



PaperVision[®] Enterprise

PaperVision E-Forms Specification

PaperVision Enterprise Release 84

Information in this document is subject to change without notice and does not represent a commitment on the part of Digitech Systems, LLC. The software described in this document is furnished under a license agreement or nondisclosure agreement. The software may be used or copied only in accordance with the terms of the agreement. It is against the law to copy the software on any medium except as specifically allowed in the license or nondisclosure agreement. No part of this manual may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording, for any purpose without the express written permission of Digitech Systems, LLC.

Copyright © 2019 Digitech Systems, LLC All rights reserved.
Printed in the United States of America.

PaperFlow, PaperFlowNOW! and the Digitech Systems, LLC logo
are trademarks of Digitech Systems, LLC

PaperVision Enterprise, ImageSilo, and PaperFlow are registered trademarks of Digitech Systems, LLC

Microsoft, Windows, SQL Server, Access, and .NET Framework are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

All other trademarks and registered trademarks are the property of their respective owners. The Microsoft Office User Interface is subject to protection under U.S. and international property laws and is used by Digitech Systems, LLC under license from Microsoft.

PaperVision Enterprise contains text search technology provided under license
by dtSearch®. All rights reserved.

PaperVision Enterprise contains technology under license
from Autonomy, Inc. All rights reserved.

PaperVision Enterprise contains portions of imaging code owned and copyrighted by Pegasus Imaging Corporation,
DBA Accusoft Pegasus, Tampa, FL. All rights reserved.



Digitech Systems, LLC

**8400 E. Crescent Parkway, Suite 500
Greenwood Village, CO 80111
Phone: 303.493.6900 Fax: 303.493.6979
www.digitechsystems.com**

Table of Contents

Chapter 1 Introduction	3
Using this Guide	3
Getting Product Support	3
Chapter 2 E-Forms Specification	4
E-Form Editor System Properties	4
E-Form Editor Properties	5
E-Form DropDownDependency Properties	5
E-Form DropDownDataDependency Properties	6
E-Form – Base - Control	7
E-Form – Accordion - Control	8
E-Form – Accordion - Item	8
E-Form - Captcha - Control	8
E-Form – Signature	8
E-Form – Tabs - Control	9
E-Form - Tab	9
E-Form - CheckBox – Control	9
E-Form - DIV (Division) - Control	10
E-Form - DropDown - Control	10
E-Form - DropDownList - Control	10
E-Form - FieldSet - Control	11
E-Form - FileInput - Control	11
E-Form - H (Heading) - Control	11
E-Form - HR (Horizontal Ruler) - Control	12
E-Form – HtmlFont - Control	12
E-Form - Hyperlink - Control	12
E-Form - Img - Control	13
E-Form - Label - Control	13
E-Form - LineBreak - Control	13
E-Form - OL (Ordered List) - Control	13
E-Form - UL (Un-Ordered List) - Control	14
E-Form - Li (List Item) - Control	14
E-Form - Paragraph - Control	14
E-Form - Password - Control	15

Table of Contents

E-Form - RadioButton - Control.....	15
E-Form - SPAN - Control.....	16
E-Form - Table - Control.....	16
E-Form - TBODY (Table Body) - Control.....	16
E-Form - TR (Table Row) - Control.....	17
E-Form - TD (Table Cell) - Control.....	17
E-Form - TextArea - Control.....	17
E-Form - TextBox - Control.....	18
E-Form – DIV (Page) - Control.....	18
Chapter 3 Custom Validation Functions.....	19
Chapter 4 Sample E-Forms Schema.....	22

Chapter 1 Introduction

Chapter 1 Introduction

Using this Guide

Welcome to the *PaperVision E-Forms Specification* guide. This document provides an overview of the various controls you can use to create E-Form definitions.

This guide is for software developers and other professionals who are familiar with developing web pages using various HTML controls. Users of this guide should also be familiar with the Microsoft® Windows operating system.

Getting Product Support

Digitech Systems takes great pride in providing software that is extremely simple to use, feature-rich, and flexible both in architecture and in integration capabilities. If, for some reason, you experience any difficulty, please do not hesitate to contact us to experience our legendary customer service toll free at (877) 374-3569 or direct at (402) 484-7777.

Additionally, you may contact us at support@digitechsystems.com. Our technical support staff is available to serve you between the hours of 8:00 AM and 6:00 PM Central Time.

Digitech Systems' product features and enhancements are driven by you, our customer. If you have an idea for a future product feature or enhancement, please send an e-mail to dev@digitechsystems.com.

