

**FUJITSU**

shaping tomorrow with you

 **SNAPROP** 機能説明

2011年 6月

株式会社富士通システムズ・ウエスト

# SNAPROPの機能構成

本資料では、SNAPROPが提供する以下の機能の特徴について説明します。

1 Visual Studio アドイン

2 業務画面クラス

3 UIコントロール部品

4 デバイス制御部品

5 データベース関連部品

6 通信機能

7 Utilクラス

8 個別アプリケーション

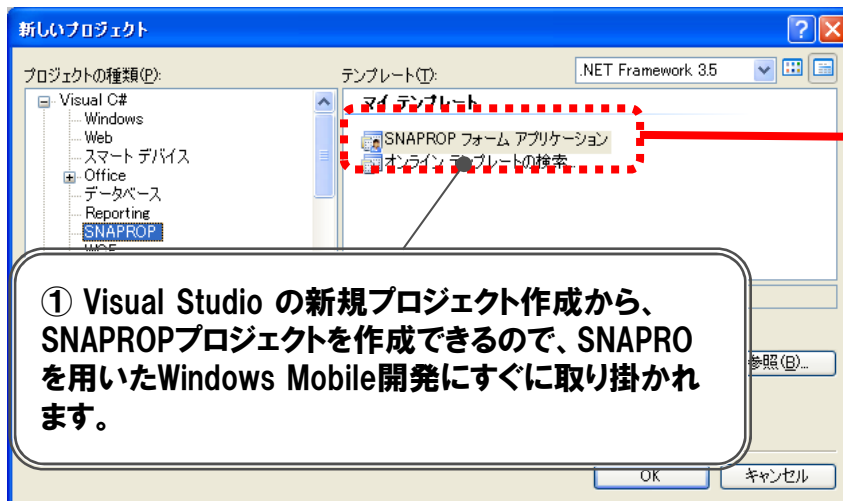


# Visual Studio アドイン

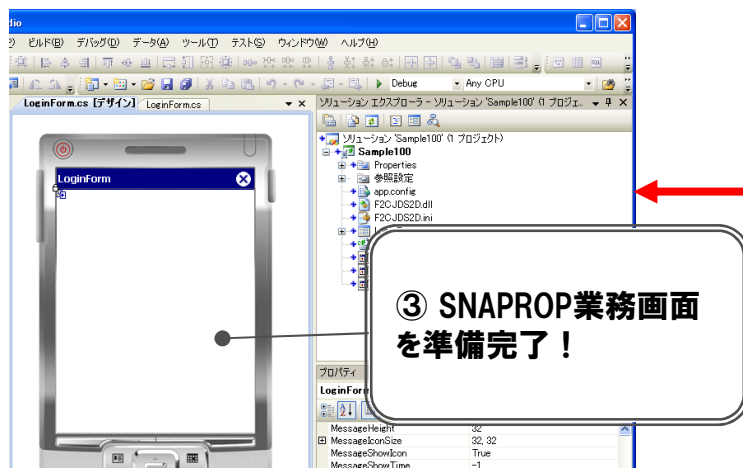
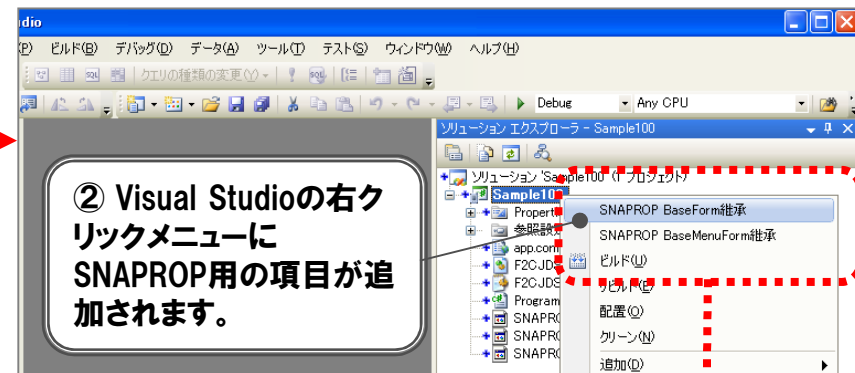
# Visual Studio の SNAPROP アドイン

Visual Studioの中にSNAPROPアドインツールが組み込まれるので、作業を止めずにプログラミングを進めることができます。

## SNAPROP プロジェクトを新規作成



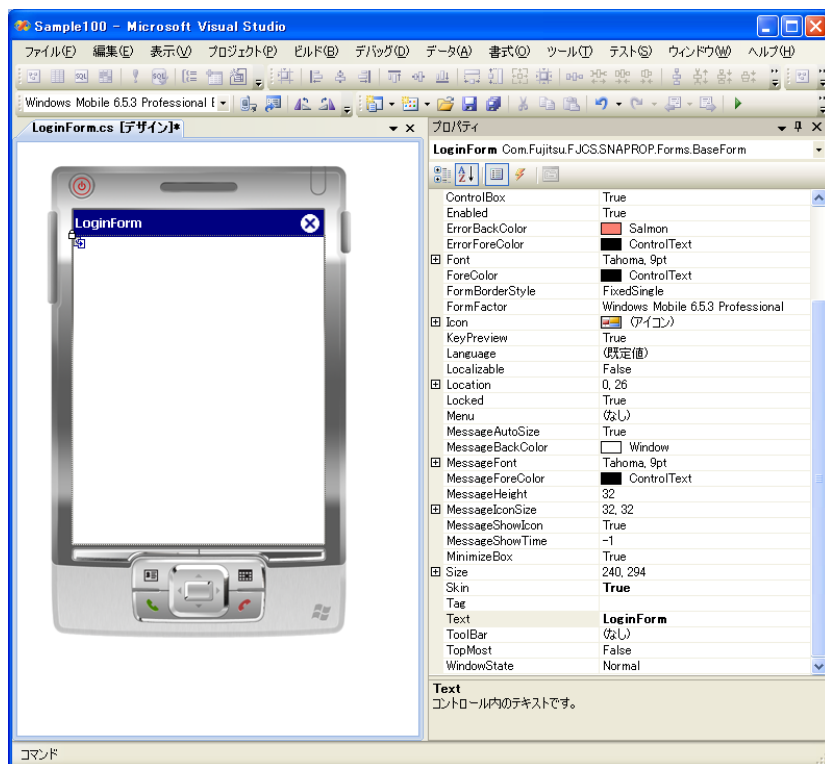
## SNAPROP 業務画面の新規作成





# 業務画面クラス

SNAPROPの業務画面クラスは、見た目は標準と同じフォーム画面ですが、Windows Mobile開発に必要な処理が埋め込まれています。



## 埋め込まれている機能

- 画面下部にメッセージを表示する機能。
- 画面の表示領域が変更になったらスクロールバーを表示する機能
- ショートカットキーによるクリックアクションを実現するための機能
- 画面に並べたコントロール間をハードキーで移動する機能
- 画面に並べたコントロールとデータ格納用クラスをマッピングする機能。(各種マップファイルとの連携機能)

# SNAPROP業務メニュークラス（1/2）

Visual StudioのSNAPROPアドインツールからメニュー用画面テンプレートを作成できます。デザインの設定や、メニュー項目を登録するだけで、メニュー機能が出来上がります。



## 出来ること

- 複数階層のメニューを簡単に実装できます。
- 色パターンによる一括デザイン設定と部位ごとの個別デザイン設定が可能です。
- ヘッダー部にタイトルを表示させたり、フッター部にボタンを並べたりが柔軟に行えます。不要な場合は、非表示に設定してメニュー領域を広げることが可能です。
- メニュー項目の登録とクリック時のイベント処理を実装できます。
- 各メニュー(行)に対しては、ショートカットキー割り当てることが可能です。

# SNAPROP業務メニュークラス (2/2)

## 色パターンを利用してデザイン設定した例



## 個別デザイン設定例



メニューリストの高さや、フォントサイズ、ショートカットキー部の表示・非表示などを個別設定できます。

## 複数階層メニュー例















# UIコントロール部品（9機能）

SNAPROPでは、9つのUIコントロール部品を用意しています。

## SNAPROP ツーバー

-  CustTextBox
-  CustButton
-  CustCheckBox
-  CustRadioButton
-  CustSelectControl
-  CustDateControl
-  CustTimeControl
-  CustLabel
-  CustListView

## UIコントロール部品を利用するメリット



指で操作しやすいサイズになっています。



デザインに頭を悩ませる必要がない。  
標準のデザインで綺麗な分かり易い画面を作れます。

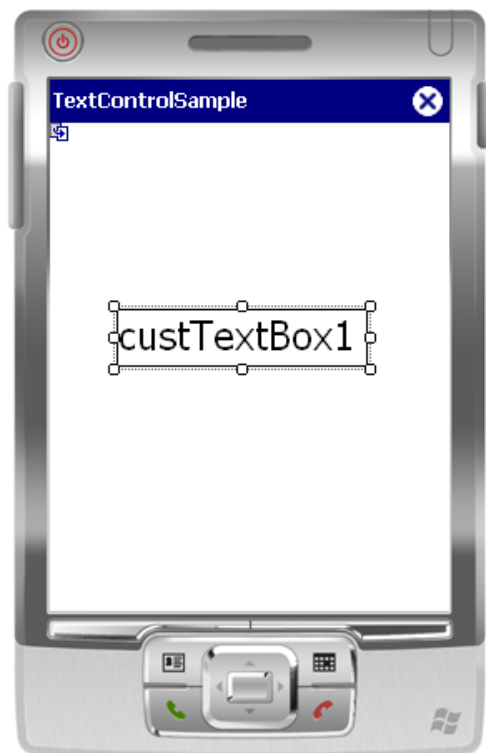
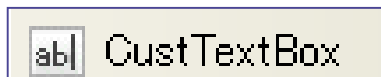


