

Garage Management System

Student name: Adam Dowling, Muhammad Fahad, Gleb Tutubalin, Taemour Bharat, Emoshoke Saliu

Student number: C00298483, C00311349, C00290944, C00295140, C00297032

Project Supervisor's Name: Aine Byrne

Programme: CW_KCSOF_B.

Year: 2

Module: Project SEM 2

Submission Date: 27/03/25

Title of Assignment: Garage Management System

Declaration of Ownership: We declare that the attached work is entirely our own and that all sources have been acknowledged:

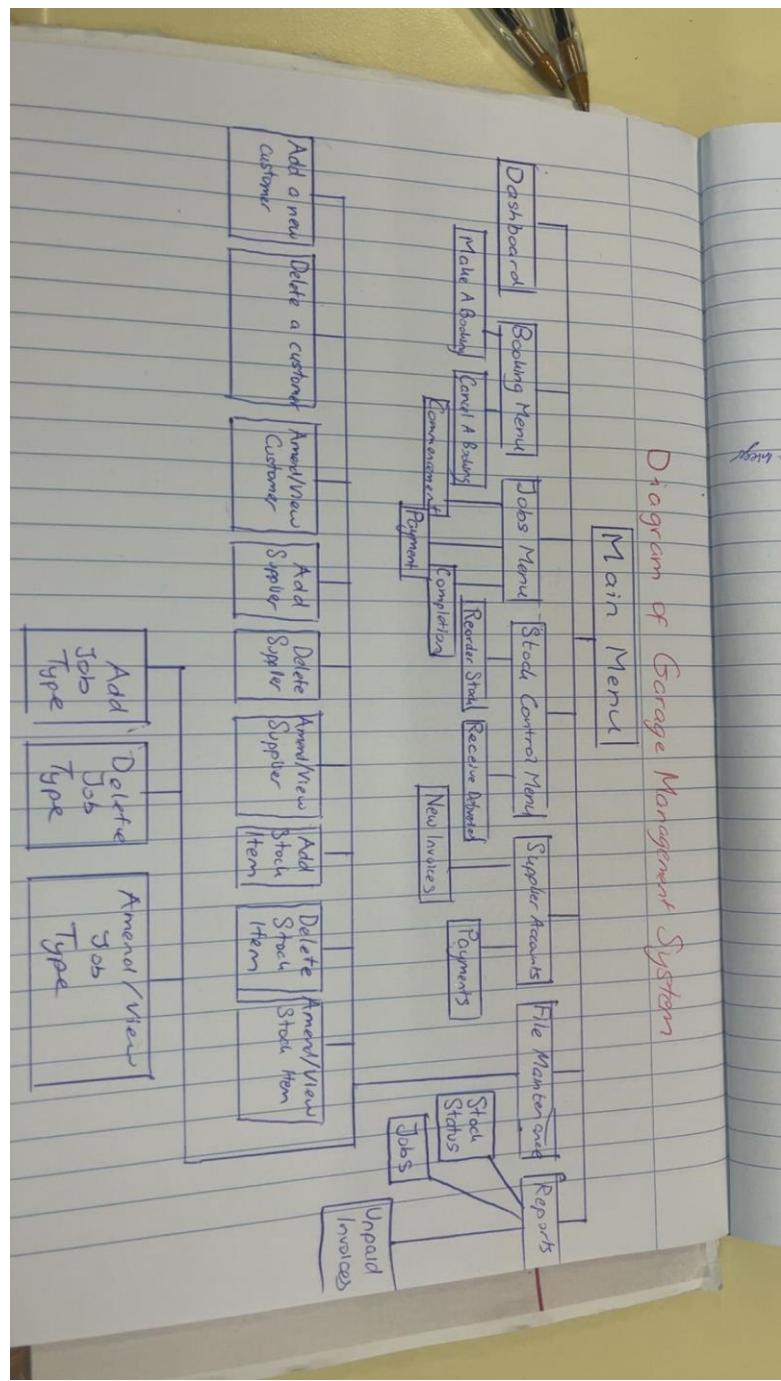
Garage Management Diagram.....	3
Garage-Management-System (Main Folder)	4
User Interface	5
Menu.....	6
Add Screens.....	7
Delete Screens	9
Amend / View Screens	10
Common files for all screens.....	11
Original ER Diagram	29
Database Tables	30
Booking Table.....	30
Customer Table	31
Invoice Table	32
Job Table	33
Job Parts Table	34
Job Type Table	35
Mechanics Table	36
Order Table	37
Payment Table	38
Stock Item Table (Adam Dowling).....	39
Stock Item / Order Table (Adam Dowling).....	40
Supplier Table (Muhammad Fahad)	41
GROUP WORK	42
Admin DashBoard	42
Dashboard Header	45
Dashboard Graphs	45
Dashboard Booking List	49
Fetch.php	51
Code	53
Adam Dowling.....	53
Add New Stock Item.....	53
Delete Screen	59
Amend / View Screen.....	66
AI Disclosure Documents for Adam Dowling	72

Gleb Tutubalin.....	95
New Invoices from Supplier	95
Reorder Stock.....	100
Payment to Supplier.....	109
Unpaid Invoices Report.....	114
Muhammad Fahad.....	117
Make New Booking	118
Add Supplier.....	124
Delete Supplier.....	129
Amend / View Supplier	134

Garage Management Diagram

Below is a diagram of how the garage management system is structured. All of this is located in the sidebar that is on the left side of the admin dashboard.

(Adam Dowling)



File Structure

Garage-Management-System (Main Folder)

-Assets

- CSS** (includes all the CSS files)
- Graph_data** (include all the graph data files)
- images** (include all the images)
- js** (include all the JavaScript Files)

Rest of all the files

✓ GARAGE-MANAGEMENT-SYSTEM	
✓ .vscode	
{} settings.json	
> AI-Disclosure-Muhammad-fa...	
✓ Assets	
✓ css	
css Admindashboard.css	
css login.css	
css ShowSuppliers.css	
css sidebar.css	
✓ graph_data	
php bookingGraphData.php	
php incomeGraphData.php	
php jobTypeGraphData.php	
✓ images	
img admin_img.png	
img boooking.png	
img currency.png	
img currency1.png	
img customer.png	
img login_page_background.jpg	
img login_page_logo.png	
img search_icon.png	
✓ js	
JS admin_dashboard.js M	
JS side_bar.js M	
	add_customer_form.php add_customer.php addJobType.php addJobTypeForm.php AddNewBooking.php AddNewBookingForm.php AddNewStock.php addNewStock1.php 1 AddNewStockForm.php addNewStockPage1.php AddSupplier.php AddSupplierForm.php AdminDashboard.php AmendJobType.php AmendJobTypeForm.php AmendViewStock.php AmendViewStockPage.php amendViewSupplier.php amendViewSupplierPage.php bookingList.php completeJob.php completeJobForm.php DBconnection.php DeleteJobType.php DeleteJobTypePage.php DeleteStock.php DeleteStockPage.php deleteSupplier.php deleteSupplierPage.php fetch.php fetchsupplier.php getStockItem.php header.php JobTypeList.php login.php README.md reorderStockForm.php sidebar.php stockList.php suppliersList.php Table of attributes and ER dia... unpaidInvoices.php unpaidInvoicesReport.php

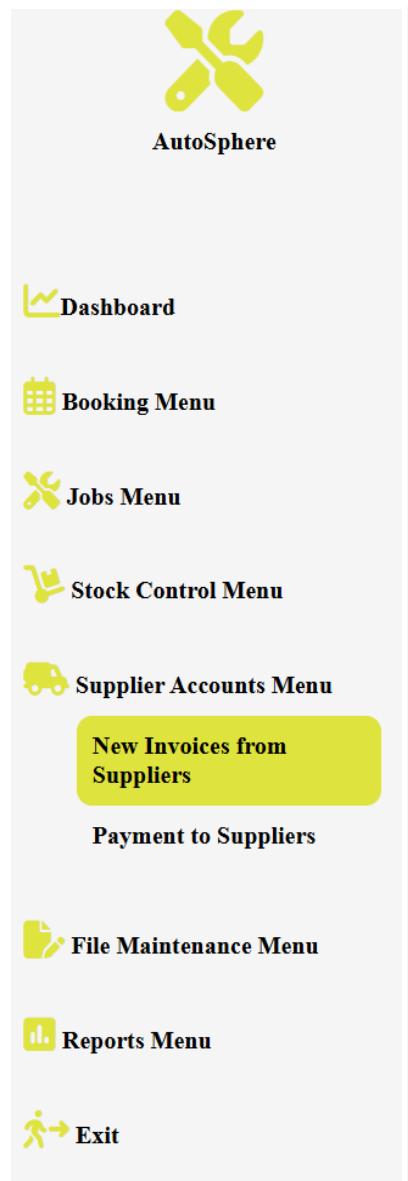
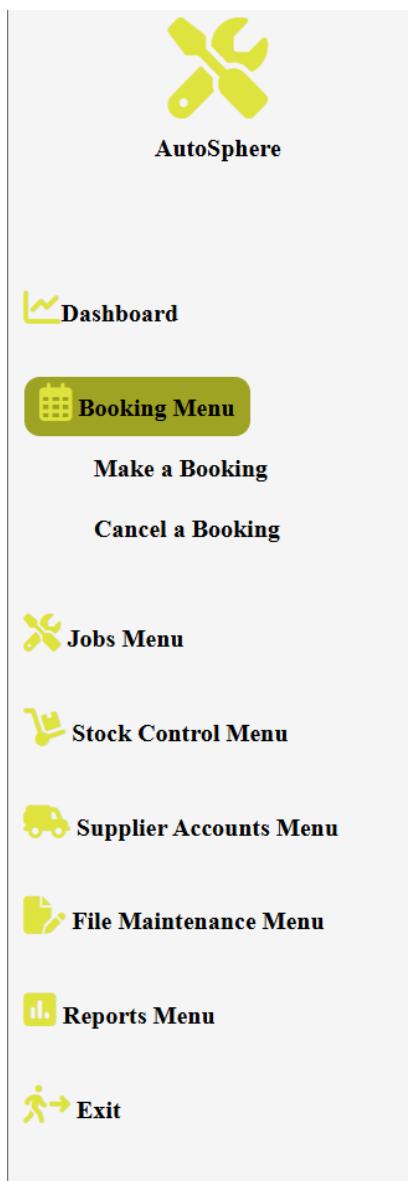
User Interface

Menu

The main menu includes the logo for the management system in the top left corner of the page along with the name of the site “AutoSphere”. There is also charts in the top-center of the screen where the total gross income, total garage bookings and all maintenance services can be viewed in a graph-like form. All of the images used for the main logo, visuals for the charts and images beside buttons are of green hue. Along with this, all text is black, and the majority of the background takes a white colour. To the left-hand side, the sidebar can be seen, which is the main tool used to navigate the garage management system. On the sidebar, the dashboard, booking menu, jobs menu, stock control menu, supplier accounts menu, file maintenance menu, reports menu and exit can be found. When these buttons are clicked, they collapse into subheadings that are also clickable (this is where all add, delete etc screens can be found). In the top right is the profile picture of the user logged in (hardcoded to be Admin1). The bulk of the main menu displays a list of bookings taken by the garage management system which is scrollable separate to the sidebar and the rest of the menu page.

Booking ID	Customer_ID	Customer Name	Email Address	Telephone	Car Model	Booking Date
40	10	taemour	testtseing@gmail.c	851569662	2009	2028-12-11
41	10	taemour	testtseing@gmail.c	851569662	1998	2028-11-17
43	5	Mashi	oke@gmail.com	8347384	2007	2028-09-09
39	5	Mashi	oke@gmail.com	8347384	2016	2027-07-09
57	18	umair	umair@hotmail.com	hffsdjflkdsh	2023	2026-12-12
58	19	tariq	tariq@yahoo.com	kljsjbv nsda	2025	2026-08-12
50	17	ahamd	ahamd@yahoo.com	0	2020	2026-03-12
48	5	Mashi	oke@gmail.com	8347384	2021	2025-12-18
56	16	john	john@gmail.com	0	2024	2025-12-12
45	5	Mashi	oke@gmail.com	8347384	2023	2025-12-12
60	17	ahamd	ahamd@yahoo.com	0	2025	2025-12-12

Continued....



Add Screens

All of the add screens for the system follow a similar structure where post methods are used to process the form entries, along with prepared statements which we used for SQL injection prevention while data is being inserted into the database. For the form itself, we used javascript to validate the data. This is done in different methods as some add screens link external javascript files such as NewStockValidation.js for the Add a new Stock Item screen and SupplierFormValidation.js for the Add a new Supplier screen, whereas the Add a new Customer screen uses internal javascript inside add_customer_form.php to validate the data. The validation for New Supplier Invoices is found in the HTML code for the form in newInvoicesForm.php. Each page/form file for our add screens posts to the php add file where the data can be successfully inserted into the database. The common CSS used for these files is the ShowSuppliers.css file which aids in displaying each page in the container, rather than redirecting the file to open up in a new tab. The button that needs to be pressed In order for new data to be inserted into the database is located at the bottom of each form, with an option to clear the fields.

Delete Screens

All of the delete screens goal is to set the deleted flag for each of the relevant tables in the database to 0, or to set a different status flag. In all of the delete screens there is a link to a dropdown list that appears at the top of the delete form which gives a list of all of the data which can be selected for deletion (JobTypeList.php, suppliersList.php, stockList.php), except the delete a customer screen uses an SQL query in the form to fetch the data to insert it into the dropdown list from there. The common CSS files used for these are the ShowSuppliers.css file (which serves the same purpose as it does in the add screen), and the Admindashboard.css file, which is used to style the dashboard header in all files and more styling for the container that each file will sit into once they have been clicked. There is also validation in each of these delete screens to ensure that no data can be set for deletion when there is an active use of the data somewhere else in the database. An example of this is stock items not being able to be deleted if it is already in stock or on order. Another example is a supplier can not be deleted if there is an unpaid invoice. The button that has to be pressed in order for the data to be flagged for deletion is located at the bottom of each form.

[Amend / View Screens](#)

The amend / view screens purpose is to be able to view the details of a certain attribute within a table in the database and be able to edit it and update the data, excluding primary keys. Some of the forms uses AJAX to fetch details when it has been selected from the dropdown menu and populates the fields. Prepared statements are used once again to prevent injection in the SQL queries. Validation is used in the amend / view files to ensure that no primary or foreign keys can be modified, also row count checks are carried out to make sure that the data has been updated in the database. The button that has to be pressed to amend the details is located at the bottom of each form, with an option to cancel which will populate the fields with the previous data that is stored in relation to the file.

Common files for all screens

The two common CSS files used across various screens whether they are add, delete, amend/view or other, are the files Admindashboard.css and ShowSuppliers.css. Admindashboard.css handles the layout of each page in relation to the display of the containers and how it remains a dynamic display. ShowSuppliers.css deals with actually displaying each form to the screen when clicked such as the different textbox functions etc. Two php files are used throughout different screens. The first one being DBconnection.php, this file ensures that there is a connection to the database. Without this file, no data would be able to be inserted, deleted or updated. This file includes the database name, the host and the password to the database. The other php file that is used frequently throughout any screens is fetch.php. This file is used for dropdown menus. It gets active data from multiple database tables and structures them so they can be used in forms that use AJAX.

Common CSS Files for All of the Forms And Pages OF AutoSphere (Garage Management System).

Side Bar sidebar.css File

```
1 /* File name : Admin Dashboard CSS
2 Description : This File contain the styling of for side bar.
3
4 Group Members :
5 C00298483 Adam Dowling
6 C00295140 Taemour Basharat
7 C00311349 Muhammad Fahad
8 C00297032 Emoshoke Saliu
9 C00290944 Gleb Tutubalin */
10
11 ul li{
12     padding: 10px ;
13 }
14 .dashboard_sidebar{
15     grid-area: sidebar;
16     background-color:var(--fourth_color);
17     display: flex;
18     flex-direction: column;
19     gap: 60px;
20 }
21 .logo_name_container{
22     display: flex;
23     flex-direction: column;
24     align-items: center;
25     justify-content: center;
26     width: 100%;
27     padding: var(--container_padding);
28     gap: 10px;
29 }
30
31 .logo{
32     font-size: 4rem;
33     color: var(--first_color);
34 }
35 .logo_name.Container .name{
36     font-size: 1rem;
37 }
38 /* navabr */
39 .dashboard_navbar ul, li, a {
40     text-decoration: none;
41     list-style: none;
42     color: black;
43     font-weight: bold;
44 }
45 .dashboard_navbar .main_menu_continer {
46     display: flex;
47     flex-direction: column;
48     justify-content: center;
49     gap: 15px;
50 }
51
52 .main_menu_continer .main_menu:hover {
53     background-color: var(--second_color);
54     padding: var(--buttonPadding);
55     border-radius: var(--border-radius);
56 }
57 .icon{
58     color: var(--first_color);
59     font-size: x-large;
60 }
61 .currency{
62     color: var(--second_color);
63 }
64 /* from here the styling for sub-menne is start */
65 .submenu{
66     display: none;
67     padding-left: 35px;
68 }
69 .submenu.show{
70     margin-top: 10px;
71     display: flex;
72     flex-direction: column;
73     align-items: first baseline;
74     text-align: left;
75     animation: fadeIn 0.4s ease-in-out;
76 }
77 .submenu li{
78     width: 200px;
79 }
80 .submenu li:hover{
81     text-overflow: ellipsis;
82     background-color: var(--first_color);
83     border-radius: var(--border-radius);
84 }
85 /* animation for sub-menne */
86 @keyframes fadeIn {
87     from{
88         opacity: 0; transform: translate(-5px);
89     }
90     to{
91         opacity: 1; transform: translate(0);
92     }
93 }
```

Dashboard Admin dashboard.css File

```
● ● ●
1 /* File name :      Admin Dashboard CSS
2 Description : This File contain the styling of Dashboard contents and its also a common styling file for all the pages
3 Group Members :
4 c00298483 Adam Dowling
5 C00295140 Taemour Sharat
6 C00311349 Muhammad Fahad
7 C00297032 Emoshoke Saliu
8 C00200944 Gleb Tutubalin */
9 /* Universal selector* used for reset the default styling of whole page */
10 * {
11   padding: 0;
12   margin: 0;
13   box-sizing: border-box;
14 }
15 body {
16   background-color: var(--third_color);
17 }
18 /* CSS variables store the styling values like colors , padding, marding , border style, then used it the calling them its name inside the var() */
19 :root {
20   --container_padding: 10px;
21   --border-radius: 10px;
22   --first_color: #dee3e;
23   --second_color: #9fa324;
24   --third_color: #e9e7e7;
25   --fourth_color: #000000;
26   --font_sub_color: #515153;
27   --fourth_color: #f5f5f5;
28   --buttonPadding: 10px;
29   --border-radius: 10px;
30 }
31 /* <!-- main container with display grip to handle the layout of page --> */
32 .main_container {
33   display: grid;
34   grid-template-columns: 270px auto;
35   grid-template-rows: auto auto 1fr;
36   grid-template-areas:
37     "sidebar header"
38     "sidebar dashboard"
39     "sidebar dashboard"
40     "sidebar dashboard"
41     "footer footer";
42 }
43 .admin_dashboard_main {
44   grid-area: dashboard;
45 }
46 /* this is styling for the Dashboard header which is include on all the files*/
47 .dashboard_header {
48   grid-area: header;
49   padding: 10px 0;
50   gap: 20px;
```

Continued....

```
1  text-align: center;
2  background-color: var(--third_color);
3  display: flex;
4  justify-content: space-around;
5  align-items: center;
6  background-color: var(--fourth_color);
7 }
8 .center_header input[type="search"] {
9  border: 2px solid rgba(0, 0, 0, 0.088);
10 border-radius: 10px;
11 background-image: url("./Assets/images/search_icon.png");
12 background-color: var(--fourth_color);
13 background-size: 20px;
14 background-position: 10px 10px;
15 background-repeat: no-repeat;
16 padding: 12px 20px 12px 40px;
17 }
18 input::placeholder {
19  padding: 5px;
20 }
21 .right_header {
22  display: flex;
23  align-items: center;
24  gap: 10px;
25 }
26 .right_header .admin_img {
27  width: 40px;
28  border-radius: 60%;
29  border: 1px solid black;
30  padding: 2px;
31 }
32 /* dashboard section ,this is the main container/section that contains the whole content of dashboard like dashboard GRAPHS and Booking Table*/
33 .dashboard_content_Section,
34 .delete_supplier_main_container {
35  padding: 20px;
36 }
37 .top_dashboard {
38  display: grid;
39  grid-template-columns: repeat(3, 350px);
40  grid-template-areas: "graph1 graph2 graph3";
41  column-gap: 10px;
42  justify-content: space-between;
43  padding: 0 0 10px 0;
44 }
45 .income_container_title {
46  width: 100%;
47  text-align: center;
48  padding: var(--container_padding);
49 }
```

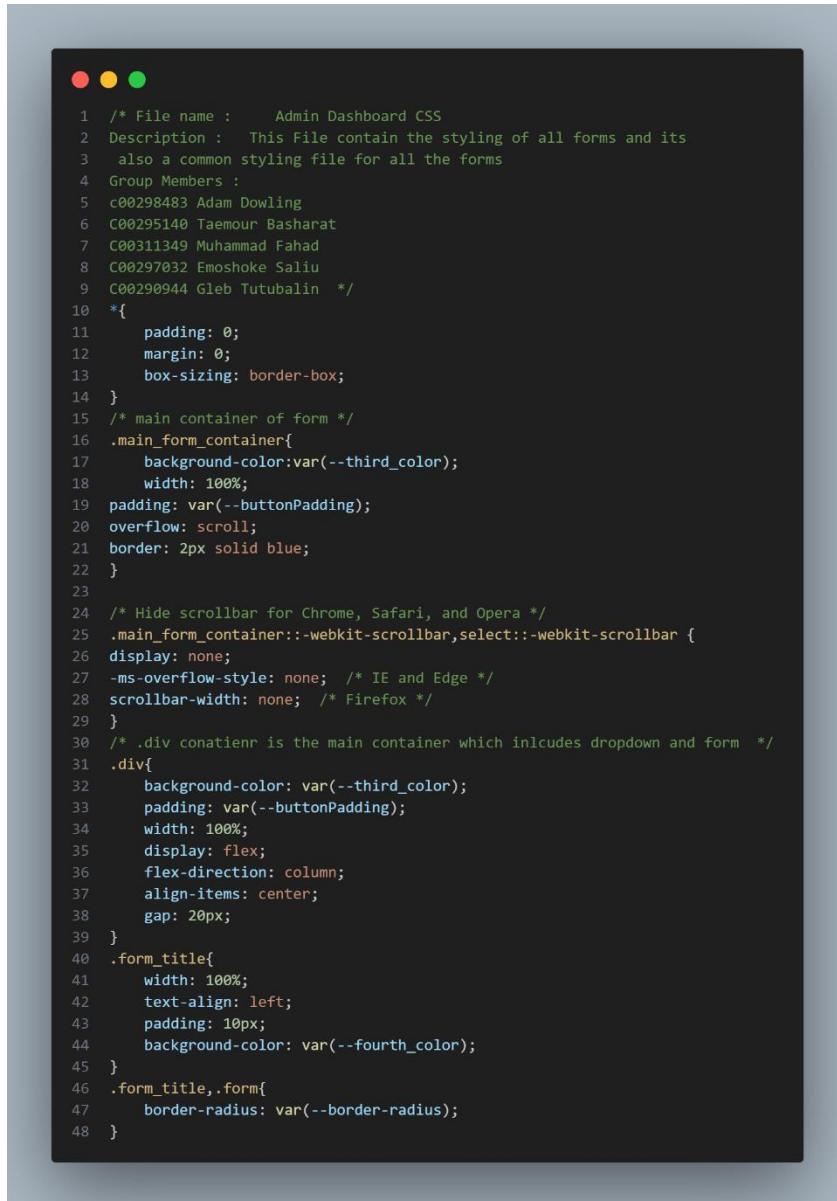
Continued....

```
1 .graph_container {  
2   background-color: var(--fourth_color);  
3   border-radius: 10px;  
4 }  
5 .currency {  
6   font-size: 40px;  
7 }  
8 .gc1 {  
9   grid-area: graph1;  
10 }  
11 .gc2 {  
12   grid-area: graph2;  
13 }  
14 .gc3 {  
15   grid-area: graph3;  
16 }  
17 .income_status {  
18   width: 100%;  
19   display: flex;  
20   flex-direction: row;  
21   justify-content: center;  
22   gap: 20px;  
23   align-items: center;  
24   text-align: center;  
25 }  
26 p {  
27   color: var(--font_sub_color);  
28   text-align: center;  
29 }  
30 p span {  
31   color: var(--second_color);  
32 }  
33 /* this is the style for all the tables that will be display on it  
34 delete supplier page */  
35 .recent_booking_section_container,  
36 .suppliers_list_section_container {  
37   background-color: var(--fourth_color);  
38   padding: 0 var(--buttonPadding);  
39   border-radius: 10px;  
40 }  
41 .booking_table_container,  
42 .supplier_table_container {  
43   width: 100%;  
44   height: 460px;  
45   background-color: var(--fourth_color);  
46   overflow: scroll;  
47   /* Hide scrollbar */  
48   -ms-overflow-style: none; /* IE and Edge */  
49   scrollbar-width: none; /* Firefox */  
50 }
```

Continued....

```
1  /* Hide scrollbar for Chrome, Safari, and Opera */
2  .main_form_container::-webkit-scrollbar {
3    display: none;
4  }
5  .heading,
6  .heading_container_supplier_list {
7    color: var(--font_sub_color);
8    padding: 20px 0;
9  }
10 .table {
11   width: 100%;
12   border-collapse: collapse;
13   position: sticky;
14 }
15 th,
16 td {
17   padding: 10px;
18 }
19 .table-column {
20   border: 2px solid black;
21 }
22 td {
23   border-bottom: 2px solid rgba(0, 0, 0, 0.064);
24   text-align: center;
25 }
26 .dashboard_recent_booking_container,
27 .supplier_list_container {
28   padding: 10px;
29   width: 100%;
30   height: 100%;
31 }
32 /* Delete supplier page >supplier drop down */
33 .delete_supplierheader {
34   display: flex;
35   width: 100%;
36 }
37 /* this is styling for the container where all the form and pages will display dynamically */
38 #dynamicDisplayForms {
39   width: 100%;
40   display: flex;
41   flex-direction: row;
42   justify-content: center;
43 }
```

Forms And Other Pages ShowSuppliers.css File.



```
1 /* File name : Admin Dashboard CSS
2 Description : This File contain the styling of all forms and its
3 also a common styling file for all the forms
4 Group Members :
5 c00298483 Adam Dowling
6 C00295140 Taemour Basharat
7 C00311349 Muhammad Fahad
8 C00297032 Emoshoke Saliu
9 C00290944 Gleb Tutubalin */
10 *{
11     padding: 0;
12     margin: 0;
13     box-sizing: border-box;
14 }
15 /* main container of form */
16 .main_form_container{
17     background-color: var(--third_color);
18     width: 100%;
19     padding: var(--buttonPadding);
20     overflow: scroll;
21     border: 2px solid blue;
22 }
23
24 /* Hide scrollbar for Chrome, Safari, and Opera */
25 .main_form_container::-webkit-scrollbar, select::-webkit-scrollbar {
26     display: none;
27     -ms-overflow-style: none; /* IE and Edge */
28     scrollbar-width: none; /* Firefox */
29 }
30 /* .div conatiern is the main container which includes dropdown and form */
31 .div{
32     background-color: var(--third_color);
33     padding: var(--buttonPadding);
34     width: 100%;
35     display: flex;
36     flex-direction: column;
37     align-items: center;
38     gap: 20px;
39 }
40 .form_title{
41     width: 100%;
42     text-align: left;
43     padding: 10px;
44     background-color: var(--fourth_color);
45 }
46 .form_title,.form{
47     border-radius: var(--border-radius);
48 }
```

Continued....



