

The IIE Varsity College

CLDV6211

Cloud Development A

ST10088532

POE

Aobakwe Kegoriloe

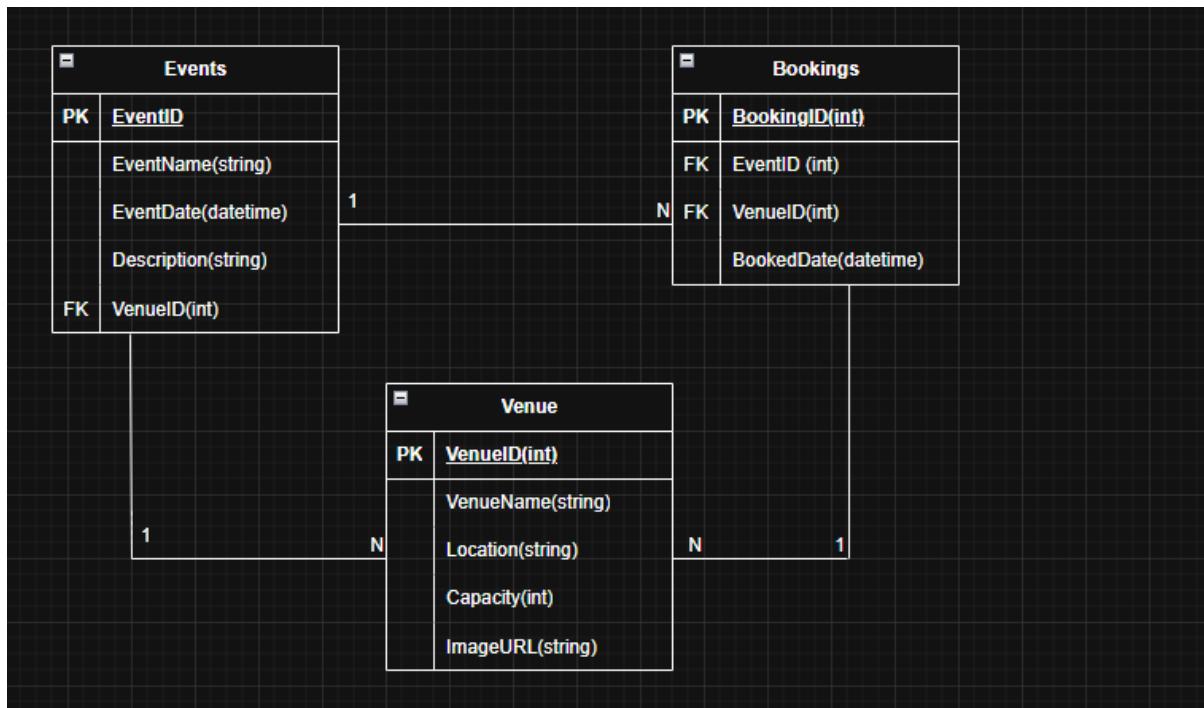
Due Date: 23 June 2025

## Table of Contents

Part 1 .....	3
Part 2 .....	22
Part 3 .....	41
Bibliography.....	57

# Part 1

## ERD diagram



## Screenshots of Web App

EventEasePOE Home Venues Event Booking Privacy

Welcome to EventEase

EventEase is your go-to platform for managing events and venue bookings efficiently. Our user-friendly interface allows booking specialists to easily browse available venues, view scheduled events, and manage bookings without the hassle of double bookings or scheduling conflicts.

With EventEase, you can:

- Quickly view and manage venue information.
- Schedule and organize events seamlessly.
- Ensure that all bookings are handled securely and efficiently.

View Venues View Events View Bookings

© 2025 - EventEasePOE - [Privacy](#)

Index - EventEasePOE

localhost:5035/Venues

EventEasePOE Home Venues Event Booking Privacy

## Index

[Create New](#)

VenueName	Location	Capacity	ImageUrl
Ntsie street	Soweto	200	<a href="https://www.google.com/url?sa=i&amp;url=https%3A%2F%2Fwww.schemecolor.com%2Fshades-of-blue.php&amp;psig=AOvVaw3Deo4OO0LUU_LpiWNnRlb8&amp;tust=1744132122518000&amp;source=images&amp;cd=vfe&amp;opi=89978449&amp;ved=0CBQC">https://www.google.com/url?sa=i&amp;url=https%3A%2F%2Fwww.schemecolor.com%2Fshades-of-blue.php&amp;psig=AOvVaw3Deo4OO0LUU_LpiWNnRlb8&amp;tust=1744132122518000&amp;source=images&amp;cd=vfe&amp;opi=89978449&amp;ved=0CBQC</a>

