chisimba cms

Form Builder

User Manual

Date: 9 November 2010

Written By:
Salman Noor

Revisions	*Revised by*	*Date*	*Accepted*	*Description*
0.1	Salman Noor	10 November 2010		Draft

Table of Contents

List of Figures and Tables

List of Abbreviations, Acronyms and Symbols

Abbreviations, Acronyms and Symbols	**Meaning**
WYSIWYG	What You See Is What You Get
HTML	Hyper Text Markup Language
URL	Uniform Resource Locator
URI	Uniform Resource Indicator

Acknowledgments

This module and its documentation has been solely designed, developed, tested, evaluated and documented by Salman Noor who is currently a student belonging to the School of Electrical and Information Engineering. The design constraints, specifications and requirements for this module were stringently stipulated by Armond Furmie and Ary Van Der Pijl who are both senior software and applications developers and designers.

1. Introduction

Form Builder is a standalone Chisimba module that permits users to create HTML forms through a user friendly WYSIWYG interface without having to code anything. This module surrenders control over to the user and provide over eleven different types of form elements; each one that can be built and configured differently.

Once users have built their forms, the form builder can construct it in any location, module or page depending on the user's volition. Form submissions can be e-mailed to the user, saved in the form builder database for future viewing or both.

Additionally, publishing options are offered. Users are also given control over what actions are to be preformed once their form is submitted. This module is designed to be powerful and concurrently user friendly.

2. Brief Module Flow

2.1.Discrepancy between Users and Designers

It is imperative not to confuse and conflate the designers and users of the form. These are two separate entities as explicitly described below.

- *Designers:* The designers of the form are the readers of this document who want to create forms for their users. The designers will build the forms with the desired form elements and configure them according to their needs.
- *Users:* The users of forms are the individuals who fill in the forms created by the designers and submit them.

2.2.Basic Module Flow

The form builder module is separated into four major entities. These entities succeed each other in solely one direction.

The first step in creating a form is to insert the form's metadata. The metadata of the form can be loosely described as the information about the form such as the form's name and description. Additionally, the designer's name and email address are also considered as metadata for the form. Once the metadata of the form gets stored, the actual form can be built.

The WYSIWYG form editor is the most integral part of the form builder that allows designers to build the actual form [1]. Designers can insert, edit, rearrange and delete form elements according to their needs. When users are satisfied with the contents of their actual form, it needs to be finalized by publishing it.

Once the form is built in the WYSIWYG form editor, it cannot be rendered for users of the form to submit unless it gets published [2]. The form needs published so it can be used. Besides publishing, the designers of the form can select a set of actions to perform after submissions of their forms occur. Once designers have published their form, it is ready to be used.

The form listings consist of all the forms that have been created with the form builder module. Designers can search for their forms in the form listings [3]. All possible actions for each form are provided. Designer's can edit their existing form contents, publishing details and metadata. Additionally, form contents and publishing settings can be previewed. Saved submit results of the form can be viewed. The designer can also delete all for submissions or contents completely. A brief schematic of these four entities is illustrated in figure 1.

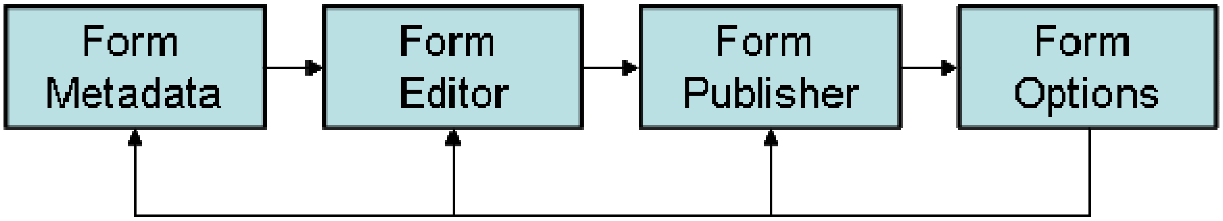


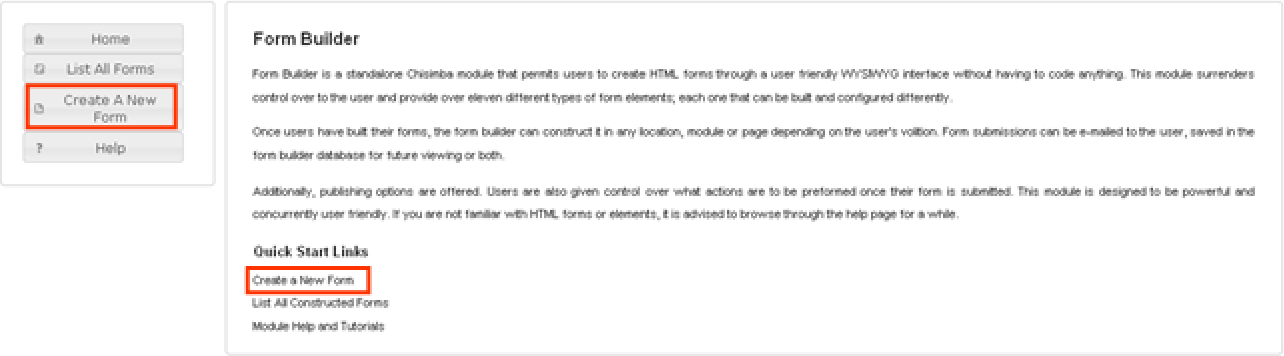
Figure 1: A simple block diagram of the major module entities and the flow relationship with each other.

When a designer first creates a form, the work flow of figure 1 has to be stringently followed. The user has to insert the form's metadata followed by actually building the form and finally publishing it for use. Once the form has been successfully created and this flow has been followed, designers can edit any part of the form in the form listings under form options.

3. Form Metadata

3.1. Create a New Form

The color themes and design of this module might possibly differ. However, the main functionality will remain the same. To create a new form, click on the links on the home page of the form builder module depicted in figure 2 inside the orange rectangles.



Form Builder

Form Builder is a standalone Chisimba module that permits users to create HTML forms through a user friendly WYSIWYG interface without having to code anything. This module surrenders control over to the user and provide over eleven different types of form elements; each one that can be built and configured differently.

Once users have built their forms, the form builder can construct it in any location, module or page depending on the user's volition. Form submissions can be e-mailed to the user, saved in the form builder database for future viewing or both.

Additionally, publishing options are offered. Users are also given control over what actions are to be preformed once their form is submitted. This module is designed to be powerful and concurrently user friendly. If you are not familiar with HTML forms or elements, it is advised to browse through the help page for a while.

Quick Start Links

Create a New Form
List All Constructed Forms
Module Help and Tutorials

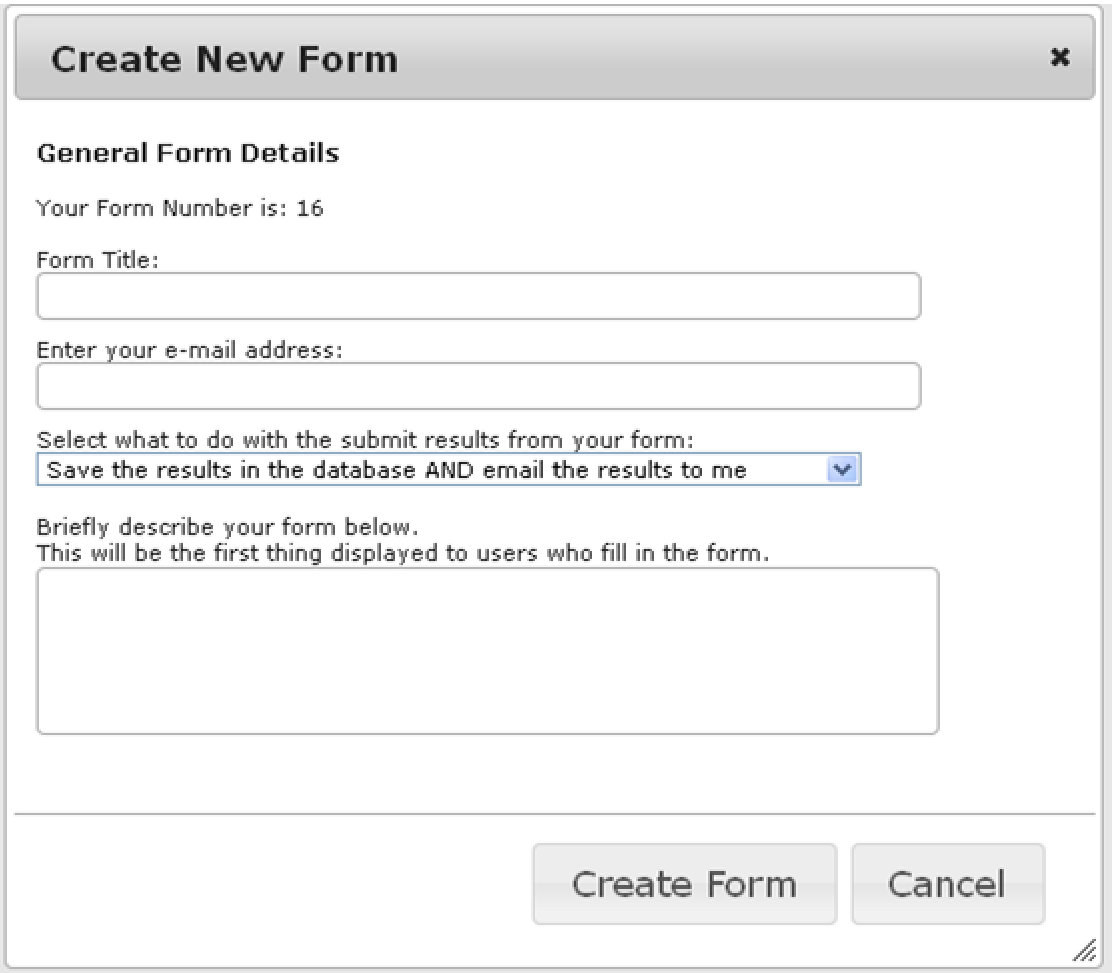
Figure 2: Home page of the form builder module is shown.

Upon clicking the new form link, a modal window depicted in figure 3 will appear containing all the form metadata that is required to be inserted. A modal window is a special dialog menu that prevents users from interacting with elements outside this window until this window been closed [4]. The form number indicator displays the rank of the form. For instance, according to figure 3 this will be the 16th form being created by the form builder module. Additionally, this number will also help in the finding of this form in the form listings as it will be displayed 16th on the list.

3.2. Form Submission Actions

The form title and description is self explanatory information about the designer's form. The drop down consists of three options that are explicated below.

- *“Save submit results in a database which can be view anytime”:* If this option is selected, all user submissions of the designer’s form will be stored in the form builder database. The submission results can be viewed at anytime.
- *“Email the results to email address entered above”:* If this selection is selected, the results of all form submissions will be emailed to the designer of the form to the email address inserted in the relevant text field above the drop down list. No submit results will be saved in the form builder database for future viewing. The submit results of each form submission will be sent in HMTL format.
- *“Save the results in the database AND email the results to me”*: After each form submission, both the first two options will be executed.


Create New Form

General Form Details

Your Form Number is: 16

Form Title:

Enter your e-mail address:

Select what to do with the submit results from your form:
Save the results in the database AND email the results to me

Briefly describe your form below.
This will be the first thing displayed to users who fill in the form.

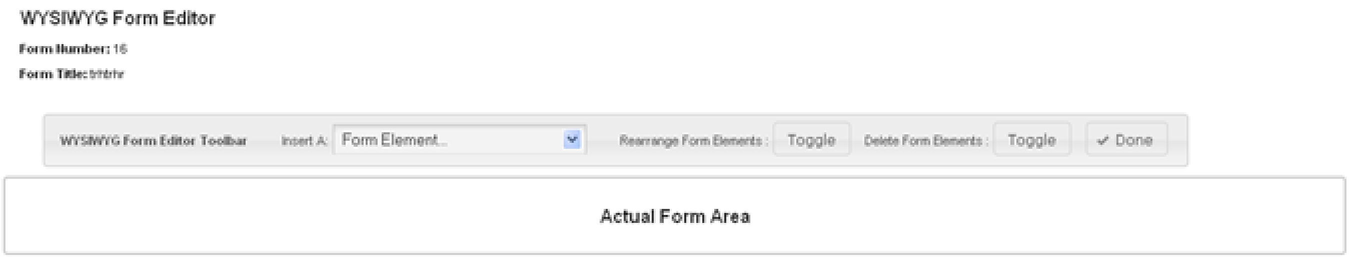
Figure 3: The form metadata modal window is shown.

Once all fields are correctly filled in, the designer can proceed to the next step in form design to the WYSIWYG form editor.

4. WYSIWYG Form Editor

4.1.Basic Editor Layout

The form editor allows designers to actually build their forms through a simple WYSIWYG form editor toolbar according their volition. The form editor layout is depicted in figure 4 with some form metadata information, form editor toolbar and the actual WYSIWYG form area beneath one another.



WYSIWYG Form Editor

Form Number: 16

Form Title: trhtrhr

Figure 4: The form editor layout is shown.

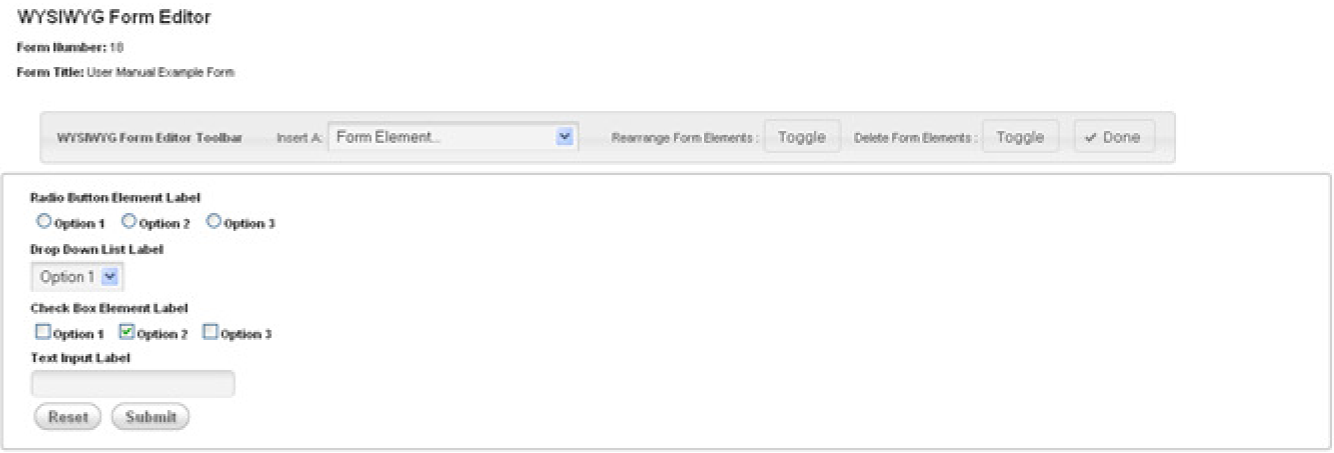
4.2. Form Editor Toolbar Drop Down List

The WYSIWYG form editor toolbar provides all possible functions to create form content. The actual form area enclosed in a gray rectangle under the form editor toolbar contains the actual form. The form area will show the actual look of the created form.

The toolbar contains a drop down list and three buttons. The drop down list contains all the possible form elements that designers can insert into their forms. These include the labels, radio buttons, drop down lists, multi-selectable drop down lists, check boxes, text inputs, text areas, HTML areas, submit buttons, reset buttons, html headings and date picker elements. Appendix A contains details regarding all these form elements. Designers that are not familiar with HTML form elements should browse through this appendix. This appendix also offers suggestions to insert which form elements for certain cases and scenarios in forms.

When a desired form element is chosen, a modal window will open in which the designer has to fill in the relevant parameters belonging to this form element to insert it. Form element modal windows are streamlined to simply insertion of form elements. A detailed explication of modal windows belonging to all of the form elements are given in Appendix B. If designers are unsure of the impacts of certain parameters to form elements contained in the modal windows, appendix B is available for a quick reference.

Using the form element toolbar an example form is built with a radio buttons followed by a drop down list, check boxes, a text input field, a reset and lastly a submit button shown in figure 5. The form is enclosed in the gray rectangle is exactly what the form would look like when it is constructed for submission later hence the term WYSIWYG.



WYSIWYG Form Editor

Form Number: 18

Form Title: User Manual Example Form

Radio Button Element Label

○ Option 1 ○ Option 2 ○ Option 3

Drop Down List Label

Option 1

Check Box Element Label

□ Option 1 ☑ Option 2 □ Option 3

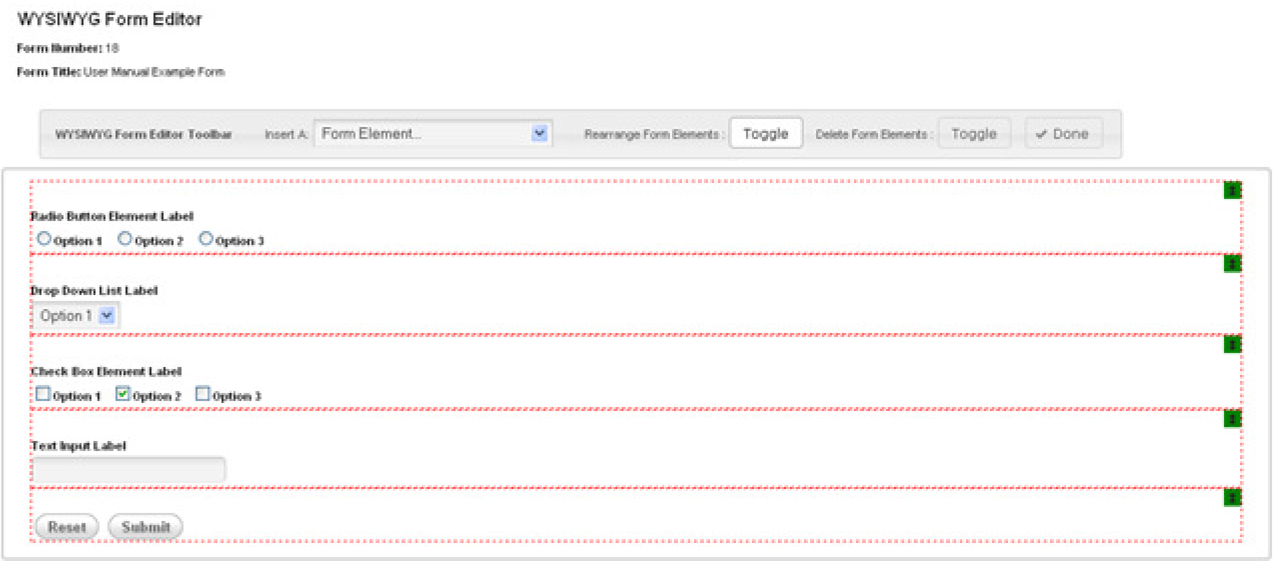
Text Input Label

Figure 5: An example form editor page is shown with a hypothetical built form.

