

PEARL OF THE INDIAN OCEAN | DOCUMENTATION

06/01/2022



PRESENTED BY: M. D. IPALAWATTA

PROBLEMS AND SOLUTIONS

All the problem occurred while implementing the website and solutions for them

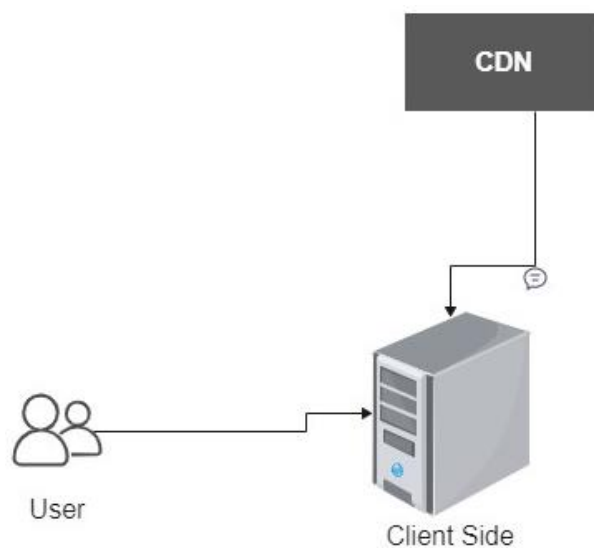
Problems	Solution
Having troubles when positioning elements in different positions	Position elements as grid
Required high resolution images without get more time to load from viewer perspective (Network speed)	Make all the images available locally(downloaded)
Not comfortable when loading images and vue application as a viewer	Add a loader animation

CONSTRAINTS

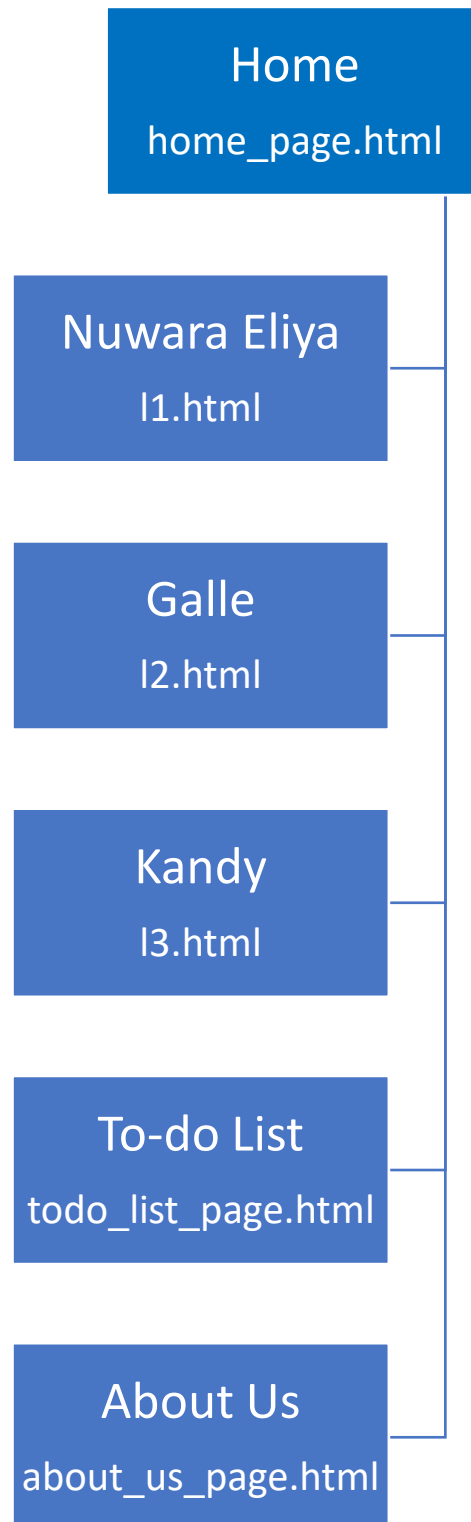
All the constraints for the website

- Limiting Event field input value for 33 characters
- Assume that viewer browsing the website from 1440x900 window resolution
- After refresh the to-do list page, all events will disappear