マップファイルと組み合わせて、  
項目チェック処理をすぐ実装できます。



フォーカスが当たったときに色を変えたり、  
ボタンにショートカットキーを設定できたりと、  
細かいところまで手が届いています。

# ① CustTextBox コントロール (1 / 3)



## 出来ること

- 単一行、複数行のテキストを入力できます。
- パスワード文字のマスクが行えます。
- 入力できるテキストの文字数を制限できます。
- 入力可能・読み取り専用を切り替えることが出来ます。
- フォーカスが当たった際の背景色を設定できます。
- フォーカスが当たった時の入力文字モードを設定可能です。  
→ フォーカスが当たったら数値入力モードにする、など
- マップファイルと連携することで、  
「必須チェック」、「文字列長チェック」、  
「文字データ種別チェック」、などを実現できます。
- フォーカスが当たったときと、外れたときの表示フォーマットを切り替えることができます。

# ① CustTextBox コントロール (2/3)

## 製品付属のサンプルプログラム

### 単一行・複数行入力

単一行

1234567890

複数行

abcdefghijklmnop  
nopqrstu

フォーカス時の色指定可能

### 必須入力チェック

⚠ '入力欄1'は必須入力です。

### 文字列長チェック

▲4桁以上入れると文字列長エラーになるように設定しています

⚠ '入力欄1'の入力は4文字までです。

### 文字種別チェック

▼半角文字を入力可能(type="Half")

1 2345

▼全角文字を入力可能(type="Full")

▼全角ひらがなを入力可能  
(type="Hiragana")

文字データ種別チェック

⚠ '入力欄2'が半角文字ではありません。

#### ▼チェックできる文字種別

- 半角文字
- 全角文字
- 全角ひらがな
- 全角カタカナ
- 半角カタカナ
- 半角英数字
- 全角英数字

# ① CustTextBox コントロール (3/3)

## 製品付属のサンプルプログラム

### 数値データチェック

▼整数値を入力可能(type="Integer")

12.5

▼整数値を入力可能、桁数指定  
(type="Integer", precision="桁数")

3桁までで設定



'入力欄1' が整数値ではありません。

#### ▼チェックできる項目

- 整数値チェック
- 実数値チェック
- 桁数チェック
- 最小値・最大値チェック
- 全角カタカナ
- 半角カタカナ
- 半角英数字
- 全角英数字

### 入力フォーマットと表示フォーマットを制御

フォーカスが当たったときは、  
入力しやすいフォーマット

フォーカスが外れたときは、  
分かり易いフォーマット

▼整数(カンマ区切り)

123456

▼整数(カンマ区切り)

123,456

▼整数(前ゼロ埋め6桁)

123

▼整数(前ゼロ埋め6桁)

000123

▼郵便番号

1234567

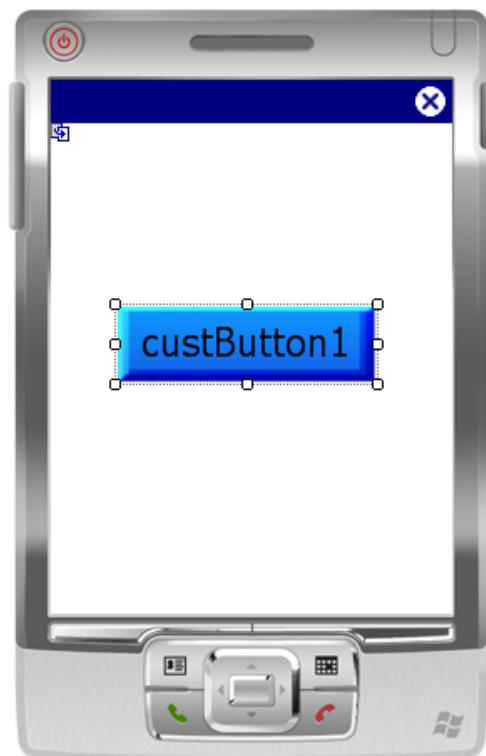
▼郵便番号

123-4567

## ② CustButton コントロール (1/2)



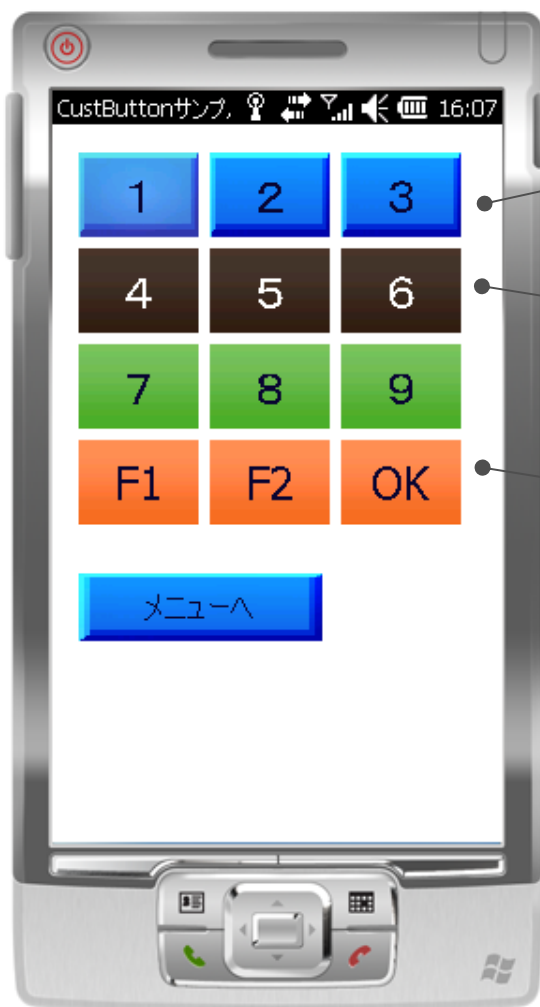
### 出来ること



- 以下の状態に応じた画像を用意して、ボタンの外観を変更できます。
  - ① フォーカスが当たっている状態
  - ② フォーカスが外れている状態
  - ③ ボタンがクリックされた状態
  - ④ ボタンが無効である状態
- ボタンのクリックをショートカットキーで実行することができます。ショートカットキーに設定できるのは、1～9、F1、F2、OKボタン
- 擬似透過設定が行えます。

## ② CustButton コントロール (2/2)

### 製品付属のサンプルプログラム



標準イメージ

画像を用意してデザインを変更できる

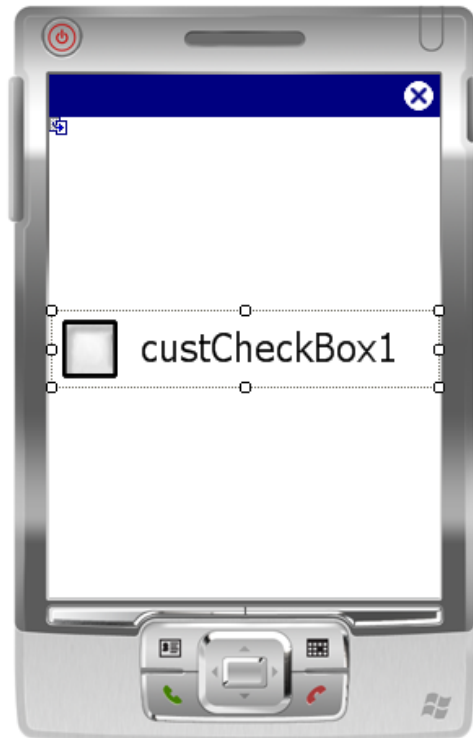
ショートカットキーでボタンをクリックできる

擬似透過が可能

※画像の1行目を背景色と同一色に設定することで擬似透過する



### ③ CustCheckBox コントロール (1 / 2)



#### 出来ること

- チェックボックスをクリックすると、その状態を「オン」から「オフ」に、または「オフ」から「オン」に変更出来ます。
- 表示方法を、2種類から選ぶことが出来ます。
  - ① イメージ + ラベル
  - ② ボタンによるトグル表示
- フォーカスが当たった際の枠線色を設定出来ます。
- 色パターンによって色味を変更できます。
- 個別画像を用意して、概観を変更できます。



# ③ CustCheckBox コントロール (2/2)

## 製品付属のサンプルプログラム

### 表示方法

通常パターン

- りんご
- みかん

トグルボタンパターン

- りんご
- みかん

トグルボタンパターン (チェックボックスなし)

- りんご
- みかん

### 色設定

- ▼ColorPattern=Blue
  - 青(デフォルト)
- ▼ColorPattern=Brown
  - 茶
- ▼ColorPattern=Green
  - 緑
- ▼ColorPattern=Orange
  - オレンジ
- ▼ColorPattern=Red
  - 赤

