

RESERVATION SYSTEM AND METHOD THEREFOR

FIELD OF THE INVENTION

5 This invention relates in general to a system for booking or reserving a particular product or service from a number of different products or services which might be available. More particularly, the invention relates to a reservation or booking system for seats on aeroplanes, in theatres, cinemas and stadiums and the like. The invention also relates to a server and device for displaying a map of different products distributed at different positions
10 within the map which is accessible by the user device.

BACKGROUND OF THE INVENTION

In general, people using a mobile telephone or a tablet computer to book a seat have
15 difficulty using online seatmaps. A seatmap may be thought of as an interactive online representation of the real world and might symbolise the seats in a cinema, theatre, stadium, or seats on an aeroplane or train.

Online seatmaps have existed for many years but they were not designed for mobile
20 devices; they were designed for use with desktop computers, which, in general, have a large display, a keyboard and a mouse. Such a seatmap is shown in figure 1a of the drawings.

As will be appreciated, using a desktop computer equipped with large screen and mouse, it
25 is very easy to select a seat by hovering over circle representing the seat with a mouse and clicking the red circle. The same webpage viewed on a mobile device is very difficult to use, as can be seen in the left hand drawing of figure 1b of the drawings. The user has to zoom in an enormous amount in order to select a seat. It is hard, using a finger rather than a mouse, to select the required seat rather than any of the others surrounding it, and this is
30 illustrated in figure 1b. As can be seen in the right hand drawing of figure 1b, having zoomed in, the row-numbers are no longer visible. This is because the JavaScript is overlaying the header and this further reduces the available screen size.

Some service providers have attempted to overcome this by creating a separate mobile
35 website. In such solutions, the website recognises that a mobile device is being used and

switches the device to an entirely different website. This approach works but it is very expensive for the service provider to maintain multiple websites.

Other service providers offer a mobile application, which customers can download, install and then use to select seats and book tickets. Again, this works but it is expensive to maintain, difficult to support all mobile and tablet devices and it is off putting to customers who, rather than installing an application might visit a competitor's website instead.

SUMMARY OF THE INVENTION

The invention is defined in the appended claims to which reference should now be made. According to a first aspect of the present invention there is provided a reservation system for aeroplane seats comprising communication means for receiving, from a remote device, an availability request for a product or service sending, to the device, a first list element data representing a distribution of a plurality of different products or services within a map in response to receiving the request; and receiving, from the remote device, a reservation request for one of the plurality of different products or services. Preferably, the first list element data is ordered list element data, although this is in fact optional.

Embodiments of the invention seek to address the above problems by providing a single e-commerce server which is configured to support a responsive seat map website which is operable with any device, regardless of whether it is a desktop computer, a tablet or a mobile telephone. The website may operate without switching a user query to a different website which is specifically configured for the user device making the request.

This is particularly important for e-commerce websites, such as websites for booking or reserving seats, where the seat availability changes rapidly, and in real time.

Embodiments of the invention seek to provide an improved seatmap website which is accessible and easily usable, irrespective of the device being used, for example a screen reader or desktop PC. Embodiments of the invention also seek to provide a seatmap website which is operable without the use of JavaScript. Embodiments of the invention seek to provide a seatmap website which minimises the size and number data files which need to be downloaded, such as seat images in GIF, PNG, or JPEG format.

BRIEF DESCRIPTION OF THE DRAWINGS

An embodiment of the invention will now be described, by way of example only, and with reference to the accompanying drawings, in which:

5 Figure 1a shows a prior art seatmap for use with a desktop personal computer;
 Figure 1b shows how the seatmap of figure 1a is rendered on a mobile device;
 Figure 1c shows a schematic representation of seatmap rendered using an html
table;

 Figure 2 shows the main functional components of a server embodying the
10 invention as well as a remote user device embodying the invention which is used for
accessing the system;

 Figure 3 is a flow diagram showing the main steps performed by an embodiment of
the invention;

 Figure 4 shows how a seatmap embodying the invention is rendered on a device
15 such as a screen reader device for use by people with disabilities;