© 2025 - EventEasePOE - [Privacy](#)

Events - EventEasePOE

localhost:5035/Events

EventEasePOE Home Venues Event Booking Privacy

## Index

[Create New](#)

EventName	EventDate	Description	VenuelId
-----------	-----------	-------------	----------

© 2025 - EventEasePOE - [Privacy](#)

Create - EventEasePOE

localhost:5035/Events/Create

EventEasePOE Home Venues Event Booking Privacy

## Create

### Event

---

EventName  
The Sundae Chronicles

EventDate  
2025/04/27 13:00

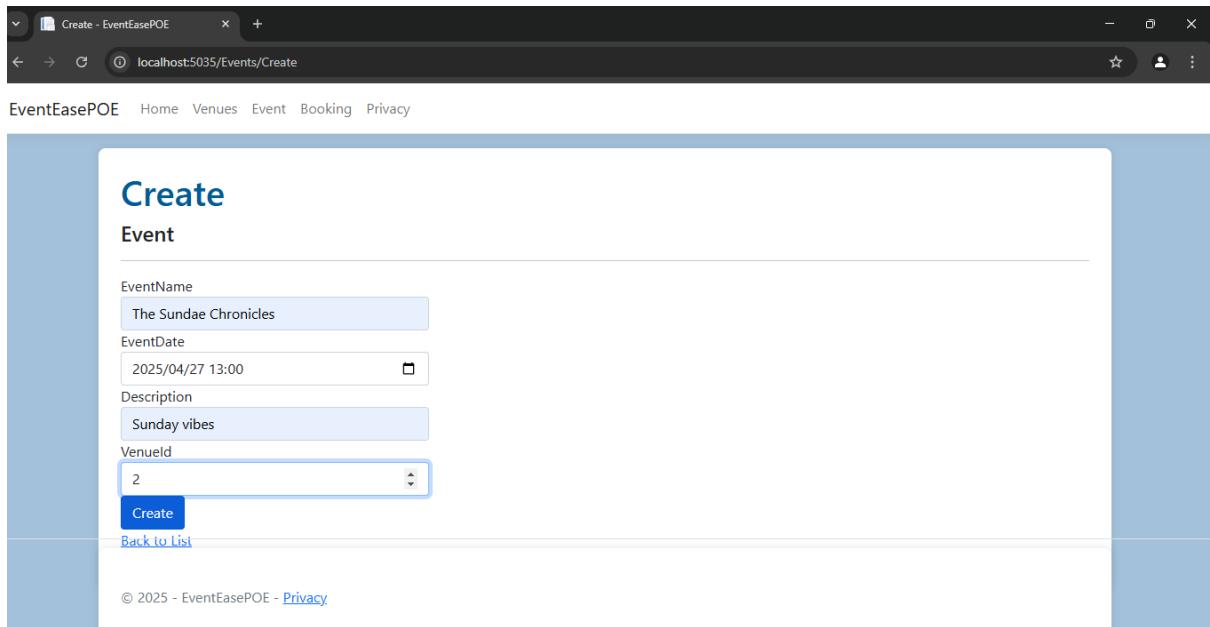
Description  
Sunday vibes

VenuelId  
2

[Create](#)

[Back to List](#)

© 2025 - EventEasePOE - [Privacy](#)



