

## **Should I use a drop-down? Four steps for choosing form elements on the Web**

*Sarah Miller and Caroline Jarrett*

*You are designing a form for the Web and you want to make it easy for your form user to choose answers to your questions. There are five HTML form elements available: drop-down box, radio buttons, check boxes, type-in box and hyperlinks. Which one is the most usable?*

*We present here a four-step process for choosing form elements.*

*First, consider whether the dominant purpose of your page is for navigation or for information gathering.*

*Second, ask yourself six questions designed to narrow down the choice of form element for each question on your form.*

*Third, look at the impact of your choice on the form as a whole. Choose a sensible order for the options, keep the options short and avoid too many different input methods.*

*Fourth, think about your users and their interaction with the browser technology. Will the user be able to access the choice you make?*

### **INTRODUCTION**

Most of us have experienced the frustration of inappropriate use of drop-downs. Who enjoys choosing their country from a drop-down list of over 100? What about scrolling through 31 days to choose the day for a date? But a drop-down can be a convenient way of showing choices to users, and they can help to make your form look smaller and less daunting.

We have developed a four-stage process for choosing between the five types of HTML forms elements. Of course, it is possible to create much more sophisticated forms with the use of other technologies such as Java applets or image maps embedded in your web page — but as these technologies can create further problems, we will assume here that you want to stick with HTML.

To illustrate our process, we have chosen some questions from a typical booking form for a hotel chain. We wanted the user to tell us:

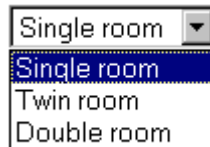
- name of guest
- number of guests
- name of hotel
- type of room (single, double or twin)
- credit card type.

## THE FIVE HTML FORM ELEMENTS

HTML offers five form elements which can be used to allow the user to make a choice. These are all common on the Web, and users expect them to work in a particular way.

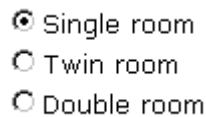
### Form element

1. The drop-down box



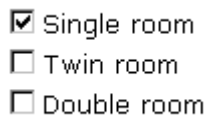
A drop-down menu with a small downward arrow on the right. The text 'Single room' is visible in the box. Below the box, a list of options is shown: 'Single room' (highlighted in blue), 'Twin room', and 'Double room'.

2. Radio buttons



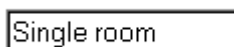
Three radio buttons are arranged vertically. The first one is selected and has a small black dot in the center. The labels are 'Single room', 'Twin room', and 'Double room'.

3. Check boxes



Three check boxes are arranged vertically. The first one is checked and has a small black square in the center. The labels are 'Single room', 'Twin room', and 'Double room'.

4. The type-in box



A rectangular text input box with a thin border. The text 'Single room' is entered inside the box.

5. Hyperlinks

[Single room](#)  
[Twin room](#)  
[Double room](#)

### How users expect it to work

View default choice. If not acceptable,  
- click on down-arrow to show choices  
- scroll through list if necessary  
- click on one option to select it.

One of the options could be blank.

View default choice. If not acceptable,  
click on another option to select it.

Expect to select one option.

Click on all acceptable choices.

Can leave all the boxes blank if none are acceptable.

Type in a response or leave the box blank.

Click on one link to see a new page.

## STEP 1: THE DOMINANT PURPOSE OF THE PAGE

First of all, think about why you are asking the user to make a choice. Is the purpose of the page to allow the user to navigate to another page or part of the site? Or is the purpose of the page to gather information from the user?

### *Navigation pages*

On a navigation page, our objective is to offer an easy route for the user to the next page of interest. If you have a large site with many choices, the lists of options can get excessive. You might think the answer is to offer a drop-down list with a 'go' button. This will save space on screen, and you can offer a variable length list of choices to the user within a predictable, small area within your design.

