

C:\Users\Public\GCC_Monitoring\gcc_monitoring\ui\layout.py

```
1  from nicegui import ui
2  from core.auth import current_user, logout
3
4  dark_mode = True
5
6
7  def layout(title: str = "GCC Monitoring System"):
8      user = current_user()
9
10     # Force dark mode (prevents white text on white cards)
11     ui.run_javascript('document.body.classList.remove("light")')
12
13     ui.add_head_html("""
14     <style>
15         :root {
16             --bg: #07150f;
17             --card: #0b231a;
18             --text: #f0f0f0;
19             --muted: #aaaaaa;
20             --accent: #16a34a;
21
22             /* NEW: line colors (subtle but visible in dark theme) */
23             --line: rgba(255,255,255,0.14);
24             --line-strong: rgba(255,255,255,0.22);
25             --border: rgba(255,255,255,0.18);
26
27         }
28
29         body {
30             background: var(--bg);
31             color: var(--text);
32         }
33
34         .gcc-card, .card {
35             background: var(--card);
36             border: 1px solid var(--line);
37             border-radius: 12px;
38             box-shadow: 0 4px 12px rgba(0,0,0,0.12);
39             color: var(--text) !important;
40         }
41
42         .gcc-muted { color: var(--muted) !important; }
43
44         /* NEW: universal separators */
45         .gcc-sep { border-top: 1px solid var(--line) !important; }
46         .gcc-sep-strong { border-top: 1px solid var(--line-strong) !important; }
47
48         /* NEW: soft, readable table grid lines everywhere */
```

```
49     .gcc-soft-grid .q-table__middle td,
50     .gcc-soft-grid .q-table__middle th {
51         border-color: var(--line) !important;
52     }
53     .gcc-soft-grid thead tr th {
54         border-bottom-color: var(--line-strong) !important;
55     }
56
57     .menu-link {
58         display: block;
59         padding: 8px 12px;
60         color: var(--text);
61         text-decoration: none;
62         border-radius: 8px;
63     }
64     .menu-link:hover {
65         background: var(--accent);
66         color: white;
67     }
68
69     /* ===== FIX: Make dialogs readable in dark mode ===== */
70
71     /* The dialog card itself */
72     .q-dialog .q-card,
73     .q-dialog .q-dialog__inner > div {
74         background: var(--card) !important;
75         color: var(--text) !important;
76
77         /* border: 1px solid var(--border) !important;*/
78
79         border: 1px solid var(--line) !important;
80
81     }
82
83     /* Dialog title/labels/text */
84     .q-dialog .q-card * {
85         color: var(--text) !important;
86     }
87
88     /* Inputs in dialog: label + typed text */
89     .q-dialog .q-field__label,
90     .q-dialog .q-field__native,
91     .q-dialog input,
92     .q-dialog textarea {
93         color: var(--text) !important;
94     }
95
96     /* Input borders / underline so you can see fields */
97     .q-dialog .q-field__control::before,
98     .q-dialog .q-field__control::after {
```

```
99     border-color: var(--border) !important;
100 }
101
102 /* Placeholder text */
103 .q-dialog input::placeholder,
104 .q-dialog textarea::placeholder {
105     color: rgba(240,240,240,0.55) !important;
106 }
107
108 /* Checkbox label */
109 .q-dialog .q-checkbox__label {
110     color: var(--text) !important;
111 }
112
113 /* ===== FIX: Make ALL inputs/selects readable in dark mode (not only dialogs) ===== */
114
115 /* input typed text */
116 .q-field__native,
117 .q-field__input,
118 .q-select__input,
119 .q-input input,
120 .q-input textarea,
121 input,
122 textarea {
123     color: var(--text) !important;
124 }
125
126 /* labels + helper text */
127 .q-field__label,
128 .q-field__bottom {
129     color: var(--muted) !important;
130 }
131
132 /* field background so you can see the box */
133 .q-field__control {
134     background: rgba(255,255,255,0.04) !important;
135 }
136
137 /* field borders / underline */
138 .q-field__control::before,
139 .q-field__control::after {
140     border-color: var(--line-strong) !important;
141 }
142
143 /* placeholder */
144 input::placeholder,
145 textarea::placeholder {
146     color: rgba(240,240,240,0.55) !important;
147 }
148
```

```

149  /* dropdown menu for selects */
150  .q-menu,
151  .q-list {
152      background: var(--card) !important;
153      color: var(--text) !important;
154  }
155  .q-item,
156  .q-item__label {
157      color: var(--text) !important;
158  }
159
160
161
162  </style>
163  """)
164
165  with ui.left_drawer(value=True).props('bordered width=260').classes('p-4') as drawer:
166      drawer.style('background: var(--card);')
167
168      ui.label('GCC Monitoring').classes('text-xl font-bold')
169      ui.label('Dashboard').classes('text-xs gcc-muted -mt-1')
170
171      ui.separator().classes('my-4 gcc-sep-strong')
172
173      ui.link('🏠 Dashboard', '/').classes('menu-link')
174      ui.link('📦 Units', '/equipment').classes('menu-link')
175      ui.link('🔔 Alerts', '/alerts').classes('menu-link')
176      ui.link('📊 Trends', '/trends').classes('menu-link')
177
178      ui.separator().classes('my-4 gcc-sep-strong')
179
180      ui.label('Administration').classes('text-sm font-bold gcc-muted mt-3')
181
182      ui.link('👥 Clients', '/clients').classes('menu-link')
183      ui.link('📍 Locations', '/locations').classes('menu-link')
184      ui.link('🔧 Admin', '/admin').classes('menu-link')
185
186      with ui.header().classes('items-center justify-between px-4 py-3'):
187          with ui.row().classes('items-center gap-3'):
188              ui.button('☰', on_click=drawer.toggle).props('flat dense').style('font-size:
24px; color: var(--text);')
189              ui.label(title).classes('text-xl font-bold')
190
191          with ui.row().classes('items-center gap-4'):
192              if user:
193                  ui.label(user.get('email', 'Guest')).classes('text-sm gcc-muted')
194
195              def toggle_theme():
196                  global dark_mode
197                  dark_mode = not dark_mode

```

```
198         if dark_mode:
199             ui.run_javascript('document.body.classList.remove("light")')
200         else:
201             ui.run_javascript('document.body.classList.add("light")')
202
203         ui.button('🌙 / ☀️', on_click=toggle_theme).props('flat dense').style('font-size: 20px;')
204
205         ui.button('Logout', on_click=lambda: (logout(),
206 ui.navigate.to('/login'))).props('flat')
207
208     return ui.column().classes('p-5 w-full max-w-[1300px] mx-auto gap-6')
```