТ-3.1.8 (К2) Сформулировать, в чем заключается необходимость тестирования доступности мобильных приложений.

3.2 Дополнительные уровни тестирования, применимые к мобильным приложениям

- МАТ-3.2.1 (К2) Описать дополнительные уровни тестирования, например, тестирование в условиях эксплуатации, и связанные с ними дополнительные активности, необходимые для эффективного тестирования мобильных приложений.
- МАТ-3.2.2 (К2) Описать тесты, необходимые для получения разрешения от магазина приложений на публикацию приложения.

3.3 Методики тестирования на основе опыта

- МАТ-3.3.1 (К1) Вспомнить управление тестированием на основе сеансов, тест-персоны и мнемоники в контексте исследовательского мобильного тестирования.
- НО-3.3.1 (Н2) Выбрать мнемонику (или ее часть), характерную для тестирования мобильных приложений, чтобы протестировать приложение, используя управление тестированием на основе сеансов.
- Примечание: НО-3.1.5 и НО-3.3.1, НО-3.3.2 и НО-3.3.3 могут быть объединены.
- МАТ-3.3.2 (К2) Описать использование тест-туров и эвристик как методов исследовательского тестирования мобильных приложений.

- НО-3.3.2 (H2) Протестировать мобильное приложение, используя подходящую эвристику.
Примечание: НО-3.1.5 и НО-3.3.1, НО-3.3.2 и НО-3.3.3 могут быть объединены.
- МАТ-3.3.3 (K3) Воспользоваться тест-туром (например, туром “Функция”) для тестирования мобильного приложения.
- НО-3.3.3 (H2) Протестировать мобильное приложение, с использованием подходящего тест-тура.
Примечание: НО-3.1.5 и НО-3.3.1, НО-3.3.2 и НО-3.3.3 могут быть объединены.

3.4 Процесс тестирования и подходы к тестированию мобильных приложений

- МАТ-3.4.1 (K2) Соотнести процесс тестирования, описанный в [ISTQB_CTFL_2018], с особенностями процесса тестирования мобильных приложений.
- МАТ-3.4.2 (K2) Описать подходы к тестированию на каждом уровне, характерные для тестирования мобильных приложений.

3.1 Распространенные типы тестирования, применимые к мобильным приложениям

3.1.1 Установочное тестирование

Тестировщикам необходимо проверять установку, обновление и деинсталляцию приложения, используя следующие способы:

- Магазины приложений

Процесс установки может отличаться для разных пользователей приложения. Пользователи могут установить приложение из таких магазинов, как Google Play Store или App Store от Apple. Для корпоративных приложений необходимо выполнять установочные тесты, используя ссылку или сервис дистрибуции, например, HockeyApp или App Center.

- Загрузка неопубликованного приложения (копирование и установка)

Некоторые операционные системы позволяют устанавливать приложения копированием их на мобильное устройство и последующей установкой из файла.

- Desktopные приложения

С помощью десктопных приложений, таких как Apple iTunes (для iOS) или Android App Installer, можно устанавливать приложения на смартфон. Тестировщик должен сначала загрузить приложение на ПК, а затем установить его на смартфон при помощи кабеля. Большинство таких приложений также позволяют деинсталлировать приложения.

Установка может производиться следующими способами:

- По беспроводной сети (используя Wi-Fi или мобильные данные)
- С помощью кабеля передачи данных

К тестовым условиям, которые следует учитывать, относятся:

- Установка, удаление и обновление приложения на встроенных и внешних запоминающих устройствах (если таковые поддерживаются).
- Переустановка приложения в случае, когда при деинсталляции была выбрана опция “Сохранить данные приложения”.
- Переустановка приложения в случае, когда при деинсталляции не была выбрана опция “Сохранить данные приложения”.
- Отмена или прерывание установки/деинсталляции приложения, например, при выключении мобильного устройства в процессе установки/деинсталляции или при потере соединения с Интернет.

- Возобновление установки, деинсталляции и обновления после отмены или прерывания.
- Тестирование прав доступа в процессе установки. К примеру, некоторые приложения при установке запрашивают разрешение на использование контактов. Важно выполнить тест на проверку работы приложения в том случае, если пользователь запретил доступ (например, было ли ему отправлено соответствующее сообщение).
- Обновление приложения с последующей проверкой, что данные не были потеряны.

Для установки некоторых приложений требуются устройства, на которых была проведена процедура взлома для iOS или Android для предоставления пользователю прав на администрирование. Большинство поставщиков платформ не поддерживает взлом, поскольку он может иметь юридические последствия. Приложения, не требующие взлома, не обязательно проверять на таких устройствах.

3.1.2 Стрессовое тестирование

Стрессовое тестирование направлено на определение эффективности производительности приложения в условиях повышенной нагрузки. Стресс-тест в этом контексте ориентирован только на мобильные устройства. Стресс-тесты серверной части приложения описаны в программе обучения ISTQB® по тестированию производительности ([ISTQB_CTFL_PT_2018]). При необходимости можно обращаться к этому документу за дополнительной информацией.

К тестовым условиям, которые следует учитывать при стрессовом тестировании, относятся:

- Высокая загрузка центрального процессора
- Нехватка памяти
- Нехватка места на диске
- Загрузка батареи
- Отказы
- Низкая пропускная способность сети
- Большое количество взаимодействий пользователя с приложением (для этого может понадобиться имитация реальных условий состояния сети)

Некоторые из этих стрессовых условий могут быть созданы с помощью таких инструментов, как Monkey. Это консольная утилита, запускаемая из командной строки оболочки ADB [URL3] или, если это возможно, вручную, например, используя большие файлы или другие приложения с высокой загрузкой ЦП или потреблением памяти.

3.1.3 Тестирование защищенности

Поскольку тестирование защищенности является сложной темой, у ISTQB® имеется отдельная программа обучения [ISTQB_CTAL_SEC_2016]. Основные проблемы безопасности для мобильных приложений включают:

- Доступ к конфиденциальным данным на устройстве
- Передачу незашифрованной информации или небезопасное хранение

К тестовым условиям, которые следует учитывать при тестировании защищенности, относятся:

- Проверка входных данных для внедрения кода и переполнения буфера
- Шифрование передаваемых данных
- Шифрование локально хранимых данных
- Удаление временных данных после использования или после аварийного завершения
- Очистка текста в полях для ввода пароля

Также следует изучить 10 основных уязвимостей, связанных с мобильными устройствами, в рамках проекта Open Web Application Security Project (OWASP) [URL2].

3.1.4 Тестирование производительности

Если пользователь устанавливает приложение, и оно не отображается достаточно быстро (например, в течение трех секунд), оно может быть удалено в пользу другого приложения. Аспекты потребления времени и ресурсов являются важными факторами успеха для приложения, и для измерения этих аспектов проводится тестирование производительности.

Эффективность производительности должна быть проверена на самом устройстве, а также при его взаимодействии с серверной частью системы и другими мобильными устройствами.

Тестирование производительности всей системы должно выполняться так, как определено в стратегии тестирования; оно не является уникальным для мобильных устройств. Для получения дополнительной информации обратитесь к программе обучения ISTQB® по тестированию производительности [ISTQB_CTFL_PT_2018].

Тест производительности самого приложения должен содержать хронометрические данные для наиболее важных рабочих процессов. Вот некоторые примеры рабочих процессов для мобильного банковского приложения: «Логин», «Изменить адрес» или «Банковский перевод с помощью персонального идентификационного номера (PIN) и аутентификационного номера транзакции (TAN)». Тестировщик должен сравнивать эти хронометрические данные с аналогичными приложениями.