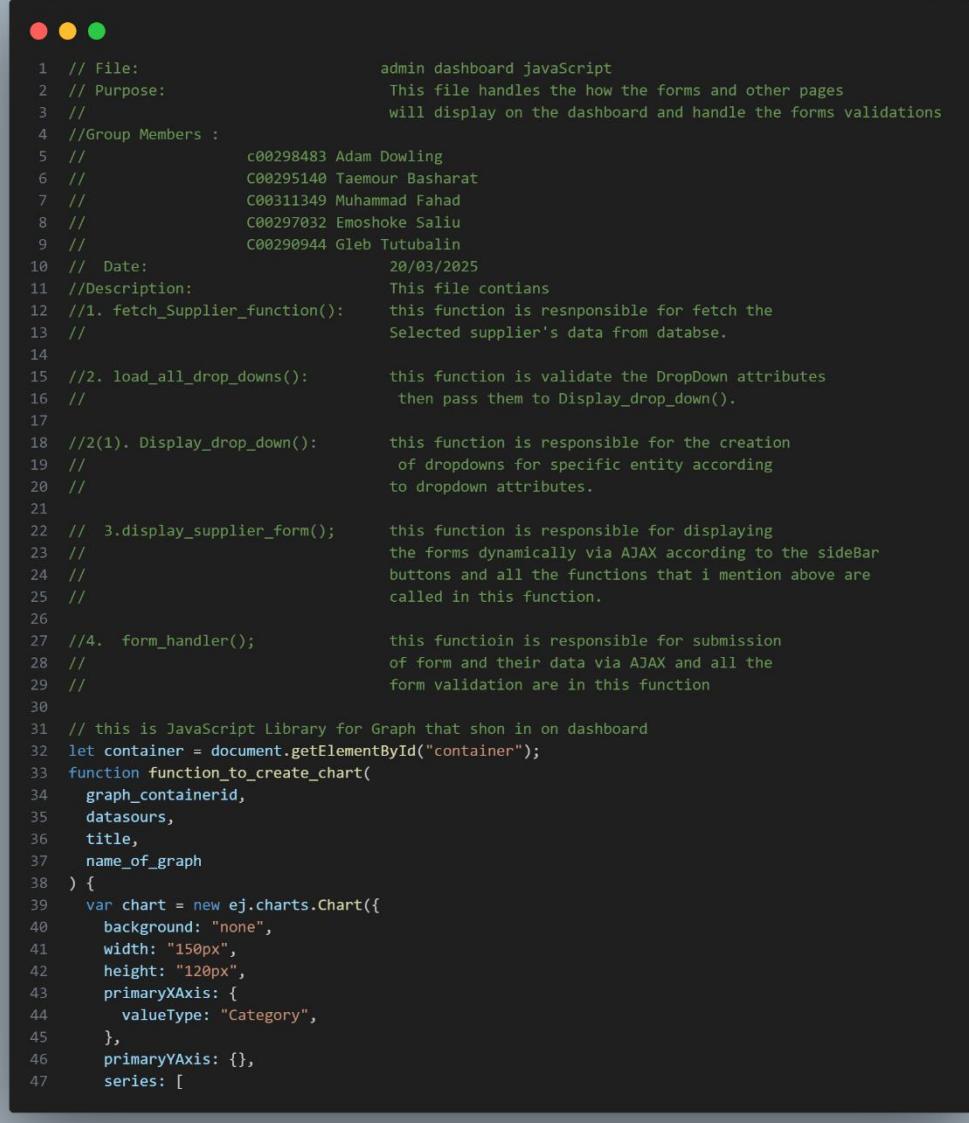
A screenshot of a code editor displaying a block of CSS code. The code is organized into several classes and a fieldset selector. It includes styling for the form itself, input fields, and a button group. The CSS uses modern properties like flexbox and calc(). The code editor has a dark theme with red, yellow, and green window control buttons at the top.

```
1 .form{  
2     width: 100%;  
3     background-color: var(--fourth_color);  
4     display: flex;  
5     flex-direction: column;  
6     justify-content: center;  
7     align-items: center;  
8     gap: 40px;  
9     margin: 0 auto;  
10    padding: calc(var(--buttonPadding) * 2);  
11 }  
12 .inputMain{  
13     width: 100%;  
14     padding: var(--buttonPadding);  
15     border-radius: var(--border-radius);  
16     display: flex;  
17     flex-direction: column;  
18     justify-content: left;  
19     &:hover{  
20         box-shadow: 1px 3px 7px -2px black;  
21     }  
22 }  
23 .feildset{  
24     height: fit-content;  
25     width: 100%;  
26 }  
27 input[type=text],  
28 input[type=email],  
29 input[type=date],  
30 input[type=number],  
31 input[type=url],  
32 select,  
33 textarea{  
34     width: calc(100% - 20px);  
35     padding: 8px;  
36     margin-top: 5px;  
37     border-radius: 8px;  
38     border: 1px solid #ddd;  
39     border: none;  
40     background: none;  
41     border-bottom: 1px solid rgba(0, 0, 0, 0.555);  
42 }  
43 .form_buttons{  
44     display: flex;  
45     gap: 30px;  
46 }
```

Continued....

```
● ● ●
1 .formButton {
2   width: 150px;
3   background-color: var(--first_color);
4   border: none;
5   border-radius: var(--border-radius);
6   padding: 10px;
7   cursor: pointer;
8   color: var(--color01);
9   font-size: 1rem;
10  transition: transform 0.02s ease-in-out, background-color 0.2s ease-in-out;
11 }
12 .formButton:hover {
13   background-color: var(--second_color);
14   color: var(--third_color);
15   border: 2px solid var(--);
16 }
17 .formButton:active {
18   transform: scale(0.5);
19 }
20 /* delete supplier */
21 .supplier_delete_list_container{
22   display: flex;
23   justify-content: center;
24   align-items: center;
25   flex-direction: row;
26   border: 2px solid rgba(19, 18, 18, 0.234);
27   width: 100%;
28 }
29 .sup_list{
30   display: flex;
31   justify-content: center;
32   align-items: center;
33   flex-direction: row;
34   width: 50%;
35 }
36 .sup_but{
37   text-align: right;
38   width: 50%;
39 }
40 /* amend view supplier */
41 .overflow_container{
42   overflow: scroll;
43   border: 2px solid red;
44 }
```

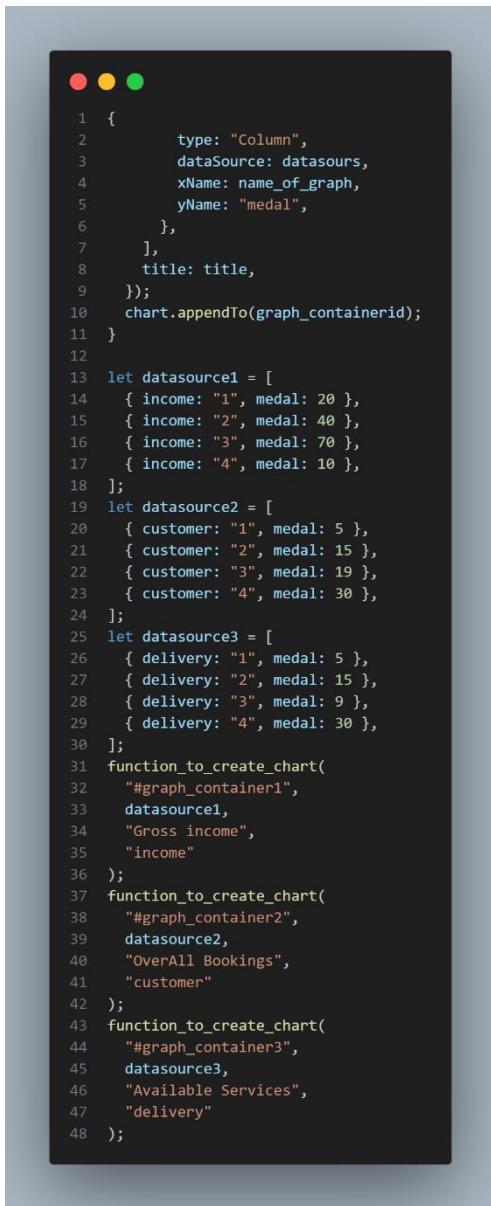
Common Java Script admin_dashboard.js File That Handles the Overall Layout and how the forms and other pages are display on the Dashboard Dynamically



```
1 // File: admin dashboard javaScript
2 // Purpose: This file handles the how the forms and other pages
3 // will display on the dashboard and handle the forms validations
4 //Group Members :
5 // c00298483 Adam Dowling
6 // C00295140 Taemour Basharat
7 // C00311349 Muhammad Fahad
8 // C00297032 Emoshoke Saliu
9 // C00290944 Gleb Tutubalin
10 // Date: 20/03/2025
11 //Description: This file contians
12 //1. fetch_Supplier_function(): this function is resposnible for fetch the
13 // Selected supplier's data from databse.
14
15 //2. load_all_drop_downs(): this function is validate the DropDown attributes
16 // then pass them to Display_drop_down().
17
18 //2(1). Display_drop_down(): this function is responsible for the creation
19 // of dropdowns for specific entity according
20 // to dropdown attributes.
21
22 // 3.display_supplier_form(): this function is responsible for displaying
23 // the forms dynamically via AJAX according to the sideBar
24 // buttons and all the functions that i mention above are
25 // called in this function.
26
27 //4. form_handler(): this functioin is responsible for submission
28 // of form and their data via AJAX and all the
29 // form validation are in this function
30
31 // this is JavaScript Library for Graph that shon in on dashboard
32 let container = document.getElementById("container");
33 function function_to_create_chart(
34   graph_containerid,
35   datasours,
36   title,
37   name_of_graph
38 ) {
39   var chart = new ej.charts.Chart({
40     background: "none",
41     width: "150px",
42     height: "120px",
43     primaryXAxis: {
44       valueType: "Category",
45     },
46     primaryYAxis: {},
47     series: [

```

continued....



```
1  {
2     type: "Column",
3     dataSource: datasours,
4     xName: name_of_graph,
5     yName: "medal",
6     },
7   ],
8   title: title,
9 });
10 chart.appendTo(graph_containerid);
11 }
12
13 let datasource1 = [
14   { income: "1", medal: 20 },
15   { income: "2", medal: 40 },
16   { income: "3", medal: 70 },
17   { income: "4", medal: 10 },
18 ];
19 let datasource2 = [
20   { customer: "1", medal: 5 },
21   { customer: "2", medal: 15 },
22   { customer: "3", medal: 19 },
23   { customer: "4", medal: 30 },
24 ];
25 let datasource3 = [
26   { delivery: "1", medal: 5 },
27   { delivery: "2", medal: 15 },
28   { delivery: "3", medal: 9 },
29   { delivery: "4", medal: 30 },
30 ];
31 function_to_create_chart(
32   "#graph_container1",
33   datasource1,
34   "Gross income",
35   "income"
36 );
37 function_to_create_chart(
38   "#graph_container2",
39   datasource2,
40   "OverAll Bookings",
41   "customer"
42 );
43 function_to_create_chart(
44   "#graph_container3",
45   datasource3,
46   "Available Services",
47   "delivery"
48 );
```

Continued....

```
1 // this function with AJAX is for fetch th selected supplier's data from database and display them in form
2 let fetch_Supplier_function = () => {
3   let selected_supplier = document.querySelector(
4     "[data-amend-supplier-dropdown='Supplier']")
5   );
6   // null check for the form who dont have supplier dropdown
7   if (!selected_supplier) {
8     console.log("No supplier dropdown found in this form");
9     return;
10 }
11 selected_supplier.addEventListener("change", (s) => {
12   let supplierID = s.target.value;
13   console.log("supplier ID is: ", supplierID);
14   if (supplierID) {
15     // fetch data from database and display them in form
16     fetch('fetchsupplier.php', {
17       method: "POST",
18       headers: {
19         "Content-Type": "application/x-www-form-urlencoded",
20       },
21
22       body: `supplier_id=${supplierID}`
23     })
24     .then((response) => {
25       // debugging in case of any errors
26       if (!response.ok) {
27         throw new Error(
28           `HTTP error! Status: ${response.status}, Message: ${response.statusText}`
29         );
30       }
31       return response.json();
32     })
33     .then((data) => {
34       if (!data) {
35         console.log("data not found", data);
36       }
37       console.log("Data received:", data);
38       document.getElementById("id").value = data.Supplier_ID;
39       document.getElementById("supplier_name").value = data.Name;
40       document.getElementById("supplier_email").value = data.Email;
41       document.getElementById("supplier_telephone").value = data.Telephone;
42       document.getElementById("supplier_address").value = data.Address;
43       document.getElementById("web_url").value = data.web_url;
44     })
45     .catch((error) =>
46       console.error(
47         "Error during fetching supplier details from dropdown",
48         error
49       )
50     );
51   }
52 });
53 };
```

Continued....

```
1 //function for automatically load the drop list
2 function load_all_drop_downs() {
3     // get all the dropdown from the form
4     let all_drop_downs = document.querySelectorAll(".drop_down_list");
5     all_drop_downs.forEach((drop_down) => {
6         let file_link = drop_down.getAttribute("data_file_link");
7         let id = drop_down.getAttribute("data_value");
8         let text = drop_down.getAttribute("data_text");
9         if (!file_link || !drop_down.id || !id || !text) {
10             alert("This Button is not active yet for:", drop_down.id);
11             return;
12         }
13         Display_drop_down(file_link, drop_down.id, id, text);
14     });
15 }
16 // this function is for display the drop down list inside the form for selecting the details
17 let Display_drop_down = (link, drop_down_table, id, text) => {
18     console.log("fetching data from database (fetch.php).", link);
19     fetch(link)
20         .then((response) => {
21             if (!response.ok) {
22                 throw new Error(
23                     `HTTP error! Status: ${response.status}, Message: ${response.statusText}`
24                 );
25             }
26             return response.json();
27         })
28         .then((data) => {
29             console.log("Data received:", data); // Debugging
30             let drop_down = document.getElementById(drop_down_table);
31             if (!drop_down) {
32                 console.error("Drop down element not found(form_ajax.js).");
33             }
34             drop_down.innerHTML = "<option value=''-Select---</option>"; // Clear existing options
35             let items = [];
36             if (drop_down_table === "Customer") {
37                 items = data.customer;
38             } else if (drop_down_table === "Supplier") {
39                 items = data.supplier;
40             } else if (drop_down_table === "Job_Type") {
41                 items = data.job_type;
42             } else if (drop_down_table === "Stock_Item") {
43                 items = data.stock_item;
44             }
45             items.forEach((item) => {
46                 let option = document.createElement("option");
47                 option.value = item[id];
48                 option.textContent = item[text];
49                 drop_down.appendChild(option);
50             });
51         })
52         .catch((error) => console.error("Error during submitting form", error));
53     };

```

Continued....

```

1 // from here the JavaScript (AJAX start) for dynamically display all forms in dashboard
2 // And also for send data to database through AJAX
3 document.addEventListener("DOMContentLoaded", () => {
4 // here "event" is the parameter of function"
5 //event delegation: Event delegation is a technique in JavaScript where you attach a single
6 //event listener to a parent element instead of adding event listeners to multiple child elements.
7 // This takes advantage of event bubbling, which allows the event to "bubble up" from a child element to its parent.
8 let sidebar = document.querySelector(".dashboard_sidebar");
9 sidebar.addEventListener("click", (event) => {
10 if (event.target.classList.contains("load-file")) {
11     let file_links = event.target.getAttribute("file-link");
12     if (!file_links) {
13         alert("This button is not activate yet!.");
14         return;
15     } else {
16         event.preventDefault();
17         display_supplier_form(file_links);
18         // store the last opened page in local storage to prevent from go back to dashboard page even after reload the page
19         sessionStorage.setItem("lastOpenedForm", file_links);
20     }
21 }
22 });
23
24 // this function is for display the form and hide the dashboard content which is initially display in dashboard
25 let display_supplier_form = (file) => {
26 // AJAX start for display forms in dashboard
27 // fetch request the form from server
28 fetch(file)
29     // convert the server response into html "response is predefined"
30     .then((response) => response.text())
31     .then((data) => {
32         //access the dashboard content section for hiding while showing the form
33         let dashboard_content = document.getElementById("dashboard_content");
34         //access form container (this is not actual for container ) its form container in dashboard where we will show actual form dynamically
35         let formContainer = document.getElementById("dynamicDisplayForms");
36         //give the form data(actual form) to form variable which contain the dynamically form container
37         formContainer.innerHTML = data;
38         formContainer.style.display = "flex";
39         load_all_drop_downs();
40         form_handler();
41         fetch_Supplier_function();
42     })
43     .catch((error) => console.log("Error during loading form", error));
44 };

```

Continued....

```
1 // this function handles the form data then send it to php file then php file send data to database
2 let form_handler = () => {
3     let form = document.getElementById("form");
4     if (!form) return;
5     // Check if phone field exists in THIS form
6     let phoneInput = form.querySelector(['name="phone"], #supplier_telephone');
7     // Setup real-time validation ONLY if phone field exists
8     if (phoneInput) {
9         phoneInput.addEventListener("input", function () {
10            const isValid = /^[0-9]{10,20}$/.test(this.value.trim());
11            this.setCustomValidity(isValid ? "" : "Phone: 10-20 digits required");
12            this.reportValidity();
13        });
14    }
15    //Booking date validation setup
16    const bookingInput = form.querySelector(
17        '#booking_date, [name="booking_date"], [type="date"]'
18    );
19    if (bookingInput) {
20        // Real-time validation
21        bookingInput.addEventListener("input", function () {
22            this.setCustomValidity("");
23            this.reportValidity();
24        });
25    }
26
27    form.addEventListener("submit", (event) => {
28        // event.preventDefault(); : this stop the normal form submission (without AJAX)
29        event.preventDefault();
30        let isFormValid = true;
31
32        // validate form
33        if (phoneInput) {
34            const phoneValid = /^[0-9]{10,20}$/.test(phoneInput.value.trim());
35            if (!phoneValid) {
36                phoneInput.setCustomValidity("Invalid phone number");
37                phoneInput.reportValidity();
38                console.log("Phone validation failed");
39                isFormValid = false;
40            }
41        }
42
43        //Booking date validation
44        if (bookingInput) {
45            let selectedDate = new Date(bookingInput.value);
46            let today = new Date();
47
```

continued....

```
1 if (selectedDate < today) {
2     bookingInput.setCustomValidity(
3         "Please select today's date or a future date"
4     );
5     bookingInput.reportValidity();
6     isFormValid = false;
7 } else {
8     bookingInput.setCustomValidity(""); // Clear if valid
9 }
10 }

11 if (!isFormValid) return false;
12 // window.confirm : predefined function in JS whenever we submit form it will ask for confirmation
13 let is_confirm = window.confirm("Are you sure to make changes ?");
14 if (!is_confirm) {
15     // FormData() :predefined function that extrats the form data
16     return false;
17 }
18 let form_data = new FormData(form);
19 let action_url = form.getAttribute("action");
20 fetch(action_url, {
21     method: "POST",
22     body: form_data,
23 })
24 .then((response) => response.text())
25 .then((msg) => {
26     let messge = document.getElementById("message");
27     // show response (message success/error)
28     messge.innerHTML = msg;
29     // reset the form after send data to php file(AddSupplier)
30     form.reset();
31 })
32 .catch((error) => console.log("Error during submitting form", error));
33 });
34 });
35 };
36 let lastOpenedForm = sessionStorage.getItem("lastOpenedForm");
37 if (lastOpenedForm) {
38     display_supplier_form(lastOpenedForm);
39 }
40 });
41 
```

Side bar JavaScript side_bar.js that handles the dropdown functionality

```
1 // File:           Side bar JavaScript
2 // Purpose:        This file handles the Side Bar buttons initially
3 //                  the Sub-buttons/Sub-menu of each Main side bar
4 //                  button are hidden, they just appear when the user
5 //                  click on main button
6 // Created By :      Muhammad Fahad c00311349
7 // Group Members :  c00298483 Adam Dowling
8 //                   C00295140 Taemour Basharat
9 //                   C00311349 Muhammad Fahad
10 //                  C00297032 Emoshoke Saliu
11 //                  C00290944 Gleb Tutubalin
12 // Date:            25/03/2025
13 //
14 document.addEventListener("DOMContentLoaded", () => {
15     //Select all main menu items
16     let main_menus = document.querySelectorAll(".main_menu");
17
18     main_menus.forEach(menu => {
19         menu.addEventListener("click", (e) => {
20             e.preventDefault(); // Stop default link behavior
21
22             //Get the submenu next to the clicked menu
23             let sub_menu = menu.nextElementSibling;
24
25             //check if the submenu exists (some menus don't have submenus)
26             if (sub_menu && sub_menu.classList.contains("submenu")) {
27                 let isOpen = sub_menu.classList.contains("show");
28
29                 //Close all other submenus
30                 document.querySelectorAll(".submenu").forEach
31                 (sub => sub.classList.remove("show"));
32
33                 //Toggle the clicked submenu (show if it was hidden, hide if it was open)
34                 if (!isOpen) {
35                     sub_menu.classList.add("show");
36                 }
37             }
38         });
39     });
40 });
41 });
42 });
43 }
```

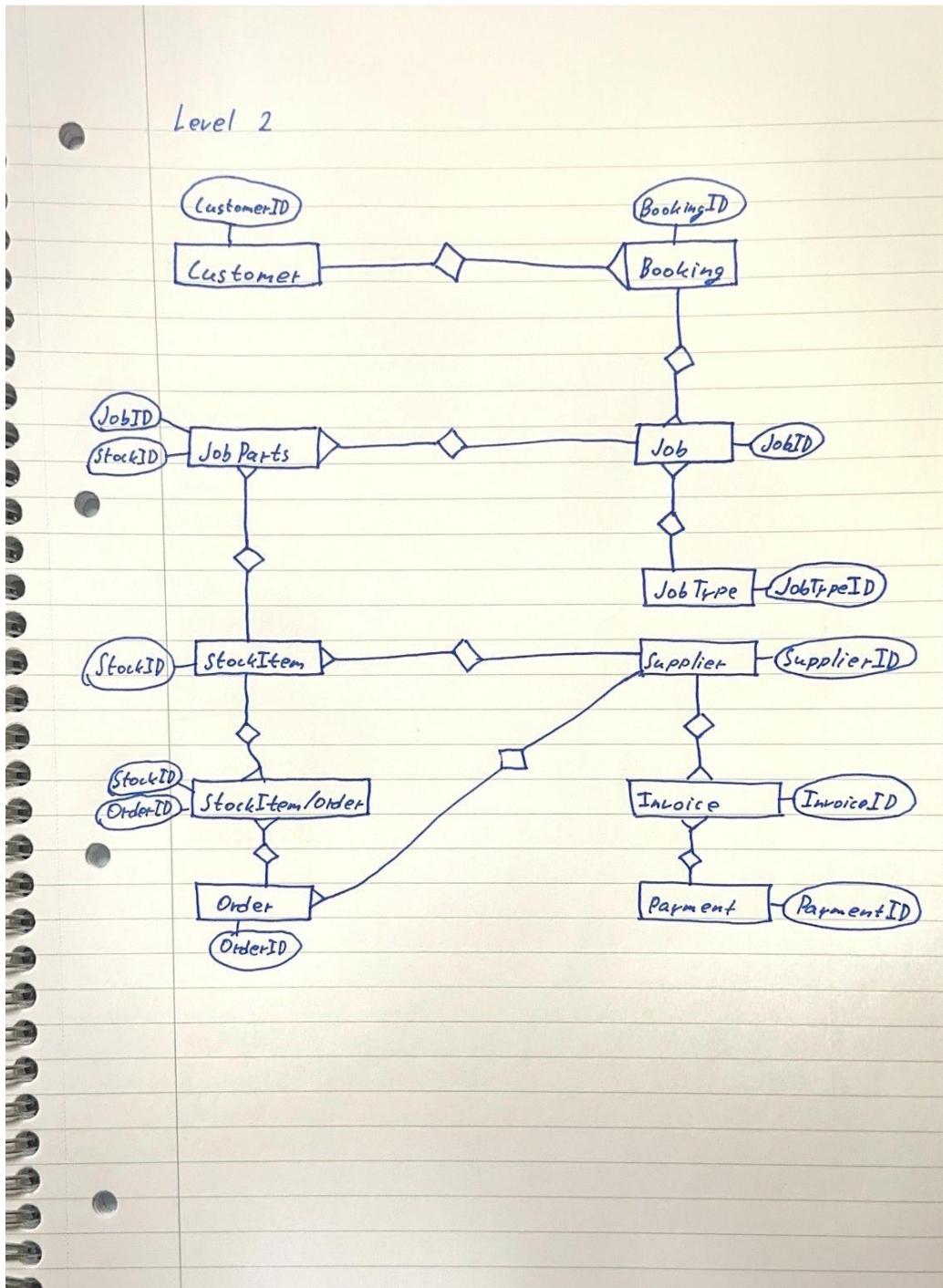
Database DBconnection.php

```
1  <!--
2      File:          DBconnection.php
3      Purpose:        contain the database connectivity
4      Created By:    Muhammad Fahad (c00311349)
5      Group Members :
6                  c00298483 Adam Dowling
7                  C00295140 Taemour Basharat
8                  C00311349 Muhammad Fahad
9                  C00297032 Emoshoke Saliu
10                 C00290944 Gleb Tutubalin
11             Date:        26/03/2025
12     -->
13 <!-- header -->
14 <?php
15 $hostName = "localhost";
16 $userName="garageuser";
17 $password="garageUserPass4";
18 $dbName="garage_system";
19 try {
20     // PDO > PHP data object
21     // create new pdo object to create connection with database
22     $con = new PDO("mysql:host=$hostName;dbname=$dbName;charset=utf8",$userName, $password);
23     // set PDO to throw exception on errors
24     $con->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
25 }
26 catch (PDOException $e) {
27     die("Database connection failed.: ".$e->getMessage());
28 }
29 ?>
```

Original ER Diagram

(Adam Dowling, Muhammad Fahad, Gleb Tutubalin, Taemour Basharat, Emoshoke Saliu)

Below is the 2nd level ER system diagram submitted in January 2025.



Database Tables

Booking Table

Screenshot of the MySQL Workbench interface showing the structure and data of the 'Booking' table.

Table Structure View:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	Booking_ID	int(11)			No	None	AUTO_INCREMENT		Change Drop More
2	Customer_ID	int(11)			No	None			Change Drop More
3	Model	varchar(20)	utf8_general_ci		No	None			Change Drop More
4	Registration_Number	varchar(11)	utf8_general_ci		No	None			Change Drop More
5	Mileage	int(11)			No	None			Change Drop More
6	Details	varchar(30)	utf8_general_ci		No	None			Change Drop More
7	Origin_Date	date			No	None			Change Drop More
8	Booking_Date	date			No	None			Change Drop More
9	Status	tinyint(1)			No	None			Change Drop More
10	Payment	tinyint(1)			No	None			Change Drop More
11	job_type_id	int(11)			No	None			Change Drop More

Indexes View:

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
	PRIMARY	BTREE	Yes	No	Booking_ID	19	A	No	

Data View:

Booking_ID	Customer_ID	Model	Registration_Number	Mileage	Details	Origin_Date	Booking_Date	Status	Payment	job_type_id
40	10	2009	34x2h9	87865	Check bateerry please	2025-03-22	2028-12-11	0	0	0
41	10	1998	132z6m	2847349	Engine Work	2025-03-22	2028-11-17	0	0	0
43	5	2007	23wq34r	12212123	just check	2025-03-23	2028-09-09	0	0	0
39	5	2016	343k8g	32244	please check wheel alignment	2025-03-22	2027-07-09	0	0	0
57	18	2023	12qw34rt	34532	Replace Battery old one was no	2025-03-26	2026-12-12	0	0	0
58	19	2025	23cv56	29898		2025-03-26	2026-08-12	0	0	0
50	17	2020	abs32	123421	Sometimes engine make too noise	2025-03-25	2026-03-12	0	0	0
48	5	2021	12qw34rt5	12345	headlight work	2025-03-24	2025-12-18	0	0	0
45	5	2023	98v56w	84758	Please can you change the head	2025-03-23	2025-12-12	0	0	0
56	16	2024	12qw34	2322	work on engine	2025-03-26	2025-12-12	0	0	0
36	10	2002	1232343d21q	243424	sometimes the engine becomes t	2025-03-21	2025-12-11	0	0	0
53	7	2024	8718vc	12322		2025-03-25	2025-09-11	0	0	0
49	5	1998	23qa4	12345	detail work	2025-03-24	2025-09-09	0	0	0
52	10	2009	12er42	123212	replaced battery	2025-03-25	2025-08-12	0	0	0
37	5	1998	23c34u	283746	Battery Work	2025-03-21	2025-07-09	0	0	0
55	7	2025	23asd21	123422	Replace Battery	2025-03-25	2025-06-12	0	0	0
44	5	2023	08v39x	43434	Wash the car please	2025-03-23	2025-06-09	0	0	0
38	5	2021	2q13c45	2334	Battery is not working	2025-03-21	2025-05-25	0	0	0
54	5	2002	1223wq2	12121	check it out	2025-03-25	2025-04-12	0	0	0

Screenshot of the MySQL Workbench interface showing the data of the 'Booking' table.

