index.js

```
1   /**
2    * index.js - Main Server Entry Point
3    */
4
5   // ------------------------------ START NEW CODE -----------------------
6   // Import express library.
7   // Creating object for express.
8   // The server is created here, it's in the computers memory.
9   // However, it's "sleeping" yet until is called.
10  // SOURCE: https://expressjs.com/en/4x/api.html
11  const express = require('express');
12  const app = express();
13
14  // Selecting port 3000.
15  const port = 3000;
16
17  // Import sqlite3 library.
18  const sqlite3 = require('sqlite3').verbose();
19
20  // Middlewares.
21  // Middleware functions are functions that have access to the request object (req),
22  // the response object (res), and the next middleware function in the application's
23  // request-response cycle.
24  // Summarizing: middlewere are functions that are executed before the route (between
    requisition and route).
25  /*
26  Exemple:
27  1. The person types the URL and presses Enter
28  2. The browser sends the request
29  3. The server receives the request
30  4. The middleware is executed
31  5. The route is executed
32  6. The server sends the response
33  7. The browser receives the respons
34  */
35
36  app.use(express.urlencoded({ extended: true })); // for parsing application/x-www-form-
    urlencoded
37  app.set('view engine', 'ejs');
38  app.use(express.static(__dirname + '/public'));
39
40  // Database Connection
41  global.db = new sqlite3.Database('./database.db',function(err){
42      if(err){
43          console.error(err);
44          process.exit(1);
45      } else {
46          console.log("Database connected");
47          global.db.run("PRAGMA foreign_keys=ON");
48      }
49  });
50
```

```
51   // ------------------------ Import Routes --------------------------
52   const mainRoutes = require('./routes/main');
53   const organiserRoutes = require('./routes/organiser');
54   const attendeeRoutes = require('./routes/attendee');
55
56   // Use Routes. It's here that is defined the name from the URL.
57   // The URL is always: index + inside route.
58   // E.g. route organiser, inside organiser there's add-event ->
     http://localhost:3000/organiser/add-event.
59   app.use('/', mainRoutes);
60   app.use('/organiser', organiserRoutes);
61   app.use('/attendee', attendeeRoutes);
62
63
64   // Turn on server. "Opens" door 3000.
65   // Literally listen a port.
66   app.listen(port, () => {
67       console.log(`App listening on port ${port}`);
68   });
69   // ---------------------------- END NEW CODE ----------------------
```

routes\main.js

```
 1  /* ----------------START NEW CODE ---------------------------- */
 2  const express = require("express");
 3
 4  // Create a router.
 5  // App is the entire application, therefore we need to create a small piece of it.
 6  // Create a router, which unite the routes.
 7  // It must be connected with the app to work properly.
 8  const router = express.Router();
 9
10  /**
11   * @route GET /
12   * @purpose Main landing page. Just links to other areas.
13   * @inputs None
14   * @outputs Renders 'index.ejs' with settings.
15   */
16
17  // Get data.
18  // Request and response.
19  // Response is what to send.
20  // Request is what is received.
21  router.get("/", (req, res) => {
22      global.db.get("SELECT * FROM settings WHERE id = 1", (err, settings) => {
23          if (err) console.error(err);
24          res.render("index", { settings: settings || { site_title: 'Event Manager',
    author_name: 'Admin' } });
25      });
26  });
27
28  module.exports = router;
29  /* ---------------- END NEW CODE ---------------------------- */
```

routes\attendee.js

```
1   /* ----------------START NEW CODE ---------------------------- */
2   const express = require("express");
3   const router = express.Router();
4
5   /**
6    * @route GET /attendee
7    * @purpose List published events sorted by date.
8    */
9   router.get("/", (req, res) => {
10      const search = req.query.search || "";
11
12      const sql = `SELECT * FROM events
13                  WHERE status='published' AND title LIKE ?
14                  ORDER BY event_date ASC`;
15
16      global.db.all(sql, [`%${search}%`], (err, events) => {
17          if (err) {
18              console.error(err);
19              return res.send("Database error");
20          }
21          res.render("attendee-home", { events, search });
22      });
23  });
24
25  /**
26   * @route GET /attendee/event/:id
27   * @purpose Display event details and booking form.