In addition to our legendary technical support staff, we also offer the following resources to help answer your questions and resolve your issues.

- **MyDSI** - MyDSI is an interactive tool for all Digitech Systems customers. Log on to MyDSI at <http://mydsi.digitechsystems.com> to download product updates, license purchased software, view support contract renewals, and check the status of your software support cases and requests.
- **User Forums** - Log in to the User Forums at <http://forums.digitechsystems.com> to exchange answers and ideas with other users in our moderated community.
- **Knowledge Base** - Log in to search our extensive Knowledge Base at <http://kb.digitechsystems.com> for articles on all Digitech Systems products.
- **Documentation** - For documentation describing the use and features of the PaperVision Capture Administration Console, refer to the *PaperVision Capture Administration Guide*. Online help is also available from the Administration Console. To access the online help, click the **Help** menu, and then click **Help Topics**.

Chapter 2 E-Forms Specification

Chapter 2 E-Forms Specification

E-Form Editor System Properties

Description: The following properties are used explicitly by the E-Form engine. The engine auto populates the fields when a form is submitted.

Properties:

Name	Type	Description
EntityID	String	Entity ID that contains the E-Form
DocumentID	String	ID of the document when an E-Form is added to the project
Version	String	E-Form definition version number
SubmissionDateTime	DateTime	Date/Time when an E-Form is added to the project.
DefinitionID	Int	E-Form definition ID
SourceIP	String	Source IP of the user submitting the E-Form
InstanceID	Int	E-Form definition version database ID
Name	String	E-Form definition name
Attachments	EFormAttachment[]	Collection of E-Form attachment files added to an E-Form

E-Form Editor Properties

Description: The following properties can be changed by the user.

Properties:

Name	Type	Description
Style	String	Used to define the inline style information, such as Width and Height (e.g. Style='width:100px; height:200px;')
BackgroundImage	String	Add a background image into the E-Form main container 'div'. Note: You can point to an external image source or embed a base64 image. Each browser and version will have unique file size limitations that need to be accounted for when using this option.
DropDownDependencies	DropDownDependency[]	Collection of E-Form drop down dependencies settings. (see the DropDownDependency Properties section for more info)

E-Form DropDownDependency Properties

Description: The DropDownDependency object allows users to define relationships between two drop down controls. For example, one drop down control would contain a list of countries while a second (different) drop down control would contain a list of states.

Properties:

Name	Type	Description
ParentID	String	Represents the primary DropDown ID, which can trigger a selection change event that populates a different dependent drop down
ChildID	String	Represents the dependent DropDown ID
ChildParentData	DropDownDataDependency[]	Collection of 'DropDownDataDependency' objects (see the DropDownDataDependency Properties section for more info)

E-Form DropDownDataDependency Properties

Description: The DropDownDataDependency object represents data that will populate the child drop down list which is based on the parent drop down selected value.

Properties:

Name	Type	Description
Value	String	Specifies the value of the dependent drop down item
Text	String	Specifies the text displayed in the dependent drop down item
RelationID	String	Specifies the Parent drop down item ID, which ties the dependency record item to the Parent drop down item using a one-to-many relationship

E-Form – Base - Control

Description: EFormBaseControl is a base class that contains properties shared by all controls.

Properties:

Name	Type	Description
Style	String	Used to define the inline style information, such as Width and Height (e.g. Style='width:100px; height:200px;')
Name	String	Specifies the name of an <input> element. Used to reference elements in a JavaScript, or to reference form data after a form is submitted. Note: Only form elements with a name attribute will have their values passed when submitting a form.
ID	String	Specifies a unique ID for an HTML element (the value must be unique within the HTML document). Used most to point to a style in a style sheet, and by JavaScript (via the HTML DOM) to manipulate the element with the specific ID.
IsVisible	Boolean	Indicates whether an element is displayed or not. Note: The parent element must be visible for a child element to be visible. The user permission UI uses this property to persist group element permissions; therefore it will override any inline value.
Tooltip	String	Specifies extra info about a specific element (e.g. tooltip)
Class	String	Specifies one or more class names for an element. To specify multiple classes, separate the class names with a space, e.g. . This allows a user to combine several CSS classes for one HTML element. Class naming rules: <ul style="list-style-type: none"> • Must begin with a letter (A-Z or a-z) • Can be followed by: letters (A-Za-z), digits (0-9), hyphens (“-“), and underscored (“_”). • All values are case-insensitive
IndexName	String	Reserved for future use.
HostTopOffset	Int	Reserved for use by the E-Forms Designer
HostLeftOffset	Int	Reserved for use by the E-Forms Designer

E-Form – Accordion - Control

Description: Displays collapsible content panels for presenting information in a limited amount of space.