Data View:

Booking_ID	Customer_ID	Model	Registration_Number	Mileage	Details	Origin_Date	Booking_Date	Status	Payment	job_type_id
40	10	2009	34x2h9	87865	Check bateerry please	2025-03-22	2028-12-11	0	0	0
41	10	1998	132z6m	2847349	Engine Work	2025-03-22	2028-11-17	0	0	0
43	5	2007	23wq34r	12212123	just check	2025-03-23	2028-09-09	0	0	0
39	5	2016	343k8g	32244	please check wheel alignment	2025-03-22	2027-07-09	0	0	0
57	18	2023	12qw34rt	34532	Replace Battery old one was no	2025-03-26	2026-12-12	0	0	0
58	19	2025	23cv56	29898		2025-03-26	2026-08-12	0	0	0
50	17	2020	abs32	123421	Sometimes engine make too noise	2025-03-25	2026-03-12	0	0	0
48	5	2021	12qw34rt5	12345	headlight work	2025-03-24	2025-12-18	0	0	0
45	5	2023	98v56w	84758	Please can you change the head	2025-03-23	2025-12-12	0	0	0
56	16	2024	12qw34	2322	work on engine	2025-03-26	2025-12-12	0	0	0
36	10	2002	1232343d21q	243424	sometimes the engine becomes t	2025-03-21	2025-12-11	0	0	0
53	7	2024	8718vc	12322		2025-03-25	2025-09-11	0	0	0
49	5	1998	23qa4	12345	detail work	2025-03-24	2025-09-09	0	0	0
52	10	2009	12er42	123212	replaced battery	2025-03-25	2025-08-12	0	0	0
37	5	1998	23c34u	283746	Battery Work	2025-03-21	2025-07-09	0	0	0
55	7	2025	23asd21	123422	Replace Battery	2025-03-25	2025-06-12	0	0	0
44	5	2023	08v39x	43434	Wash the car please	2025-03-23	2025-06-09	0	0	0
38	5	2021	2q13c45	2334	Battery is not working	2025-03-21	2025-05-25	0	0	0
54	5	2002	1223wq2	12121	check it out	2025-03-25	2025-04-12	0	0	0

Customer Table

Server: localhost:3306 » Database: garage_system » Table: Customer

[Browse](#) [Structure](#) [SQL](#) [Search](#) [Insert](#) [Export](#) [Import](#) [Operations](#) [Tracking](#) [Triggers](#)

[Table structure](#) [Relation view](#)

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	Customer_ID	int(11)	utf8_general_ci		No	None		AUTO_INCREMENT	Change Drop More
2	Name	varchar(25)	utf8_general_ci		No	None			Change Drop More
3	Address	varchar(50)	utf8_general_ci		No	None			Change Drop More
4	Telephone	varchar(11)	utf8_general_ci		No	None			Change Drop More
5	E_Mail	varchar(20)	utf8_general_ci		No	None			Change Drop More
6	Delete_Flag	tinyint(1)			No	None			Change Drop More

[Check all](#) With selected: [Browse](#) [Change](#) [Drop](#) [Primary](#) [Unique](#) [Index](#) [Spatial](#) [Fulltext](#)

[Print](#) [Propose table structure](#) [Track table](#) [Move columns](#) [Normalise](#)

[Add](#) 1 column(s) after Delete_Flag [Go](#)

[Indexes](#)

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Rename Drop	PRIMARY	BTREE	Yes	No	Customer_ID	13	A	No	

Server: localhost:3306 » Database: garage_system » Table: Customer

[Browse](#) [Structure](#) [SQL](#) [Search](#) [Insert](#) [Export](#) [Import](#) [Operations](#) [Tracking](#) [Triggers](#)

Showing rows 0 - 14 (15 total, Query took 0.0002 seconds.)

SELECT * FROM `Customer`

[Profiling](#) [Edit inline](#) [Edit](#) [Explain SQL](#) [Create PHP code](#) [Refresh](#)

Show all | Number of rows: 25 Sort by key: None

[Extra options](#)

	Customer_ID	Name	Address	Telephone	E_Mail	Delete_Flag
Edit Copy Delete	5	Mashi	104 Arm Strong	8347384	oke@gmail.com	0
Edit Copy Delete	6	hello	98 lake jack	899887898	hello@gmail.com	1
Edit Copy Delete	7	testing 1	111testing1	111	testing1@gms.com	0
Edit Copy Delete	8	testing 2	111testing1	111	testing1@gms.com	0
Edit Copy Delete	9	testing 2	2testing2	222	testing2@gms.com	0
Edit Copy Delete	10	taemour	setu carlow it university	851569662	testtesteing@gmail.c	0
Edit Copy Delete	11	hello	98 southern house	899887878	hello@gmail.com	0
Edit Copy Delete	12	goofy goober	goober house	899887878	goofy@goober.com	0
Edit Copy Delete	13	hello	23 hello	0	hello@hello.ie	1
Edit Copy Delete	14	hello	23 hello	0	hello@hello.ie	0
Edit Copy Delete	15	hello	23 hello	899889888	hello@hello.ie	0
Edit Copy Delete	16	john	23 pearse road	0	john@gmail.com	0
Edit Copy Delete	17	ahamd	34 old road	0	ahamd@yahoo.com	0
Edit Copy Delete	18	umair	78 armstrong range	hffsdjfkdh	umair@hotmail.com	0
Edit Copy Delete	19	tariq	67 fordgeland	klsjbv nsda	tariq@yahoo.com	0

[Check all](#) With selected: [Edit](#) [Copy](#) [Delete](#) [Export](#)

Show all | Number of rows: 25 Sort by key: None

Invoice Table

Server: localhost:3306 » Database: garage_system » Table: Invoice

[Browse](#) [Structure](#) [SQL](#) [Search](#) [Insert](#) [Export](#) [Import](#) [Operations](#) [Tracking](#) [Triggers](#)

[Table structure](#) [Relation view](#)

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	Invoice_ID	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	Supplier_ID	varchar(11)	utf8_general_ci		No	None			Change Drop More
3	Invoice_Ref	varchar(11)	utf8_general_ci		No	None			Change Drop More
4	Invoice_Date	date			No	None			Change Drop More
5	Payment_ID	varchar(11)	utf8_general_ci		No	None			Change Drop More
6	Amount	int(10)			No	None			Change Drop More
7	Status	varchar(15)	utf8_general_ci		No	ISSUED			Change Drop More

[Check all](#) [With selected:](#) [Browse](#) [Change](#) [Drop](#) [Primary](#) [Unique](#) [Index](#) [Spatial](#) [Fulltext](#)

[Print](#) [Propose table structure](#) [Track table](#) [Move columns](#) [Normalise](#)

[Add](#) column(s) [after Status](#) [Go](#)

[Indexes](#)

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Rename Drop	PRIMARY	BTREE	Yes	No	Invoice_ID	8	A	No	

Server: localhost:3306 » Database: garage_system » Table: Invoice

[Browse](#) [Structure](#) [SQL](#) [Search](#) [Insert](#) [Export](#) [Import](#) [Operations](#) [Tracking](#) [Triggers](#)

Showing rows 0 - 7 (8 total, Query took 0.0002 seconds.)

```
SELECT * FROM `Invoice`
```

[Profiling](#) [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)] [[Refresh](#)]

[Show all](#) | Number of rows: [25](#) Filter rows: Sort by key: [None](#)

[Extra options](#)

	Edit	Copy	Delete	Invoice_ID	Supplier_ID	Invoice_Ref	Invoice_Date	Payment_ID	Amount	Status
<input type="checkbox"/>	Edit	Copy	Delete	45	25	A13	2025-03-25	20	1500	PAID
<input type="checkbox"/>	Edit	Copy	Delete	46	25	B45	2025-03-25	20	1800	PAID
<input type="checkbox"/>	Edit	Copy	Delete	47	27	GH89JI	2025-03-25	16	568	PAID
<input type="checkbox"/>	Edit	Copy	Delete	48	37	FG67IO90	2025-03-25		45	ISSUED
<input type="checkbox"/>	Edit	Copy	Delete	49	34	G76FG	2025-03-25		35	ISSUED
<input type="checkbox"/>	Edit	Copy	Delete	50	33	JIUJ90	2025-03-24	17	10000	PAID
<input type="checkbox"/>	Edit	Copy	Delete	51	35	FG89	2025-03-25	18	5600	PAID
<input type="checkbox"/>	Edit	Copy	Delete	52	50	FGT789	2025-03-26	19	450	PAID

[Check all](#) [With selected:](#) [Edit](#) [Copy](#) [Delete](#) [Export](#)

[Show all](#) | Number of rows: [25](#) Filter rows: Sort by key: [None](#)

Job Table

Server: localhost:3306 » Database: garage_system » Table: Job

[Browse](#) [Structure](#) [SQL](#) [Search](#) [Insert](#) [Export](#) [Import](#) [Operations](#) [Tracking](#) [Triggers](#)

[Table structure](#) [Relation view](#)

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	Job_ID	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	Booking_ID	int(11)			No	None			Change Drop More
3	Job_Type_ID	int(11)			No	None			Change Drop More
4	Time_Taken	int(11)			No	None			Change Drop More
5	Status	varchar(20)	utf8_general_ci		No	None			Change Drop More

[Check all](#) *With selected:* [Browse](#) [Change](#) [Drop](#) [Primary](#) [Unique](#) [Index](#) [Spatial](#) [Fulltext](#)

[Print](#) [Propose table structure](#) [Track table](#) [Move columns](#) [Normalise](#)

[Add](#) column(s) [after Status](#) [Go](#)

[Indexes](#)

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Rename Drop	PRIMARY	BTREE	Yes	No	Job_ID	25	A	No	

Create an index on columns [Go](#)

Server: localhost:3306 » Database: garage_system » Table: Job

[Browse](#) [Structure](#) [SQL](#) [Search](#) [Insert](#) [Export](#) [Import](#) [Operations](#) [Tracking](#) [Triggers](#)

[Promoting](#) [Edit inline](#) [Edit](#) [Explain SQL](#) [Create PHP code](#) [Refresh](#)

Show all | Number of rows: Filter rows: Sort by key:

[Extra options](#)

	Job_ID	Booking_ID	Job_Type_ID	Time_Taken	Status
<input type="checkbox"/>	24	36	14	60	Completed
<input type="checkbox"/>	25	36	15	4	Completed
<input type="checkbox"/>	26	36	16	0	
<input type="checkbox"/>	27	37	10	1	Completed
<input type="checkbox"/>	28	37	15	0	
<input type="checkbox"/>	29	37	16	0	
<input type="checkbox"/>	30	38	10	0	
<input type="checkbox"/>	31	39	9	0	
<input type="checkbox"/>	32	40	10	0	
<input type="checkbox"/>	33	41	9	0	
<input type="checkbox"/>	34	41	12	0	
<input type="checkbox"/>	35	42	12	20	Completed
<input type="checkbox"/>	36	44	14	0	
<input type="checkbox"/>	38	46	12	0	
<input type="checkbox"/>	40	48	13	0	
<input type="checkbox"/>	41	49	9	0	
<input type="checkbox"/>	42	50	10	0	
<input type="checkbox"/>	49	56	13	0	
<input type="checkbox"/>	50	57	10	0	
<input type="checkbox"/>	51	58	13	0	

[Edit](#) [Copy](#) [Delete](#)

[Check all](#) *With selected:* [Edit](#) [Copy](#) [Delete](#) [Export](#)

Show all | Number of rows: Filter rows: Sort by key:

Job Parts Table

Server: localhost:3306 » Database: garage_system » Table: Job_Parts

Browse Structure SQL Search Insert Export Import Operations Tracking Triggers

Table structure Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	Job_Part_ID	int(11)			No	None		AUTO_INCREMENT	Change Drop Primary Unique Index Spatial Fulltext
2	Job_ID	int(11)			No	None			Change Drop Primary Unique Index Spatial Fulltext
3	Stock_ID	int(11)			No	None			Change Drop Primary Unique Index Spatial Fulltext
4	Quantity	int(11)			No	None			Change Drop Primary Unique Index Spatial Fulltext

Check all With selected: Browse Change Drop Primary Unique Index Spatial Fulltext

Print Propose table structure Track table Move columns Normalise

Add 1 column(s) after Quantity Go

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Rename Drop	PRIMARY	BTREE	Yes	No	Job_Part_ID	6	A	No	

Server: localhost:3306 » Database: garage_system » Table: Job_Parts

Browse Structure SQL Search Insert Export Import Operations Tracking Triggers

Showing rows 0 - 5 (6 total, Query took 0.0002 seconds.)

```
SELECT * FROM `Job_Parts`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all Number of rows: 25 Filter rows: Search this table Sort by key: None

Extra options

	Job_Part_ID	Job_ID	Stock_ID	Quantity
	1	24	1	2
	2	25	1	1
	3	25	1	1
	4	25	1	5
	5	27	1	1
	6	35	1	1

Check all With selected: Edit Copy Delete Export

Show all Number of rows: 25 Filter rows: Search this table Sort by key: None

Job Type Table

Server: localhost:3306 » Database: garage_system » Table: Job_Type

Browse Structure SQL Search Insert Export Import Operations Tracking Triggers

Table structure Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	Job_Type_ID	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	Job_Type_Name	varchar(30)	utf8_general_ci		No	None			Change Drop More
3	Description	varchar(25)	utf8_general_ci		No	None			Change Drop More
4	Price	int(11)			No	None			Change Drop More
5	Lead_Mechanic	varchar(25)	utf8_general_ci		No	None			Change Drop More
6	Delete_Flag	tinyint(4)			No	None			Change Drop More

Check all With selected: Change Drop Primary Unique Spatial

Add 1 column(s) after Delete_Flag

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Rename Drop	PRIMARY	BTREE	Yes	No	Job_Type_ID	6	A	No	

Server: localhost:3306 » Database: garage_system » Table: Job_Type

Browse Structure SQL Search Insert Export Import Operations Tracking Triggers

Showing rows 0 - 5 (6 total, Query took 0.0002 seconds.)

SELECT * FROM `Job_Type`

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

Extra options

	Job_Type_ID	Job_Type_Name	Description	Price	Lead_Mechanic	Delete_Flag
<input type="checkbox"/>	10	Battery work	444	33	1	0
<input type="checkbox"/>	12	paint	paint vehicle	55	1	0
<input type="checkbox"/>	13	Headlights Work	full	110	2	0
<input type="checkbox"/>	14	cleaning	clean car	50	1	0
<input type="checkbox"/>	15	Engine Tunning	nothing	23	2	0
<input type="checkbox"/>	17	Bumper Work	About Bumper	34	23	1

Check all With selected: Edit Copy Delete

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

Mechanics Table

Server: localhost:3306 » Database: garage_system » Table: Mechanics

Browse Structure SQL Search Insert Export Import Operations Tracking Triggers

Table structure Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	Mechanic_ID	int(11)	utf8_general_ci		No	None		AUTO_INCREMENT	<input type="button" value="Change"/> <input type="button" value="Drop"/> <input type="button" value="More"/>
2	Mechanic_Name	text	utf8_general_ci		No	None			<input type="button" value="Change"/> <input type="button" value="Drop"/> <input type="button" value="More"/>
3	Deleted_Flag	tinyint(4)			No	None			<input type="button" value="Change"/> <input type="button" value="Drop"/> <input type="button" value="More"/>

Check all With selected:

Add 1 column(s) after Deleted_Flag

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
<input type="button" value="Edit"/> <input type="button" value="Rename"/> <input type="button" value="Drop"/>	PRIMARY	BTREE	Yes	No	Mechanic_ID	3	A	No	

Server: localhost:3306 » Database: garage_system » Table: Mechanics

Browse Structure SQL Search Insert Export Import Operations Tracking Triggers

Showing rows 0 - 2 (3 total, Query took 0.0004 seconds.)

```
SELECT * FROM `Mechanics`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all Number of rows: 25 Filter rows: Search this table Sort by key: None

Extra options

	Mechanic_ID	Mechanic_Name	Deleted_Flag
<input type="checkbox"/> <input type="button" value="Edit"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/>	1	John carl	0
<input type="checkbox"/> <input type="button" value="Edit"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/>	2	Fred Jones	0
<input type="checkbox"/> <input type="button" value="Edit"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/>	3	Aine Byrne	0

Check all With selected:

Show all Number of rows: 25 Filter rows: Search this table Sort by key: None

Order Table

Server: localhost:3306 » Database: garage_system » Table: Order

Browse Structure SQL Search Insert Export Import Operations Tracking Triggers

Table structure Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	Order_ID	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	Supplier_ID	int(11)			No	None			Change Drop More
3	Order_Date	date			No	None			Change Drop More
4	Delivery_Date	date			No	None			Change Drop More
5	Status	tinyint(1)			Yes	0			Change Drop More

With selected: Browse Change Drop Primary Unique Index Spatial Fulltext

Print Propose table structure Track table Move columns Normalise

Add 1 column(s) after Status Go

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Rename Drop	PRIMARY	BTREE	Yes	No	Order_ID	13	A	No	

Server: localhost:3306 » Database: garage_system » Table: Order

Browse Structure SQL Search Insert Export Import Operations Tracking Triggers

Showing rows 0 - 12 (13 total, Query took 0.0003 seconds.)

SELECT * FROM `Order`

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

Extra options

	Order_ID	Supplier_ID	Order_Date	Delivery_Date	Status
Edit Copy Delete	1	30	2025-03-17	2025-03-27	0
Edit Copy Delete	2	33	2025-03-09	2025-03-26	0
Edit Copy Delete	17	50	2025-03-25	2025-04-08	0
Edit Copy Delete	19	25	2025-03-25	2025-04-08	0
Edit Copy Delete	20	50	2025-03-25	2025-04-08	0
Edit Copy Delete	21	50	2025-03-25	2025-04-08	0
Edit Copy Delete	23	50	2025-03-25	2025-04-08	0
Edit Copy Delete	24	50	2025-03-25	2025-04-08	0
Edit Copy Delete	25	25	2025-03-25	2025-04-08	0
Edit Copy Delete	26	50	2025-03-25	2025-04-08	0
Edit Copy Delete	27	25	2025-03-25	2025-04-08	0
Edit Copy Delete	28	25	2025-03-25	2025-04-08	0
Edit Copy Delete	29	50	2025-03-26	2025-04-09	0

With selected: Edit Copy Delete Export

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

Payment Table

Server: localhost:3306 » Database: garage_system » Table: Payment

[Browse](#) [Structure](#) [SQL](#) [Search](#) [Insert](#) [Export](#) [Import](#) [Operations](#) [Tracking](#) [Triggers](#)

[Table structure](#) [Relation view](#)

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 Payment_ID	int(11)			No	None		AUTO_INCREMENT	Change Drop More
<input type="checkbox"/>	2 Date_Of_Payment	date			No	None			Change Drop More
<input type="checkbox"/>	3 Total_Price	int(11)			No	None			Change Drop More

[Check all](#) *With selected:* [Browse](#) [Change](#) [Drop](#) [Primary](#) [Unique](#) [Index](#) [Spatial](#) [Fulltext](#)

[Print](#) [Propose table structure](#) [Track table](#) [Move columns](#) [Normalise](#)

[Add](#) column(s) [after Total_Price](#) [Go](#)

[Indexes](#)

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Rename Drop	PRIMARY	BTREE	Yes	No	Payment_ID	5	A	No	

Server: localhost:3306 » Database: garage_system » Table: Payment

[Browse](#) [Structure](#) [SQL](#) [Search](#) [Insert](#) [Export](#) [Import](#) [Operations](#) [Tracking](#) [Triggers](#)

Showing rows 0 - 4 (total, Query took 0.0002 seconds.)

```
SELECT * FROM `Payment`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 [Filter rows:](#) Search this table Sort by key: None

[Extra options](#)

	Payment_ID	Date_Of_Payment	Total_Price
<input type="checkbox"/>	16	2025-03-25	568
<input type="checkbox"/>	17	2025-03-25	10000
<input type="checkbox"/>	18	2025-03-25	5600
<input type="checkbox"/>	19	2025-03-26	450
<input type="checkbox"/>	20	2025-03-26	3300

[Edit](#) [Copy](#) [Delete](#) [Edit](#) [Copy](#) [Delete](#) [Edit](#) [Copy](#) [Delete](#) [Edit](#) [Copy](#) [Delete](#)

[Check all](#) *With selected:* [Edit](#) [Copy](#) [Delete](#) [Export](#)

[Show all](#) | Number of rows: 25 [Filter rows:](#) Search this table Sort by key: None

Stock Item Table (Adam Dowling)

Server: localhost:3306 » Database: garage_system » Table: Stock_Item

Browse Structure SQL Search Insert Export Import Operations Tracking Triggers

Table structure Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	Stock_ID	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	Description	varchar(30)	utf8_general_ci		No	None			Change Drop More
3	Stock_Quantity	int(11)			No	None			Change Drop More
4	Reorder_Level	int(11)			No	10			Change Drop More
5	Reorder_Quantity	int(11)			No	None			Change Drop More
6	Cost_Price	double			No	None			Change Drop More
7	Retail_Price	double			No	None			Change Drop More
8	Supplier_ID	int(11)		Yes	NULL				Change Drop More
9	Status	varchar(15)	utf8_general_ci		No	0			Change Drop More

Check all With selected: Browse Change Drop Primary Unique Index Spatial Fulltext

Print Propose table structure Track table Move columns Normalise

Add 1 column(s) after Status Go

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Rename Drop	PRIMARY	BTREE	Yes	No	Stock_ID	14	A	No	
Edit Rename Drop	FK_Supplier	BTREE	No	No	Supplier_ID	14	A	Yes	

Server: localhost:3306 » Database: garage_system » Table: Stock_Item

Browse Structure SQL Search Insert Export Import Operations Tracking Triggers

Showing rows 0 - 11 (12 total, Query took 0.0003 seconds.)

```
SELECT * FROM `Stock_Item`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all Number of rows: 25 Filter rows: Search this table Sort by key: None

Extra options

	Stock_ID	Description	Stock_Quantity	Reorder_Level	Reorder_Quantity	Cost_Price	Retail_Price	Supplier_ID	Status
	1	Wheel	-1	10	4	33	100	25	0
	4	Wheel Type 1	34	10	40	65	110	25	0
	5	Wheel Type 2	112	10	130	50	150	25	0
	6	Wheel Type 3	0	10	50	140	200	25	0
	7	Wheel Type 4	25	4	25	70	78	25	0
	13	Wheel Type 5	0	10	55	44	90	25	0
	14	Wheel Type 5	0	10	10	44	90	25	0
	15	Box of 500 Bolts	0	10	0	33	65	50	1
	18	Standard Box of Nails (50)	75	2	90	5	22	50	0
	21	Bolt & Nail Medium Box Set	20	10	0	20	80	50	0
	22	Head light	23	10	40	23	56	30	0
	23	Window Wiper	12	10	34	47	67	37	0

Check all With selected: Edit Copy Delete Export

Stock Item / Order Table (Adam Dowling)

Server: localhost:3306 » Database: garage_system » Table: Stock_Item/Order

[Browse](#) [Structure](#) [SQL](#) [Search](#) [Insert](#) [Export](#) [Import](#) [Operations](#) [Tracking](#) [Triggers](#)

[Table structure](#) [Relation view](#)

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	Stock_Item/Order_ID	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	Order_ID	int(11)			No	None			Change Drop More
3	Stock_ID	int(11)			No	None			Change Drop More
4	Quantity	int(5)			No	None			Change Drop More

[Check all](#) [With selected:](#) [Browse](#) [Change](#) [Drop](#) [Primary](#) [Unique](#) [Index](#) [Spatial](#) [Fulltext](#)

[Print](#) [Propose table structure](#) [Track table](#) [Move columns](#) [Normalise](#)

[Add](#) column(s) [after Quantity](#) [Go](#)

[Indexes](#)

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Rename Drop	PRIMARY	BTREE	Yes	No	Stock_Item/Order_ID	9	A	No	

Server: localhost:3306 » Database: garage_system » Table: Stock_Item/Order

[Browse](#) [Structure](#) [SQL](#) [Search](#) [Insert](#) [Export](#) [Import](#) [Operations](#) [Tracking](#) [Triggers](#)

Showing rows 0 - 8 (9 total, Query took 0.0005 seconds.)

```
SELECT * FROM `Stock_Item/Order`
```

[Profiling](#) [Edit inline](#) [Edit](#) [Explain SQL](#) [Create PHP code](#) [Refresh](#)

Show all | Number of rows: [25](#) Filter rows: Sort by key: [None](#)

[Extra options](#)

	Stock_Item/Order_ID	Order_ID	Stock_ID	Quantity
Edit Copy Delete	14	26	18	90
Edit Copy Delete	15	26	15	200
Edit Copy Delete	16	27	13	55
Edit Copy Delete	17	27	6	100
Edit Copy Delete	18	27	4	40
Edit Copy Delete	19	28	5	260
Edit Copy Delete	20	29	21	180
Edit Copy Delete	21	29	15	70
Edit Copy Delete	22	29	18	100

[Check all](#) [With selected:](#) [Edit](#) [Copy](#) [Delete](#) [Export](#)

[Show all](#) | Number of rows: [25](#) Filter rows: Sort by key: [None](#)

Supplier Table (Muhammad Fahad)

Server: localhost:3306 » Database: garage_system » Table: Supplier

Browse Structure SQL Search Insert Export Import Operations Tracking Triggers

Table structure Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	Supplier_ID	int(11)	utf8_general_ci		No	None		AUTO_INCREMENT	Change Drop Primary Unique Index Spatial Fulltext
2	Name	varchar(25)	utf8_general_ci		No	None			Change Drop Primary Unique Index Spatial Fulltext
3	Address	varchar(50)	utf8_general_ci		No	None			Change Drop Primary Unique Index Spatial Fulltext
4	Telephone	varchar(20)	utf8_general_ci		No	None			Change Drop Primary Unique Index Spatial Fulltext
5	Email	varchar(30)	utf8_general_ci		No	None			Change Drop Primary Unique Index Spatial Fulltext
6	web_url	varchar(30)	utf8_general_ci		No	None			Change Drop Primary Unique Index Spatial Fulltext
7	Delete_Flag	tinyint(1)			No	0			Change Drop Primary Unique Index Spatial Fulltext

Check all With selected: Browse Change Drop Primary Unique Index Spatial Fulltext

Print Propose table structure Track table Move columns Normalise

Add 1 column(s) after Delete_Flag Go

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Rename Drop	PRIMARY	BTREE	Yes	No	Supplier_ID	17	A	No	

Showing rows 0 - 13 (14 total). Query took 0.0004 seconds.

SELECT * FROM `Supplier`

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all Number of rows: 25 Filter rows: Search this table Sort by key: None

Supplier

Supplier_ID	Name	Address	Telephone	Email	web_url	Delete_Flag
25	Kyle Wheel House	Shop 98 pearse road	09948747938	kwh@gms.com	https://wheelHouse.com	0
29	Spare house Traders	19 dublin city center	9648294	spare23@gms.com	https://spareh11@hotmail.com	1
30	Kasha exterior house	09 classic building shop no 27	0335218474637	Kasha45@gms.com	https://kashaeexterior.ie	0
33	Anushka Traders	98 civil line road near civic centre	099890678783	anushka78@gms.com	https://anuskatrade.com.ie	0
34	Jack exhaust house	87 hill view centre shop no 76	00445329876884	jack98@gms.com	https://jackexhausthouse.com	0
35	Dunnes Traders	Shop 33 city centre galway	087797887787	traders98@hotmail.com	https://dunestraders.com.ie	0
37	Moeez Auto Parts	shop no 01 city center galway	08698798478475	moeez23@hotmail.com	https://autoparts.com	0
40	muhmmad fahad	armstrong centre	08786474648	fahadimran038@gmail.com	https://autogarage.ie	0
50	josh screw house	12 ashborke near city centre dublin	09488588574	screwhouse@hotmail.com	https://joshcrew.com	0
51	Adam Traders	12 cork road	09888888899	adamtraders@yahoo.co	https://testing.com	1
52	Tehseen Oil Tarders	12 tallagh centre dublin	089787347738	tehseen23@hotmail.com	https://tehseenTraders.ie	1
53	Asim Wheel House	shop 23 industrial state wexford	08974847398	asim23@yahoo.com	https://asoil.com	1
54	Kyle Exterior House	56 Ashbroke broke road enniscorthy	003353333677	kyleex@gmail.com	https://kyleExtHouse.ie	1
55	Maani Wire House	67 fordgeland enniscorthy county wexford	09878787675	maaniwire45@hotmail.com	https://wirehouse.ie	1

Check all With selected: Edit Copy Delete Export

Show all Number of rows: 25 Filter rows: Search this table Sort by key: None

GROUP WORK

Admin DashBoard



AutoSphere
Comprehensive auto management ecosystem

-  [Dashboard](#)
-  [Booking Menu](#)
-  [Jobs Menu](#)
-  [Stock Control Menu](#)
-  [Supplier Accounts Menu](#)
-  [File Maintenance Menu](#)
-  [Reports Menu](#)
-  [Exit](#)



19,998.00
Total gross income



20
Total Garage Bookings



6
All maintenance services

List of Bookings						
Booking ID	Customer_ID	Customer Name	Email Address	Telephone	Car Model	Booking Date
40	10	taemour	testesteing@gmail.c	851569662	2009	2028-12-11
41	10	taemour	testesteing@gmail.c	851569662	1998	2028-11-17
43	5	Mashi	oke@gmail.com	8347384	2007	2028-09-09
39	5	Mashi	oke@gmail.com	8347384	2016	2027-07-09
57	18	umair	umair@hotmail.com	hffsdjfkdh	2023	2026-12-12
58	19	tariq	tariq@yahoo.com	kljsjbv nsda	2025	2026-08-12
50	17	ahamad	ahamad@yahoo.com	0	2020	2026-03-12
48	5	Mashi	oke@gmail.com	8347384	2021	2025-12-18
56	16	john	john@gmail.com	0	2024	2025-12-12
45	5	Mashi	oke@gmail.com	8347384	2023	2025-12-12
50	17	ahamad	ahamad@yahoo.com	0	2025	2025-12-12