But we always try to avoid drop-downs on navigation pages because:

- a drop-down box hides the menu options; unless the user is very familiar with the site, s/he will need to review the options in order to make a choice
- hyperlinks only involve one click; a drop-down box gives the user more work

- a well designed list of hyperlinks is easier to scan than a drop-down box
- if there are lots of menu items, it will be impossible for the user to see all the options in the drop-down box without scrolling.

If you find that you are being forced into a drop-down on a navigation page, maybe you should think about the overall structure of your site. Can you 'design out' the need to show the user so many options by dividing the site up differently?

### ***Information gathering***

In most cases, the dilemma over web form elements arises when you need to gather information from the user. You have devised a question to ask the user and have determined that there is a finite set of valid responses. What is the best way to present this set of choices to the user in order to maximise the chance of getting the right answer?

## **STEP 2: SIX QUESTIONS**

When we are choosing elements for an information-gathering form, we find that asking ourselves these six questions helps us to make a good choice:

1. Is it more natural for the user to type the answer rather than select it?
2. Are the answers easily mis-typed?
3. Does the user need to review the options to understand the question?
4. How many options are there?
5. Is the user allowed to select more than one option?
6. Are the options visually distinctive?

We have to put them into an order for this discussion, but we find that they are all worth asking and the actual order is much less important than thinking about all of them.

### ***1. Is it more natural for the user to type the answer rather than select it?***

Personal details such as name, address, and date of birth are so well known to us that it is much easier and more natural to type them in directly rather than selecting from a list. We found that name of person and number of guests are values which are naturally typed rather than selected. The form element of choice is therefore a type-in (data entry) box.

The table on the next page looks at each of our data items from the hotel form example.

<b>Field</b>	<b>Is it more natural for the user to type the answer rather than select it?</b>	<b>Likely choice of form element</b>
Name of guest	yes, more natural to type than select	type-in box
Number of guests	yes, more natural to type than select	type-in box
Name of hotel	no	
Type of room	no	
Credit card type	no	

It may help the user if you can validate the type-in box. For example, if the user types more than 4 into ‘number of guests’, you might offer some help with a group booking – or the option to change the entry to a lower number.

## **2. Are the answers easily mis-typed?**

Although the hotel staff are likely to type the full, official name of a hotel correctly (“Cheviot Country Hotel and Golfing Centre”), a user booking a room on the web might easily make a mistake – by spelling, or by using an abbreviation that differs from the hotel’s own preference (e.g. “Cheviot Golfing Center”). A type-in box is therefore ruled out. But depending upon the design of the form and the web site, any of the other form elements could be used.

<b>Field</b>	<b>Are the answers easily mis-typed?</b>	<b>Choice of form element</b>
Name of guest	no	
Number of guests	no	
Name of hotel	yes, easily mis-typed	hyperlinks
Type of room	yes, easily mis-typed	radio buttons, check boxes
Credit card type	no	

## **3. Does the user need to review the options to understand the question?**

The user may not know what the question is asking until the list of options is available. Questions such as “*Type of room*” could have more than one set of answers to specify preferences from:

- single, double, twin
- two double beds, king-size bed
- accessible by wheel-chair, inaccessible
- smoking/non-smoking
- ground floor, higher floor
- Western-style, Japanese-style.

The most visible method of displaying the options is via radio buttons or check boxes. If you use a drop-down, the user will have work to do before the question makes sense.

<b>Field</b>	<b>Does the user need to review the options to understand the question?</b>	<b>Choice of form element</b>
Name of guest	no	
Number of guests	no	
Name of hotel	no	
Type of room	yes, need to review options	radio buttons, check boxes
Credit card type	no	

#### **4. How many options are there?**

Most form elements become less usable when there is a large number of options (30 or more), because either the page or the drop-down box will require scrolling.

Where there are very few options (4 or less), radio buttons or check boxes work better than a drop-down box. The user can see all options at a glance, it takes one click to make a choice and, with so few options, space on the page is not an issue.

