

37. ИНСТРУМЕНТЫ ВЕБ-ДИЗАЙНА ДЛЯ АДАПТАЦИИ ПОД МОБИЛЬНЫЕ УСТРОЙСТВА

Маркевич М.А., Судникович П.В., студентки гр. 173602, Скриган В.А., магистрант гр. 176541

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Ефремов А.А. – канд. эк. наук, доцент кафедры ЭИ

Аннотация. Рассмотрены адаптивный и отзывчивый дизайны сайтов. Предоставлены рекомендации по разработке мобильного веб-дизайна. Рассмотрен язык Swift. Изложены отличия разработки на операционных системах iOS и Android. Приведены списки отсутствующих компонентов веб-дизайна в зависимости от операционной системы.

Ключевые слова: веб-дизайн, мобильные устройства, адаптивный дизайн, разработка.

В 2019 году 80% пользователей использовали мобильные устройства для поиска в Интернете. То, как сайт отображается на мобильных устройствах, может укрепить или разрушить бренд. Одно из важнейших решений при разработке для мобильных устройств – выбрать адаптивный или отзывчивый дизайн. Отзывчивый дизайн динамически адаптируется под размер экрана любого устройства. Макет использует медиа-запросы CSS для изменения размера таких аспектов веб-страницы, как ее ширина и высота. Адаптивный дизайн удобен для мобильных устройств, но требует больше ресурсов и использует другой подход. Он отображает статические макеты для экранов различных размеров. Дизайнеру потребуется разработать макеты адаптивного сайта для шести экранов шириной: 320, 480, 760, 960, 1200 и 1600 пикселей [1].

Рекомендации по разработке мобильного веб-дизайна [2]:

- Исследовать пути перемещения пользователей. У мобильных устройств нет такой же площади экрана, как у настольных компьютеров и планшетов. Это означает, что нужно расставить приоритеты для элементов, которые будут отображаться для мобильных пользователей.

- Отказаться от полноценной панели навигации. Панели навигации позволяют пользователям переходить к нужному разделу сайта. Обычно они расположены в верхней или боковой части сайта, где пользователи могут их четко видеть. Однако внедрение панелей навигации для мобильных устройств – проблема. Меньшие размеры экрана означают, что полноценная панель навигации займет слишком много места и будет неэффективна.

- Ограничить количество вариантов действий на одной странице. Слишком большое количество вариантов может сбить с толку и затруднить навигацию по сайту.

- Упростить элементы и действия. Элемент, что не служит цели или не является необходимым, должен быть удален. Также не рекомендуется использование длительной, сложной анимации без прямой необходимости.

- Использовать изображения и видео. Добавление изображений на страницу сайта – отличный способ привлечь внимание и повысить вовлеченность. Изображения должны соответствовать предложению и иметь правильный размер, подходящий для дизайна сайта.

- Размещать элементы ближе к центру. Важно, чтобы навигационные и интерактивные области были расположены ближе к центру. Обязательно протестировать места размещения, независимо от выбора дизайна отзывчивого или адаптивного.

- Сделать контактную информацию ссылками. При работе с мобильным устройством на выполнение задачи должно уходить минимальное количество времени. Номера телефонов должны быть ссылками, чтобы при нажатии автоматически открывалось приложение для сотовой связи с набранным набором цифр. Таким образом, посетителям не придется переключаться с сайта на мобильное приложение и обратно вручную. То же самое касается и имен контактных лиц, почт и тд. При нажатии должно открываться соответствующее приложение.

Swift – это мощный и простой язык программирования, представленный Apple в 2014 году, с помощью него можно создать приложения для iOS, MacOS, Apple TV и Apple Watch. Данным языком были унаследованы идеи от Objective-C, Rust, Haskell, Ruby, Python, C# и CLU. Язык использует паттерны безопасного программирования и содержит новые функции, что совершенствует процесс разработки. Созданный Крисом Латтнером, язык получился надежным, быстрым и открытым. Данный язык программирования используется для написания приложений для телефонов, умных

часов и компьютеров компании Apple. Это универсальный язык, который можно использовать на любой платформе от компании из Купертино [3].