Admin Dashboard Structure AdminDashboard.php

```
1  <!--
2      File:      admin_dashboard.html
3      Purpose:   Main administrative interface for Garage Management System
4      Features:
5          Three key metric graphs (income, bookings, services)
6          Recent bookings list
7          Dynamic form display area
8          Responsive layout with sidebar navigation
9      Student:   Muhammad Fahad (C00311349)
10     Date:      26/03/2025
11     Dependencies:
12         AdminDashboard.css (main layout)
13         sidebar.css (navigation styling)
14         Syncfusion EJ2 (graphing library)
15         Multiple PHP data files
16         JavaScript for dynamic behavior
17     -->
18     <!DOCTYPE html>
19     <html lang="en">
20     <head>
21         <meta charset="UTF-8" />
22         <meta name="viewport" content="width=device-width, initial-scale=1.0" />
23         <title>GMS|Admin Dashboard</title>
24         <link rel="stylesheet" href=".//Assets/css/AdminDashboard.css" />
25         <link rel="stylesheet" href=".//Assets/css/sidebar.css" />
26         <script src="https://kit.fontawesome.com/0cd0283581.js" crossorigin="anonymous"></script>
27         <script src="https://cdn.syncfusion.com/ej2/dist/ej2.min.js" type="text/javascript"></script>
28     </head>
29     <body>
30         <!-- main container with display grip to handle the layout of page -->
31         <main class="main_container">
32             <?php include 'header.php';
33             ?>
34             <?php include 'sidebar.php';
35             ?>
36             <!-- main dashboard content -->
37             <main class="admin_dashboard_main">
38                 <!-- dashboard section -->
39                 <section class="dashboard_content_Section" id="dashboard_content">
40                     <div class="dashboard_main_container">
41                         <!-- weekly booking income container -->
42                         <div class="top_dashboard">
43                             <!-- income -->
44                             <div class="graph_container gc1">
45                                 <div class="income_status">
46                                     <div class="income_status_sub_container">
47                                         <?php
48                                         // this file contain the total gross income
49                                         include './Assets/graph_data/incomeGraphData.php'
50                                         ?>
51                                         <p>Total gross income</p>
52                                     </div>
53                                     <div id="graph_container1"></div>
54                                 </div>
55                         </div>
```

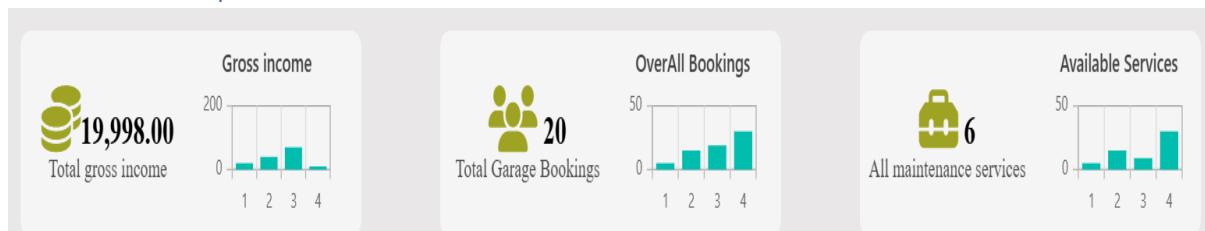
Continued....

```
1  <!-- bookings -->
2      <div class="graph_container gc2">
3          <div class="income_status">
4              <div class="income_status_sub_container">
5                  <?php
6                      // this file contain the total number of bookings
7                      include './Assets/graph_data/bookingGraphData.php'
8                  ?><p>Total Garage Bookings</p>
9                  </div>
10                 <div id="graph_container2"></div>
11             </div>
12         </div>
13         <!-- services -->
14         <div class="graph_container gc3">
15             <div class="income_status">
16                 <div class="income_status_sub_container">
17                     <?php
18                         // this file contain the total number of job types
19                         include './Assets/graph_data/jobTypeGraphData.php'
20                     ?><p> All maintenance services</p>
21                     </div>
22                     <div id="graph_container3"></div>
23                 </div>
24             </div>
25         </div>
26     <main class="recent_booking_main_conatiner">
27         <section class="recent_booking_section_container">
28             <?php include './bookingList.php'; ?>
29         </section>
30     </main>
31 </section>
32 <!-- suppiler form container here we will show form dynamically -->
33     <div id="dynamicDisplayForms"></div>
34
35     </main>
36 </main>
37     <script src="./Assets/js/side_bar.js"></script>
38     <script src="./Assets/js/admin_dashboard.js"></script>
39 </body>
40
41 </html>
```

Dashboard Header header.php

```
1 <!--
2     File:          header.php
3     Purpose:       display headbar on every page
4     Created By:    Muhammad Fahad (c00311349)
5     Group Members :
6             c00298483 Adam Dowling
7             C00295140 Taemour Basharat
8             C00311349 Muhammad Fahad
9             C00297032 Emoshoke Saliu
10            C00290944 Gleb Tutubalin
11            Date:        26/03/2025
12 -->
13 <!-- header -->
14     <header class="dashboard_header">
15         <!-- left header -->
16         <div class="left_header">
17             <h2 class="garage_name">AutoSphere</h2>
18             <p class="slogan">Comprehensive auto management ecosystem</p>
19         </div>
20         <!-- centre header -->
21         <div class="center_header">
22             <input type="search" placeholder="Search" />
23         </div>
24         <!-- right header -->
25         <div class="right_header">
26             
27             <h3 class="admin_name">Admin1</h3>
28         </div>
29     </header>
30 
```

Dashboard Graphs



Income Graph incomeGraphData.php

```
1  <!--
2   File:          income Graph Data
3   Purpose:       This file retrieves income data from the database to display
4   in the dashboard graph container
5   Member:        Muhammad Fahad (C00311349)
6   Dependencies: Requires DBconnection.php for database access
7   Security:     Uses prepared statements to prevent SQL injection
8   Error Handling: Includes try-catch block for database operations
9   Date:          26/03/2025
10  -->
11
12 <?php
13 include './DBconnection.php';
14
15 try {
16     /**
17      * Query to calculate total income of the garage
18      * Uses COUNT(*) to get the total records in the payment table
19      * Prepared statement is used for security against SQL injection
20     */
21     $payment = $con->prepare("SELECT SUM(Total_Price) AS total_payments FROM `Payment`");
22     $payment->execute();
23     $total_payment = $payment->fetch(PDO::FETCH_ASSOC);
24
25     if (isset($total_payment['total_payments'])) {
26         $total = $total_payment['total_payments'];
27     } else {
28         $total = 0 ;
29     }
30     // formated into 2 decimal places
31     $formated = number_format($total,2);
32
33     echo "<h2 class='income'>".<i class='fa-solid fa-coins currency'></i>" .
34     . $formated .
35     "</h2>";
36 } catch (PDOException $e) {
37     // shows error log in case of any error for debugging
38
39     error_log("Database Error: " . $e->getMessage());
40     echo "<h2>System error occurred. Please try again.</h2>";
41 }
```

Booking Graph bookingGraphData.php

```
1  <!--
2      File:          Booking Graph Data
3      Purpose:       This file retrieves booking data from the database to display in
4                      the dashboard graph container
5      Member:        Muhammad Fahad (C00311349)
6      Dependencies: Requires DBconnection.php for database access
7      Security:     Uses prepared statements to prevent SQL injection
8      Error Handling: Includes try-catch block for database operations
9      Date:          26/03/2025
10     -->
11     <?php
12     include './DBconnection.php';
13
14     try {
15         /**
16         * Query to calculate total number of bookings in the system
17         * Uses COUNT(*) to get the total records in the Booking table
18         * Prepared statement is used for security against SQL injection
19         */
20         $booking = $con->prepare("SELECT COUNT(*) AS total_bookings FROM `Booking`");
21         $booking->execute();
22         $total_booking = $booking->fetch(PDO::FETCH_ASSOC);
23
24         if (isset($total_booking['total_bookings'])) {
25             $total = $total_booking['total_bookings'];
26         } else {
27             $total = 0 ;
28         }
29         echo "<h2 class='income'>".<i class='fa-solid fa-users currency'> </i> "
30         . $total .
31         "</h2>";
32     } catch (PDOException $e) {
33         // shows error log in case of any error for debugging
34         error_log("Database Error: " . $e->getMessage());
35         echo "<h2>System error occurred. Please try again.</h2>";
36     }
}
```

Job type/services Graph jobTypeGraphData.php

```
1  <!--
2      File:          job type /services Graph Data
3      Purpose:       This file retrieves job type /services provided by garage,
4                      data from the database to display in the dashboard graph container
5      Member:        Muhammad Fahad (C00311349)
6      Dependencies: Requires DBconnection.php for database access
7      Security:     Uses prepared statements to prevent SQL injection
8      Error Handling: Includes try-catch block for database operations
9      Date:          26/03/2025
10     -->
11
12     <?php
13     include './DBconnection.php';
14
15     try {
16         /**
17         * Query to calculate total job types/ services of the garage
18         * Uses COUNT(*) to get the total records in the job type table
19         * Prepared statement is used for security against SQL injection
20         */
21         $job_types = $con->prepare("SELECT COUNT(*) AS total_job_types FROM `Job_Type`;");
22         $job_types->execute();
23         $total_job_types = $job_types->fetch(PDO::FETCH_ASSOC);
24
25         if (isset($total_job_types['total_job_types'])) {
26             $total = $total_job_types['total_job_types'];
27         } else {
28             $total = 0 ;
29         }
30         echo "<h2 class='income'>"."<i class='fa-solid fa-toolbox currency'> </i> "
31             . $total .
32             "</h2>";
33     } catch (PDOException $e) {
34         // shows error log in case of any error for debugging
35
36         error_log("Database Error: " . $e->getMessage());
37         echo "<h2>System error occurred. Please try again.</h2>";
38     }
}
```

Dashboard Booking List

List of Bookings

Booking ID	Customer ID	Customer Name	Email Address	Telephone	Car Model	Booking Date
40	10	taemour	testtesteing@gmail.c	851569662	2009	2028-12-11
41	10	taemour	testtesteing@gmail.c	851569662	1998	2028-11-17
43	5	Mashi	oke@gmail.com	8347384	2007	2028-09-09
39	5	Mashi	oke@gmail.com	8347384	2016	2027-07-09
57	18	umair	umair@hotmail.com	hffsdjfkds	2023	2026-12-12
58	19	tariq	tariq@yahoo.com	klsljbv nsda	2025	2026-08-12
50	17	ahamd	ahamd@yahoo.com	0	2020	2026-03-12
48	5	Mashi	oke@gmail.com	8347384	2021	2025-12-18
56	16	john	john@gmail.com	0	2024	2025-12-12
45	5	Mashi	oke@gmail.com	8347384	2023	2025-12-12
50	17	ahamd	ahamd@yahoo.com	0	2025	2025-12-12

bookingList.php

```

1  <!--
2      File:          booking_list.php
3      Purpose:       Displays a list of all bookings with customer details records in the database
4      Member:        Muhammad Fahad (C00311349)
5      Functionality:
6          Retrieves booking records joined with customer data
7          Displays in a responsive HTML table
8          Sorts by booking date (newest first)
9          Security:    Uses prepared statements to prevent SQL injection
10         Error Handling: Includes try-catch block for database operations
11         Database Impact: Reads from Booking and Customer tables
12         Date:          26/03/2025
13     -->
14 <?php
15 include 'DBconnection.php';
16 // SQL query to fetch booking data with customer information
17 $sql = "SELECT
18     b.Booking_ID,
19     b.Booking_Date,
20     b.Model,
21     b.Customer_ID,
22     c.Name ,
23     c.E_Mail ,
24     c.Telephone
25 FROM
26     Booking b
27 JOIN
28     Customer c ON b.Customer_ID = c.Customer_ID
29 ORDER BY
30     b.Booking_Date DESC;";
31     // Execute query and fetch results
32 $result = $con->query($sql);
33     $row = $result->fetchAll(PDO::FETCH_ASSOC);
34 ?>
```

Continued....

```
1 <div class="suppliers_list_container_heading">
2     <h2 class="heading">List of Bookings</h2>
3 </div>
4 <div class="supplier_table_conatainer">
5     <table class="table">
6         <tr class="table-column">
7             <th>Booking ID</th>
8             <th>Customer_ID</th>
9             <th>Customer Name</th>
10            <th>Email Address</th>
11            <th>Telephone</th>
12            <th>Car Model</th>
13            <th>Booking Date</th>
14        </tr>
15        <?php
16        if (!empty($row)) {
17            foreach ($row as $rs) {
18                echo "
19                <tr class='table_row'>
20                    <td> {$rs['Booking_ID']} </td>
21                    <td> {$rs['Customer_ID']} </td>
22                    <td> {$rs['Name']} </td>
23                    <td> {$rs['E_Mail']} </td>
24                    <td> {$rs['Telephone']} </td>
25                    <td> {$rs['Model']} </td>
26                    <td> {$rs['Booking_Date']} </td>
27                </tr> ";
28            }
29        } else {
30            echo "
31            <tr>
32                <td><h2>No Records Available Yet ComeBack later!</h2></td>
33            </tr>
34            ";
35        }
36    }
37    ?>
38    </table>
39 </div>
```

Fetch.php

This file contains the Dropdown which is Used in
AmendViewSupplierPage.php,
DeleteSupplierPage.php,
AddNewBooking.php,
AmendViewStockItem.php

```
1 <!--
2   File:          fetch.php
3   Purpose:       Provides JSON data for dynamic dropdown menus
4   Member:        Muhammad Fahad (C00311349)
5   Functionality:
6           Fetches active records from multiple tables:
7           Customers
8           Suppliers
9           Job Types
10          Stock Items
11          Returns data as JSON for AJAX consumption
12  Security:     Uses prepared statements to prevent SQL injection
13  Error Handling: Includes try-catch block for database operations
14  Database Impact: Reads from Booking and Customer tables
15  Date:          26/03/2025
16 -->
17 <?php
18 include 'DBconnection.php';
19 $response = [
20   'customer' => [],      // Will hold customer dropdown data
21   'supplier' => [],      // Will hold supplier dropdown data
22   'job_type' => [],      // Will hold job type dropdown data
23   'stock_item' => []     // Will hold stock item dropdown data
24 ];
25 // drop down for customer
26 try {
27   $Customer = $con->prepare("SELECT `Customer_ID`, `Name` FROM `Customer` WHERE `Delete_Flag` = '0';");
28   $Customer->execute();
29   $response['customer'] = $Customer->fetchAll(PDO::FETCH_ASSOC);
30 } catch (PDOException $e) {
31
32   error_log("Database Error while fetching dropdown". $e->getMessage());
33   echo json_encode([]); // Return empty array on error
34 }
```

Continued...

```
 1 //Drop Down For Supplier
 2 try {
 3     $Supplier = $con->prepare("SELECT `Supplier_ID`, `Name` FROM `Supplier` WHERE `Delete_Flag` = '0' ;");
 4     $Supplier->execute();
 5     $response['supplier'] = $Supplier->fetchAll(PDO::FETCH_ASSOC);
 6 } catch (PDOException $e) {
 7     error_log("Database Error while fetching dropdown".$e->getMessage());
 8     echo json_encode([]); // Return empty array on error
 9 }
10 // drop down for job type
11 try {
12     $jobtype = $con->prepare("SELECT `Job_Type_ID`, `Job_Type_Name` FROM `Job_Type` WHERE `Delete_Flag` = '0' ;");
13     $jobtype->execute();
14     $response['job_type'] = $jobtype->fetchAll(PDO::FETCH_ASSOC);
15 } catch (PDOException $e) {
16     error_log("Database Error while fetching dropdown".$e->getMessage());
17     echo json_encode([]);
18 }
19
20 // drop down for stock item
21 try {
22     $stockitem = $con->prepare("SELECT `Stock_ID`, `Description` FROM `Stock_Item` WHERE `Status` = '0';");
23     $stockitem->execute();
24     $response['stock_item'] = $stockitem->fetchAll(PDO::FETCH_ASSOC);
25 } catch (PDOException $e) {
26     error_log("Database Error while fetching dropdown".$e->getMessage());
27     echo json_encode([]);
28 }
29 // Set the content type to JSON
30 header('Content-Type: application/json');
31 echo json_encode($response);
32 ?>
```

Code

Adam Dowling

Add New Stock Item

AddNewStock.php:

```
1 <?php
2 /*
3 Name: Adam Dowling
4 Student ID: C00298483
5 Date: 20/02/25
6 For: Add screen
7 */
8 include './DBconnection.php'; // Include database connection
9
10 if ($_SERVER['REQUEST_METHOD'] === 'POST') { // Is it a post?
11     $Description = $_POST['Description']; // Gets descr from form
12     $Stock_Quantity = $_POST['Stock_Quantity'];
13     $Reorder_Quantity = $_POST['Reorder_Quantity'];
14     $Cost_Price = $_POST['Cost_Price'];
15     $Retail_Price = $_POST['Retail_Price'];
16     $Supplier_ID = $_POST['Supplier_ID'];
17
18     try {
19         // SQL to inset data into Stock_Item
20         $insert = $con->prepare("INSERT INTO `Stock_Item`(`Description`, `Stock_Quantity`, `Reorder_Quantity`, `Cost_Price`, `Retail_Price`, `Supplier_ID`)
21         VALUES (:Description, :Stock_Quantity, :Reorder_Quantity, :Cost_Price, :Retail_Price, :Supplier_ID)");
22
23         // Binding for SQL inj prevention
24         $insert->bindParam(':Description', $Description);
25         $insert->bindParam(':Stock_Quantity', $Stock_Quantity);
```

AddNewStock.php continued:

```
1  $insert->bindParam(':Reorder_Quantity', $Reorder_Quantity);
2  $insert->bindParam(':Cost_Price', $Cost_Price);
3  $insert->bindParam(':Retail_Price', $Retail_Price);
4  $insert->bindParam(':Supplier_ID', $Supplier_ID);
5
6  if ($insert->execute()) // Checks if data inserted fully
7  {
8      echo "<h2>Data is inserted into the database successfully.</h2>"; //Success
9  } else {
10     echo "Error occurred during inserting data into the database."; //Fail
11     error_log("Database Error: " . print_r($insert->errorInfo(), true)); //Error log
12 }
13
14 } catch (PDOException $e) //catch errors
15 {
16     echo("Database error: " . $e->getMessage()); //display error
17     error_log("Database error: " . $e->getMessage()); //Shows error log
18 }
19 }
20 ?>
```

AddNewStockForm.php:

```
1  <!--
2  Name: Adam Dowling
3  Student ID: C00298483
4  Date: 20/02/25
5  For: Add screen
6  -->
7  <!DOCTYPE html>
8  <html lang="en">
9
10 <head>
11   <meta charset="UTF-8">
12   <meta name="viewport" content="width=device-width, initial-scale=1.0">
13   <title>Add New Stock Item|GMS</title> <!--Add title-->
14   <link rel="stylesheet" href=".//Assets/css>ShowSuppliers.css"> <!--CSS link-->
15 </head>
16
17 <body>
18   <main class="main_form_container"> <!-- Main container for form -->
19     <div class="div">
20       <div class="form_title">
21         <h1>Add New Stock Item</h1> <!-- Form heading -->
22       </div>
23
24       <form action="AddNewStock.php" method="post" id="form" class="form"> <!--form for adding a new stock item -->
25         <!-- Description -->
```

AddNewStockForm.php continued:

```
1 <div class="inputMain">
2     <label for="Description">Description: </label> <!--Label for description-->
3     <input type="text" id="Description" placeholder="Enter a description of the new stock item" name="Description" required><!--text input for description-->
4 </div>
5
6     <!-- Stock Quantity -->
7     <div class="inputMain">
8         <label for="Stock_Quantity">Stock Quantity: </label>
9         <input type="text" id="Stock_Quantity" placeholder="Enter the quantity of stock" name="Stock_Quantity" required>
10 </div>
11
12     <!-- Reorder Quantity -->
13     <div class="inputMain">
14         <label for="Reorder_quantity">Reorder Quantity: </label>
15         <input type="text" id="Reorder_quantity" placeholder="Enter the reorder quantity for the stock item" name="Reorder_Quantity" required>
16 </div>
17
18     <!-- Cost Price -->
```

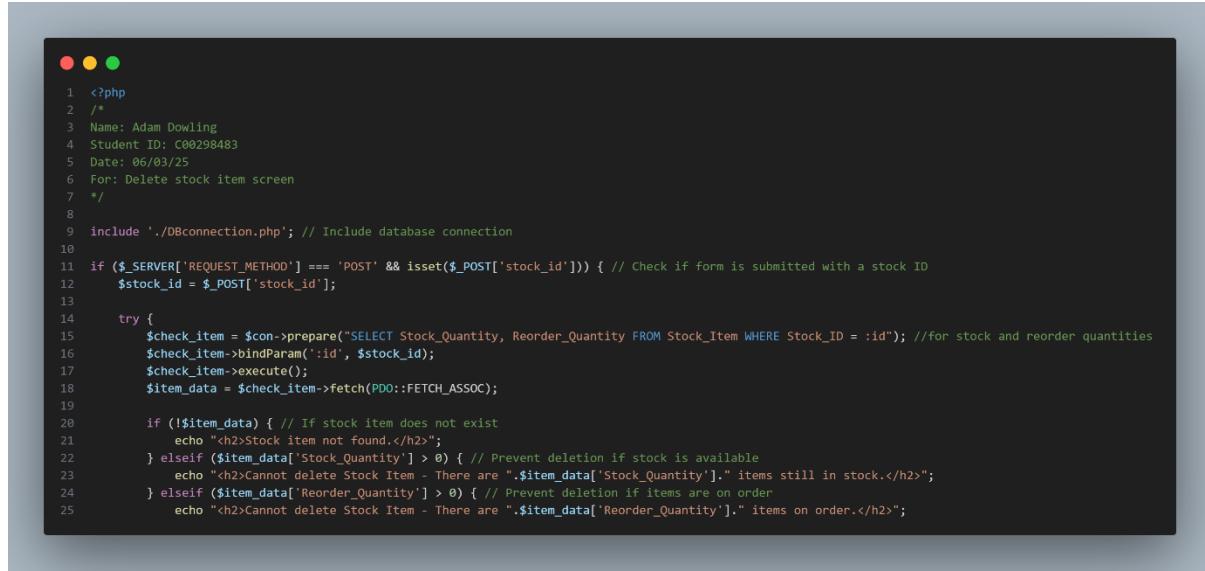
AddNewStockForm.php continued:

```
1 <label for="Retail_Price">Retail Price: </label>
2         <input type="text" id="Retail_Price" placeholder="Enter the retail price of the new stock item" name="Retail_Price" required>
3     </div>
4
5     <!-- Supplier Selection Dropdown -->
6     <div class="inputMain">
7         <label for="Supplier_ID">Supplier: </label> <!-- Label for dropdown -->
8         <select id="Supplier_ID" name="Supplier_ID" class="supplier-list" required> <!--Dropdown for supplier-->
9             <option value="">-- Select Supplier --</option> <!--Placeholder -->
10            <?php
11            include 'DBconnection.php'; //database connection
12
13            $suppliers = $con->query("SELECT Supplier_ID, Name FROM Supplier ORDER BY Name"); // SQL to get id and name from database
14
15            // Loop for each supplier to be added to dropdown
16            while ($supplier = $suppliers->fetch(PDO::FETCH_ASSOC)) {
17                echo '<option value="'.$supplier['Supplier_ID'].'>'.$supplier['Name'].'</option>';
18            }
19        ?>
20    </select>
21 </div>
22
23     <!-- Form Buttons) -->
24     <div class="form_buttons">
25         <input type="reset" value="Cancel" class="formButton"> <!-- Button to reset-->
```

AddNewStockForm.php continued:

```
1      <input type="submit" value="Add New Stock Item" class="formButton"> <!-- Button to submit-->
2      </div>
3
4      <!-- Placeholder-->
5      <p id="message"></p>
6      </form>
7      </div>
8  </main>
9
10     <!-- External JavaScript file for form validation -->
11     <script src=".Assets/js/NewStockValidation.js"></script>
12 </body>
13 </html>
```

Delete Screen deleteStock.php:



```
1 <?php
2 /*
3  Name: Adam Dowling
4  Student ID: C00298483
5  Date: 06/03/25
6  For: Delete stock item screen
7 */
8
9 include './DBconnection.php'; // Include database connection
10
11 if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['stock_id'])) { // Check if form is submitted with a stock ID
12     $stock_id = $_POST['stock_id'];
13
14     try {
15         $check_item = $con->prepare("SELECT Stock_Quantity, Reorder_Quantity FROM Stock_Item WHERE Stock_ID = :id"); //for stock and reorder quantities
16         $check_item->bindParam(':id', $stock_id);
17         $check_item->execute();
18         $item_data = $check_item->fetch(PDO::FETCH_ASSOC);
19
20         if (!$item_data) { // If stock item does not exist
21             echo "<h2>Stock item not found.</h2>";
22         } elseif ($item_data['Stock_Quantity'] > 0) { // Prevent deletion if stock is available
23             echo "<h2>Cannot delete Stock Item - There are ".$item_data['Stock_Quantity']."' items still in stock.</h2>";
24         } elseif ($item_data['Reorder_Quantity'] > 0) { // Prevent deletion if items are on order
25             echo "<h2>Cannot delete Stock Item - There are ".$item_data['Reorder_Quantity']."' items on order.</h2>";
```

deleteStock.php continued:

```
1      } else {
2          // Flag item for deletion by updating its status
3          $update = $con->prepare("UPDATE Stock_Item SET Status = 1 WHERE Stock_ID = :id");
4          $update->bindParam(':id', $stock_id);
5
6          if ($update->execute()) { // Check if the update was successful
7              echo "<h2>Stock Item has been successfully flagged for deletion.</h2>";
8          } else { // If update fails
9              echo "<h2>Error processing deletion request.</h2>";
10             error_log("Database Error: " . print_r($update->errorInfo(), true));
11         }
12     }
13 } catch (PDOException $e) { // Catch errors
14     error_log("Database Error: " . $e->getMessage()); //shows error message in log
15     echo "<h2>System error occurred. Please try again.</h2>"; //Message when an error occurs
16 }
17 } else { // If there isn't a stock id
18     echo "<h2>Please select a valid Stock Item.</h2>";
19 }
20 ?>
```

DeleteStockPage.php:

```
● ● ●
```

```
1 <!--
2 Name: Adam Dowling
3 Student ID: C00298483
4 Date: 07/03/25
5 For: Delete stock item screen
6 -->
7 <!DOCTYPE html>
8 <html lang="en">
9
10 <head>
11   <meta charset="UTF-8">
12   <meta name="viewport" content="width=device-width, initial-scale=1.0">
13   <title>Delete Supplier</title> <!--title-->
14
15   <!-- Link CSS -->
16   <link rel="stylesheet" href="./Assets/css/AdminDashboard.css">
17   <link rel="stylesheet" href="./Assets/css>ShowSuppliers.css">
18 </head>
19
20 <style>
21   /* Style for dropdown */
22   select.drop_down_list option {
23     color: black !important;
24     font-size: 16px !important;
25 }
```

DeleteStockPage.php continued:

```
1  </style>
2
3 <body>
4   <main class="delete_supplier_main_container"> <!-- Main container for delete supplier -->
5
6   <section class="suppliers_list_section_container"> <!-- Section upplier list -->
7
8   <!-- Form for deleting a stock item -->
9   <form action="./deleteStock.php" method="post" id="form">
10
11    <section class="header_section">
12      <header class="delete_stockheader">
13        <ul>
14          <li>
15            <label for="Stock_Item">Stock Items List:</label> <!-- Label for mydropdown -->
16            <select
17              class="drop_down_list"
18              name="stock_id"
19              id="Stock_Item"
20              data_file_link="fetch.php"
21              data_value="Stock_ID"
22              data_text="Description"> <!--fetches data will fetch.php-->
23              <option value="">---Select Supplier---</option> <!--placeholder -->
24            </select>
25          </li>

```

DeleteStockPage.php continued:

```
1      <li><input type="submit" value="Delete Stock Item"></li> <!-- Submit button -->
2      </ul>
3
4      <!-- Placeholder-->
5      <p id="message"></p>
6      </header>
7      </section>
8
9      <?php
10     include './stockList.php'; // stock item list
11     ?>
12     </form>
13     </section>
14     </main>
15 </body>
16 </html>
```

stockList.php:

```
1  <!--
2   File:          stockList.php
3   Description:  This file display the stock items from the database
4   - Retrieves non-deleted stock items from database
5   - Displays them in a responsive HTML table
6   - Handles empty result set gracefully
7   Member:        Adam Dowling (c00298483)
8   Dependencies: Requires DBconnection.php for database access
9   -->
10  <?php
11 include 'DBconnection.php';
12 $sql = "SELECT `Stock_ID`, `Description`, `Stock_Quantity`, `Reorder_Quantity`, `Retail_Price`, `Supplier_ID` FROM `Stock_Item` WHERE `Status` = '0' ; ";
13 // Execute query and fetch all results as associative array
14
15 $result = $con->query($sql);
16 $row = $result->fetchAll(PDO::FETCH_ASSOC);
17 ?>
18
19 <div class="stock_list_container_heading">
20   <h2 class="heading">List of Stock Items</h2>
21 </div>
22
23 <div class="stock_table_container">
24   <table class="table">
```

stockList.php continued:

```
1      <tr class="table-column">
2          <th>Stock ID</th>
3          <th>Stock Name</th>
4          <th>Quantity</th>
5          <th>Re-Order Quantity</th>
6          <th>Price</th>
7          <th>Supplier_ID</th>
8      </tr>
9      <?php
10     // Check if any suppliers were found
11
12     if (!empty($row)) {
13
14         foreach ($row as $rs) {
15
16             echo "
17                 <tr class='table_row'>
18                     <td> {$rs['Stock_ID']} </td>
19                     <td> {$rs['Description']} </td>
20                     <td> {$rs['Stock_Quantity']} </td>
21                     <td> {$rs['Reorder_Quantity']} </td>
22                     <td> {$rs['Retail_Price']} </td>
23                     <td> {$rs['Supplier_ID']} </td>
24
25                 </tr> ";
26         }
27     } else {
28
29         echo "
30             <tr>
31                 <td><h2>No Records Available Yet ComeBack later!</h2></td>
32             </tr>
33         ";
34     }
35 }
36
37     ?>
38     </table>
39 </div>
```

