

Dog Rental Service Web Application Report

Dog Rental Service project link: <https://altitude.otago.ac.nz/mtrn/dog-rental-service>

Dog Rental Service required a web application for to manage their booking service to their customer. For the first assignment, we were required to create an interface for the web application. The application consists of 4 pages: index page, admin page, booking page and the point of interest page. The application interface is created using html5, css3 and javascript in phpStorm.

The main layout of the web pages includes a header consists of the login form and a side navigation bar leading to the Home (index), View Dog (booking) and Point of Interest pages. The login form and sidebar are fixed on scrolling with the sidebar adjusting its own size based on the width of the window.

The index page is the main interface that visitor will see when they first access the web application. It consists of a welcoming message and the reviews about the page. The reviews are stored beforehand in a separate json file and is displayed in html elements that are dynamically created in the javascript. We create html elements dynamically to support the increasing size of the reviews. The current index page loads all reviews in listing order, which can get very long, therefore, further enhancement for this page can be a carousel or animation to display the reviews in less space and more appealing.

In the booking page, the user can have a preview of the dogs and choose the available dog options. For the dog preview, all the dogs are listed together with their information, and user cannot add them to booking until they have selected date, time and number of hours for booking. The user can selection of booking detail will apply a filter to see the available dogs and allow them to add the dog to their current bookings, and they will have to make a new booking if they want to change the pick-up details. As per the requirements, up to 3 dogs are allowed per booking. In addition, we only allow a dog to be added once only. After selecting the dogs and adding user booking name, the user can save current booking to local storage. Upon moving to another page, users are prompted to confirm that the current booking will be removed if they have not saved it.

The requirements also outline the interactive map that shows the points of interest to visitor, including the company office, the surrounding parks and the walking tracks. In the Points of Interest page, the user can move along the map to check the loaded markers of the points of interest. There is also a list of points of interest underneath with a checkbox to show or hide the marker and a button to focus the map view on that marker. The markers also include small pop-up showing the name and the type of the marker. If the visitor hides the marker, the focusing button will be disabled as well. All the markers are currently only listed with only name and type in the pop-up and in the list underneath, further development can include more details on the location, such as the address or the opening hours.

We are required to have an administrative view at the current booking for staff, which are displayed in the admin page. In order to access the admin page, the user have to select "Login as Admin" in the header. Currently, the admin page only displays existing booking in table form, as per the requirements, which means that upon required, the admin page can be used to display more information in further development.

Third-party code and libraries:

In the development of Dog Rental Service web application, we use third-party libraries which are Leaflet, jQuery and jQuery UI. The leaflet library is required to create the interactive map and the jQuery is used in association with script files.

JsHint:

The final code is checked though jsHint as Code Quality Check, with their setting stored in an xml file. One of the notable settings is the warning about incompatibilities with specified ECMA Script, which was chosen to prevent usage of functions that may not render properly in other environments. In several files, we also ignore the unused local symbol warning to allow the reusability of the code as an event-handler, even when the parameter e (for event) is not used inside the code function.

Test Case:

General:

- Testing the website in various browsers: Mozilla Firefox, Google Chrome

For administration page:

- Access into administrative view by clicking the “Login as Admin” from any other web page.

For View of Dog Options:

- Disable the pickup select form once submitted, this will be re-enabled after page reload or creating a new booking
- Only generate “Add to Booking” button when the time and date for pickup has been selected.
- Date selection is within a range of 01/01/2021 to 31/12/2022, time is within the range of 9:00 to 16:00. The range outside are automatically disabled based on html attributes.
- Alert when a dog is selected twice
- Alert when more than 3 dogs are selected for one booking
- Allow adding more dog again after having removed a dog from one booking.
- Name for the booking must be not null and greater than 2 in length.
- Savings are stored in localStorage.

For Point of Interest:

- Show/hide checkbox show/hide the marker and enable/disable the focusing button
- Focusing button focuses on a marker

Existing issues and problems:

The existing issues includes the flow and the aesthetic of the View Dog page. When there is no pickup time selected, the information is displayed as null, which the visitors do not need to know. In addition, since we are using the attributes in the html to validate the time, and we are using input [type= “time”], the error message sometimes does not print out. In addition, due to the nature of the input [type= “time”], there is no consistency in time display: 24 hours against am/pm.

In addition, current bookings will be overwritten in the localStorage once a new booking is saved.