 Figure 5; shows how CSS is used to reposition seat links horizontally;

 Figure 6; shows how the rendered html is displayed on a device where the
information included for screen readers or optional JavaScript popups is hidden;

 Figure 7 shows how the styled html is rendered on a device which shows seat A
20 adjacent to seat B which, in turn is adjacent to seat C;

 Figure 8; shows how the styled html is rendered on a large display;

 Figure 9; shows how the styled html is rendered on a display, and how it can
include a popup window can show more information about the requested seat and how the
cursor changes to a representation of a person being dropped into the requested seat;

25 Figure 10 shows how the styled html is rendered on a display, and how it can
include a cursor to indicate when a seat is not available;

 Figure 11; shows how the styled html is rendered on a smaller display where seats
adjust so that they fit to the available space;

30 Figure 12 shows how the styled html is rendered on a smaller display by reducing
the text when space is limited;

 Figure 13 shows how the styled html is rendered on a very small display by
adjusting the seat size so that corresponds approximately to the font size; and

 Figure 14 shows how the styled html is rendered on three different devices having
different display sizes.

The following description is of a system for use in the aviation industry, but this is exemplary and other applications of the invention will also be discussed. For example, the system may be used in any environment a user wishes to select one service or product from another service or product, and where different products may be represented by different locations in a map. The invention also has particular application to the travel industry in general where a user may wish to book or reserve a particular seat prior to travel. Thus, embodiments of the invention find application in the travel industry in general (for example rail, air, coach and the like), but also in the ticketing industry, such as ticketing for theatre, cinema, and the like.

A seatmap is difficult to implement in html. Creating a representation of all the seats on an aircraft or the seats in a stadium is problematic when html does not easily lend itself to this sort of complex graphical display.

Many seatmaps on the web today have been implemented using html tables. Figure 1c of the drawings shows a standard seatmap included with SITA eCommerce.

Whilst this approach worked in the past, html tables and Responsive Web Design are not compatible. It is not possible to manipulate tables so that they are satisfactorily displayed on all devices. Further, code behind the seatmap shown in figure 1c is inefficient: it requires a lot of calculations and a large amount of html to implement.

It is also impossible for a screen reader, which might be used by a blind or visually impaired person, to make sense of the code behind the seatmap shown in figure 1c. As such, blind and visually impaired people would be unable to select a seat. Below, is part of the html for the seatmap shown in figure 1 c.

```
<tr>
  <td class="rownumber">12</td>
  <td class="wall">&nbsp;</td>
  <td class="airseat occ">
    
  </td>
  <td class="airseat unocc selectbl">
    <a href="#" onclick="SeatClick('S_12B', 'standard_selected_seaticon.gif',
      '/images/seats/standard_seaticon.gif', this);return false"
      onmouseover="SwapSeatImage('S_12B', '/images/seats/standard_seaticon2.gif')"
      onmouseout="SwapSeatImage('S_12B', '/images/seats/standard_seaticon.gif')">
      
    </a>
  </td>
  <td class="airseat occ">
    
  </td>
  <td class="aisle aisle">
```

```

5         
        </td>
        <td class="airseat unocc selectbl">
          <a href="#" onclick="SeatClick('S_12D', 'standard_selected_seaticon.gif',
10          '/images/seats/standard_seaticon.gif', this);return false"
            onmouseover="SwapSeatImage('S_12D', '/images/seats/standard_seaticon2.gif')"
            onmouseout="SwapSeatImage('S_12D',
              '/images/seats/standard_seaticon.gif')">
              
            </a>
        </td>

```

It should be noted that this code only outputs 4 seats. Apart from the fact that there is *a lot* of code, there is very little for a screen reader to make sense of. The JavaScript is jumbled up with the html making it even more difficult to decode.

Referring now to figure 2 of the drawings, the main functional components of a system embodying the invention will be described. The system may include 2 main functional components; a server 201 supporting a web page which is accessible, in principle, by any number of wired or wireless remote devices, referred to as a communication devices 203 comprising a display 211, via wired or wireless communication protocols which will be known to the skilled person. However, it will be appreciated that the invention may be embodied in solely the communication device 203, or solely in the server, or both.