Amend / View Screen

AmendViewStock.php:

```
1  <?php
2  /*
3  Adam Dowling
4  C00298483
5  25/03/25
6  Screen: Amend / View Stock Item
7  */
8 // Start session and connect to DB
9 session_start();
10 include 'DBconnection.php';
11
12 // Handle form submission
13 if ($_SERVER['REQUEST_METHOD'] === 'POST') {
14     try {
15         // Prepare update query
16         $update = $con->prepare("UPDATE Stock_Item SET
17             Description = :Description,
18             Stock_Quantity = :Stock_Quantity,
19             Reorder_Level = :Reorder_Level,
20             Reorder_Quantity = :Reorder_Quantity,
21             Cost_Price = :Cost_Price,
22             Retail_Price = :Retail_Price,
23             Supplier_ID = :Supplier_ID
24             WHERE Stock_ID = :Stock_ID");
25     }
```

AmendViewStock.php continued:

```
1      // Execute with form data
2      $update->execute([
3          ':Stock_ID' => $_POST['Stock_ID'],
4          ':Description' => $_POST['Description'],
5          ':Stock_Quantity' => $_POST['Stock_Quantity'],
6          ':Reorder_Level' => $_POST['Reorder_Level'],
7          ':Reorder_Quantity' => $_POST['Reorder_Quantity'],
8          ':Cost_Price' => $_POST['Cost_Price'],
9          ':Retail_Price' => $_POST['Retail_Price'],
10         ':Supplier_ID' => $_POST['Supplier_ID']
11     ]);
12
13     // Set success/error message
14     if ($update->rowCount() > 0) {
15         $_SESSION['message'] = "Stock Item updated successfully!";
16         $_SESSION['message_type'] = "success";
17     } else {
18         $_SESSION['message'] = "No changes made or item not found";
19         $_SESSION['message_type'] = "error";
20     }
21 } catch (PDOException $e) {
22     // Log errors
23     error_log("Database Error: ".$e->getMessage());
24     $_SESSION['message'] = "Error updating stock item";
25     $_SESSION['message_type'] = "error";
26 }
27
28 // Redirect back
29 header('Location: AmendViewStockPage.php');
30 exit();
31 }
32 ?>
```

AmendViewStockPage.php:

```
1  <!DOCTYPE html>
2  <!--
3  Adam Dowling
4  C00298483
5  26/03/25
6  Amend View Stock Item Screen
7  -->
8  <html lang="en">
9  <head>
10     <!-- Basic page setup -->
11     <meta charset="UTF-8">
12     <meta name="viewport" content="width=device-width, initial-scale=1.0">
13     <title>Amend / View Stock Item</title>
14
15     <!-- CSS styling -->
16     <link rel="stylesheet" href=".Assets/css/AdminDashboard.css">
17     <link rel="stylesheet" href=".Assets/css>ShowSuppliers.css">
18     <style>
19         /* Dropdown styling */
20         select.drop_down_list option {
21             color: black !important;
22             font-size: 16px !important;
23         }
24         /* Success/error message styling */
25         .success-message { color: green; padding: 10px; margin: 10px 0; }
26         .error-message { color: red; padding: 10px; margin: 10px 0; }
27     </style>
28 </head>
29 <body>
30     <!-- Main container -->
31     <main class="amend_stock_main_container">
32         <section class="amend_stock_section_container">
33             <!-- Form for editing stock items -->
34             <form action="AmendViewStock.php" method="post" id="stockForm" class="form">
35
36                 <!-- Stock item dropdown -->
37                 <label for="Stock_Item">Stock Item List:</label>
38                 <select class="drop_down_list" id="Stock_Item" name="Stock_ID" required>
39                     <option value="">---Select Stock Item---</option>
40                     <?php
41                         // Populate dropdown with active stock items
42                         include 'DBconnection.php';
```

AmendViewStockPage.php continued:

```
1      // Populate dropdown with active stock items
2      include 'DBconnection.php';
3      $stmt = $con->query("SELECT Stock_ID, Description FROM Stock_Item WHERE Status = 0");
4      while ($row = $stmt->fetch(PDO::FETCH_ASSOC)) {
5          echo '<option value="'.$row['Stock_ID'].'">'.$row['Description']. ' (ID: '.$row['Stock_ID'].')</option>';
6      }
7  ?>
8  </select>
9
10     <!-- Form fields --&gt;
11     &lt;div class="inputMain"&gt;
12         &lt;label for="id"&gt;Stock ID&lt;/label&gt;
13         &lt;input type="text" id="id" name="Stock_ID" readonly&gt;
14     &lt;/div&gt;
15
16     <!-- Other input fields (Description, Quantities, Prices) --&gt;
17     &lt;div class="inputMain"&gt;
18         &lt;label for="Description"&gt;Description&lt;/label&gt;
19         &lt;input type="text" id="Description" name="Description" required&gt;
20     &lt;/div&gt;
21
22     &lt;!-- ... (similar structure for other fields) ... --&gt;
23
24     <!-- Form buttons --&gt;
25     &lt;div class="form_buttons"&gt;
26         &lt;input type="reset" value="Cancel" class="formButton"&gt;
27         &lt;input type="submit" value="Update" class="formButton"&gt;
28     &lt;/div&gt;
29 &lt;/form&gt;
30
31     &lt;!-- Include stock list display --&gt;
32     &lt;?php include 'stockList.php'; ?&gt;
33 &lt;/section&gt;</pre>
```

AmendViewStockPage.php continued:

```
1  <!-- JavaScript functionality -->
2  <script>
3      // Load stock item details when selected
4      document.getElementById('Stock_Item').addEventListener('change', function() {
5          const stockId = this.value;
6          if (!stockId) return;
7
8          // Fetch item details and populate form
9          fetch('getStockItem.php?id=' + stockId)
10             .then(response => response.json())
11             .then(data => {
12                 document.getElementById('id').value = data.Stock_ID;
13                 document.getElementById('Description').value = data.Description;
14                 // ... (set other field values) ...
15             })
16             .catch(error => console.error('Error:', error));
17     });
18
19
20     // Confirm before submitting
21     document.getElementById('stockForm').addEventListener('submit', function(e) {
22         if (!confirm('Are you sure you want to update this stock item?')) {
23             e.preventDefault();
24         }
25     });
26     </script>
27 </body>
28 </html>
```

getStockItem.php:

```
1 <?php
2 /*
3 Adam Dowling
4 C00298483
5 26/03/25
6 Amend View Stock Item Screen
7 */
8 // Connect to database
9 include 'DBconnection.php';
10
11 // Set response as JSON
12 header('Content-Type: application/json');
13
14 // Check if ID parameter exists
15 if (!isset($_GET['id'])) {
16     echo json_encode([]); // Return empty if no ID
17     exit;
18 }
19
20 // Get stock item by ID
21 $stockId = $_GET['id'];
22 $stmt = $con->prepare("SELECT * FROM Stock_Item WHERE Stock_ID = ?");
23 $stmt->execute([$stockId]);
24 $item = $stmt->fetch(PDO::FETCH_ASSOC);
25
26 // Return item data as JSON
27 echo json_encode($item);
28 ?>
```

Project

AI Disclosure Documents for Adam Dowling

Student Name: _____ Adam Dowling _____

Student ID: _____ C00298483 _____

Course Name _____ Software Development _____

Project Title _____ Garage Management System _____

Code Unit _____ Add a new Stock Item _____

Date Submitted: _____ 27/03/25 _____

AI Assistance Disclosure

(Please expand where necessary)

AI Technologies Used

Indicate whether you used AI technologies in completing this task:

Yes, I used AI tools. (List all AI technologies used, e.g., ChatGPT, GitHub Copilot, Codeium, Google Bard, etc.)

- Deepseek

Prompts Used

If AI was used, describe the prompts or queries you provided to the AI tools:

- I copy and pasted my code into deepseek to try and see what was giving my code an error and looked for recommendations on how to fix it. I was recommended to bindparams for SQL injection and it provided me with a code snippet on how to implement an error log

Nature of Assistance

Indicate how AI technologies contributed to your work by selecting and elaborating on the relevant categories below:

- **Code Suggestions & Auto-Completion:** (e.g., AI suggested code snippets, function implementations, syntax corrections, etc.)
 - Deepseek provided me with bindparam and error log code suggestions
- **Debugging & Error Fixing:** (e.g., AI identified and corrected syntax/logical errors, suggested refactoring, etc.)
 - It helped fix the error I was having in my code by implementing this
- **Concept Explanation:** (e.g., AI provided explanations of algorithms, data structures, programming concepts, etc.)
 - One I received the output, I asked deepseek to explain steps on why this works so that I have an understanding of the code I was implementing

Use of AI-Generated Output

Describe how you used the AI-generated output in your work:

- I copy and pasted it from deepseek to my code once I read over and understood what the output was doing

Verification of AI Output

Explain the steps you took to verify the accuracy and correctness of the AI-generated output:

- I copy and pasted it into a copy of my add screen files to test it, when I realized it works, I pasted it into my main code.

AI-Generated Content Modifications

Did you modify or refine any AI-generated content before submission? (Check one and provide details.)

No, I used AI-generated content as is. (Explain why you did not modify it.)

- I used it as is because the code was now inserting into my database

Acknowledgment & Integrity Statement

By submitting this document, I acknowledge that I have fully disclosed my use of AI technologies in completing this lab sheet.

Project

AI Disclosure Document

Student Name: _____ Adam Dowling _____
Student ID: _____ C00298483 _____
Course Name _____ Software Development _____
Project Title _____ Garage Management System _____
Code Unit _____ Delete Stock Item _____
Date Submitted: _____ 27/03/25 _____

AI Assistance Disclosure

(Please expand where necessary)

AI Technologies Used

Indicate whether you used AI technologies in completing this task:

Yes, I used AI tools. (List all AI technologies used, e.g., ChatGPT, GitHub Copilot, Codeium, Google Bard, etc.)

- Deepseek

Prompts Used

If AI was used, describe the prompts or queries you provided to the AI tools:

- There were many fatal errors in my code, I fixed the bulk of them myself but I couldn't figure out the other issue so I copy and pasted it into deepseek and asked for an explanation of what is wrong with the code
- _____
- _____

Nature of Assistance

Indicate how AI technologies contributed to your work by selecting and elaborating on the relevant categories below:

- **Code Suggestions & Auto-Completion:** (e.g., AI suggested code snippets, function implementations, syntax corrections, etc.)
 - It suggested I use an associative array to validate stock quantity and check reorder quantity instead of my previous method
- **Debugging & Error Fixing:** (e.g., AI identified and corrected syntax/logical errors, suggested refactoring, etc.)
 - I got it to fix one of my errors which ended up just being a variable spelled wrong

- **Concept Explanation:** (e.g., AI provided explanations of algorithms, data structures, programming concepts, etc.)
 - I got it to tell me more about associative arrays and how their keys work to get a better understanding of my code
- **Optimization & Efficiency Improvements:** (e.g., AI suggested more efficient algorithms, reduced complexity, optimized loops, etc.)
 - The associative array is used for efficiency improvements in my code

Use of AI-Generated Output

Describe how you used the AI-generated output in your work:

- I copy and pasted it into my code, and also tried it out myself in one of my webdev labs to see how I can implement associative arrays on different programs

Verification of AI Output

Explain the steps you took to verify the accuracy and correctness of the AI-generated output:

- Once I copy and pasted it and ran the code, I realised that the status in the stock item table was changing as I wanted it, so I knew it worked

AI-Generated Content Modifications

Did you modify or refine any AI-generated content before submission? (Check one and provide details.)

No, I used AI-generated content as is. (Explain why you did not modify it.)

- I did not modify it as it was making my code run and update the database successfully and the validation was all working fine.

Acknowledgment & Integrity Statement

By submitting this document, I acknowledge that I have fully disclosed my use of AI technologies in completing this lab sheet.

Project

AI Disclosure Document

Student Name: _____ Adam Dowling _____
Student ID: _____ C00298483 _____
Course Name _____ Software Development _____
Project Title _____ Garage Management System _____
Code Unit _____ Amend / View a Stock Item _____
Date Submitted: _____ 27/03/25 _____

AI Assistance Disclosure

(Please expand where necessary)

AI Technologies Used

Indicate whether you used AI technologies in completing this task:

Yes, I used AI tools. (List all AI technologies used, e.g., ChatGPT, GitHub Copilot, Codeium, Google Bard, etc.)

- Deepseek

Prompts Used

If AI was used, describe the prompts or queries you provided to the AI tools:

- I wanted to include some AJAX to my code to fetch data and wanted to learn a bit about it so I pasted my code into it and asked how could I change it to add some AJAX into it

Nature of Assistance

Indicate how AI technologies contributed to your work by selecting and elaborating on the relevant categories below:

- **Code Suggestions & Auto-Completion:** (e.g., AI suggested code snippets, function implementations, syntax corrections, etc.)
- I asked deepseek to give me another version of my code but with some AJAX functionalities added to it which can be seen in the code along with deepseek added a new file called getStockItem.php which handles the AJAX aswell
- **Concept Explanation:** (e.g., AI provided explanations of algorithms, data structures, programming concepts, etc.)
 - All of the code it outputted I asked for a breakdown and explanation of each part to understand AJAX better

Use of AI-Generated Output

Describe how you used the AI-generated output in your work:

- I copy and pasted it as is into my code once I had read over the output and researched about AJAX

Verification of AI Output

Explain the steps you took to verify the accuracy and correctness of the AI-generated output:

- After researching about AJAX based on the output from deepseek, some notes from stack overflow, and an introduction on W3Schools, I added it to my code and it worked.

AI-Generated Content Modifications

Did you modify or refine any AI-generated content before submission? (Check one and provide details.)

Yes, I made modifications. (Explain what was changed and why.)

- It changed some parts of my code I didn't want to be changed so had to modify the output slightly so that all variable / post names matched across my files, but as for the AJAX part, I used the output as is.

No, I used AI-generated content as is. (Explain why you did not modify it.)

- _____

Acknowledgment & Integrity Statement

By submitting this document, I acknowledge that I have fully disclosed my use of AI technologies in completing this lab sheet.

Emoshoke Saliu

Add Job Type

AddJobType.php:

```
1  <!--
2  Emma Saliu
3  C00297032
4  GARAGE PROJECT
5  Add job type
6  -->
7
8  <?php
9  include 'DBconnection.php';
10
11 // Initialize variables for job type details
12 $jobTypeName = '';
13 $description = '';
14 $price = '';
15 $leadMechanic = ''; // Changed variable name
16
17 // Handle form submission
18 if ($_SERVER['REQUEST_METHOD'] === 'POST') {
19     // Capture form data
20     $jobTypeName = $_POST['Job_Type_Name'];
21     $description = $_POST['Description'];
22     $price = $_POST['Price'];
23     $leadMechanic = $_POST['Lead_Mechanic_ID']; // We will insert the mechanic's name, not ID
24
25     // Check if lead mechanic is selected
26     if (empty($leadMechanic)) {
27         echo "<h2>Error: You must select a Lead Mechanic.</h2>";
28     } else {
29         try {
30             // Insert the new job type into the database (with mechanic name instead of ID)
31             $insert = $con->prepare("INSERT INTO `Job_Type` (`Job_Type_Name`, `Description`, `Price`, `Lead_Mechanic`)
32                                     VALUES (:Job_Type_Name, :Description, :Price, :Lead_Mechanic)");
33
34             // Bind parameters
35             $insert->bindParam(':Job_Type_Name', $jobTypeName, PDO::PARAM_STR);
36             $insert->bindParam(':Description', $description, PDO::PARAM_STR);
37             $insert->bindParam(':Price', $price, PDO::PARAM_INT); // Ensure price is an integer
38             $insert->bindParam(':Lead_Mechanic', $leadMechanic, PDO::PARAM_STR); // Bind mechanic's name
39
40             // Execute the query
41             if ($insert->execute()) {
42                 echo "<h2>Job Type added successfully!</h2>";
43             } else {
44                 echo "<h2>Error occurred while adding the Job Type.</h2>";
45             }
46         } catch (PDOException $e) {
47             echo "<h2>Database error: " . htmlspecialchars($e->getMessage()) . "</h2>";
48         }
49     }
50 }
51 ?>
```

addJobTypeForm.php;

```
1  <!--
2  Emma Saliu
3  C00297032
4  GARAGE PROJECT
5  Add job type
6  -->
7
8  <?php
9  // Include the database connection
10 include 'DBconnection.php';
11 ?>
12
13 <!DOCTYPE html>
14 <html lang="en">
15 <head>
16     <meta charset="UTF-8">
17     <meta name="viewport" content="width=device-width, initial-scale=1.0">
18     <title>Add Job Type | GMS</title>
19     <link rel="stylesheet" href="./Assets/css>ShowSuppliers.css">
20 </head>
21     <style>
22         /* Add this to your CSS file */
23         select.drop_down_list option {
24             color: black !important;
25             font-size: 16px !important;
26         }
27     </style>
28 <body>
29     <main class="main_form_container">
30         <div class="form_wrapper">
31             <div class="form_title">
32                 <h1>Add Job Type</h1>
33             </div>
34             <form action="addJobType.php" method="post" id="form" class="form">
35                 <p>
36                     <label for="Lead_Mechanic_ID">Lead Mechanic:</label>
37                     <select name="Lead_Mechanic_ID" id="Lead_Mechanic_ID" required>
38                         <option value="" disabled selected>Select Lead Mechanic</option>
39                         <?php
40                             // Fetch mechanics from the database
41                             try {
42                                 // Corrected query
43                                 $sql = "SELECT `Mechanic_ID`, `Mechanic_Name` FROM Mechanics WHERE `Deleted_Flag` = 0";
44                                 $stmt = $con->prepare($sql);
45                                 $stmt->execute();
46
47                                 // Fetch all mechanics
48                                 $mechanics = $stmt->fetchAll(PDO::FETCH_ASSOC);
49             
```

Cont.

```
1      if ($mechanics) {
2          // Output each mechanic as an option in the select dropdown
3          foreach ($mechanics as $row) {
4              echo '<option value="' . htmlspecialchars($row['Mechanic_ID']) . '">' . htmlspecialchars($row['Mechanic_Name']) . '</option>';
5          }
6      } else {
7          echo "<option value='>No active mechanics available</option>";
8      }
9  } catch (PDOException $e) {
10      // Handle error and display message if the query fails
11      echo "<option value='>Error fetching mechanics: " . htmlspecialchars($e->getMessage()) . "</option>";
12  }
13  ?>
14  </select>
15  </p>
16  <div class="inputMain">
17      <p>
18          <label for="Job_Type_Name">Job Type Name:</label>
19          <input type="text" name="Job_Type_Name" id="Job_Type_Name" required/>
20      </p>
21  </div>
22  <div class="inputMain">
23      <p>
24          <label for="Description">Description:</label>
25          <input type="text" name="Description" id="Description" required/>
26      </p>
27  </div>
28  <div class="inputMain">
29      <p>
30          <label for="Price">Base Price:</label>
31          <input type="text" name="Price" id="Price" required pattern="\d+(\.\d{2})?" title="Please enter a valid price (e.g., 100 or 100.00)"/>
32      </p>
33  </div>
34  <div class="form_button">
35      <input type="reset" value="Cancel" class="formButton">
36      <input type="submit" value="Add Job Type" class="formButton">
37  </div>
38  <p id="message"></p>
39  </form>
40  </div>
41  </main>
42 </body>
43 </html>
44
```

Delete Job Type

DeleteJobType.php;

```
1  <!--
2  Emma Saliu
3  C00297032
4  GARAGE PROJECT
5  Delete job type
6  -->
7  <?php
8  include 'DBconnection.php';
9
10 // If you have your job type functions in a separate file, include them here:
11 // include 'jobTypeFunctions.php';
12
13 // For this example, we define the two needed functions inline:
14 function check_upcoming_jobs($con, $job_type_id) {
15     $sql = "SELECT 1 FROM Booking
16             WHERE job_type_id = :job_type_id
17                 AND Booking_Date > NOW()
18                     AND Status = 1";
19     $stmt = $con->prepare($sql);
20     $stmt->bindParam(':job_type_id', $job_type_id, PDO::PARAM_INT);
21     $stmt->execute();
22     return $stmt->rowCount() > 0;
23 }
24
25 function mark_job_type_as_deleted($con, $job_type_id) {
26     $sql = "UPDATE Job_Type SET Delete_Flag = 1 WHERE Job_Type_ID = :job_type_id";
27     $stmt = $con->prepare($sql);
28     $stmt->bindParam(':job_type_id', $job_type_id, PDO::PARAM_INT);
29     return $stmt->execute();
30 }
31
32 // Initialize messages and flag
33 $error_message = $success_message = $message = "";
34 $confirm_deletion = false;
35 $job_type_id = "";
36
37 // Process the initial delete request
38 if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['delete']) && isset($_POST['Job_Type_ID'])) {
39     $job_type_id = $_POST['Job_Type_ID'];
40     if (empty($job_type_id)) {
41         $error_message = "Please select a Job Type.";
42     } else {
43         if (check_upcoming_jobs($con, $job_type_id)) {
44             $error_message = "This Job Type cannot be deleted because it is part of an upcoming booking.";
45         } else {
46             $confirm_deletion = true;
47         }
48     }
49 }
```

Cont.

```
● ● ●
1 // Process the confirmation step
2 if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['confirm']) && isset($_POST['Job_Type_ID'])) {
3     $job_type_id = $_POST['Job_Type_ID'];
4     if ($_POST['confirm'] === 'Y') {
5         if (mark_job_type_as_deleted($con, $job_type_id)) {
6             $success_message = "Job Type has been successfully deleted.";
7         } else {
8             $error_message = "Error occurred while deleting the Job Type.";
9         }
10    } else {
11        $message = "Deletion cancelled.";
12    }
13 }
14
15 // If any messages exist, redirect back to the page with messages in the query string.
16 if (!empty($error_message) || !empty($success_message) || !empty($message)) {
17     $params = http_build_query([
18         'error_message' => $error_message,
19         'success_message' => $success_message,
20         'message' => $message
21     ]);
22     header("Location: DeleteJobTypePage.php?$params");
23     exit();
24 }
25 ?>
26 <!DOCTYPE html>
27 <html lang="en">
28 <head>
29     <meta charset="UTF-8">
30     <meta name="viewport" content="width=device-width, initial-scale=1.0">
31     <title>Confirm Delete Job Type</title>
32     <link rel="stylesheet" href="./Assets/css/AdminDashboard.css">
33     <link rel="stylesheet" href="./Assets/css>ShowSuppliers.css">
34 </head>
35 <style>
36     select.drop_down_list option {
37         color: black !important;
38         font-size: 16px !important;
39     }
40 </style>
41 <body>
42     <main class="delete_supplier_main_conatainer">
43         <section class="suppliers_list_section_container">
44             <?php
45                 // If deletion needs confirmation, display the confirmation form here.
46                 if ($confirm_deletion) {
47                     echo "<h2>Are you sure you want to delete this Job Type?</h2>";
48                     echo "<form action='DeleteJobType.php' method='post'>
49                         <input type='hidden' name='Job_Type_ID' value='$job_type_id'>
50                         <label>
51                             <input type='radio' name='confirm' value='Y' required> Yes
52                         </label>
53                         <label>
54                             <input type='radio' name='confirm' value='N' required> No
55                         </label>
56                         <input type='submit' value='Confirm'>
57                     </form>";
58                 }
59             ?>
60         </section>
61     </main>
62 </body>
63 </html>
```

DeleteJobTypeForm.php;

```
1  <!--
2  Emma Saliu
3  C00297032
4  GARAGE PROJECT
5  Delete job type
6  -->
7  <?php
8  include 'DBconnection.php';
9  ?>
10 <!DOCTYPE html>
11 <html lang="en">
12 <head>
13   <meta charset="UTF-8">
14   <meta name="viewport" content="width=device-width, initial-scale=1.0">
15   <title>Delete Job Type</title>
16   <link rel="stylesheet" href=".//Assets/css/Admindashboard.css">
17   <link rel="stylesheet" href=".//Assets/css>ShowSuppliers.css">
18 </head>
19 <style>
20   select.drop_down_list option {
21     color: black !important;
22     font-size: 16px !important;
23   }
24 </style>
25 <body>
26   <main class="delete_supplier_main_container">
27     <section class="suppliers_list_section_container">
28       <!-- Header & Form for Deleting Job Type -->
29       <form action=".//DeleteJobType.php" method="post" id="form">
30         <section class="header_section">
31           <header class="delete_supplierheader">
32             <ul>
33               <li>
34                 <label for="Job_Type_ID">Job Type List:</label>
35                 <select class="drop_down_list" name="Job_Type_ID" id="Job_Type_ID" data_file_link="fetchJobTypes.php" data_value="Job_Type_ID" data_text="Job_Type_Name">
36                   <option value="">---Select Job Type---</option>
37                   <?php
38                     // Populate the dropdown with active job types
39                     $sql = "SELECT Job_Type_ID, Job_Type_Name FROM Job_Type WHERE Delete_Flag = '0'";
40                     $result = $con->query($sql);
41                     if ($result) {
42                       while ($row = $result->fetch(PDO::FETCH_ASSOC)) {
43                         echo "<option value='{$row['Job_Type_ID']}'>{$row['Job_Type_Name']}
```

Cont.