Properties:

Name	Type	Description
Items	AccordionItem[]	Collection of collapsible panels

E-Form – Accordion - Item

Description: Displays a single collapsible content panel

Properties:

Name	Type	Description
Header	String	Represents the header section of a collapsible panel
Controls	EformControlBase[]	Collection of E-Form controls

E-Form - Captcha - Control

Description: The Captcha control is used to secure your site from random automated scripts and to ensure the form is submitted by human interaction.

Properties:

Name	Type	Description
TabIndex	String	Specifies the tab order of an element when the tab button is used for navigation

E-Form - Signature - Control

Description: Allows a user to create a signature using a mouse or touch device

Properties:

Name	Type	Description
Width	Int	Represents the width of the signature pad canvas
Height	Int	Represents the height of the signature pad canvas
Value	String	Used by the system to store the signature created
Required	Boolean	Specifies that a user must make a selection
ValidationMessage	String	Customizes the validation message shown to the user

E-Form – Tabs - Control

Description: A single content area with multiple panels; each one is associated with a header in a list.

Properties:

Name	Type	Description
Items	Tab	Collection of panels
TabMode	TabsMode	Represents the orientation of the tab header (Values: Top, Bottom, Vertical)

E-Form - Tab - Control

Description: Displays a single content panel with a header

Properties:

Name	Type	Description
Header	String	Represents the header area of the tab section
Controls	EformControlBase[]	Collection of E-Form controls. You can add multiple nested controls using this property

E-Form - CheckBox – Control

Description: The CheckBox control is mapped directly to an HTML <input> tag (type = ‘text’) which lets a user select one or more options from a limited number of choices.

Properties:

Name	Type	Description
IsChecked	String	Sets or returns the checked state of a checkbox
Value	String	Sets or returns the value of the checkbox
CustomValidation	String	Allows a user to define advanced validation; such as the min/max number of check boxes that must be selected or validation will fail. Example: Min2=> validate[minCheckbox[2]] checkbox Max2=> validate[maxCheckbox[2]] checkbox (See Custom Validation Functions for more information)
Required	Boolean	Specifies that a user must make a selection
ValidationMessage	String	Customizes the validation message shown to the user
TabIndex	String	Defines the tab order of an element when the tab button is used for navigation

E-Form - DIV (Division) - Control

Description: The DIV control is mapped directly to an HTML <div> tag used to define a division (or a section) in an HTML document. This tag will group block-elements to format them with a CSS.

Properties:

Name	Type	Description
Controls	EformBaseControl[]	Collection of E-Form controls

E-Form - DropDown - Control

Description: The DropDown control is mapped directly to an HTML <select> tag which is used to create a drop-down list.

Properties:

Name	Type	Description
OnChange	String	Reserved for future use
Source	String	Reserved for future use
Items	DropDownItem	Collection of drop-down items control
Required	Boolean	Specifies that a user must select an option
ValidationMessage	String	Customizes the validation message shown to the user
TabIndex	String	Defines the tab order of an element when the tab button is used for navigation

E-Form - DropDownItem - Control

Description: The DropDownItem control is mapped directly to an HTML <option> tag that is used to define options for a select list.

Properties:

Name	Type	Description
Text	String	Specifies the text used for an option. (Note: This value is displayed to the user)
Value	String	Specifies the value to be sent to a server
IsSelected	Boolean	Specifies that an option should be pre-selected when the page loads

E-Form - FieldSet - Control

Description: The FieldSet control is mapped directly to an HTML <fieldset> tag which is used to group related elements in a form. The <fieldset> tag is used to draw a box around related elements.

Properties:

Name	Type	Description
Controls	EformBaseControl[]	Collection of E-Form controls. You can add multiple nested controls using this property
Legend	String	Defines a caption for the <fieldset> element

E-Form - FileInput - Control

Description: The FileInput control defines a file-select field and a “Browse...” button (for file uploads). This control allows users to upload multiple files and restricts uploaded files.

