

ASP.NET

モバイル Web アプリケーション再入門

デベロッパー&プラットフォーム推進部

エバンジェリスト

物江 修

アジェンダ

- ASP.NET モバイルアプリケーションについて
- モバイル Web プロジェクトの作成方法
- モバイル Web フォームとコントロールの特徴
- 標準のモバイル Web コントロールで行えない機能の実現
- 開発に必要な情報の入手と反映
- モバイル Web アプリケーションのセッション管理について
- まとめ

ASP.NET モバイルアプリケーションについて

ASP.NET モバイル Web アプリケーション の目的

- 多種多様なモバイルデバイスの仕樣的な違いを吸収し、開発者に共通した開発環境を提供するもの
 - 携帯電話やPDAと言ったクライアントのモバイルデバイスに左右されない、単一のアプリケーション開発を行なうことができる

モバイルデバイスの仕様の違い

- マークアップ言語の違い

例) cHTML, WML, XHTML, HTML など

- ブラウザの性能の違い

例) 色、画像の表示、リダイレクト、cookie, SSL の使用可、不可等

ASP.NET モバイル Web アプリケーション がサポートしているマークアップ言語

- **HTML**
 - PC や PDA など
- **cHTML**
 - NTT DoCoMo
- **WML 1.1, 1.2**
 - Open wave (国外)
- **XHTML**
 - SOFT BANK (旧 J-Phone, vodafone), 最近の au (WAP 2.0)

注) 古い au や *TUKA* が採用している HDML はサポートしてない

モバイル Web プロジェクト の作成方法

モバイル Web プロジェクトの作成

-作成方法は Visual Studio のバージョンごとに異なる-

- Visual Studio.NET (2002)

Microsoft の Web サイトより Mobile Internet Toolkit ダウンロードしてインストール

- Visual Studio.NET 2003

プロジェクト新規作成の際に「ASP.NET モバイル Web アプリケーション」を選択する

- Visual Studio 2005

ASP.NET と同じく “Web サイト” を作成し、[既存項目の追加]メニューで モバイル Web アプリケーション関連のファイル類を追加していく

Visual Studio.NET (2002) でのモバイル Web アプリケーションの開発

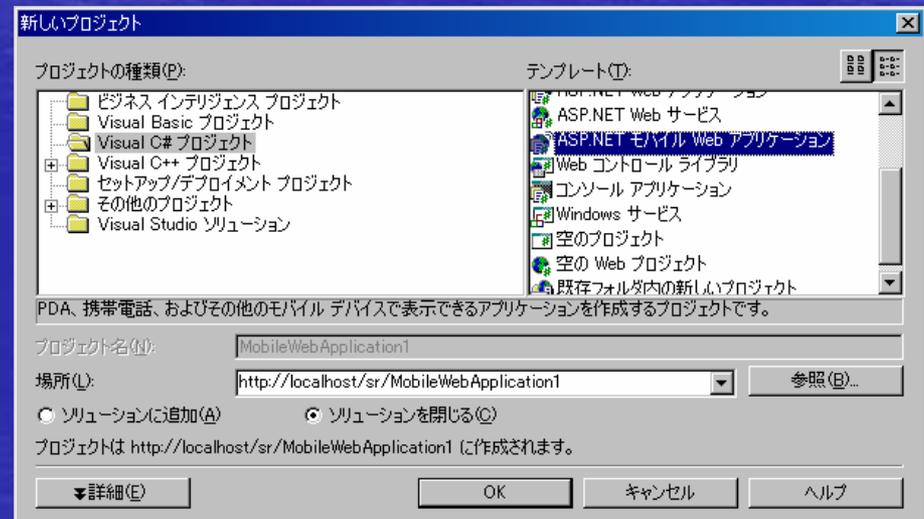
1. Microsoft の Web サイトより Mobile Internet Toolkit をダウンロードしてインストール