A screenshot of a Mac OS X desktop environment showing a code editor window. The window has a dark theme and features a tab bar at the top with three colored dots (red, yellow, green). The main area of the window displays a block of PHP code. The code includes HTML structures like forms and sections, along with PHP logic for handling messages via GET requests. Lines are numbered from 1 to 27 on the left side of the code.

```
1  ?>
2      </select>
3      </li>
4      <li>
5          <input type="submit" name="delete" value="Delete Job Type">
6      </li>
7  </ul>
8  <!-- Message display area -->
9  <p id="message">
10     <?php
11     // If any messages are passed via GET (from processing), display them.
12     if(isset($_GET['error_message'])) echo "<span style='color:red;'>" . $_GET['error_message'] . "</span>";
13     if(isset($_GET['success_message'])) echo "<span style='color:green;'>" . $_GET['success_message'] . "</span>";
14     if(isset($_GET['message'])) echo "<span>" . $_GET['message'] . "</span>";
15     ?>
16     </p>
17     </header>
18     </section>
19     </form>
20
21     <!-- Include the list of active job types -->
22     <?php include 'JobTypeList.php'; ?>
23
24     </section>
25     </main>
26 </body>
27 </html>
```

Amend Job Type

AmendJobType.php

```
1  <!--
2  Emma Saliu
3  C00297032
4  GARAGE PROJECT
5  Amend job type
6  -->
7  <?php
8  include 'DBconnection.php';
9
10 if ($_SERVER['REQUEST_METHOD'] === 'POST') {
11     // Display the POST data for debugging
12     echo "<pre>";
13     print_r($_POST);
14     echo "</pre>";
15
16     // Retrieve form values
17     $job_type_id = $_POST['job_type_id'];
18     $job_type_name = $_POST['job_type_name'];
19     $description = $_POST['description'];
20     $price = $_POST['price'];
21     $lead_mechanic = $_POST['lead_mechanic'];
22
23     // Ensure a job type is selected
24     if (!$job_type_id) {
25         echo "<h2>Please select a Job Type</h2>";
26         exit();
27     }
28
29     try {
30         // Update the Job_Type record
31         $update = $con->prepare("UPDATE `Job_Type`
32             SET `Job_Type_Name` = :name,
33                 `Description` = :description,
34                 `Price` = :price,
35                 `Lead_Mechanic` = :lead_mechanic
36             WHERE `Job_Type_ID` = :id");
37         $update->bindParam(':id', $job_type_id);
38         $update->bindParam(':name', $job_type_name);
39         $update->bindParam(':description', $description);
40         $update->bindParam(':price', $price);
41         $update->bindParam(':lead_mechanic', $lead_mechanic);
42
43         if ($update->execute()) {
44             echo "<h2>Data is updated in the database successfully.</h2>";
45         } else {
46             echo "<h2>Error during updating data in the database!</h2>";
47             error_log("Database Error: " . print_r($update->errorInfo(), true));
48         }
49     } catch (PDOException $e) {
50         error_log("Database Error: " . $e->getMessage());
51     }
52 }
53 ?>
```

AmendJobTypeForm.php;



```
1  <!--
2  Emma Saliu
3  C00297032
4  GARAGE PROJECT
5  Amend job type
6  -->
7  <?php
8  include 'DBconnection.php';
9
10 if ($_SERVER['REQUEST_METHOD'] === 'POST') {
11     // Display the POST data for debugging
12     echo "<pre>";
13     print_r($_POST);
14     echo "</pre>";
15
16     // Retrieve form values
17     $job_type_id = $_POST['job_type_id'];
18     $job_type_name = $_POST['job_type_name'];
19     $description = $_POST['description'];
20     $price = $_POST['price'];
21     $lead_mechanic = $_POST['lead_mechanic'];
22
23     // Ensure a job type is selected
24     if (!$job_type_id) {
25         echo "<h2>Please select a Job Type</h2>";
26         exit();
27     }
28
29     try {
30         // Update the Job_Type record
31         $update = $con->prepare("UPDATE `Job_Type`"
32                               . " SET `Job_Type_Name` = :name,
33                               . `Description` = :description,
34                               . `Price` = :price,
35                               . `Lead_Mechanic` = :lead_mechanic
36                               . WHERE `Job_Type_ID` = :id");
37         $update->bindParam(':id', $job_type_id);
38         $update->bindParam(':name', $job_type_name);
39         $update->bindParam(':description', $description);
40         $update->bindParam(':price', $price);
41         $update->bindParam(':lead_mechanic', $lead_mechanic);
42
43         if ($update->execute()) {
44             echo "<h2>Data is updated in the database successfully.</h2>";
45         } else {
46             echo "<h2>Error during updating data in the database!</h2>";
47             error_log("Database Error: " . print_r($update->errorInfo(), true));
48         }
49     } catch (PDOException $e) {
50         error_log("Database Error: " . $e->getMessage());
51     }
52 }
53 ?>
```

Cont.

```
1 <!-- Form to amend a job type -->
2 <form action="./AmendJobType.php" method="post" id="form" class="form">
3 <label for="JobType">Job Type List:</label>
4 <select data-amend-jobtype-dropdown="JobType" class="drop_down_list" name="job_type_id" id="JobType" data_file_link="fetchJobTypes.php" data_value="Job_Type_ID" data_text="Job_Type_Name">
5 <option value="">---Select Job Type---</option>
6 </select>
7 // Populate the dropdown from the Job_Type table.
8 $sql = "SELECT Job_Type_ID, Job_Type_Name FROM Job_Type WHERE Delete_Flag = '0'";
9 $result = $con->query($sql);
10 if ($result) {
11     while ($row = $result->fetch(PDO::FETCH_ASSOC)) {
12         echo "<option value='{$row['Job_Type_ID']}'>{$row['Job_Type_Name']}";
13     }
14 }
15 >
16 </select>
17
18 <!-- Job Type ID (read-only) -->
19 <div class="inputMain">
20     <label for="job_type_id_display">Job Type ID: </label>
21     <input type="text" id="job_type_id_display" placeholder="Job Type ID" name="job_type_id_display" readonly>
22 </div>
23
24 <!-- Job Type Name -->
25 <div class="inputMain">
26     <label for="job_type_name">Job Type Name: </label>
27     <input type="text" id="job_type_name" placeholder="Enter job type name" name="job_type_name">
28 </div>
29
30 <!-- Description -->
31 <div class="inputMain">
32     <label for="description">Description: </label>
33     <input type="text" id="description" placeholder="Enter description" name="description">
34 </div>
35
36 <!-- Base Price -->
37 <div class="inputMain">
38     <label for="price">Base Price: </label>
39     <input type="number" id="price" placeholder="Enter base price" name="price">
40 </div>
41
```

Cont.

```
1  <!-- Lead Mechanic -->
2      <div class="inputMain">
3          <label for="lead_mechanic">Lead Mechanic: </label>
4          <select class="drop_down_list" name="lead_mechanic" id="lead_mechanic">
5              <option value="">---Select Lead Mechanic---</option>
6              <?php
7                  $sql2 = "SELECT Mechanic_ID, Mechanic_Name FROM Mechanics WHERE Deleted_Flag = '0'";
8                  $result2 = $con->query($sql2);
9                  if ($result2) {
10                      while ($row2 = $result2->fetch(PDO::FETCH_ASSOC)) {
11                          echo "<option value='{$row2['Mechanic_ID']}'>{$row2['Mechanic_Name']}
```

JobTypeList.php;

```
1  <!--
2  Emma Saliu
3  C00297032
4  GARAGE PROJECT
5  job type list
6  -->
7  <?php
8  include 'DBconnection.php';
9
10 $sql = "SELECT jt.Job_Type_ID, jt.Job_Type_Name, jt.Description, jt.Price, m.Mechanic_Name AS Lead_Mechanic
11      FROM Job_Type jt
12      LEFT JOIN Mechanics m ON jt.Lead_Mechanic = m.Mechanic_ID
13      WHERE jt.Delete_Flag = '0'";
14 $result = $con->query($sql);
15 $row = $result->fetchAll(PDO::FETCH_ASSOC);
16 ?>
17
18 <div class="suppliers_list_container_heading">
19   <h2 class="heading">List of Active Job Types</h2>
20 </div>
21
22 <div class="supplier_table_conatainer">
23   <table class="table">
24     <tr class="table-column">
25       <th>Job Type ID</th>
26       <th>Job Type Name</th>
27       <th>Description</th>
28       <th>Base Price</th>
29       <th>Lead Mechanic</th>
30     </tr>
31   <?php
32   if (!empty($row)) {
33     foreach ($row as $rs) {
34       echo "
35         <tr class='table_row'>
36           <td>{$rs['Job_Type_ID']}</td>
37           <td>{$rs['Job_Type_Name']}</td>
38           <td>{$rs['Description']}</td>
39           <td>{$rs['Price']}</td>
40           <td>{$rs['Lead_Mechanic']}</td>
41         </tr>";
42     }
43   } else {
44     echo "
45       <tr>
46         <td colspan='5'><h2>No Records Available Yet, Come Back Later!</h2></td>
47       </tr>";
48   }
49   ?>
50   </table>
51 </div>
```

Complete Job

CompleteJob.php

```
1  <!--
2  Emma Saliu
3  C00297032
4  GARAGE PROJECT
5  complete job
6  -->
7  <?php
8 // completeJob.php
9 include 'DBconnection.php';
10
11 if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['complete_job'])) {
12     $job_id = $_POST['job_id'];
13     $time_taken = $_POST['time_taken'];
14     $spare_parts_stock = $_POST['spare_part_stock']; // Array of stock numbers.
15     $spare_parts_qty = $_POST['spare_part_qty']; // Array of corresponding quantities.
16
17     try {
18         // Begin transaction for atomicity.
19         $con->beginTransaction();
20
21         // 1. Update the Job table: set the time taken and mark the job as 'Completed'
22         $stmt = $con->prepare("UPDATE Job SET Time_Taken = ?, Status = 'Completed' WHERE Job_ID = ?");
23         $stmt->execute([$time_taken, $job_id]);
24
25         // 2. Process each spare part used.
26         for ($i = 0; $i < count($spare_parts_stock); $i++) {
27             $stock_id = trim($spare_parts_stock[$i]);
28             $quantity = intval($spare_parts_qty[$i]);
29
30             // Insert into Job_Parts.
31             $stmtInsert = $con->prepare("INSERT INTO Job_Parts (Job_Part_ID, Job_ID, Stock_ID, Quantity) VALUES (?, ?, ?, ?)");
32             $stmtInsert->execute([$newJobPartID, $job_id, $stock_id, $quantity]);
33
34             // Update the Stock_Item table to deduct the used quantity.
35             $stmtUpdate = $con->prepare("UPDATE Stock_Item SET Stock_Quantity = Stock_Quantity - ? WHERE Stock_ID = ?");
36             $stmtUpdate->execute([$quantity, $stock_id]);
37
38         }
39
40         // Commit the transaction.
41         $con->commit();
42
43         // Redirect to a payment or confirmation page.
44         header("Location: paymentForm.php?job_id=" . urlencode($job_id));
45         exit();
46     } catch (Exception $e) {
47         // Roll back the transaction if an error occurs.
48         $con->rollBack();
49         echo "Transaction failed: " . $e->getMessage();
50     }
51 } else {
52     // If accessed directly, redirect back to the form.
53     header("Location: completeJobForm.php");
54     exit();
55 }
56
57 ?>
```

Cont.

```
1  if ($stock_id !== '' && $quantity > 0) {
2      // Manually generate a unique Job_Part_ID.
3      $stmtMax = $con->query("SELECT COALESCE(MAX(Job_Part_ID), 0) AS max_id FROM Job_Parts");
4      $row = $stmtMax->fetch(PDO::FETCH_ASSOC);
5      $newJobPartID = $row['max_id'] + 1;
6
7      // Insert into Job_Parts.
8      $stmtInsert = $con->prepare("INSERT INTO Job_Parts (Job_Part_ID, Job_ID, Stock_ID, Quantity) VALUES (?, ?, ?, ?)");
9      $stmtInsert->execute([$newJobPartID, $job_id, $stock_id, $quantity]);
10
11     // Update the Stock_Item table to deduct the used quantity.
12     $stmtUpdate = $con->prepare("UPDATE Stock_Item SET Stock_Quantity = Stock_Quantity - ? WHERE Stock_ID = ?");
13     $stmtUpdate->execute([$quantity, $stock_id]);
14
15 }
16
17 // Commit the transaction.
18 $con->commit();
19
20 // Redirect to a payment or confirmation page.
21 header("Location: paymentForm.php?job_id=" . urlencode($job_id));
22 exit();
23 } catch (Exception $e) {
24     // Roll back the transaction if an error occurs.
25     $con->rollBack();
26     echo "Transaction failed: " . $e->getMessage();
27 }
28 } else {
29     // If accessed directly, redirect back to the form.
30     header("Location: completeJobForm.php");
31     exit();
32 }
33
34 ?>
```

CompleteJobTypeForm.php;

```
1  <!--
2  Emma Saliu
3  C00297032
4  GARAGE PROJECT
5  complete job form
6  -->
7  <?php
8  include 'DBconnection.php';
9
10 // Query for jobs in progress with their job type names.
11 $stmt = $con->prepare("SELECT j.Job_ID, jt.Job_Type_Name AS Job_Name
12                         FROM Job j
13                         JOIN Job_Type jt ON j.Job_Type_ID = jt.Job_Type_ID
14                         WHERE j.Status IS NULL OR j.Status = '' OR j.Status = '0'");
15 $stmt->execute();
16 $jobs = $stmt->fetchAll(PDO::FETCH_ASSOC);
17 ?>
18 <!DOCTYPE html>
19 <html lang="en">
20 <head>
21   <meta charset="UTF-8" />
22   <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
23   <title>Complete Job | GMS</title>
24   <link rel="stylesheet" href=".//Assets/css>ShowSuppliers.css"/>
25   <script>
26     // Adds an additional spare part row.
27     function addSparePartRow() {
28       var table = document.getElementById("sparePartsTable");
29       var row = table.insertRow(-1);
30       var cell1 = row.insertCell(0);
31       var cell2 = row.insertCell(1);
32       cell1.innerHTML = '<input type="text" name="spare_part_stock[]" required>';
33       cell2.innerHTML = '<input type="number" name="spare_part_qty[]" required min="1">';
34     }
35
```

Cont.

```
1 // Optional: Fetch additional job details when a job is selected.
2     function fetchJobDetails(jobId) {
3         if (!jobId) {
4             document.getElementById("job_details").innerHTML = "";
5             return;
6         }
7         fetch(`fetch_job_details.php?job_id=${jobId}`)
8             .then(response => response.text())
9             .then(html => {
10                 document.getElementById("job_details").innerHTML = html;
11             });
12     }
13     </script>
14 </head>
15 <body>
16     <main class="main_form_container">
17         <div class="div">
18             <div class="form_title">
19                 <h1>Complete Job</h1>
20             </div>
21             <form method="POST" action="completeJob.php" class="form">
22                 <!-- Job Selection Drop-Down -->
23                 <div class="inputMain">
24                     <label for="job_id">Select Job:</label>
25                     <select name="job_id" id="job_id" class="drop_down_list" required onchange="fetchJobDetails(this.value)">
26                         <option value="">--Select a job--</option>
27                         <?php foreach ($jobs as $job): ?>
28                             <option value=<?php echo htmlspecialchars($job['Job_ID']); ?>>
29                             <?php echo "Job ID: " . htmlspecialchars($job['Job_ID']) . " - " . htmlspecialchars($job['Job_Name']); ?>
30                         </option>
31                     <?php endforeach; ?>
32                     </select>
33                 </div>
34
35                 <!-- Job Details (loaded dynamically via AJAX if needed) -->
36                 <div id="job_details" class="inputMain">
37                     <!-- Additional job details can appear here. -->
38                 </div>
39
40             <!-- Total Time Taken
```

Cont.

```
1  <!-- Total Time Taken -->
2      <div class="inputMain">
3          <label for="time_taken">Total Time Taken (in minutes):</label>
4          <input type="number" name="time_taken" id="time_taken" required min="1">
5      </div>
6
7  <!-- Spare Parts Used Section -->
8      <div class="inputMain">
9          <h2>Spare Parts Used</h2>
10         <table id="sparePartsTable" border="1">
11             <tr>
12                 <th>Stock Number</th>
13                 <th>Quantity</th>
14             </tr>
15             <tr>
16                 <td><input type="text" name="spare_part_stock[]" required></td>
17                 <td><input type="number" name="spare_part_qty[]" required min="1"></td>
18             </tr>
19         </table>
20         <button type="button" onclick="addSparePartRow()">Add Another Spare Part</button>
21     </div>
22
23  <!-- Form Buttons -->
24      <div class="form_buttons">
25          <input type="reset" value="Cancel" class="formButton">
26          <input type="submit" name="complete_job" value="Complete Job" class="formButton">
27      </div>
28  </form>
29 </div>
30 </main>
31 </body>
32 </html>
```

Gleb Tutubalin

New Invoices from Supplier

newInvoicesForm.php:

```
1  <!--
2      Student: Gleb Tutubalin C00290944
3      Date:    02.2025
4      Description: Creating a new invoice from a supplier and adding it to the database
5  -->
6  <!DOCTYPE html>
7  <html lang="en">
8  <head>
9      <meta charset="UTF-8">
10     <meta name="viewport" content="width=device-width, initial-scale=1.0">
11     <title>New Invoice|GMS</title>
12     <link rel="stylesheet" href=".//Assets/css/invoicesForm.css">
13 </head>
14 <body>
15     <main class="main_form_container">
16         <div class="div">
17             <div class="form_title">
18                 <h1>New Invoice from Supplier</h1>
19             </div>
20             <!-- Form for adding a new invoice -->
21             <form method="post" action="AddNewInvoices.php" id="form" class="form" onsubmit="return validateForm()">
22
23             <!-- Select the supplier -->
24             <div class="inputMain">
25                 <label for="supplier_id">Supplier:</label>
```

newInvoicesForm.php continued:

```
1      <select name="supplier_id" id="supplier_id" class="inputMain">
2          <?php
3              include 'DBconnection.php';
4              // Getting all suppliers from the database
5              try {
6                  // Selecting all suppliers that are not deleted
7                  $select = $con->prepare("SELECT `Supplier_ID`, `Name`, `Address` FROM `Supplier` WHERE `Delete_Flag`=0");
8                  // Executing the query
9                  $select->execute();
10                 // Fetching the result
11                 $result = $select->fetchAll(PDO::FETCH_ASSOC);
12
13                 // Displaying the suppliers in the dropdown
14                 if ($result) {
15                     foreach ($result as $row) {
16                         echo "<option value='" . $row["Supplier_ID"] . "'>" . $row["Name"] . " - " . $row["Address"] . "</option>";
17                     }
18                 } else {
19                     echo "<option value='>Suppliers are not found</option>";
20                 }
21             } catch (PDOException $e) {
22                 echo "Error: " . $e->getMessage();
23             }
24         ?>
25     </select>
26
```

newInvoicesForm.php continued:

```
1      </div>
2
3      <!-- Reference Number -->
4      <div class="inputMain">
5          <label for="invoice_reference_number">Reference Number:</label>
6          <input type="text" name="invoice_reference_number" id="invoice_reference_number" placeholder="Enter reference number" required pattern="[A-Za-z0-9]*" title="Only letters and numbers are allowed">
7      </div>
8
9      <!-- Date of Invoice -->
10     <div class="inputMain">
11         <label for="invoice_date">Date of Invoice:</label>
12         <input type="date" name="invoice_date" id="invoice_date" required>
13     </div>
14
15     <!-- Amount of Invoice -->
16     <div class="inputMain">
17         <label for="invoice_amount">Amount of Invoice:</label>
18         <input type="number" name="invoice_amount" id="invoice_amount" placeholder="Enter amount" required>
19     </div>
20
21     <!-- Form Buttons -->
22     <div class="form_buttons">
23         <input type="reset" value="Cancel" class="formButton">
24         <input type="submit" value="Add Invoice" class="formButton">
25     </div>
26
```

newInvoicesForm.php continued:

```
1      <!-- success /error message will show here -->
2      <p id="message"></p>
3      </form>
4      </div>
5  </main>
6  <script>
7      // Setting the current date as the default value for the date field
8      document.addEventListener("DOMContentLoaded", function() {
9          let dateField = document.getElementById("invoice_date");
10         let today = new Date().toISOString().split('T')[0];
11         dateField.value = today;
12     });
13
14     // Validation for the date field and confirmation message
15     function validateForm() {
16         let dateField = document.getElementById("invoice_date");
17         let selectedDate = new Date(dateField.value);
18         let today = new Date();
19
20         // Ensure the selected date is not in the future
21         if (selectedDate > today) {
22             alert("The date of the invoice cannot be in the future.");
23             return false;
24         }
25     }
```

newInvoicesForm.php continued:

```
1 // Show confirmation message
2 return confirm("Are you sure you want to submit this invoice?");
3 }
4 </script>
5 </body>
6 </html>
```

AddNewInvoices.php:

```
1 <!--
2   Student: Gleb Tutubalin C00290944
3   Date:    20.02.2025
4   Description: Creating a new invoice from a supplier and adding it to the database
5 -->
6 <?php
7 include 'DBconnection.php'; // Database connection
8 if ($_SERVER['REQUEST_METHOD'] === 'POST') {
9   // Getting the values from the form
10  $supplier_id = $_POST['supplier_id'];
11  $invoice_reference_number = $_POST['invoice_reference_number'];
12  $invoice_date = $_POST['invoice_date'];
13  $invoice_amount = $_POST['invoice_amount'];
14 try {
15   // insert the new invoice into the database
16   $insert = $con->prepare("INSERT INTO `Invoice`(`Supplier_ID`, `Invoice_Ref`, `Invoice_Date`, `Amount`)
17   VALUES (:supplier_id, :invoice_reference_number, :invoice_date, :invoice_amount)");
18
19   // Bind the parameters
20   $insert->bindParam(':supplier_id', $supplier_id);
21   $insert->bindParam(':invoice_reference_number', $invoice_reference_number);
22   $insert->bindParam(':invoice_date', $invoice_date);
23   $insert->bindParam(':invoice_amount', $invoice_amount);
24
25   // Execute the query
26 }
```

AddNewInvoices.php continued:

```
● ● ●
1      if($insert->execute())
2      {
3          echo "<h2>New invoice was added successfully.</h2>";
4      }
5      else
6      {
7          echo "error occur during inserting data to database.";
8          error_log("Database Error: ".print_r($insert->errorInfo(),true));
9      }
10     }
11 }
12 }
13 catch (PDOException $e)
14 {
15     echo("Database error: ".$e->getMessage());
16     error_log("database error: ".$e->getMessage());
17 }
18 }
19 ?>
```

Reorder Stock

reorderStockForm.php:

```
1  <!--
2      Student: Gleb Tutubalin C00290944
3      Date:    03.2025
4      Description: This file is responsible for reordering stock items.
5  -->
6  <!DOCTYPE html>
7  <html lang="en">
8  <head>
9      <meta charset="UTF-8">
10     <meta name="viewport" content="width=device-width, initial-scale=1.0">
11     <title>Reorder Stock</title>
12     <link rel="stylesheet" href=".Assets/css/AdminDashboard.css">
13     <link rel="stylesheet" href=".Assets/css>ShowSuppliers.css">
14     <style>
15         .disabled-row {
16             opacity: 0.5; /* Make the row appear faded */
17             pointer-events: none; /* Prevent interaction with the row */
18         }
19     </style>
20 </head>
21 <body>
22     <main class="delete_supplier_main_container">
23         <section class="suppliers_list_section_container">
24             <div class="form_title">
25                 <h2>Reorder Stock</h2>
26             </div>
27         </section>
28     </main>
29     <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js" integrity="sha384-ka7Sk0Gln4ZUI0q6E2d+uq9jwZ85c+sdVzbnQ+Ql9u9Zngx3Zn2zWcWUcB1M4" crossorigin="anonymous"></script>
30 
```

reorderStockForm.php continued:

```
1      </div>
2      <!-- Stock Listing -->
3      <div class="supplier_table_conatainer">
4          <table class="table">
5              <tr class="table-column">
6                  <th>Stock Number</th>
7                  <th>Stock Description</th>
8                  <th>Qty in Stock</th>
9                  <th>Reorder Level</th>
10                 <th>Reorder Qty</th>
11                 <th>Cost Price</th>
12                 <th>Retail Price</th>
13                 <th>Supplier</th>
14                 <th>Action</th>
15             </tr>
16             <?php include 'reorderStock.php'; ?>
17         </table>
18     </div>
19     <!-- Current Order Section -->
20     <div id="current-order">
21         <h2>Current Reorder Items</h2>
22         <form action="createOrder.php" method="POST" id="form" class="form">
23             <div id="selected-items"></div>
24             <input type="hidden" name="supplier_id" id="supplier-id">
25
```

reorderStockForm.php continued:

```
1          <!--<button type="button" onclick="addMoreItems()" id="add-more" disabled>Add More Items</button>-->
2          <button type="submit">Generate Order</button>
3      </form>
4  </div>
5 </section>
6 </main>
7 <script>
8     // JavaScript code to handle the reorder stock form
9     let selectedSupplierId = null;
10    // Get all stock rows
11    const stockRows = document.querySelectorAll('tr[data-supplier-id]');
12
13    // Add event listeners to the select buttons
14    function selectItem(stockId, supplierId, suggestedQty) {
15        if (!selectedSupplierId) {
16            selectedSupplierId = supplierId;
17            document.getElementById('supplier-id').value = supplierId;
18            //document.getElementById('add-more').disabled = false;
19            filterSupplierRows(supplierId); // Show only the selected supplier's stock items
20        }
21
22        // Prompt the user to enter the quantity
23        let qty = prompt(`Enter order quantity (Suggested: ${suggestedQty})`, suggestedQty);
24
25        // Validate the input quantity
26    }

```

reorderStockForm.php continued:

```
 1  if (qty !== null) {
 2      qty = parseInt(qty, 10); // Convert the input to an integer
 3      if (isNaN(qty) || qty <= 0) {
 4          // If the input is not a valid positive number, show an alert and return
 5          alert("Please enter a valid positive number for the quantity.");
 6          return;
 7      }
 8      addToOrder(stockId, qty, supplierId); // Add the selected item to the order
 9  }
10 }
11
12 function filterSupplierRows(supplierId) {
13     stockRows.forEach(row => {
14         const selectButton = row.querySelector('button'); // Get the "Select" button in the row
15         if (row.dataset.supplierId !== supplierId) {
16             // Disable the button for rows that do not match the selected supplier
17             selectButton.disabled = true;
18             row.classList.add('disabled-row'); // Optional: Add a visual indicator for disabled rows
19         } else {
20             // Enable the button for rows that match the selected supplier
21             selectButton.disabled = false;
22             row.classList.remove('disabled-row'); // Remove the visual indicator
23         }
24     });
25 }
26 }
```

reorderStockForm.php continued:

```
1 // Add the selected item to the order
2 function addToOrder(stockId, qty, supplierId) {
3     const container = document.getElementById('selected-items');
4     const existingInput = container.querySelector(`input[name="stock_ids[]"][value="${stockId}"]`);
5
6     // Check if the supplier ID matches the selected supplier
7     if (supplierId !== selectedSupplierId) {
8         alert("You cannot add items from a different supplier. Please select items from the same supplier.");
9         return; // Exit the function without adding the item
10    }
11
12    if (existingInput) {
13        // If the stock ID already exists, update the quantity
14        const quantityInput = existingInput.nextElementSibling; // Get the corresponding quantity input
15        const currentQty = parseInt(quantityInput.value, 10); // Get the current quantity
16        const newQty = currentQty + qty;
17        quantityInput.value = newQty;
18
19        // Update the displayed quantity
20        const displayDiv = quantityInput.nextElementSibling;
21        displayDiv.textContent = `Stock #${stockId} - Quantity: ${newQty}`;
22    } else {
23        // If the stock ID does not exist, add a new entry
24        container.innerHTML += `
25
```

reorderStockForm.php continued:

```
1          <input type="hidden" name="stock_ids[]" value="${stockId}">
2          <input type="hidden" name="quantities[]" value="${qty}">
3          <div>Stock #${stockId} - Quantity: ${qty}</div>;
4      }
5  }
6
7 // Add more items to the order
8 function addMoreItems() {
9     filterSupplierRows(selectedSupplierId);
10 }
11
12 // Add confirmation before form submission
13 document.getElementById('form').addEventListener('submit', function (event) {
14     const confirmation = confirm("Are you sure you want to submit this order?");
15     if (!confirmation) {
16         event.preventDefault(); // Prevent form submission if the user cancels
17     }
18 });
19 </script>
20 </body>
21 </html>
```

reorderStock.php:

```
1 <!--
2     Student: Gleb Tutubalin C00290944
3     Date:    03.2025
4     Description: This file is responsible for reordering stock items.
5 -->
6 <?php
7 // reorderStock.php
8 include 'DBconnection.php';
9
10 try {
11     $sql = "SELECT Stock_Item.Stock_ID, Stock_Item.Description, Stock_Item.Stock_Quantity,
12             Stock_Item.Reorder_Level, Stock_Item.Reorder_Quantity, Stock_Item.Cost_Price,
13             Stock_Item.Retail_Price, Supplier.Name, Supplier.Supplier_ID
14         FROM Stock_Item
15     INNER JOIN Supplier ON Stock_Item.Supplier_ID = Supplier.Supplier_ID";
16
17     $result = $con->query($sql);
18
19     if ($result) {
20         if ($result->rowCount() > 0) {
21             while ($row = $result->fetch(PDO::FETCH_ASSOC)) {
22                 echo "<tr class='table-row' data-supplier-id='{$row['Supplier_ID']}'>";
23                 echo "<td>" . htmlspecialchars($row['Stock_ID']) . "</td>";
24                 echo "<td>" . htmlspecialchars($row['Description']) . "</td>";
25
26                 echo "<td>" . htmlspecialchars($row['Stock_Quantity']) . "</td>";
27                 echo "<td>" . htmlspecialchars($row['Reorder_Level']) . "</td>";
28                 echo "<td>" . htmlspecialchars($row['Reorder_Quantity']) . "</td>";
29                 echo "<td>" . htmlspecialchars($row['Cost_Price']) . "</td>";
30                 echo "<td>" . htmlspecialchars($row['Retail_Price']) . "</td>";
31                 echo "<td>" . htmlspecialchars($row['Name']) . "</td>";
32                 echo "<td><button onclick='selectItem(\"{$row['Stock_ID']}\", \"{$row['Supplier_ID']}\", {$row['Reorder_Quantity']})'>
33                     Select
34                 </button></td>";
35             }
36         } else {
37             echo "<tr><td colspan='9'>No stock items found</td></tr>";
38         }
39     }
40 } catch (PDOException $e) {
41     echo "<tr><td colspan='9'>Database error: " . htmlspecialchars($e->getMessage()) . "</td></tr>";
42 }
43 ?>
```

reorderStock.php continued:

```
1     echo "<td>" . htmlspecialchars($row['Stock_Quantity']) . "</td>";
2     echo "<td>" . htmlspecialchars($row['Reorder_Level']) . "</td>";
3     echo "<td>" . htmlspecialchars($row['Reorder_Quantity']) . "</td>";
4     echo "<td>" . htmlspecialchars($row['Cost_Price']) . "</td>";
5     echo "<td>" . htmlspecialchars($row['Retail_Price']) . "</td>";
6     echo "<td>" . htmlspecialchars($row['Name']) . "</td>";
7     echo "<td><button onclick='selectItem(\"{$row['Stock_ID']}\", \"{$row['Supplier_ID']}\", {$row['Reorder_Quantity']})'>
8         Select
9     </button></td>";
10    echo "</tr>";
11
12    }
13} else {
14    echo "<tr><td colspan='9'>No stock items found</td></tr>";
15}
16
17
18
19
```

createOrder.php:

```
1  <!--
2      Student: Gleb Tutubalin C00290944
3      Date:    03.2025
4      Description: This file is responsible for creating an order for the selected supplier and stock items.
5  -->
6 <?php
7 // Connection to the database
8 include 'DBconnection.php';
9
10 // Check if the form was submitted
11 // If the form was submitted, create the order and display the reorder letter
12 if ($_SERVER['REQUEST_METHOD'] === 'POST') {
13     try {
14         // Start a transaction
15         $con->beginTransaction();
16
17         // Get supplier details
18         $supplierId = $_POST['supplier_id'];
19         $stmt = $con->prepare("SELECT * FROM Supplier WHERE Supplier_ID = ?");
20         $stmt->execute([$supplierId]);
21         $supplier = $stmt->fetch(PDO::FETCH_ASSOC);
22
23         // Check if the supplier exists
24         if (!$supplier) {
25             throw new Exception("Supplier not found.");
26
27     }
28
29     // Create order
30     $orderStmt = $con->prepare("INSERT INTO `Order` (Supplier_ID, Order_Date, Delivery_Date) VALUES (?, CURDATE(), DATE_ADD(CURDATE(), INTERVAL 14 DAY))");
31     $orderStmt->execute([$supplierId]);
32     // Get the last inserted Order_ID
33     $orderId = $con->lastInsertId();
34
35     // Add order items
36     foreach ($_POST['stock_ids'] as $key => $stockId) {
37         $quantity = $_POST['quantities'][$key];
38
39         // Insert order item
40         $itemStmt = $con->prepare("INSERT INTO `Stock_Item/Order` (Order_ID, Stock_ID, Quantity) VALUES (?, ?, ?)");
41         $itemStmt->execute([$orderId, $stockId, $quantity]);
42     }
43
44     $con->commit();
45
46     // Generate reorder letter
47     echo "<div style='font-family: Arial, sans-serif; line-height: 1.6;'>";
48     echo "<p style='text-align: right;'>Gerry's Garage,<br>High Street,<br>Carlow<br>" . date("Y-m-d") . "</p>";
49     echo "<p>" . htmlspecialchars($supplier['Name']) . ",<br>" . nl2br(htmlspecialchars($supplier['Address'])) . "</p>";
50     echo "<p><strong>Order Number:</strong> " . htmlspecialchars($orderId) . "</p>";
51
52 }
```

createOrder.php continued:

```
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
```

createOrder.php continued:

```
1      echo "<p>Please supply the following stock item(s):</p>";
2      echo "<table style='width: 100%; border-collapse: collapse; margin-top: 20px;'>";
3      echo "<thead>";
4      echo "<tr style='background-color: #f2f2f2;'>";
5      echo "<th style='border: 1px solid #dd; padding: 8px; text-align: left;'>Stock</th>";
6      echo "<th style='border: 1px solid #dd; padding: 8px; text-align: left;'>Item Description</th>";
7      echo "<th style='border: 1px solid #dd; padding: 8px; text-align: left;'>Quantity</th>";
8      echo "</tr>";
9      echo "</thead>";
10     echo "<tbody>";
11
12     // Fetch stock item details
13     foreach ($_POST['stock_ids'] as $key => $stockId) {
14         $quantity = $_POST['quantities'][$key];
15         $stockStmt = $con->prepare("SELECT Description FROM Stock_Item WHERE Stock_ID = ?");
16         $stockStmt->execute([$stockId]);
17         $stock = $stockStmt->fetch(PDO::FETCH_ASSOC);
18
19         echo "<tr>";
20         echo "<td style='border: 1px solid #dd; padding: 8px;'>" . htmlspecialchars($stockId) . "</td>";
21         echo "<td style='border: 1px solid #dd; padding: 8px;'>" . htmlspecialchars($stock['Description']) . "</td>";
22         echo "<td style='border: 1px solid #dd; padding: 8px;'>" . htmlspecialchars($quantity) . "</td>";
23         echo "</tr>";
24     }
25
26
```

createOrder.php continued:

```
1      echo "</tbody>";
2      echo "</table>";
3      echo "<p style= 'margin-top: 20px;'>Yours sincerely,<br><br>xxxxxxxxxxxxxxxxxxxxx<br>Stock Controller</p>";
4      echo "</div>";
5
6  } catch (PDOException $e) {
7      // Rollback the transaction if an error occurred
8      $con->rollBack();
9      die("Order creation failed: " . $e->getMessage());
10 } catch (Exception $e) {
11     // Rollback the transaction if an error occurred
12     $con->rollBack();
13     die("Order creation failed: " . $e->getMessage());
14 }
15 }
16 ?>
```