4.3. Form Editor Toolbar Rearrange Form Elements Button

When a new form element is made, it is automatically inserted as the last form element in the form. What if designers want to place this form element some in the middle of their forms? Two major form element manipulation buttons are provided. The "rearrange form elements" button allows designers to change the order of their form elements. The "rearrange form elements button" can simply be clicked to be toggled. A button is turned on when it is toggled. It will automatically change to a distinct color denoting its changed sate [5].

Red dotted rectangles will appear around each form element to separate and distinguish the boundary of each form element. A green up/down arrow button will appear on the top right hand corner of each form element. Click this button belonging to desired form element and drag it to its desired position. Simply clicking and dragging the form elements to their desired positions would change the order of the form elements as they would appear in this WYSIWYG interface. Once designers are satisfied with the order the form elements, click on the "rearrange form elements" button to untoggle or turn it off. The red dotted rectangles and green icons will automatically disappear [6].


WYSIWYG Form Editor

Form Number: 18

Form Title: User Manual Example Form

Radio Button Element Label
○ Option 1 ○ Option 2 ○ Option 3

Drop Down List Label
Option 1

Check Box Element Label
□ Option 1 ☑ Option 2 □ Option 3

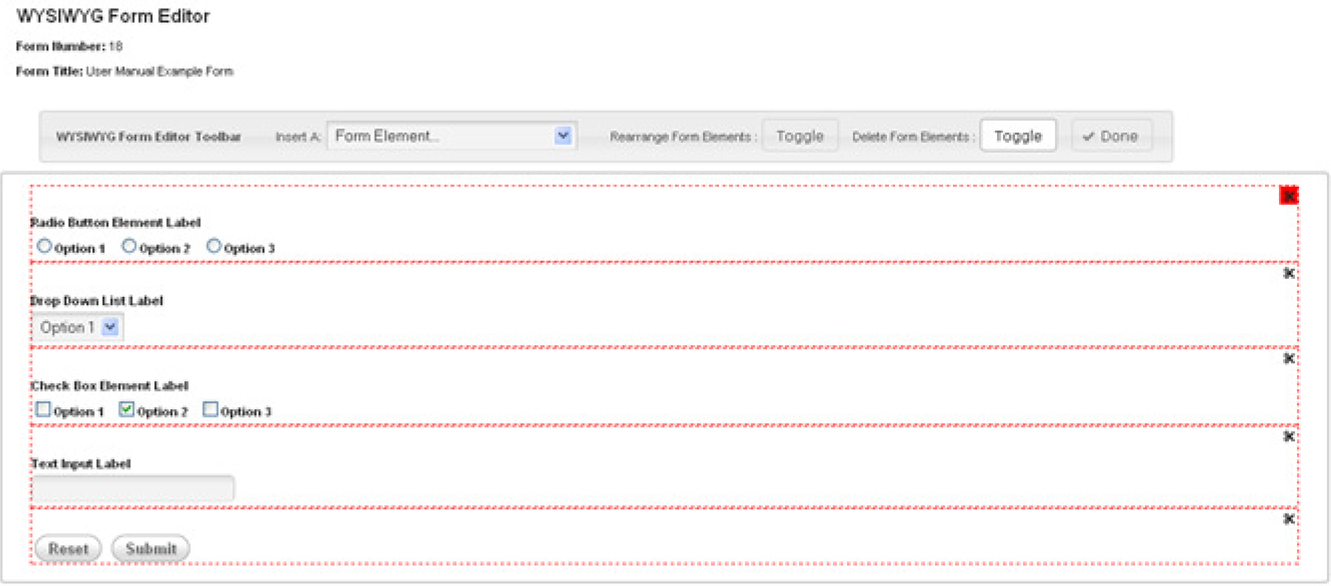
Text Input Label

WYSIWYG Form Editor Toolbar
Insert A: Form Element...
Rearrange Form Elements : Toggle
Delete Form Elements : Toggle
✓ Done
Reset
Submit

Figure 6: An example form editor page is shown with a hypothetical built form and the "rearrange form elements" button toggled.

4.4. Form Editor Toolbar Delete Form Elements Button

How do designers remove unwanted form elements their form? Enter the "delete form elements" button that allows designers to easily remove form elements from their form. There is a design limitation that exists within form editor. Once a certain form element has been inserted into a form, it cannot be edited. If designers want to change the properties a particular form element they will have delete the existing unwanted form element and insert the same form element again with the desired properties.


WYSIWYG Form Editor

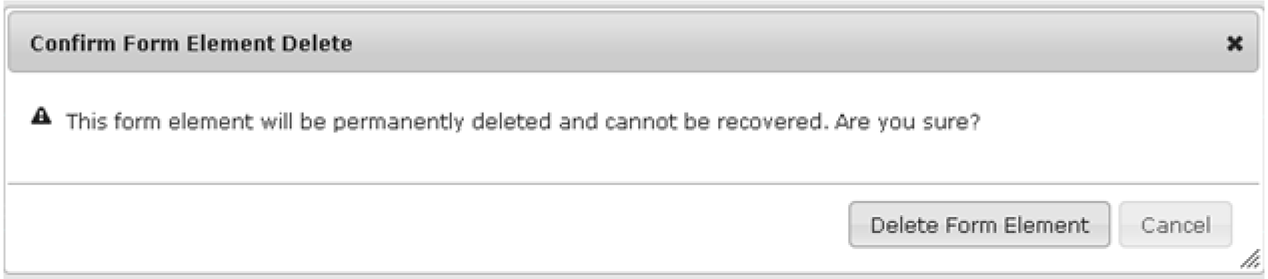
Form Number: 18

Form Title: User Manual Example Form

WYSIWYG Form Editor Toolbar
Insert A: Form Element...
Rearrange Form Elements : Toggle
Delete Form Elements : Toggle
✓ Done
Radio Button Element Label
○ Option 1 ○ Option 2 ○ Option 3
Drop Down List Label
Option 1
Check Box Element Label
□ Option 1 ☑ Option 2 □ Option 3
Text Input Label
Reset
Submit

Figure 7: An example form editor page is shown with a hypothetical built form and the "delete form elements" button toggled.

To delete a form element simply click on the "delete form elements" button to toggle it. Similar to the rearrange form elements button, a red dotted border will appear around each form element to separate and distinguish between each form element as depicted in figure 7. A cut icon will appear on the top right corner of each form element. To delete a form element simply click on the relevant cut icon for that form element. It will glow red on mouse hover as in the radio form element in figure 7. Upon clicking the cut icon, a modal window will appear that will request a delete confirmation show in figure 8. If the user clicks on the delete form element button, the pertinent form element will be deleted and cannot be recovered [7]. Once the desired form elements have been deleted simply click on the "delete form elements" button again to untoggle or turn it off. This will remove the delete form element state with all the cut icons and red dotted borders [8].


Confirm Form Element Delete

This form element will be permanently deleted and cannot be recovered. Are you sure?

Figure 8: The form element delete confirmation dialog box is depicted.

4.5. *Form Editor Toolbar Done Button*

When form designers are satisfied with their forms simply click on the "done" button inside the WYSIWYG form editor toolbar. Every form requires a submit button to be complete. The form editor will check if designers have added a submit button. If a form is devoid of a submit button the form editor will prompt designers with a message indicating the form is incomplete. However, designers are still allowed to continue with the form building process and enter the final stage of publishing this form. Despite this, designer's can still come back and complete, alter or augment their forms according to their volition anytime in the future. This can be done by clicking on the "edit form in WYSWIG editor” button which can be found when clicking on the “form options” button for your form in the form listings. If a form seems to be complete and comprises of a submit button, a modal window will appear for publishing the form.

5. Publishing Form Parameters

5.1. Significance of Publishing Forms

Once the “done” button is clicked on, a modal window will appear to publish this form. Two options are given to publish the form or leave it unpublished as illustrated in figure 9. The designer can choose to leave the form unpublished and click on the “Set Publishing Parameters” button to proceed with an unpublished form. The ramification of a form being unpublished is that it cannot be constructed for *users* for submission.

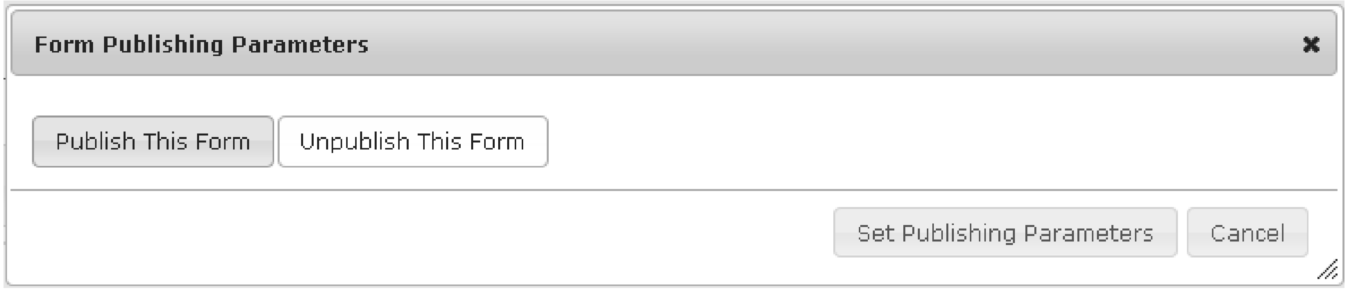
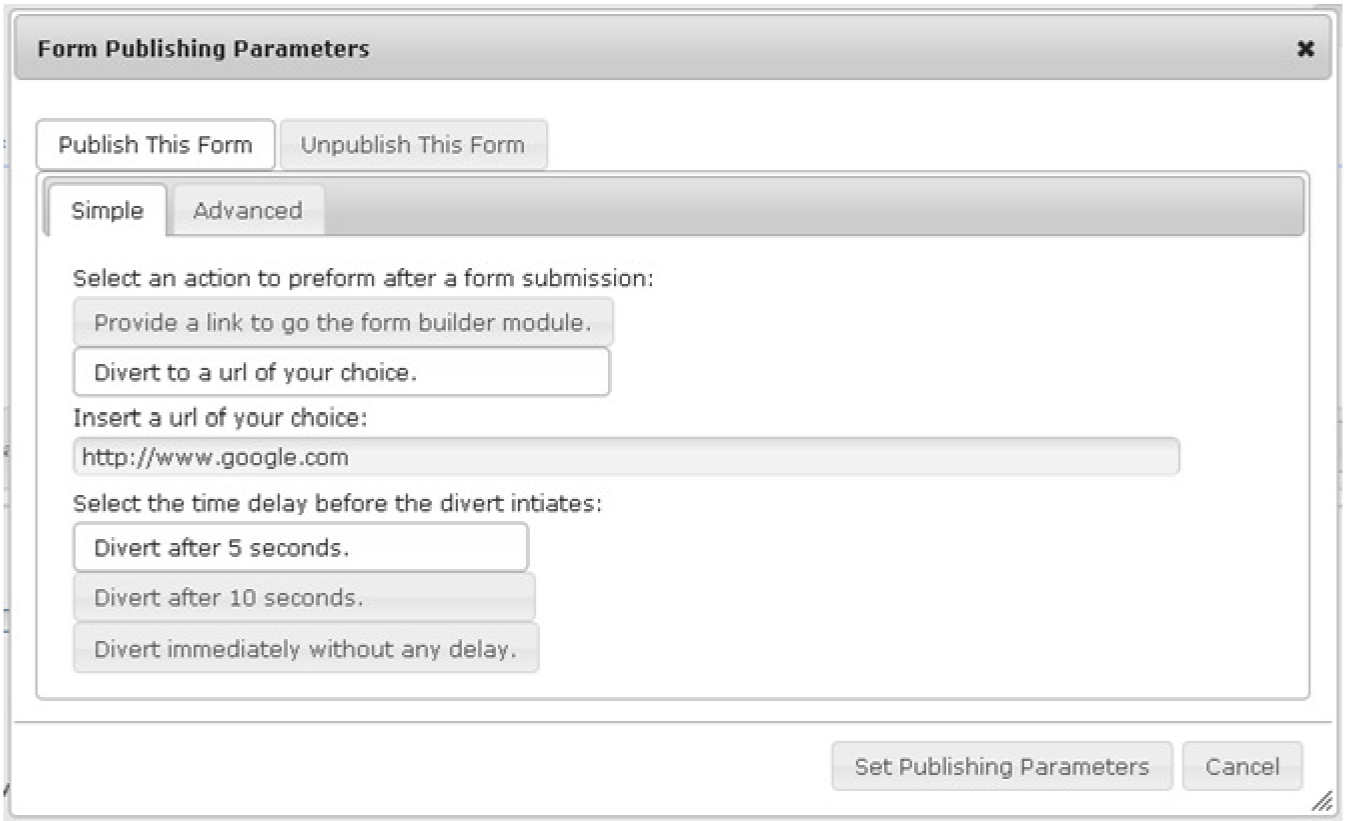


Figure 9: A publishing form modal window is shown.

5.2. Simple Publishing Parameters

If designers choose to publish their form, a certain set of actions have to be set when their forms are submitted. Designers can prefer simple or advanced publishing parameters. For simple publishing parameters, once a form has been submitted, the form can always divert to a URL set in the text field labeled “Insert a URL of your choice”. In figure 10, the form is set to divert to google’s homepage. Additionally, a time delay can be set before the form builder diverts to the specified URL or designers can opt to divert without any delay. On the other hand, if designers’ do not want to divert to any URL, the “provide a link to go the form builder module” can be selected. Once a form gets submitted, a new page will load that contains a button that allows users to return to the home page of the form builder module.


Form Publishing Parameters
×
Publish This Form
Unpublish This Form
Simple
Advanced

Select an action to preform after a form submission:

Provide a link to go the form builder module.
Divert to a url of your choice.

Insert a url of your choice: http://www.google.com

Select the time delay before the divert intiates:

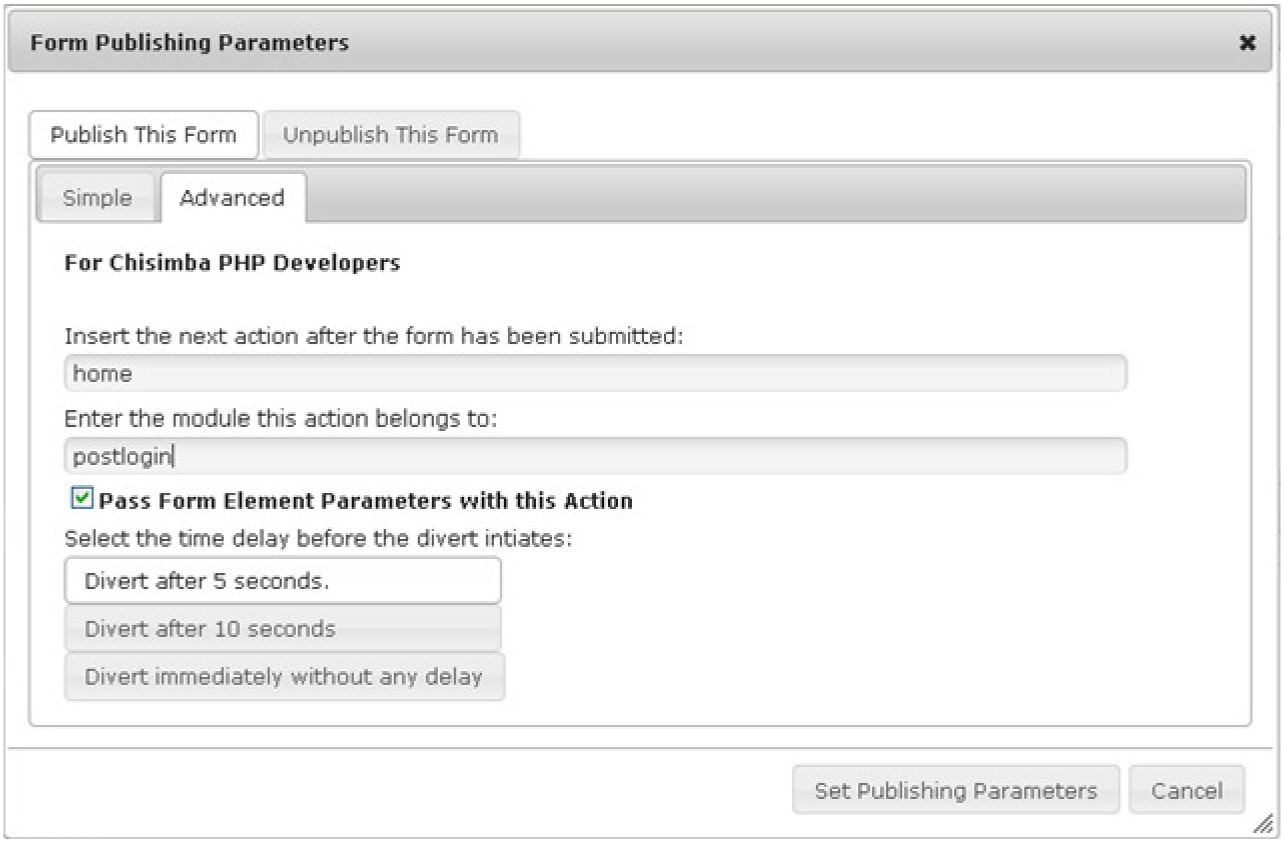
Divert after 5 seconds.
Divert after 10 seconds.
Divert immediately without any delay.
Set Publishing Parameters
Cancel

Figure 10: A publishing form modal window is shown with simple publishing parameters.