<http://www.microsoft.com/japan/msdn/vstudio/device/mitdownload.aspx>

2. Visual Studio.NET のメニュー [ファイル]-[新規作成] を選択
3. [新しいプロジェクト] ダイアログボックス内 [テンプレート] リストから「ASP.NET モバイル Web アプリケーション」を選択

Visual Studio.NET 2003 でのモバイル Web アプリケーションの開発

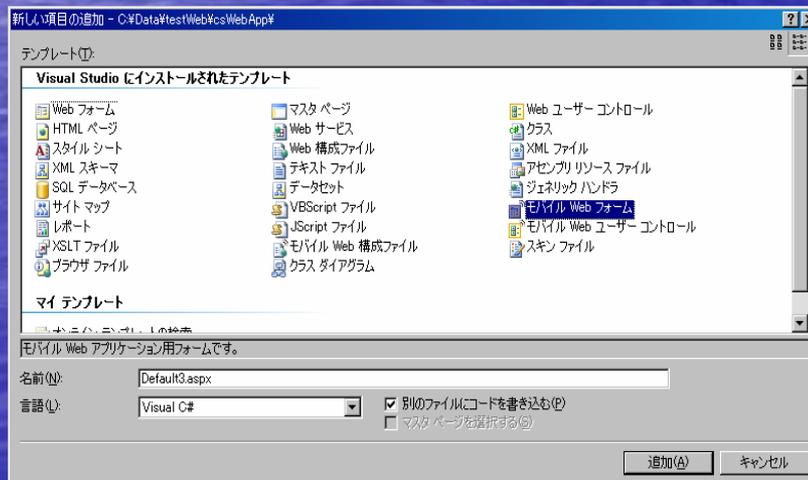
1. Visual Studio.NET のメニュー [ファイル]-[新規作成] を選択
2. [新しいプロジェクト] ダイアログボックス内 [テンプレート] リストから「ASP.NET モバイル Web アプリケーション」を選択



Visual Studio 2005でのモバイル Web アプリケーションの開発

Visual Studio 2005 はプロジェクトテンプレートが用意されていないので、[新しい Web サイト] 作成後、ソリューションエクスプローラーで [新しい項目の追加] を使用して以下の項目を追加する必要がある

1. モバイル Web フォーム
2. モバイル Web 構成ファイル



モバイル Web フォームと コントロールの特徴

モバイル Web フォームの特徴

- 1ページに複数の画面が持てる
- モバイルデバイスごとに自動的にマークアップ言語が出力される
- モバイルデバイスごとに動作を変更させることができる
- モバイル Web フォームには モバイル Web フォームコントロールしか使用できない
- HTML ソースにモバイル Web コントロールにないタグを記述しても出力されない

デモ：ASP.NET モバイル Web アプリケーションの動作

- ASP.NET モバイル Web アプリケーションのページ遷移
- 出力されるマークアップ言語の違い
- モバイルデバイスごとに異なる動作

モバイル Web コントロール

- モバイル Web コントロールの種類
 - コンテナ
 - 文字列を表示
 - 列挙(リストとセレクション)
 - ページ遷移
 - イメージの表示
 - カレンダー
 - バリデーション
 - ユーザーコントロール
 - カスタムコントロール

モバイル Web フォーム のコントロール :
コンテナ



コンテナコントロール

- フォームコントロール

- 個別にリンク可能な分割されたコントロール用のコンテナ
- 一ページに複数のフォーム
- 任意のデバイスに対応した複数の“スクリーン”
- 一度の接続に対し一つの画面がアクティブ(既定では最初のフォームがアクティブ)

- パネルコントロール

- 多数のコントロールをグループ化する機能を提供
- 複数のコントロールに対し、表示非表示、無効有効が一度の操作で設定可能
- スタイルを適用するとパネル内の各コントロールに継承される
- 動的に生成されたプレースフォルダ

モバイル Web フォーム のコントロール :
テキストの表示

MobileControl

Ad Rotator

Calendar

Image

PagedControl

Panel

Selection List

StyleSheet

TextControl

TextView

ValidationSummary

List

ObjectList

Form

BaseValidator

PhoneCall

Command

Label

Link

TextBox

CompareValidator

CustomValidator

RangeValidator

RegularExpression
Validator

RequiredField
Validator

文字列を表示するコントロール

- Label コントロール
 - 少量の内容
 - 読み取り専用
- TextBox コントロール
 - シングルラインの入力用テキストボックス
- TextView コントロール
 - Mobile Internet Tool専用
 - テキストの大きなフィールド
 - 少量のマークアップのサポート:ボールド, <i>イタリック,
改行, <p>段落, <a>アンカータグ
 - ページ分割のサポート