Помимо хронометрических измерений, важно учитывать субъективную оценку производительности пользователем. Впечатления от использования приложения могут оказать огромное влияние на то, как долго пользователь будет ждать завершения определенной функции.

3.1.5 Тестирование практичности

Данные показывают, что многие пользователи удаляют приложения со своих устройств в течение нескольких минут после установки из-за их непрактичности или плохой производительности, см. [URL4]. Именно поэтому важно проводить тестирование практичности приложения.

В связи с этим рекомендуется, чтобы дизайн пользовательского интерфейса учитывал внешний вид и поведение платформы, на которой должно использоваться приложение. Если интерфейс не соответствует ожиданиям пользователя в отношении выбранной платформы, это может вызвать серьезные негативные последствия. Таким образом, тестировщик должен иметь представление о внешнем виде и поведении используемой платформы.

Тесты практичности могут выполняться тестировщиком с использованием различных доступных эвристик и тест-туров (см. 3.3.3). При проведении тестирования практичности также помогает рассмотрение тест-персон (см. 3.3.1). Кроме того, при необходимости можно использовать лабораторию эргономики (практичности).

Результаты, выявленные в ходе тестирования практичности реальных проектов, представляют собой в основном замечания, а не дефекты. Тестировщик должен иметь возможность объяснить эти замечания команде, владельцу продукта или другим заинтересованным сторонам. Для достижения удовлетворительного уровня практичности приложение должно:

- Быть интуитивно понятным
- Позволять пользователю допускать ошибки
- Быть последовательным в формулировках и логике работы
- Не противоречить руководствам по проектированию платформ
- Обеспечивать видимость и доступность необходимой информации для каждого размера и типа экрана

Для получения дополнительной информации обратитесь к программе обучения ISTQB® по тестированию практичности [ISTQB_FLUT_2018].

3.1.6 Тестирование баз данных

Многим приложениям требуется хранить данные локально, используя различные механизмы хранения данных, такие как плоские файлы или базы данных. К тестовым условиям, которые следует учитывать при тестировании баз данных мобильных приложений, относятся:

- Проверка правильности хранения данных:
 - Синхронизация
 - Конфликты загрузки
 - Безопасность данных
 - Ограничения на данные
 - Функциональность CRUD (Create/Read/Update/Delete - Создание/Чтение/Обновление/Удаление)
 - Поиск
- Тестирование интеграции данных для данных, предоставляемых устройством (например, контактов) или сторонними приложениями (например, изображений, видео и сообщений)
- Производительность хранения данных на устройстве

3.1.7 Тестирование глобализации и локализации

Тестирование интернационализации (I18N)/глобализации приложения включает тестирование приложения для различных местоположений, форматов дат, чисел и валют, а также замену фактических строк псевдостроками.

Тестирование локализации (L10N) включает тестирование приложения с локализованными строками, изображениями и рабочими процессами для определенного региона. Например, русские и немецкие слова могут быть намного длиннее, чем в других языках. Поскольку мобильные устройства имеют разные размеры и разрешения экрана, это может привести к проблемам с переведенными строками. Эти проблемы должны быть проверены в ходе стандартных тестов глобализации/локализации.

Очень важным аспектом, который необходимо проверять, является используемый формат даты, например, ГОД - МЕСЯЦ - ДЕНЬ или ДЕНЬ - МЕСЯЦ - ГОД.

3.1.8 Тестирование доступности

Тестирование доступности проводится для того, чтобы выявить, насколько легко пользователи с ограниченными возможностями могут использовать компонент или систему. Для мобильных приложений это можно сделать, используя настройки доступности устройства и тестируя приложение для каждого параметра.

У поставщиков платформ можно получить руководства по доступности, которые следует использовать. Например, как Google [URL5], так и Apple [URL6] опубликовали руководства по обеспечению доступности для своих платформ. Также полезно получать отзывы от людей, которые испытывают необходимость в данной функции.

Для мобильного веб-контента W3C опубликовал руководство по доступности, которое также следует рассмотреть [URL7].

3.2 Дополнительные уровни тестирования, применимые к мобильным приложениям

В дополнение к обычным уровням тестирования, начиная с компонентного и вплоть до приемочного тестирования, описанным в [ISTQB_CTFL_2018], существует необходимость в дополнительных уровнях тестирования мобильных приложений.

3.2.1 Тестирование в условиях эксплуатации

Некоторые мобильные приложения необходимо тестировать в условиях эксплуатации, чтобы убедиться в их правильной работе при сценариях использования, ожидаемых от реальных пользователей. Сюда может входить тестирование в различных сетях и на различных типах коммуникационных технологий, таких как Wi-Fi или сотовые данные.

Тесты в условиях эксплуатации должны включать использование мобильных вышек, сетей, Wi-Fi и сотовой коммутации данных во время использования приложения. Тесты должны проводиться с различными скоростями загрузки и уровнями сигнала и включать обработку "мертвых зон".

Тестирование в условиях эксплуатации требует тщательного планирования и определения всех элементов, необходимых для выполнения тестов, таких как подходящие типы устройств, Wi-Fi, тарифные планы различных операторов и доступ к различным видам транспорта, необходимым для обеспечения достаточного покрытия. Кроме того, необходимо запланировать использование различных видов транспорта и их маршруты, а также время суток, когда должны проводиться испытания.

Практичность приложения - еще один важный аспект, который необходимо учитывать при проведении тестирования в условиях эксплуатации. Испытания должны включать факторы окружающей среды, такие как температура и аналогичные условия, связанные со сценарием использования.

3.2.2 Тестирование для получения разрешения от магазина приложений и пост-релизное тестирование

Перед отправкой приложения на публикацию необходимо выполнить некоторые тесты на основе чек-листов, чтобы обеспечить получение разрешения на публикацию от магазина приложений. Если выпускается обновление, то помимо этого должны быть проведены тесты, связанные с обновлением.

Чек-листы, как правило, основаны на руководствах (в том числе специфических для операционных систем) по разработке пользовательского интерфейса и использованию библиотек и API, предоставляемых магазинами приложений.

Процесс получения разрешения может занять некоторое время после подачи заявки на рассмотрение. Если в процессе утверждения будут обнаружены какие-либо проблемы, возможно, потребуется отправить новую версию, что займет дополнительное время. Эта ситуация требует тщательного рассмотрения при планировании и тестировании проекта.

Следующим уровнем тестирования является пост-релизное тестирование. Оно включает в себя загрузку и установку приложения из магазинов приложений.

3.3 Методики тестирования на основе опыта

3.3.1 Тест-персоны и мнемоники

Тест-персоны - это вымышленные персонажи, представляющие реальных клиентов. У них есть мотивы, ожидания, проблемы, привычки и цели, и тестировщики могут использовать их, когда нужно имитировать реальное поведение пользователя.

Тест-персона может иметь имя, пол, возраст, доход, образование и местоположение. В контексте мобильных устройств они могут использовать другие приложения, проверять свое устройство x раз в час, могут иметь другие устройства, а также личные качества.

Мнемоника - это средство для запоминания чего-либо. В контексте тестирования каждая буква в

мнемонике обозначает технику, метод тестирования или фокус для тестирования. Примером мнемоники является SFIDPOT [URL8]. Буквы в этой мнемонике означают следующее:

S – Structure (структура) - элементы пользовательского интерфейса, другие элементы приложения, их порядок и иерархия вызовов и т. п.

