Creating accessible forms

Jon Whiting
webaim.org
@jonwhiting
3 steps

1. Select the correct form control.
2. Ensure instructions, labels, and cues are accessible to all users.
3. Minimize user error and facilitate error recovery.
1. Correct input type
Select from the following to make group reservations.

☐ School Group
☐ Other Group
☐ Tour Operator
What if there isn’t a standard form control?
“Standardize” the interaction

Maximum bid: $72

$10  $99
• Arrow keys to navigate
• Use ARIA to tell the screen reader what it is/does
What if it isn’t standardized
2. Form labeling
Form Labels

First Name: 

<label for="fname">First Name:</label>
<input type="text" id="fname"/>

Text boxes, text areas, select menus, checkboxes, and radio buttons.
Implicit Form Labels

<label>Name:
<input type="text"></label>
<fieldset> and <legend>

Choose a shipping method:

- Overnight
- Two day
- Ground

<fieldset>
<legend>Choose a shipping method:</legend>
<input id="overnight" type="radio"/>
<label for="overnight">Overnight</label>
...
</fieldset>
Limitations with `<label>`

1:1 relationship between `<label>` and a form control.

- A `<label>` cannot apply to more than one form control.
- A form control cannot have more than one `<label>`.
ARIA (Accessible Rich Internet Applications) can be used to overcome this limitation.
First Name: 

```html
<span id="fnamelabel">First name:</span>
<input type="text" aria-labelledby="fnamelabel" />
```
<label for="fname" id="fnamelabel">First name:</label>

<input type="text" id="fname" aria-labelledby="fnamelabel">

Redundant, but does not cause any problems
One label for multiple controls

<input type="text" name="name1" id="namelabel" aria-labelledby="namelabel"/>

<table>
<thead>
<tr>
<th>Name</th>
<th>Age</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Multiple labels for one control

<table>
<thead>
<tr>
<th>Name</th>
<th>Office Number</th>
<th>Phone</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cyndi</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jared</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jonathan</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<input type="text" name="office1" aria-labelledby="cyndi officenum">
Form values as labels for other inputs

<table>
<thead>
<tr>
<th>Name</th>
<th>Office Number</th>
<th>Phone</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cyndi</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jared</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jonathan</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Username: <input type="text" id="user" aria-describedby="usernamedetails"/>

<p id="usernamedetails">Username must be 8-15 characters</p>
Password:* 

<label for="password">Password:</label>*
<input type="password" id="password" aria-required="true">

A screen reader indicates the field is required... and that's all.
A screen reader indicates the field is invalid or broken... and that’s all.

Use ARIA attributes to control styling:

```
[aria-invalid=true] {border : 2px solid red;}
```
'Invisible' Labels

1. Hidden `<label>` or
2. The `title` attribute or
3. The `aria-label` attribute
Hidden <label>

<label class="hidden" for="s">
Search Terms</label>
<input type="text" id="s"/>
Visually Hiding Content

• `display:none` and `visibility:hidden` hide from everyone
• Avoid 0 pixels, same color as background, etc.
• Position off-screen left or clip with CSS for screen readers
• Use judiciously
Visually Hiding Content

.hidden {
  position:absolute;
  left:-10000px;
  top:auto;
  width:1px;
  height:1px;
  overflow:hidden;
}
Title Attribute

<input title="Search Terms"...>
Title Attribute

• Advisory information only
• Ignored by screen readers, except...
  – form elements missing labels
  – `<frame title="navigation">`
  – `<acronym>/ `<abbr>`...usually
• Not accessible to keyboard users, touch screen users, etc.
aria-label Attribute

<input aria-label="Search Terms"...>
Do **NOT** use only Placeholder as a Label

<input placeholder="Search Terms"...>

(Placeholder can be used, just not in place of a label)
Other Form Issues

- Avoid JavaScript “jump” menus
- Be very careful with autofocus attribute or scripting
- Be very careful with “Reset” buttons
- Avoid multiple select menus
Form Validation and Error Recovery

• Avoid
  – Make the form as simple as possible
  – Only collect necessary information
  – Do not force users to provide formatting unless necessary

• Alert
  – Let the users know there is an error
  – Direct users directly to errors

• Advise
  – What is the error. How can you fix it.
Form Validation Types

- Errors at the top
- Alert, then focus
- Inline errors
Errors at the top

There was a problem with the form.

- Please enter a subject or choose a subject from the drop-down menu.
Alert, then focus

Please enter a value for the "NAME" field.
Password

Please enter at least 8 characters.

Please enter at least 1 number or symbol (!@#$%^).
Other Form Issues

• Disabled controls have very low contrast and cannot receive keyboard focus
  – Use `aria-disabled="true"` instead

• Ensure image buttons have alt text
  `<input type="image" src="submit.jpg" alt="Search">`

• Ensure standard buttons have a value
  `<input type="submit" value="Search">
  <button>Search</button>`
Thank You!

webaim.org

- Web based forums
- E-mail discussion list
- Tutorials, articles, and resources
- Blog