MobileControl

Ad Rotator

Calendar

Image

PagedControl

Panel

Selection List

StyleSheet

TextControl

TextView

ValidationSummary

モバイル Web フォーム のコントロール :
リスト と セレクション

List

ObjectList

Form

BaseValidator

Call

Command

Label

Link

TextBox

CompareValidator

CustomValidator

RangeValidator

RegularExpression
Validator

RequiredField
Validator

リスト と セレクション リスト

	リスト	セレクションリスト	オブジェクトリスト
データ連結	任意	任意	必須
ページ分割	可	不可	可
装飾	なし, 中黒, 数字	ドロップダウンリスト, リストボックス, ラジオボタン, チェックボックス, マルチセレクト	カスタマイゼーションでのみ
対話式	任意	可	任意
ASP.NET コントロールで同様の機能を提供するもの	データリスト	リストボックス, チェックボックス, チェックボックスリスト, ラジオボタン, ラジオボタンリスト, とドロップダウンリスト	データグリッド

MobileControl

Ad Rotator

Calendar

Image

PagedControl

Panel

Selection List

StyleSheet

TextControl

TextView

ValidationSummary

モバイル Web フォーム のコントロール :
遷移コントロール

List

ObjectList

Form

BaseValidator

PhoneCall

Command

Label

Link

TextBox

CompareValidator

CustomValidator

RangeValidator

RegularExpression
Validator

RequiredField
Validator

遷移コントロール

- Link コントロール
 - モバイルのページ、任意のURLあるいは別のフォームへのテキストベースのハイパーリンク
 - Softkey プロパティ
 - ASP.NET のHyperLink コントロールと同様

```
<mobile:Link id="lnkfrmResults" runat="server"
NavigateUrl="#frmResults">Go to Results</mobile:Link>
```

- PhoneCall コントロール
 - Mobile Internet Tool 独自
 - 電話をかけるか電話番号を表示するためのマークアップを自動的に生成

遷移コントロール Command コントロール

- ASP.NET の Button コントロールと同様の機能を提供
- UI から ASP.NET イベントを取得

```
<mobile:Command id="cmdSelectProduct" runat="server" softkeylabel="Next">Select Product</mobile:Command>
```

- SoftkeyLabelプロパティ
 - 携帯電話操作支援のためのソフトキーテキストを指定

モバイル Web フォーム のコントロール：
イメージの表示

MobileControl

Ad Rotator

Calendar

Image

PagedControl

Panel

Selection List

StyleSheet

TextControl

TextView

ValidationSummary

List

ObjectList

Form

BaseValidator

PhoneCall

Command

Label

Link

TextBox

CompareValidator

CustomValidator

RangeValidator

RegularExpression
Validator

RequiredField
Validator

イメージ表示コントロール

- **Image コントロール**

- 画像の表示
- ImageUrl プロパティで画像ファイルを指定
指定可能なファイル形式
*.gif, *.jpg, *.jpeg, *.bmp, *.wmf, *.png, *.wbmp

- **AdRotator コントロール**

- 広告見出しを表示する
- AdvertisementFile プロパティに XML 形式の広告ファイルを指定する

MobileControl

Ad Rotator

Calendar

Image

PagedControl

Panel

Selection List

StyleSheet

TextControl

TextView

ValidationSummary

List

ObjectList

Form

BaseValidator

PhoneCall

Command

Label

Link

TextBox

CompareValidator

CustomValidator

RangeValidator

RegularExpression
Validator

RequiredField
Validator

モバイル Web フォーム のコントロール :
カレンダー

カレンダーコントロール

- ユーザーに日付の選択を行なわせる
- 使用するデバイスによってインターースが大きく変わる

SelectedDate プロパティ
選択された日を指定する

VisibleDate プロパティ
表示される月を指定する

FirstDayOfWeek プロパティ
週の最初の日を指定する

SelectionMode プロパティ
日付の選択の範囲や選択方法を指定する

MobileControl

Ad Rotator

Calendar

Image

PagedControl

Panel

Selection List

StyleSheet

TextControl

TextView

ValidationSummary

List

ObjectList

Form

BaseValidator

PhoneCall

Command

Label

Link

TextBox

CompareValidator

CustomValidator

RangeValidator

RegularExpression
Validator

RequiredField
Validator

モバイル Web フォーム のコントロール :
検証コントロール

検証コントロール

- 同じフォーム上の SelectList および TextBox コントロールのデータを検証する

検証コントロール	使用目的
CompareValidator	値と比較
CustomValidator	独自の検証を定義
RangeValidator	範囲を検証
RegularExpressionValidator	パターンと一致するかどうか検証
RequiredFieldValidator	値を入力したかどうか検証
ValidationSummary	エラーの一覧を表示します