F – Function (функция) - заданные функции работают, доступны и функционируют в соответствии с требованиями и т. п.

i – Input (ввод) - все предусмотренные входные данные (например, вводимые с клавиатуры, датчиков и камеры) доступны и обрабатываются должным образом и т. п.

D – Data (данные) - данные хранятся (в том числе на SD-карте), модифицируются, добавляются и удаляются, как определено в требованиях, и т. п.

P – Platform (платформа) - конкретные функции операционной системы доступны в зависимости от настроек устройства, включая магазин для загрузки приложения, и т. п.

O – Operations (операции) - доступны действия обычного пользователя, такие как перемещение между сетями мобильной связи и Wi-Fi, и т. п.

T – Time (время) - обработка и отображение часовых поясов, времени, дат и т. п.

Примером мнемоники и эвристики, относящихся непосредственно к мобильному тестированию, является фраза “I SLICED UP FUN” [URL9]. Буквы в этой мнемонике означают следующее:

I – Inputs (входные данные)

S – Store (магазин)

L – Location (местоположение)

I – Interactions and interruptions (взаимодействия и прерывания)

C – Communication (связь)

E – Ergonomics (эргономика)

D – Data (данные)

U – Usability (практичность)

P – Platform (платформа)

F – Function (функция)

U – User scenarios (пользовательские сценарии)

N – Network (сеть)

3.3.2 Эвристика

Эвристический подход - это универсальный практический метод для решения проблем, обучения и исследований. Данный подход не является гарантированно оптимальным или совершенным, но может считаться достаточным для достижения краткосрочных целей.

Существует много эвристик для мобильного тестирования. Большинство мнемоник можно использовать как эвристику, но не каждая эвристика является мнемоникой.

3.3.3 Тест-туры

Тест-туры используются в исследовательском тестировании, чтобы дать возможность изучить приложение с определенной точки зрения и фокуса. Они позволяют понять, как работает

приложение, и создать модели рабочего процесса. Использование тест-туров является эффективным методом тестирования в условиях эксплуатации.

Примером тест-тура является тур “Достопримечательность”, где пользователь имитирует визит туриста в город и посещение им известных достопримечательностей. В следующей таблице показано, как места назначения в тест-туре можно использовать в качестве аналогии для шагов, которые необходимо выполнить в мобильном тестировании.

Места назначения в туре “Достопримечательность”	Аналогия для мобильного тестирования
Исторический квартал	Унаследованный код
Деловой район Час пик	Бизнес-логика приложения Запуск и закрытие приложения
Туристический квартал	Часть приложения, используемая новичками
Гостиничный квартал	Части приложения, активные только в спящем режиме

Объединение тестирования на основе сеансов (см. раздел 3.3.4) с тест-турами, включая использование эвристик и мнемоник, помогает повысить эффективность тестирования мобильных приложений.

В следующей таблице приведено несколько хороших примеров тест-туров и областей, которые они охватывают, для предоставления некоторых идей по тестированию. Некоторые из них можно найти в [Kohl17].

Тур для тестирования приложения	Охватываемая область
Супермодель	Внешний вид, поведение и практичность
Достопримечательность	Наиболее важные функции в приложении
Саботаж	Устойчивость к нарушениям
Функция	Новые функции
Сценарий	Весь рабочий процесс в приложении в сочетании с пользовательскими историями
Подключение	Используемые типы подключения (Wi-Fi, GSM и т. п.)

Местоположение	Корректный язык, даты, числа
Освещение	Видимость в различных условиях освещения (в темноте, на открытом воздухе, в красном свете).
Низкий уровень заряда батареи	Потери данных в приложении, вызванные низким уровнем энергии.
Дополнительные туры для тестирования приложения	Область, охватываемая в [Koh117]
Жест	Использовать все жесты везде, где это возможно
Ориентация	Изменять ориентацию экрана
Возвращение	Возвращаться назад
Движение	Делать различные движения
Местоположение	Переходить с места на место
Подключение	Изменять тип соединения или местоположение, перемещаясь
Сравнение	Сравнить с другими типами устройств
Согласованность	Проверить согласованность экранов, графический интерфейс

3.3.4 Управление тестированием на основе сеансов

Управление тестированием на основе сеансов (SBTM - session-based test management) позволяет проводить исследовательское тестирование с соблюдением временных рамок. Сеанс состоит из трех задач:

- Подготовка сеанса
- Проектирование и выполнение тестов
- Поиск дефектов и отчетность

Управление тестированием на основе сеансов обычно использует протокол сеанса, содержащий концепцию тестирования, в которой указаны цели тестирования. Кроме того, протокол сеанса используется для документирования действий по выполнению тестов.

Исследовательское тестирование - это методика тестирования на основе опыта, которая является эффективным подходом к тестированию мобильных приложений. Методики тестирования на основе опыта описаны в [ISTQB_CTFL_2018].

3.4 Процесс тестирования и подходы к тестированию мобильных приложений

3.4.1 Процесс тестирования

Основные активности процесса тестирования ISTQB® описаны в [ISTQB_CTFL_2018] и применимы также к тестированию мобильных приложений.

Существуют дополнительные аспекты, специфичные для мобильного тестирования, которые всегда следует рассматривать как часть процесса тестирования ISTQB®.

Основная группа активностей процесса тестирования	Типичные области для рассмотрения в мобильном тестировании	Ссылки на программу обучения
Планирование тестирования	<ul style="list-style-type: none"> • Комбинации устройств, на которых необходимо провести тестирование. • Использование мобильных эмуляторов и симуляторов как части тестового окружения. • Специфические трудности в тестировании мобильных приложений. • Типы тестирования, применимые к мобильным приложениям. 	<ul style="list-style-type: none"> • Раздел 4.3 • Раздел 1.7 • Раздел 3.2
Анализ и проектирование	<ul style="list-style-type: none"> • Тестирование получения разрешения от магазина приложений. • Тестирование в условиях эксплуатации. • Совместимость с устройствами. • Вид используемых лабораторий. • Типы тестирования мобильных приложений. 	<ul style="list-style-type: none"> • Раздел 3.2.2 • Раздел 3.2.1 • Раздел 3.2
Реализация и выполнение тестов	<ul style="list-style-type: none"> • Тестирование в условиях эксплуатации. • Пост-релизное тестирование загрузки и установки. 	<ul style="list-style-type: none"> • Раздел 3.2.1 • Раздел 3.1.1

	<ul style="list-style-type: none"> • Методики на основе опыта. 	[ISTQB_CTFL_2018]
Реализация и выполнение тестов	<ul style="list-style-type: none"> • Тесты основаны на руководствах платформ по пользовательскому интерфейсу и магазинам приложений. • Тесты, основанные на руководствах, обычно проводятся поставщиками платформ в рамках процесса предоставления разрешения от магазинов приложений. • Рекомендуется проводить эти тесты со стороны поставщика приложения перед передачей поставщикам платформы, чтобы избежать возможных отказов. 	

3.4.2 Подходы к тестированию

Тестирование мобильных приложений включает в себя активности, которые должны выполняться как разработчики, так и тестировщики.