Bookings - EventEasePOE

localhost:5035/Bookings

EventEasePOE Home Venues Event Booking Privacy

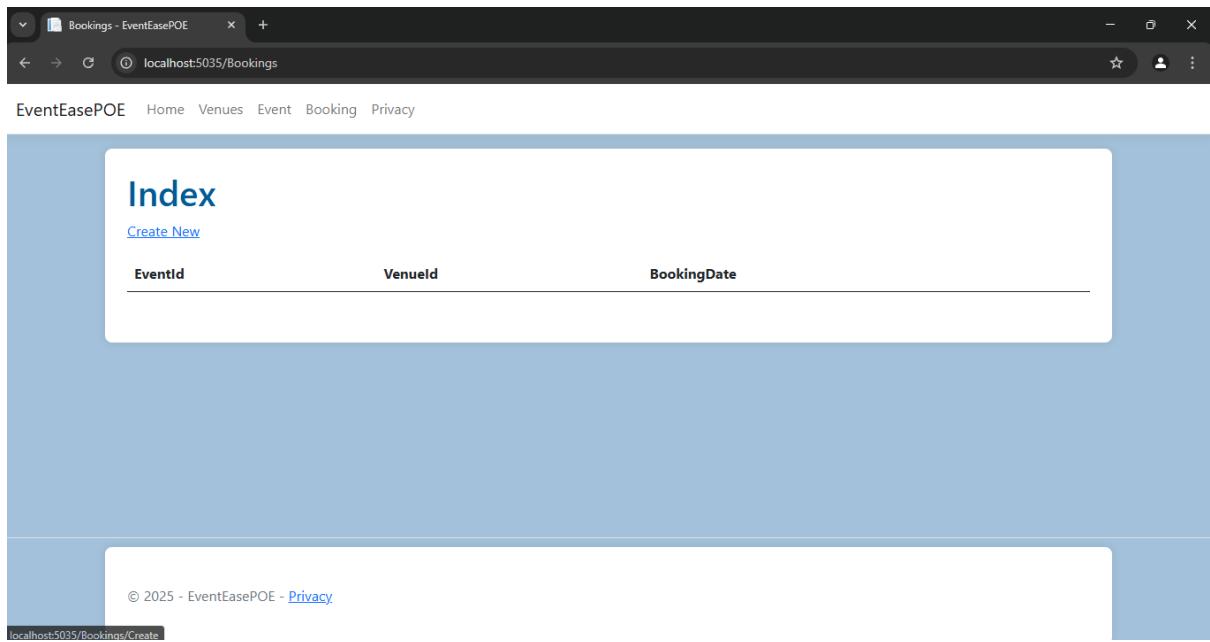
## Index

[Create New](#)

EventId	VenuelId	BookingDate
---------	----------	-------------

© 2025 - EventEasePOE - [Privacy](#)

localhost:5035/Bookings/Create



Create - EventEasePOE

localhost:5035/Bookings/Create

EventEasePOE Home Venues Event Booking Privacy

## Create

### Booking

---

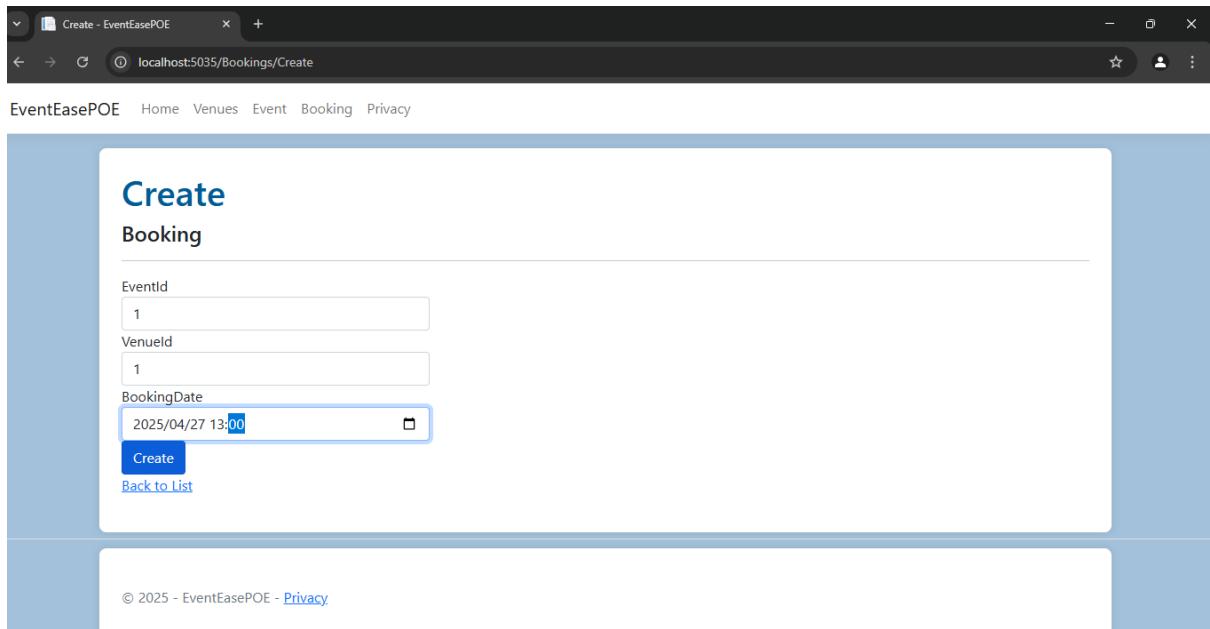
EventId  
1

VenueId  
1

BookingDate  
2025/04/27 13:00

[Back to List](#)