Properties:

Name	Type	Description
Value	String	Used by the system
Required	Boolean	Specifies whether a user must attach a file to the form
ValidationMessage	String	Customizes the message shown to the user
AllowedExtensions	String	Used when only certain file extensions are to be selected. Separate valid extensions using a “ ” character (e.g. “jpg gif png”)
MaxNumberFiles	Int	Used to limit the number of files selected (attached)
TabIndex	String	Defines the tab order of an element when the tab button is used for navigation

E-Form - H (Heading) - Control

Description: The H control is mapped directly to an HTML <h1,2,3,4,5, or 6> tag used to define HTML headings. The <h1> tag defines the most important heading. The <h6> tag defines the least important heading.

Properties:

Name	Type	Description
Text	String	Specifies the text shown by the heading tag
Number	String	Specifies the importance of the heading (values range from 1 – 6)

E-Form - HR (Horizontal Ruler) - Control

Description: The HR control is mapped directly to an HTML <hr> tag which defines a thematic break in an HTML page (e.g. a shift of topic). Used to separate content (or define a change) in an HTML page.

Properties: N/A

E-Form – HtmlFont - Control

Description: The HtmlFont control is mapped directly to an HTML tag which specifies the font face, font size, and font color of text.

Properties:

Name	Type	Description
Text	String	Specifies a text value
Size	String	Specifies the size of the text
Controls	EformBaseControl[]	Collection of E-Form controls. You can add multiple nested controls using this property

E-Form - Hyperlink - Control

Description: The Hyperlink control is mapped directly to an HTML <a> tag used to define a hyperlink that will link one page to another. The most important attribute of the <a> element is the ‘href’ attribute, which indicates the link’s destination. By default, links will appear in all browsers as follows:

- An unvisited link is underlined and blue
- A visited link is underlined and purple
- An active link is underlined and red

Properties:

Name	Type	Description
Text	String	Specifies the text shown as a hyperlink
Href	String	Specifies the URL of the page the link goes to
Target	String	Specifies where to open the linked document (e.g. _blank, _parent, _self, _top)

E-Form - Img - Control

Description: The Img control is mapped directly to an HTML tag which allows users to define an image. The tag defines an image in an HTML page. Note: Images are not technically inserted into an HTML page, images are linked to HTML pages. The tag creates a holding space for the referenced image.

Properties:

Name	Type	Description
AlternateText	String	Specifies an alternate text for an image
Source	String	Specifies the URL of an image

E-Form - Label - Control

Description: The Label control is mapped directly to an HTML <label> tag which can be used to specify short text using this control.

Properties:

Name	Type	Description
Text	String	Sets the text shown by the label control

E-Form - LineBreak - Control

Description: The LineBreak control is mapped directly to an HTML
 tag which is used to insert a single line break.

Properties: N/A

E-Form - OL (Ordered List) - Control

Description: The OL control is mapped directly to an HTML tag which defines an ordered list. An ordered list can be numerical or alphabetical.

Properties:

Name	Type	Description
Items	Li[]	Collection of list items

E-Form - UL (Un-Ordered List) - Control

Description: The UL control is mapped directly to an HTML tag which defines an un-ordered (bulleted) list.

Properties:

Name	Type	Description
Items	Li[]	Collection of list items

E-Form - Li (List Item) - Control

Description: The Li control is mapped directly to an HTML tag used to define a list item. This tag is used in both ordered or un-ordered lists.

Properties:

Name	Type	Description
Controls	EformBaseControl[]	Collection of E-Form controls. You can add multiple nested controls using this property
Text	String	Specifies raw text

E-Form - Paragraph - Control

Description: The Paragraph control is mapped directly to an HTML <p> tag which is used to define a paragraph. Browsers automatically add some space (margin) before and after each <p> element. The margins can be modified with a CSS (using the margin properties).

Properties:

Name	Type	Description
Text	String	Specifies the text displayed in the Paragraph control

E-Form - Password - Control

Description: The Password control is mapped directly to an HTML <input> tag (type = 'password') which is used to represent a single-line password field. The contents of the password field are masked (appears as blobs or asterisks) in the browser.

Properties:

Name	Type	Description
Text	String	Sets or returns the value
Required	Boolean	Specifies that a user must populate the control
TabIndex	String	Defines the tab order of an element when the tab button is used for navigation
ValidationMessage	String	Customizes the validation message shown to the user
CustomValidation	String	Allows a user to define advanced validation options that are built into the system. (See Custom Validation Functions for more information)

E-Form - RadioButton - Control

Description: The RadioButton control is mapped directly to an HTML <input> tag (type = 'radio') which allows a user to select only one of a limited number of choices. Note: To group multiple radio buttons, give all the radio buttons the same name in the 'Name' property field.

