

В.В. Молотихин, А.А. Лазарев

Научный руководитель: канд. техн. наук, доц. Е.П. Догадина
Муромский институт Владимирского государственного университета
602264, г. Муром Владимирской обл., ул. Орловская, д.23
E-mail: kaf-eivt@yandex.ru

Основные концепции разработки приложения для Windows Phone

В данной статье рассматриваются базовые принципы разработки приложений для Windows Phone. Описывается создание приложений, состоящих из нескольких страниц, и навигация между страницами. Рассматривается поддержка работы приложения в различных ориентациях экрана, при изменении тем и акцентных цветах. Также описывается работа с экранной клавиатурой, системным треем и панелью приложения.

Все телефоны на операционной системой Windows Phone имеет аппаратную кнопку Назад, которая позволяет перемещаться между страницами независимо от того, какому приложению принадлежит конкретная страница. Это напоминает работу Web-браузера. Навигация между страницами осуществляется при помощи класса `NavigationService`. Кроме прямого вызова методов класса `NavigationService`, навигация может осуществляться с помощью гиперссылок на страницах приложения. Гиперссылки представляют собой объекты класса `HyperlinkButton`. Кроме использования элемента управления `HyperlinkButton` осуществляется возможность перехода на другую страницу с помощью кода на языке C#.

Возможность смены ориентации экрана — одна из базовых функций операционной системы. Однако разработчик может сам решать, в каких ориентациях экрана будет работать его приложение. Ориентация экрана, когда его высота больше ширины, называется портретной (`Portrait`). В портретной ориентации работает стартовая страница телефона и запускается эмулятор. Ориентация экрана, когда ширина больше высоты, называется ландшафтной (`Landscape`). В такой ориентации экрана работает большинство игр. В XAML-коде каждой страницы у элемента `PhoneApplicationPage` присутствуют два свойства: `SupportedOrientations="Portrait"` и `Orientation="Portrait"`. Первое свойство определяет поддерживаемые ориентации, второе задает ориентацию по умолчанию.

Операционная система Windows Phone предоставляет возможность выбора между светлой (`Light`) и темной (`Dark`) темами. И акцентным цветом выделены слова "accent color". Также акцентный цвет может использоваться и в ваших приложениях. Если требуется выделить что-то цветом, то пусть цветом выделения будет системный акцентный цвет. При создании приложений для Windows Phone часто хочется изменить цвет какого-либо элемента. И платформа не накладывает на это никаких ограничений. Но при смене цвета необходимо иметь понятие о том, как будет выглядеть интерфейс в другой теме и при другом акцентном цвете. Например, элемент, нарисованный белым, будет хорошо отображаться в темной теме, но если у пользователя установлена светлая тема, то такой элемент вообще не будет виден. Предпочтительнее не задавать значения цветов напрямую, а использовать цвета системной темы. Иногда в процессе работы приложения требуется узнать, какая тема выбрана на телефоне — темная или светлая. Это можно сделать, проверив значение одного из ресурсов — `PhoneDarkThemeVisibility` или `PhoneLightThemeVisibility`. Первый ресурс возвращает значение видимости темной темы, а второй — светлой.

Для использования экранной клавиатуры на Windows Phone, не требуется никаких действий. Если в приложении присутствует элемент ввода данных, то клавиатура автоматически появится при нажатии пользователем на этот элемент, например, на текстовое поле. Но кроме этого, можно заставить экранную клавиатуру отображать только те символы, которые нужны в конкретном контексте. Например, при вводе номера телефона или стоимости покупки это будут цифры, а при вводе e-mail-адреса пользователь должен иметь возможность легко вставить символ @. В интерфейсе же чата часто требуется вставить смайл. За то, какая именно клавиатура будет отображаться, отвечает контекст ввода (`inputscope`). Каждое текстовое поле (`TextBox`) и поле ввода пароля (`PasswordBox`) позволяет настроить для себя контекст ввода. Такая настройка не обязательна, но пользователю будет удобнее работать с подходящей в каждом конкретном случае клавиатурой.

Во встроенном графическом дизайнера Visual Studio, а также при отладке приложений в верхней

части экрана можно увидеть системную информацию телефона, такую как уровень сигнала, заряд батареи и т. д. Эта информация важна для пользователя. Область, где она отображается, называется системным треем (System Tray). Его можно скрыть. По умолчанию системный трей показывается на всех страницах. За это отвечает свойство `shell:SystemTray.IsVisible="True"`. Чтобы скрыть системный трей, присвойте данному свойству значение `False`. В этом случае приложению будет доступна на 32 пиксела большая область экрана.

Панель приложения (Application Bar) — это элемент управления, который размещается в нижней части экрана телефона и содержит несколько круглых кнопок-значков. Максимальное количество кнопок равно четырем. Больше разместить на панели приложения не получится. Кроме того, в правом верхнем углу панели приложения отображаются три точки, нажав на которые можно раскрыть панель. В этом случае появятся подписи к кнопкам и текстовые пункты меню (в случае их наличия). Кроме того, панель приложения может отображаться в свернутом состоянии, в котором отображаются только три точки, раскрывающие панель. Несмотря на то, что название панели содержит слово Application, эта панель задается индивидуально для каждой страницы. Поэтому на каждой странице возможно реализовать свою уникальную панель приложения, а на каких-то страницах они и вовсе могут отсутствовать. За круглые кнопки-значки отвечает элемент управления `ApplicationBarIconButton`. Значки для панели приложения можно нарисовать самостоятельно либо их можно найти в Интернете.

Литература

1. Разработка приложений для Windows Phone 7.5 / С. В. Пугачев, С. И. Павлов, Д. В. Сошников. — СПб.: БХВ-Петербург, 2012. — 384 с.
2. [<http://www.samomudr.ru/>] Разработка приложений для Windows Phone 7. Программирование мобильных приложений и игр. Мобильная разработка и мобильные технологии Microsoft