Payment to Supplier

paymentForm.php:

```
1  <!--
2      Student: Gleb Tutubalin C00290944
3      Date:    03.2025
4      Description: This file is responsible for processing payments to suppliers.
5  -->
6  <!DOCTYPE html>
7  <html lang="en">
8  <head>
9      <meta charset="UTF-8">
10     <meta name="viewport" content="width=device-width, initial-scale=1.0">
11     <title>Payment to Suppliers</title>
12     <link rel="stylesheet" href=".Assets/css/AdminDashboard.css">
13     <link rel="stylesheet" href=".Assets/css>ShowSuppliers.css">
14     <script>
15         function confirmSubmission() {
16             return confirm("Are you sure you want to process this payment?");
17         }
18     </script>
19 </head>
20 <body>
21     <main class="delete_supplier_main_container">
22         <div class="form-title">
23             <h1>Payment to Suppliers</h1>
24         </div>
25 
```

paymentForm.php continued:

```
1  <!-- Combined Form -->
2  <form method="post" action="" id="paymentForm" class="form" onsubmit="return confirmSubmission();">
3      <label for="supplier_id">Supplier:</label>
4      <select name="supplier_id" id="supplier_id" class="drop_down_list" required onchange="this.form.submit()">
5          <option value="" disabled selected><--Select--><-- Default option -->
6      </select>
7      <?php
8          include 'DBconnection.php';
9          // Getting all suppliers from the database
10         try {
11             $select = $con->prepare("SELECT `Supplier_ID`, `Name`, `Address` FROM `Supplier` WHERE `Delete_Flag` = 0");
12             $select->execute();
13             $result = $select->fetchall(PDO::FETCH_ASSOC);
14         }
15         if ($result) {
16             foreach ($result as $row) {
17                 // Pre-select the supplier if the form was submitted
18                 $selected = (isset($_POST['supplier_id']) && $_POST['supplier_id'] == $row['Supplier_ID']) ? "selected" : "";
19                 echo "<option value='".$row['Supplier_ID']."' ". $selected .". htmlspecialchars($row['Name']) . " - " . htmlspecialchars($row['Address']) . "</option>";
20             }
21         } else {
22             echo "<option value='>No suppliers found</option>";
23         }
24     } catch (PDOException $e) {
25         echo "Error: " . htmlspecialchars($e->getMessage()); 
```

paymentForm.php continued:

```
1      }
2      ?>
3      </select><br><br>
4  </form>
5
6  <?php
7  // Check if the form was submitted and the supplier was selected
8  if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['supplier_id'])) {
9      $supplier_id = $_POST['supplier_id']; // Get the supplier ID from the form submission
10
11     // Getting all unpaid invoices for the selected supplier
12     try {
13         $stmt = $con->prepare("SELECT `Invoice_Ref`, `Amount` FROM `Invoice` WHERE `Supplier_ID` = ? AND `Status` = 'ISSUED'");
14         $stmt->execute([$supplier_id]);
15         $invoices = $stmt->fetchAll(PDO::FETCH_ASSOC);
16
17         // Display the unpaid invoices
18         if (!empty($invoices)) {
19             $total_amount = 0;
20
21             // Calculate the total amount of unpaid invoices
22             foreach ($invoices as $invoice) {
23                 $total_amount += $invoice['Amount'];
24             }
25
26     
```

paymentForm.php continued:

```
1      // Generating the letter to the supplier
2      $stmt = $con->prepare("SELECT `Name`, `Address` FROM `Supplier` WHERE `Supplier_ID` = ?");
3      $stmt->execute([$supplier_id]);
4      $supplier = $stmt->fetch(PDO::FETCH_ASSOC);
5
6      // Display the unpaid invoices
7      if ($supplier) {
8          echo "<div class='form_title'>";
9          echo "<h2 class='heading'>Unpaid Invoices for " . htmlspecialchars($supplier['Name']) . "</h2>";
10         echo "</div>";
11         echo "<div class='supplier_table_container'>";
12         echo "<table border='1' style='width: 100%; border-collapse: collapse;'>";
13         echo "<tr class='table-column'><th>Your Invoice Reference</th><th>Amount</th></tr>";
14         foreach ($invoices as $invoice) {
15             echo "<tr class='table-row'><td>" . htmlspecialchars($invoice['Invoice_Ref']) . "</td><td>" . htmlspecialchars($invoice['Amount']) . "</td></tr>";
16         }
17         echo "<tr class='table-column'><td><strong>Total Payment</strong></td><td><strong>" . htmlspecialchars($total_amount) . "</strong></td></tr>";
18         echo "</table>";
19         echo "</div>";
20     } else {
21         echo "<p>Supplier not found.</p>";
22     }
23 } else {
24     echo "<p>No unpaid invoices found for the selected supplier.</p>";
25 }
```

paymentForm.php continued:

```
1      } catch (PDOException $e) {
2          // Display an error message if there was an error fetching the invoices
3          echo "<p>Error: " . htmlspecialchars($e->getMessage()) . "</p>";
4      }
5  ?>
6
7      <!-- New form to execute payment.php -->
8      <div class="form_buttons">
9          <form method="post" action="payment.php" onsubmit="return confirmSubmission();" id="processPaymentForm" class="form">
10             <input type="hidden" name="supplier_id" value=<?php echo isset($supplier_id) ? htmlspecialchars($supplier_id) : ''; ?>>
11             <input type="submit" value="Process Payment" class="formButton">
12         </form>
13     </div>
14   </main>
15 </body>
16 </html>
```

payment.php:

```
1  <!--
2      Student: Gleb Tutubalin C00290944
3      Date:    03.2025
4      Description: This file is responsible for processing payments for suppliers.
5 -->
6 <?php
7 include 'DBconnection.php';
8
9 if ($_SERVER["REQUEST_METHOD"] == "POST") {
10     $supplier_id = $_POST['supplier_id'];
11
12     try {
13         // Getting all unpaid invoices for the selected supplier
14         $stmt = $con->prepare("SELECT `Invoice_Ref`, `Amount`, `Invoice_ID` FROM `Invoice` WHERE `Supplier_ID` = ? AND `Status` = 'ISSUED'");
15         $stmt->execute([$supplier_id]);
16         $invoices = $stmt->fetchAll(PDO::FETCH_ASSOC);
17
18         if (!empty($invoices)) {
19             $total_amount = 0;
20
21             foreach ($invoices as $invoice) {
22                 $total_amount += $invoice['Amount'];
23             }
24
25             // Inserting a new payment record
26     }
```

payment.php continued:

```
1      $stmt = $con->prepare("INSERT INTO `Payment` (`Date_Of_Payment`, `Total_Price`) VALUES (NOW(), ?)");
2      $stmt->execute([$total_amount]);
3
4      // Get the last inserted Payment_ID
5      $payment_id = $con->lastInsertId();
6
7      // Updating the status of the invoices to paid and setting the Payment_ID
8      $stmt = $con->prepare("UPDATE `Invoice` SET `Status` = 'PAID', `Payment_ID` = ? WHERE `Supplier_ID` = ? AND `Status` = 'ISSUED'");
9      $stmt->execute([$payment_id, $supplier_id]);
10
11     // Generating the letter to the supplier
12     $stmt = $con->prepare("SELECT `Name`, `Address` FROM `Supplier` WHERE `Supplier_ID` = ?");
13     $stmt->execute([$supplier_id]);
14     $supplier = $stmt->fetch(PDO::FETCH_ASSOC);
15
16     if ($supplier) {
17         echo "<div style='font-family: Arial, sans-serif; line-height: 1.6;'>";
18         echo "<p style='text-align: right;'>Gerry's Garage,<br>High Street,<br>Carlow<br>" . date("Y-m-d") . "</p>";
19         echo "<p>" . htmlspecialchars($supplier['Name']) . "<br>" . nl2br(htmlspecialchars($supplier['Address'])) . "</p>";
20         echo "<p>Enclosed please find cheque for <strong>total amount</strong> in payment of the following invoices:</p>";
21         echo "<table style='width: 100%; border-collapse: collapse; margin-top: 20px;'>";
22         echo "  <thead>";
23         echo "    <tr style='background-color: #f2f2f2;'>";
24         echo "      <th style='border: 1px solid #ddd; padding: 8px; text-align: left;'>Your Invoice Reference</th>";
25     }
```

payment.php continued:

```
1      echo "<th style='border: 1px solid #ddd; padding: 8px; text-align: left;'>Amount</th>";
2      echo "</tr>";
3      echo "</thead>";
4      echo "<tbody>";
5
6      foreach ($invoices as $invoice) {
7          echo "<tr>";
8          echo "  <td style='border: 1px solid #ddd; padding: 8px;'>" . htmlspecialchars($invoice['Invoice_Ref']) . "</td>";
9          echo "  <td style='border: 1px solid #ddd; padding: 8px; '>€" . number_format($invoice['Amount'], 2)) . "</td>";
10         echo "</tr>";
11
12         echo "<tr style='font-weight: bold;'>";
13         echo "  <td style='border: 1px solid #ddd; padding: 8px;'>Total Payment</td>";
14         echo "  <td style='border: 1px solid #ddd; padding: 8px; '>€" . htmlspecialchars(number_format($total_amount, 2)) . "</td>";
15         echo "</tr>";
16         echo "</tbody>";
17         echo "</table>";
18         echo "<p style='margin-top: 20px;'>Yours sincerely,<br><br>xxxxxxxxxxxxxxxxxxxx<br>Manager</p>";
19         echo "</div>";
20     } else {
21         echo "Supplier not found.";
22     }
23 } else {
24     echo "No unpaid invoices found for the selected supplier.";
```

payment.php continued:

```
1      }
2  } catch (PDOException $e) {
3      echo "Error: " . $e->getMessage();
4  }
5 }
6 ?>
```

Unpaid Invoices Report

unpaidInvoicesReport.php:

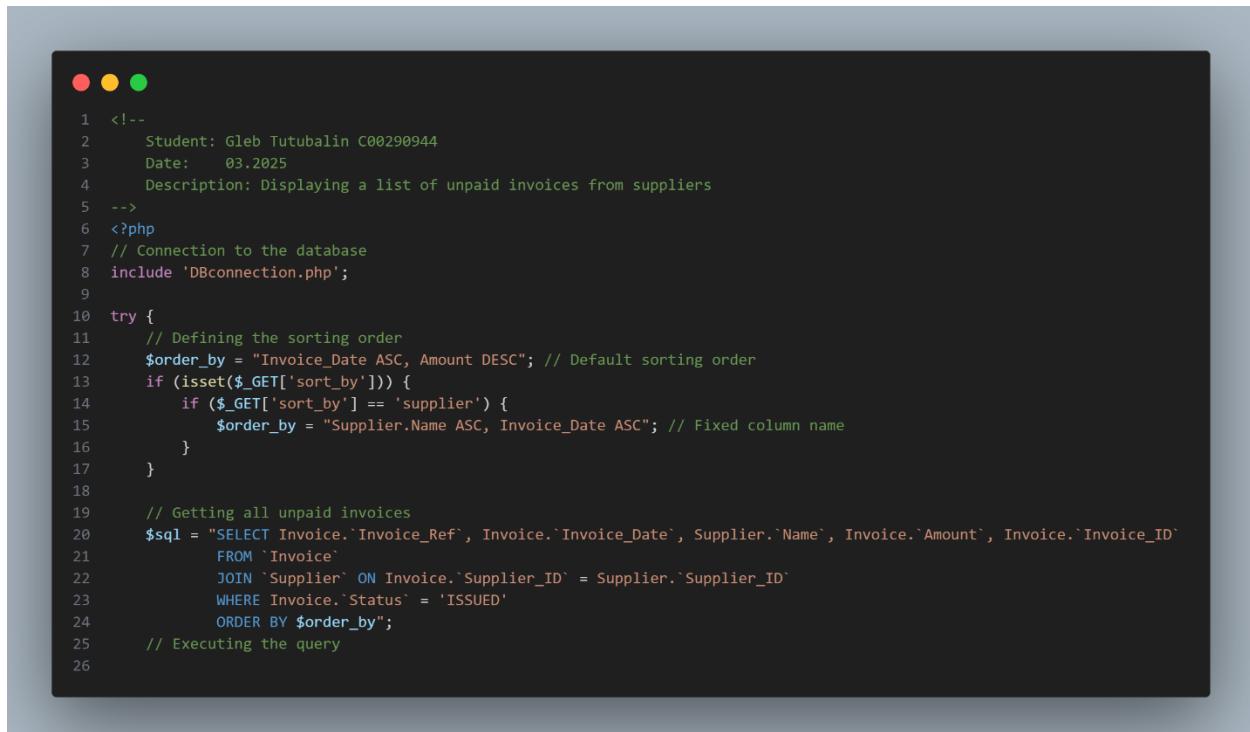
```
1  <!--
2      Student: Gleb Tutubalin C00290944
3      Date:    03.2025
4      Description: Displaying a list of unpaid invoices from suppliers
5  -->
6  <?php
7 // Connection to the database
8 include 'DBconnection.php';
9 ?>
10
11 <!DOCTYPE html>
12 <html lang="en">
13 <head>
14     <meta charset="UTF-8">
15     <meta name="viewport" content="width=device-width, initial-scale=1.0">
16     <title>Unpaid Invoices Report</title>
17     <link rel="stylesheet" href=".Assets/css/AdminDashboard.css">
18     <link rel="stylesheet" href=".Assets/css>ShowSuppliers.css">
19 </head>
20 <body>
21     <main class="delete_supplier_main_container">
22         <section class="suppliers_list_section_container">
23             <div class="form_title">
24                 <h2>List of Unpaid Invoices</h2>
25             
```

unpaidInvoicesReport.php continued:



```
1 </div>
2 <!-- Sorting buttons -->
3 <form method="get" action="unpaidInvoicesReport.php" id="form">
4     <button type="submit" name="sort_by" value="date" ><?php if (!isset($_GET['sort_by']) || $_GET['sort_by'] == 'date') echo 'disabled'; ?>Date</button>
5     <button type="submit" name="sort_by" value="supplier" ><?php if (isset($_GET['sort_by']) && $_GET['sort_by'] == 'supplier') echo 'disabled'; ?>Supplier</button>
6 </form>
7
8 <!-- Table with the invoices -->
9 <div class="supplier_table_container">
10    <table class="table">
11        <tr class="table-column">
12            <th>Supplier's Invoice Reference</th>
13            <th>Date of Invoice</th>
14            <th>Supplier Name</th>
15            <th>Amount of Invoice</th>
16            <th>Our Invoice Number</th>
17        </tr>
18        <!-- Displaying the invoices -->
19        <?php include "unpaidInvoices.php"; ?>
20    </table>
21 </div>
22 </section>
23 </main>
24 </body>
25 </html>
```

unpaidInvoices.php:



```
1 <!--
2     Student: Gleb Tutubalin C00290944
3     Date: 03.2025
4     Description: Displaying a list of unpaid invoices from suppliers
5 -->
6 <?php
7 // Connection to the database
8 include 'DBconnection.php';
9
10 try {
11     // Defining the sorting order
12     $order_by = "Invoice.Date ASC, Amount DESC"; // Default sorting order
13     if (isset($_GET['sort_by'])) {
14         if ($_GET['sort_by'] == 'supplier') {
15             $order_by = "Supplier.Name ASC, Invoice.Date ASC"; // Fixed column name
16         }
17     }
18
19     // Getting all unpaid invoices
20     $sql = "SELECT Invoice.Invoice_Ref, Invoice.Invoice_Date, Supplier.Name, Invoice.Amount, Invoice.Invoice_ID"
21         . " FROM `Invoice`"
22         . " JOIN `Supplier` ON Invoice.Supplier_ID = Supplier.Supplier_ID"
23         . " WHERE Invoice.Status = 'ISSUED'"
24         . " ORDER BY $order_by";
25 // Executing the query
26
```

unpaidInvoices.php continued:

```
1      $result = $con->query($sql);
2
3  if ($result) {
4      // Displaying the table with the invoices
5      if ($result->rowCount() > 0) {
6          while ($row = $result->fetch(PDO::FETCH_ASSOC)) {
7              echo "<tr class='table_row'>";
8              echo "<td>" . htmlspecialchars($row['Invoice_Ref']) . "</td>";
9              echo "<td>" . htmlspecialchars($row['Invoice_Date']) . "</td>";
10             echo "<td>" . htmlspecialchars($row['Name']) . "</td>";
11             echo "<td>" . htmlspecialchars($row['Amount']) . "</td>";
12             echo "<td>" . htmlspecialchars($row['Invoice_ID']) . "</td>";
13             echo "</tr>";
14         }
15     } else {
16         echo "<tr><td colspan='5'>No unpaid invoices found</td></tr>";
17     }
18 } else {
19     echo "An error occurred while fetching data from the database.";
20     error_log("Database Error: " . print_r($con->errorInfo(), true));
21 }
22 } catch (PDOException $e) {
23     echo("Database error: " . $e->getMessage());
24     error_log("Database error: " . $e->getMessage()); // Fixed typo in variable name
25 }
26 ?>
```

Student Muhammad Fahad
Make New Booking Screen
Add Supplier Screen
Delete Supplier Screen
Amen View Supplier Screen

Muhammad Fahad

Make New Booking

Make New Booking Screen :

AI-Disclosure :

- **Booking Creation First:** When creating a new booking, the database auto-generates a booking ID
- **Job Table Requirement:** Need to insert this new booking ID along with selected job type IDs into the Job table
- **Challenge:** The booking ID doesn't exist until after the booking is created
- **Solution Approach:**
 - **Transaction Block:** Wrap both operations in a database transaction
 - **Step 1:** Insert the booking record into Booking table
 - **Get Booking ID:** Use lastInsertId() to retrieve the auto-generated booking ID
 - **Step 2:** Insert records into Job table using the obtained booking ID
 - **Commit:** Only commit if both operations succeed
- _____

Verification of AI Output

Explain the steps you took to verify the accuracy and correctness of the AI-generated output: after demonstration I implement the explanation into actual code while doing this I also use AI for debugging

What Was Fixed?

This should resolve all the errors and ensure secure data insertion.

- It suggests me to use transaction to achieve
- _____

AI-Generated Content Modifications

- Did you modify or refine any AI-generated content before submission? (Check one and provide details.)
Clarified the use of lastInsertId() and transactions for data integrity.

No, I used AI-generated content as is. (Explain why you did not modify it.)

- Yes I did

Acknowledgment & Integrity Statement

By submitting this document, I acknowledge that I have fully disclosed my use of AI technologies in completing this lab sheet.

Make Booking Form (Front-End)

Add New Booking

Customer:
--Select--

Job Type:
--Select--
Headlights Work
cleaning
Engine Tuning

Booking Date:
dd / mm / yyyy

Vehicle Information

Vehicle Model:
Enter vehicle model

Registration Number:
Enter Registration Number

Current Mileage:
Enter Vehicle's Mileage

Details:
Enter Details of booking

This AddNewBooking.php contain the operations for create a booking (Back-End).

```
1 <!--
2     File:          add_booking.php
3     Purpose:       Handles creation of new vehicle bookings and associated jobs
4     Member:        Muhammad Fahad (C00311349)
5     Functionality:
6             Processes booking form submissions
7             Inserts records into Booking and Job tables
8             Validates required job type selections
9     Security:      Uses prepared statements to prevent SQL injection
10    Error Handling: Includes try-catch block for database operations
11    Date:          26/03/2025
12 -->
13 <?php
14 include 'DBconnection.php';
15 // include 'fetch_customer.php' ;
16 if ($_SERVER['REQUEST_METHOD'] === 'POST') {
17     $customer_id = $_POST['customer_id'];
18     $vehicle_model = $_POST['vehicle_model'];
19     $registration_num = $_POST['registration_num'];
20     $mileage = $_POST['mileage'];
21     $form_date = date("y-m-d");
22     $booking_date = $_POST['booking_date'];
23     $details = $_POST['details'];
24
25     // $job_type_ids is the array who stores the multiple ids of job types as we set its
26     // name in form as name=job_type_id[].
27     $job_type_ids = $_POST['job_type_id'];
28
29     // make sure the job_type_ids array should not be empty
30     if (empty($job_type_ids)) {
31         echo "<h2>Please select atleast one job type.</h2>";
32         exit();
33     }
34     try {
35         // begin transaction is sequence in database operations that work as a single unit all
36         // operation in the transaction work as a single unit.
37         // if All the operations/queries successful then commit into the database.
38         // if any of the operation/query fails ,catch the transaction by calling
39         // rollBack() function, to prevent insert any data to database
40         $con->beginTransaction();
41         // this query for booking table
42         $insert = $con->prepare("INSERT INTO `Booking`(`Customer_ID`, `Model`,
43             `Registration_Number`, `Mileage`, `Details`, `Origin_Date`, `Booking_Date`)
44             VALUES (:customer_id, :vehicle_model, :registration_num, :mileage, :details,
45             :form_date, :booking_date) ");
46
```

Continue...

```
1 $insert->bindParam(':customer_id', $customer_id);
2 $insert->bindParam(':vehicle_model', $vehicle_model);
3 $insert->bindParam(':registration_num', $registration_num);
4 $insert->bindParam(':mile_age', $mile_age);
5 $insert->bindParam(':details', $details);
6 $insert->bindParam(':form_date', $form_date);
7 $insert->bindParam(':booking_date', $booking_date);
8
9 if ($insert->execute()) {
10
11     // fetch booking ID from database
12     //the lastInsertId() is the PHP predefined function for retrieve/fetch
13     //the latest Auto generate (value) column from table that is AUTO generated,
14     //this function just return the AUTO generated column/ID/any Auto genrated data.
15     //if the the column data is not AUTO generated then it returns false.
16     $booking_id = $con->lastInsertId();
17
18     // this query for adding job type_id and booking_id to job table
19     $insert_to_job = $con->prepare("INSERT INTO `Job`(`Booking_ID`, `Job_Type_ID`)
20     VALUES (:booking_id,:job_type_id) ;");
21     // execute query for inserting into job table
22     foreach ($job_type_ids as $job_type_id) {
23         $insert_to_job->bindParam(':job_type_id', $job_type_id);
24         $insert_to_job->bindParam(':booking_id', $booking_id);
25         $insert_to_job->execute();
26     }
27     $con->commit();
28     echo "<h2>Data is inserted to database Successfully.</h2>";
29 } else {
30     echo "Error during inserting data into Database!";
31     error_log("Database Error: " . print_r($insert->errorInfo(), true));
32 }
33 } catch (PDOException $e) {
34     $con->rollBack();
35     error_log("Database Error: " . $e->getMessage());
36 }
37 }
38
39 ?>
```

This AddNewBookingForm.php contain the Structure of Make New Booking Form.

```
1 <!--
2     File:          AddNewBookingForm.html
3     Purpose:       Form for creating new vehicle service bookings
4     Member:        Muhammad Fahad (C00311349)
5     Functionality:
6             Customer selection dropdown (dynamic population)
7             Multiple job type selection
8             Vehicle information section
9             Date picker for booking
10            Form validation
11     Dependencies:
12             ShowSuppliers.css for styling
13             fetch.php for dynamic dropdown data
14     Date:          26/03/2025
15 -->
16 <!DOCTYPE html>
17 <html lang="en">
18
19 <head>
20     <meta charset="UTF-8">
21     <meta name="viewport" content="width=device-width, initial-scale=1.0">
22     <title>Add Supplier|GMS</title>
23     <link rel="stylesheet" href=".//Assets/css>ShowSuppliers.css">
24     <style>
25         /* Add this to your CSS file */
26         select.drop_down_list option {
27             color: black !important;
28             font-size: 16px !important;
29         }
30     </style>
31 </head>
32
33 <body>
34     <main class="main_form_container">
35         <div class="div">
36             <div class="form_title">
37                 <h1>Add New Booking</h1>
38             </div>
39             <form action="AddNewBooking.php" method="post" id="form" class="bookingform form">
40
41                 <!-- DropDown For customer -->
42                 <div class="inputMain">
43                     <label for="Customer">Customer:</label>
44                     <select class="drop_down_list customer_select_in_booking"
45                         name="customer_id"
46                         id="Customer"
47                         data_file_link="fetch.php"
48                         data_value="Customer_ID"
49                         data_text="Name" required>
50                         <option value="">---Select Customer---</option>
51                     </select>
52                 </div>
```

continued....

```
1      <!-- DropDownList For job type -->
2      <div class="inputMain">
3          <label for="Job_Type">Job Type:</label>
4          <select class="drop_down_list job_type_select_in_booking"
5              name="job_type_id[]"
6              id="Job_Type"
7              data_file_link="fetch.php"
8              data_value="Job_Type_ID"
9              data_text="Job_Type_Name"
10             multiple required>
11             <option value="">---Select Job Type---</option>
12         </select>
13     </div>
14     <!-- Booking Date -->
15     <div class="inputMain">
16         <label for="booking_date">Booking Date: </label>
17         <input type="date" id="booking_date" name="booking_date" required>
18     </div>
19     <!-- Vehicle Information -->
20     <fieldset class="inputMain">
21         <legend>Vehicle Information</legend>
22         <!-- Model -->
23         <div class="inputMain">
24             <label for="vehicle_model">Vehicle Model:</label>
25             <input type="number" id="vehicle_model"
26                 placeholder="Enter vehicle model" name="vehicle_model" required>
27         </div>
28         <!-- Registration Number -->
29         <div class="inputMain">
30             <label for="reg_num">Registration Number: </label>
31             <input type="text" id="reg_num"
32                 placeholder="Enter Registration Number" name="registration_num" required>
33         </div>
34         <!-- Mileage -->
35         <div class="inputMain">
36             <label for="mile_age">Current Mileage: </label>
37             <input type="number" id="mile_age"
38                 placeholder="Enter Vehicle's Mileage" name="mile_age" required>
39         </div>
40         <!-- Details -->
41         <div class="inputMain">
42             <label for="supplier_address">Details: </label>
43             <textarea type="text" id="supplier_address"
44                 placeholder="Enter Details of booking" name="details"></textarea>
45         </div>
46     </fieldset>
47     <!-- Form Buttons -->
48     <div class="form_buttons">
49         <input type="reset" value="Cancel" class="formButton">
50         <input type="submit" value="Add Booking" class="formButton">
51     </div>
52     <!-- Success/Error Message -->
53     <p id="message"></p>
54   </form>
55 </div>
56 </main>
57 </body>
58 </html>
```

Add Supplier
ADD Supplier Screen

AI-Disclosure;

AI Disclosure Document

Student Name: _____ Muhammad fahad _____
Student ID: _____ C00311349 _____
Course Name & Code: _____ Project SEM 2 CW_KCSOF_B _____

Add New Supplier: _____
Date Submitted: _____ 26/03/2025 _____

AI Assistance Disclosure

AI Technologies Used

Indicate whether you used AI technologies in completing this task:

✓ yes, I used AI tools. (ChatGPT, DeepSeek and Copilot)

- I used copilot for examine the typo mistakes in code and to maintain the structure of code
- DeepSeek, in this screen was stuck on form validation as the form validation did not work when the form displayed in the dashboard but works when run the form individually.
- _____

Prompts Used

- I was stuck on how to add booking ID into job table as have to add booking and job typeID into job Table as well, the problem was the booking was create at that time.