Операционная система Android все также занимает большую часть рынка, что автоматически делает разработчиков, которые создают для нее приложения, востребованными и высокооплачиваемыми специалистами. Начинающих программистов всегда мучает вопрос «какой язык программирования учить, чтобы создавать приложения для Android?» Порядка 90% приложений, представленных на Play Market, разработаны на языке программирования Java. Это абсолютно оправдано, т.к. этот язык долгое время был основным для ОС Android и в целом используется повсеместно для создания различных продуктов. Еще несколько лет назад он был на первом месте по популярности и востребованности использования среди остальных. Альтернативные языки для работы с приложениями Android: Kotlin – это второй базовый язык программирования для создания приложений Android; Python; C/C++ – эти языки отлично подойдут для грузных приложений в любой среде, в том числе и Android; JavaScript – подходит для создания полного жизненного цикла мобильного приложения, но при условии отличного владения React Native и использования его в работе; Dart – альтернатива JavaScript; Lua[4].

Существуют отличия дизайна мобильного приложения на iOS и Android. Адаптация дизайна под другую платформу – неотъемлемая часть работы дизайнера мобильных приложений. Цель этой работы – органично вписать в дизайн паттерны взаимодействия пользователя. К тому же проработанная адаптация упрощает разработку за счет использования нативных компонентов платформы.

Чтобы адаптировать дизайн правильно, нужно соблюдать гайдлайны платформ: Human Interface Guidelines (HIG) у iOS и Material Design у Android.

Базовые отличия:

- единицы измерения: iOS – pt, Android – dp;
- системный шрифт: iOS – San Francisco, Android – Roboto;
- способы верхнеуровневой навигации: iOS – Tab bar, Android – Navigation Drawer, Bottom Navigation Bar и Tabs;
- особый паттерн вызова Navigation Drawer, жест edge swipe слева вправо: iOS – переход с дочерней на родительскую страницу, Android – открытие навигационной шторки.

Отличия в компонентах (UI).

В iOS нет компонентов:

- Navigation Drawer. iOS не реализует бургер-меню. В iOS верхнеуровневая навигация только по Tab Bar.
- Banner. Не нативный компонент для iOS. С помощью Banner пользователю сообщается важная информация и предлагаются связанные с ней действия.
- Snackbar. Не нативный компонент для iOS. Snackbar применяется, чтобы донести до пользователя короткое сообщение о результате его действия.
- Chips. Не нативный компонент для iOS. Используются для ввода информации, описания и действия.
- FAB. Не нативный компонент для iOS. FAB – это кнопка для совершения основного действия на экране. Например, в почтовом приложении FAB будет создавать новое письмо.
- Expanding Bottom Sheet. Не нативный компонент для iOS. Expanding Bottom Sheet – это поверхность, которая прикреплена к низу страницы. При нажатии поверхность расширяется до полноценной страницы.

В Android нет компонентов:

- Page Controls. Page Control показывает, на какой из страниц находится пользователь. Его нет среди нативных компонентов Android.
- Steppers. Steppers – стандартный control iOS, не описан в Material. Используется для ввода небольших значений. Пример: количество копий при печати.
- Popover. Popover – всплывающая панель, которая в основном используется на iPad.

Кроме адаптивных и отзывчивых дизайнов, адаптацию под мобильные устройства можно осуществить с помощью разработки мобильной версии сайта. Мобильная версия сайта – это сайт, открывающийся при посещении основного ресурса с мобильного телефона или планшета. Адрес мобильного сайта обычно начинается с «m».

В целях повышения эффективности разработки дизайна для веб-приложений на мобильных устройствах и повышения конкурентоспособности этих веб-приложений в интернет-пространстве была изучена тема «Инструменты веб-дизайна для адаптации под мобильные устройства». Для поставщиков интернет-сервисов данное исследование способно помочь в увеличении частоты использования их предложений за счет увеличения пользователей на мобильных платформах.

Список использованных источников:

1. Новая адаптивность: веб-дизайн в мире компонентов [Электронный ресурс]. –<https://web.dev/i18n/ru/>

59-я Научная Конференция Аспирантов, Магистрантов и Студентов БГУИР, Минск, 2023

2. Интернет технологии.ру [Электронный ресурс]. – <https://www.internet-technologies.ru/articles/adaptivnyy-ili-otzyvchivyy-7-luchshih-praktik-mobilnogo-veb-dizayna.html>
3. Swift. Язык программирования [Электронный ресурс]. – <https://www.apple.com/ru/swift/>
4. ItProger [Электронный ресурс]. – <https://itproger.com/news/>