28   */
29  router.get("/event/:id", (req, res) => {
30      global.db.get("SELECT * FROM events WHERE id = ? AND status='published'",
    [req.params.id], (err, event) => {
31          if (err || !event) return res.send("Event not found or not published.");
32          res.render("attendee-event", { event });
33      });
34  });
35
36  /**
37   * @route POST /attendee/book/:id
38   * @purpose Handle ticket booking and update database quantities.
39   */
40  router.post("/book/:id", (req, res) => {
41      const eventId = req.params.id;
42
43      const numFull = parseInt(req.body.num_full) || 0;
44      const numConc = parseInt(req.body.num_conc) || 0;
45      const attendeeName = req.body.attendee_name;
46
47      if (numFull === 0 && numConc === 0) {
48          return res.send("Please select at least one ticket.");
49      }
50
51      global.db.get("SELECT * FROM events WHERE id=?", [eventId], (err, event) => {
```

```
52            if (event.full_price_tickets < numFull || event.concession_tickets < numConc) {
53                return res.send("Error: Not enough tickets available.");
54            }
55
56            global.db.run(`UPDATE events SET full_price_tickets = full_price_tickets - ?,
   concession_tickets = concession_tickets - ? WHERE id = ?`,
57                [numFull, numConc, eventId], (err) => {
58                    if (err) return res.send("Error updating tickets");
59
60                    global.db.run(`INSERT INTO bookings (event_id, attendee_name,
   num_full_tickets, num_concession_tickets) VALUES (?, ?, ?, ?)`,
61                        [eventId, attendeeName, numFull, numConc], (err) => {
62                            res.redirect("/attendee");
63                        });
64                });
65        });
66  });
67
68  router.get("/my-bookings", (req, res) => {
69      const sql = `
70          SELECT bookings.*, events.title, events.event_date
71          FROM bookings
72          JOIN events ON bookings.event_id = events.id
73          ORDER BY bookings.id DESC`;
74
75      global.db.all(sql, [], (err, bookings) => {
76          if (err) {
77              return res.send("Erro ao carregar reservas.");
78          }
79          res.render("my-bookings", { bookings });
80      });
81  });
82
83  module.exports = router;
84  /* ----------------END NEW CODE ----------------------------- */
```

routes\organiser.js

```
1   /* ----------------START NEW CODE ---------------------------- */
2   const express = require("express");
3   const router = express.Router();
4
5   /**
6    * @route GET /organiser
7    * @purpose Display dashboard with separate lists for Drafts and Published events.
8    * @inputs None
9    * @outputs Renders 'organiser.ejs' with settings, publishedEvents, draftEvents.
10   */
11  router.get("/", (req, res) => {
12      global.db.get("SELECT * FROM settings WHERE id = 1", (err, settings) => {
13          global.db.all("SELECT * FROM events ORDER BY created_at DESC", (err, events) => {
14              const publishedEvents = events.filter(e => e.status === 'published');
15              const draftEvents = events.filter(e => e.status === 'draft');
16
17              // Note: We pass separate lists to the view
18              res.render("organiser", { settings, publishedEvents, draftEvents });
19          });
20      });
21  });
22
23  /**
24   * @route GET /organiser/settings
25   * @purpose Show site settings form.
26   * @inputs None
27   * @outputs Renders 'settings.ejs'
28   */
29  router.get("/settings", (req, res) => {
30      global.db.get("SELECT * FROM settings WHERE id = 1", (err, settings) => {
31          res.render("settings", { settings });
32      });
33  });
34
35  /**
36   * @route POST /organiser/settings
37   * @purpose Update site configuration.