モバイル Web コントロールの詳細について

- .NET Framework 開発者ガイド -『ASP.NET モバイル コントロールを使用した作業の開始』
 - <http://msdn.microsoft.com/library/ja/default.asp?url=/library/ja/mw sdk/html/mworiControlReference.asp>

標準のモバイル Web コントロールで 行えない機能の実現

モバイル Web コントロールの作成

- 既定で用意されていないモバイル Web コントロールは以下の方法で作成することが可能
 - ユーザーコントロール
 - *.ascx ファイルに既存のコントロールを含めていく
 - アプリケーションの共通ユーザーインターフェースの再利用
 - カスタムコントロール
 - アプリケーション、あるいはクラスライブラリにプログラムコードで作成する
 - 継承されたコントロール
 - 既存のモバイルWebコントロールを拡張
 - グループ化した複数のモバイルコントロール
 - コントロールの新規作成
 - アセンブリに新規のモバイルコントロールを作成
 - コントロールの為のアダプタを要求

モバイル Web ユーザーコントロールの作成

- Visual Studio 2005 のメニュー [Web サイト] - [新しい項目の追加] を使用して “モバイル Web ユーザーコントロール” を追加する
- 追加された *.ascx に既存のコントロールを配置する
- 複合コントロールの作成が簡単
- 管理が簡単
- 実行速度はカスタムコントロールには劣る
- Web フォームデザイナー上では配置されたコントロールは描画されない
- 使用する際は *.aspx ファイル内に宣言が必要 (VisualStudio を使用すると自動的に宣言が行なわれる)

例)

```
<%@ Register Src="WebUserControl.ascx"  
    TagName="WebUserControl" TagPrefix="uc1" %>
```

モバイル Web カスタムコントロールの作成

- クラスライブラリ用のプロジェクトを作成し、`System.Web.UI.MobileControls.MobileControl` クラスか、基となるモバイル Web コントロールクラスを継承させる
- `MobileControl` クラスはデザイン時の機能をサポートしておらず、そのためデザイナのツールボックスへの追加は行えない
(※) 既存のモバイル Web コントロールクラスを継承した場合は問題ない

モバイル Web カスタムコントロールの使用

- MobileControl クラスを継承元としたカスタムコントロールは [ツールボックス] に登録できないため以下のいずれかの方法を使用する
 - ユーザーコード内でインスタンスを生成する
(※記述例は後述)
 - Web フォームに HTML ソースにタグを記述する
(※記述例は後述)

モバイル Web カスタムコントロールの使用

- ユーザーコード内でインスタンスを生成する

例:C#) 名前空間が UserControl で クラスが ComplexCtrl である場合

```
UserCtrl.ComplexCtrl nwCtrl = new UserControl.ComplexCtrl();  
nwCtrl.ID = "nwCtrl";  
this.Form1.Controls.Add(nwCtrl);
```

モバイル Web カスタムコントロールの使用

- Web フォームに HTML ソースにタグを記述する

例) アセンブリ名が UserControl でかつ名前空間が UserControl でクラスが ComplexCtrl である場合

宣言

```
<%@ Register Assembly="UserCtrl" Namespace="UserCtrl"  
TagPrefix="cc1" %>
```

コントロールのタグ

```
<cc1:ComplexCtrl ID="ComplexCtrl1" runat="server" />
```

デモ：モバイル Web コントロールの作成

- ユーザーコントロールの作成
- カスタムコントロールの作成

モバイル Web コントロールへのカスタム属性の追加について

- モバイルブラウザ用マークアップ言語には、タグに独自の属性を追加することで機能を拡張しているものがあり、その属性を追加したいというご要望は多い

例)

- `<input>` タグへの `istyle` 属性の追加 (NTT DoCoMo 端末にて入力文字種の指定が可能)
- `<A>` タグへの `utn` 属性の追加 (NTT DoCoMo 端末にて携帯電話の固体識別番号の取得が可能)

モバイル Web フォームは *.aspx の HTML ソースの記述をそのままには出力しないので、いずれも独自にデバイスアダプタを作成しないと実現できない

デバイスアダプタの作成 (1/2)