5.3.Advanced Publishing Parameters

Advanced publishing parameters are for Chisimba PHP developers. This is shown in figure 11. Instead of a URL, the form builder diverts to an action belonging to a Chisimba module. In figure 11, the form is set to divert in 5 seconds after a form submission to the default action called home from the Post Login Chisimba module. Additionally, a checkbox option is given to pass all the form parameters to this action. If this checkbox is selected to be on, the names and values of all the form elements inside the form get transformed into a text string in standard URL-encoded notation. Chisimba code 1 gets executed.

```
$this->nextAction($chisimbaAction,$nextActionParameterArray,$chisimbaModule); (1)
```



For Chisimba PHP Developers

Insert the next action after the form has been submitted: home

Enter the module this action belongs to: postlogin

☑ **Pass Form Element Parameters with this Action**

Select the time delay before the divert intiates:

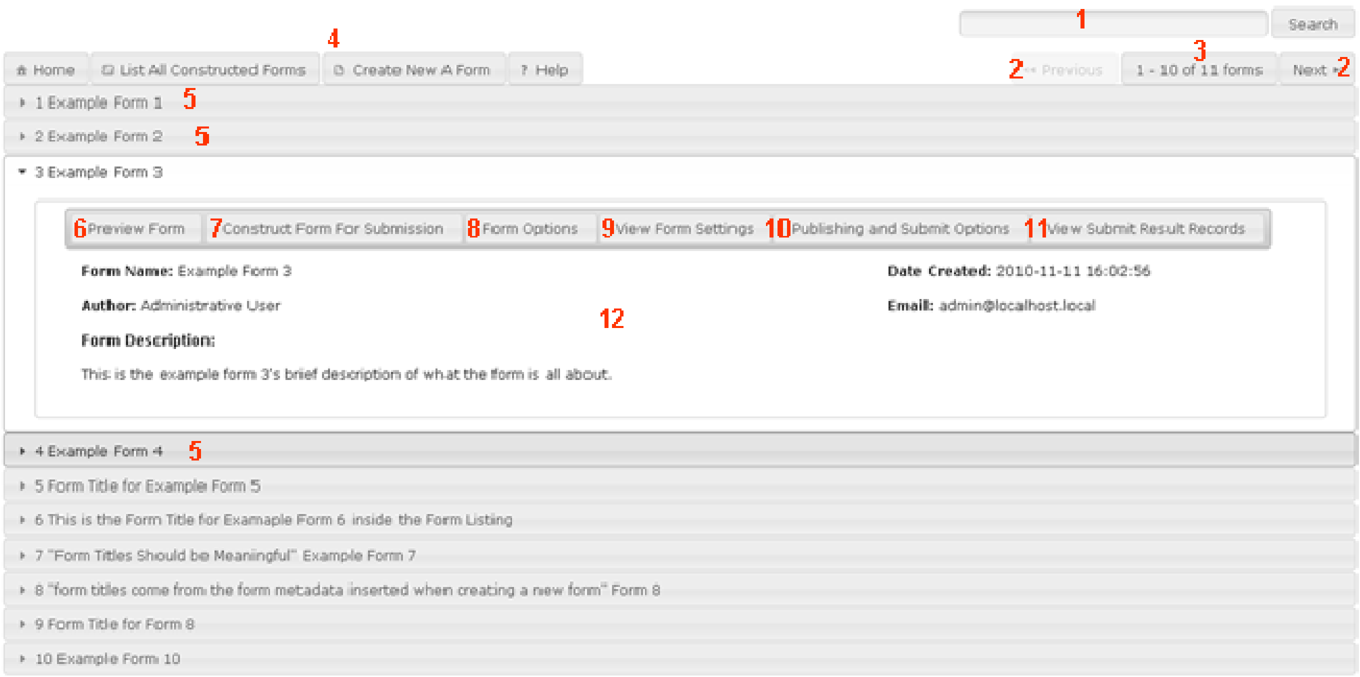
Figure 11: A publishing form modal window is shown with advanced publishing parameters.

Once the publishing parameters are set, the designer's form is ready for use!

6. Form Options inside the Form Listings

6.1.Basic Overview

From figure 2, if designers click on the "List All Forms" button or link, a list of all designed forms are shown in a format shown in figure 12.


1
Search
4
Home
List All Constructed Forms
Create New A Form
? Help
2 « Previous
3
1 - 10 of 11 forms
Next » 2
1 Example Form 1 5
2 Example Form 2 5
3 Example Form 3
6 Preview Form
7 Construct Form For Submission
8 Form Options
9 View Form Settings
10 Publishing and Submit Options
11 View Submit Result Records
12
4 Example Form 4 5
5 Form Title for Example Form 5
6 This is the Form Title for Examaple Form 6 inside the Form Listing
7 "Form Titles Should be Meaningful" Example Form 7
8 "form titles come from the form metadata inserted when creating a new form" Form 8
9 Form Title for Form 8
10 Example Form 10

Form Name: Example Form 3

Date Created: 2010-11-11 16:02:56

Author: Administrative User

Email: admin@localhost.local

Form Description:

This is the example form 3's brief description of what the form is all about.

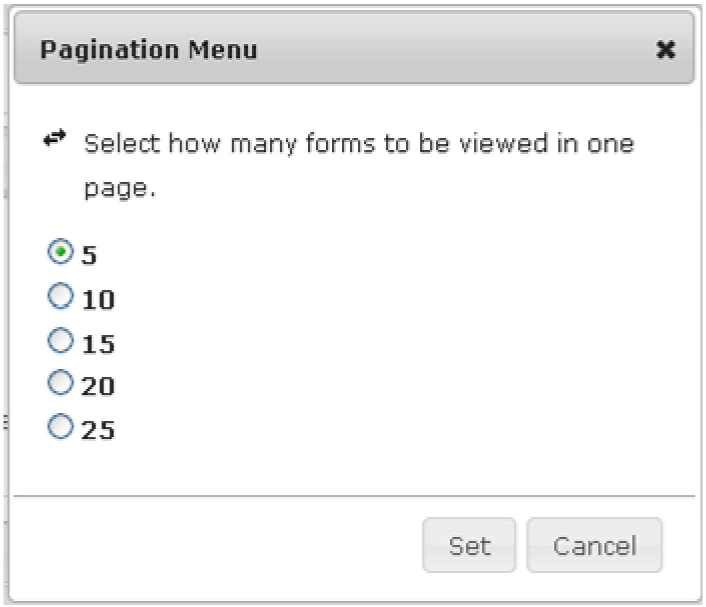
Figure 12: The page that lists all forms is shown with colour coded numbers to help denote each element within this page.

6.2. Quick Search Toolbar

Once the process of designing a form is completed, all designed forms are listing on this page. The entity labeled 1 in figure 1 is a quick search bar. It would be innate that hundreds of forms would be built by dozens of form designers. A quick search toolbar is helpful tool list a set of desired forms according to the search query. The quick search tool bar can take all form metadata parameters such as form titles, descriptions, designer names, email addresses of designers and creation dates.

6.3. Form Pagination

By default, 10 forms are listed on each page. If there are more forms than this number, click on the next and previous buttons labeled 2 to show next and previous pages of results. The button labeled as 3 provides a pagination indicator. Additionally, if this button is clicked upon, a modal window pops up that permits the change of default number of forms being displayed on a page as depicted figure 13 [9].



Pagination Menu ✖

⇄ Select how many forms to be viewed in one page.

● 5
○ 10
○ 15
○ 20
○ 25

Figure 13: A modal window that prompts the change of the default number of forms being listed in a page.

6.4.Form Accordion Listing

The entity labeled 4 is the quick start toolbar. The entity is labeled 5 is an accordion list of designed forms. Each form item can be expanded or stretched to reveal content associated with that form. In figure 12, form 3 is selected revealing its details and options. When an accordion form item is selected some form metadata labeled as 12 and a tool bar of form options are provided [10].

6.5.Form Accordion Toolbar Buttons – Preview Form Button

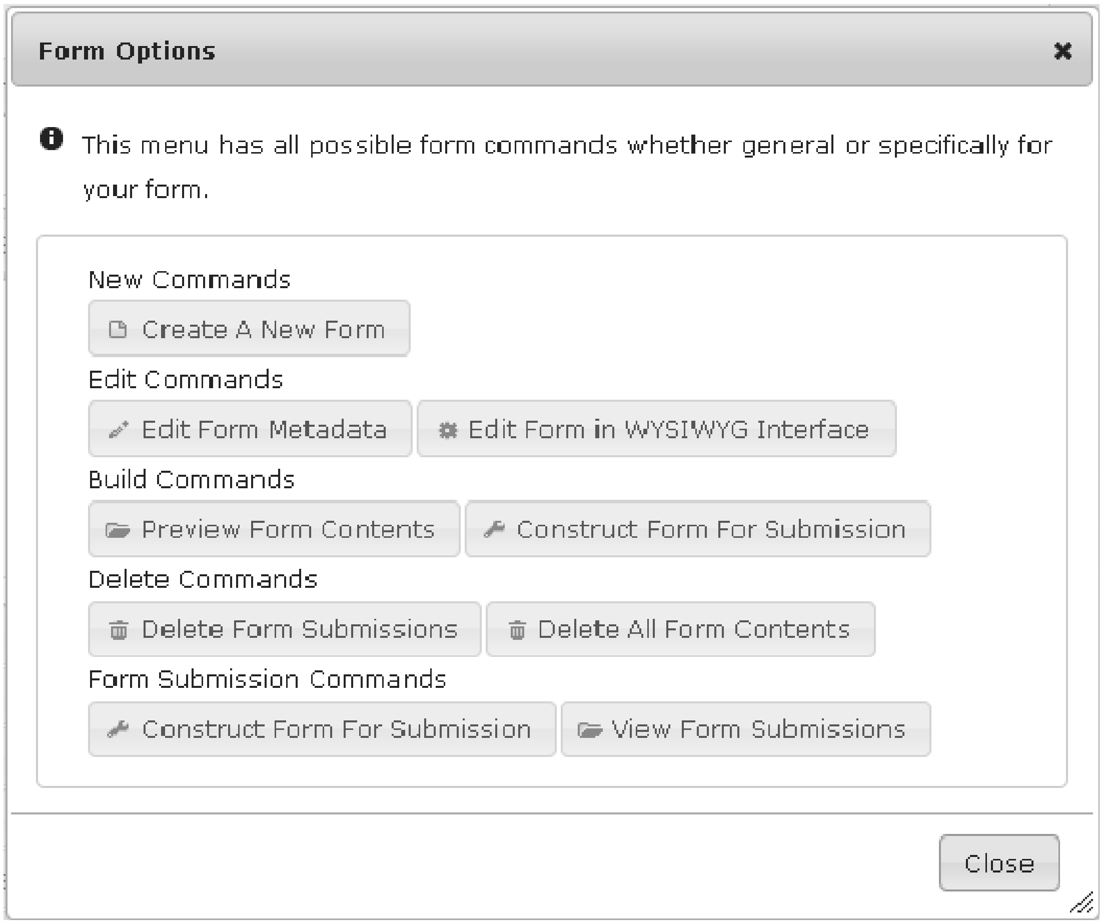
The "Preview Form" button labeled 6 on the form options list opens a modal window that shows a read only version of the designed form. It cannot be altered or augmented in any way.

6.6.Form Accordion Toolbar Buttons – Construct Form for Submission Button

Button 7 called "Construct form for Submission" builds the designer's form for the *user* to fill in and submit. Refer to section 2.1 to determine the discrepancy between designers and users of a form.

6.7.Form Accordion Toolbar Buttons – Form Options Button

If the button "form options" is clicked on, a modal window opens with all the possible form commands whether general or specific to the designer's form depicted in figure 14. The new commands will allow designers to create new forms from scratch. If designers want to alter the form metadata, the button "Edit Form Metadata" exists for this use. If designers want to alter or augment their existing form contents which are the form elements, designers can go back to the WYSIWYG form editor to make willful changes. The build command buttons will either build the designed for preview or for users of that form for submission. Two delete options are given. The "Delete Form Submissions" button will delete all the submission results of users for that form. The "Delete All Form Contents" button will delete everything belonging to this form including the form metadata, form element contents, publishing settings and submission results. The form submission commands include a button for users of the form for submitting results and for designers to view results submitted by the users.



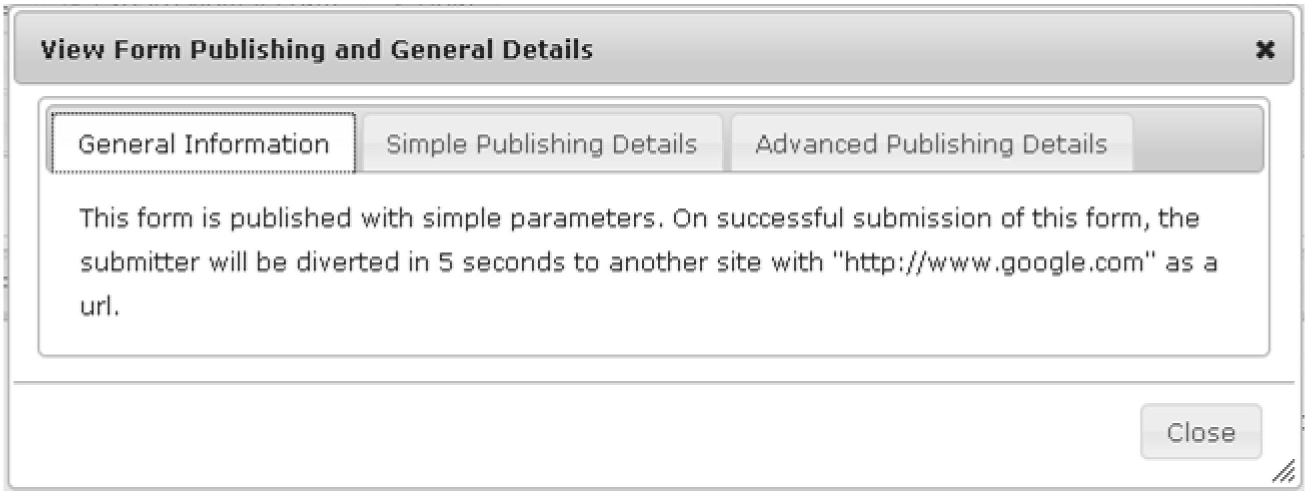
Form Options

This menu has all possible form commands whether general or specifically for your form.

Figure 14: A modal window that lists all the possible commands for forms.

6.8.Form Accordion Toolbar Buttons – View Form Settings Button

Returning to figure 12, the "View Form Settings" button colored coded as 9 will display a modal window displaying the publishing details selected by the designer as well as simple and advanced directions to construct or render designers' forms. Figure 15 shows the form settings modal window with the general information tab opened. This tab simply provides the brief description of the publishing details selected by the user [11].

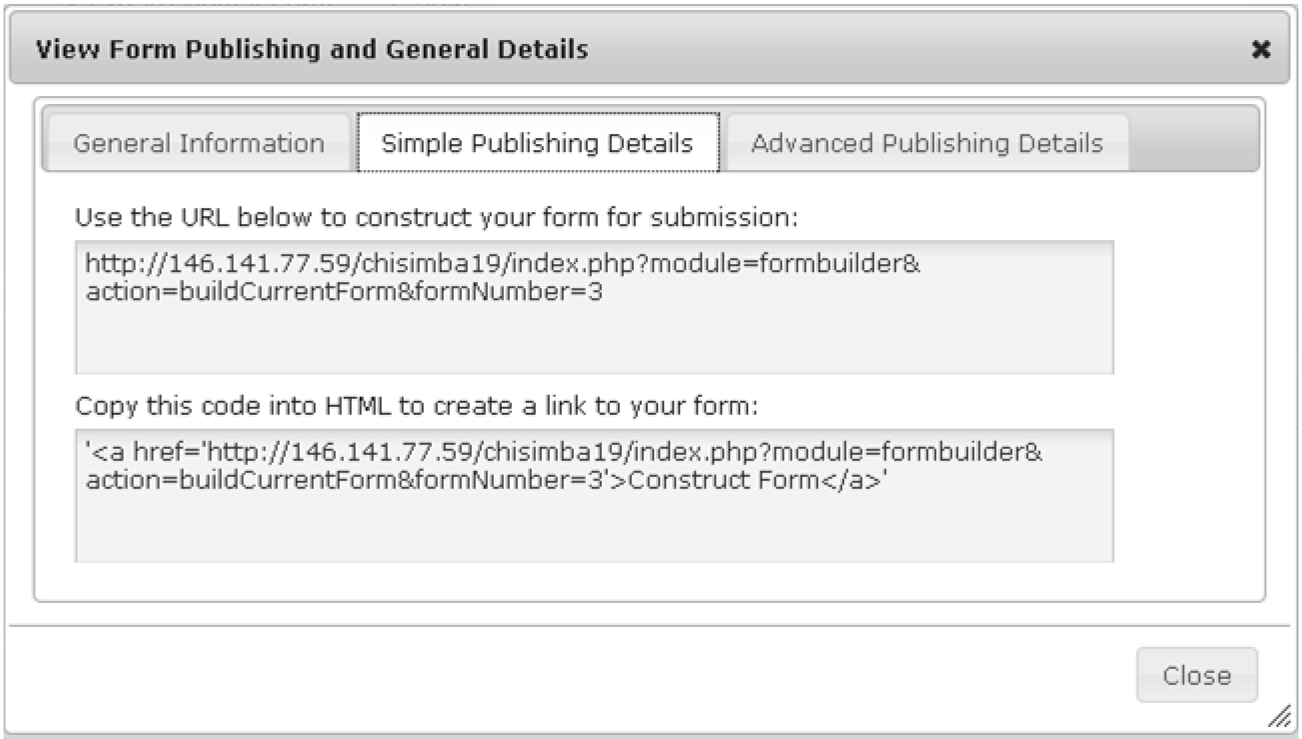


View Form Publishing and General Details

This form is published with simple parameters. On successful submission of this form, the submitter will be diverted in 5 seconds to another site with "http://www.google.com" as a url.