<b>Field</b>	<b>How many options are there?</b>	<b>Choice of form element</b>
Name of guest	Very large number	Type-in box
Number of guests	Could be a large number for a conference or big party	Type-in box
Name of hotel	Not many, unless this is a big hotel chain	Drop-down box, radio buttons
Type of room	Very few (4 or less)	Radio buttons, check boxes
Credit card type	Not many	Drop-down box, radio buttons

#### **5. Is the user allowed to select more than one option?**

Where the user is stating preferences or, for example, selecting brochures, more than one response may be acceptable. Many users know that check boxes allow for multiple selections. If you want to avoid the problem of writing instructions to tell the user about multiple choices, or if you think your users will have problems with them, then you can offer radio buttons instead with an extra button for 'any'.

<b>Field</b>	<b>Is the user allowed to select more than one option?</b>	<b>Likely choice of form element</b>
Name of guest	No	
Number of guests	No	
Name of hotel	No	
Type of room	Yes	radio buttons (with 'any' option) check boxes
Credit card type	No	

### **6. Are the options visually distinctive?**

It is easier to scan a list of options if they are visually distinctive. For example, the lists “single”, “double”, and “twin” (type of room) or “Visa”, “Mastercard” and “American Express” (credit card type) are each short lists of options which are easily distinguished from each other. In contrast, the set of numbers from 01 to 10, because only 2 characters in length, are not so easily scanned when presented in a list. If your options look similar, then consider a type-in box in preference to offering users a list.

If you need to offer a list, then consider varying or abbreviating the options in some way so that it is easier to pick out each item. For example, if a chain of hotels shares similar names then it might be better to offer a list of hotel locations instead. Alternatively, pick up the hotel name from a hyperlink choice earlier.

<b>Field</b>	<b>Are the options visually distinctive?</b>	<b>Likely choice of form element</b>
Name of guest	Options not known in advance	
Number of guests	No	
Name of hotel	Could be similar	drop-down, hyperlink
Type of room	yes, visually distinctive	radio buttons, check boxes, drop-down
Credit card type	yes, visually distinctive	radio buttons, check boxes, drop-down

## **STEP 3: THE IMPACT ON THE FORM**

Once you have chosen which form element to use, we have found that the following considerations will ensure that you implement your choice in the most usable way.

### ***Avoid too many different input methods***

Even after making all the ‘correct’ choices using the questions above, the result may be a form which looks like a demonstration of every different sort of form element. Using the results from above, our hotel reservation form could end up like this:

Field	Choice of form element
Name of guest	Type-in box
Name of hotel	Hyperlink
Type of room	Radio buttons with 'any' option
Number of guests	Type-in box
Credit card type	Drop-down box

Too many different form elements will confuse your users, so look back at your choices to see whether an alternative would offer a more consistent experience.

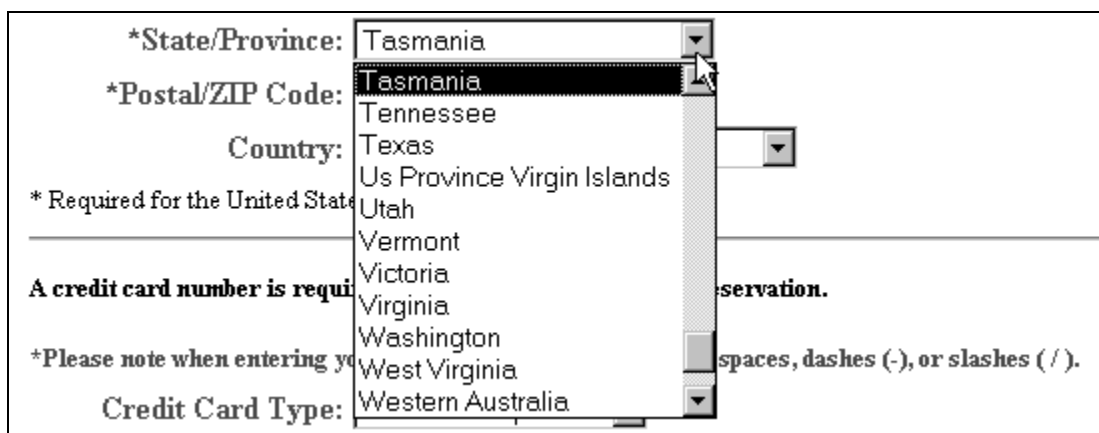
### ***Keep the options short and easily readable***