- 以下の情報をもとにデバイスアダプタを作成

.NET Framework 開発者ガイド『ウォークスルー：デバイスのサポートの追加』

<http://msdn.microsoft.com/library/ja/default.asp?url=/library/ja/mw/sdk/html/mwconwalkthroughaddingsupportfordevices.asp>

- 以下の情報をもとにデバイスアダプタを追加

.NET Framework 開発者ガイド『アダプタのマッピング』

<http://msdn.microsoft.com/library/ja/default.asp?url=/library/ja/mw/sdk/html/mwconadaptermappings.asp>

デバイスアダプタの作成 (2/2)

- 参考となるデバイスアダプタのソースコード
 - ASP.NET Mobile Controls XHTML Adapter Source

<http://www.microsoft.com/downloads/details.aspx?FamilyID=9642963f-7c8d-4b26-a440-400e03ced4cb&DisplayLang=en>
- HTML,cHTML,WAP のデバイスアダプタのコード

デモ：デバイスアダプタの作成と実装

- cHTML デバイスアダプタによる iStyle 属性の追加
- XHTML デバイスアダプタによる format 属性の追加

モバイル Web デバイスの機種情報 の入手と反映

モバイルデバイスの判断の仕組み

- ブラウザの User-Agent 情報をキーとして、
machine.config ファイル、*. Browser ファイル
(ASP.NET 2.0 の場合) の内容を検索

ASP.NET 1.1

machine.config ファイルの
<browserCaps> 要素にデバイス情報を追加する

※この方法は ASP.NET 2.0 では使用していないがサポートはされている。

よって ASP.NET でも有効に動作する

ASP.NET 2.0

C:\Windows\Microsoft.NET\Framework
k\v2.0.50727\CONFIG\Browsers
下にインストールされている *.browser
ファイルにデバイス情報を追加する

ASP.NET 1.1 での機種情報の追加

- web.config、あるいは machine.config に <browserCaps> 要素を使用して機種情報を追加

machine.config に機種情報を追加した場合はワーカークロセスの再起動が必要

記述例)

```
<browserCaps>
  <result type="System.Web.Mobile.MobileCapabilities, System.Web.Mobile,
Version=2.0.0.0, Culture=neutral, PublicKeyToken=b03f5f7f11d50a3a"/>
  <use var="HTTP_USER_AGENT"/>
  <filter>
    <case match="Vemulator/.*">
  <case
match="Vemulator/(?'majorVersion'¥d+)(?'minorVersion'¥.¥d+)/(?'deviceID'¥S*)">
  browser="vodafone"
  version="{majorVersion}{minorVersion}"
  majorver="{majorVersion}"
  minorver="{minorVersion}"
  preferredRenderingType = "xhtml-basic"
  preferredImageMime = "image/jpeg"
  preferredRenderingMime = "text/html"
  javascript = "false"
  supportsAccesskeyAttribute = "true"
  <case match="V802N" with="{deviceID}">
    screenCharactersHeight = "16"
    screenCharactersWidth = "20"
    screenPixelsHeight = "240"
    screenPixelsWidth = "320"
  </case>
  <case match="V703SH" with="{deviceID}">
    screenCharactersHeight = "8"
    screenCharactersWidth = "9"
    screenPixelsHeight = "240"
    screenPixelsWidth = "320"
  </case>
  </case>
  </case>
  </filter>
</browserCaps>
```

ASP.NET 2.0 での機種情報の追加

- 以下の 3 つの方法がある
 1. アプリケーションフォルダ内に App_Browsers フォルダを手動で作成し、その中に機種情報を記述した *.browser ファイルを配置する
 2. フォルダ %v2.0.50727%\CONFIG%Browsers 下に機種情報を記述した *.browser ファイルを配置し、aspnet_regbrowsers -i コマンドを実行する
 3. フォルダ %v2.0.50727%\CONFIG%Browsers 下の *.browser ファイルに機種情報を追記し、aspnet_regbrowsers -i コマンドを実行する
- ※ただし、今後ブラウザ情報が変更される可能性があるためあまり推奨できない