Определение надлежащей глубины тестирования для каждого уровня (компонентное, интеграционное, системное тестирование, тестирование в условиях эксплуатации, тестирование для получения разрешения от магазина приложений, пост-релизное и пользовательское приемочное тестирование) важно для предоставления продуктов хорошего качества. Глубина тестирования, необходимая для каждого уровня, зависит от многих факторов, таких как архитектура приложения, сложность приложения и предполагаемая пользовательская аудитория.

Платформы разработки мобильных приложений предоставляют разнообразные инструменты поддержки тестирования на различных уровнях. Понимание инструментов и способов их применения на конкретном уровне очень важно. Например, мобильный симулятор и/или эмулятор можно использовать на уровне компонентного тестирования, если необходимо воспользоваться преимуществами предоставляемых платформой сред и программных интерфейсов API. Кроме того, мобильные симуляторы и/или эмуляторы могут использоваться на уровне системного тестирования, когда реальные устройства недоступны. Это позволяет тестировать функциональность, ограниченные аспекты практичности, а также пользовательский интерфейс.

Помимо всего прочего, ранняя реализация подходов может служить ключевым фактором для обеспечения правильной настройки устройств и своевременного выполнения всех предварительных условий.

Кроме того, важны модульные и интеграционные тесты, а также ручное тестирование (особенно на этапе тестирования в условиях эксплуатации). Для мобильных приложений характерна перевернутая пирамида тестирования [Knott15], что означает увеличение объемов ручного тестирования.

4. Платформы, инструменты и окружение мобильных приложений – 80 минут

Ключевые слова

тестирование в условиях эксплуатации, тестирование на основе сходства, удаленная тестовая лаборатория, эмулятор, симулятор

Цели обучения для главы “Платформы, инструменты и окружение мобильных приложений”

4.1 Платформы разработки мобильных приложений

МАТ-4.1.1 (K1) Вспомнить среды разработки, используемые для разработки мобильных приложений.

4.2 Общие инструменты платформ разработки

МАТ-4.2.1 (K1) Напомнить некоторые из общих инструментов, поставляемых как часть платформ разработки приложений.

НО-4.2.1 (H1) Использовать инструменты разработки программного обеспечения, чтобы создавать снимки экрана, извлекать логи и симулировать входящие события.

4.3 Эмуляторы и симуляторы

МАТ-4.3.1 (K2) Понимать разницу между эмуляторами и симуляторами.

МАТ-4.3.2 (K2) Описать использование эмуляторов и симуляторов для тестирования мобильных приложений.

НО-4.3.2 (H1) Создать и запустить симулированное/эмулированное устройство, установить на нем приложение и выполнить несколько тестов.

4.4 Создание тестовой лаборатории

МАТ-4.4.1 (K2) Провести различие между подходами к созданию тестовой лаборатории.

4.1 Платформы разработки мобильных приложений

Интегрированные среды разработки (IDE) доступны на рынке для разработки мобильных приложений. Эти IDE содержат различные инструменты, помогающие в разработке, кодировании, компиляции, установке, удалении, мониторинге, эмуляции, логировании и тестировании приложений.

Например, для разработки приложений для Android может использоваться Android Studio, а для разработки приложений для iOS - Xcode. От обычных IDE они отличаются поддержкой мобильных платформ.

Также доступны некоторые кроссплатформенные среды разработки, которые помогают при разработке мобильных приложений. Они работают на нескольких платформах и не требуют написания кода отдельно для каждой.

4.2 Общие инструменты платформ разработки

Наборы средств разработки программного обеспечения обычно предоставляют различные утилиты, которые полезны при разработке и тестировании приложений. Эти утилиты охватывают широкий спектр задач, таких как создание снимков экрана, извлечение логов, отправка случайных событий и уведомлений на устройство, создание виртуальных устройств, а также мониторинг различных параметров, таких как использование памяти и ЦП.

Некоторыми примерами таких инструментов являются диспетчер виртуальных устройств Android (AVD), Android Debug Bridge (ADB) и Android Device Monitor для Android, а также Instruments для iOS.

4.3 Эмуляторы и симуляторы

4.3.1 Обзор эмуляторов и симуляторов

В контексте данной программы термины “эмулятор” и “симулятор” относятся к мобильному эмулятору или мобильному симулятору. Термины “симулятор” и “эмулятор” иногда используются как синонимы, но это неправильно. Определения см. в глоссарии в главе 8.

Симулятор моделирует среду выполнения, в то время как эмулятор моделирует оборудование и использует ту же среду выполнения, что и физическое оборудование. Приложения, тестируемые на симуляторе, компилируются в специальную версию, которая работает на симуляторе, но не на реальном устройстве. Таким образом, оно не зависит от реальной ОС.

Напротив, приложения для развертывания и тестирования на эмуляторе компилируются в реальный байт-код, который также может использоваться реальным устройством.

Симуляторы и эмуляторы очень полезны на ранней стадии разработки, поскольку они обычно интегрируются со средами разработки и обеспечивают быстрое развертывание, тестирование и мониторинг приложений.

Симуляторы иногда используются в качестве замены реальных устройств при тестировании. Однако этот метод имеет больше ограничений, чем использование эмуляторов, поскольку тестируемое на симуляторе приложение отличается на уровне байт-кода от приложения, которое будет поставляться.

Эмуляторы также используются для снижения стоимости тестовых сред путем замены реальных устройств при тестировании. Эмулятор не способен полностью заменить мобильные устройства, поскольку его работа может отличаться от работы того устройства, которое он пытается имитировать. Кроме того, некоторые функции могут не поддерживаться, такие как мультикасание, акселерометр и другие. Это частично вызвано ограничениями платформы, используемой для запуска эмулятора.

4.3.2 Использование эмуляторов и симуляторов

Использование эмуляторов и симуляторов для мобильного тестирования может быть полезным по разным причинам.

Каждая среда разработки мобильных операционных систем обычно поставляется со своими собственными эмулятором и симулятором. Также доступны сторонние эмуляторы и симуляторы.

Тестировщик может использовать любой эмулятор или симулятор, который соответствует его цели. Использование эмулятора или симулятора предполагает их запуск, установку на них необходимого приложения, а затем тестирование приложения, как если бы оно было на реальном устройстве.

Обычно эмуляторы и симуляторы позволяют устанавливать различные параметры использования.

Эти настройки могут включать в себя эмуляцию сети на разных скоростях, с разным уровнем сигнала и потерей пакетов, изменение ориентации экрана, генерацию прерываний и данные GPS о местоположении. Некоторые из этих настроек (например, глобальная геолокация или уровни сигнала) могут быть очень полезны, так как их сложно или дорого воспроизводить на реальных устройствах.

Для подключения к эмуляторам в целях установки может потребоваться использование инструментов командной строки, таких как Android Debug Bridge (ADB) для Android, или подключение из интегрированной среды разработки, например Xcode или Android Studio.

4.4 Создание тестовой лаборатории

Для создания мобильной тестовой лаборатории используются следующие подходы:

Локальная лаборатория

В локальной лаборатории все устройства, эмуляторы и симуляторы находятся на месте проведения работ. Выбор устройства может быть сделан на основе различных факторов, таких как рейтинг устройства (найденный в Google или другой аналитике), операционная система и ее версия, размеры и плотность пикселей экрана, доступность и стоимость, специальные функции и важность для целевой аудитории.

Преимущества локальной лаборатории включают в себя доступность устройств для специфического тестирования на основе сходства, а также аспекты, характерные для датчиков и сенсоров, такие как батарея, касание и повышенная безопасность.