© 2025 - EventEasePOE - [Privacy](#)



## Visual Studio Code

File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help Search (Ctrl+Q) EventEasePOE AK Live Share

Feature 'Edit and Continue' is currently unavailable due to an internal error. Show Stack Trace  
Feature 'Symbol search' is currently unavailable due to an internal error. Show Stack Trace

Server Explorer

laptop-tktpff6\sqlexpress.Connection Data.dbo  
laptop-tktpff6\sqlexpress.DatCraft\_db.dbo  
laptop-tktpff6\sqlexpress.EventEasedata.dbo

- Tables
  - \_EFMigrationsHistory
  - Bookings
    - BookingId
    - EventId
    - VenueId
    - BookingDate
  - Events
    - EventId
    - EventName
    - EventDate
    - Description
    - VenueId
  - Venue
    - VenueId
    - VenueName
    - Location
    - Capacity
    - ImageUrl
  - Venues
    - VenueId
    - VenueName
    - Location
    - Capacity
    - ImageUrl

oft.AspNetCore.Mvc.Infrastructure  
oft.AspNetCore.Routing.Endpoint  
oft.AspNetCore.Authorization  
oft.AspNetCore.Diagnostics.DependencyInjection  
tools services..  
tools design surface service  
tools selection service  
tools tag navigation service  
tools hotkey manager service  
on connected  
host services...  
ad ignoring http://localhost  
icon connected  
tools services..  
tools design surface service  
tools selection service  
tools tag navigation service  
tools hotkey manager service  
on connected  
host services..  
x0 has exited with code 0 (0)  
'JavaScript debug adapter: 1' has exited with code 4294  
'[28324] EventEasePOE.exe' has exited with code 0 (0)

File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help Search (Ctrl+Q) EventEasePOE AK Live Share

Program.cs

```

1  using EventEasePOE.Models; // Ensure this namespace is included
2  using Microsoft.EntityFrameworkCore;
3
4  var builder = WebApplication.CreateBuilder(args);
5
6  // Add services to the container.
7  builder.Services.AddControllersWithViews();
8
9  // Configure the DbContext with SQL Server
10 builder.Services.AddDbContext<ApplicationDbContext>(options =>
11     options.UseSqlServer(builder.Configuration.GetConnectionString("DefaultConnection")));
12
13 var app = builder.Build();
14
15 // Configure the HTTP request pipeline.
16 if (app.Environment.IsDevelopment())
17 {
18     app.UseExceptionHandler("/Home/Error");
19 }
20 app.UseStaticFiles();
21
22 app.UseRouting();
23
24 app.UseAuthorization();
25
26 app.MapControllerRoute(
27     name: "default",
28     pattern: "{controller=Home}/{action=Index}/{id?}");
29
30 app.Run();

```

Output Solution Explorer

Search Solution Explorer (Ctrl+Shift+F)

- Event.cs
- Venue.cs
- Views
  - Bookings
    - Create.cshtml
    - Delete.cshtml
    - Details.cshtml
    - Edit.cshtml
    - Index.cshtml
  - Events
    - Create.cshtml
    - Delete.cshtml
    - Details.cshtml
    - Edit.cshtml
    - Index.cshtml
  - Home
    - Index.cshtml
    - Privacy.cshtml
  - Shared
    - \_Layout.cshtml
    - \_ValidationScriptsPartial.cshtml
    - Error.cshtml
  - Venues
    - \_ViewImports.cshtml
    - \_ViewStart.cshtml

appsettings.json

```

1  {
2      "Logging": {
3          "LogLevel": {
4              "Default": "Information",
5              "Microsoft.AspNetCore": "Warning"
6          }
7      },
8      "AllowedHosts": "*",
9      "ConnectionStrings": {
10         "DefaultConnection": "Server=LAPTOP-TKTPFF6\SQLEXPRESS;Initial Catalog=EventEasedata;Integrated Security=True"
11     }
12 }
13

```

Toolbox Error List SQL Server Object Explorer



Screenshot of Visual Studio showing the code editor and Solution Explorer for the EventEasePOE project.

