Getting started:

* To start the application, first you will need to use the node package manager to download the following dependencies in the react project (Copy and paste into the terminal):
  + npm install axios
  + npm install cors
  + npm install date-fns
  + npm install fs
  + npm install http
  + npm install prop-types
  + npm install react-big-calendar
  + npm install react-datepicker
  + npm install react-helmet
  + npm install react-router-dom
  + npm install @mui/material @mui/styled-engine-sc styled-components
  + npm install @mui/x-charts
  + npm install @mui/material @emotion/react @emotion/styled
    - Also don’t forget to download the express.js for the server-side files (the backend servers in the server-side folder)
* If you have any problem with any module, React will notify you of the error, and you could just simply do the following: write in the terminal npm install “module name” and it will be downloaded.
* After downloading the necessary dependencies, we have a few more steps to follow:
  + First, we open the react server through opening a CMD terminal in the Dern-Support file directory and with the CMD terminal we will run the following command: npm start.
  + After opening the react server, you will have to open the backend servers.
    - First navigate to the server-side folder in the dern-support folder. Then when inside the folder, open 3 CMD windows with the directory of the server-side folder and in these windows.
    - Run the following Commands:
      * node server.js
      * node serverInventory.js
      * node serverSubmitRequest.js
* Each of these commands should be in a different window of the cmd windows.
* And that’s how to run the Full stack solution, moving on to the testing.

Testing Methods:

* Test Plan: The plan is to use the White Box testing method.
* Unit Test Cases:

**[Test No.1]**

**[Description]**

Testing the Login page

**[Result]**

The login page works as intended and it redirects users to the pages based on their roles.

**[Test No.2]**

**[Description]**

Testing the Sign Up page

**[Result]**

The Sign Up page works and it stores the new users data into the Database.

**[Test No.3]**

**[Description]**

Testing the support request page

**[Result]**

It works as intended, and it stores data into its designated database while also displaying it into the table below.

**[Test No.4]**

**[Description]**

Testing the Spare parts inventory page

**[Result]**

It mostly works as intended, and the update button works, but the delete button isn’t functioning.

**[Test No.5]**

**[Description]**

Testing the Job Schedule Page

**[Result]**

It works but right now its mostly just a façade with no real functionality

**[Test No.5]**

**[Description]**

Testing the Analysis Page

**[Result]**

The page works as intended and the Analysis Line Chart is working perfectly and retrieving data from the database.

Review and Refinement:

* Based on the review of few of my colleagues after the last adjustments, they are satisfied with the new layout and design.
* “Most other reviews are about minor bugs that will be mentioned in the bug testing” and they have been now mentioned above.

Evaluation of the Solution:

* Efficiency: The application is efficient.
* Effectiveness: The application meets most of the requirements and the rest will be done in the future plans (The rest are minor stuff like the search function in the spare parts page)
* Compliance: It follows the industry standards
* Usability: The application is perfectly usable and easy to understand
* Robustness: The application is robust enough to handle errors most of the time but it’s not 100% guaranteed.
* Maintainability: the application is perfectly maintainable and is very modular.

Future Development Suggestions:

* Better Security and a Log out function: it’s of best interest to create a secure system.
* A search function for the spare parts inventory page: it would be convenient to add a search function for when searching for specific PC parts.
* Accessibility options like Text-to-speech, High Contrast Mode and Color Filters.
* A search function for the knowledge base to access blogs easier.
* Functionality for the contact us page.
* Functionality for the knowledge base page.