

VFPX Project Overview/Proposal

Instructions: Fill out each of the following sections. Then include any supporting files for the proposed project with this document in a zip file and email it to projects@vfpx.com.

Please note that commercial products, demo versions, or any project that does not include complete source code does not fall within the scope of VFPX and licensing will not be accepted.

What is the proposed title for this project?

Dynamic Forms

What problem does this project address?

Eliminates the manual layout and placement of UI controls in the FoxPro Form Designer. Allows for dynamically rendering a display/edit form based on properties that exist on a target object.

What are the goals of this project?

To provide a programmatic/declarative syntax for defining a user input form at run time.

What programming languages or technologies will be used for/by this project?

100% VFP, with a syntax influenced by HTML/XAML layout.

Are there existing code, examples, or documents included with this project proposal? If so, please give a brief description of the files being included.

DynamicForm.prg working sample included.

Are there any existing licenses, copyrights, or trademarks that this project would be subject to? If so, please explain.

No.

Do you have any people interested in being the project manager or participating as developers on this project at this time? What type of resources are you initially anticipating?

I will be the project leader. In addition to myself, I have one tester who has already used this tool in a production app. I am hopeful that the project will attract a few other early-adopters to influence the development tool during the beta cycle.

Who should members of VFPX contact if interested in working on this project?

Matt Slay

mattslay@jordanmachine.com

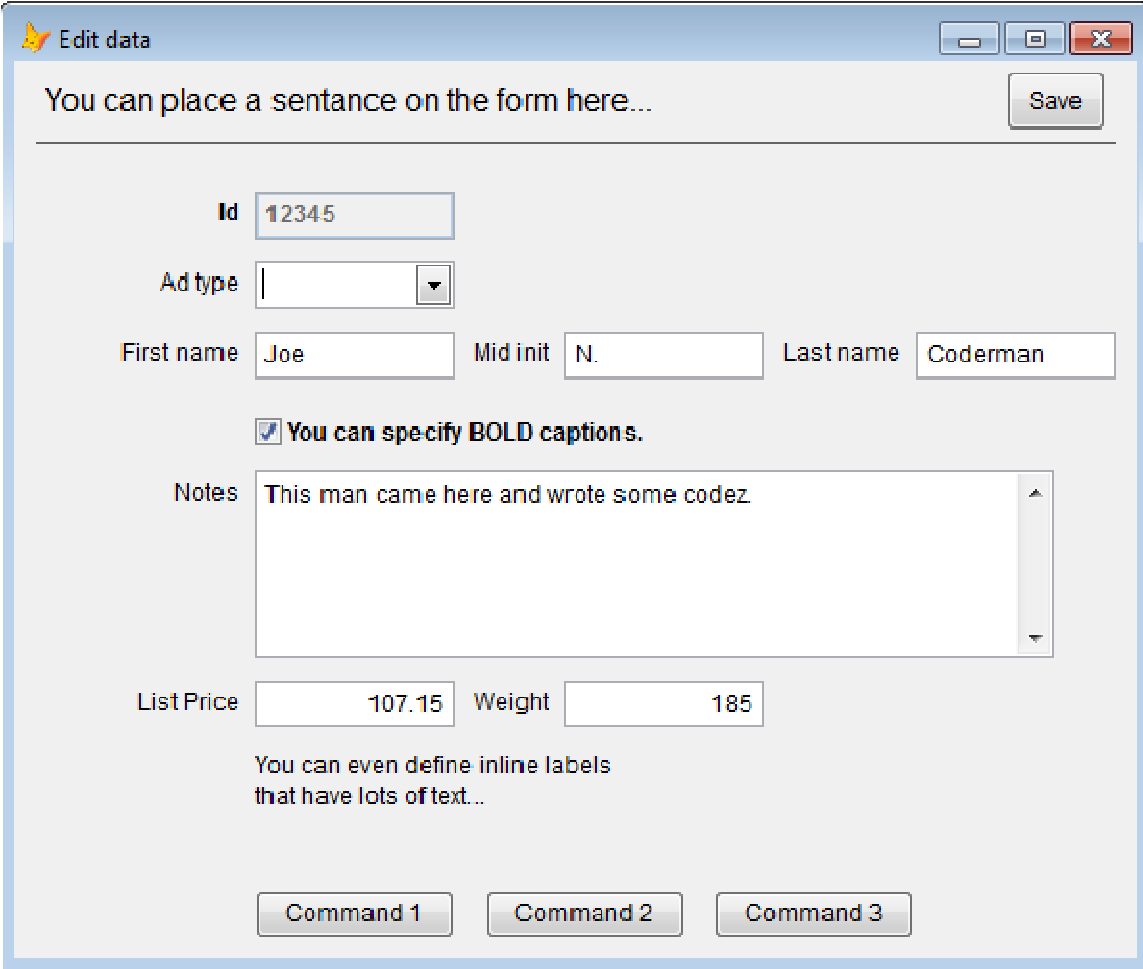
Is there an ongoing discussion out on the VFPX Message Boards regarding this project? If so, please provide the location of the thread.

Not yet.

Is there any additional information that you feel is pertinent to this project?

Screenshot... Dynamic Form sample rendered from markup code:

(See next page...)



The image shows a software window titled "Edit data" with a yellow star icon. The window has a light blue header bar with standard Windows window controls (minimize, maximize, close). Below the header, there is a large text area with the placeholder text "You can place a sentence on the form here..." and a "Save" button to its right. The main content area contains several form fields: an "Id" field with the value "12345", an "Ad type" dropdown menu, three text fields for "First name" (Joe), "Mid init" (N.), and "Last name" (Coderman). Below these is a checked checkbox labeled "You can specify BOLD captions." followed by a "Notes" text area containing the text "This man came here and wrote some codez.". Further down are two text fields for "List Price" (107.15) and "Weight" (185). Below these fields is a text label "You can even define inline labels that have lots of text...". At the bottom of the window are three buttons labeled "Command 1", "Command 2", and "Command 3".

Markup syntax sample:

```

*-- Step 2: Define the UI layout string (similar to HTML / XAML markup) -----
Text to lcFieldList PRETEXT 15 NoShow

    id      :enabled => '.f.' :FontBold => '.t.' :label_FontBold => '.t.' |
    ad_type :class => 'combobox' :RowSource => 'laOptions' :RowSourceType => '5' :name => 'cboAdType' |
    first_name |
    mid_init  :row => '0' |
    last_name :row => '0' |

    bool_3    :caption => 'You can specify BOLD captions.' :FontBold => '.t.' :width => '200' |
    notes     :class => 'editbox' :height => '100' :width => '400' |

    lnPrice   :caption => 'List Price' |
    weight    :row => '0' |

    :class => 'label' :caption => 'You can even define inline labels that have lots of text...' :width => '200' |

    :class => 'DynamicFormResultButton' :caption => 'Command 1' :Tag => "Option_1" |
    :class => 'DynamicFormResultButton' :caption => 'Command 2' :Tag => "Option_2" :row => '0' |
    :class => 'DynamicFormResultButton' :caption => 'Command 3' :Tag => "Option_3" :row => '0' |

EndText

```