通常表示

トグル表示

予め用意された色パターンで色味を変更可能

フォーカスが当たった際の枠線の色を指定可能

## ④ CustRadioButton コントロール (1/2)

 CustRadioButton

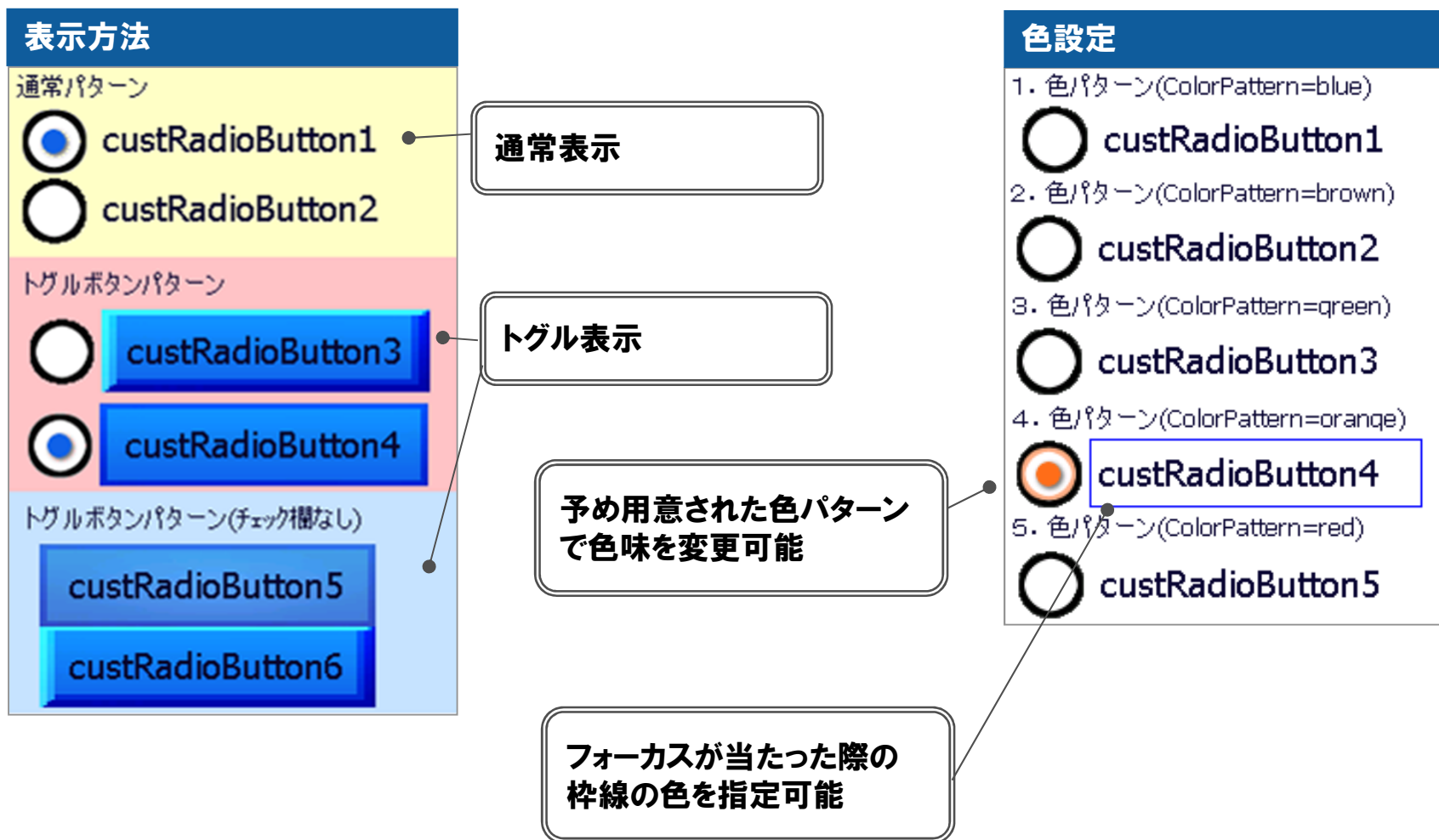


### 出来ること

- グループ内のチェックボタンを選択すると、他のチェックボタンは自動的にオフになります。
- 表示方法を、2種類から選ぶことが出来ます。
  - ① イメージ + ラベル
  - ② ボタンによるトグル表示
- フォーカスが当たった際の枠線色を設定出来ます。
- 色パターンによって色味を変更できます。
- 個別画像を用意して、概観を変更できます。

## ④ CustRadioButton コントロール (2/2)

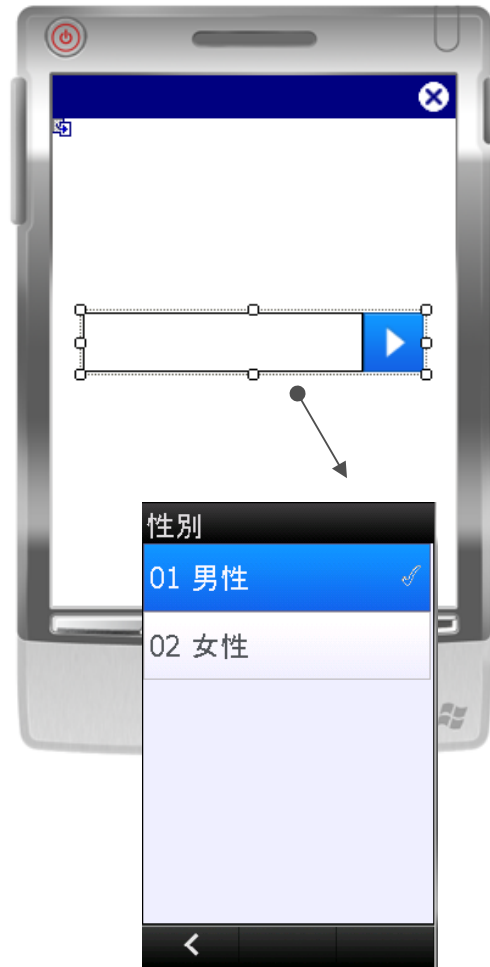
### 製品付属のサンプルプログラム



## ⑤ CustSelectControl (1/2)



### 出来ること



- 選択項目が別画面に表示されるので、使いやすいコントロールを実現できます。
- ジェスチャー操作で選択項目をスクロールできます。
- 複数項目選択が可能です。
- 色パターンを利用して、デザインを簡単に設定出来ます。
- マップファイルと連携して「必須チェック」を実装できます。

# ⑤ CustSelectControl (2/2)

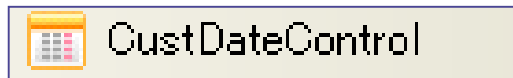
## 製品付属のサンプルプログラム

The image displays several examples of the CustSelectControl widget in a mobile application. On the left, three panels illustrate different selection modes: '単一選択' (Single Selection) with a yellow button labeled '01 男性', '複数選択' (Multiple Selection) with a yellow button labeled '01 docomo, 02 SoftB...', and '必須項目チェック' (Required Item Check) with red buttons for '単一項目' and '複数項目', and a 'チェック実行' button. A warning icon and text at the bottom of the third panel state: '性別'は必須入力です。 (Gender is a required input).

In the center, two dropdown menus are shown. The first is titled '性別' (Gender) and lists '01 男性' (checked) and '02 女性'. The second is titled '保有ケータイのキャリア' (Carrier of the mobile phone you own) and lists '01 docomo' (checked), '02 SoftBank' (checked), '03 AU' (checked), and '04 その他' (Others). A '選択' (Select) button is at the bottom.

On the right, a '色パターン設定' (Color Pattern Setting) screen is shown. It lists five color patterns: 1. blue (01 男性), 2. brown (02 女性), 3. green (01 男性), 4. orange (02 女性), and 5. red (01 男性). A small inset shows a preview of the '性別' dropdown with '01 男性' selected.

## ⑥ CustDateControl (1 / 2)



### 出来ること



- テキストによる日付入力と、タッチ操作による日付入力が可能です。
- フォーカスが外れた際の表示を「YYYY/MM/DD」や「YYYY年MM月DD日」といった書式に出来ます。
- 「年月日」入力、「年月」入力をサポートします。
- 入力できる年の範囲を指定出来ます。
- 初期表示日付を指定出来ます。
- 色パターンを利用して、デザインを簡単に変更出来ます。
- マップファイルと連携して「必須チェック」、「入力範囲チェック」を実装出来ます。

# ⑥ CustDateControl (2/2)

## 製品付属のサンプルプログラム

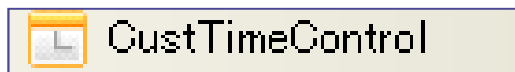
The image displays a mobile application interface with several date-related controls and settings:

- 年月日入力 (Year/Month/Day Input):** A text input field containing "20100425" with a blue play button. An arrow points to a date picker dialog showing year (2010), month (4), and day (25).
- 年月入力 (Year/Month Input):** A text input field containing "201004" with a blue play button. An arrow points to a date picker dialog showing year (2010) and month (4).
- 必須項目チェック (Required Item Check):** A red bar with a play button and a blue bar labeled "必須チェック". Below it, a warning icon and text: "'生年月日' は必須入力です。" (Birth date is required input).
- 色パターン設定 (Color Pattern Setting):** A list of five color patterns with corresponding text input fields and play buttons:
  - 1. 色パターン(ColorPattern=blue)
  - 2. 色パターン(ColorPattern=brown)
  - 3. 色パターン(ColorPattern=green)
  - 4. 色パターン(ColorPattern=orange): Contains "20100401" with a green play button. An arrow points to a date picker dialog showing year (2010), month (4), and day (1).
  - 5. 色パターン(ColorPattern=red)

**入力フォーマットと表示フォーマットを制御 (Control Input and Display Formats):**

- フォーカスが当たったとき (When focus is gained):** The input field shows "20100425" in a yellow background.
- フォーカスが外れたとき (When focus is lost):** The input field shows "2010/04/25" in a white background.

## ⑦ CustTimeControl (1/2)



### 出来ること



- テキストによる時刻入力と、タッチ操作による時刻入力が可能です。
- フォーカスが外れた際の表示を「HH:mm」や「HH時mm分」といった書式に出来ます。
- 初期表示日付を指定出来ます。
- 色パターンを利用して、デザインを簡単に変更出来ます。
- マップファイルと連携して「必須チェック」、「入力範囲チェック」を実装出来ます。



# ⑦ CustTimeControl (2/2)

## 製品付属のサンプルプログラム

### 時刻入力

0315

CustTimeControl01 時刻入力

+ +

3 15

- -

< クリア 選択

### 色パターン設定

- 色パターン(ColorPattern=blue)
- 色パターン(ColorPattern=brown)
- 色パターン(ColorPattern=green)
- 色パターン(ColorPattern=orange)
- 色パターン(ColorPattern=red)