Figure 15: A modal window that shows designer form settings with the general information tab open.

The form settings modal window with the "Simple Publishing Details" tab opened is shown in figure 16. For simple designers that possess fair knowledge about HTML can use the information in this tab to construct their form anywhere they desire. From figure 16, the first text area contains the URL address to create the designer's form. Designers can use this knowledge to build their forms anywhere they desire. To see this in action, simple copy the contents of this text area and paste it into the internet address bar of an internet browser to build the form. The second text area contains HTML code to produce a link to construct or render form.



View Form Publishing and General Details

Use the URL below to construct your form for submission:

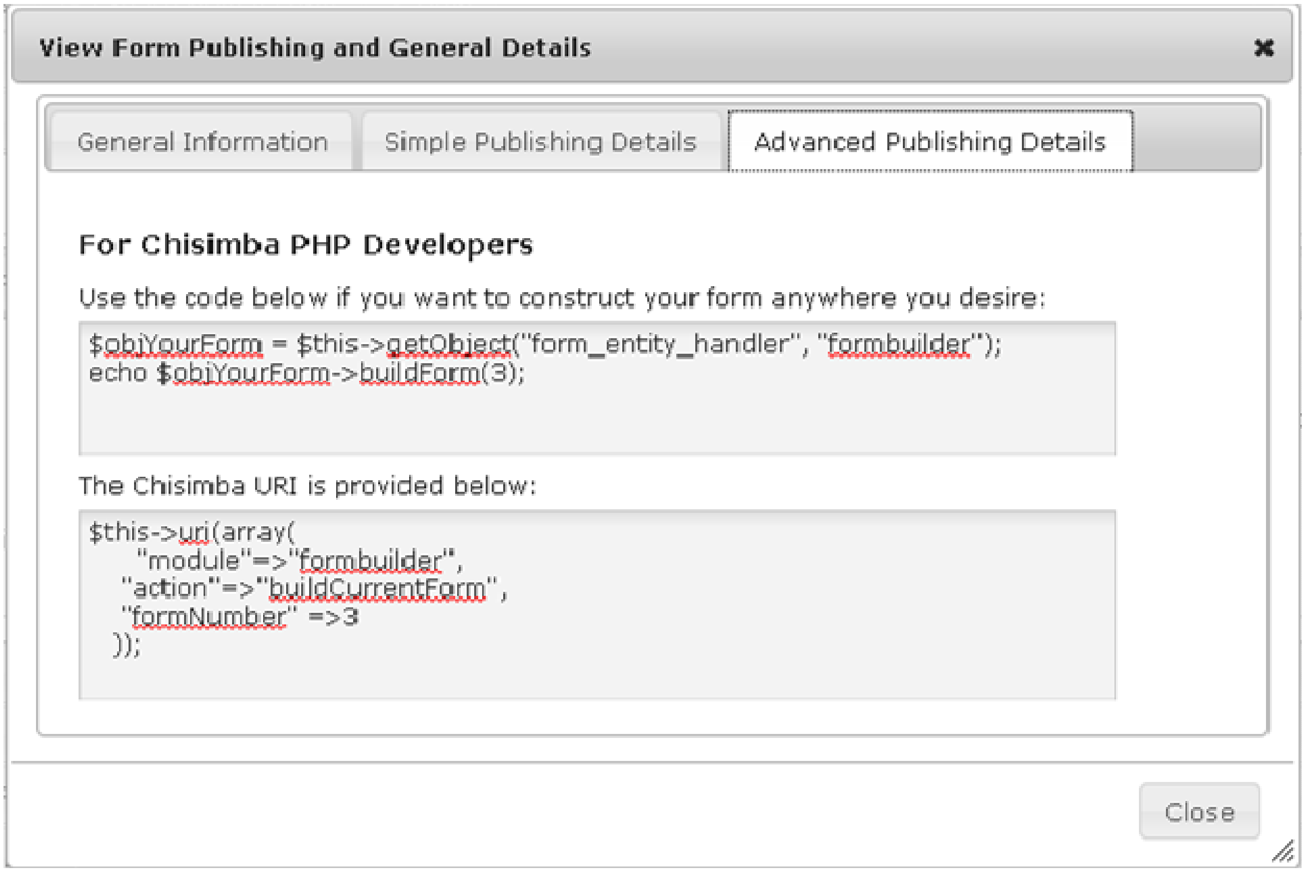
```
http://146.141.77.59/chisimba19/index.php?module=formbuilder&
action=buildCurrentForm&formNumber=3
```

Copy this code into HTML to create a link to your form:

```
'<a href='http://146.141.77.59/chisimba19/index.php?module=formbuilder&
action=buildCurrentForm&formNumber=3'>Construct Form</a>'
```

Figure 16: A modal window that shows designer form settings with the simple publishing details tab open.

The form setting modal window with the "Advanced Publishing Details" tab opened is shown in figure 17. This tab contains information for PHP developers to produce their forms anywhere in their modules. The first text area in figure 17 consists of Chisimba PHP code to construct or render the form. The second text area consists of a Chisimba PHP URI (Uniform Resource Indicator) that developers can use at their disposal.


View Form Publishing and General Details
General Information
Simple Publishing Details
Advanced Publishing Details
Close

For Chisimba PHP Developers

Use the code below if you want to construct your form anywhere you desire:

```
$objYourForm = $this->getObject("form_entity_handler", "formbuilder");
echo $objYourForm->buildForm(3);
```

The Chisimba URI is provided below:

```
$this->uri(array(
    "module"=>"formbuilder",
  "action"=>"buildCurrentForm",
  "formNumber" =>3
 ));
```

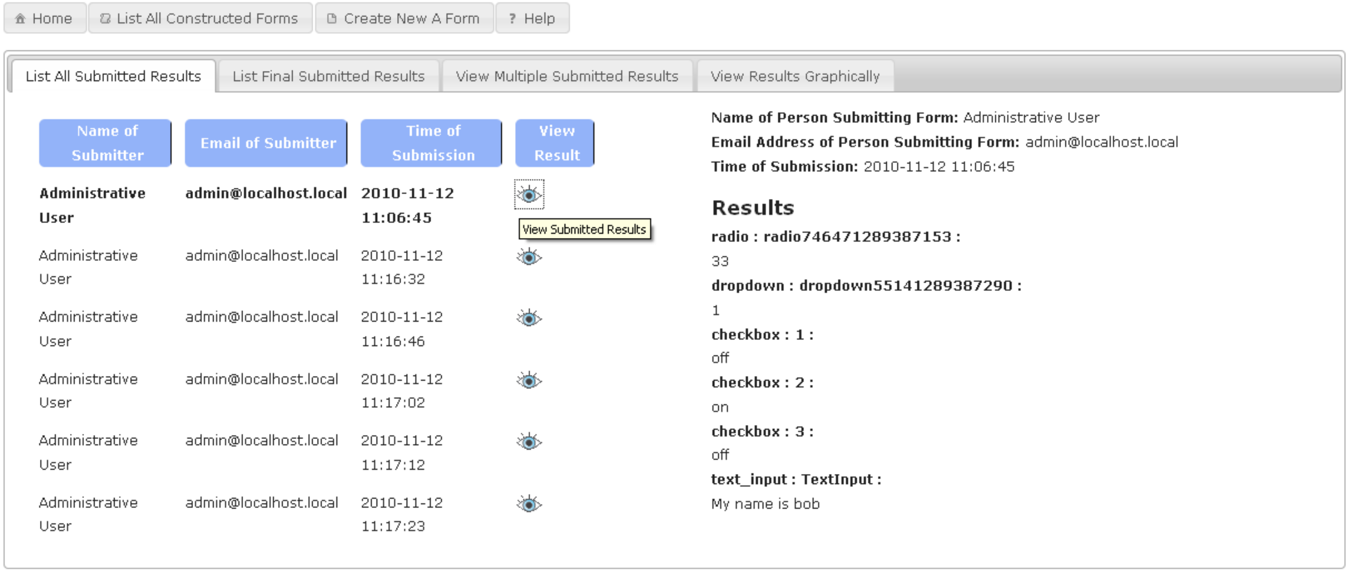
Figure 17: A modal window that shows designer form settings with the advanced publishing details tab open.

6.9.Form Accordion Toolbar Buttons – Publishing and Submit Options Button

Returning to figure 12, the "Publishing and Submit Options" button color labeled 10 will produce the same modal window explicated in section 3.3. Refer to this section for more information about this button. It is important to note that editing a form inside the WYSIWYG form editor will automatically unpublish it. When the designer decides to publish this form, the previous publishing parameters saved do not get lost.

6.10. Form Accordion Toolbar Buttons – View Submit Result Records Button

The "View Submit Result Records" button will enable designers to view all the current form submissions of this form in a page payout shown in figure 18. The default tab "List ALL Submitted Results" will display of all form submissions in a format shown below. To view the actual result, click on the link shaped like an eye and the result will slide out to right hand side of the page shown in figure 18. Sometimes users will submit the form multiple times, the second tab "List Final Submitted Results" will list only the last submission of each user in the same format as in figure 18. The "View Multiple Submitted Results" and "View Result Graphically" tabs are currently under construction by the developer of this module. It will possibly include pie charts and bar graphs to present the submit results [12].


Home
List All Constructed Forms
Create New A Form
Help
List All Submitted Results
List Final Submitted Results
View Multiple Submitted Results
View Results Graphically
View Submitted Results

Name of Submitter	Email of Submitter	Time of Submission	View Result
Administrative User	**admin@localhost.local**	**2010-11-12 11:06:45**	👁
Administrative User	admin@localhost.local	2010-11-12 11:16:32	👁
Administrative User	admin@localhost.local	2010-11-12 11:16:46	👁
Administrative User	admin@localhost.local	2010-11-12 11:17:02	👁
Administrative User	admin@localhost.local	2010-11-12 11:17:12	👁
Administrative User	admin@localhost.local	2010-11-12 11:17:23	👁

Name of Person Submitting Form: Administrative User
Email Address of Person Submitting Form: admin@localhost.local
Time of Submission: 2010-11-12 11:06:45

Results

radio : radio746471289387153 :
33
dropdown : dropdown55141289387290 :
1
checkbox : 1 :
off
checkbox : 2 :
on
checkbox : 3 :
off
text_input : TextInput :
My name is bob

Figure 18: The layout of the view submission records page is shown.

7. Conclusion

Some additional features will be added in future. Existing entities, layouts and styles would be slightly altered. However, the core functionality explicated in this manual will remain the same. When creating a new form, the form flow of inserting form metadata, form element contents inside the form editor and publishing details have to be inserted for successfully creating a new form. If any of these steps are neglected, the intended result of creating a form for users to fill in and submit will not be achieved. When forms are successfully created, the contents, metadata and publishing settings can easily be altered using the options provided in the form listings. Additionally, forms and form submissions can be deleted if it is required. The functionality of this module has been explored and explicated.

References

[1] http://cutesoft.net/aspnet/wysiwyg-html-editor.htm?gclid=CPKRs6LAoaUCFYVA4wodLxIkJA, Last accessed 6 November 2010.

[2] Armond Furmie, Senior Software Analyst, Personal Communication.

[3] http://www.softpanorama.org/Algorithms/searching.shtml, Last accessed 6 November 2010.

[4] http://jqueryui.com/demos/dialog/, Last accessed 6 November 2010.

[5] http://jqueryui.com/demos/button/, Last accessed 6 November 2010.

[6] http://jqueryui.com/demos/sortable/, Last accessed 10 November 2010.

[7] http://www.filamentgroup.com/lab/styling_buttons_and_toolbars_with_the_jquery_ui_css_framework/, Last accessed 11 November 2010.

[8] http://jqueryui.com/docs/Theming/API, Last accessed 11 November 2010.

[9] http://plugins.jquery.com/project/pagination, Last accessed 12 November 2010.

[10] http://docs.jquery.com/UI/Accordion, Last accessed 13 November 2010.

[11] http://docs.jquery.com/UI/Tabs, Last accessed 13 November 2010.

[12] http://www.jqplot.com/, Last accessed 13 November 2010.

Bibliography

http://www.chisimba.com/, Last Accessed: 5 October 2010.

http://cvs2.uwc.ac.za/trac/chisimba_documentation, Last Accessed: 5 October 2010.

http://cvs2.uwc.ac.za/trac/chisimba_documentation/browser/documentation/chisimba/pdf, Last Accessed: 22 September 2010

Play-hookey.com, A b*asic digital counter*, http://www.play-hookey.com/digital/ripple_counter.html, Last Accessed: 21 September 2010

Play-hookey.com, *A synchronous binary counter*, http://www.play-hookey.com/digital/synchronous_counter.html, Last Accessed: 29 October 2010

Play-hookey.com, *The 555 timer IC*, http://www.play-hookey.com/digital/timer_555.html, Last Accessed: 30 October 2010

W. Kleitz, *Digital and Microprocessor Fundamentals Theory and Applications.* Upper Saddle River, New Jersey, 07458: Pearson Education Inc., 2003.

Appendices

A. HTML Form Elements

A.1 HTML Forms

HTML forms contain form elements or fields that allow users to enter or fill in information. For instance, text will inserted in text inputs and area and options are selected in drop down or radio buttons. The most integral part of a form is the submit button which takes the information filled in the form and executes certain specified actions. For instance, once the submit button is hit, the information filled in the form either get emailed to the designer of the form or saved into a database for future viewing or both. There are several types of form elements that can be added into a form. These will be explained in detail. Additionally, a specific mention will be given for the uses of these form elements and the cases to utilize them in forms.

Any HTML entity can be ascribed with an ID. This includes HTML form elements. With many form elements, it becomes easier to differentiate between multiple form elements of the same type. As a general rule, all form elements that are made by the form builder require a unique ID to be inserted [1].

A.2 Text Input

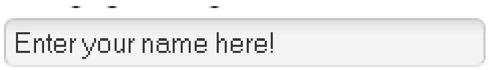
The text input is a form field that allows users to fill in text which is restricted to only one line. Designers can also add more attributes to this form element. The text input can be given a name. This name becomes important when designers view form submission records. The text input's name with the data that users input inside it will be shown.

Designers can also set the size of the text input field which defines the width of the field. The width of the field will determine the number of characters that can be visible inside the field [2].

A default value for the text input can be set. In this case the default value will be text that will appear inside the text input field by default. The value is text that the user of the form fills in that gets saved in a database or sent to designer by email with the name of the text input when the form gets submitted. Code 2 shows the html code that produces a text input.

```
<input type="text" size="35" value="Enter your name here!">....................(2)
```

Code 2 produces a text input shown in figure19. Note that the value gets shown by default inside the field which can be changed by the user of the form.



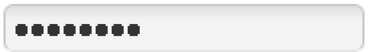
Enter your name here!

Figure 19: A text input field is shown.

A password field is like text input field which one minute difference. The only difference is that inserts data of type password and not text as seen in code 3. The data that gets entered into a password field is displayed as dots as to prevent others looking at the screen to read the password.

```
<input type="password" size="35" value="password">............................(3)
```

Code 3 produces a password field shown in figure 20. Notice that the value password shows up as eight dots.



●●●●●●●●

Figure 20: A password input field is shown.

Apart from text and password fields, an input mask can also be applied to a text input field. An input mask is defined by the designer of the form that governs what users are allowed to enter as input in a text input field. It can also aid in adding special characters such as dashes for credit card numbers or parentheses for telephone numbers. The input masks offered by the form builder are listed in table 1 [3].

Table 21: A table listing all the possible input masks provided by the form builder.

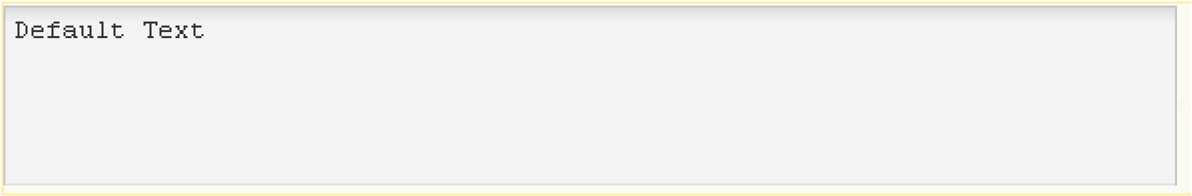
Input Mask Type	Format
Number	1234567890
Date (US Format)	day/month/year
Date (ISO Format)	year-month-day
Time	hour : minute
Phone Number	(000)000-0000
Credit Card Number	0000-0000-0000-0000

A.3 Text Area

Text areas are similar to text inputs except that text areas can span multiple lines. Text areas are unique form fields as these entities do not start with the tag <input> but rather the tag <textarea> as expressed in code 4. Text areas also possess "row" and "cols" attributes to specify the row and column size of the text area. All other attributes such as "name" and "value" behave the same as in text inputs.

```
<textarea cols="70" rows="4" name="myTextArea">Default Text</textarea>.........(4)
```

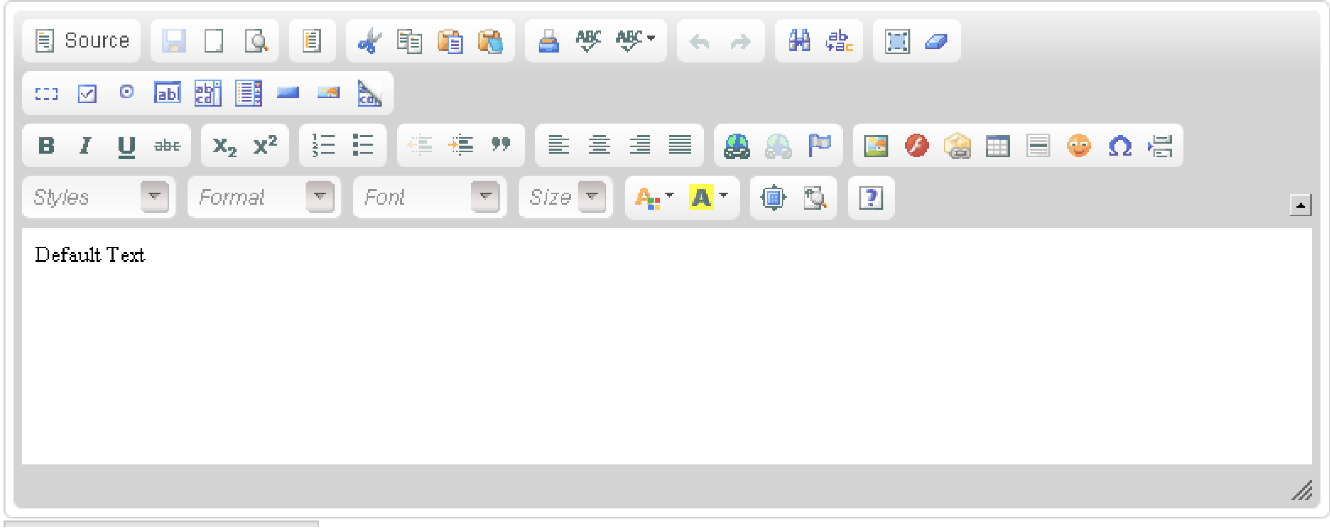
Code 4 produces a text area shown in figure 22. Notice that the value of the text area is encapsulated inside the <textarea> tags rather than the value tag. However, this attribute is flexible as specifying default text within a value attribute will also produce the same result [4].