The server may comprise a processor 205 and transmission/ receiving means 207 for transmitting and receiving communications with a mobile device and possible with an inventory control system 209 which controls seat availability. Embodiments of the invention enable a server to support a responsive seatmap where one or more seats can be booked using one website that serves all devices. The seatmap may be configured to operate with or without Javascript on all modern web browsers.

Embodiments of the invention incorporate a responsive design approach so that the website is displayed on the communication device in a manner which facilitates user interaction with the displayed page, for example, by minimising the amount of resizing, panning and scrolling required, irrespective of the device displaying the webpage.

Many content-rich websites have been made responsive so that they work well with any sort of device regardless of whether it is a desktop computer, a tablet or a mobile phone. However, very few ecommerce websites have. Mobile applications which have to be downloaded and installed on a device are also costly and can be off-putting to customers who might not necessarily want to download and install an application, just to check a fare, book a ticket or reserve a seat.

Embodiments of the invention provide a responsive seatmap by using clean, semantic html (hypertext markup language) and css (cascading stylesheets). This code is more efficient and it adapts to the device being used, and will be described in detail below.

5

Below is a slimmed down version of the html used to generate the responsive seatmap according to embodiments of the invention:

```

10  <ol class="seatmap">
    <li class="row" id="r1">
        <ol class="seats">
            <li class="seat available" id="s1A">
                <a title="Select seat A in row 1"
15  href="#s1a">
                    <span class="description">seat A</span>
                    <span class="desc">1A</span>
                    <span class="status">available</span>
                </a>
20  </li>
                Additional list items are generated to output all other seats in this row
                as well as other features such as the corridor, exits, wings, toilets
                and galley.
            </ol>
25  </li>
            <li class="row" id="r2">
                ...
            </li>
30  </ol>

```

The following should be noted about the above html:

- Lists are used to model the seat layout in html
 - The seatmap is an ordered list
 - A list element is created for each *row* of seats
 - The list element has the html class "row" so that all *rows* can be styled
 - Each list element has a unique ID (i.e. "r1") so that each *row* can be styled individually or accessed by JavaScript.
- Inside each *row* another ordered list is output
 - The internal ordered list contains a list element for each *seat*
 - The list element has the html class "seat" so that all *seats* can be styled

- The status is added as an html class (e.g. “available” / “occupied”)
- Each list element has a unique ID (i.e. “s1a”) so that each individual *seat* can be styled or accessed by JavaScript.
- A long description and a short description is output for each *seat*.
The most appropriate is shown with the other being hidden by the CSS.

Figure 4 of the drawings shows how the above html is rendered in a web browser, or is rendered by a screen reader, which might be used by a visually impaired user. This demonstrates that the seat display still works for people with disabilities who might be using a screen reader and are still able to select an available seat. It also demonstrates that seat selection is still available without JavaScript: users simply select one of the links to choose a seat.

However, embodiments of the invention may incorporate CSS, Cascading Stylesheet rules, so that the seatmap is responsive to the type of device requesting information on available seats, and displaying the seatmap.

A number of CSS rules make this possible.

The stylesheet takes a mobile-first approach. This means that the display is styled for mobile devices. The rules for wider displays (bigger screens) are enclosed in *media queries*. With this approach, many different blocks of CSS code, relating to various different browser window widths, is sent to each web browser. The web browser then applies the appropriate CSS given the current width of the browser window.

Media Queries

```
@media only screen and (min-width: 880px) {  
    ...  
}
```

Media Queries are used to implement specific CSS rules when the browser window is a certain width. 880px is used as an example above but media queries can be created for as many different widths as required. The CSS may use a Media Query to determine the width of the user's web browser. The rules may Media Queries

Media Queries are used by Cascading Style Sheets (CSS) to determine the width of the user's web browser window. The Style Sheet may contains one or more rules. For

example, the a rule may determine the browser window size. If the browser window is determined to be greater than 900px but less than 1200px then user interface is formatted in a first manner. In a further example, if the browser window size is determined to be greater than 1200px but less than 2000px then the user interface is formatted in an alternative manner. The formatting style may be rendered as shown in accordance with figures 8 to 14 of the drawings, and described in further detail below.