*.browser ファイルの記述例

```
<browsers>
<browser id="Vodafone" parentID="Up">
  <identification>
    <userAgent match="(UP¥.Browser)|(UP/)" />
    <userAgent nonMatch="Go¥.Web" />
  </identification>

  <capture>
    <userAgent
match="Vemulator/(?'majorVersion'¥d+)(?'minorVersion'¥.¥d+)/(?'deviceID'¥S*)" />
    </capture>

  <capabilities>
    <capability name="browser" value="vodafone" />
    <capability name="canInitiateVoiceCall" value="true" />
    <capability name="canSendMail" value="false" />
    <capability name="deviceID" value="{deviceID}" />
    <capability name="deviceVersion" value="{deviceVersion}" />
    <capability name="isMobileDevice" value="true" />
    <capability name="majorVersion" value="{browserMajorVersion}" />
    <capability name="maximumRenderedPageSize" value="1492" />
    <capability name="minorVersion" value="{browserMinorVersion}" />
    <capability name="numberOfSoftkeys" value="2" />
    <capability name="optimumPageWeight" value="700" />
    <capability name="preferredImageMime" value="image/jpeg" />
    <capability name="preferredRenderingMime" value="text/html" />
    <capability name="preferredRenderingType" value="xhtml-basic" />
    <capability name="requiresAdaptiveErrorReporting" value="true" />
    <capability name="rendersBreakBeforeWmlSelectAndInput" value="true" />
    <capability name="rendersWmlDoAcceptsInline" value="false" />
    <capability name="rendersWmlSelectsAsMenuCards" value="true" />
    <capability name="requiresFullyQualifiedRedirectUrl" value="true" />
    <capability name="requiresNoescapedPostUrl" value="true" />
    <capability name="requiresPostRedirectionHandling" value="true" />
    <capability name="supportsRedirectWithCookie" value="false" />
    <capability name="type" value="Phone.com${browserMajorVersion}" />
    <capability
name="version" value="{browserMajorVersion}{browserMinorVersion}" />
  </capabilities>
</browser>
```

(※)表示の関係で折り返しています

```
<browser id="V703SH" parentID="Vodafone">
  <identification>
    <capability name="deviceID" match="V703SH" />
  </identification>
  <capabilities>
    <capability name="mobileDeviceManufacturer" value="Sharp" />
    <capability name="mobileDeviceModel" value="V703SH" />
    <capability name="screenCharactersHeight" value="8" />
    <capability name="screenCharactersWidth" value="9" />
    <capability name="screenPixelsHeight" value="240" />
    <capability name="screenPixelsWidth" value="320" />
  </capabilities>
</browser>

<browser id="V802N" parentID="Vodafone">
  <identification>
    <capability name="deviceID" match="V802N" />
  </identification>
  <capabilities>
    <capability name="mobileDeviceManufacturer" value="NEC" />
    <capability name="mobileDeviceModel" value="V802N" />
    <capability name="screenCharactersHeight" value="16" />
    <capability name="screenCharactersWidth" value="20" />
    <capability name="screenPixelsHeight" value="240" />
    <capability name="screenPixelsWidth" value="320" />
  </capabilities>
</browser>
</browsers>
```

機種情報追加の補足資料

- ASP.NET 1.1 向け

- browserCaps 要素 (ASP.NET 設定スキーマ)

- [http://msdn2.microsoft.com/ja-jp/library/sk9az15a\(VS.80\).aspx](http://msdn2.microsoft.com/ja-jp/library/sk9az15a(VS.80).aspx)

- ASP.NET 2.0 向け

- ブラウザ定義ファイルのスキーマ (browsers 要素)

- [http://msdn2.microsoft.com/ja-jp/library/ms228122\(VS.80\).aspx](http://msdn2.microsoft.com/ja-jp/library/ms228122(VS.80).aspx)

モバイルデバイスの機種情報について

- モバイルデバイスのブラウザ情報は、ASP.NET 1.1 まで、Microsoft の DeviceUpdate により提供されていた

ASP.NET Mobile Control Device Update 4.0 ダウンロード
http://www.microsoft.com/japan/msdn/vstudio/device/ASPNET_MCDU4_eula.aspx

テスト済みデバイス - ASP.NET Mobile Control / Microsoft Mobile Internet Toolkit –
<http://www.microsoft.com/japan/msdn/vstudio/device/mitdevices.aspx>

ASP.NET によるモバイル Web 開発のロードマップ
<http://www.microsoft.com/japan/msdn/asp.net/asp20/mobileroadmap/default.aspx>