Default Text

Figure 22: A text area with some default text is shown.

The form builder also allows the insertion of text areas with text editing toolbars on top. There are five different types of toolbars that allow users to insert various types of data with different styles. Figure 23 shows a text area with the forms toolbar [5].



Default Text

Figure 23: A text area with some default text and the forms toolbar is shown.

A.4 Check Box Element

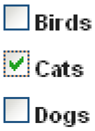
Check boxes are utilized when the designer wants the user of the form to select one or more options from a set of alternatives. If only one option is to be selected in a group of options, it is suggested that radio buttons should be used.

Similar to other form elements, the name attribute adds an internal name to the field so the form builder can uniquely identify it apart form other fields. The value attribute defines what will be submitted by the user. In this case, if the use checks this form element, the value "on" gets saved along with the element's name. If the check box remains unchecked, the value "off" gets stored. The value of the check box element should not be confused with the option text. The option text is what gets displayed next to the check box. It could be thought of as a label for the check box [6].

The attribute "checked" will allow designers to default check this field. Example HTML is given in code 5.

```
 <input type="checkbox" name="Bird">Birds<br>
<input type="checkbox" name="Cat" checked>Cats<br>............................(5)
 <input type="checkbox" name="Dog">Dogs
```

Code 5 produces the form entity displayed in figure 24. Notice the checked attribute is added in the cats option to produce this check this option by default [7].



□ Birds

☑ Cats

□ Dogs

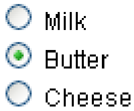
Figure 24: Check box elements are shown.

A.5 Radio Button

Radio buttons are used when the designer desires to permit the user of the form to select only one option from a set of other alternative options. If the designer wants to allow more options to be selected, it is suggested to use check box elements instead. Similar to all other form elements, the name attribute specifies a group of radio buttons the field belongs to. When the user selects one option; this causes all the other options become unselected. The value attribute defines what will be submitted if checked. Unlike the check box element, the value can be defined. Also, the option text has to be specified. It is imperative the designers not conflate the value attribute with the option text. The option text is what gets displayed next to the radio button while the value is what gets submitted when selected. Similar to the check box element, the checked attribute will allow designer to check only one option from a group of alternative options as a default. Code 6 shows some an example html to produce a group of radio buttons [8].

```
<input type="radio" name="group1" value="Milk"> Milk<br>
  <input type="radio" name="group1" value="butter" checked> Butter<br>..........(6)
<input type="radio" name="group1" value="Cheese"> Cheese
```

From code 6, the radio buttons produced is shown in figure 25. If this group of radio buttons gets submitted in this state, the value “butter” will be saved in the database or emailed to the designer of the form [9].



○ Milk
● Butter
○ Cheese

Figure 25: Check box elements are shown.

A.6 Drop Down List

Drop down lists are possibly the most flexible form elements that can be inserted in forms by designers. Designers can set drop down lists to include the same propose and behave similar to radio buttons (one selection only) or check boxes (multiple selection permitted). The drop down list consumes less space than radio buttons or checkboxes at the cost of users not able to see all the options by default. To circumvent this restriction, designers can opt to add multi-selectable drop downs that permit the setting of a specific size to show more that just one option at a time. However, when designers opt to do this, the advantage of conserving space is lost. There is a delicate balanced relationship between space and accessibility. Designers should be aware of this tradeoff [10].

In some cases, designers should replace text fields with drop down menus. This will make designers’ forms more user friendly as selecting form a menu faster and easier than typing.

Firstly, a single selectable drop down menu will be discussed. The name attribute specifies group of drop down list options. When the user selects one option, this causes all the other options to become unselected. The value attribute defines what will be submitted if checked. Unlike the check box element, the value can be defined. Also, the option text has to be specified. It is imperative the designers not conflate the value attribute with the option text. The option text is what gets displayed inside the drop down list while the value is what gets submitted when selected. Unlike radio buttons and check boxes, drop down lists have the selected attribute which serves the function similar to the checked attribute. The “selected” attribute will allow designer to check only one option from a group of alternative options as a default. Code 7 shows some an example html to produce a list inside drop down menu [11].

```
        <select name="mydropdown">
        <option value="Milk" selected=”selected”>Fresh Milk</option>
    <option value="Cheese">Old Cheese</option >...................................(7)
        <option value="Bread">Hot Bread</option>
        </select>
```

From code 7, this html code produces a drop down list shown in figure 26. This figure shows both a closed and opened menu.

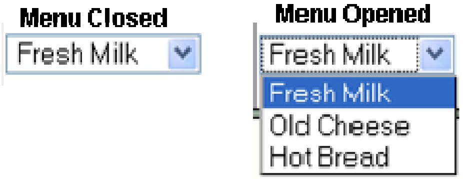


Figure 26: Opened and closed drop down menus are shown.

The multi-selectable drop down menu allows designers to display any number of options. Additionally, multiple options can be selected as default. Users can also select multiple values. Code 8 is the html to produce a multi selectable drop down menu.

```
<select multiple="multiple" name="mydropdown[]">
<option selected="selected" value="Milk">Fresh Milk</option>
   <option value="Cheese">Old Cheese</option >...................................(8)
<option selected="selected" value="Bread">Hot Bread</option>
<option value="Eggs">Large Eggs</option>
</select>
```

From code 8, a multi-selectable drop down menu is produced shown in figure 27. From code 8 and resultant figure, it can be seen that the values milk and break are selected as default. Additionally, a menu size is not specified. If a menu size is not specified, the menu will auto fit all options as seen in figure 27 [12].

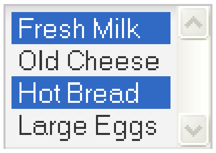


Figure 27: A multi-selectable drop down menu is shown.

A.7 Date Picker Object

If the designer wants the user of the form to select a date, two approaches can be taken. A text input field with a date input mask can be inserted. However, this is not user friendly. The date picker object can be inserted to increase simplicity and usability. As with other form elements, a name attribute has to be given. Additionally, a date format has to be selected by the designer. Once this is set, dates that are filled in by users of the form will be saved and displayed in this format. A real time or default date can also be selected. The real time is a default setting if no default date is specified and will show the current date by default. If a specific date is chosen, this date will be shown by default. The date picker is not solely an HTML entity. It also incorporates some JavaScript. Figure 28 displays the layout of date picker object. [13]

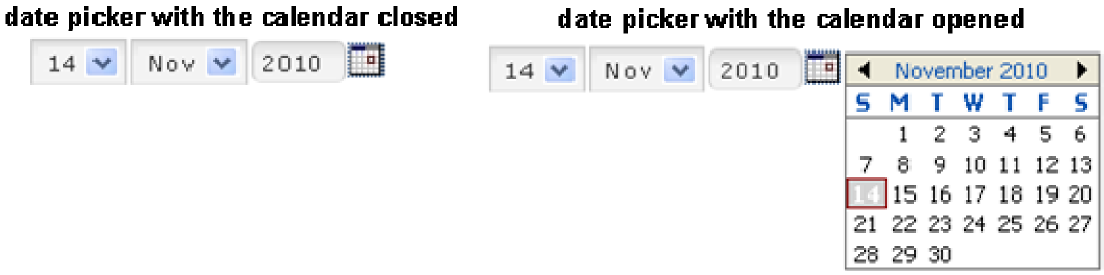


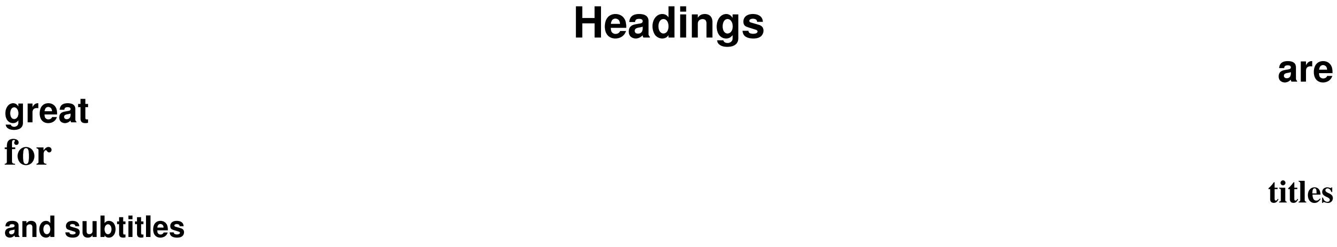
Figure 28: Two date picker objects one with a calendar closed and with the other opened is shown.

A.8 HTML Heading

A HTML heading is simply a title or subtitle that can be displayed with various sizes and alignments. The html heading sizes run from 1 through to 6 with 1 being the largest and 6 the smallest. The alignments include left, center and right alignment with the left being default if one does not specify this attribute. HTML code is shown code block 9 that produces some html headings [14].

```
<h1 align=”center”>Headings</h1>
<h2 align=”right”>are</h2>
      <h3>great</h3>..................................................................(9)
<h4 align=”left”>for</h4>
<h5 align=”right”>titles</h5>
<h6>and subtitles</h6>
```

Code 9 produces various html headings with various sizes and alignments shown in figure 29.



```
                          Headings
                                                          are
great
for
                                                       titles
and subtitles
```

Figure 29: Headings one through six with three different alignments are shown.

A.9 Label

Typically, form elements are associated with text. This text can be referred to as labels. To create a label, simply enclose the intended text inside the tags <label >and </label>. To associate these labels, the attribute “for” is used. The name form the form element is inserted inside the “for” attribute. If the label is devoid of this “for” attribute, then the label can simply be construed as text [15].

A.10 Submit and Reset Buttons

The reset button allows users of the form to reset all the form elements belonging to the form to their default values. The submit button allows users send the form to a specified address and conduct a set of particular specified actions. For instance, the action can be set to store all the results from the form into a database and divert to a particular website or send the form results via email to the designer and then divert to a particular website. The name attribute adds an internal name to the button so the form builder module handles the form and does not confuse the button with the other fields. The value attribute sets what is written on the button. It can be referred to as the label. HTML code that produces both a reset and submit button is expressed in code block 10 [16].

```
<input type="reset" value="Reset All fields!">
<input type="submit" value="Send me your name!">............................(10)
```

Code block 10 produces submit and reset buttons next to each other shown in figure 30.

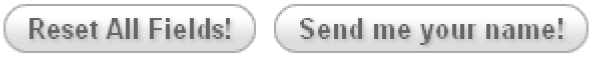


Figure 30: The reset and submit buttons next to each other are shown.

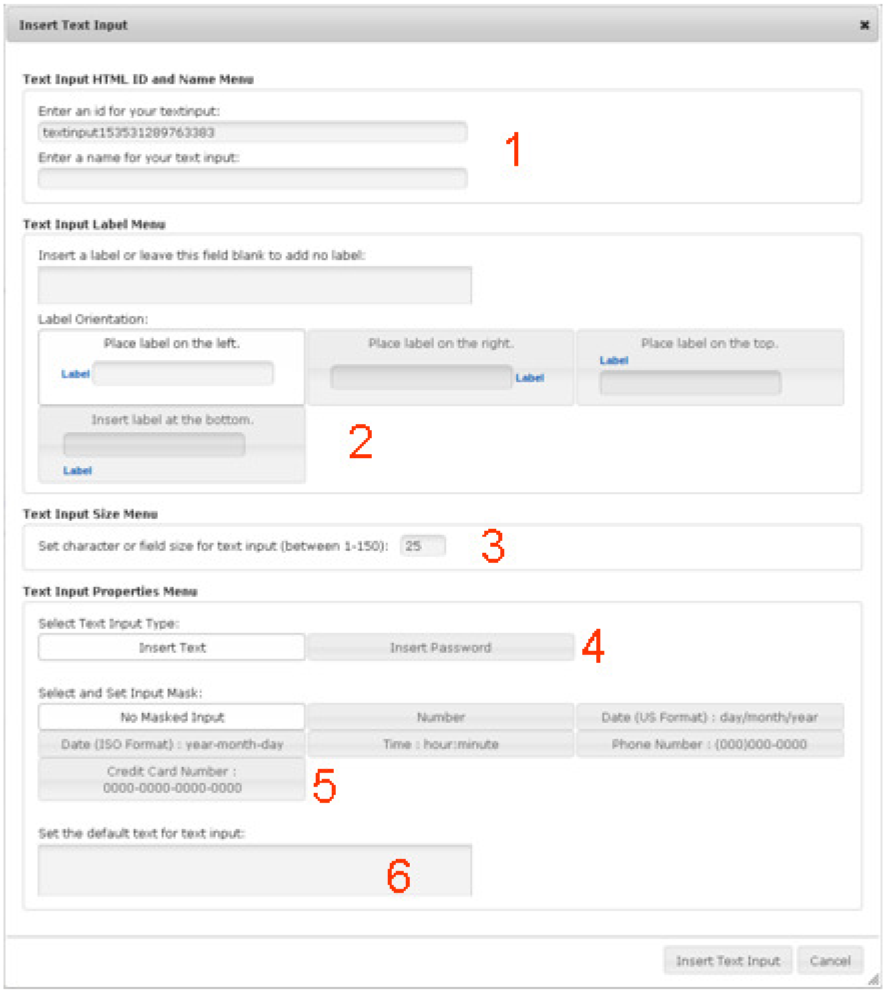
the check box element, the checked attribute will allow designer to check only one option from a group of alternative options as a default [17].

B. Form Element Insertion Modal Windows

B.1. Text Input

This modal window inserts a single text input field. When the "Insert Text Input" button is clicked on, if the fields completed inside this modal window are correct, this modal window will close and insert a new text input field. If some fields are not filled in correctly, an error message will be displayed on top of the modal window and the fields which contain the input error will be highlighted red. The main sections of this modal window displayed in figure 31 are numbered and explained below.

1. The insert id field will use the contents of this field and set the id attribute in the html div container that holds the text input element. This field value needs to be unique within this form. The name of the text input field will take the contents of this field and set the name attribute of the text input field. This value needs to be unique as no other form elements within this form can have this value. Additionally, this value needs to be intuitive, as it will be displayed in the submission of results section with the value that the user of the form has submitted.
2. This section will allow designers to insert a label for the text input element. If designers opt not to add a label for their text input fields then simply leave the text area blank. Otherwise, insert the desired text in the text area to be shown as a label. The label orientation radio buttons allow designers to select the orientation of the label with respect to the text input field. Images inside these radio buttons allow users to visualize how the label orientation would look like.
3. This section allows designers to set the width of the text input field. The default size is 25. Any number outside 1 and 150 will result in an error and has to be rectified.
4. The input text type radio button allows designers to choose between a text field and a password field. If a password field is chosen, the input mask option and the default text input field will automatically hide away since it is not required for a password field.
5. The input mask radio buttons sets the input mask type of the text input field. The no input mask option is selected as default.
6. Default text for the text input can be set by entering it inside this text area. If designers choose to leave the text input field blank, simply leave this field blank.



Insert Text Input

Text Input HTML ID and Name Menu

Enter an id for your textinput: textinput153531289763383

Enter a name for your text input:

Text Input Label Menu

Insert a label or leave this field blank to add no label:

Label Orientation: Place label on the left.

Text Input Size Menu

Set character or field size for text input (between 1-150): 25

Text Input Properties Menu

Select Text Input Type: Insert Text

Select and Set Input Mask: No Masked Input

Set the default text for text input:

×
1
Place label on the left.
Label
Place label on the right.
Label
Place label on the top.
Label
Insert label at the bottom.
Label
2
3
Insert Text
Insert Password
4
No Masked Input
Number
Date (US Format) : day/month/year
Date (ISO Format) : year-month-day
Time : hour:minute
Phone Number : (000)000-0000
Credit Card Number : 0000-0000-0000-0000
5
6
Insert Text Input
Cancel

Figure 31: The insert text input modal window is shown.