38   * @inputs Form data: site_title, author_name
39   * @outputs Redirects to /organiser
40   */
41  router.post("/settings", (req, res) => {
42      const { site_title, author_name } = req.body;
43      global.db.run("UPDATE settings SET site_title = ?, author_name = ? WHERE id = 1",
44          [site_title, author_name], (err) => res.redirect("/organiser"));
45  });
46
47  /**
48   * @route POST /organiser/add-event
49   * @purpose Create a blank draft event and redirect to edit page.
50   * @inputs None
51   * @outputs Redirects to /organiser/edit-event/:id
```

```javascript
  */
router.post("/add-event", (req, res) => {
    // Pegamos os dados do formulário
    const {
        title,
        description,
        content, // Este é o nome que está no seu HTML
        event_date,
        full_price_tickets,
        full_price_cost,
        concession_tickets,
        concession_cost
    } = req.body;

    const sql = `INSERT INTO events (
        title,
        description,
        event_date,
        status,
        full_price_tickets,
        full_price_cost,
        concession_tickets,
        concession_cost
    ) VALUES (?, ?, ?, 'draft', ?, ?, ?, ?)`;


    global.db.run(sql, [
        title,
        content,
        event_date,
        full_price_tickets || 0,
        full_price_cost || 0,
        concession_tickets || 0,
        concession_cost || 0
    ], (err) => {
        if (err) {
            console.error(err);
            return res.send("Error creating event: " + err.message);
        }
        res.redirect("/organiser");
    });
});

/**
 * @route GET /organiser/edit-event/:id
 * @purpose Show form to edit event details.
 * @inputs URL param: id
 * @outputs Renders 'edit-event.ejs'
 */
router.get("/edit-event/:id", (req, res) => {
    global.db.get("SELECT * FROM events WHERE id = ?", [req.params.id], (err, event) => {
        res.render("edit-event", { event });
    });
});
```

```
106
107    /**
108     * @route POST /organiser/edit-event/:id
109     * @purpose Update all event fields.
110     * @inputs Form data (title, description, dates, ticket info)
111     * @outputs Redirects to /organiser
112     */
113    router.post("/edit-event/:id", (req, res) => {
114        const { title, description, event_date, full_price_tickets, full_price_cost,
       concession_tickets, concession_cost } = req.body;
115        const sql = `UPDATE events SET title=?, description=?, event_date=?,
116                     full_price_tickets=?, full_price_cost=?, concession_tickets=?,
       concession_cost=?,
117                     last_modified=CURRENT_TIMESTAMP WHERE id=?`;
118        global.db.run(sql, [title, description, event_date, full_price_tickets,
       full_price_cost, concession_tickets, concession_cost, req.params.id],
119            (err) => res.redirect("/organiser"));
120    });
121
122    /**
123     * @route POST /organiser/publish/:id
124     * @purpose Change status to 'published' and set timestamp.
125     * @inputs URL param: id
126     * @outputs Redirects to /organiser
127     */
128    router.post("/publish/:id", (req, res) => {
129        global.db.run("UPDATE events SET status='published', published_at=CURRENT_TIMESTAMP
       WHERE id=?",
130            [req.params.id], (err) => res.redirect("/organiser"));
131    });
132
133    /**
134     * @route POST /organiser/delete/:id
135     * @purpose Delete an event.