2205

### 必須項目チェック

必須チェック

⚠ '到着時間' は必須入力です。

### 入力範囲チェック

▼入力値の範囲を08:00~17:00に設定した場合のサンプル

2000

範囲チェック

⚠ '時刻2' は17:00以前の時刻でなければなりません。

### 入力フォーマットと表示フォーマットを制御

フォーカスが当たったとき

フォーカスが外れたとき

0315 → 03時15分

CustTimeControl01 時刻入力

+ +

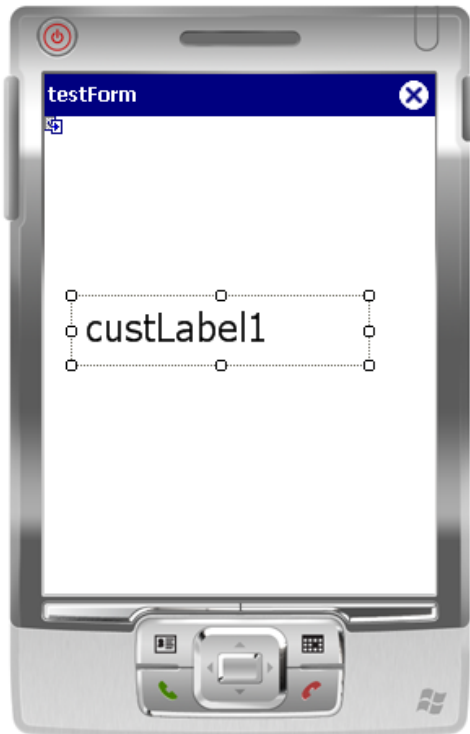
22 5

- -

< クリア 選択

## ⑧ CustLabel (1 / 2)

### A CustLabel



### 出来ること

- 境界線の描画と色指定が可能です。
- 表示するテキストの横位置と縦位置を指定可能です。
- 背景色にグラデーションを設定可能です。

## ⑧ CustLabel (2/2)

### 製品付属のサンプルプログラム

#### 背景色指定

標準

標準

背景色変更

背景色つき

#### 枠線描画・色指定

枠線設定

枠線付き

#### グラデーション指定

```
DrawGradientMode = Center  
StartColor= R:255,G:192,B:203  
EndColor= R:255,G:0,B:0
```

グラデーション1

```
DrawGradientMode = TopToBottom  
StartColor= R:255,G:255,B:0  
EndColor= R:0,G:128,B:0
```

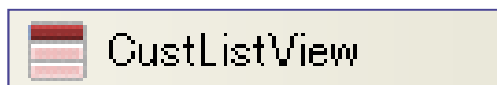
グラデーション2

```
DrawGradientMode = BottomToTop  
StartColor= R:135,G:206,B:235  
EndColor= R:0,G:0,B:255
```

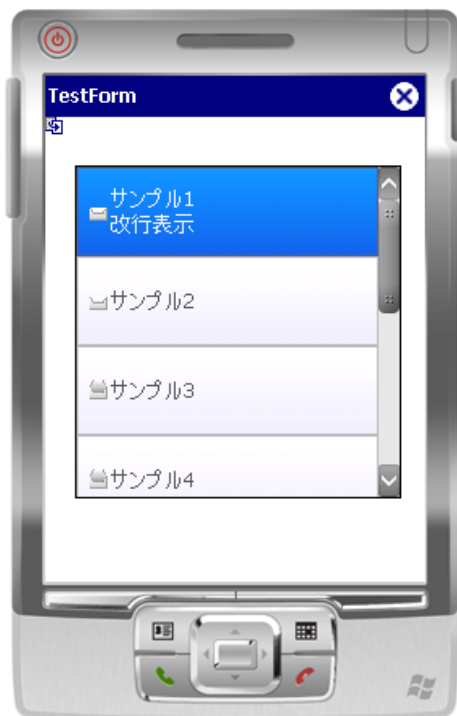
グラデーション3

- ・上から下へのグラデーション
- ・下から上へのグラデーション
- ・真ん中から外側へのグラデーション

## ⑨ CustListView (1 / 2)



### 出来ること

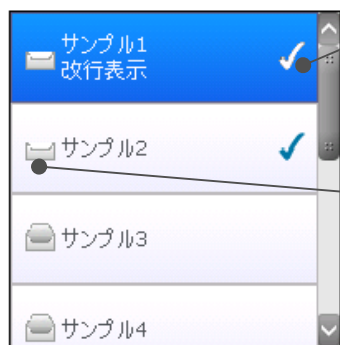


- 1列の項目データを表示することができ、項目の左側には特定の画像データを、右側にはチェック画像を表示出来ます。
- 項目の高さ調節や色パターンによる見た目の調節が可能です。
- OwnerDrawプロパティで、外観を独自描画することが出来ます。

# ⑨ CustListView (2/2)

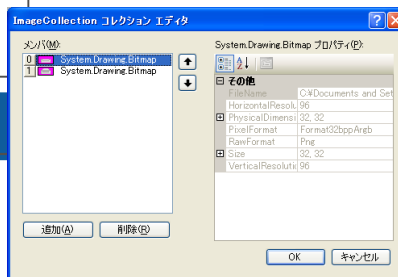
## 製品付属のサンプルプログラム

### 標準デザイン



・項目左の画像と、チェック画像は不要であれば非表示にできます。

・左に表示する画像は、ImageListクラスを利用して制御します。



### 高さ調整



・横に複数項目を並べることは出来ないため、改行して縦方向に項目を並べる形になります。

### 色パターン指定



・色パターンを変更して、見た目を切り替えることができます。



## デバイス制御部品（4機能）

SNAPROPでは、4つのデバイス制御部品を用意しています。

## SNAPROP ツーバー

-  BarCodeControl
-  BarCodeComponent
-  CameraControl
-  GpsComponent

## デバイス制御部品を利用するメリット



F-05Bを利用したバーコード読み取り機能を簡単に実装可能です。

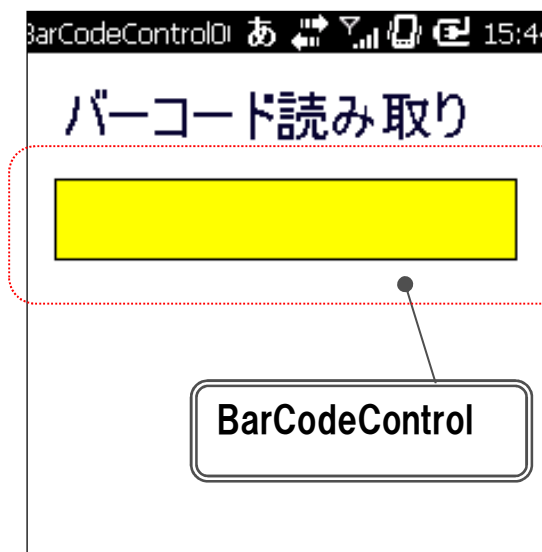
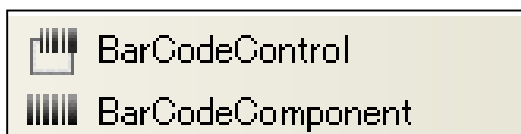


カメラ撮影機能やGPS機能を利用して、付加価値のある業務サービスを実現出来ます。



F-05B端末の「バーコード読み取り」と「カメラ撮影」は一つのスキャナデバイスで実現されています。そのため、本来、デバイス競合が発生しないような考慮が必要ですが、SNAPROP部品がその部分を吸収します。

# ① BarCodeControl、②BarCodeComponent (1/2)



## 出来ること

- バーコードを読み取って、そのデータをテキストボックスに転記する機能は、BarCodeControlを配置するだけで、基本処理が実装されます。(バーコード読み取り+テキストボックスのセットが、BarCodeControlとなります)
- バーコードをスキャンするためには、スキャナデバイスをバーコードモードで事前にOpen処理する必要があります。BarCodeControlでは、テキストボックスにフォーカス当たった際にOpen処理を自動で実行することが出来ます。
- BarCodeControl で読み取るバーコードのフォーマットをON/OFFできるので、読み取ることのない不要なバーコードフォーマットはOFFに出来ます。
- スキャン時イベントを用意しているため、データの妥当性チェック処理などを追加出来ます。
- スキャンエラー時イベントを用意しているので、エラーのハンドリングが簡単です。
- 速度が重要視される場合は、BarCodeComponentを利用して、スキャナデバイスのOpen/Close処理を任意タイミングで実効出来ます(アプリ起動時に、1度だけOpen処理するなど)。また、BarCodeComponentを利用すれば、読み込んだデータを別コンポーネントに貼り付けたりといった、柔軟な実装が可能です。



# ① BarCodeControl、②BarCodeComponent (2/2)

## 製品付属のサンプルプログラム



BarCodeControlにフォーカスが当たれば、自動でOpen状態となり、読みとったデータはテキストボックスに張り付きます



NW7、Code39といったF-05Bが対応している14のバーコードフォーマットについて、それぞれON/OFF設定が行えます。



BarCodeComponentを利用して、ListViewコントロールに読み取りデータをためて行くといったような柔軟な実装が可能になります。

### ③ CameraControl (1 / 2)

CameraControl



#### 出来ること

- CameraControlを画面に貼り付けて必要な設定を行い、撮影のタイミングを実装することで、カメラを使った写真撮影処理を実現出来ます。
- CameraControlのプロパティで以下の設定を行えます。
  - ・プレビューを行うかどうか
  - ・モノクロデータで保存するかどうか
  - ・ズーム状態の変更
  - ・カメラ解像度の変更
- カメラ撮影完了時イベントを用意しているので、そのタイミングで撮影画像データをファイル保存したり、画面表示させたりといったことが可能になります。
- 撮影エラー時イベントを用意しているので、エラーのハンドリングが簡単です。
- カメラ撮影するためには、スキャナデバイスをカメラモードで事前にOpen処理する必要があります。CameraControlでは、フォーカスがコントロールに当たった際にOpen処理を自動で実行することが出来ます。

# ③ CameraControl (2/2)

## 製品付属のサンプルプログラム

### プレビュー表示ON/OFF設定

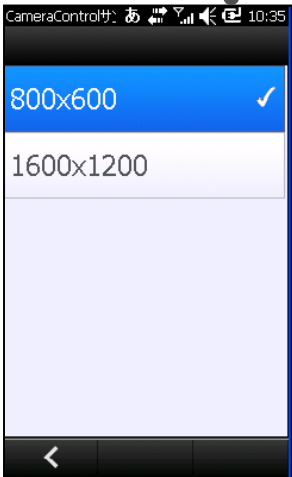
カメラ撮影時のプレビュー表示をプロパティでON/OFFすることができます。

カメラ解像度の変更を簡単に行えます。

撮影画像表示部

プレビュー部

### カメラ解像度の設定



### モーション画像で保存

撮影した後の画像をモーションで保存して、ファイルサイズを小さく出来ます。

### ズーム値変更

ズーム設定値もSNAPPROPで変更可能です。

## ④ GpsComponent (1 / 2)

 GpsComponent



mainMenu1

 epsComponent1

GpsComponent

### 出来ること

- 現在位置の緯度、経度の情報が取得可能です。
- 周期的に現時位置の緯度、経度の情報が取得可能です。また、前回取得した位置からの移動距離(メートル)を取得出来ます。
- 世界測地系、日本測地系の2種類の情報を使用する事が出来ます。
- 世界測地系から日本測地系への変換が可能です。
- 世界測地系の2点を指定し、距離を計算出来ます。
- 捕捉している衛星の数を知ることが出来ます。

