TMS Advanced Smooth Touch Keyboard
DEVELOPERS GUIDE

June 2009
Copyright © 2009 by tmssoftware.com bvba
Web: http://www.tmssoftware.com
Email: info@tmssoftware.com

Index

TAdvSmoothTouchKeyBoard & TAdvSmoothPopupTouchKeyBoard

TAdvSmoothTouchKeyBoard description

Enhance your applications with this restyled variant of the TAdvTouchKeyBoard.

TAdvSmoothTouchKeyBoard features

Configurable on-screen keyboard for touchscreen applications with support for QWERTY, AZERTY, DVORAK, NUMERIC, CELLPHONE keyboard layouts as well as fully customizable keyboard layouts. Includes a keyboard that can be dropped on a form as well as a popup touchscreen keyboard that can optionally automatically follow focus.

- configurable keys
- automatic highlighting of Shift / Alt-Gr key values
- automatic display of shift state
- customizable background color or image for normal & down state
- popup version that can automatically follow focus control
- method to load & save keyboard layouts
- keys can have text or images
- different keys can have different colors
- sub keys when pressing a key
- autocompletion list that popups up when typing

TAdvSmoothTouchKeyBoard use

Using TAdvSmoothTouchKeyBoard is simple. The built-in keyboard layouts can be selected with the property AdvSmoothTouchKeyBoard.KeyboardType property. Drop the component on the form and when the property AutoPostKey = true, it will automatically post the keyvalue for the keyboard key pressed to the active focused control. Alternatively, if it is not desirable to have TAdvSmoothTouchKeyBoard automatically send keyboard keys, key presses can be handled through the event OnKeyDown or OnKeyClick event. The OnKeyDown event returns the key code and shift state. The OnKeyClick event returns the index of the key clicked in the TAdvSmoothTouchKeyBoard.Keys collection.

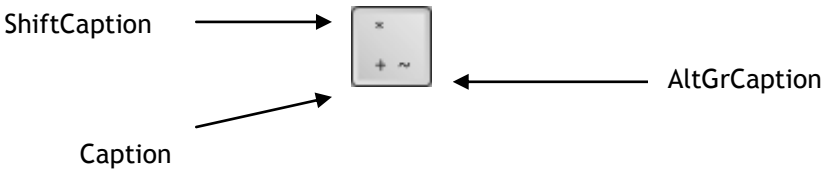
Special keys such as Shift, Alt, Ctrl, Alt Gr are sticky keys. This means that the first click on the key puts the key in down state and the second click puts the key back in normal state. The sticky behaviour makes it possible to access all key states with single clicks.
TAdvSmoothPopupTouchkeyboard works very similar except that it is shown in a popup window. To use TAdvSmoothPopupTouchkeyboard, drop it on a form and call TAdvSmoothPopupTouchkeyboard.Show to show at default position or TAdvSmoothPopupTouchkeyboard.ShowAtXY(x,y) to show at x,y screen coordinates.
TAdvPopupTouchkeyboard has the additional capability that its position can track the edit or memo control that has focus and can automatically display and hide when an edit/memo control has focus or not. To do this, set TAdvSmoothPopupTouchkeyboard.AutoFollowFocus and TAdvSmoothPopupTouchkeyboard.AutoHide to true.

Appearance

TAdvSmoothTouchkeyboard has many options to control the appearance. The background fill is set through the TAdvSmoothTouchkeyboard.fill property. The keys have a rounded advanced button look adjusted with TAdvSmoothTouchKeyItem.Color and TAdvSmoothTouchKeyItem.ColorDown. Unlike normal physical keyboards, the software touchscreen keyboard can also display keys different for Shift & Alt-Gr states. When AutoCapsDisplay is true, the normals key caption displays either in uppercase or lowercase. When HighlightCaps or HighlightAltGr is a color different from clNone, keys with multiple values depending on Shift or Alt-Gr key state will show the caption in the appropriate color. For example, if the key has the value of Euro currency (€), when Alt-Gr is pressed, the key can show € in a special color set by HighlightAltGr.

Customization

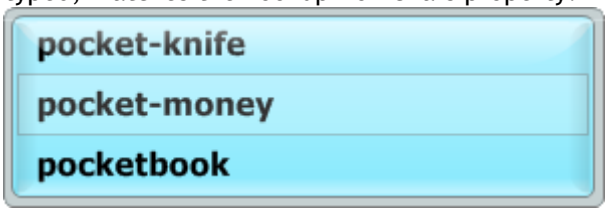
Keyboard layouts can be fully customized with TAdvSmoothTouchkeyboard. To allow this, the TAdvSmoothTouchkeyboard has a collection of TAdvSmoothTouchkeyItem objects that control position, appearance and behaviour of each key on the keyboard. A TAdvSmoothTouchKeyItem has following properties:
AltGrCaption : text on the key to display for Alt-Gr combination (bottom right of key)
AltGrKeyValue : value of key when pressed in combination with Alt-Gr key, default -1
Caption : text on the key (centered in key)
Color : color of the key in normal state
ColorDown : color of the key in down state
KeyValue : value of key when pressed in normal state, default -1
ShiftCaption : text on the key to display for Shift combination (top of key)
ShiftKeyValue : value of key when pressed in shift state, default -1
SpecialKey : sets key as special key : Caps, Ctrl, Alt, Alt-Gr, Shift, Enter, Tab, Spacebar, Return, Multiply, Divide, Delete, Subtract, Add, App, Win, Scroll, Num
TextColor : color of text on key in normal state
TextColorDown : color of text on key in down state



When SpecialKey is skNone, a key can send values to the keyboard for normal state, shift state and in combination with Alt-Gr key. When KeyValue, ShiftKeyValue, AltGrKeyValue are default -1, the key that is sent is the first letter of the text for the key state, ie. the first letter of Caption, ShiftCaption, AltGrCaption. When the KeyValue property is different from -1, the keyvalue specified is sent. For keys where SpecialKey is different from skNone, the special key value is automatically sent.

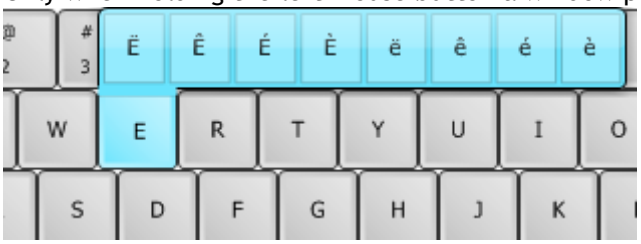
AutoCompletion

When typing, it can be useful to show a list of words that can be used to complete the sentence you are writing. The autocompletion feature has some extra properties. LookupFromChars, to set the number of characters to type before the autocompletion window pops up. LookupCaseSensitive, to search for case sensitive words when typing.
And the Mode property, to change the behaviour of the autocompletion window, whether or not it should be permanently shown, hidden, or shown when the amount of characters that are already typed, matches the LookupFromChars property.



SubKeys

This is an extra feature to allow certain Keys that are not included in the default keyboard layout to be shown. When adding characters to the SubKey collection of a single key, visually nothing will change by default.
Only when holding the left mouse button a window pops up which shows the added subkeys.



Saving and loading keyboard layouts

Keyboard layouts can be saved & loaded at any time. For this, TAdvTouchkeyboard exposes two methods:
SaveKeybLayout(FileName: string);
LoadKeybLayout(FileName: string);