ARCHITECHTURAL DIAGRAM OF THE SYSTEM



SITEMAP



USED WEB TECHNOLOGIES AND EXAMPLES

CLIENT-SIDE SCRIPTING /CODING

- HTML
- CSS
- SASS
- JavaScript

WEB APPLICATION FRAMEWORKS

- Vue.js (CDN)

TEST RESULTS

LOADER ANIMATION .JS FILE TEST

Test whether query selector selecting loader class correctly. If it works correctly, browser should log the console the right element.

```
js > JS loader.js > vanish
1 //when window is fully loaded, loader will be vanished
2
3 var loader = document.querySelector(".loader");
4 console.log(loader)
5
6 window.addEventListener("load", vanish);
7
8 function vanish() {
9   loader.classList.add("disappear");
10 }
```



ADDEVENT FUNCTION TEST

If addEvent function works correctly, browser should log the console with outputs array and the right object in it.

```

52 //add events to the liast
53 addEvent () {
54   if ((this.inputs[0].value == '' && this.inputs[1].value == '') || this.inputs[0].value == '') {
55     this.showError = true
56   } else{
57     this.outputs.push({event: this.inputs[0].value, date: this.inputs[1].value})
58     console.log(this.outputs)
59     this.inputs[0].value = ''
60     this.inputs[1].value = ''
61     this.showError = false
62   }
63 }

```

The screenshot displays a web application with a dark theme. At the top, there's a section titled 'Event (0/33)' with two input fields: 'Event' and 'Date' (formatted as mm/dd/yyyy). A green '+' button and a red 'Clear All' button are next to the inputs. Below the inputs, a list of events is shown, with the first one being 'go to kandy' dated '2022-01-06'. A checkbox labeled 'Not Completed' is next to it. At the bottom, there's a 'Contact Us' section with social media icons for Instagram, Twitter, Facebook, and YouTube. The footer text reads '© 2022 Pearl of the Indian Ocean'.

The browser console on the right shows several Vue.js warnings: 'Extraneous non-props attributes (id) were passed vue@next:8035 to component but could not be automatically inherited because component renders fragment or text root nodes.' These warnings are repeated for the 'Mainwindow id="main-window"' and 'App' components. A log entry shows the event object: `{event: 'go to kandy', date: '2022-01-06'}`. The console also shows a 'What's New' section with updates from Chrome 96.

CLEARALL FUNCTION TEST

If clearAll function works correctly, browser should log the console with empty outputs array.

```

64 //clear all events in the list
65 clearAll () {
66     length = this.outputs.length
67     this.outputs.splice(0, length)
68     this.showError = false
69     console.log(this.outputs)
70 }

```

The screenshot displays a web application titled "To-Do List" with a background image of a bridge at night. The form includes a header "You can easily add your future tourist events you hope to do and track them here.", an "Event (0/33)" input field, a "Date" input field with a calendar icon, and a "Clear All" button. Below the form is a "Contact Us" section with social media links for Instagram, Twitter, Facebook, and YouTube, and a footer "© 2022 Pearl of the Indian Ocean".

The Chrome DevTools console on the right shows several Vue.js warnings: "Extraneous non-props attributes (id) were passed to component but could not be automatically inherited because component renders fragment or text root nodes." These warnings are repeated for the 'MainView' component. Below the warnings, the console shows the state of the 'Proxy' object, which is an empty array (length: 0). At the bottom, there are "Highlights from the Chrome 96 update" including a new CSS Overview panel and options to enable emulating prefers-contrast media and auto dark mode.

DELETEEVENT FUNCTION TEST

If deleteEvent function works correctly, browser should log the console with outputs array and without the deleted object in it.

```

121     methods: {
122       //delete selected event
123       deleteEvent () {
124         index = this.$parent.outputs.findIndex(x => x.event ===this.eventName && x.date ===this.date);
125         this.$parent.outputs.splice(index, 1)
126         console.log(this.$parent.outputs)
127       },

```

The screenshot displays a web application interface and its corresponding browser console. The web application, titled "To-Do List", features a header image of a bridge over water. Below the header, there is a text input field for "Event (0/33)", a date input field for "Date" with the placeholder "mm/dd/yyyy", a green "+" button, and a red "Clear All" button. The footer includes social media links for Instagram, Twitter, Facebook, and YouTube, along with a copyright notice "© 2022 Pearl of the Indian Ocean".