**Code Editor:**

```

43     <button>
44         <div class="collapse navbar-collapse" id="navBarNav">
45             <ul class="nav nav-tabs mb-2 mb-lg-0">
46                 <li class="nav-item">
47                     <a class="nav-link" asp-area="" asp-controller="Home" asp-action="Index">Home</a>
48                 </li>
49                 <li class="nav-item">
50                     <a class="nav-link" asp-area="" asp-controller="Venues" asp-action="Index">Venues</a>
51                 </li>
52                 <li class="nav-item">
53                     <a class="nav-link" asp-area="" asp-controller="Events" asp-action="Index">Event</a>
54                 </li>
55                 <li class="nav-item">
56                     <a class="nav-link" asp-area="" asp-controller="Bookings" asp-action="Index">Bookings</a>
57                 </li>
58                 <li class="nav-item">
59                     <a class="nav-link" asp-area="" asp-controller="Home" asp-action="Privacy">Privacy</a>
60                 </li>
61             </ul>
62         </div>
63     </div>
64 </header>
65 <div class="container">
66     <main role="main" class="pb-3">
67         @RenderBody()
68     </main>
69 </div>
70 </div>
71 <footer class="border-top footer text-muted">
72     <div class="container">
73         <copy> © 2025 - EventEasePOE - <a href="#" asp-area="" asp-controller="Home" asp-action="Privacy">Privacy</a>
74     </div>
75 </footer>
76 </div>
77 <script src="~/lib/jquery/dist/jquery.min.js"></script>
78 <script src="~/lib/bootstrap/dist/js/bootstrap.bundle.min.js"></script>
79 <script src="~/js/site.js" asp-append-version="true"></script>
80 <@await RenderSectionAsync("Scripts", required: false) >
81 </body>
82 </html>

```

**Solution Explorer:**

- Views
  - Bookings
    - Create.cshtml
    - Delete.cshtml
    - Details.cshtml
    - Edit.cshtml
    - Index.cshtml
  - Events
    - Create.cshtml
    - Delete.cshtml
    - Details.cshtml
    - Edit.cshtml
    - Index.cshtml
  - Home
    - Index.cshtml
    - Privacy.cshtml
  - Shared
    - \_Layout.cshtml
    - \_ValidationScriptsPartial.cshtml
    - Error.cshtml
  - Venues
    - Create.cshtml
    - Delete.cshtml
    - Details.cshtml
    - Edit.cshtml
    - Index.cshtml

Screenshot of Visual Studio showing the code editor and Solution Explorer for the EventEasePOE project.

**Code Editor:**

```

1 @{
2     ViewData["Title"] = "Home";
3 }
4 <div class="container mt-4">
5     <h1>Welcome to EventEase</h1>
6     <div class="introduction">
7         <p>EventEase is your go-to platform for managing events and venue bookings efficiently. Our user-friendly interface allows booking specific details quickly and easily.</p>
8         <p>With EventEase, you can:</p>
9         <ul>
10             <li>Quickly view and manage venue information.</li>
11             <li>Schedule and organize events seamlessly.</li>
12             <li>Ensure that all bookings are handled securely and efficiently.</li>
13         </ul>
14     </div>
15     <div class="buttons">
16         <a class="btn btn-primary" href="@Url.Action("Index", "Venues")">View Venues</a>
17         <a class="btn btn-primary" href="@Url.Action("Index", "Events")">View Events</a>
18         <a class="btn btn-primary" href="@Url.Action("Index", "Bookings")">View Bookings</a>
19     </div>
20 </div>
21 
```

**Solution Explorer:**

- Views
  - Bookings
    - Create.cshtml
    - Delete.cshtml
    - Details.cshtml
    - Edit.cshtml
    - Index.cshtml
  - Events
    - Create.cshtml
    - Delete.cshtml
    - Details.cshtml
    - Edit.cshtml
    - Index.cshtml
  - Home
    - Index.cshtml
    - Privacy.cshtml
  - Shared
    - \_Layout.cshtml
    - \_ValidationScriptsPartial.cshtml
    - Error.cshtml
  - Venues
    - Create.cshtml
    - Delete.cshtml
    - Details.cshtml
    - Edit.cshtml
    - Index.cshtml

The screenshot shows the Microsoft Visual Studio IDE interface. The top menu bar includes File, Edit, View, Git, Project, Build, Debug, Test, Analyze, Tools, Extensions, Windows, Help, and Search (Ctrl+F). The toolbar has icons for Undo, Redo, Cut, Copy, Paste, Find, Replace, and others. The title bar says "EventEasePOE".

The left sidebar contains the Toolbox, Error List, and SQL Server Object Explorer.