Properties:

Name	Type	Description
IsChecked	String	Specifies that the radio button will be enabled automatically
Text	String	Displays a label before the radio button, including a value for this property
Value	String	Value passed to the code when the form is submitted
Required	Boolean	Specifies that a user must select a single choice
ValidationMessage	String	Customizes the validation message shown to the user
CustomValidation	String	Allows a user to define advanced validation options that are built into the system. (See Custom Validation Functions for more information)
TabIndex	String	Defines the tab order of an element when the tab button is used for navigation

E-Form - SPAN - Control

Description: The SPAN control is mapped directly to an HTML tag which is used to group inline-elements in a document. The tag provides no visual change by itself. This tag provides a way to add a hook to an area of text or an area of a document.

Properties:

Name	Type	Description
Controls	EformBaseControl[]	Collection of E-Form controls. You can add multiple nested controls using this property
Text	String	Specifies raw text

E-Form - Table - Control

Description: The Table control is mapped directly to an HTML <table> tag. An HTML table consists of the <table> element and one or more <tr>, <th>, and <td> elements. The <tr> element defines a table row, the <th> element defines a table header, and the <td> element defines a table cell.

Properties:

Name	Type	Description
TBODY	TBODY	Used to group the body content in an HTML table
Items	TR[]	Collection of <tr> tags
Caption	String	Defines a table caption. Only one caption can be specified per table.

E-Form - TBODY (Table Body) - Control

Description: The TBODY control is mapped directly to an HTML <tbody> tag which is used to group the body content in an HTML table. A tbody must have one or more <tr> tags inside.

Properties:

Name	Type	Description
Items	TR[]	Collection of <tr> tags

E-Form - TR (Table Row) - Control

Description: The TR control is mapped directly to an HTML <tr> tag which defines a row in an HTML table. A <tr> tag must have one or more <td> elements.

Properties:

Name	Type	Description
Items	TD[]	Collection of <td> tags

E-Form - TD (Table Cell) - Control

Description: The TD control is mapped directly to an HTML <td> tag which defines a standard cell in an HTML table. The <td> tag can contain one or more E-Form controls.

Properties:

Name	Type	Description
Controls	EformBaseControl[]	Collection of E-Form controls. You can add multiple nested controls using this property
ColSpan	String	Specifies the number of columns a cell should span
RowSpan	String	Specifies the number of rows a cell should span
Text	String	Specifies the text

E-Form - TextArea - Control

Description: The TextArea control is mapped directly to an HTML <textarea> tag which defines a multi-line text input control. A text area can hold an unlimited number of characters, and renders in a fixed-width font (e.g. Courier). The size of a text area can be specified by the cols and rows attributes, or through the CSS' height and width properties.

Properties:

Name	Type	Description
Column	String	Specifies the visible width of a text area
Rows	String	Specifies the visible number of lines in a text area
Text	String	Specifies the text
Required	Boolean	Specifies that a text area is required (must be filled out)
ValidationMessage	String	Customizes the message shown to the user
TabIndex	String	Specifies the tabbing order of the element. If blank, the control will dictate the tab order

E-Form - TextBox - Control

Description: The TextBox control is mapped directly to an HTML <input> tag (type = 'text') which defines a one-line input field that a user can enter text into. Note: The textbox control can be used as a calendar picker by setting the 'ValidationMode' property to date.

Properties:

Name	Type	Description
Mask	String	Formats user input automatically
Text	String	Specifies the default value displayed in the text box
CustomValidation	String	Allows a user to define advance validation that is built into the system using the functions described in Custom Validation Functions
Required	Boolean	Specifies that a user must enter text.
ValidationMessage	String	Customizes the validation message shown to the user
ValidationMode	String	Predefined regular expression validation function (e.g. Phone, URL, Email, Date, Number, Integer, OnlyLetter, OnlyLetterNumber)
TabIndex	String	Defines the tab order of an element when the tab button is used for navigation
DateFormat	String	Allows a user to customize the date format displayed from the calendar control. This option only applies when the 'ValidationMode' property is set to 'date'

E-Form – DIV (Page) - Control

Description: This control is mapped directly to an HTML <div> tag used to define a division (section or page) in an HTML document. This tag is used by the E-Forms Designer to group controls into printable pages. By default, a page size is automatically set to 8 ½ X 11.