※F-05B端末は、GPSに対応しておりますが、基地局情報を利用した測位には対応していません。

# ④ GpsComponent (2/2)

## 製品付属のサンプルプログラム

### 位置情報の取得 (任意タイミング)

位置情報を取得

(緯度)世界測地系:35.5555260100233  
(経度)世界測地系:139.720879801738  
(緯度)日本測地系:35.5522865244778  
(経度)日本測地系:139.724079706391  
補足している衛星の数:6

ボタンを押したタイミングなどで位置情報を取得します。

### 位置情報の取得 (周期処理)

周期的に位置情報を取得

[取得日時]11/06/15 18:28:30  
(緯度)世界測地系:35.5559797907162  
(経度)世界測地系:139.720656753373  
(緯度)日本測地系:35.5527403576031  
(経度)日本測地系:139.723856660397  
移動距離(メートル):0  
補足している衛星の数:5

180秒間隔で取得



位置情報取得間隔を設定して周期処理をスタートさせれば、一定タイミングで位置情報を取得できます。また、前回取得からの移動距離を取得出来ます。

### 2点間の距離算出

距離を求める

▼東京駅  
(緯度2)世界測地系:35.681382  
(経度2)世界測地系:139.766084  
▼現在地  
(緯度1)世界測地系:35.5555260100233  
(経度1)世界測地系:139.720879801738  
補足している衛星の数:6

距離(km)14.5952094744913

2点の緯度経度を指定して、その距離を算出できます。

### 世界測地系から日本測地系への変換

変換する

▼東京駅の位置情報(世界測地系)  
世界測地系:35.681382  
世界測地系:139.766084

▼東京駅の位置情報(日本測地系)  
日本測地系:35.6781551864295  
日本測地系:139.769293454107



## 通信制御クラス（4クラス）

# ① Connection Managerクラス（1 / 2）

Connection Manager 機能を利用することで、Windows Mobileのネットワーク接続を一元的に管理出来ます。

## 出来ること

- 無線LANや3G接続などの接続を一元的に管理出来ます。
- 全てのネットワーク(Wi-Fi、mopera U)を切断する処理が可能です。
- 接続するネットワークを明に指定したり、優先順位をつけて接続を行ったりが可能です。
- Wi-FiのON/OFFを切り替えることが出来ます。
- RASエントリの一覧を取得することが出来ます。

# ① Connection Managerクラス (2/2)

## mopera U でネットワーク接続するサンプル



## 優先順位をつけてネットワーク接続するサンプル





## ② TerminalCommunicationクラス (1 / 2)

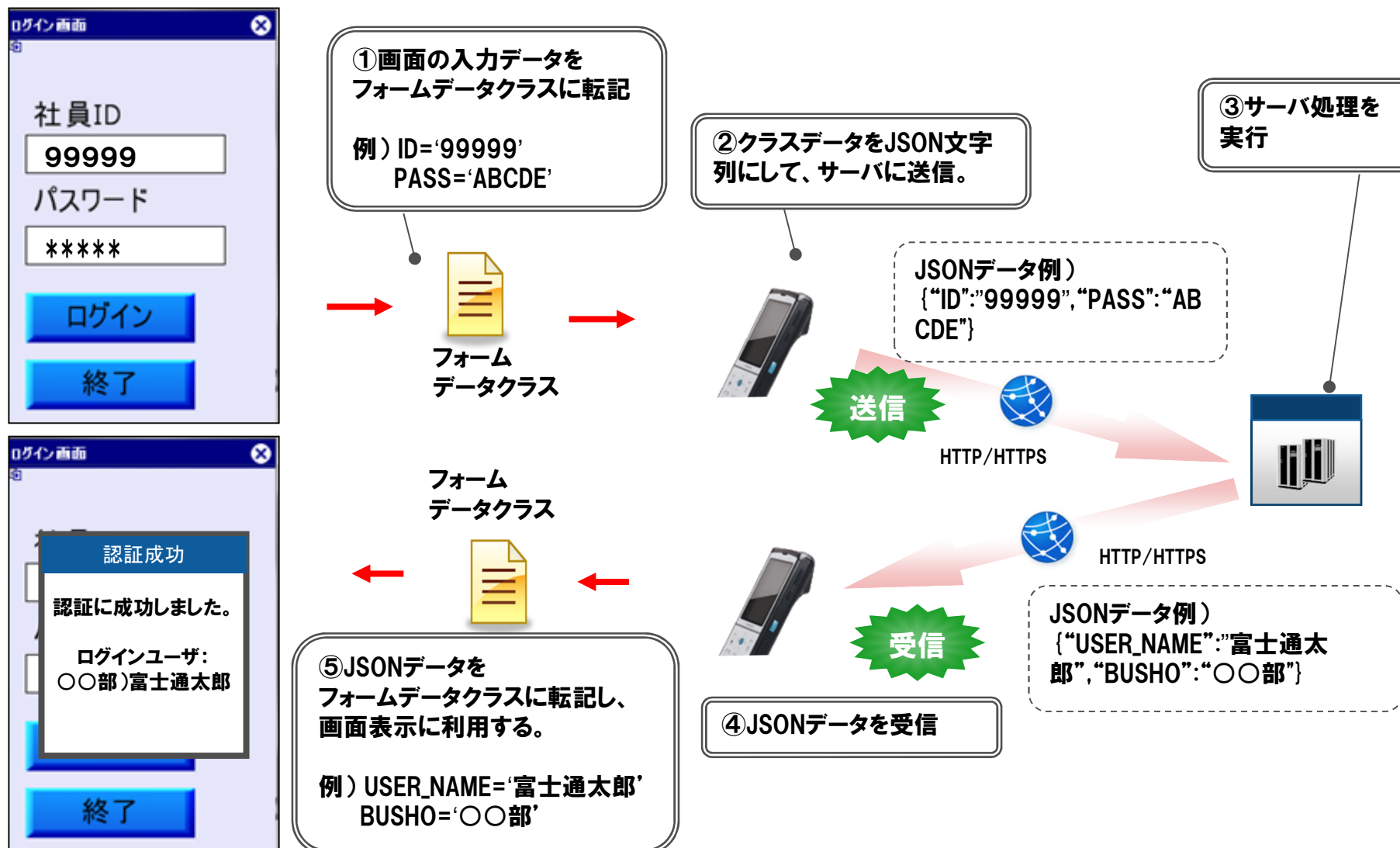


### 出来ること

- 端末上のファイルを一括でサーバに送信できます。  
(HTTPのマルチパートフォーマットで送信)
- サーバ上のファイルを一括で受信できます。  
(HTTPのマルチパートフォーマットで受信)
- 端末のクラスデータをJSON形式に変換して送信することが可能です。また、サーバ側からJSON形式のデータをレスポンスとして受け取るとり、端末側のクラスデータに格納することが可能です。
- データ送信時に端末を一意に認識する為のサブスクライバID、端末識別番号、電話番号をフォームデータとして送信することができます(出力しない事も可能)。

## ② TerminalCommunicationクラス (2/2)

### JSON形式でのデータ送受信イメージ



### ③ 周期通信クラス

SNAPROPがもつ周期通信機能を用いることで、業務作業で端末側に溜めておいたデータを1時間に1回などの定期的なタイミングで、サーバへ一括送信することが出来ます。

#### 出来ること

- 同期マップ(XML)に定義するだけで、端末側で更新された情報をサーバ側に送信可能です。
- 集荷テーブル、集荷明細テーブルというような関連するテーブルをサーバ側に送信し、2テーブルを同時にコミットする事が可能です。
- テーブルにファイル(イメージ等)の格納場所が保存されていた場合、そのフィールド名を同期マップに定義する事でファイルをサーバ側に送信する事が可能です。

※サーバ側のデータ受けプログラムは、個別に開発する必要があります。

## ④ メール送信クラス

MailComponent クラスを利用して、メールの送受信機能を実装することができます。

### 製品付属のサンプルプログラム

#### メール送信サンプル

メール送信

▼説明

- メールの送信はMailComponentのSendメソッドを使用します。
- メール送信 (Sendメソッドの実行)にあたっては、MailComponentにメールサーバ情報を事前にセットする必要があります。
- 情報のセットについては、ロジックで個別にセットするか、設定マップ (cmap) ファイルでそれをコード上で反映させます。
- 処理の箇所は、必ずtry, catch でラージの処理を実装してください
- 本サンプルでは、mopera Uの回線にてmoperaのメールサーバを利用して送受信するサンプルにしています。

#### メール受信サンプル

メール受信

受信したメールをサーバから削除

▼説明

- メール受信処理にはMailComponentのRecieveメソッドを使用します。
- メール受信 (Recieveメソッドの実行)にあたっては、MailComponentにメールサーバ情報を事前にセットする必要があります。
- 情報のセットについては、ロジックで個別にセットするか、設定マップ (cmap) ファイルに定義してそれをコード上で反映させます。
- 処理の箇所は、必ずtry, catch で囲んで、エラーの処理を実装してください

### 出来ること

- POP3でメールを受信することが可能です。
- SMTPでのメールの送信が可能です。  
(ただし、添付ファイルの送信は行えません)
- SMTP-AUTH認証でのメール送信に対応しています。