The main area shows the code editor with the file "Index.cshtml" open. The code is an ASP.NET MVC view for displaying a list of events:

```
<thead>
    <tr>
        <th>@Html.DisplayNameFor(model => model.EventName)</th>
        <th>@Html.DisplayNameFor(model => model.EventDate)</th>
        <th>@Html.DisplayNameFor(model => model.Description)</th>
        <th>@Html.DisplayNameFor(model => model.VenueId)</th>
    </tr>
</thead>
<tbody>
@foreach (var item in Model) {
    <tr>
        <td>@Html.DisplayFor(modelItem => item.EventName)</td>
        <td>@Html.DisplayFor(modelItem => item.EventDate)</td>
        <td>@Html.DisplayFor(modelItem => item.Description)</td>
        <td>@Html.DisplayFor(modelItem => item.VenueId)</td>
        <td>
            <a href="#" asp-action="Edit" asp-route-id="@item.EventId">Edit</a> | 
            <a href="#" asp-action="Details" asp-route-id="@item.EventId">Details</a> | 
            <a href="#" asp-action="Delete" asp-route-id="@item.EventId">Delete</a>
        </td>
    </tr>
}
</tbody>
</table>
```

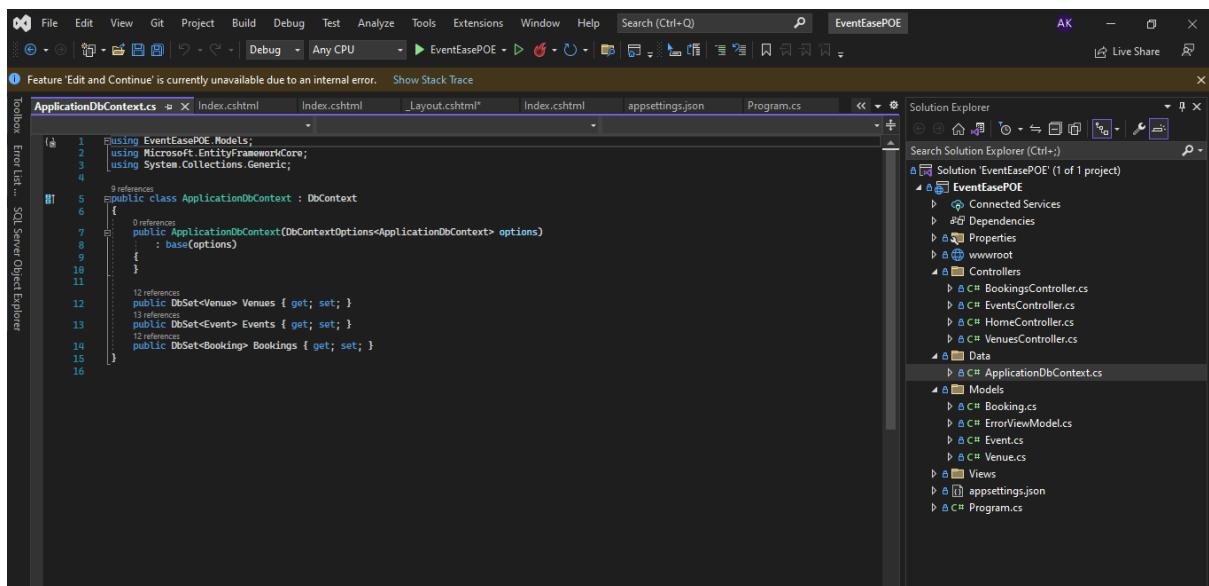
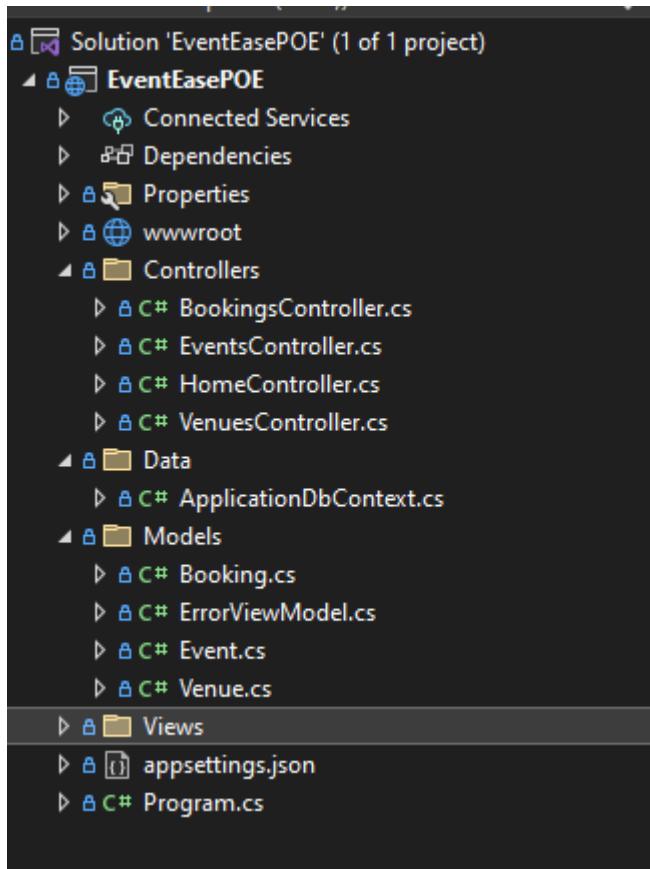
The Solution Explorer on the right lists the project structure:

- Views
  - Venue.cs
  - Booking
    - Create.cshtml
    - Delete.cshtml
    - Details.cshtml
    - Edit.cshtml
    - Index.cshtml
  - Events
    - Create.cshtml
    - Delete.cshtml
    - Details.cshtml
    - Edit.cshtml
    - Index.cshtml
  - Home
    - Index.cshtml
    - Privacy.cshtml
  - Shared
    - \_Layout.cshtml
    - ValidationScriptsPartial.cshtml
    - Error.cshtml
  - Venues
    - Create.cshtml
    - Delete.cshtml
    - Details.cshtml
    - Edit.cshtml
    - Index.cshtml

The screenshot shows the Visual Studio IDE interface with the following details:

- Solution Explorer:** Located on the right side, it displays the project structure. It includes the **Data**, **Models**, and **Views** folders. Under **Views**, there are sub-folders for **Bookings**, **Events**, **Home**, and **Venues**, each containing files like Create.cshtml, Delete.cshtml, Details.cshtml, Edit.cshtml, and Index.cshtml.
- Code Editor:** The main window shows the **Index.cshtml** file. The code is a Razor view for displaying bookings. It starts with a header section and then a table with columns for Event ID, Venue ID, and Booking Date. Each row contains links for Edit, Details, and Delete actions.

```
1 @model IEnumerable<EventEasePOE.Models.Booking>
2 
3 @{
4     ViewData["Title"] = "Bookings";
5 }
6 
7 <h1>Index</h1>
8 
9 <p>
10    <a href="#" asp-action="Create">Create New</a>
11 </p>
12 <table class="table">
13     <thead>
14         <tr>
15             <th>
16                 @Html.DisplayNameFor(model => model.EventId)
17             </th>
18             <th>
19                 @Html.DisplayNameFor(model => model.VenueId)
20             </th>
21             <th>
22                 @Html.DisplayNameFor(model => model.BookingDate)
23             </th>
24         </tr>
25     </thead>
26     <tbody>
27         @foreach (var item in Model) {
28             <tr>
29                 <td>
30                     @Html.DisplayFor(modelItem => item.EventId)
31                 </td>
32                 <td>
33                     @Html.DisplayFor(modelItem => item.VenueId)
34                 </td>
35                 <td>
36                     @Html.DisplayFor(modelItem => item.BookingDate)
37                 </td>
38                 <td>
39                     <a href="#" asp-action="Edit" asp-route-id="@item.BookingId">Edit</a> |
40                     <a href="#" asp-action="Details" asp-route-id="@item.BookingId">Details</a> |
41                     <a href="#" asp-action="Delete" asp-route-id="@item.BookingId">Delete</a>
42                 </td>
43             </tr>
44         }
45     </tbody>
46 </table>
```



The screenshot shows the Visual Studio IDE with the 'Booking.cs' file open in the code editor. The code defines a 'Booking' class with properties: BookingId, EventId, VenueId, and BookingDate. The 'BookingId' property is annotated with a 'BookId' attribute. The code editor interface includes tabs for 'ApplicationDbContext.cs', 'Index.cshtml', 'Index.cshtml', '\_Layout.cshtml\*', 'Index.cshtml', and 'appsettings.json'. The status bar at the bottom indicates 'Any CPU' and 'EventEasePOE'. A message bar at the top left says 'Feature 'Edit and Continue' is currently unavailable due to an internal error.'

```
1  namespace EventEasePOE.Models
2  {
3      public class Booking
4      {
5          public int BookId { get; set; }
6          public int EventId { get; set; }
7          public int VenueId { get; set; }
8          public DateTime BookingDate { get; set; }
9      }
10 }
11 
```