Properties:

Name	Type	Description
Controls	EformBaseControl[]	Collection of E-Form controls

Chapter 3 Custom Validation Functions



Chapter 3 Custom Validation Functions

Custom Validation Functions:

Name	Syntax	Description
Predefined Regex	validate[required,custom[url]]	Validates if the user input is a URL
Equals	validate[required,equals[password]]	Validates if two controls have the same value. Note: password is the control 'ID' to compare to
Function	validate[required,funCall[checkHELLO]]	Executes user-defined custom javascript functions
Conditional required	validate[condRequired[dep1]]	Field is required if field ID 'dep1' has a value
Conditional required	validate[condRequired[dep2a,dep2b]]	Field is required if either the control id 'dep2a' or dep2b' contains a value
MinSize	validate[required,minSize[6]]	Specifies the minimum characters allowed
MaxSize	validate[required,maxSize[6]] - or - validate[maxSize[6]]	Specifies the maximum characters allowed. Note this field will allow blank values unless [required] is specified.
Min	validate[required,custom[integer],min[-5]]	Specifies the value entered must be equal or greater than the predefined value (-5)
Max	validate[required,custom[integer],max[50]]	Specifies the value entered must be equal or less than the predefined value (50)
Past	validate[custom[date],past[2010/01/01]]	Validate that the value is a date in the past.

Future	<pre>validate[custom[date],future[20/01/01]]</pre> <pre>validate[custom[date],future[NOW]]</pre>	<p>Validate that the value is a date in the future. (e.g. Please enter a date later than today's date)</p>
Group Required	<pre>validate[groupRequired[payments]]</pre>	<p>Specifies at least one of the input fields for the 'group required' option is not empty or invalid.</p> <p>Note: all input fields must be grouped together with 'groupRequired'</p>
Date Range	<pre>validate[dateRange[grp1]]</pre>	<p>Validates that the start date is before the end date.</p> <p>Note: both fields must have the same name 'grp1'</p>
Date Time Range	<pre>validate[dateTimeRange[grp2]]</pre>	<p>Validates that the start date and time are before the end date and time.</p> <p>Note: Both fields must have the same name 'grp2'</p>
Date Compare	<pre>validate[custom[date],past[#dateCmp2]]</pre> <pre>validate[custom[date],future[#dateCmp1]]</pre>	<p>Validates that the first date is before the second date.</p> <p>Note: dateCmp2 & dateCmp1 are field IDs</p>
Credit Card	<pre>validate[required,creditCard]</pre>	<p>Validates that the credit card number is valid, per the <u>Luhn checksum algorithm</u>. Note: since credit cards are often presented in different formats, spaces and hyphens are ignored.</p>

Masking Sample

- **a** - Represents an alpha character (A-Z, a-z)
- **9** - Represents a numeric character (0-9)
- ***** - Represents an alphanumeric character (A-Z,a-z,0-9)

Date	99/99/9999
Phone	(999) 999-9999
Phone + Ext	(999) 999-9999?X99999
Tax ID	99-9999999
SSN	999-99-9999
Product Key	a*-999-a999
Eye Script	~9.99~9.99 999