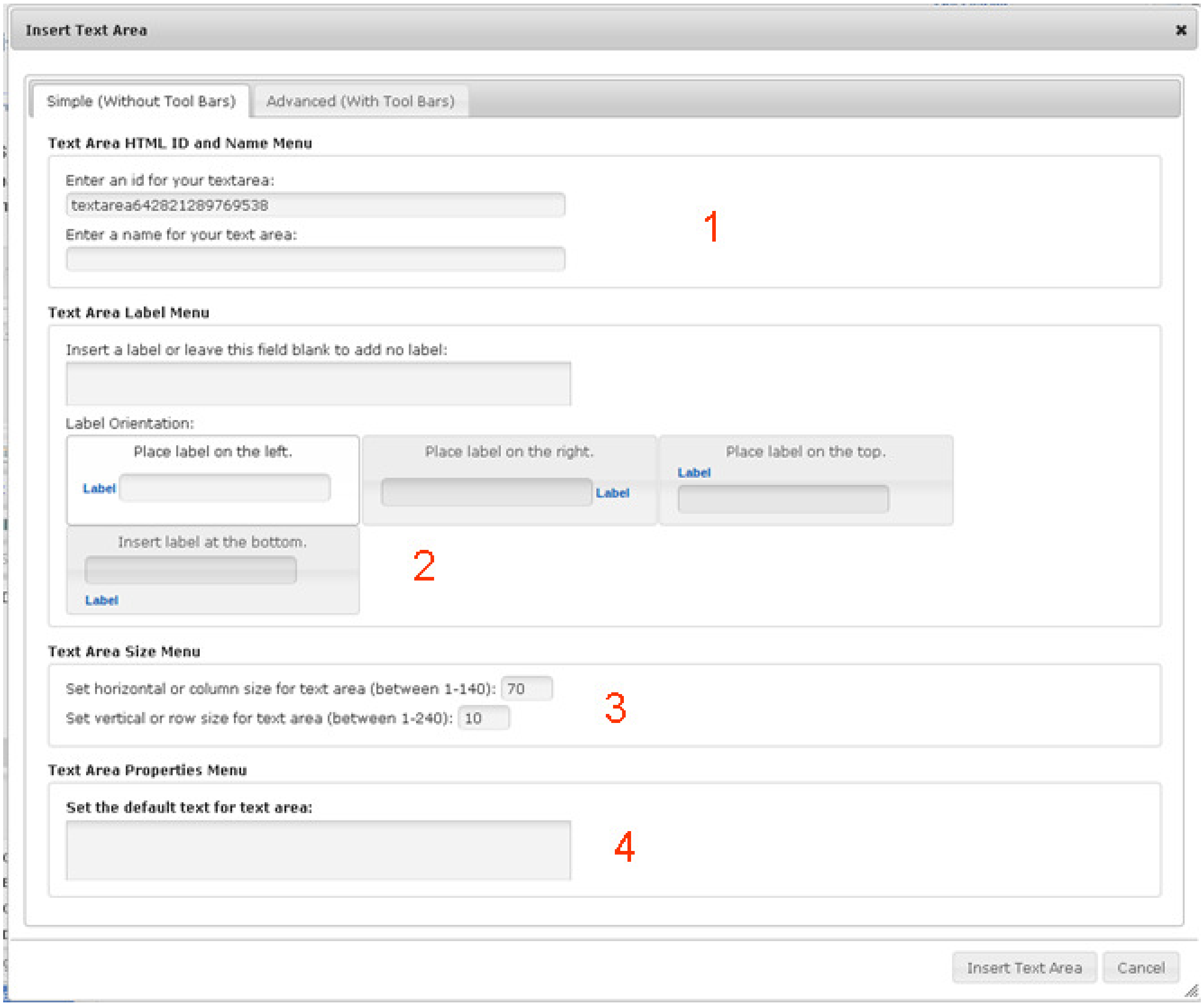
B.2. Text Area

This modal window inserts a single text area field. When the "Insert Text Area" button is clicked on, if the fields completed inside this modal window are correct, this modal window will close and insert a new text area field. If some fields are not filled in correctly, an error message will be displayed on top of the modal window and the fields which contain the input error will be highlighted red. The main sections of this modal window displayed in figure 32 are numbered and explained below. Note that there are two tabs on top. If the simple tab is selected, a text area without toolbars on top will be inserted. If the advanced tab is selected, a text area with toolbars on top will be displayed. The only difference between the two tabs is that in the advanced tab there is a radio button entity that allows the choosing of a toolbar for the text area.

1. The insert id field will use the contents of this field and set the id attribute in the html div container that holds the text area element. This field value needs to be unique within this form. The name of the text area field will take the contents of this field and set the name attribute of the text area field. This value needs to be unique as no other form elements within this form can have this value. Additionally, this value needs to be intuitive, as it will be displayed in the submission of results section with the value that the user of the form has submitted.
2. This section will allow designers to insert a label for the text area element. If designers opt not to add a label for their text area fields then simply leave the text area blank. Otherwise, insert the desired text in the text area to be shown as a label. The label orientation radio buttons allows

designers to select the orientation of the label with respect to the text area. Images inside these radio buttons allow users to visualize how the label orientation would look like.

3. This section allows designers to set the width and height of the text area field. The default sizes are 70 and 10 for column and row sizes. The columns field will set the number of text characters that can be displayed. The rows field will set the number of rows of text that can be displayed.
4. Default text for the text area can be set by entering it inside this text area. If designers choose to leave the text area field blank, simply leave this field blank.


Insert Text Area
×
Simple (Without Tool Bars)
Advanced (With Tool Bars)
Text Area HTML ID and Name Menu
1
Text Area Label Menu
Place label on the left.
Label
Place label on the right.
Label
Place label on the top.
Label
Insert label at the bottom.
Label
2
Text Area Size Menu
3
Text Area Properties Menu
4
Insert Text Area
Cancel

Enter an id for your textarea: textarea642821289769538

Enter a name for your text area:

Insert a label or leave this field blank to add no label:

Label Orientation:

Set horizontal or column size for text area (between 1-140): 70

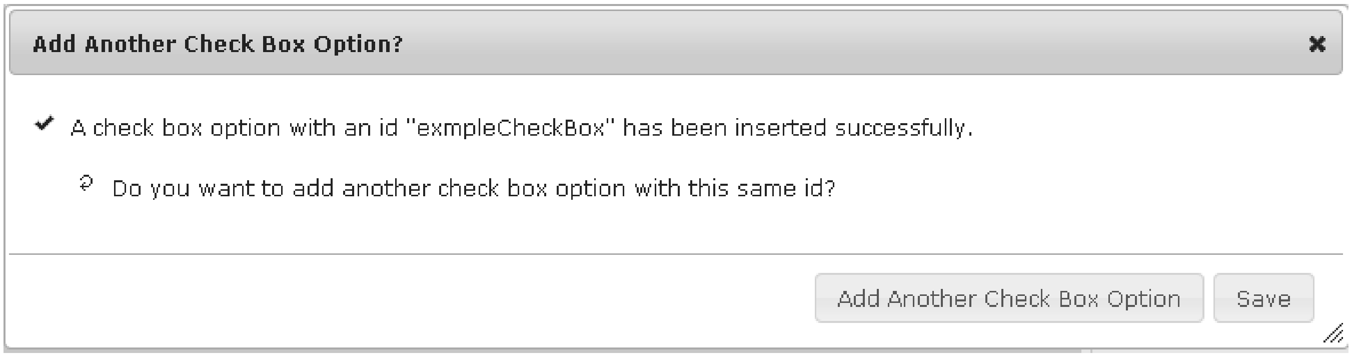
Set vertical or row size for text area (between 1-240): 10

Set the default text for text area:

Figure 32: The insert text area modal window is shown.

B.3. Check Box

This modal window inserts check box elements within a single html div container. When the "Insert Check Box Option" button is clicked on, if the fields completed inside this modal window are correct, this modal window will close and insert a new check box. If some fields are not filled in correctly, an error message will be displayed on top of the modal window and the fields which contain the input error will be highlighted red. Upon the successful insertion of a check box option, another modal window displayed in figure 33 will ask whether or not to add more check box options. If the designer is done inserting check boxes, simply click on the “save” button otherwise click the “add another check box option” button to add more options.



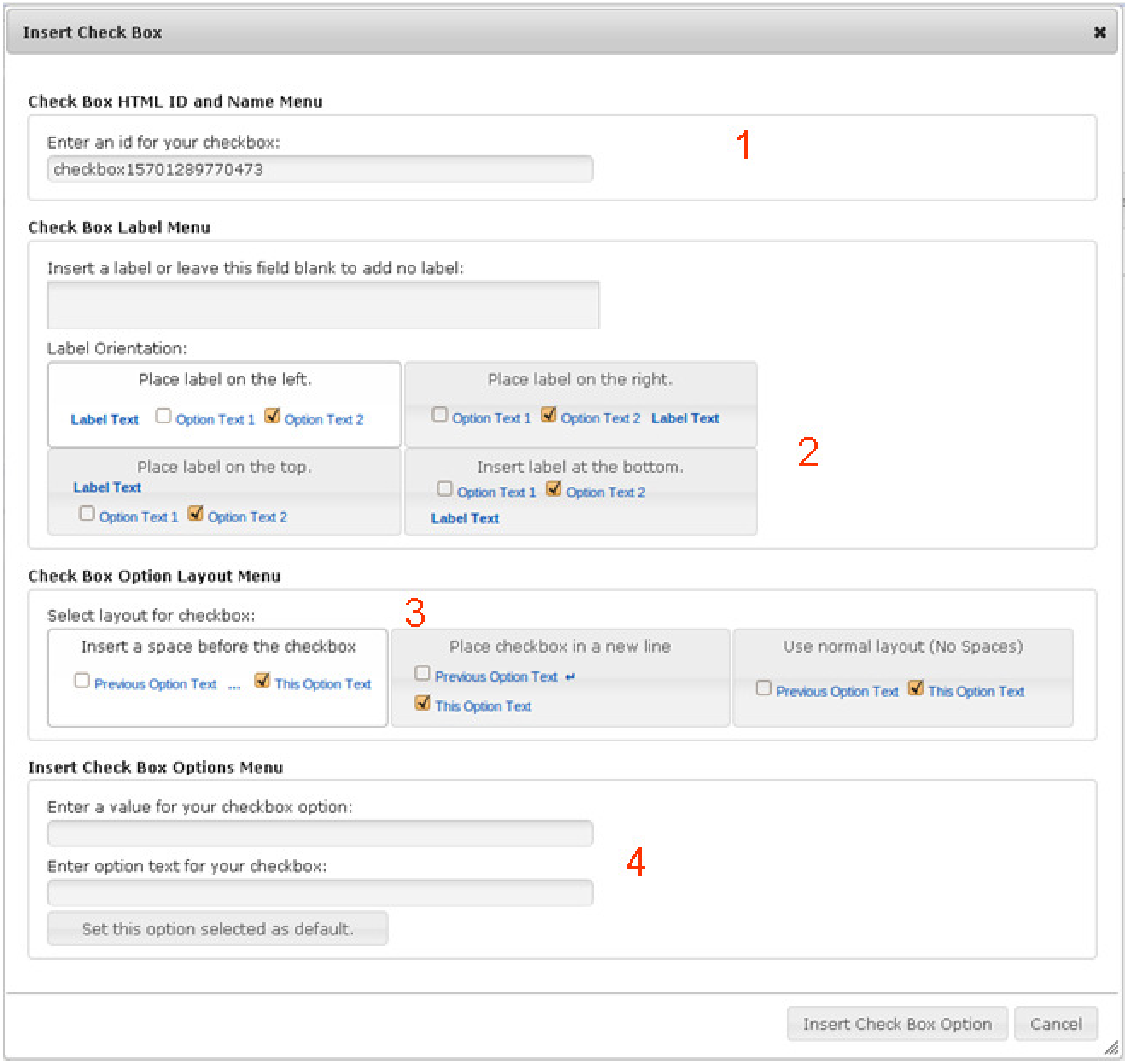
✓ A check box option with an id "exmpleCheckBox" has been inserted successfully.

? Do you want to add another check box option with this same id?

Figure 33: The “Add another Check Box Option” modal window is shown.

The main sections of this modal window displayed in figure 34 are numbered and explained below.

1. The insert id field will use the contents of this field and set the id attribute in the html div container that hold the check box options. This field value needs to be unique within this form.
2. This section will allow designers to insert a label for the check box options. If designers opt not to add a label for their check box options then simply leave the text area blank. Otherwise, insert the desired text in the text area to be shown as a label. The label orientation radio buttons allows designers to select the orientation of the label with respect to the check box options. Images inside these radio buttons allow users to visualize how the label orientation would look like.
3. This section allows designers to set what layout to insert their check box option. Three different spacing and orientations can be chosen. Images inside the radio buttons will allow designers to visualize what the layout of the checkbox option would look like.
4. This section will allow designers to set the value and text of the check box option. It is imperative not to confuse and conflate these two entities together. The value of the check box option is actually the name attribute which adds an internal name to the check box option so the form builder can uniquely identify it apart form other fields. Additionally, this value needs to be intuitive, as it will be displayed in the submission of results section with the value that the user of the form has submitted. The value of the check box element should not be confused with the option text. The option text is what gets displayed next to the check box. It could be thought of as a label for the check box. The button “set this option selected as default” will apply the attribute “checked” to the option to default check this option.



Insert Check Box ✖

Check Box HTML ID and Name Menu

Enter an id for your checkbox: checkbox15701289770473

Check Box Label Menu

Insert a label or leave this field blank to add no label:

Label Orientation:

- Place label on the left. — Label Text □ Option Text 1 ☑ Option Text 2
- Place label on the right. — □ Option Text 1 ☑ Option Text 2 Label Text
- Place label on the top. — Label Text □ Option Text 1 ☑ Option Text 2
- Insert label at the bottom. — □ Option Text 1 ☑ Option Text 2 Label Text

Check Box Option Layout Menu

Select layout for checkbox:

- Insert a space before the checkbox — □ Previous Option Text ... ☑ This Option Text
- Place checkbox in a new line — □ Previous Option Text ↵ ☑ This Option Text
- Use normal layout (No Spaces) — □ Previous Option Text ☑ This Option Text

Insert Check Box Options Menu

Enter a value for your checkbox option:

Enter option text for your checkbox:

Set this option selected as default.

Insert Check Box Option | Cancel

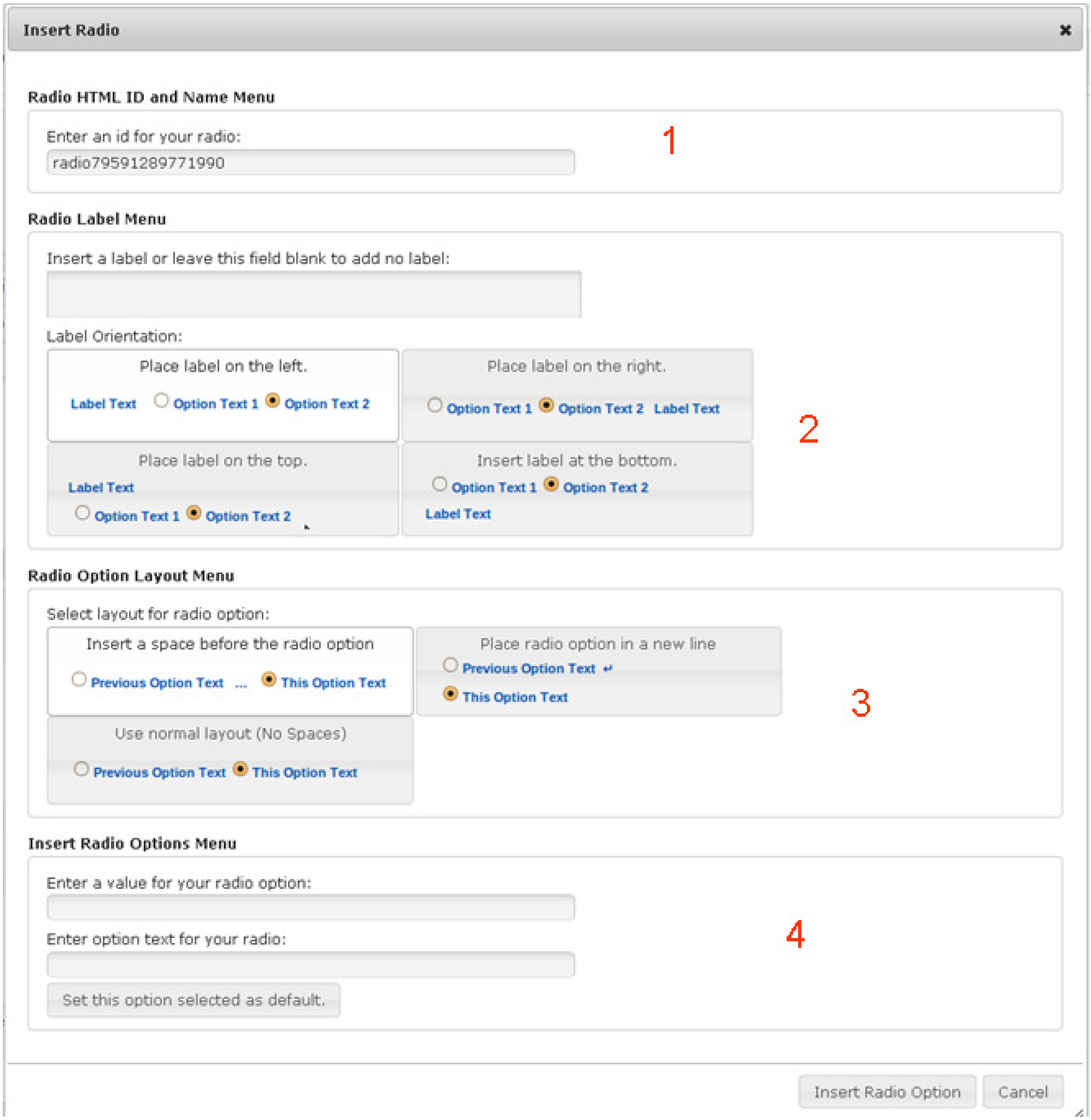
Figure 34: The insert check box option modal window is shown.

B.4. Radio Button

This modal window inserts radio button elements within a single html div container. When the "Insert Radio Option" button is clicked on, if the fields completed inside this modal window are correct, this modal window will close and insert a new radio button. If some fields are not filled in correctly, an error message will be displayed on top of the modal window and the fields which contain the input error will be highlighted red. Upon the successful insertion of a radio button option, another modal window like the one displayed in figure 33 will ask whether or not to add more radio button options. If the designer is done inserting radio buttons, simply click on the "save" button otherwise click the "add another radio button option" button to add more options.

The main sections of this modal window displayed in figure 35 are numbered and explained below.