All the rules are sent to all web browsers - provided that the web browser understands CSS and has CSS enabled or switched on. When CSS disabled or switched off, the web browser does not request the CSS files.

With media queries, all the CSS is sent to the web browser. The browser then uses its own knowledge about the current width of the window, to decide which CSS to implement.

Seat styling

To make a seatmap, we are faced with the problem of making of standard html mimic the layout of seats in an auditorium or aeroplane, for example.

This is implemented as follows.

Firstly, the html is created: as mentioned previously the html for a single seat is shown below.

```
<li class="seat available" id="s1A">
  <a title="Select seat A in row 1" href="#s1a">
    <span class="description">seat A</span>
    <span class="desc">1A</span>
    <span class="status">available</span>
  </a>
</li>
```

Embodiments of the invention use a list (LI) so that the displayed text makes sense to people who have the CSS disabled (such as those using a screen reader).

Inside the list, we use an anchor tag (A) so that users can click the seat they want to sit in.

This works for all users, all devices and does not require JavaScript.

All the text is enclosed in span tags so that it can be styled later.

This user sees the following in his web browser: (two additional seats are shown)

- seat A 1A available
- seat B 1B available
- seat C 1C available

5 Clearly this looks nothing like a seat and so some styling is required.

Seat row

In the real world, the seats are side by side and so we need to represent that on the page. We do this by floating each seat *left*. To make the seat responsive to all devices and screen widths, we set the seat width as a percentage rather than an absolute value. Rows of seats now use the full width of the display.

```

15      LI{
          float:left;
          white-space:nowrap;
          text-align:center;
        }
        LI.seat{
20          margin-right:0.4%;
          width:8.5%;
        }

```

This code would be rendered as shown in Figure 5 of the drawings. Figure 5 shows how CSS is used to reposition seat links horizontally so that seats that would be side-by-side in the real world are side-by-side on the display. Figure 5 also shows text included for people with disabilities.

Now we can hide the information included only for screen readers or optional JavaScript popups. We can also hide the bullet points / numbered list:

```

30      .status{
          display:none;
        }
        UL, OL{
35          list-style-type: none;
        }

```

This code would be rendered as shown in Figure 6 of the drawings.

40 Seats that look like seats

The seat links are now side-by-side but they still look very much like standard html links. The final step is to style the links and make them look more 'seat' like.

This is achieved by styling the anchor tag: removing the 'underline', adding a border, rounding the edges at the top and changing the background colour. The rounded corners (border-radius) mimic the rounded edge on a seat / cushion.

```
5      A {  
        text-decoration: none;  
        border: 1px solid #666;  
        border-top-left-radius: 10px;  
10       border-top-right-radius: 10px;  
        padding: 10%;  
        margin-right: 0.1%;  
        margin-bottom: 0.1%;  
        color: #222;  
        background-color: #aaa;  
15       white-space: nowrap;  
        display: block;  
      }
```

20

This code is rendered as shown in Figure 7 of the drawings.

Using the above rules, the code is rendered on a screen reader as shown in figure 4 of the drawings, and is rendered on a large display as shown in figure 8 of the drawings. On such
25 a large, wide display the seatmap uses as much space as it needs and a richer interface is available.

As shown in figure 9 of the drawings, a popup window can show more information about the requested seat and it is possible to change the cursor so that it looks like a person is
30 being dropped into the requested seat. As shown in figure 10 of the drawings, another cursor can be used to indicate when seats are not available.

Further, as shown in figure 11 of the drawings, on smaller displays, the seats adjust so that they always fit.

35

Figure 12 shows how the text is reduced when space is limited, while figure 14 shows how the seatmap is displayed on the smallest display.

Accordingly, embodiments of the invention provide a seatmap that:

- 40
- works with all devices adapting to all screen widths
 - is suitable for users with disabilities (using a screen reader)
 - requires the smallest possible download (for users with slow connections)

Without images or JavaScript, this seatmap, containing 80 seats, requires a download of just 26kb. It is anticipated that some JavaScript would be required but this would not add significantly to the total. Each additional row of 10 seats uses just 3kb of extra data.