Создание этого типа лаборатории может потребовать больших расходов в зависимости от устройств, которые будут закупаться и обслуживаться. Дополнительные сложности могут возникнуть со своевременной доступностью и с тестированием в разных локациях и окружениях.

Удаленная тестовая лаборатория

Такие лаборатории важны и полезны для тестирования, когда устройства или сети физически недоступны на месте проведения работ. Удаленный доступ к устройству (RDA) позволяет получить доступ через сетевое соединение к различным устройствам, расположенным в центре обработки данных поставщика. Каждый потенциальный поставщик RDA должен быть оценен на соответствие требованиям, особенно в отношении безопасности.

Некоторые удаленные лаборатории предоставляют следующие дополнительные функции:

- Отдельные версии физических устройств (например, лаборатория мобильных устройств Samsung)
- Однотипные устройства, предназначенные только для конкретной операционной системы и версии
- Роботизированные руки для выполнения операций, связанных с касанием и жестами
- Соединения виртуальной частной сети (VPN) для предоставления доступа к устройству
- Сотовая связь, предоставляемая различными провайдерами сотовой сети
- Инструменты и сервисы автоматизации

Некоторые из факторов, которые следует учитывать при использовании удаленных лабораторий тестирования, включают медленную скорость реакции устройства и ограниченные возможности взаимодействия с устройствами (мультикасание и жесты). Такие лаборатории могут быть экономически эффективными для разового использования, но они, как правило, более дорогие, если используются в течение продолжительных периодов времени для широкого спектра устройств.

Другие факторы включают доступность запрашиваемой платформы по сравнению с необходимостью получения доступа к отсутствующим устройствам в локальной лаборатории, а также масштабируемость лаборатории, поскольку она может увеличиваться и уменьшаться по мере

развития проекта.

Тестовые сценарии, включающие такие датчики, как NFC/Bluetooth, или уровень заряда батареи, часто сложно протестировать в облаке. Тем не менее, различное географическое местоположение удаленных лабораторий может помочь с тестами, которые требуют подключения к сети и GPS.

Тестовая лаборатория может использовать один из этих подходов или их комбинацию в зависимости от типа тестов, которые необходимо выполнить.

5. Автоматизация выполнения тестов – 55 минут

Ключевые слова

программный интерфейс (API), тестирование на основе агента пользователя, тестирование на основе устройства, отчет о тестировании

Цели обучения для главы “Автоматизация выполнения тестов”

5.1 Подходы к автоматизации

MAT-5.1.1 (K2) Провести различие между общими подходами и средами автоматизации тестирования мобильных приложений.

5.2 Методы автоматизации

MAT-5.2.1 (K2) Описать различные методы автоматизации тестирования мобильных приложений.

5.3 Оценка инструментов автоматизации

MAT-5.3.1 (K1) Вспомнить различные параметры, которые следует учитывать при оценке инструментов автоматизации мобильного тестирования.

5.4 Подходы к созданию лаборатории автоматизации тестирования

MAT-5.4.1 (K2) Провести различие между общими подходами к созданию тестовых лабораторий, учитывая их преимущества и недостатки в контексте автоматизации тестирования.

5.1 Подходы к автоматизации

Существуют различные подходы и среды автоматизации, которые можно использовать при тестировании мобильных приложений. Выбор подхода будет частично определяться типом приложения.

Существует два общих подхода к автоматизации тестирования:

- Тестирование на основе агента пользователя
- Тестирование на основе устройства

Тестирование на основе агента пользователя использует строку идентификатора, отправленную определенным браузером для его имитации на определенном устройстве. Этот подход может использоваться для мобильных веб-приложений. Тестирование на основе устройства, в свою очередь, включает в себя запуск тестируемого приложения непосредственно на устройстве. Этот подход может использоваться для всех типов мобильных приложений.

Тип приложения также может определять среду автоматизации тестирования, которая подходит для этого приложения. Мобильные веб-приложения можно протестировать с помощью обычных инструментов автоматизации веб-приложений на рабочем столе, в то время как приложения для

конкретной платформы могут потребоваться специальные инструменты. Поставщики платформ также могут предоставлять инструменты автоматизации, предназначенные для конкретной платформы.

Подходы к автоматизации, используемые для обычных приложений, часто применимы и к мобильным приложениям. К ним относятся захват/воспроизведение, тестирование на основе данных, ключевых слов и поведения, как описано в документах “Программа подготовки ISTQB Базового уровня” [ISTQB_CTFL_2018] и ISTQB® Advanced Level Specialist Test Automation Engineer syllabus [ISTQB_CTAL_TAE_2016].

Ключевые возможности, которые обычно должна включать среда автоматизированного тестирования мобильных приложений:

- Идентификация объекта
- Операции с объектами
- Отчеты о тестировании
- Программные интерфейсы и расширенные возможности
- Полная и понятная документация
- Интеграция с другими инструментами
- Независимость от практики разработки тестов

5.2 Методы автоматизации

Чтобы разрабатывать автоматизированные тесты, тестировщик должен понимать механизм записи или создания сценариев автоматизации и то, как получить доступ к графическим объектам приложения, таким как кнопки, списки и поля ввода, и взаимодействовать с ними.

Существует несколько методов идентификации графического объекта, используемых для автоматизации мобильного тестирования. К ним относятся распознавание изображений, символов/текста, а также распознавание объектов (встроенных или веб-объектов, в зависимости от типа приложения).

Тестировщик мобильных приложений должен не только применять методы обнаружения и идентификации графических объектов, но также понимать, какой метод будет наиболее эффективным для обеспечения успешного запуска тестов параллельно и непрерывно на большом количестве мобильных устройств.

Ключевые различия между методами создания сценария:

Элемент сравнения	Идентификация объекта	Сравнение изображения/оптического распознавания символов
Надежность	Если идентификатор постоянен, ориентация экрана может быть изменена. Риск состоит в том, что могут быть идентифицированы и задействованы объекты внутри кода, скрытые от пользователя. Это может привести к ложноотрицательным результатам тестирования.	Изображения можно масштабировать в соответствии с размером экрана, но при изменении ориентации тесты будут считаться непройденными.
Пользовательский интерфейс	Обычно требуются ручные сценарии, по крайней мере,	Полное тестирование на основе графического

	для улучшения удобства чтения и сопровождения записанных сценариев.	интерфейса без создания сценариев.
Скорость выполнения	Быстрее, чем сравнение изображения/распознавания, особенно при использовании встроенных инструментов, предоставленных производителем системы.	Обычно медленнее из-за необходимости попиксельно сравнивать экран с базовым изображением.
Сопровождение	Зависит от качества сценариев тестирования.	В основном по предоставлению измененных базовых изображений.
Сложности разработки	Требуется знание языка сценариев и методов проектирования программного обеспечения, чтобы создать эффективное решение для автоматизации.	Необходимость генерации базовых изображений, особенно при частых изменениях приложения.

5.3 Оценка инструментов автоматизации

Чтобы успешно создавать решения для автоматизации тестирования, командам по автоматизации тестирования необходимо выбрать подходящий набор инструментов. Необходимо учитывать основные отличия доступных инструментов и их пригодность для требований проекта (см. также [ISTQB_CTAL_TAE_2016]).

Параметры оценки инструментов автоматизации тестирования можно разделить на две категории:

- Организационное соответствие
- Техническое соответствие

Параметры организационного соответствия описаны в главе 6.2 документа “Программа подготовки ISTQB Базового уровня” [ISTQB_CTFL_2018].