Chapter 4 Sample E-Forms Schema



Chapter 4 Sample E-Forms Schema

Sample E-Form Schema:

```
<?xml version="1.0"?>
<xs:schema elementFormDefault="qualified" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="Eform" nillable="true" type="Eform" />
  <xs:complexType name="Eform">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="EntityID" type="xs:string" />
      <xs:element minOccurs="0" maxOccurs="1" name="DocumentID" type="xs:string" />
      <xs:element minOccurs="0" maxOccurs="1" name="Style" type="xs:string" />
      <xs:element minOccurs="0" maxOccurs="1" name="Name" type="xs:string" />
      <xs:element minOccurs="0" maxOccurs="1" name="Version" type="xs:string" />
      <xs:element minOccurs="1" maxOccurs="1" name="CreationDateTime" nillable="true" type="xs:dateTime"
/>
      <xs:element minOccurs="1" maxOccurs="1" name="SubmissionDateTime" nillable="true"
type="xs:dateTime" />
      <xs:element minOccurs="1" maxOccurs="1" name="DefinitionID" type="xs:int" />
      <xs:element minOccurs="0" maxOccurs="1" name="SourceIP" type="xs:string" />
      <xs:element minOccurs="1" maxOccurs="1" name="InstanceID" type="xs:int" />
      <xs:element minOccurs="0" maxOccurs="1" name="BackgroundImage" type="xs:string" />
      <xs:element minOccurs="0" maxOccurs="1" name="OnLoad" type="xs:string" />
      <xs:element minOccurs="0" maxOccurs="1" name="DropDownDependencies"
type="ArrayOfDropDownDependency" />
      <xs:element minOccurs="0" maxOccurs="1" name="Controls" type="ArrayOfControl" />
      <xs:element minOccurs="0" maxOccurs="1" name="Attachments" type="ArrayOfEFormAttachment" />
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="ArrayOfDropDownDependency">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="DropDownDependency" nillable="true"
type="DropDownDependency" />
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="DropDownDependency">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="ParentID" type="xs:string" />
      <xs:element minOccurs="0" maxOccurs="1" name="ChildID" type="xs:string" />
      <xs:element minOccurs="0" maxOccurs="1" name="ChildParentData"
type="ArrayOfDropDownDataDependency" />
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="ArrayOfDropDownDataDependency">
    <xs:sequence>
```



```

</xs:complexType>
<xs:complexType name="DropDownDataDependency">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Value" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="Text" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="RelationID" type="xs:string" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="ArrayOfControl">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="Control" nillable="true" type="Control" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="Control" abstract="true">
  <xs:sequence>
    <xs:element minOccurs="1" maxOccurs="1" name="HostTopOffset" type="xs:int" />
    <xs:element minOccurs="1" maxOccurs="1" name="HostLeftOffset" type="xs:int" />
    <xs:element minOccurs="0" maxOccurs="1" name="Styles" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="Name" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="ID" type="xs:string" />
    <xs:element minOccurs="1" maxOccurs="1" name="IsVisible" type="xs:boolean" />
    <xs:element minOccurs="0" maxOccurs="1" name="Tooltip" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="Class" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="IndexName" type="xs:string" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="Signature">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="1" maxOccurs="1" name="Width" type="xs:int" />
        <xs:element minOccurs="1" maxOccurs="1" name="Height" type="xs:int" />
        <xs:element minOccurs="0" maxOccurs="1" name="Value" type="xs:string" />
        <xs:element minOccurs="1" maxOccurs="1" name="Required" type="xs:boolean" />
        <xs:element minOccurs="0" maxOccurs="1" name="ValidationMessage" type="xs:string" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="Accordion">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Items" type="ArrayOfAccordionItem" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

```

</xs:complexContent>
</xs:complexType>
<xs:complexType name="ArrayOfAccordionItem">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="AccordionItem" nillable="true"
type="AccordionItem" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="AccordionItem">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Header" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="Controls" type="ArrayOfControl" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="Page">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Controls" type="ArrayOfControl" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="HtmlFont">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Text" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="Size" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="Controls" type="ArrayOfControl" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="Label">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Text" type="xs:string" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>

```

```

</xs:complexType>
<xs:complexType name="Password">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Text" type="xs:string" />
        <xs:element minOccurs="1" maxOccurs="1" name="Required" type="xs:boolean" />
        <xs:element minOccurs="0" maxOccurs="1" name="ValidationMessage" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="CustomValidation" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="TabIndex" type="xs:string" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="SPAN">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Text" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="Controls" type="ArrayOfControl" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="TextArea">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Columns" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="Rows" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="Text" type="xs:string" />
        <xs:element minOccurs="1" maxOccurs="1" name="Required" type="xs:boolean" />
        <xs:element minOccurs="0" maxOccurs="1" name="ValidationMessage" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="TabIndex" type="xs:string" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="UL">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Items" type="ArrayOfLI" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>

```