136     * @inputs URL param: id
137     * @outputs Redirects to /organiser
138     */
139    router.post("/delete/:id", (req, res) => {
140        global.db.run("DELETE FROM events WHERE id=?", [req.params.id], (err) =>
       res.redirect("/organiser"));
141    });
142
143    module.exports = router;
144
145    /* ---------------- END NEW CODE ---------------------------- */
```

views\index.ejs

```
1   <!-- --------------------- START NEW CODE --------------------->
2   <!DOCTYPE html>
3   <html lang="en">
4   <head>
5       <meta charset="UTF-8">
6       <title><%= settings.site_title %></title>
7       <link rel="stylesheet" href="/main.css">
8   </head>
9   <body>
10      <header>
11          <h1>Welcome to <%= settings.site_title %></h1>
12          <p>Managed by: <%= settings.author_name %></p>
13      </header>
14
15      <div class="nav-box">
16          <p>Are you an Organiser?</p>
17          <a href="/organiser">Go to Organiser Dashboard</a>
18      </div>
19
20      <div class="nav-box">
21          <p>Are you an Attendee?</p>
22          <a href="/attendee">Go to Events List</a>
23      </div>
24  </body>
25  </html>
26  <!-- --------------------- END NEW CODE --------------------->
```

views\add-event.ejs

```
1   <!-- -------------------- START NEW CODE -------------------->
2
3   <!DOCTYPE html>
4   <html lang="en">
5   <head>
6       <meta charset="UTF-8">
7       <title>Add New Event</title>
8       <link rel="stylesheet" href="/main.css">
9   </head>
10  <body>
11      <header>
12          <h1>Create New Event</h1>
13          <a href="/organiser">Back to Dashboard</a>
14      </header>
15
16      <main>
17          <form action="/organiser/add-event" method="POST">
18              <h3>General Information</h3>
19
20              <label for="title">Event Title:</label>
21              <input type="text" id="title" name="title" placeholder="e.g. Summer Concert"
    required>
22
23              <label for="description">Short Description:</label>
24              <input type="text" id="description" name="description" placeholder="Short
    summary of the event" required>
25
26              <label for="content">Full Content / Details:</label>
27              <textarea id="content" name="content" rows="5" placeholder="Detailed
    information about the event..." required></textarea>
28
29              <label for="event_date">Event Date:</label>
30              <input type="date" id="event_date" name="event_date" required>
31
32              <hr>
33              <h3>Ticket Settings</h3>
34              <p><em>Initial ticket configuration</em></p>
35
36              <label for="price_tickets">Base Ticket Price (£):</label>
37              <input type="number" id="price_tickets" name="price_tickets" step="0.01"
    value="0.00" required>
38
39              <div style="margin-top: 20px;">
40                  <button type="submit">Create Event as Draft</button>
41                  <a href="/organiser" class="cancel" style="margin-left: 15px;">Cancel</a>
42              </div>
43          </form>
44      </main>
45  </body>
46  </html>
47  <!-- -------------------- END NEW CODE -------------------->
```

views\settings.ejs

```
1   <!-- -------------------- START NEW CODE -------------------->
2   <!DOCTYPE html>
3   <html>
4   <head>
5       <title>Settings</title>
6       <link rel="stylesheet" href="/main.css">
7   </head>
8   <body>
9       <header>
10          <h1>System Settings</h1>
11          <a href="/organiser">Back to Dashboard</a>
12      </header>
13
14      <form action="/organiser/settings" method="POST">
15          <label>Site Title:</label>
16          <input type="text" name="site_title" value="<%= settings.site_title %>" required>
17
18          <label>Author Name:</label>
19          <input type="text" name="author_name" value="<%= settings.author_name %>" required>
20
21          <button type="submit">Save Changes</button>
22      </form>
23  </body>
24  </html>
25  <!-- -------------------- END NEW CODE -------------------->
```

views\organiser.ejs

```
1   <!-- -------------------- START NEW CODE -------------------->
2   <!DOCTYPE html>
3   <html>
4   <head>