Wherever possible, and especially when using a drop-down box, make the options short and concise. Again, this will help the user to easily scan the list.

It is possible to present phrases or short sentences as options for radio buttons or check boxes as long as space on the page isn't an issue, and there are not a large number of options. Alternatively, redesign the question to avoid the need to present a list of lengthy options.

### ***Choose a sensible order for the options***

To enable visual scanning of the lists, order the options in a way which will make sense to the user. There may be a natural order to the options, for example, months of the year. In many other cases, sorting the options alphabetically makes it easy for the user to scan the list, although this doesn't always work as the following example shows. Sorting the Australian and USA states together makes it more confusing for Australian and for USA users.



## **STEP 4: USERS AND THEIR INTERACTION WITH THE BROWSER TECHNOLOGY**

The Web is used by a diverse group of people of varying abilities using a variety of different technologies. For instance, users may be using a screenreader due to visual impairment, or they may have physical limitations which affect the type of technology

they use. Not all users will be accessing the Web via a standard desktop PC; they may be using other technologies which use only a keyboard instead of a keyboard and mouse (for example, WAP phones). There are at least two ways in which you can help these users to get a better experience from your form:

- A drop-down box controlled by JavaScript for navigation is impossible to use without a mouse. For greater accessibility, use a 'submit' button instead of JavaScript to initiate the action.
- In describing the page to the user, screenreaders use the 'name' attribute of form elements. Ensure that all your form elements have unique and meaningful names.

## CONCLUSION

Although the choice of form elements on the Web is limited, it is still possible to reach a dilemma over which element to use. We have shown that by following a few simple guidelines, you can almost always be sure of creating a satisfying experience for your user.

- Think about the dominant purpose of your page: navigation or information gathering?
- Ask yourself six questions about the nature of the options
- Check your decisions: choose a sensible order for the options, keep the options short, avoid too many different input methods
- Finally, consider your users and their interaction with the browser technology to ensure that your form is accessible to all.

## RESOURCES

On the web: [www.formsthatwork.com](http://www.formsthatwork.com), the forms advice web site (Caroline Jarrett and Gerry Gaffney)

In print: try Jeff Johnson's book *GUI Bloopers*. Academic Press.

The authors would like to thank Debbie Stone for her help in the preparation of this paper.



**Sarah Miller**

60 Hollow Wood  
Olney  
MK46 5LZ  
United Kingdom

SMiller@ukgateway.net

Sarah Miller is a systems analyst specialising in web development at the Open University. Recently she has decided to broaden her skills to include usability techniques relevant to the web development lifecycle. Before joining the Open University, Sarah worked for Intel Corporation (UK) Ltd. as a systems analyst in web development and Electronic Data Interchange (EDI). Sarah also undertakes consultancy work for Effortmark Ltd.

**Caroline Jarrett**

Usability Consultant  
Effortmark Limited  
16 Heath Road  
Leighton Buzzard  
LU7 3AB  
United Kingdom

Caroline.Jarrett@Effortmark.co.uk

Caroline Jarrett is a usability consultant who specializes in forms: evaluating paper forms, design of paper and on-line forms, and effective implementation of business processes that include forms. She works mostly with large organisations that are trying to establish new usability groups. She is also the practitioner member of the Open University course team developing the User Interface Design and Development course. Caroline is Education and Employment officer for the UK chapter of STC.