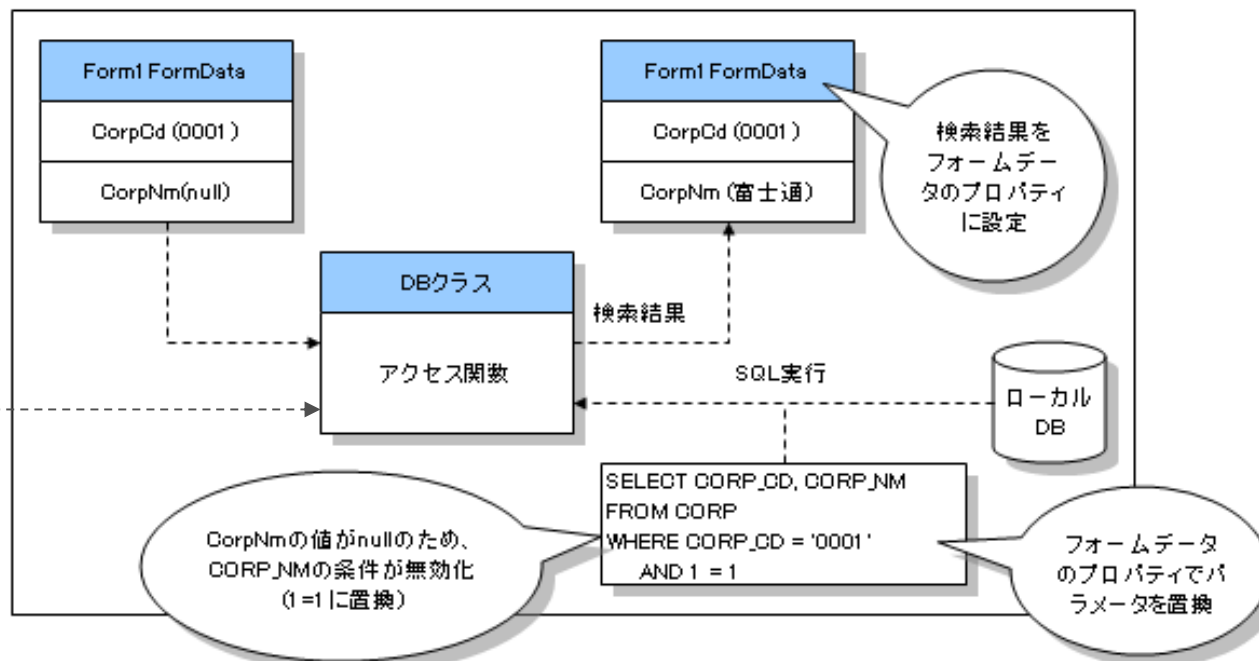
## データベース関連部品（3機能）

# ① SQLマッピング機能（1/2）

SQLマップファイル (XML形式) に定義したSQLを使用して、SQL Server Compact 3.5へのアクセスを行う機能を提供します。データベースへの接続の管理や、トランザクション管理を行うことができます。SQL実行時のパラメータにnullを指定した場合、WHERE句のパラメータ部分が“1=1”のように置換する機能があり、AND検索などで一部の条件を除外したい場合に使用出来ます。

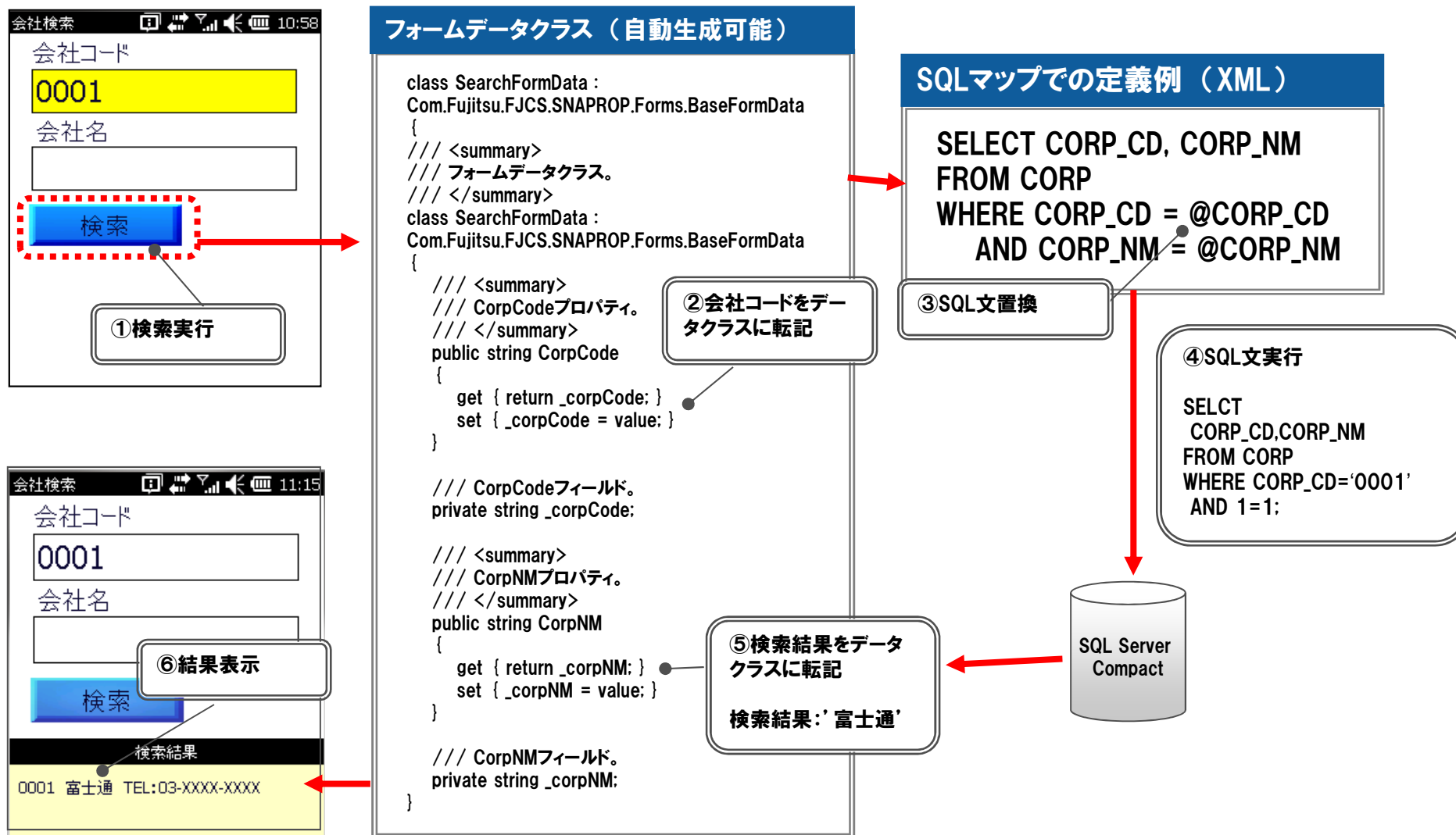
## SQLマップでの定義例 (XML)

```
SELECT CORP_CD, CORP_NM  
FROM CORP  
WHERE CORP_CD = @CORP_CD  
AND CORP_NM = @CORP_NM
```



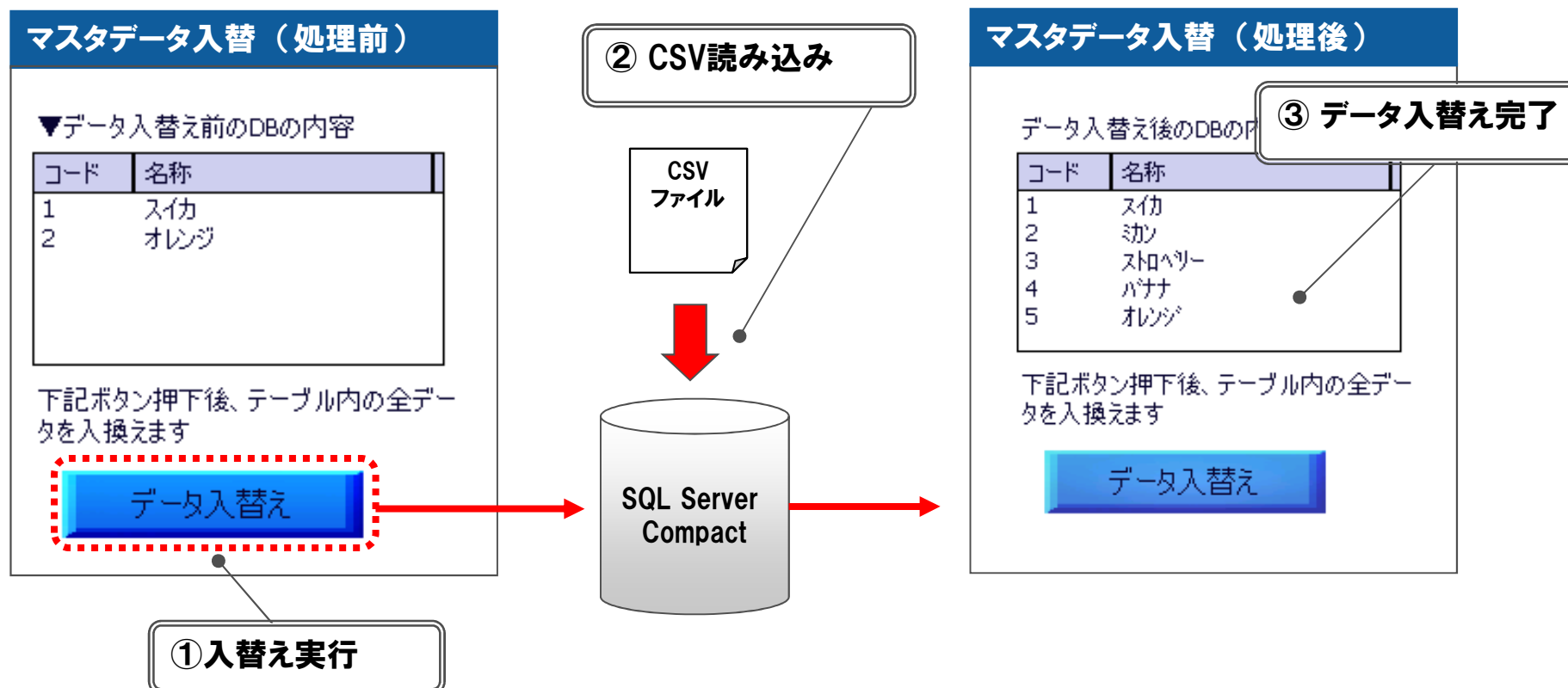
# ① SQLマッピング機能 (2/2)