5 Other seatmaps supply seats as images in GIF, PNG or JPEG format. Whilst this can create a more realistic seatmap, it adds to the number of downloads and the amount of data required.

10 An alternative to this design is to use SVG stacks. SVG (Scalable Vector Graphics) is a vector image format and can be used to create image files that are very small on disc. The images need to be reasonably simple and so a simple seat icon could be easily created and would use only a small number of kilobytes. Importantly, SVG files – being vector graphics – can be output at any size on the web page which is vital for a responsive seatmap that must work at all screen sizes. SVG stacks allow us to ‘stack’ several images in one file, which makes them quicker to download.

15

A further advantage of a responsive seatmap is cost. Providing just one website serving all devices is vastly cheaper than providing a separate site for mobile devices.

CLAIMS

1. A reservation system comprising:

communication means for:

receiving, from a remote device, an availability request for a product or service;

sending, to the device, at least a first ordered list element data representing a distribution of a plurality of different products or services within a map in response to receiving the request; and

receiving, from the remote device, a reservation request for one of the plurality of different products or services;

wherein the communication means is further arranged to receive a styling request associated with the display of the products or services on a display and in particular wherein the communication means is arranged to send the styling data only if a styling request has been received;

wherein the communication means is further configured to send style sheet data for each ordered list element for rendering on a display, wherein the style sheet data comprises a plurality of different style data associated with different viewing area pixel sizes, and in particular wherein the style sheet data floats each product or service so that each row of the distribution of the plurality of different products or services within the map occupies a full display width and wherein the style data sets the display width of each product or service as a percentage value; and

wherein the communication means is further configured to send descriptor data associated with the product or service, wherein each product or service is associated with a plurality of descriptors for selection based on a determined viewing area pixel size.

2. The reservation system of claim 1 wherein the communication means is further arranged to send a unique identifier associated with the ordered list element data.
3. The system of claim 1 or 2 further comprising a processing means configured to associate each product or service with status element information indicative of the availability of each product or service, and in particular wherein the communication means is arranged to send the status element information to the remote device and preferably wherein each status element is associated with the ordered list element.

4. The system of any preceding claim wherein the processing means is configured to represent a distribution of seats within a seat map by associating the first ordered list element with a plurality of different seats distributed within the map and wherein the communication means is further arranged to send styling data associated with each ordered list and to send a media query data for determining a viewing area pixel size.
5. The system of any preceding claim wherein the processing means is further configured to associate the first ordered list element data with a first row of products or services and in particular in which the processing means is further configured to associate a plurality of ordered list elements with a plurality of rows of products or services.
6. The system of any preceding claim wherein the processing means is further configured to associate each list element with a unique identifier and preferably wherein each product or service is associated with at least one descriptor of the product or service.
7. The system of any preceding claim wherein each product or service is associated with a plurality of descriptors.
8. The system of any preceding claim wherein the communication means is further configured to send style sheet data for each ordered list element for rendering on a display and in particular wherein the style sheet data hides bullet points or hides a numbered list.
9. The system of any preceding claim wherein the communication means is further configured to send style sheet data for each ordered list element for rendering on a display and in particular wherein the style sheet data styles an anchor tag by removing an underline, including a border, rounding the edges and changing a background colour.
10. The system of any preceding claim wherein the communication means is further configured to send style sheet data for each ordered list element for rendering the product or service as a representation of a seat within a seatmap.
11. A portable reservation system comprising:

communication means for sending an availability request for a product or service and for receiving from a remote reservation system, a first ordered list element data associated with a plurality of different products or services and for sending a unique identifier associated with the list element data in response to receiving the request;

wherein the communication means is arranged to send, to the remote reservation system, a reservation request for one of the plurality of different products or services and a styling request associated with the display of the products or services on a display;

wherein the communication means is arranged to receive style sheet data for each ordered list element for rendering on a display wherein the style sheet data comprises a plurality of different style data associated with different viewing area pixel sizes and wherein the style sheet data floats each product or service so that each row of the distribution of the plurality of different products or services within the map occupies a full display width and wherein the style data sets the display width of each product or service as a percentage value; and

wherein the communication means is further arranged to receive descriptor data associated with the product or service, wherein each product or service is associated with a plurality of descriptors and wherein one of the descriptors is selected based on a determined viewing area pixel size.