- _____

Nature of Assistance

Indicate how AI technologies contributed to your work by selecting and elaborating on the relevant categories below:

- **Debugging & Error Fixing:** (e.g., AI identified and corrected syntax/logical errors, suggested refactoring, etc.)
 - _____
- **Concept Explanation:** (e.g., AI provided explanations of algorithms, data structures, programming concepts, etc.)
 - _____

Continued....

- **Optimization & Efficiency Improvements:** (e.g., AI suggested more efficient algorithms, reduced complexity, optimized loops, etc.)

Use of AI-Generated Output

Describe how you used the AI-generated output in your work:

- I make some changes in AI-generated output to fulfill the structure of my forms
- _____

Verification of AI Output

Explain the steps you took to verify the accuracy and correctness of the AI-generated output:

What Was Fixed?

This should resolve all the errors and ensure secure data insertion.

- It suggests that use validation on that function which handles the form display role and submission
- _____
- _____

AI-Generated Content Modifications

Did you modify or refine any AI-generated content before submission? (Check one and provide details.)

- AI just validate the form after when the user submit form then I make changes to validation at real time to user

Repeat the above for each task on the lab sheet

Acknowledgment & Integrity Statement

By submitting this document, I acknowledge that I have fully disclosed my use of AI technologies in completing this lab sheet.

Add New Supplier (Front-End)

Add Suppiler

Supplier's Name:
Enter supplier's name

Supplier's Email:
Enter supplier's email

Supplier's Telephone:
Enter supplier's Telephone

Supplier's Address:
Enter supplier's address

Website URL:
Enter website URL, <https://example.com>

Cancel Add Supplier

This AddSupplier.php contain the operations for add supplier (Back-End).

```
1 <!--
2     File:          add_supplier.php
3     Purpose:       Handles creation of new supplier records in the database
4     Member:        Muhammad Fahad (C00311349)
5     Functionality:
6             Processes supplier form submissions
7             Inserts records into Supplier table
8             Implements error handling and logging
9     Security:      Uses prepared statements to prevent SQL injection
10    Error Handling: Includes try-catch block for database operations
11    Database Impact: Inserts records into Supplier table
12    Date:          26/03/2025
13 -->
14 <?php
15 include './DBconnection.php';
16 if ($_SERVER['REQUEST_METHOD'] === 'POST') {
17     $name = $_POST['name'];
18     $email = $_POST['email'];
19     $telephone = $_POST['telephone'];
20     $address = $_POST['address'];
21     $web_url = $_POST['web_url'];
22     try {
23         // insert data into supplier table
24         $insert = $con->prepare("INSERT INTO `Supplier`(`Name`, `Address`, `Telephone`, `Email`, `web_url`)
25         VALUES (:name, :address, :telephone, :email, :web_url)");
26
27         $insert->bindParam(':name', $name);
28         $insert->bindParam(':address', $address);
29         $insert->bindParam(':telephone', $telephone);
30         $insert->bindParam(':email', $email);
31         $insert->bindParam(':web_url', $web_url);
32
33         if ($insert->execute()) {
34             echo "<h2>Data is inserted into database successfully.</h2>";
35         } else {
36             echo "Error occurred while inserting data into database.";
37             error_log("Database Error: " . print_r($insert->errorInfo(), true));
38         }
39     } catch (PDOException $e) {
40         // display error log in case of any error
41         echo ("Database error: " . $e->getMessage());
42         error_log("database error: " . $e->getMessage());
43     }
44 }
45 ?>
```

This AddSupplierForm.php contain the structure of Add Supplier Form

```
1 <!--
2   File:          AddSupplierForm.html
3   Purpose:       Form for adding new suppliers into garage
4   Member:        Muhammad Fahad (C00311349)
5   Functionality:
6     Supplier information collection
7     Form validation (client-side)
8     Responsive design
9     Required field validation
10    Form validation
11  Dependencies:
12    ShowSuppliers.css for styling
13    Admin_dashboard.js for form validation
14  Date:          26/03/2025
15 -->
16 <!DOCTYPE html>
17 <html lang="en">
18 <head>
19   <meta charset="UTF-8">
20   <meta name="viewport" content="width=device-width, initial-scale=1.0">
21   <title>Add Supplier|GMS</title>
22   <link rel="stylesheet" href="./Assets/css>ShowSuppliers.css">
23 </head>
24 <body>
25   <main class="main_form_container">
26     <div class="div">
27       <div class="form_title">
28         <h1>Add Supplier</h1>
29       </div>
30       <form action="AddSupplier.php" method="post" id="form" class="form">
31         <!-- name -->
32         <div class="inputMain">
33           <label for="supplier_name">Supplier's Name: </label>
34           <input type="text" id="supplier_name" placeholder="Enter supplier's name" name="name" required>
35         </div>
36         <!-- Email -->
37         <div class="inputMain">
38           <label for="supplier_email">Supplier's Email: </label>
39           <input type="email" id="supplier_email" placeholder="Enter supplier's email" name="email" required>
40         </div>
41         <!-- telephone-->
42         <div class="inputMain">
43           <label for="supplier_telephone">Supplier's Telephone: </label>
44           <input type="text" id="supplier_telephone" placeholder="Enter supplier's Telephone" name="telephone" required>
45         </div>
46         <!-- Address -->
47         <div class="inputMain">
48           <label for="supplier_address">Supplier's Address: </label>
49           <input type="text" id="supplier_address" placeholder="Enter supplier's address" name="address" required>
50         </div>
51         <!-- Website URL -->
52         <div class="inputMain">
53           <label for="web_url">Website URL: </label>
54           <input type="url" id="web_url" placeholder="Enter website URL, https://example.com" title="Include http:// or https://" name="web_url">
55         </div>
56         <!-- Form Buttons -->
57         <div class="form_buttons">
58           <input type="reset" value="Cancel" class="formButton">
59           <input type="submit" value="Add Supplier" class="formButton">
60         </div>
61         <!-- success /error message will show here -->
62         <p id="message"></p>
63       </form>
64     </div>
65   </main>
66 </body>
67 </html>
```

Delete Supplier

Delete Supplier Screen

AI-Disclosure

AI Disclosure Document

Student Name: _____Muhammad fahad_____

Student ID: _____C00311349_____

Course Name & Code: _____Project SEM 2 CW_KCSOF_B_____

Delete a Supplier Screen: _____

Date Submitted: _____26/03/2025_____

AI Assistance Disclosure

AI Technologies Used

Indicate whether you used AI technologies in completing this task:

✓ NO I didnot used AI tools. (List all AI technologies used, e.g., ChatGPT, GitHub Copilot, Codeium, Google Bard, etc.)

- NO I didnot used AI in this Screen.
 - _____
 - _____
-

Acknowledgment & Integrity Statement

By submitting this document, I acknowledge that I have fully disclosed my use of AI technologies in completing this lab sheet.

Delete Supplier Screen (Front-End)

Delete Supplier

Supplier List:

List of Active Suppliers

Supplier ID	Supplier Name	Address	Telephone	Email Address	Website Address
25	Kyle Wheel House	Shop 98 pearse road	09948747938	kwh@gms.com	https://wheelHouse.com
30	Kasha exterior house	09 classic building shop no 27	0335218474637	Kashaex45@gms.com	https://kashaexterior.ie
33	Anushka Traders	98 civil line road near civic centre	099890678783	anushka78@gms.com	https://anuskatrade.com.ie
34	Jack exhaust house	87 hill view centre shop no 76	00445329876884	jack98@gms.com	https://jackexhausthouse.com
35	Dunnes Traders	Shop 33 city centre galway	087797887787	traders98@hotmail.com	https://dunnestraders.com.ie
37	Moeez Auto Parts	shop no 01 city center galway	08698798478475	moeez23@hotmail.com	https://autoparts.com
40	muhammad fahad	armstrong centre	08786474648	fahadimran038@gmail.com	https://autogarage.ie
50	josh screw house	12 ashborke near city centre dublin	09488588574	screwhouse@hotmail.com	https://joshscREW.com
58	Adam Traders	45 ashbroke road near city center	095785484657	adam45@hotmail.com	https://interiorHouse.ie

This deleteSupplier.php contain the Operation of Delete Screen (Back-End)

```
1  <!--
2      File:          delete_supplier.php
3      Purpose:       Handles safe deletion (soft delete) of supplier records
4      Member:        Muhammad Fahad (C00311349)
5      Functionality:
6          Checks for unpaid invoices before deletion
7          Verifies no open orders exist
8          Implements soft delete via Delete_Flag
9          Provides comprehensive error handling
10         Block deletion if unpaid invoices exist
11         Block deletion if open orders exist
12     Security:      Uses prepared statements to prevent SQL injection
13     Error Handling: Includes try-catch block for database operations
14     Database Impact: Reads from Invoice and Order tables
15             Updates Supplier table (Delete_Flag)
16     Date:          26/03/2025
17 -->
18 <?php
19 include './DBconnection.php';
20 if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['supplier_id'])) {
21     $supplier_id = $_POST['supplier_id'];
22     try {
23         // Check for unpaid invoices
24         $check_invoice = $con->prepare("SELECT COUNT(*) FROM `Invoice`"
25         WHERE `Supplier_ID` = :id AND `Status` = 'ISSUED'");
26         $check_invoice->bindParam(':id', $supplier_id);
27         $check_invoice->execute();
28         $unpaid_invoice = $check_invoice->fetchColumn();
29         // Check for open orders (fixed missing AND)
30         $check_order = $con->prepare("SELECT COUNT(*) FROM `Order`"
31         WHERE `Supplier_ID` = :id AND `Status` = '0'");
32         $check_order->bindParam(':id', $supplier_id);
33         $check_order->execute();
34         $open_order = $check_order->fetchColumn();
35         if ($unpaid_invoice > 0) {
36             echo "<h2>Cannot delete supplier - there are $unpaid_invoice unpaid invoices.</h2>";
37         } elseif ($open_order > 0) {
38             echo "<h2>Cannot delete supplier - there are $open_order open orders.</h2>";
39         } else {
40             // Update delete flag
41             $update = $con->prepare("UPDATE `Supplier` SET `Delete_Flag` = '1' WHERE `Supplier_ID` = :id");
42             $update->bindParam(':id', $supplier_id);
43             if ($update->execute()) {
44                 echo "<h2>Supplier has been successfully flagged for deletion.</h2>";
45             } else {
46                 echo "<h2>Error processing supplier deletion.</h2>";
47                 error_log("Database Error: " . print_r($update->errorInfo(), true));
48             }
49         }
50     } catch (PDOException $e) {
51         // display error logs in case of any errors
52         error_log("Database Error: " . $e->getMessage());
53         echo "<h2>System error occurred. Please try again.</h2>";
54     }
55 } else {
56     echo "<h2>Please select a valid supplier.</h2>";
57 }
```

This deleteSupplierPage.php contain the Strucure of Delete supplier Screen

```
1  <!--
2      File:          deleteSupplierPage.html
3      Purpose:       Form for selecting and deleting supplier records
4      Member:        Muhammad Fahad (C00311349)
5      Functionality:
6              Dynamic supplier dropdown list
7              Delete confirmation workflow
8              Responsive design
9              Required field validation
10             Form validation
11             Integrated with suppliers list display
12     Dependencies:
13             ShowSuppliers.css for styling
14             Admindashboard.css (main layout)
15             fetch.php (dynamic dropdown data)
16             deleteSupplier.php (form processor)
17             suppliersList.php (supplier table display)
18             Admin_dasboard.JS for form validarion
19     Date:          26/03/2025
20 -->
21 <!DOCTYPE html>
22 <html lang="en">
23 <head>
24     <meta charset="UTF-8">
25     <meta name="viewport" content="width=device-width, initial-scale=1.0">
26     <title>Delete Supplier</title>
27     <link rel="stylesheet" href="./Assets/css/Admindashboard.css">
28     <link rel="stylesheet" href="./Assets/css>ShowSuppliers.css">
29 </head>
30 <style>
31     select.drop_down_list option {
32         color: black !important;
33         font-size: 16px !important;
34     }
35 </style>
```

Continued....

```
1 <body>
2     <main class="main_form_container">
3         <section class="div">
4             <div class="form_title">
5                 <h1>Delete Supplier</h1>
6             </div>
7             <form action="./deleteSupplier.php" method="post" id="form" class="form bookingform">
8                 <!-- <section class="header_section"> -->
9                 <!-- <header class="delete_supplierheader"> -->
10                <div class="form_title supplier_delete_list_container">
11                    <div class="sup_list">
12                        <label for="Supplier">Supplier List:</label>
13                        <select
14                            class="drop_down_list"
15                            name="supplier_id"
16                            id="Supplier"
17                            data_file_link="fetch.php"
18                            data_value="Supplier_ID"
19                            data_text="Name">
20                            <option value="">---Select Supplier---</option>
21                        </select>
22                    </div>
23                    <div class="sup_but">
24                        <input class="formButton" type="submit" value="Delete Supplier">
25                    </div>
26                </div>
27                <!-- Success/Error Message -->
28                <p id="message"></p>
29            <!-- </header> -->
30            <!-- </section> -->
31            <?php
32                include './suppliersList.php';
33            ?>
34            </form>
35        </section>
36    </main>
37 </body>
38 </html>
```

Amend / View Supplier
Amend View Supplier Screen
AI-Disclosure

AI Disclosure Document

Student Name: _____ Muhammad fahad _____
Student ID: _____ C00311349 _____
Course Name & Code: _____ Project SEM 2 CW_KCSOF_B _____

Amend View Supplier _____
Date Submitted: _____ 26/03/2025 _____

AI Assistance Disclosure

AI Technologies Used

Indicate whether you used AI technologies in completing this task:

✓ yes, I used AI tools. (ChatGPT, DeepSeek and Copilot)

- I used copilot for examine the typo mistakes in code and to maintain the structure of code
- DeepSeek, ChatGPT , as stuck on show th supplier data in the form when the user select supplier from dropdown list.
- _____

Prompts Used

- i have form called amend view supplier with supplier Dropdown (contains real data from database) , when the user select the supplier from the dropdown then the supplier details should be display in the form. how can i achieve this ?
-

Nature of Assistance

Indicate how AI technologies contributed to your work by selecting and elaborating on the relevant categories below:

- **Debugging & Error Fixing:** (e.g., AI identified and corrected syntax/logical errors, suggested refactoring, etc.)
 - _____
- **Concept Explanation:** (e.g., AI provided explanations of algorithms, data structures, programming concepts, etc.)
 - _____

Continued...

- **Optimization & Efficiency Improvements:** (e.g., AI suggested more efficient algorithms, reduced complexity, optimized loops, etc.)

Use of AI-Generated Output

Describe how you used the AI-generated output in your work:

- I make some changes in AI-generated output to fulful the structure of my forms
- _____

Verification of AI Output

Explain the steps you took to verify the accuracy and correctness of the AI-generated output:

What Was Fixed?

This should resolve all the errors and ensure secure data insertion.

- fetchSuppliers.php didn't call \$select->execute(); before fetchAll().
Fix: Added \$select->execute(); before fetching data.
- **Bug:** Dropdown options were being **appended multiple times** on every fetch.
- **Fix:** Added dropdown.innerHTML = "" before adding new options.
- _____
- _____

AI-Generated Content Modifications

Did you modify or refine any AI-generated content before submission? (Check one and provide details.)

Original Issue: AI-generated PHP scripts didn't encode data properly.

Fix Applied: i encode the json responce into array then the data from databse fetches as array ;

Repeat the above for each task on the lab sheet

Acknowledgment & Integrity Statement

Amend View Supplier Screen (Front-End)

Amend View Supplier

Supplier List:

---Select---

Supplier's ID:

Supplier's ID

Supplier's Name:

Enter supplier's name

Supplier's Email:

Enter supplier's email

Supplier's Telephone:

Enter supplier's Telephone

Supplier's Address:

Enter supplier's address

Website URL:

Enter website URL, https://example.com

Cancel

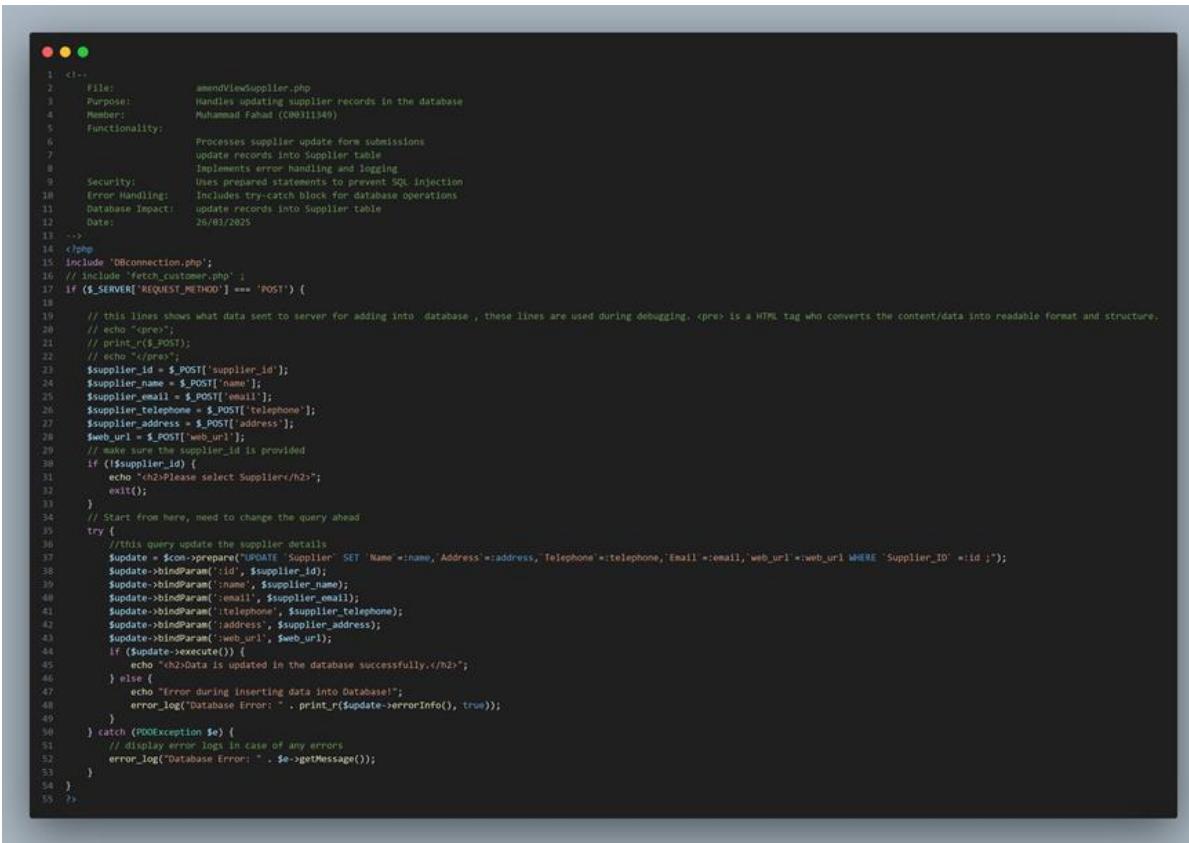
Update Info

Continued....

List of Active Suppliers

Supplier ID	Supplier Name	Address	Telephone	Email Address	Website Address
25	Kyle Wheel House	Shop 98 pearse road	09948747938	kwh@gms.com	https://wheelHouse.com
30	Kasha exterior house	09 classic building shop no 27	0335218474637	Kashaex45@gms.com	https://kashaexterior.ie
33	Anushka Traders	98 civl line road near civic centre	099890678783	anushka78@gms.com	https://anuskatrade.com.ie
34	Jack exhaust house	87 hill view centre shop no 76	00445329876884	jack98@gms.com	https://jackexhausthouse.com
35	Dunnes Traders	Shop 33 city centre galway	087797887787	traders98@hotmail.com	https://dunnestraders.com.ie
37	Moeez Auto Parts	shop no 01 city center galway	08698798478475	moeez23@hotmail.com	https://autoparts.com
40	muhammad fahad	armstrong centre	08786474648	fahadimran038@gmail.com	https://autogarage.ie
50	josh screw house	12 ashborke near city centre dublin	09488588574	screwhouse@hotmail.com	https://joshscREW.com
58	Adam Traders	45 ashbroke road near city center	095785484657	adam45@hotmail.com	https://interiorHouse.ie

This amendViewSupplier.php contain the Operation of Amend View Supplier (Back-End)



```
1 <!--
2   File:           amendViewSupplier.php
3   Purpose:        Handles updating supplier records in the database
4   Member:         Muhammad Fahad (C08311349)
5   Functionality:
6     Processes supplier update form submissions
7     update records into Supplier table
8     Implements error handling and logging
9   Security:
10    Uses prepared statements to prevent SQL injection
11    Error Handling: Includes try-catch block for database operations
12    Database Impact: update records into Supplier table
13    Date:          26/03/2025
14 -->
15 </php>
16 include 'DBconnection.php';
17 // include 'fetch_customer.php';
18 if ($_SERVER['REQUEST_METHOD'] === 'POST') {
19   // this lines shows what data sent to server for adding into database , these lines are used during debugging. <pre> is a HTML tag who converts the content/data into readable format and structure.
20   // echo "<pre>";
21   // print_r($_POST);
22   // echo "</pre>";
23   $supplier_id = $_POST['supplier_id'];
24   $supplier_name = $_POST['name'];
25   $supplier_email = $_POST['email'];
26   $supplier_telephone = $_POST['telephone'];
27   $supplier_address = $_POST['address'];
28   $web_url = $_POST['web_url'];
29   // make sure the supplier_id is provided
30   if (!$supplier_id) {
31     echo "<h2>Please select Supplier</h2>";
32     exit();
33   }
34   // Start from here, need to change the query ahead
35   try {
36     //this query update the supplier details
37     $update = $con->prepare("UPDATE `Supplier` SET `Name`=:name, `Address`=:address, `Telephone`=:telephone, `Email`=:email, `Web_url`=:web_url WHERE `Supplier_ID` =:id ");
38     $update->bindParam(':id', $supplier_id);
39     $update->bindParam(':name', $supplier_name);
40     $update->bindParam(':email', $supplier_email);
41     $update->bindParam(':telephone', $supplier_telephone);
42     $update->bindParam(':address', $supplier_address);
43     $update->bindParam(':web_url', $web_url);
44     if ($update->execute()) {
45       echo "<h2>Data is updated in the database successfully.</h2>";
46     } else {
47       echo "Error during inserting data into Database!";
48       error_log("Database Error: " . print_r($update->errorInfo(), true));
49     }
50   } catch (PDOException $e) {
51     // display error logs in case of any errors
52     error_log("Database Error: " . $e->getMessage());
53   }
54 }
55 ?>
```

This amendViewSupplierPage.php contain the structure of Amend View Supplier Form.

```
1  <!--
2      File:          AddSupplierForm.html
3      Purpose:       Form for viewing and updating supplier information
4      Member:        Muhammad Fahad (C00311349)
5      Functionality:
6          Dynamic supplier selection dropdown via javascript,AJAX
7          Form auto-population when supplier selected
8          Integrated suppliers list display(suppliersList.php)
9          Form validation (client-side)
10         Responsive design
11         Required field validation
12     Dependencies:
13         ShowSuppliers.css for styling
14         Admindashboard.css (main layout)
15         fetch.php (dynamic dropdown data)
16         amendViewSupplier.php (form processor)
17         Admin_dashboard.JS for form validarion
18     Date:          26/03/2025
19 -->
20 <!DOCTYPE html>
21 <html lang="en">
22
23 <head>
24     <meta charset="UTF-8">
25     <meta name="viewport" content="width=device-width, initial-scale=1.0">
26     <title>Delete Supplier</title>
27     <link rel="stylesheet" href=".Assets/css/Admindashboard.css">
28     <link rel="stylesheet" href=".Assets/css>ShowSuppliers.css">
29 </head>
30 <style>
31     /* Add this to your CSS file */
32     select.drop_down_list option {
33         color: black !important;
34         font-size: 16px !important;
35     }
36 </style>
37
38 <body>
39     <main class="delete_supplier_main_conatainer">
40         <!-- <section class="suppliers_list_section_container"> -->
41         <div class="div">
42             <div class="form_title">
43                 <h1>Amend View Supplier</h1>
44             </div>
45             <form action=".amendViewSupplier.php" method="post" id="form" class="form">
46                 <!-- Supplier Selection Dropdown -->
47                 <div class="inputMain">
48                     <label for="Supplier">Suppleir List:</label>
49                     <select
50                         data-amend-supplier-dropdown="Supplier"
51                         class="drop_down_list"
52                         name="supplier_id"
53                         id="Supplier"
54                         data_file_link="fetch.php"
55                         data_value="Supplier_ID"
56                         data_text="Name" required>
57                         <option value="">---Select Supplier---</option>
58                     </select>
59                 </div>
```

Continued....

```
1  <!-- Supplier ID -->
2      <div class="inputMain">
3          <label for="id">Supplier's ID: </label>
4          <input type="text" id="id" placeholder="Supplier's ID"
5              name="supplier_id" readonly>
6      </div>
7  <!-- Supplier Name -->
8      <div class="inputMain">
9          <label for="supplier_name">Supplier's Name: </label>
10         <input type="text" id="supplier_name"
11             placeholder="Enter supplier's name" name="name" required>
12     </div>
13  <!-- Supplier Email -->
14      <div class="inputMain">
15          <label for="supplier_email">Supplier's Email: </label>
16          <input type="email" id="supplier_email"
17              placeholder="Enter supplier's email" name="email" required>
18      </div>
19  <!-- Supplier Telephone -->
20      <div class="inputMain">
21          <label for="supplier_telephone">Supplier's Telephone: </label>
22          <input type="text" id="supplier_telephone"
23              placeholder="Enter supplier's Telephone" name="telephone" required>
24      </div>
25  <!-- Supplier Address -->
26      <div class="inputMain">
27          <label for="supplier_address">Supplier's Address: </label>
28          <input type="text" id="supplier_address"
29              placeholder="Enter supplier's address" name="address" required>
30      </div>
31  <!-- Website URL -->
32      <div class="inputMain">
33          <label for="web_url">Website URL: </label>
34          <input type="url" id="web_url"
35              placeholder="Enter website URL, https://example.com" name="web_url" required>
36      </div>
37  <!-- Form Buttons -->
38      <div class="form_buttons">
39          <input type="reset" value="Cancel" class="formButton">
40          <input type="submit" value="Update Info" class="formButton">
41      </div>
42  <!-- Success/Error Message -->
43      <p id="message"></p>
44  </form>
45  <?php
46      // this file contain the supplier list table
47      include './suppliersList.php';
48  ?>
49      </div>
50  </main>
51  </body>
52
53  </html>
```

This fetchsupplier.php fetches the selected supplier details from database

```
1 <!--
2   File:          fetch_supplier_details.php
3   Purpose:       fetch supplier data from database
4   Member:        Muhammad Fahad (C00311349)
5   Functionality:
6           Fetches active records from supplier table
7           Accepts POST requests with supplier_id
8   Security:      Uses prepared statements to prevent SQL injection
9   Error Handling: Includes try-catch block for database operations
10  Database Impact: Reads from supplier tables
11  Date:          26/03/2025
12 -->
13 <?php
14 include 'DBconnection.php';
15 if ($_SERVER['REQUEST_METHOD'] === 'POST') {
16     // echo "<pre>";
17     // print_r($_POST);
18     // echo "</pre>";
19     // check if the Supplier is select or not
20     if (!isset($_POST['supplier_id'])) {
21         header('Content-Type: application/json');
22         echo json_encode(["error" => "Supplier ID is required"]);
23         exit;
24     }
25     $supplier_id = $_POST['supplier_id'];
26     try {
27         // this is SQL Query for fetch the Supplier details , here i used
28         //named placeholder instead of direct insert the variable
29         $fetch_supplier = $con->prepare("SELECT `Supplier_ID`, `Name`, `Address`,
30                                         `Telephone`, `Email`, `web_url` FROM `Supplier` WHERE `Supplier_ID` =:supplier_id");
31         $fetch_supplier->bindParam(':supplier_id', $supplier_id);
32         $fetch_supplier->execute();
33         $row = $fetch_supplier->fetch(PDO::FETCH_ASSOC);
34
35         header('Content-Type: application/json');
36         if (!$row) {
37             // Set the content type to JSON
38             echo json_encode(["error" => "No record found"]);
39         } else {
40             echo json_encode($row);
41         }
42     }
43     // its a catch clock to display error with error details ,
44     catch (PDOException $e) {
45         header('Content-Type: application/json');
46         echo json_encode(["error" => "Database Error" . $e->getMessage()]);
47     }
48 }
49 // if the action method is not POST then it shows below line
50 else {
51     header('Content-Type: application/json');
52     echo json_encode(["error" => "invalid action method"]);
53 }
54 ?>
```