モバイルデバイスの機種情報について

- DeviceProfilingTool が www.asp.net にて提供されている

DeviceProfilingTool

<http://www.asp.net/mobile/profile/> (ツール)

<http://www.microsoft.com/japan/msdn/vstudio/device/help.aspx> (ヘルプ)

- DaviceUpdate4 適用後、および ASP.NET 2.0 では 200 以上のモバイルデバイスのブラウザをサポートしている

メーカーサイトからの入手 (ユーザーエージェント情報)

- NTT DoCoMo

<http://www.nttdocomo.co.jp/service/imode/make/content/spec/useragent/index.html>

- Soft Bank

http://developers.softbankmobile.co.jp/dp/tech_svc/info/useragent.php

- au

http://www.au.kddi.com/ezfactory/tec/spec/4_4.html

メーカーサイトからの入手 (機種情報)

- NTT DoCoMo

- FOMA (フォーマ) 製品一覧

- <http://www.nttdocomo.co.jp/product/foma/index.html>

- Mova (ムーバ) 製品一覧

- <http://www.nttdocomo.co.jp/product/mova/index.html>

- Soft Bank

- 製品情報 (個々の製品のページを参照)

- <http://mb.softbank.jp/mb/product/3G/>

- au

- 機種別情報一覧

- http://www.au.kddi.com/ezfactory/tec/spec/new_win/ezkishu.html

メーカーサイトからの入手

(エミュレーター)

- NTT DoCoMo

- iモードHTMLシミュレータII

- <http://www.nttdocomo.co.jp/service/imode/make/content/html/tool2.html>

- Soft Bank

- ウェブコンテンツツビューア

- http://developers.softbankmobile.co.jp/dp/tool_dl/web/wcv.php

- au

- Openwave® SDK 6.2K

- <http://www.au.kddi.com/ezfactory/tool/ue/index.html>

実機での検証

- Soft Bank では『Soft Bank Solution Provider (SSP)』になることによって無料で検証センターを使用することができる
- NTT DoCoMo、au の検証センターに関する情報は見つからない
- その他、『株式会社ケータイラボラトリー』が有料で検証センターを提供している

<http://www.ktai-labo.com/>

デモ：ASP.NET モバイル Web アプリケーションへのデバイス情報の追加

- ASP.NET 1.1 アプリケーションへのデバイス情報の追加
- ASP.NET 2.0 アプリケーションへのデバイス情報の追加

ASP.NET モバイルアプリケーションでのセッション管理

- Visual Studio.NET 200x で追加される web.config は既定の状態では <sessionState> の cookieless 属性が true(セッション管理に cookie を使用しない)
- Visual Studio 2005 でモバイル Web 構成ファイルでは AutoDetect が指定されている

cookieless とは?

- ASP.NET アプリケーションは既定の状態では cookie 内に一意にキーを設けてセッション管理を行なうが、cookieless 属性を true にすると、代わりに URL に一意な識別子を付加して使用する。

黄色い部分が識別子

[http://localhost/mobiApp/\(S\(xq5xee555xaulnulo3ffjt\)\)/MobileWebForm1.aspx](http://localhost/mobiApp/(S(xq5xee555xaulnulo3ffjt))/MobileWebForm1.aspx)

ASP.NET 2.0 で追加された cookieless 属性の指定

- cookieless 属性に AutoDetect と UseDeviceProfile が指定可能
 - 要求元のブラウザが Cookie をサポートしているかどうかを判断 (UseDeviceProfile は HttpBrowserCapabilities 設定に基づいて判断) し、要求元のブラウザが Cookie をサポートしている場合は Cookie を使用してユーザーデータの保持を行なう。それ以外の場合は、クエリ文字列 (URL) に識別子が使用される。

```
<sessionState cookieless ="AutoDetect">
```

cookieless 属性を有効にした場合の問題

- URL 中に含まれる一意な文字列を識別子としてセッション情報の取り出しを行なうため、URL をコピーするとセッションが共有されてしまう
- 既定の状態では期限切れのセッション ID が指定された場合にその ID を再利用してセッション管理が開始される。そのため同じセッション ID が存在した場合はセッションが共有されてしまう

ASP.NET 2.0 では <sessionState> 要素に regenerateExpiredSessionId 属性が追加され、規定の状態では true であるためこの問題は発生しない