1. The insert id field will use the contents of this field and set the id attribute in the html div container that hold the radio button options. This field value needs to be unique within this form. This id field will also serve as setting the name attribute of the radio button. Similar to other form elements, the name attribute specifies a group of radio buttons the field belongs to. When the user selects one option; this causes all the other options become unselected. Additionally, this value needs to be intuitive, as it will be displayed in the submission of results section with the value that the user of the form has submitted.
2. This section will allow designers to insert a label for the radio button options. If designers opt not to add a label for their radio button options then simply leave the text area blank. Otherwise, insert the desired text in the text area to be shown as a label. The label orientation radio buttons allows designers to select the orientation of the label with respect to the radio button options. Images inside these radio buttons allow users to visualize how the label orientation would look like.
3. This section allows designers to set what layout to insert their radio button option. Three different spacing and orientations can be chosen. Images inside the radio buttons will allow designers to visualize what the layout of the radio button option would look like.
4. This section will allow designers to set the value and text of the radio button option. It is imperative not to confuse and conflate these two entities together. The value of the radio button option is what gets submitted when this option gets selected. The value of the radio button option should not be confused with the option text. The option text is what gets displayed next to the radio button. It could be thought of as a label for the radio button. The button “set this option selected as default” will apply the attribute “checked” to the option to default check this option. Once an option is set to be default selected, no other options can be set checked as default.


Insert Radio

Radio HTML ID and Name Menu

Enter an id for your radio: radio79591289771990

Radio Label Menu

Insert a label or leave this field blank to add no label:

Label Orientation:

Place label on the left.
Label Text ○ Option Text 1 ● Option Text 2

Place label on the right.
○ Option Text 1 ● Option Text 2 Label Text

Place label on the top.
Label Text
○ Option Text 1 ● Option Text 2

Insert label at the bottom.
○ Option Text 1 ● Option Text 2
Label Text

Radio Option Layout Menu

Select layout for radio option:

Insert a space before the radio option
○ Previous Option Text ... ● This Option Text

Place radio option in a new line
○ Previous Option Text ↵
● This Option Text

Use normal layout (No Spaces)
○ Previous Option Text ● This Option Text

Insert Radio Options Menu

Enter a value for your radio option:

Enter option text for your radio:

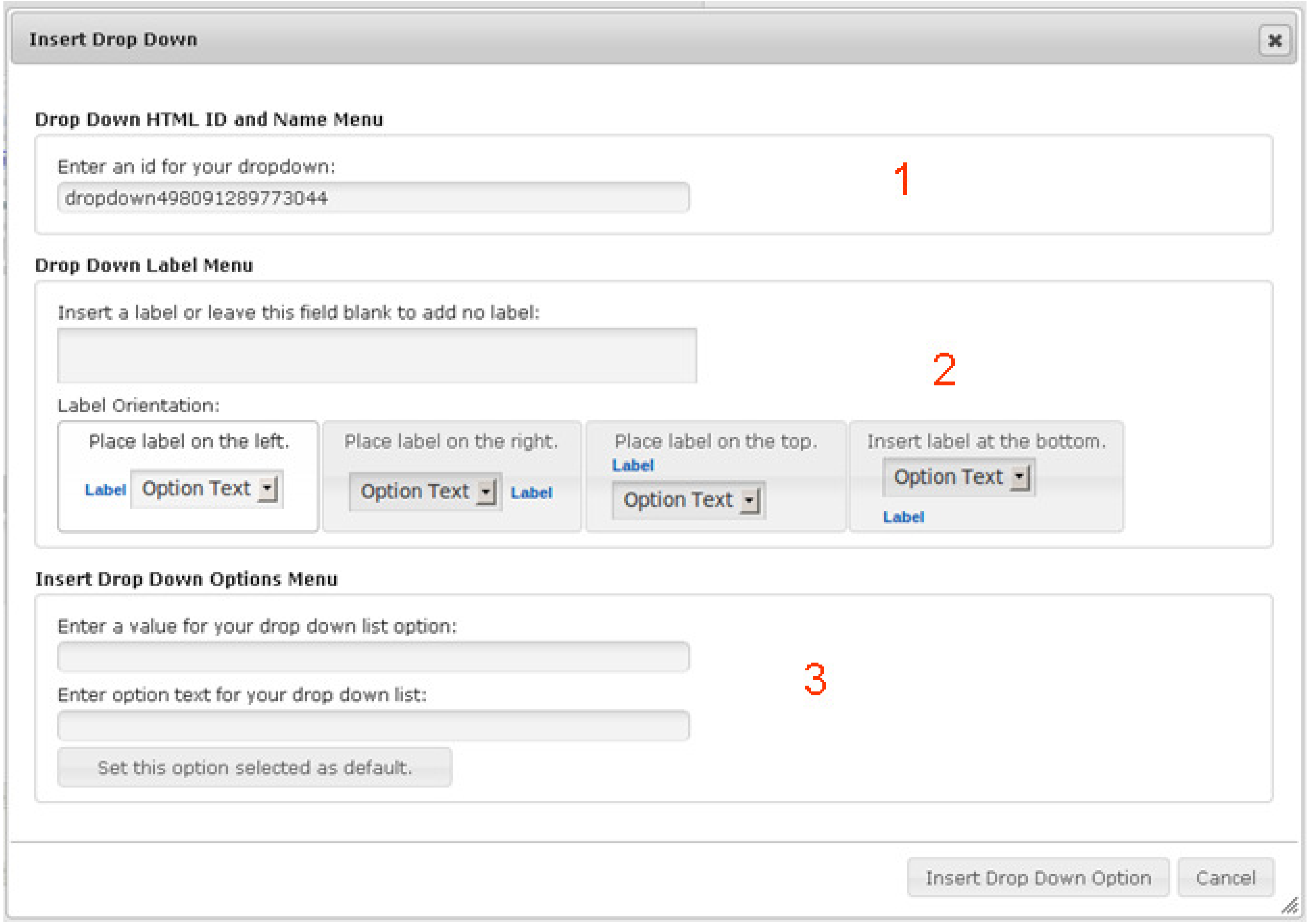
Figure 35: The insert radio button option modal window is shown.

B.5. Drop Down List

This modal window inserts drop down lists within a single html div container. When the “Insert Drop down Option” button is clicked on, if the fields completed inside this modal window are correct, this modal window will close and insert a new drop down list button. If some fields are not filled in correctly, an error message will be displayed on top of the modal window and the fields which contain the input error will be highlighted red. Upon the successful insertion of a drop down option, another modal window like the one displayed in figure 33 will ask whether or not to add more drop down list options. If the designer is done inserting drop down options, simply click on the “save” button otherwise click the “add another drop down option” button to add more options.

The main sections of this modal window displayed in figure 36 are numbered and explained below.

1. The insert id field will use the contents of this field and set the id attribute in the html div container that holds the drop down menu. This field value needs to be unique within this form. This id field will also serve as setting the name attribute of the drop down list. Similar to other form elements, the name attribute specifies a group of drop down options the field belongs to. When the user selects one option; this causes all the other options become unselected and dormant. Additionally, this value needs to be intuitive, as it will be displayed in the submission of results section with the value that the user of the form has submitted.
2. This section will allow designers to insert a label for the drop down menu. If designers opt not to add a label for their drop down menu then simply leave the text area blank. Otherwise, insert the desired text in the text area to be shown as a label. The label orientation radio buttons allows designers to select the orientation of the label with respect to the drop down menu. Images inside these radio buttons allow users to visualize how the label orientation would look like.
3. This section will allow designers to set the value and text of the drop down option. It is imperative not to confuse and conflate these two entities together. The value of the drop down option is what gets submitted when this option gets selected. The value of the drop down option should not be confused with the option text. The option text is what gets displayed inside the drop down menu as an option. It could be thought of as a label for the drop down option. The button “set this option selected as default” will apply the attribute “selected” to the option to default select this option. Once an option is set to be default selected, no other options can be set selected as default.


Insert Drop Down
Drop Down HTML ID and Name Menu
1
Drop Down Label Menu
Insert a label or leave this field blank to add no label:
2
Label Orientation:
Place label on the left.
Label
Option Text
Place label on the right.
Option Text
Label
Place label on the top.
Label
Option Text
Insert label at the bottom.
Option Text
Label
Insert Drop Down Options Menu
Enter a value for your drop down list option:
Enter option text for your drop down list:
3
Set this option selected as default.
Insert Drop Down Option
Cancel

Enter an id for your dropdown: dropdown498091289773044

Figure 36: The insert drop down list modal window is shown.

B.6. Multi-Selectable Drop Down List

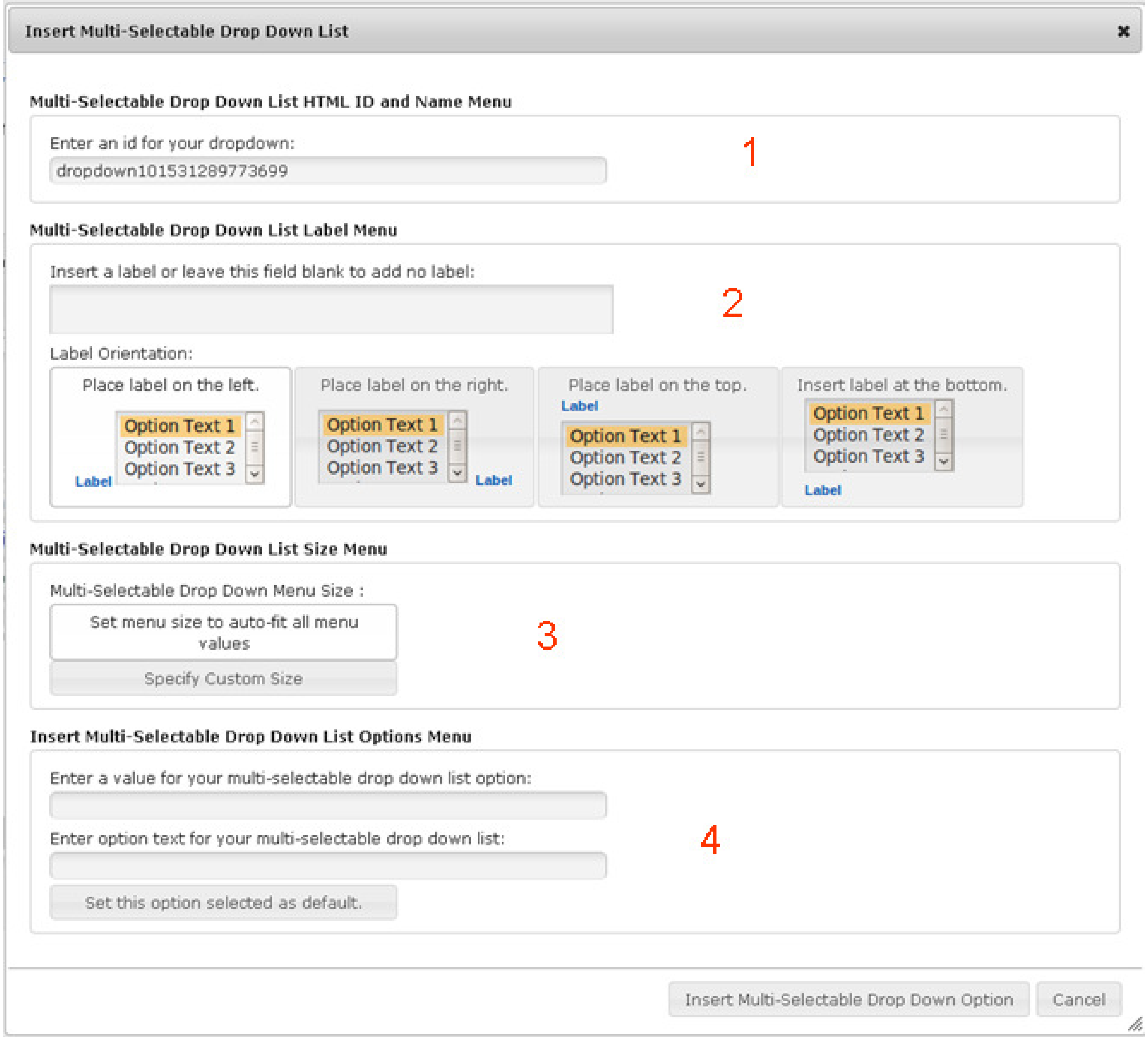
This modal window inserts multi-selectable drop down lists within a single html div container. When the "Insert Multi-Selectable Drop down Option" button is clicked on, if the fields completed inside this modal window are correct, this modal window will close and insert a new drop down list button. If some fields are not filled in correctly, an error message will be displayed on top of the modal window and the fields which contain the input error will be highlighted red. Upon the successful insertion of a drop down option, another modal window like the one displayed in figure 33 will ask whether or not to add more drop down list options. If the designer is done inserting drop down options, simply click on the "save" button otherwise click the "add another drop down option" button to add more options.

The main sections of this modal window displayed in figure 37 are numbered and explained below.

1. The insert id field will use the contents of this field and set the id attribute in the html div container that holds the drop down menu. This field value needs to be unique within this form. This id field will also serve as setting the name attribute of the drop down list. Similar to other form elements, the name attribute specifies a group of drop down options the field belongs to. Additionally, this value needs to be intuitive, as it will be displayed in the submission of results section with the value that the user of the form has submitted.
2. This section will allow designers to insert a label for the drop down menu. If designers opt not to add a label for their drop down menu then simply leave the text area blank. Otherwise, insert the

desired text in the text area to be shown as a label. The label orientation radio buttons allows designers to select the orientation of the label with respect to the drop down menu. Images inside these radio buttons allow users to visualize how the label orientation would look like.

3. This section sets the size of the drop down menu. A radio button is given with two options. The auto-fit option will fit all options with the menu and the menu is automatically grown and fit all the options enabling to be visible. The other option is to specify the menu size. When this option is selected a text input will appear in which the user will be asked to enter any number greater than 1. This number will set the number of options that can be visible.
4. This section will allow designers to set the value and text of the drop down option. It is imperative not to confuse and conflate these two entities together. The value of the drop down option is what gets submitted when this option gets selected. The value of the drop down option should not be confused with the option text. The option text is what gets displayed inside the drop down menu as an option. It could be thought of as a label for the drop down option. The button "set this option selected as default" will apply the attribute "selected" to the option to default select this option.



Insert Multi-Selectable Drop Down List

Multi-Selectable Drop Down List HTML ID and Name Menu

Enter an id for your dropdown: dropdown101531289773699

Multi-Selectable Drop Down List Label Menu

Insert a label or leave this field blank to add no label:

Label Orientation:

Multi-Selectable Drop Down List Size Menu

Multi-Selectable Drop Down Menu Size :

Insert Multi-Selectable Drop Down List Options Menu

Enter a value for your multi-selectable drop down list option:

Enter option text for your multi-selectable drop down list:

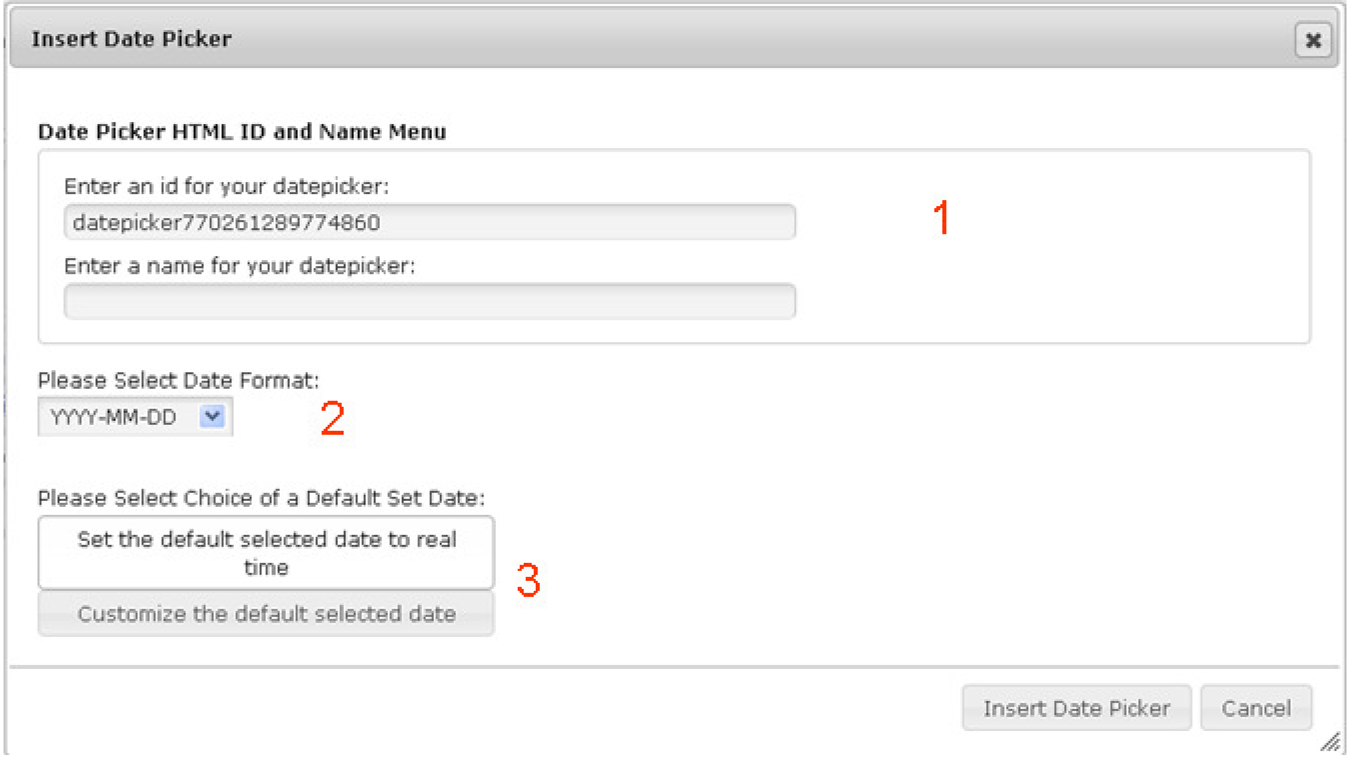
×
1
2
Place label on the left.
Place label on the right.
Place label on the top.
Insert label at the bottom.
Label
Option Text 1
Option Text 2
Option Text 3
3
Set menu size to auto-fit all menu values
Specify Custom Size
4
Set this option selected as default.
Insert Multi-Selectable Drop Down Option
Cancel

Figure 37: The insert multi-selectable drop down list modal window is shown.

