

Software Development **Proposal**

```
index.js
author (You)
File 2 CSS + 1 JavaScript

// handles because the labels are providing a larger hit area for the check to be made.
2 + document.querySelectorAll(".style2").checkSet label;
document.querySelectorAll(".arrow");

// handles adding the arrowMotion class so if the arrow gets animated and then removes it if the box
is to remove the dashboard icon if it is unchecked
checkboxes) {
  < this.parentElement.querySelector("input").checked;
  & querySelector(".arrow").classList.add("arrowMotion");
  & querySelector(".arrow").classList.remove("arrowMotion");
  classList.remove("dashboardIcon");
  classList.add("circleIcon");

// handles switching the dashboard icon after the checkbox is checked
300ms. This is a work-around
as has to be added as we are not able to modify parent elements directly from javascript as they do not
work around {
  check = this.parentElement.querySelector("input").checked;
  check ? this.parentElement.classList.add("circleIcon");
  else this.parentElement.classList.remove("circleIcon");
  this.parentElement.classList.add("arrowMotion");

// for a click event so as to add the animation to the arrow class
2.checkbox2 => checkbox2.addEventListener("click", handleArrowAnimation);
// for the animation of the arrow to end and then the box will be checked (the duration will appear). It
will be checked instantly when the item is clicked but the duration will wait for the animation to end
appear. />
checkbox2.addEventListener("animationend", handleCheck);

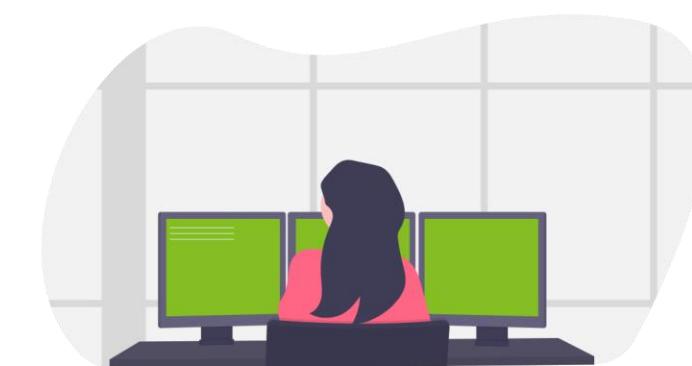
style.css
customRadioButtonsCheckboxes & index.html & body
  @media screen and (-ms-high-contrast: none), (-ms-high-contrast: active) {
    html, body {
      margin: 0;
      padding: 0;
      font-size: 10px;
      font-family: sans-serif;
      color: #333;
      background-color: #f9f9f9;
    }
  }

  /* General */
  * {
    margin: 0;
    padding: 0;
    box-sizing: border-box;
  }

  /* Header */
  header {
    padding: 10px;
    display: flex;
    justify-content: center;
    align-items: center;
    background-color: var(--teal);
  }

  header a {
    color: var(--black);
  }

  .header {
    position: absolute;
    top: 0;
    left: 0;
    width: 100px;
    height: 100px;
    background-color: var(--teal);
    display: flex;
    align-items: center;
    justify-content: center;
    font-size: 10px;
    font-weight: bold;
    color: white;
  }
```



Software Development **Proposal**

Deloitte.

1. Overview

Deloitte proposes to design and implement a private dashboard solution for Daikibo, aimed at monitoring the health status of nine machines across each of Daikibo's four factories. This dashboard will aggregate telemetry data and provide users with real-time insights into the status of each machine, while enabling access through Daikibo's intranet, secured by internal authentication.

Please find enclosed this document our Software Development Proposal for Daikibo's Real-time Telemetry Dashboard.

2. Scope

Here are the main functionalities of the project:

- Private dashboard showing the health status of the 9 machines in each of Daikibo's 4 factories for which they collect telemetry.
- Access to the page happens only within the client's intranet.
- Authentication is synced to an internal authentication server (users can leverage their company-wide accounts).
- The dashboard consists of a single page listing the current status of all monitored devices.
- The view is collapsible/expandable at the factory and device levels (showing a history of statuses).

You can refer to the wireframe image located on the next page for visual reference. Please note this is not the final design, it is a mock-up visual representation of the functionality.

Software Development **Proposal**

✓  Daikibo Factory Meijo	Last update: <1min ago	◀
✓  Daikibo Factory Seiko	Last update: <1min ago	◀
✓  Daikibo Berlin	Last update: <1min ago	◀
✗  Daikibo Shenzhen	Last update: <1min ago	▽
✗  CNC	Last update: 2min ago	▽
✗  Status: Unhealthy	2min ago	
✓  Status: Healthy	12min ago	
Load More		
✓  LaserCutter	Last update: <1min ago	◀
✓  HeavyDutyDrill	Last update: <1min ago	◀
✓  SpotWelder	Last update: <1min ago	◀
✓  LaserWelder	Last update: <1min ago	◀
✓  MetalPress	Last update: <1min ago	◀
✓  Furnace	Last update: <1min ago	◀
✓  ConveyorBelt	Last update: <1min ago	◀
✓  AirWrench	Last update: <1min ago	◀

3. Estimate

The total number of man-hours needed for this project is 120 hours.

Design	Development	Testing	Integration	Total
20	70	20	40	120

We are going to form an internal team of 3 software engineers and 1 graphic designer.

We will require the help of at least 1 IT engineer from Daikibo to hand off the finished product and help us with access to authentication and access to telemetry databases/servers.

4. Timeline

1. [1st of September 2025] **Design starts**
2. [4th of September 2025] **Design is discussed with Daikibo for feedback**
3. [6th of September 2025] **Design is finalized, and Development starts**
4. [19th of September 2025] **Development is done, and the product is demonstrated to Daikibo**
5. [21st of September 2025] **Development is finalized, and Testing starts**
6. [25th of September 2025] **Testing is done, and Integration starts**
7. [2nd of October 2025] **Integration is completed**

5. Support

This proposal's focus is the development of the project. After the product is successfully deployed with Daikibo's infrastructure, our team will remain available for continuous support.

Further support can be submitted through our internal support system. The estimate of work described earlier does not included in the continuous support we provide. Any future invoices, bug fixes, updates, and improvements will be invoiced separately.