5       <title>Organiser Dashboard</title>
6       <link rel="stylesheet" href="/main.css">
7   </head>
8   <body>
9       <header>
10          <h1>Organiser Dashboard</h1>
11          <nav>
12              <a href="/">Exit to Home</a> |
13              <a href="/organiser/settings">Global Settings</a>
14          </nav>
15      </header>
16
17      <form action="/organiser/add-event" method="POST">
18          <button type="submit">Create New Event</button>
19      </form>
20
21      <h2>Published Events</h2>
22      <ul>
23          <% publishedEvents.forEach(function(event) { %>
24              <li>
25                  <h3><%= event.title %></h3>
26                  <p>Date: <%= event.event_date %></p>
27                  <form action="/organiser/delete/<%= event.id %>" method="POST"
    style="background:none; border:none; padding:0;">
28                      <button type="submit" style="background-color: var(--danger-
    color);">Delete</button>
29                  </form>
30              </li>
31          <% }); %>
32      </ul>
33
34      <h2>Drafts</h2>
35      <ul>
36          <% draftEvents.forEach(function(event) { %>
37              <li>
38                  <strong><%= event.title %></strong><br>
39                  <a href="/organiser/edit-event/<%= event.id %>">Edit Event</a> |
40                  <form action="/organiser/publish/<%= event.id %>" method="POST"
    style="display:inline; background:none; border:none; padding:0;">
41                      <button type="submit" style="padding: 5px 10px;
    margin:0;">Publish</button>
42                  </form>
43              </li>
44          <% }); %>
45      </ul>
46  </body>
47  </html>
48  <!-- -------------------- END NEW CODE -------------------->
```

views\my-bookings.ejs

```
1   <!-- -------------------- START NEW CODE -------------------->
2   <!DOCTYPE html>
3   <html>
4   <head>
5       <title>My Bookings</title>
6       <link rel="stylesheet" href="/main.css">
7   </head>
8   <body>
9       <header>
10          <h1>My Bookings</h1>
11          <nav>
12              <a href="/attendee">Back to Events</a>
13          </nav>
14      </header>
15
16      <main>
17          <% if (bookings.length > 0) { %>
18              <table style="width:100%; border-collapse: collapse; margin-top: 20px;">
19                  <thead>
20                      <tr style="background-color: #f2f2f2;">
21                          <th style="padding: 10px; border: 1px solid #ddd;">Event</th>
22                          <th style="padding: 10px; border: 1px solid #ddd;">Date</th>
23                          <th style="padding: 10px; border: 1px solid #ddd;">Attendee
    Name</th>
24                          <th style="padding: 10px; border: 1px solid #ddd;">Ticket Type</th>
25                      </tr>
26                  </thead>
27                  <tbody>
28                      <% bookings.forEach(function(booking) { %>
29                          <tr>
30                              <td style="padding: 10px; border: 1px solid #ddd;"><%=
    booking.title %></td>
31                              <td style="padding: 10px; border: 1px solid #ddd;"><%=
    booking.event_date %></td>
32                              <td style="padding: 10px; border: 1px solid #ddd;"><%=
    booking.attendee_name %></td>
33                              <td style="padding: 10px; border: 1px solid #ddd;">
34                                  <%= booking.ticket_type === 'full' ? 'Full Price' :
    'Concession' %>
35                              </td>
36                          </tr>
37                      <% }); %>
38                  </tbody>
39              </table>
40          <% } else { %>
41              <p>You haven't made any bookings yet.</p>
42              <a href="/attendee" class="btn">Browse Events</a>
43          <% } %>
44      </main>
45  </body>
46  </html>
47  <!-- -------------------- END NEW CODE -------------------->
```

views\edit-event.ejs

```
1   <!-- -------------------- START NEW CODE -------------------->
2   <!DOCTYPE html>
3   <html lang="en">
4   <head>
5       <meta charset="UTF-8">
6       <title>Edit Event - University of London</title>
