

Шпаргалка

Веб-приложения используют клиент-серверную архитектуру и адаптированы под мобильные устройства. Веб-приложения не нужно устанавливать.

Нативные приложения нужно устанавливать: сначала нужно скачать установочный файл.

Гибридные приложения комбинируют свойства двух предыдущих: в основе «гибрида» — нативное приложение, его пишут отдельно под каждую платформу. Веб-содержание наполняют HTML, CSS и JavaScript — они могут быть едины для всех платформ.

Android — операционная система на основе Linux, её разрабатывает Google. Работает на смартфонах, планшетах, фитнес-браслетах, умных часах, электронных книгах, и даже телевизорах.

iOS — ОС от компании Apple. У неё закрыт исходный код, поэтому его не могут применять сторонние производители. iOS есть только на устройствах компании Apple: планшетах, смартфонах, плеерах.

Разрешение экрана — количество пикселей на экране в длину и ширину. Чем их больше, тем чётче изображение. Обычно описывают в портретной ориентации экрана: сначала указывают разрешение по вертикали, затем — по горизонтали. Например, 1920x1080.

Размер экрана — это длина и ширина устройства. Обычно в документации приводят диагональ экрана в дюймах. Например, диагональ iPhone 8 Plus — 5.5 дюймов.

Эмуляторы — это приложения, которые позволяют имитировать работу мобильного устройства с разными параметрами и функциональностью. Например, можно задать определённую версию ОС и разрешение экрана.

Лог — запись о событии, которое происходит в приложении. Логи помогают разработчику быстрее выяснить, в чём именно проблема.

Уровни логирования

- **Verbose.** Все сообщения логов. Это значение по умолчанию — самый высокий уровень логирования, который применяют для детальной отладки.
- **Debug.** Подробные уведомления, которые помогают отладить систему. Второй уровень.
- **Info.** Сообщения о процессах в системе и в приложении. Третий уровень.
- **Warning.** Предупреждение о том, что произошло событие, на которое необходимо обратить внимание. Четвёртый уровень.
- **Error.** Ошибки системы. Пятый уровень.
- **Assert.** Специальный уровень, которым пользуются разработчики, чтобы отладить приложение.

Аппаратные особенности — это:

- Прерывание работы приложения — все действия, которые прерывают работу текущего приложения. Например, блокировка экрана.
- Датчики девайса — то, что реагирует на окружающую среду. Например, датчик освещённости на корпусе смартфона.
- Интернет — скорость и качество связи соединения.
- Геолокация — способ определения координат местонахождения устройства и точность этого определения.
- Производительность — характеристики процессора, памяти и времени отклика на действия пользователя.

Прерывания бывают нескольких типов:

- **Входящий вызов.** Приложение должно продолжить работу, а не «вылетать», если на твой номер кто-то позвонил.
- **Переключение на другое приложение и возврат в текущее.** Например, тебе пришло всплывающее уведомление о новом сообщении в мессенджере в то время, как ты заполняешь форму регистрации в

приложении. Если ты переходишь в мессенджер и возвращаешься в форму, нужно убедиться, что данные не сбросились.

- Блокировка экрана. Если заблокировать экран, например, Яндекс.Музыка должна продолжить проигрывать аудио-файл.
- Действия при низком заряде батареи. При экстренно низком заряде батареи некоторые смартфоны принудительно закрывают приложения, чтобы сэкономить энергопотребление.

При тестировании **датчиков** смартфона обрати внимание, как работают:

- Гироскоп. Определяет положение смартфона в пространстве. Например, он помогает распознать, что пользователь повернул устройство в другую ориентацию.
- Акселерометр. Помогает измерить скорость перемещения устройства. Например, тебе нужно протестировать, корректно ли в фитнес-приложении отслеживается количество шагов.
- Датчик освещённости. Измеряет светосилу, и система подстраивает яркость экрана смартфона.
- NFC-чип. Позволяет работать с NFC-совместимыми устройствами, считывая с них данные. Например, можно считать ID транспортной карты, а затем приложение запросит данные о состоянии счёта у поставщика услуг по этому ID.
- Биометрия. Смартфоны работают со сканером отпечатка пальца и распознаванием лица. Важно проверять, как это влияет на работу приложения, если в нём есть такая функциональность. Например, в мобильном приложении банка.

Обязательно тестируй, как работает приложение при разных **параметрах связи**:

- Тип сети: мобильная сеть или Wi-Fi. За каждый тип отвечает свой модуль. Это может спровоцировать «провалы», когда один тип сети переключается на другой. Также стоит помнить, что в реальной жизни

качество мобильной связи неидеально: она может пропадать в метро или быть неустойчивой в новых районах. Приложение должно обрабатывать такие ситуации корректно.

- Скорость мобильной сети: 2G, 3G, LTE. 2G считают низкой скоростью, 3G и LTE — высокой скоростью мобильного интернета.
- Авиарежим. Отсутствие и Wi-Fi, и мобильного интернета. Если приложение совсем не работает без интернета, оно должно оповещать об отсутствии соединения. Если функциональность частично работает в офлайн-режиме, у пользователя должна быть возможность работать с приложением.

При тестировании **производительности** обращай внимание на:

- Длительность запуска приложения. Уточни в требованиях, сколько должно загружаться приложение, и проверь, что оно действительно загружается столько или меньше. Чтобы точно зафиксировать это, можно снять видео экрана.
- Количество кадров в секунду (Frames per Second). Если скорость — больше или равна 60 кадров/сек, анимация будет плавной. Посмотреть эту статистику можно в специальном инструменте. Например, FPS Meter.
- Расход заряда батареи. Один из показателей эффективности приложения — то, как оно расходует энергию. Есть два показателя: когда ты пользуешься приложением, и когда оно свёрнуто. Расход энергии можно посмотреть в настройках телефона или через специальные инструменты. Самый надёжный способ — подключиться напрямую к аккумулятору устройства.

Когда тебе нужно протестировать **новую версию приложения**, проверяй:

- Что обновление корректно устанавливается не только с предыдущей, но и более ранних версий.
- Обновление при уже запущенном приложении. Например, можно посмотреть, что информация о его версии изменилась.

- Запуск после обновления. Нужно убедиться, что приложение не будет "падать" при запуске. В этом случае пользователю придется удалить и заново установить приложение. При удалении потеряются его данные.
- Данные пользователя после обновления. Документы, персональные данные, настройки должны открываться и работать без изменений.