The browser console shows several Vue.js warnings: "Extraneous non-props attributes (id) were passed to component but could not be automatically inherited because component renders fragment or text root nodes." These warnings are repeated multiple times. Additionally, the console shows a log output from the deleteEvent function, which is an empty array: `Array(0)`.

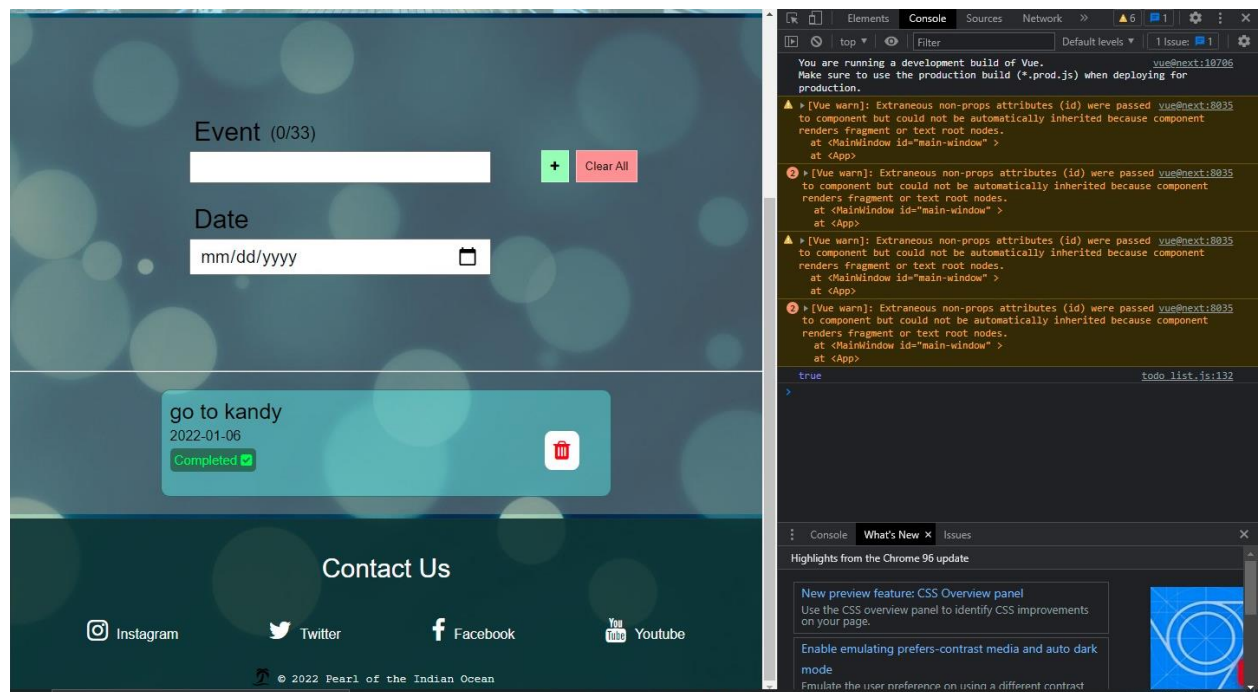
DONEEVENT FUNCTION TEST

If doneEvent function works correctly, browser should log the console with the Boolean value of isDoneEvent property.

```

128 //mark as completed selected event
129 doneEvent () {
130   isDoneEvent = this.$parent.isDone
131   this.isDoneEvent = !this.isDoneEvent
132   console.log(this.isDoneEvent)
133 }
134

```



CONCLUSION

Pearl of the Indian Ocean Website is the result of EEI3346-Mini Project. It took one month and a half for the completion of this project. This website contains 5 separate pages and designed using HTML, CSS, SASS, JavaScript, Vue.js(CDN) and Font Awesome Icons(CDN). It contains all the images as downloaded. In the test phase, 5 tests were done for very important segments.

~END OF THE DOCUMENTATION~