7       <link rel="stylesheet" href="/main.css">
8   </head>
9   <body>
10      <header>
11          <h1>Edit Event Details</h1>
12          <a href="/organiser">← Back to Dashboard</a>
13      </header>
14
15      <main>
16          <form action="/organiser/edit-event/<%= event.id %>" method="POST">
17              <h3>General Information</h3>
18              <label>Title:</label>
19              <input type="text" name="title" value="<%= event.title %>" required>
20
21              <label>Description:</label>
22              <textarea name="description" rows="4" required><%= event.description %>
    </textarea>
23
24              <label>Event Date:</label>
25              <input type="date" name="event_date" value="<%= event.event_date %>" required>
26
27              <hr>
28              <h3>Ticket & Financial Settings</h3>
29              <p><em>Adjust the availability and pricing for this event.</em></p>
30
31              <div style="display: grid; grid-template-columns: 1fr 1fr; gap: 20px;">
32                  <div>
33                      <label>Full Price Quantity:</label>
34                      <input type="number" name="full_price_tickets" value="<%=
    event.full_price_tickets %>" min="0">
35                  </div>
36                  <div>
37                      <label>Full Price Cost (£):</label>
38                      <input type="number" step="0.01" name="full_price_cost" value="<%=
    event.full_price_cost %>" min="0">
39                  </div>
40              </div>
41
42              <div style="display: grid; grid-template-columns: 1fr 1fr; gap: 20px; margin-
    top: 15px;">
43                  <div>
44                      <label>Concession Quantity:</label>
45                      <input type="number" name="concession_tickets" value="<%=
    event.concession_tickets %>" min="0">
46                  </div>
47                  <div>
```

```
48                    <label>Concession Cost (£):</label>
49                    <input type="number" step="0.01" name="concession_cost" value="<%=
   event.concession_cost %>" min="0">
50                </div>
51             </div>
52
53             <div style="margin-top: 30px;">
54                 <button type="submit">Update Event</button>
55             </div>
56          </form>
57       </main>
58    </body>
59 </html>
60 <!-- --------------------- END NEW CODE --------------------->
```

views\attendee-home.ejs

```
1  <!-- -------------------- START NEW CODE -------------------->
2  <!DOCTYPE html>
3  <html>
4  <head>
5      <title>Find Events</title>
6      <link rel="stylesheet" href="/main.css">
7  </head>
8  <body>
9      <header>
10         <h1>Available Events</h1>
11
12         <nav style="margin-bottom: 20px;">
13             <a href="/" class="btn-secondary">Home</a> |
14             <a href="/attendee/my-bookings" class="btn-secondary" style="font-weight: bold;
   color: #3498db;">
15                 View My Bookings
16             </a>
17         </nav>
18     </header>
19
20     <section style="background: white; padding: 20px; border: 1px solid #bfc1c3; margin-
   bottom: 20px;">
21         <form action="/attendee" method="GET" style="box-shadow: none; padding: 0; border:
   none; display: flex; gap: 10px; align-items: flex-end;">
22             <div style="flex-grow: 1;">
23                 <label for="search" style="margin-top: 0;">Search Events by Title:</label>
24                 <input type="text" name="search" id="search" placeholder="e.g. Graduation"
   value="<%= search %>">
25             </div>
26             <button type="submit" style="margin-top: 0; padding: 10px;">Search</button>
27         </form>
28     </section>
29
30     <main>
31         <% if (events.length > 0) { %>
32             <ul>
33                 <% events.forEach(function(event) { %>
34                     <li>
35                         <h3><%= event.title %></h3>
36                         <p><strong>Date:</strong> <%= event.event_date %></p>
37                         <p><strong>Price from:</strong> £<%= event.full_price_cost %></p>
38                         <a href="/attendee/event/<%= event.id %>">View & Book</a>
39                     </li>
40                 <% }); %>