12. The portable reservation system of claim 11 further comprising determining a current viewing area pixel size and selecting a style based on the determined viewing area pixel size.

13. A reservation method comprising:

receiving, from a remote device, an availability request for a product or service;

sending, to the device, a first ordered list element data representing a distribution of a plurality of different products or services within a map in response to receiving the request;

receiving, from the remote device, a reservation request for one of the plurality of different products or services;

receiving a styling request associated with the display of the products or services on a display and in particular further comprising sending the styling data only if a styling request has been received;

sending style sheet data for each ordered list element for rendering on a display, wherein the style sheet data comprises a plurality of different style data associated with different viewing area pixel sizes, and in particular wherein the style sheet data floats each product or service so that each row of the distribution of the plurality of different products or services within the map occupies a full display width and wherein the style data sets the display width of each product or service as a percentage value; and

sending descriptor data associated with the product or service, wherein each product or service is associated with a plurality of descriptors for selection based on a determined viewing area pixel size.

14. The reservation method of claim 13 further comprising sending a unique identifier associated with the ordered list element data.
15. The method of claim 13 or 14 further comprising associating each product or service with a status element information indicative of the availability of each product or service and in particular further comprising sending the status element information to the remote device and preferably wherein each status element is associated with a list element.
16. The method of any of claims 13 to 15 further comprising representing a distribution of seats within a seat map by associating the first ordered list element with a plurality of different seats distributed within the map and further comprising sending styling data associated with each ordered list and to send a media query data for determining a viewing area pixel size.
17. The method of any of claims 13 to 16 further comprising associating the first ordered list element data with a first row of products or services and in particular further comprising associate a plurality of ordered list elements with a plurality of rows of products or services.
18. The method of any of claims 13 to 17 further comprising associating each list element with a unique identifier and preferably wherein each product or service is associated with at least one descriptor of the product or service.
19. The method of any of claims 13 to 18 wherein each product or service is associated with a plurality of descriptors.

20. The method of any of claims 13 to 19 further comprising sending style sheet data for each ordered list element for rendering on a display and in particular wherein the style sheet data hides bullet points or hides a numbered list.
21. The method of any of claims 13 to 20 further comprising sending style sheet data for each ordered list element for rendering on a display and in particular wherein the style sheet data styles an anchor tag by removing an underline, including a border, rounding the edges and changing a background colour.
22. The method of any of claims 13 to 21 further comprising sending style sheet data for each ordered list element for rendering the product or service as a representation of a seat within a seatmap.
23. A portable reservation method comprising:
 - sending, using a communication means, an availability request for a product or service and for receiving from a remote reservation system, a first ordered list element data associated with a plurality of different products or services and for sending a unique identifier associated with the list element data in response to receiving the request;
 - sending, to the remote reservation system, a reservation request for one of the plurality of different products or services and a styling request associated with the display of the products or services on a display;
 - receiving, at the communication means, style sheet data for each ordered list element for rendering on a display wherein the style sheet data comprises a plurality of different style data associated with a plurality of different viewing area pixel sizes and wherein the style sheet data causes each product or service to float so that each row of the distribution of the plurality of different products or services within the map occupies a full display width and wherein the style data sets the display width of each product or service as a percentage value; and
 - receiving, at the communication means, descriptor data associated with the product or service, wherein each product or service is associated with a plurality of descriptors and wherein one of the descriptors is selected based on a determined viewing area pixel size.

24. The method of claim 23 further comprising further comprising determining a current viewing area pixel size of the plurality of different viewing area pixel sizes and selecting a style based on the determined viewing area pixel size.
25. A computer readable medium which when executed undertakes the method of any of method claims 13 to 24.