Параметры технического соответствия включают в себя следующее:

- Требования к автоматизации тестирования и сложности ее реализации, например использование приложением новых функций, таких как сканер лица (FaceID), отпечатки пальцев и чат-боты.
- Требования к среде тестирования, такие как изменяющиеся условия сети, импорт или создание тестовых данных и создание виртуальной среды на стороне сервера.
- Отчетность тестирования и возможности обратной связи.
- Способность среды управлять выполнением тестов в больших масштабах либо локально, либо в тестовой лаборатории в облаке.
- Интеграция тестовой среды с другими инструментами, используемыми в организации.
- Доступность поддержки и документации для текущих и будущих обновлений.

5.4 Подходы к созданию лаборатории автоматизации тестирования

Выполняя тестирование мобильных приложений, разработчики и тестировщики имеют возможность выбора лаборатории тестирования устройств, которую они будут использовать для автоматизации тестирования.

- Локальная лаборатория тестирования устройств
- Удаленная лаборатория тестирования устройств

Могут применяться различные комбинации этих подходов. Описание и сравнение их основных характеристик приводится в разделе 4.4.

Обслуживание локальных лабораторий тестирования устройств, как правило, сложно и требует много времени. Для ранних этапов разработки и тестирования мобильных приложений наилучшим образом подходит тестирование на локальных устройствах параллельно с использованием эмуляторов и симуляторов.

По достижении более поздней стадии разработки приложения командам необходимо выполнять полное регрессионное, функциональное и нефункциональное тестирование. Такие тесты лучше всего выполнять в лаборатории, обладающей полным комплектом необходимых устройств. На этой стадии используется удаленная лаборатория тестирования устройств, которая управляется, непрерывно обновляется и поддерживается в облаке. Такие лаборатории дополняют локальную лабораторию тестирования устройств и обеспечивают наличие и актуальность достаточного числа комбинаций устройств и операционных систем. Используя преимущества наиболее распространенных удаленных лабораторий, команды получают доступ к большему набору поддерживаемых возможностей, включая более подробные отчеты о тестировании и расширенные возможности автоматизации тестирования.

Наконец, при проведении масштабного тестирования с помощью среды автоматизации или в рамках непрерывной интеграции стабильность тестовой лаборатории является ключом к эффективности и надежности тестирования. Такие лаборатории обычно предназначены для того, чтобы устройства и операционные системы были всегда доступны и стабильны.

Удаленные лаборатории тестирования устройств не всегда необходимы на более поздних этапах разработки приложений. Хорошо спроектированные и поддерживаемые локальные лаборатории тестирования устройств могут подходить для тестирования так же, как и любая удаленная лаборатория тестирования, а иногда даже лучше.

6. Список использованных источников

6.1 Документы ISTQB®

- [ISTQB_CTFL_2018]:
ISTQB® Certified Tester – Foundation Level Syllabus – Version 2018
Программа подготовки ISTQB Базового уровня 2018
- [ISTQB_FLAT_2014]:
ISTQB® Certified Tester – Foundation Level Extension Syllabus – Agile Tester – Version 2014
Программа подготовки ISTQB Базовый уровень - Тестировщик в сфере Гибких Методологий 2014
- [ISTQB_FLUT_2018]:
ISTQB® Certified Tester – Foundation Level Specialist Syllabus – Usability Testing – Version 2018
- [ISTQB_CTFL_PT_2018]:
ISTQB® Certified Tester – Foundation Level Specialist Syllabus – Performance Testing – Version 2018
Программа подготовки ISTQB Базовый уровень - Тестирование Производительности
- [ISTQB_CTAL_SEC_2016]:
ISTQB® Certified Tester – Advanced Level Specialist Syllabus – Security Testing – Version 2016
- [ISTQB_CTAL_TAE_2016]:
ISTQB® Certified Tester – Advanced Level Specialist Syllabus - Test Automation Engineer - Version 2016
- [ISTQB_GLOSSARY]:
ISTQB®'s Glossary of Terms used in Software Testing, Version 3.2

6.2 Литература

- [Knott15] Knott, D., “Hands-On Mobile App Testing”, Addison-Wesley Professional, 2015, ISBN 978-3-86490-379-3
- [Kohl17] Kohl, J., “Tap into mobile application testing”, leanpub.com, 2017, ISBN 978-0-9959823-2-1

6.3 Книги и статьи для дальнейшего изучения

- Boris Beizer, “Black-box Testing”, John Wiley & Sons, 1995, ISBN 0-471-12094-4
Борис Бейзер, “Тестирование черного ящика”, Питер, 2004, ISBN 5-94723-698-2
- Rex Black, “Agile Testing Foundations”, BCS Learning & Development Ltd: Swindon UK, 2017, ISBN 978-1-78017-33-68
- Rex Black, “Managing the Testing Process”(3e), John Wiley & Sons: New York NY, 2009, ISBN 978-0-470-40415-7
- Hans Buwalda, “Integrated Test Design and Automation”, Addison-Wesley Longman, 2001, ISBN 0-201-73725-6
- Lee Copeland, “A Practitioner's Guide to Software Test Design”, Artech House, 2003, ISBN 1-58053-791-X
- Rick David Craig, Stefan P. Jaskiel, “Systematic Software Testing”, Artech House, 2002, ISBN 1-580-53508-9

6.4 Ссылки (Веб/Интернет)

Примечание: Все ссылки являются рабочими по состоянию на 5 января 2019 года.

- [URL1] <http://gs.statcounter.com/>
- [URL2] www.owasp.org
- [URL3] <https://developer.android.com/studio/test/monkey>
- [URL4] <https://www.google.de/amp/s/techcrunch.com/2016/05/31/nearly-1-in-4-people-abandon-mobile-apps-after-only-one-use/amp/>
- [URL5] <https://www.google.com/accessibility/>
- [URL6] <https://www.apple.com/uk/accessibility/>
- [URL7] <https://www.w3.org/WAI/standards-guidelines/mobile/>
- [URL8] <https://www.slideshare.net/karennjohnson/kn-johnson-2012-heuristics-mnemonics>
- [URL9] <http://www.kohl.ca/articles/SLICEDUPFUN.pdf>

7. Приложение А – Цели обучения/Уровень знаний

Следующие цели обучения определены применительно к текущей программе обучения. Каждая тема в программе будет проверяться в соответствии с целями обучения, определенными для нее.

7.1 Уровень 1: Запомнить (K1)

Кандидат может различать, запоминать и вспоминать термины или понятия.

Ключевые слова: определить, запомнить, найти, вспомнить, различать, узнать

Примеры:

Определить понятие "отказ" как:

- "Отказ в предоставлении услуги конечному пользователю или заказчику" или
- "Отклонение поведения компонента или системы от ожидаемого поведения, действия или результата"

7.2 Уровень 2: Понять (K2)

Кандидат может указать причины или пояснить утверждения, относящиеся к теме, а также резюмировать, сравнивать, классифицировать, разделять по категориям и приводить примеры.

Ключевые слова: резюмировать, обобщать, абстрагировать, классифицировать, сравнивать, изображать, сопоставлять, приводить примеры, интерпретировать, переводить, формулировать, делать выводы и заключения, разделять по категориям, строить модели

Примеры:

Пояснить причину, почему анализ и проектирование тестов должны выполняться как можно раньше:

- Исправление найденных дефектов будет стоить дешевле.
- Наиболее серьезные дефекты будут найдены раньше.