41             </ul>
42         <% } else { %>
43             <p>No events found for "<%= search %>".</p>
44             <a href="/attendee">Show all events</a>
45         <% } %>
46     </main>
47  </body>
48  </html>
```

```
49    <!-- --------------------- END NEW CODE --------------------->
```

views\attendee-event.ejs

```
1   <!-- -------------------- START NEW CODE -------------------->
2   <!DOCTYPE html>
3   <html>
4   <head>
5       <title>Book Event</title>
6       <link rel="stylesheet" href="/main.css">
7   </head>
8   <body>
9       <header>
10          <h1><%= event.title %></h1>
11          <a href="/attendee">Back to Events List</a>
12      </header>
13
14      <main>
15          <h3>Event Details</h3>
16          <p><strong>Date:</strong> <%= event.event_date %></p>
17          <p><strong>Description:</strong> <%= event.description %></p>
18
19          <div>
20              <h4>Ticket Availability</h4>
21              <p>Full Price: £<%= event.full_price_cost %> (<%= event.full_price_tickets %>
    available)</p>
22              <p>Concession: £<%= event.concession_cost %> (<%= event.concession_tickets %>
    available)</p>
23          </div>
24
25          <h3>Book Your Tickets</h3>
26          <form action="/attendee/book/<%= event.id %>" method="POST">
27              <label>Your Name:</label>
28              <input type="text" name="attendee_name" placeholder="John Doe" required>
29
30              <label>Full Price tickets?</label>
31              <input type="number" name="num_full" min="0" max="<%= event.full_price_tickets
    %>" value="0">
32
33              <label>Concession tickets?</label>
34              <input type="number" name="num_conc" min="0" max="<%= event.concession_tickets
    %>" value="0">
35
36              <button type="submit">Confirm Booking</button>
37          </form>
38      </main>
39  </body>
40  </html>
41  <!-- -------------------- END NEW CODE -------------------->
```

public\main.css

```
 1  /* ----------------START NEW CODE ---------------------------- */
 2  :root {
 3      --primary-color: ■ #002147;
 4      --accent-color: ■ #003366;
 5      --success-color: ■ #1d70b8;
 6      --danger-color: ■ #b10e1e;
 7      --bg-color: □ #f3f2f1;
 8      --card-bg: □ #ffffff;
 9      --text-color: ■ #0b0c0c;
10      --border-radius: 2px;
11  }
12
13  body {
14      font-family: "Helvetica Neue", Helvetica, Arial, sans-serif;
15      background-color: var(--bg-color);
16      color: var(--text-color);
17      margin: 0;
18      padding: 40px 20px;
19      display: flex;
20      flex-direction: column;
21      align-items: center;
22  }
23
24  body > * { width: 100%; max-width: 800px; }
25
26  header {
27      background: var(--card-bg);
28      padding: 30px;
29      border-top: 8px solid var(--primary-color);
30      border-bottom: 1px solid #bfc1c3;
31      margin-bottom: 30px;
32      text-align: center;
33  }
34
35  h1, h2, h3 { color: var(--primary-color); }
36
37  h1 { border-bottom: 4px solid var(--primary-color); padding-bottom: 15px; }
38
39  .nav-box {
40      background: var(--card-bg);
41      border: 1px solid #bfc1c3;
42      padding: 30px;
43      margin: 20px 0;
44      text-align: center;
45  }
46
47
48  form {
49      background: var(--card-bg);
50      padding: 30px;
51      border: 1px solid #bfc1c3;
```

```css
 52        margin-bottom: 20px;
 53    }
 54
 55
 56    .search-box form {
 57        display: flex;
 58        padding: 20px;
 59        gap: 0;
 60        border: none;
 61        background: none;
 62    }
 63
 64    .search-box input[type="text"] {
 65        flex: 1;
 66        margin-top: 0;
 67        height: 45px;
 68        border-right: none;
 69        box-sizing: border-box;
 70    }
 71
 72    .search-box button {
 73        height: 45px;
 74        padding: 0 30px;
 75        margin-top: 0;
 76        border-radius: 0;
 77        white-space: nowrap;
 78    }
 79    /* -------------------------------------------- */
 80
 81    label { display: block; margin-top: 20px; font-weight: 700; }
 82
 83    input, textarea {
 84        width: 100%;
 85        padding: 10px;
 86        margin-top: 8px;