```

</xs:complexType>
<xs:complexType name="ArrayOfLI">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="Li" nillable="true" type="Li" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="Li">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Text" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="Controls" type="ArrayOfControl" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="H">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Text" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="Number" type="xs:string" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="HR">
  <xs:complexContent mixed="false">
    <xs:extension base="Control" />
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="DIV">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Controls" type="ArrayOfControl" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="FieldSet">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Legend" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="Controls" type="ArrayOfControl" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

```
</xs:sequence>
  </xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="FileInput">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Value" type="xs:string" />
        <xs:element minOccurs="1" maxOccurs="1" name="Required" type="xs:boolean" />
        <xs:element minOccurs="0" maxOccurs="1" name="ValidationMessage" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="TabIndex" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="AllowedExtensions" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="MaxNumberFiles" type="xs:string" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="Captcha">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="TabIndex" type="xs:string" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="RadioButton">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="1" maxOccurs="1" name="IsChecked" type="xs:boolean" />
        <xs:element minOccurs="0" maxOccurs="1" name="Text" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="Value" type="xs:string" />
        <xs:element minOccurs="1" maxOccurs="1" name="Required" type="xs:boolean" />
        <xs:element minOccurs="0" maxOccurs="1" name="ValidationMessage" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="CustomValidation" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="TabIndex" type="xs:string" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="DropDown">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
```

```

    <xs:element minOccurs="0" maxOccurs="1" name="OnChange" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="Source" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="Items" type="ArrayOfDropDownItem" />
    <xs:element minOccurs="1" maxOccurs="1" name="Required" type="xs:boolean" />
    <xs:element minOccurs="0" maxOccurs="1" name="ValidationMessage" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="TabIndex" type="xs:string" />
  </xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="ArrayOfDropDownItem">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="DropDownItem" nillable="true"
type="DropDownItem" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="DropDownItem">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Text" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="Value" type="xs:string" />
    <xs:element minOccurs="1" maxOccurs="1" name="IsSelected" type="xs:boolean" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="CheckBox">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="1" maxOccurs="1" name="IsChecked" type="xs:boolean" />
        <xs:element minOccurs="0" maxOccurs="1" name="Value" type="xs:string" />
        <xs:element minOccurs="1" maxOccurs="1" name="Required" type="xs:boolean" />
        <xs:element minOccurs="0" maxOccurs="1" name="ValidationMessage" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="CustomValidation" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="TabIndex" type="xs:string" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="HiddenField">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Value" type="xs:string" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

```
<xs:complexType name="Button">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Value" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="TabIndex" type="xs:string" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="TextBox">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Mask" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="Text" type="xs:string" />
        <xs:element minOccurs="1" maxOccurs="1" name="Required" type="xs:boolean" />
        <xs:element minOccurs="0" maxOccurs="1" name="ValidationMode" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="CustomValidation" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="ValidationMessage" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="DateFormat" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="TabIndex" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="DataFormat" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="DataFormula" type="xs:string" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="OL">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Items" type="ArrayOfLI" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="Img">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="AlternateText" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="Source" type="xs:string" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

```
</xs:complexType>
<xs:complexType name="LineBreak">
  <xs:complexContent mixed="false">
    <xs:extension base="Control" />
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="Paragraph">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Text" type="xs:string" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="Hyperlink">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Text" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="Href" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="Target" type="xs:string" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="TR">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Text" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="Items" type="ArrayOfTD" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="ArrayOfTD">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="TD" nillable="true" type="TD" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="TD">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Text" type="xs:string" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```



```
<xs:element minOccurs="0" maxOccurs="1" name="ColSpan" type="xs:string" />
<xs:element minOccurs="0" maxOccurs="1" name="RowSpan" type="xs:string" />
<xs:element minOccurs="0" maxOccurs="1" name="Controls" type="ArrayOfControl" />
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="TBODY">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Items" type="ArrayOfTR" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="ArrayOfTR">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="TR" nillable="true" type="TR" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="TABLE">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Caption" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="TBODY" type="TBODY" />
        <xs:element minOccurs="0" maxOccurs="1" name="Items" type="ArrayOfTR" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="Tab">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Header" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="Controls" type="ArrayOfControl" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="Tabs">
  <xs:complexContent mixed="false">
    <xs:extension base="Control">
      <xs:sequence>
```

```

    <xs:element minOccurs="1" maxOccurs="1" name="TabMode" type="TabsMode" />
    <xs:element minOccurs="0" maxOccurs="1" name="Items" type="ArrayOfTab" />
  </xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:simpleType name="TabsMode">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Top" />
    <xs:enumeration value="Bottom" />
    <xs:enumeration value="Vertical" />
  </xs:restriction>
</xs:simpleType>
<xs:complexType name="ArrayOfTab">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="Tab" nillable="true" type="Tab" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="ArrayOfEFormAttachment">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="EFormAttachment" nillable="true"
type="EFormAttachment" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="EFormAttachment">
  <xs:sequence>
    <xs:element minOccurs="1" maxOccurs="1" name="ProjectID" type="xs:int" />
    <xs:element minOccurs="0" maxOccurs="1" name="ProjectName" type="xs:string" />
    <xs:element minOccurs="1" maxOccurs="1" name="DocID" type="xs:int" />
    <xs:element minOccurs="0" maxOccurs="1" name="FileName" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="Url" type="xs:string" />
  </xs:sequence>
</xs:complexType>
</xs:schema>

```