データのマッピングにより、DB検索結果・DBパラメータとフォームデータとの間で相互に値の転送が可能です。



## ② マスタデータ取り込み機能

CSVファイルの内容で、端末側のDB内のテーブルデータを全件差し替えすることが可能です。  
また、追加でデータを入れ込むことも可能です。







## Utilクラス（2クラス）

# ① 端末情報取得機能（TerminalInfoクラス）

SNAPROPが用意する端末情報取得機能を用いて、端末の電話番号、デバイスID、OSバージョンなどの様々な情報を簡単に取得することが出来ます。

## 取得できる端末情報

- 端末のメーカー名
- 端末のモデル名
- 端末のリビジョンを取得
- 端末識別番号 (IMEI)（返される値は端末固有となります）
- サブスクライバID (IMSI)（返される値はユーザ毎に一意的な識別番号となります）
- 端末のOEM情報
- OSのバージョン
- CompactFrameworkのバージョン
- 端末の電話番号
- 端末のデバイスID
- 画面の全体のサイズ
- 画面の作業領域（タスクバーの領域を含まない領域）

## ② Log出力機能

アプリケーション開発に必ず必要となるログの出力に関しても、SNAPROPの部品を利用出来ます。

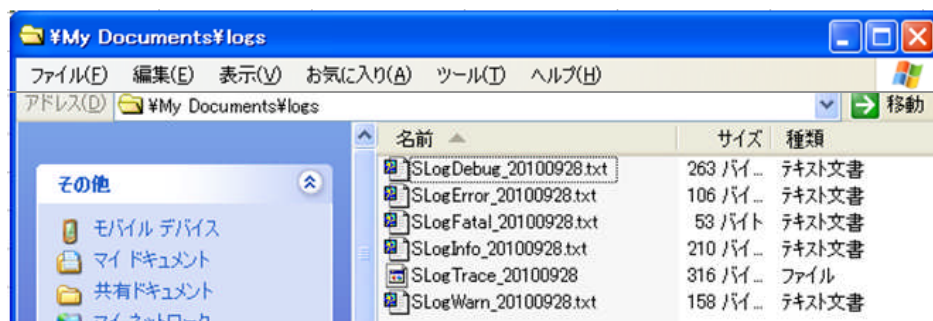
### 出来ること

- 出力ファイル名、ログ出力レベルを指定しログの出力が可能です。
- Logファイルの世代管理の設定が可能です。

### ロガーの登録

```
Log.AddLogInfo(new LogInfo("Logテスト1", @"%My Documents%logs%SLogTrace", 10, LogLevel.Trace));
Log.AddLogInfo(new LogInfo("Logテスト2", @"%My Documents%logs%SLogInfo.txt", 10, LogLevel.Info));
Log.AddLogInfo(new LogInfo("Logテスト3", @"%My Documents%logs%SLogDebug.txt", 10, LogLevel.Debug));
Log.AddLogInfo(new LogInfo("Logテスト4", @"%My Documents%logs%SLogWarn.txt", 10, LogLevel.Warn));
Log.AddLogInfo(new LogInfo("Logテスト5", @"%My Documents%logs%SLogError.txt", 10, LogLevel.Error));
Log.AddLogInfo(new LogInfo("Logテスト6", @"%My Documents%logs%SLogFatal.txt", 10, LogLevel.Fatal));
```

### ログファイルの出力





## 個別アプリケーション（3機能）

# ① ランチャー機能 (1/2)

SNAPROPが提供するランチャー機能を利用することで、必要な機能をすぐに呼び出すことができます。また、許可されていないアプリケーションはランチャーの背面に隠される作りになっていますので、業務で不要なアプリケーションは利用を制限することが出来ます。



# ① ランチャー機能 (2/2)

## ▼ 設定ファイルイメージ

```
<?xml version="1.0" encoding="utf-8" ?>
<settings>

  <launcher>
    <caption>業務アプリ</caption>
    <icon>Images/sample.png</icon>
    <focusIcon>Images/sample.png</focusIcon>
    <clickIcon>Images/sample.png</clickIcon>
    <execute>%Windows%gyomu.exe</execute>
    <args></args>
  </launcher>
  <launcher>
    <caption>メモ帳</caption>
    <icon>%Windows%Start_Icon_Notes.png</icon>
    <focusIcon>%Windows%Start_Icon_Notes.png</focusIcon>
    <clickIcon>%Windows%Start_Icon_Notes.png</clickIcon>
    <execute>%Windows%word.exe</execute>
    <args></args>
  </launcher>
  <launcher>
    <caption>予定表</caption>
    <icon>%Windows%Start_Icon_Calendar.png</icon>
    <focusIcon>%Windows%Start_Icon_Calendar.png</focusIcon>
    <clickIcon>%Windows%Start_Icon_Calendar.png</clickIcon>
    <execute>%Windows%outlook.exe</execute>
    <args>calendar</args>
  </launcher>

```

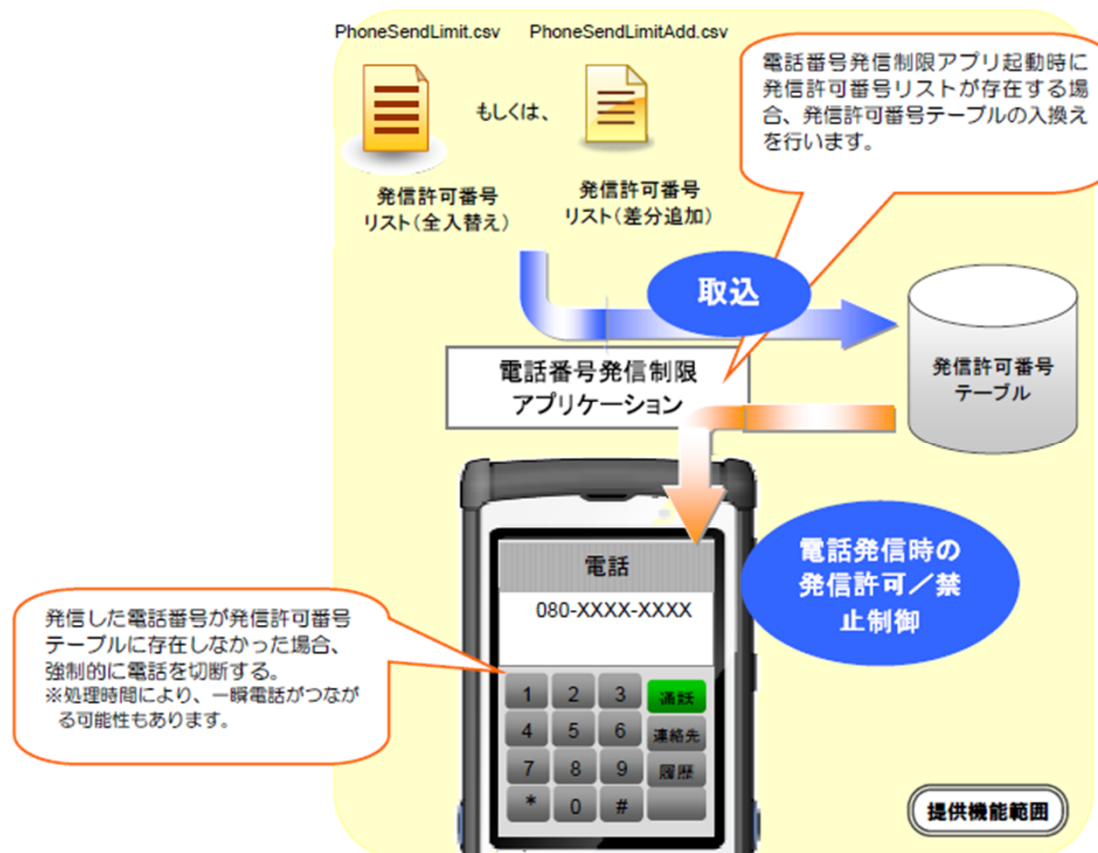
・・・途中省略・・・

## 出来ること

- ランチャーに表示させる項目は、XMLファイルで定義出来ます。
- 個別に画像を用意すれば、ランチャーのデザインを変更することが出来ます。
- 画面下部のボタンでページの切り替えが可能です。
- ランチャーはユーザによって終了させることは出来ません。  
※開発中などでランチャーを終了させる必要がある場合は、終了機能を設定することも出来ます。
- ランチャーから起動したアプリケーションを除き、常にランチャーの画面は前面に表示されます。

## ② 電話発信制限機能

SNAPROPでは、指定した電話番号以外への発信を制限する個別アプリケーションを提供します。発信を許可する電話番号については、CSVファイルを用いて用意にメンテナンスが可能です。



### 特徴

- 発信許可番号リスト(CSV)ファイルが存在する場合、発信許可番号テーブルの入換えを行います。
- 発信した電話番号が発信許可番号テーブルに存在しなかった場合、強制的に電話を切断します。

## ③ アプリケーション機能制限機能

不要なアプリケーションを起動できないようにする、「アプリ起動制限機能」を提供しています。（ランチャー機能の場合は、禁止アプリケーションは背面に隠すだけですが、アプリケーション制限機能は禁止アプリケーションを停止します）

### ▼ 設定ファイルイメージ

```
<?xml version="1.0" encoding="utf-8" ?>
<config>
  <interval>[監視間隔]</interval>
  <execute order="[比較順番]">
    <wait>[終了待機時間]</wait>
    <deny>
      <path>[実行モジュールのパス]</path>
    </deny>
    <allow>
      <path>[実行モジュールのパス]</path>
      <startup />
    </allow>
  </execute>
</config>
```