 87        border: 2px solid var(--text-color);
 88        box-sizing: border-box;
 89    }
 90
 91    button, .btn-uol {
 92        background-color: var(--primary-color);
 93        color: ☐ white;
 94        border: none;
 95        padding: 12px 24px;
 96        font-weight: 700;
 97        cursor: pointer;
 98        transition: background 0.2s;
 99    }
100
101    button:hover { background-color: ■ #001126; }
102
103    a { color: var(--success-color); font-weight: bold; text-decoration: none; }
104    a:hover { text-decoration: underline; }
105
```

```css
106  ul { list-style: none; padding: 0; }
107
108  li {
109      background: var(--card-bg);
110      border-left: 10px solid var(--primary-color);
111      padding: 20px;
112      margin-bottom: 15px;
113      border-right: 1px solid #bfc1c3;
114      border-top: 1px solid #bfc1c3;
115      border-bottom: 1px solid #bfc1c3;
116  }
117
118
119  table {
120      width: 100%;
121      border-collapse: collapse;
122      background: ☐ white
123  }
124
125  th, td {
126      border: 1px solid #bfc1c3;
127      padding: 12px;
128      text-align: left;
129  }
130
131  th { background-color: var(--bg-color); color: var(--primary-color); }
132  /* ---------------- END NEW CODE ---------------------------- */
```

db_schema.sql

```sql
1   --------------------------- START NEW CODE ---------------------------
2   PRAGMA foreign_keys=ON;
3
4   CREATE TABLE IF NOT EXISTS settings (
5       id INTEGER PRIMARY KEY AUTOINCREMENT,
6       site_title TEXT DEFAULT 'My Event Manager',
7       author_name TEXT DEFAULT 'Event Organiser'
8   );
9
10  CREATE TABLE IF NOT EXISTS events (
11      id INTEGER PRIMARY KEY AUTOINCREMENT,
12      title TEXT DEFAULT 'New Event',
13      description TEXT,
14      event_date TEXT, -- Store as YYYY-MM-DD
15      status TEXT DEFAULT 'draft', -- 'draft' or 'published'
16      created_at DATETIME DEFAULT CURRENT_TIMESTAMP,
17      published_at DATETIME,
18      last_modified DATETIME DEFAULT CURRENT_TIMESTAMP,
19      -- Requisito específico: Tipos de bilhetes separados
20      full_price_tickets INTEGER DEFAULT 0,
21      full_price_cost REAL DEFAULT 0.0,
22      concession_tickets INTEGER DEFAULT 0,
23      concession_cost REAL DEFAULT 0.0
24  );
25
26  CREATE TABLE IF NOT EXISTS bookings (
27      id INTEGER PRIMARY KEY AUTOINCREMENT,
28      event_id INTEGER,
29      attendee_name TEXT NOT NULL,
30      num_full_tickets INTEGER DEFAULT 0,
31      num_concession_tickets INTEGER DEFAULT 0,
32      FOREIGN KEY(event_id) REFERENCES events(id) ON DELETE CASCADE
33  );
34
35  INSERT INTO settings (id, site_title, author_name) VALUES (1, 'University Events', 'Student
    Admin');
36
37  --Insert initial events.
38  INSERT INTO events (title, description, event_date, status, full_price_tickets,
    full_price_cost, concession_tickets, concession_cost)
39  VALUES (
40      'Global Tech Summit 2026',
41      'Join the biggest tech conference. Topics include AI, Web Development and Ethics.',
42      '2026-05-20',
43      'published',
44      100, 50.0, 50, 25.0
45  );
46
47  INSERT INTO events (title, description, event_date, status, full_price_tickets,
    full_price_cost, concession_tickets, concession_cost)
48  VALUES (
49      'Sunset Yoga by the River',
```

```sql
50         'Relaxing yoga session for all levels. Bring your own mat.',
51         '2026-02-14',
52         'published',
53         5, 15.0, 2, 10.0
54  );
55
56  INSERT INTO events (title, description, event_date, status, full_price_tickets,
    full_price_cost, concession_tickets, concession_cost)
57  VALUES (
58         'Italian Cooking Masterclass',
59         'Learn to make authentic pasta from scratch with Chef Mario.',
60         '2026-08-10',
61         'published',
62         15, 45.0, 10, 30.0
63  );
64
65  INSERT INTO events (title, description, event_date, status, full_price_tickets,
    full_price_cost, concession_tickets, concession_cost)
66  VALUES (
67         'Birthday Party',
68         'Come to my birthday party!',
69         '2026-10-10',
70         'draft',
71         20, 30.0, 5, 30.0
72  );
73
74  ----------------------------- END NEW CODE --------------------------
```