Найти сходства и различия между интеграционным и системным тестированием:

- Сходства: объекты тестирования для интеграционного и системного тестирования включают более чем один компонент и в обоих случаях включают нефункциональные типы тестирования.
- Различия: в интеграционном тестировании упор делается на интегрируемость и взаимодействие компонентов, в системном - на всю систему в целом от начала до конца.

7.3 Уровень 3: Применить (K3)

Кандидат умеет выбирать правильное применение понятия или техники и использовать их в данном контексте.

Ключевые слова: реализовать, выполнить, использовать, следовать процедуре, применить процедуру

Примеры:

- Правильно определять граничные значения допустимых и недопустимых классов.
- Правильно выбирать тестовые сценарии из заданной диаграммы переходов, чтобы покрыть

все переходы.

Ссылка (к уровням целей обучения):

Anderson, L. W. and Krathwohl, D. R. (eds) (2001) *A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives*, Allyn & Bacon: Boston MA

8. Приложение Б – Глоссарий специальных терминов

Термин глоссария	Определение
ADB	Отладочный мост Android (Android Debug Bridge) - инструмент командной строки, который позволяет взаимодействовать с устройством.
ADM	Монитор устройств Android (Android Device Monitor) - автономный инструмент, предоставляющий пользовательский интерфейс для инструментов отладки и анализа приложений на Android.
Android Studio	Официальная интегрированная среда разработки (IDE) для Android. Android Studio предоставляет инструменты для создания приложений на устройствах Android любого типа.
AVD	Виртуальное устройство Android (Android Virtual Device).
Bluetooth	Технология беспроводной связи ближнего действия.
CRUD	Мнемоника для действий, которые применяются к данным (Создание/Чтение/Обновление/Удаление) (Create/Read/Update/Delete).
DPI/PPI	Количество точек/пикселей на дюйм (Dots/pixels per inch) - число, выражающее разрешение экрана в точках или пикселях.
Force touch	Технология, разработанная корпорацией Apple и позволяющая сенсорным панелям и сенсорным экранам распознавать различные значения силы, приложенной к их поверхностям.
freemium-приложение	Бизнес-модель, при которой пользователи ничего не платят за загрузку приложения, но им предлагается сделать дополнительные покупки в приложении.
GPS	Система глобального позиционирования (Global Positioning System). Сеть спутников рассылает сигналы времени по всему миру. Учитывая сигналы не менее 3 спутников, приемник может рассчитать свое положение относительно спутников с помощью триангуляции.

GSM	Глобальная система мобильной связи (первоначально Groupe Spécial Mobile) - стандарт, разработанный Европейским институтом телекоммуникационных стандартов (ETSI) для описания протоколов цифровых сотовых сетей второго поколения, используемых мобильными устройствами. В настоящее время самый распространенный стандарт мобильной связи в мире.
GSM-сота	Часть сети GSM, которая может быть определена по уникальному идентификатору соты.
I18N	Нумероним (числовое слово), означающий "интернационализация".
IDE	Интегрированная среда разработки (Integrated Development Environment) - программное приложение, которое предоставляет программистам ЭВМ широкие возможности для разработки программного обеспечения.
Instruments	Средства анализа и тестирования производительности, входящие в состав набора инструментов Xcode.
IoT-устройство	Интернет вещей (Internet of Things) - устройство или, например, необходимый датчик, подключенный к Интернету.
L10N	Нумероним (числовое слово), означающий "локализация".
NFC	Коммуникация ближнего поля (Near Field Communication) - технология беспроводной связи ближнего действия.
OCR	Оптическое распознавание символов (Optical Character Recognition) - распознавание текста, содержащегося в электронном изображении, и его преобразование в текстовые данные, используемые для представления символов в компьютере.
OTA	Обновление по воздуху (over the air) - способ передачи данных с помощью радиосигналов, обычно используемый для обозначения установки приложения на устройство напрямую от источника, не подключенного через кабель.
QR-код	Код быстрого реагирования (Quick Response Code) - товарный знак для матричных, или двумерных, штрих-кодов.
SDK	Комплект для разработки программного

	обеспечения (Software Development Kit) - набор инструментов и библиотек для разработки программного обеспечения на определенной платформе.
VPN	Виртуальная частная сеть (Virtual Private Network) - зашифрованный частный канал в общедоступной сети.
Xcode	Интегрированная среда разработки приложений для платформ OS X и iOS, разработанная Apple.
“мёртвая зона”	Место, где отсутствует беспроводная телекоммуникационная сеть.
2G	Второе поколение мобильных беспроводных телекоммуникационных технологий.
3d-touch	см. “Force Touch”
3G	Третье поколение мобильных беспроводных телекоммуникационных технологий.
4G	Четвертое поколение мобильных беспроводных телекоммуникационных технологий.
5G	Пятое поколение мобильных беспроводных телекоммуникационных технологий.
активное приложение	Приложение, которое запускается на переднем плане устройства для прямого взаимодействия с пользователем.
альбомный режим	Ориентация устройства, при которой ширина экрана больше его высоты.
асинхронная передача данных	Тип передачи данных, при котором данные могут передаваться время от времени, а не непрерывно.
байт-код	Набор команд, разработанный для эффективного выполнения программным интерпретатором. Также называется переносимым кодом или р-кодом.
библиотека	Набор энергонезависимых ресурсов, используемых компьютерными программами, то есть функциями приложения.
браузерное (веб-) приложение	Приложение, которое размещается в Интернете и используется через браузер.
взлом для Android	Процесс получения root-доступа к операционной системе устройства. Термин обычно используется применительно к платформе Android. Аналогвзлома для iOS.
взлом для iOS	Расширение прав доступа с целью устранения ограничений программного обеспечения,

	налагаемых операционной системой. Термин обычно используется на iOS. Аналогичновзлому для Android.
внешний вид и поведение	Визуальное и эмоциональное впечатление от чего-либо.
внешняя память	Дополнительная память, которая добавляется в устройство через стандартный интерфейс. В настоящее время для мобильных телефонов наиболее распространены SD-карты.
внутренняя память	Память, которая является частью аппаратного обеспечения устройства.
гибридное приложение	Приложение, сочетающее в себе встроенные веб-технологии. Обычно гибридное приложение использует встроенный фрейм, который должен быть установлен на устройстве, для взаимодействия с библиотеками устройства и т. п. Кроме того, отображается контент, полученный с веб-сервера.
глобализация	См. “интернационализация”.
датчик	Устройство, модуль или подсистема, целью которых является обнаружение событий или изменений в своем окружении и отправка информации другому электронному устройству, часто компьютерному процессору.
жест	Определенный шаблон взаимодействия, такой как сведение/разведение пальцев или свайп, для активации определенных функций устройства. Например, сведение/разведение пальцев обычно используется для увеличения или уменьшения масштаба экрана интеллектуального устройства.
интернационализация	Процесс подготовки приложения к его последующей локализации.
коммуникатор	Класс мобильных телефонов, который предоставляет больше функций, чем обычный телефон (например, браузер), но не обеспечивает полную функциональность смартфона.
конфиденциальные данные	Данные, которые нуждаются в особой защите, например пароли и личные данные.
конфликт загрузки	Ошибка при попытке загрузить файл, который уже присутствует в месте назначения загрузки.
корпоративное приложение	Приложение, созданное для внутреннего использования в организации и не предназначенное для публичного