B.7. Date Picker Object

This modal window inserts a single date picker object. When the "Insert Date Picker" button is clicked on, if the fields completed inside this modal window are correct, this modal window will close and insert a new date picker object. If some fields are not filled in correctly, an error message will be displayed on top of the modal window and the fields which contain the input error will be highlighted red. The main sections of this modal window displayed in figure 38 are numbered and explained below.

1. The insert id field will use the contents of this field and set the id attribute in the html div container that holds the date picker object. This field value needs to be unique within this form. The name of the date picker field will take the contents of this field and set the name attribute of the date picker. This value needs to be unique as no other form elements within this form can have this value. Additionally, this value needs to be intuitive, as it will be displayed in the submission of results section with the value that the user of the form has submitted.
2. A date format has to be selected by the designer. Once this is set, dates that are filled in by users of the form will be saved and displayed in this format.
3. A real time or default date can also be selected. The real time is a default setting if no default date is specified and will show the current date by default. If a specific date is chosen, this date will be shown by default.



Insert Date Picker

Date Picker HTML ID and Name Menu

Enter an id for your datepicker: datepicker770261289774860

Enter a name for your datepicker:

Please Select Date Format: YYYY-MM-DD

Please Select Choice of a Default Set Date:

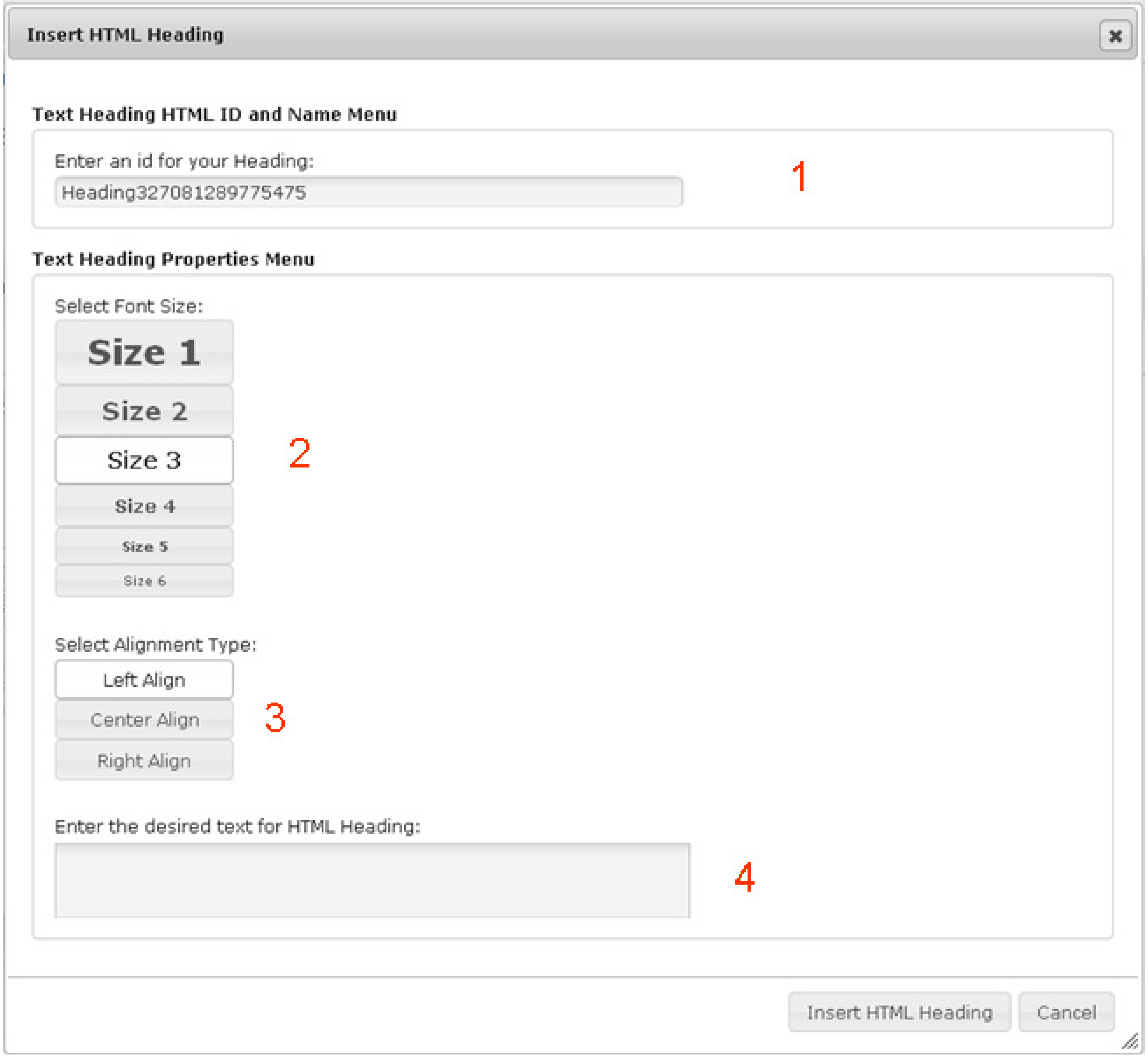
✖
1
2
3
Set the default selected date to real time
Customize the default selected date
Insert Date Picker
Cancel

Figure 38: The insert date picker object modal window is shown.

B.8. HTML Heading

This modal window inserts a single html heading. When the "Insert HTML Heading" button is clicked on, if the fields completed inside this modal window are correct, this modal window will close and insert a new html heading. If some fields are not filled in correctly, an error message will be displayed on top of the modal window and the fields which contain the input error will be highlighted red. The main sections of this modal window displayed in figure 39 are numbered and explained below.

1. The insert id field will use the contents of this field and set the id attribute in the html div container that holds the html heading. This field value needs to be unique within this form.
2. The html heading sizes run from 1 through to 6 with 1 being the largest and 6 the smallest. The designer has to select only one of these radio buttons.
3. The alignments include left, center and right alignment with the left being default if one does not specify this attribute.
4. This section allows designers to add the actual text for the HTML heading.



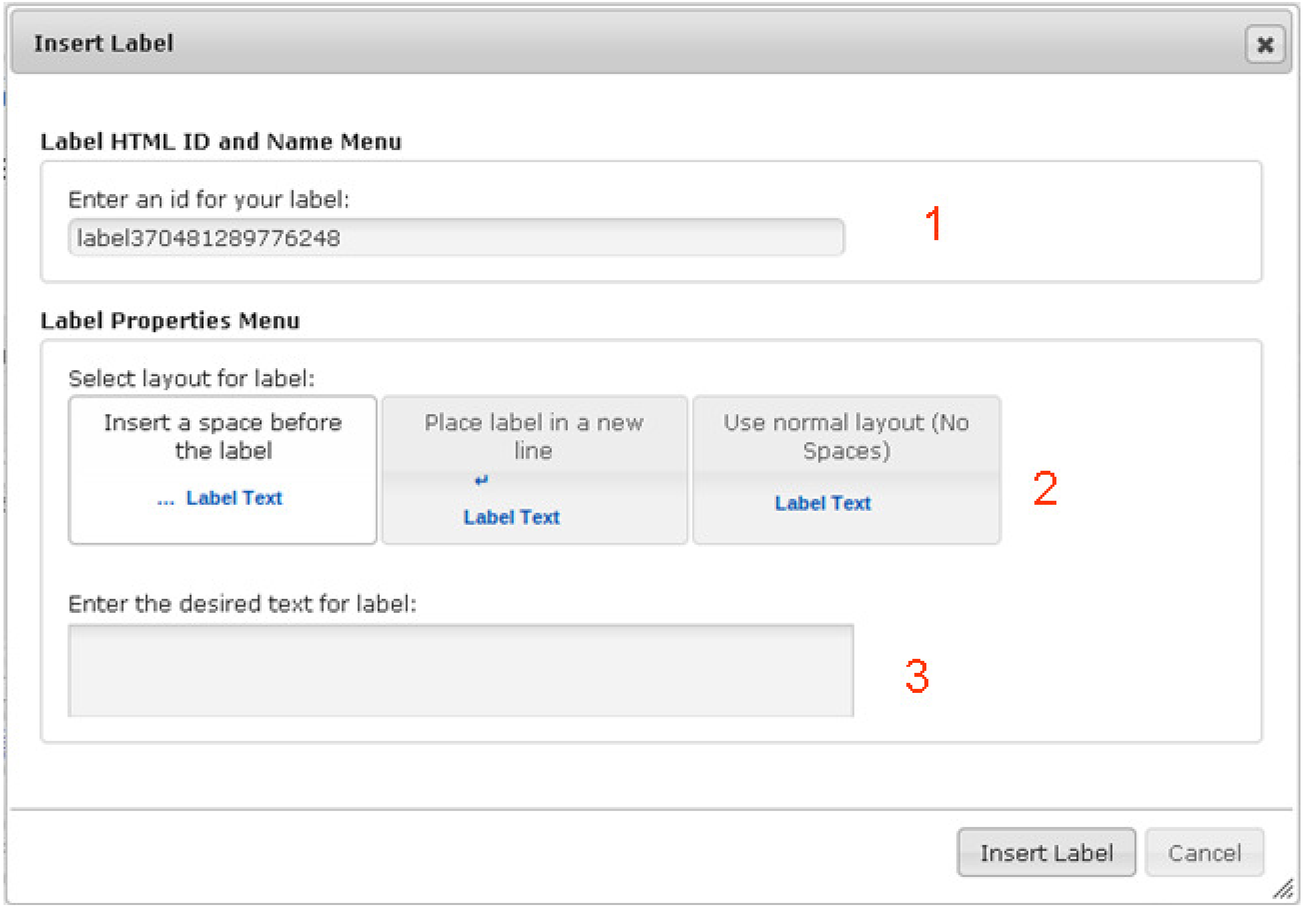
Enter an id for your Heading: Heading327081289775475

Figure 39: The insert HTML heading modal window is shown.

B.9. Label

This modal window inserts a single label. However, the attribute "for" will be set to null. Inserting this kind of label is similar to simply inserting normal text. When the "Insert Label" button is clicked on, if the fields completed inside this modal window are correct, this modal window will close and insert a new label or simply text. If some fields are not filled in correctly, an error message will be displayed on top of the modal window and the fields which contain the input error will be highlighted red. The main sections of this modal window displayed in figure 40 are numbered and explained below.

1. The insert id field will use the contents of this field and set the id attribute in the html div container that holds the label or simply text. This field value needs to be unique within this form.
2. This section allows designers to set what layout to insert their label. Three different spacing and orientations can be chosen. Images inside the radio buttons will allow designers to visualize what the layout of the label would look like.
3. This section allows designers to add the actual text for the label.



Insert Label

Label HTML ID and Name Menu

Enter an id for your label: label370481289776248

Label Properties Menu

Select layout for label:

Insert a space before the label
... Label Text
Place label in a new line
↵
Label Text
Use normal layout (No Spaces)
Label Text
1
2
3
Insert Label
Cancel

Enter the desired text for label:

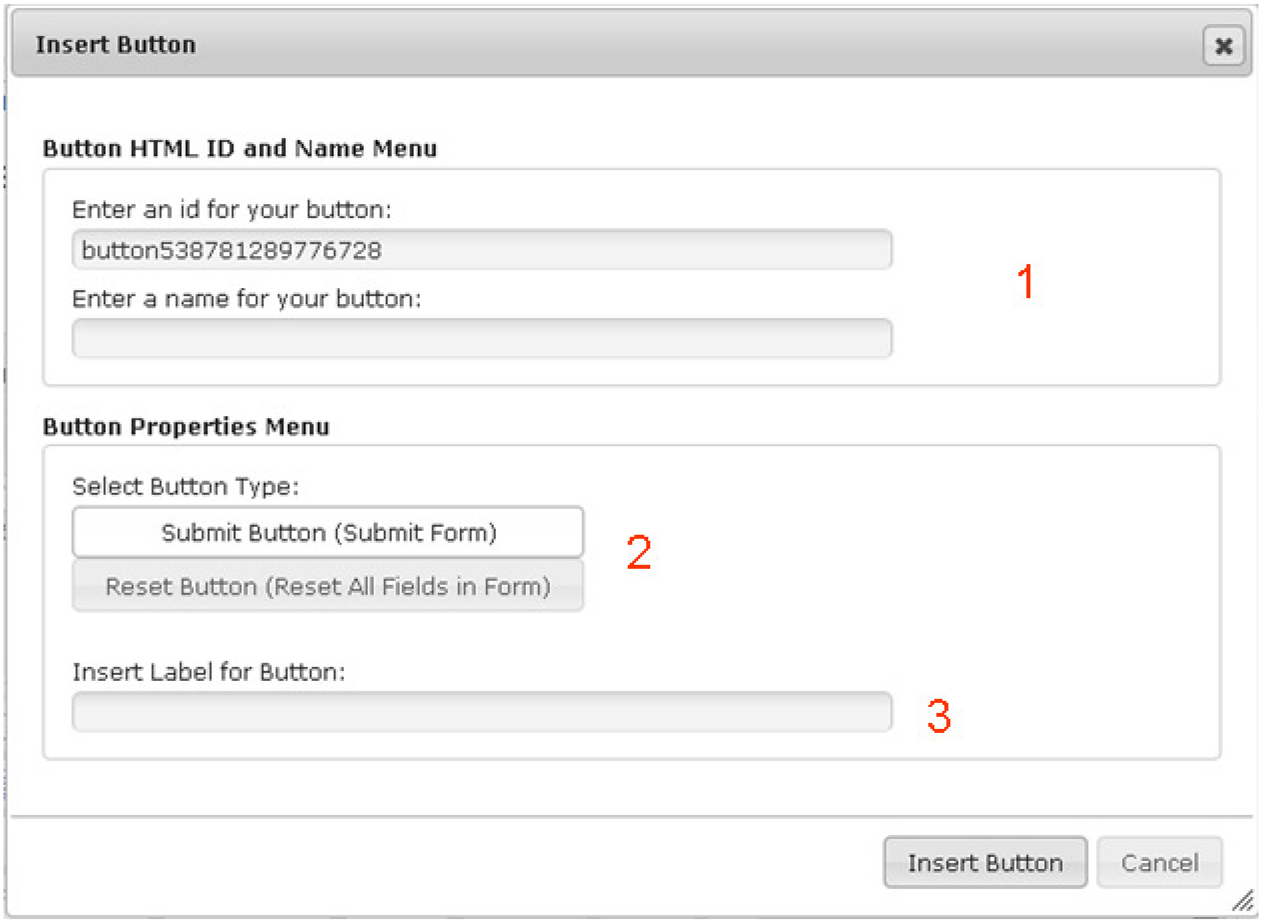
Figure 40: The insert label modal window is shown.

B.10. Submit or Reset Button

This modal window inserts submit or reset buttons within a single html div container. When the “Insert Button” button is clicked on, if the fields completed inside this modal window are correct, this modal window will close and insert a new submit or reset button. If some fields are not filled in correctly, an error message will be displayed on top of the modal window and the fields which contain the input error will be highlighted red. Upon the successful insertion of a button, another modal window like the one displayed in figure 33 will ask whether or not to add another button next to newly created one. If the designer is done inserting buttons, simply click on the “save” button otherwise click the “add another button” button to add another button next to the newly created one.

The main sections of this modal window displayed in figure 41 are numbered and explained below.

1. The insert id field will use the contents of this field and set the id attribute in the html div container that hold submit or reset buttons. This field value needs to be unique within this form. The name attribute adds an internal name to the button so the form builder module handles the form and does not confuse the button with the other fields.
2. This radio button permits either submit or reset buttons to be inserted.
3. The value attribute sets what is written on the button. It can be referred to as the label.


Enter an id for your button: button538781289776728

Insert Button
Button HTML ID and Name Menu
Enter a name for your button:
1
Button Properties Menu
Select Button Type:
Submit Button (Submit Form)
Reset Button (Reset All Fields in Form)
2
Insert Label for Button:
3
Insert Button
Cancel

Figure 41: The insert button modal window is shown.

References for Appendices

[1] http://www.w3schools.com/html/html_primary.asp, Last accessed 6 November 2010.

[2] http://www.w3schools.com/html/html_forms.asp, Last accessed 6 November 2010.

[3] http://echoecho.com/htmlforms05.htm, Last accessed 6 November 2010.

[4] http://www.w3schools.com/TAGS/tag_textarea.asp, Last accessed 6 November 2010.

[5] http://echoecho.com/htmlforms08.htm, Last accessed 6 November 2010.

[6] http://echoecho.com/htmlforms09.htm, Last accessed 6 November 2010.

[7] http://www.tizag.com/htmlT/htmlcheckboxes.php, Last accessed 6 November 2010.

[8] http://html-codes.1keydata.com/html-radio-button.php, Last accessed 6 November 2010.

[9] http://echoecho.com/htmlforms10.htm, Last accessed 6 November 2010.

[10] http://echoecho.com/htmlforms11.htm, Last accessed 6 November 2010.

[11] http://www.htmlcodetutorial.com/linking/linking_famsupp_114.html, Last accessed 6 November 2010.

[12] http://www.w3schools.com/TAGS/tag_Select.asp, Last accessed 6 November 2010.

[13] http://jqueryui.com/demos/datepicker/, Last accessed 6 November 2010.

[14] http://www.tizag.com/htmlT/htmlheadings.php, Last accessed 6 November 2010.

[15] http://www.w3schools.com/tags/tag_label.asp, Last accessed 6 November 2010.

[16] http://echoecho.com/htmlforms12.htm, Last accessed 6 November 2010.

[17] http://echoecho.com/htmlforms13.htm, Last accessed 6 November 2010.