### 出来ること

- 許可されていないアプリケーションが実行された場合、強制的に終了させることができます。
- ユーザは、アプリ起動制限機能をタスクマネージャーから終了をすることは出来ません。
- アプリ起動制限機能をインストーラーでスタートアップにインストールすることで、端末が起動されている間中、アプリ起動制限を有効にすることができます。
- 許可／禁止設定は、XML形式の設定ファイルで簡単に設定が行えます。





# 開発スピードを上げるためのツール

# ① 自動生成機能

SNAPROPでは、画面項目の入力チェックを行うためのマップファイルや、入力データを格納するためのフォームクラスを自動生成することができます。

① Visual Studioで画面の項目を作成します。

② 右クリックして、フォームマップを作成します

③ ファイル名などを指定して自動生成を実行します。

④ 自動生成完了！

フォームマップ (XMLファイル)    ドメインマップ (XMLファイル)    フォームデータクラス

- 画面モード制御
- 必須入力項目制御
- 入力チェック制御
- 文字列書式制御

出力先  
フォームマップ: LoginForm.fmap    参照  
ドメインマップ: LoginForm.dmap    参照

出力モード

オプション  
 フォームデータの作成

キャンセル    確定

## ② マップファイル編集ツール（1/2）

マップファイルを編集するためのSNAPROPDeveloperを使えば、必須入力チェックや文字種チェック、表示フォーマット設定、入力モード設定を簡単に登録できます。

The screenshot shows the 'マップファイル管理' (Map File Management) application. The main window displays the 'LoginForm.dmap' domain map. A table lists the items:

ドメインID	項目名	データ種別	入力モード	入力桁数	最大桁数	数値の桁数	小数部有効桁数	数値の最小
LoginForm_tbxID	社員ID	Integer	Numeric	8	8			
LoginForm_tbxPa...	パスワード	Character	Numeric	8	8			

Below the table is the 'ドメインマップ編集' (Domain Map Edit) panel for the selected item. It contains various configuration fields:

- ドメインID (id): LoginForm\_tbxID
- 項目名 (name): 社員ID
- データ種別 (type): 整数値(Integer)
- 入力モード (inputMode): 数字(Numeric)
- 入力桁数 (inputLength): 8
- 最大桁数 (maxLength): 8
- 数値の桁数 (precision):
- 小数部有効桁数 (scale):
- 数値の最小値 (minValue):
- 数値の最大値 (maxValue):
- デフォルト値 (defaultValue):
- 項目説明のメッセージID (information): SNAP9000
- 表示書式 (viewFormat): 数値8桁(前ゼロ埋め)
- 表示パターン (viewPattern):
- 入力書式 (inputFormat): 数値(標準の入力書式)
- 入力パターン (inputPattern): #
- データ書式 (dataFormat): 数値8桁(前ゼロ埋め)
- データパターン (dataPattern):

At the bottom, there are buttons for 'フォルダ変更' (Change Folder), '更新' (Update), '保存' (Save), and 'メニューへ戻る' (Return to Menu).

## ② マップファイル編集ツール (2/2)

### SNAPROPDeveloper を用いて必須入力チェックを実装する手順

① Visual Studioから  
SNAPROPDeveloperを  
起動します。



② SNAPROPDeveloperでフォー  
ムマップファイルを編集します。

マッピングID	プロパティ名	プロパティ型	フォーム名	マッピングID	プロパティ名	プロパティ型	フォーム名	
LoginForm			LoginForm	LoginForm			LoginForm	
blID			LoginForm	blID			LoginForm	
tbxID	TbxID	string	LoginForm	tbxID	LoginForm_tbxID	false	true	true
blPassword			LoginForm	blPassword			true	true
tbxPassword	TbxPassword	string	LoginForm	tbxPassword	LoginForm_tbxPa...	false	true	true
btnLogin			LoginForm	btnLogin			true	true
btnExit			LoginForm	btnExit			true	true

① → 該当項目を選択します。↵

② → 入力必須フラグ(indispensable)をtrueに設定します↵

③ → 保存します↵

フォームマップ編集

マッピングID: tbxID  
プロパティ名: TbxID  
プロパティ型: string  
フォーム名: LoginForm  
コントロール名: tbxID  
ドメインID: LoginForm\_tbxID

入力必須フラグ: indispensable (true)  
表示フラグ: visible (true)  
有無効フラグ: enabled (true)  
デフォルト値: default value  
カラム値: column  
コントロールの値: control value

保存    メニューへ戻る

④ デバックします。



③ 必須チェックを行うようにタイミングをコーディングします。  
(ボタンを押したタイミングなど)



# 機能理解に役立つレッスン用 サンプルプログラムの提供

# レッスン用サンプルプログラム（1／3）

SNAPROPの機能を段階的に習得するため、レッスン用のサンプルプログラムを提供します。実装手順についてチュートリアルで細かく説明しています。

STEP1. ログイン画面を作成する

STEP2. メニュー画面を作成する

STEP3. 社員検索処理を作る(データベース処理の実装)

STEP4. 商品登録処理を作る(DBトランザクション処理の実装)

STEP5. 入力モード設定と実機でのデバック

STEP6. アプリ配布用インストーラ作成(CABファイルの作成)

STEP7. 実機での動作確認



**SNAPROPの基礎知識習得**

# レッスン用サンプルプログラム（2/3）

### ログイン画面

社員ID

パスワード

**ログイン**

**終了**

### メニュー画面

トップメニュー

**社員検索**

**商品登録**

ログアウト

### 社員検索画面

社員ID

社員名

**検索**

社員ID	社員名	入社年月
00000001	富士通太郎	200104
00000002	富士通次郎	199804
00000003	富士通花子	200504

**メニューへ**

### 商品一覧画面

**新規登録**

3件の商品が登録されています

商品ID	商品名	単価
100	りんご	100
101	みかん	80
102	バナナ	70

**メニューへ**

### 商品登録画面

商品ID

商品名

単価

**登録**

**一覧へ**

*i* 商品IDを入力してください。

### 商品編集画面

商品ID

商品名

単価

**更新**

**一覧へ**

*i* 商品名を入力してください。

## レッスン用プログラムで習得できること

- 新規にSNAPROPプロジェクトを作成する手順を習得します。
- SNAPROPのUIコントロール部品を画面に配置し、イベント処理を実装する手順を習得します。
- マップファイルを使って画面項目に必須入力チェックや文字種チェックを実装する手順を習得します。
- 画面遷移の方法を習得します。
- 画面間でパラメータを受け渡す方法を習得します。
- 作成したアプリケーションをエミュレータでデバッグする手順を習得します。
- 業務メニュー画面を作成する手順を習得します。
- データベース処理を実装する手順を習得します(検索、登録、更新)。
- マップファイルを使ったメッセージ管理方法を習得します。
- 「画面モード機能」を利用してコントロールの状態制御を行う手順を習得します。
- マップファイルを使ってデータ入力時に「初期文字入力モード」を設定する方法を習得します。
- CABファイル(インストーラ)を作成して、実機にへ転送し、インストールする手順を習得します。





## 製品価格／お問合せ

## Windows Mobile スマートフォン向け .NET アプリケーションフレームワーク

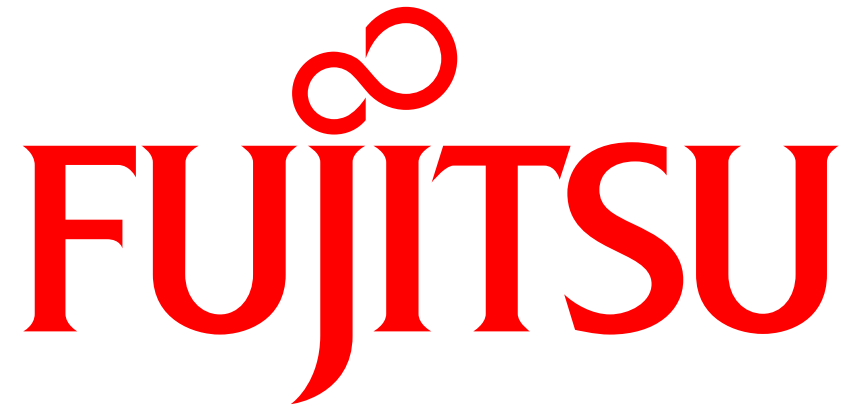
No.	商品名	ベンダ手配 コード	提供形式	提供 媒体	販売 区分	標準価格 (円)	特記事項
<b>開発キット</b>							
1-1	SNAPROP 開発キット媒体	SNAP-DEVCD	媒体のみ	CDROM	一括	10,000	
1-2	SNAPROP 開発キット 1ライセンス	SNAP-DEV-1	ライセンス	証書	一括	250,000	
<b>実行ライセンス</b>							
2-1	SNAPROP 30 端末ライセンス	SNAP-V1-30	ライセンス	証書	一括	250,000	
2-2	SNAPROP 50 端末ライセンス	SNAP-V1-50	ライセンス	証書	一括	410,000	
2-3	SNAPROP 100 端末ライセンス	SNAP-V1-100	ライセンス	証書	一括	790,000	
2-4	SNAPROP 300 端末ライセンス	SNAP-V1-300	ライセンス	証書	一括	2,300,000	
2-5	SNAPROP 500 端末ライセンス	SNAP-V1-500	ライセンス	証書	一括	3,800,000	
<b>サポートサービス</b>							
3-1	SNAPROP サポート3ヶ月 10インシデント	SNAP-SP-10	保守サポート	証書	一括	80,000	3ヶ月有効
3-2	SNAPROP サポート3ヶ月 20インシデント	SNAP-SP-20	保守サポート	証書	一括	155,000	3ヶ月有効
3-3	SNAPROP サポート3ヶ月 50インシデント	SNAP-SP-50	保守サポート	証書	一括	379,000	3ヶ月有効

### SNAPROPに関するお問合せ

☎ 電話でのお問合せ **03-5480-1185**

受付時間 9時～17時30分 (土曜・日曜・祝日・当社指定の休業日を除く)

株式会社 富士通システムズ・ウエスト  
 ビジネスソリューション本部 第二ソリューション事業部第一  
 ソリューション部  
 〒144-0035 東京都大田区南蒲田2-16-2



shaping tomorrow with you