	использования.
кросс-платформенная среда разработки	Среда для разработки приложения для различных платформ, использующих одну и ту же кодовую базу.
локализация	Процесс адаптирования приложения или продукта для определенного региона с помощью таких действий, как перевод и настройка форматов.
локальная лаборатория	Лаборатория, которая физически расположена в том же месте, что и пользователь лаборатории.
магазин приложений	Платформа для распространения приложений, на которую разработчики могут загружать свои приложения, а пользователи - искать приложения для загрузки и установки их на свою платформу.
мнемоника	Техника запоминания.
многоуровневая архитектура	Подход к разработке серверной части приложений, когда 2 или более сервера предоставляют специализированные функции.
мобильная ОС	Операционная система, специально разработанная для мобильных устройств.
мобильная платформа	Опорная инфраструктура вокруг мобильной операционной системы, обычно включающая инструменты разработки, саму операционную систему и канал распространения приложений.
мобильное пространство	Обобщающий термин, который включает в себя все, что касается технологии мобильных устройств, от рынка и его участников до устройств и приложений.
мобильный симулятор	Виртуальная среда выполнения. Например, симулятор iOS имитирует iOS, но на самом деле эта iOS не является настоящей.
мобильный эмулятор	Виртуальное представление аппаратной платформы. Например, эмулятор Android - это виртуальное оборудование, на котором запущен образ настоящей ОС Android. На оборудовании может быть развернут образ ОС, и он будет работать так же, как настоящая ОС.
мультикасание	Тип взаимодействия с устройством, при котором одновременно происходят различные события касания.
мультиплатформенные приложения	Приложения, предназначенные и разработанные для работы на нескольких

	платформах с использованием одной и той же кодовой базы для всех платформ.
настройки	Общие параметры конфигурации устройства или приложения, которые могут быть изменены пользователем.
носимое устройство (“умная” одежда)	Компьютерное устройство, которое надевается на тело. Пример: часы или очки.
область просмотра	Виртуальный экран, используемый браузером для адаптации макета к экрану.
обратная совместимость	Возможность работы приложения на предыдущих версиях платформ.
обычный телефон	Мобильный телефон с минимальным набором функций, таких как звонки, хранение телефонных номеров, отправка SMS, часы и будильник.
одноуровневая архитектура	Подход к разработке серверной части приложения, при котором один сервер предоставляет все необходимые сервисы для приложения.
ориентация	Расположение объекта в мобильном устройстве; обычно используется для обозначения способа использования устройства. Может быть альбомной или портретной.
переполнение	Ситуация, при которой входящие данные превышают объем свободного места на устройстве.
планшет	Тип мобильных устройств с экранами 7” и более.
платное приложение	Приложение, которое приносит прибыль, будучи продаваемым в магазинах приложений.
плоский файл	Файл, не имеющий внутренней иерархии.
покупки в приложении	Дополнительный контент и функции, доступные прямо из приложения.
портретный режим	Ориентация устройства, при которой высота экрана больше его ширины.
предустановленное приложение	Мобильное приложение, установленное производителем устройства. Обычно пользователь не может удалить эти приложения.
прерывание	Событие, которое происходит во время другого события.
приложение для конкретной платформы	Приложение, специально разработанное для определенной платформы с использованием

	программных интерфейсов и инструментов разработки, предоставляемых этой платформой.
приложение на основе транзакций	Приложение, в котором с пользователя снимается плата за транзакцию.
проверка данных	Оценка того, являются ли данные правильными, точными, последовательными и полезными.
промежуточное хранение данных	Способ синхронизации данных, при котором данные хранятся локально и перенаправляются на сервер при наличии соответствующего сетевого подключения.
пространство экрана	Количество места, предоставляемого экраном.
протокол сеанса	A document to scope and record a test session. Документ для охвата и записи сеанса тестирования.
режим блокировки/“не беспокоить”	Режим работы мобильных устройств, который может быть активирован пользователем для блокировки определенных функций - общих уведомлений и звонков.
режим полета	Специальный режим работы мобильных устройств, при котором отключены радиопередатчики, чтобы предотвратить помехи в системах связи/обеспечения полетов.
рекламные приложения	Модель монетизации приложения, при которой организации-разработчики зарабатывают деньги с помощью рекламы, показываемой в приложении.
сеанс	Событие, ограниченное по времени.
серверная часть	Часть системы, которая обеспечивает функциональность других систем.
синхронизация данных	Процесс приведения данных из двух или более источников в одно и то же состояние.
синхронная передача данных	Способ передачи данных, при котором потоки данных отправляются и принимаются с одинаковой скоростью и разделяются сигналами синхронизации.
смартфон	Карманный персональный компьютер с мобильной операционной системой и встроенной широкополосной мобильной сотовой связью для совершения звонков, отправки SMS (служба коротких сообщений) и передачи данных через Интернет.
соотношение сторон	Отношение ширины экрана или изображения к

	его высоте.
состояние электропитания	Установленные пользователем или заранее заданные параметры потребления энергии, которые могут быть активированы на мобильном устройстве.
сотовая сеть	Сотовая сеть - это сеть, созданная из нескольких независимых, но соединенных ячеек.
сотовые данные	Данные, передающиеся по сотовой сети.
сохранение данных приложения	Созданные пользователем данные и/или содержимое приложения сохраняются на устройстве после деинсталляции приложения и остаются доступными для других приложений или для того же самого приложения в случае его переустановки.
среда выполнения	Реализация модели выполнения. Также известна как система поддержки выполнения.
сторонний магазин приложений	Платформа для распространения приложений, которая не управляется поставщиком платформы.
сторонняя загрузка	Загрузка/установка приложения с помощью средств, отличных от магазина приложений.
тест-персона	Модель/архетип определенной группы пользователей.
тип мобильного устройства	Классификация мобильных устройств по их основным характеристикам. К типичным устройствам относятся обычный телефон, коммуникатор, смартфон, фаблет, планшет и носимое устройство.
толстый клиент	В клиент-серверных приложениях - клиент, созданный для обработки части данных.
тонкий клиент	Тип клиента в клиент-серверных приложениях, который переносит основную часть задач по обработке данных на сервер.
уведомление	Объявление, отправляемое устройством.
удаленный доступ к устройству	Взаимодействие с устройством, физически расположенным в месте, отличном от местонахождения пользователя, обычно через Интернет.
устройство-компаньон	Компьютерное устройство, предназначенное для совместной работы с зависимым интеллектуальным устройством.
фоновое приложение	Приложение, которое работает в фоновом режиме.
фрагментация устройств	Разнообразие доступных устройств, их

	уникальная конфигурация оборудования и влияние на приложения и взаимодействие с пользователем.
целостность данных	Точность и согласованность данных в течение всего жизненного цикла, включая хранение, обработку и поиск.
частота ЦП	Тактовая частота процессора.
штрих-код	Оптическое представление данных, распознаваемое ЭВМ.
экранная клавиатура	Виртуальная клавиатура, изображенная на экране устройства.
эмулятор	Программное приложение, которое имитирует поведение оборудования.
энергопотребление	Количество потребляемой энергии.
энергосберегающий режим	Режим работы мобильных устройств, который может быть активирован пользователем или самим устройством для экономии энергии.
ярлык приложения	Ярлык для заданного разработчиками набора действий в приложении на Android 7.1 и выше.