補足資料

- cookieless 属性の値について
sessionState 要素 (ASP.NET 設定スキーマ)
[http://msdn2.microsoft.com/ja-jp/library/h6bb9cz9\(VS.80\).aspx](http://msdn2.microsoft.com/ja-jp/library/h6bb9cz9(VS.80).aspx)
- regenerateExpiredSessionId について
SessionStateSection.RegenerateExpiredSessionId プロパティ
[http://msdn2.microsoft.com/ja-jp/library/system.web.configuration.sessionstatesection.regenerateexpiredsessionid\(VS.80\).aspx](http://msdn2.microsoft.com/ja-jp/library/system.web.configuration.sessionstatesection.regenerateexpiredsessionid(VS.80).aspx)

モバイル Web アプリケーションにおけるリダイレクト処理

- モバイルデバイスは相対 URL のリダイレクトをサポートしていなものがあるため以下の方法を使用する必要がある
 - `MobilePage.RedirectToMobilePage` メソッドを使用する
 - `<httpRuntime>` 要素の `useFullyQualifiedRedirectUrl` 属性を `True` に設定する

絶対 URL でリダイレクト処理を行なう際の 問題

- クライアントと IIS の間に何らかの処理を行なうサーバーがあり、その処理が URL へ情報を付加している場合、意図しない URL がリダイレクトされる例)
 - SSL アクセラレーター を使用した場合
 - SSL アクセラレーターから IIS に送られるリクエスト URL は “http://” なので、クライアントへリダイレクトされる URL も “https://” ではなく “http://” でリダイレクトされる
 - 何らかの変換サーバー (ポート番号) 使用している場合
 - 変換サーバーから IIS に送られるリクエスト URL に “http://hoge:8080” のようにポート番号が付加されている場合は、同様にポート番号が付加された URL がクライアントにリダイレクトされる

絶対 URL でリダイレクト処理を行なう際の 問題 (回避策)

- <httpRuntime> 要素の useFullyQualifiedRedirectUrl 属性を False に設定する
 - ただしクライアントに送られるのは絶対 URL ではなくなってしまう
- global.asax の Application_EndRequest イベントハンドラ内で、Response.RedirectLocation プロパティの内容を改竄する

```
protected void Application_EndRequest(Object sender, EventArgs e)
{
    if (Response.RedirectLocation != null)
    {
        Response.RedirectLocation =
Response.RedirectLocation.Replace("http", "https");
    }
}
```

Response.RedirectLocation プロパティはクライアントに送信されるリダイレクト用の URL が保持されている (.NET Framework 1.1 から)

参考

- ASP.NET モバイル コントロール クイック スタート
[http://msdn2.microsoft.com/ja-jp/library/c55eyak3\(VS.80\).aspx](http://msdn2.microsoft.com/ja-jp/library/c55eyak3(VS.80).aspx)
- チュートリアル：モバイル デバイス用 Web ページの作成
[http://msdn2.microsoft.com/ja-jp/library/z8h56a3f\(VS.80\).aspx](http://msdn2.microsoft.com/ja-jp/library/z8h56a3f(VS.80).aspx)
- ブラウザ定義ファイルのスキーマ (browsers 要素)
[http://msdn2.microsoft.com/ja-jp/library/ms228122\(VS.80\).aspx](http://msdn2.microsoft.com/ja-jp/library/ms228122(VS.80).aspx)
- Web コントロールとモバイル コントロールの比較
[http://msdn2.microsoft.com/ja-jp/library/yz7d0feb\(VS.80\).aspx](http://msdn2.microsoft.com/ja-jp/library/yz7d0feb(VS.80).aspx)

参考

- アダプタの種類

- [http://msdn2.microsoft.com/ja-jp/library/99tteheb\(VS.80\).aspx](http://msdn2.microsoft.com/ja-jp/library/99tteheb(VS.80).aspx)

- ASP.NET for Mobiles

- <http://www.asp.net/mobile/>

- 個人的におすすめ

- @IT - .NETでモバイル・サイト開発を始めよう

- <http://www.atmarkit.co.jp/fdotnet/aspnetmobile/index/index.html>

The image features the Microsoft logo and slogan centered on a background of a blue ocean with a bright light reflecting on the surface. The logo is in a bold, white, sans-serif font, and the slogan is in a white, italicized, sans-serif font.

Microsoft®

Your potential. Our passion.™

© 2007 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.

The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.

MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.