The screenshot shows the Visual Studio IDE with the 'Event.cs' file open in the code editor. The code defines an 'Event' class with properties: EventId, EventName, EventDate, Description, and VenueId. The 'EventId' property is annotated with an 'EventId' attribute. The code editor interface includes tabs for 'Booking.cs', 'ApplicationDbContext.cs', 'Index.cshtml', 'Index.cshtml', '\_Layout.cshtml\*', 'Index.cshtml', and 'appsettings.json'. The status bar at the bottom indicates 'Any CPU' and 'EventEasePOE'. A message bar at the top left says 'Feature 'Edit and Continue' is currently unavailable due to an internal error.' The Solution Explorer on the right shows the project structure with files like 'BookingController.cs', 'EventsController.cs', 'HomeController.cs', 'VenuesController.cs', 'ApplicationDbContext.cs', 'Booking.cs', 'ErrorViewModel.cs', 'Event.cs', and 'Venue.cs'.

```
1  namespace EventEasePOE.Models
2  {
3      public class Event
4      {
5          public int EventId { get; set; }
6          public string EventName { get; set; }
7          public DateTime EventDate { get; set; }
8          public string Description { get; set; }
9          public int VenueId { get; set; }
10     }
11 }
12 
```

The screenshot shows the Visual Studio IDE with the 'Venue.cs' file open in the code editor. The code defines a 'Venue' class with properties: VenueId, VenueName, Location, Capacity, and ImageUrl. The 'VenueId' property is annotated with a 'VenueId' attribute. The code editor interface includes tabs for 'Event.cs', 'Booking.cs', 'ApplicationDbContext.cs', 'Index.cshtml', 'Index.cshtml', 'Layout.cshtml\*', and 'appsettings.json'. The status bar at the bottom indicates 'Any CPU' and 'EventEasePOE'. A message bar at the top left says 'Feature 'Edit and Continue' is currently unavailable due to an internal error.' The Solution Explorer on the right shows the project structure with files like 'BookingController.cs', 'EventsController.cs', 'HomeController.cs', 'VenuesController.cs', 'ApplicationDbContext.cs', 'Booking.cs', 'ErrorViewModel.cs', 'Event.cs', 'Venue.cs', and 'appsettings.json'.

```
1  namespace EventEasePOE.Models
2  {
3      public class Venue
4      {
5          public int VenueId { get; set; }
6          public string VenueName { get; set; }
7          public string Location { get; set; }
8          public int Capacity { get; set; }
9          public string ImageUrl { get; set; }
10     }
11 }
12 
```

The screenshot shows the Visual Studio IDE with the VenuesController.cs file open in the code editor. The code defines a controller for managing venues. It includes methods for listing all venues and retrieving a specific venue by ID. The controller uses an ApplicationDbContext context to interact with the database.

```
10  namespace EventEasePOE.Controllers
11  {
12      [ApiController]
13      public class VenuesController : Controller
14      {
15          private readonly ApplicationDbContext _context;
16
17          public VenuesController(ApplicationDbContext context)
18          {
19              _context = context;
20          }
21
22          // GET: Venues
23          public async Task<IActionResult> Index()
24          {
25              return _context.Venues != null ?
26                  View(await _context.Venues.ToListAsync()) :
27                  Problem("Entity set 'ApplicationDbContext.Venues' is null.");
28          }
29
30          // GET: Venues/Details/5
31          public async Task<IActionResult> Details(int? id)
32          {
33              if (id == null || _context.Venues == null)
34              {
35                  return NotFound();
36              }
37
38              var venue = await _context.Venues
39                  .FirstOrDefaultAsync(m => m.VenueId == id);
40              if (venue == null)
41              {
42                  return NotFound();
43              }
44
45              return View(venue);
46          }
47      }
48  }
```

The screenshot shows the Visual Studio IDE with the EventsController.cs file open in the code editor. The code defines a controller for managing events. It includes methods for listing all events and retrieving a specific event by ID. The controller uses an ApplicationDbContext context to interact with the database.

```
10  namespace EventEasePOE.Controllers
11  {
12      [ApiController]
13      public class EventsController : Controller
14      {
15          private readonly ApplicationDbContext _context;
16
17          public EventsController(ApplicationDbContext context)
18          {
19              _context = context;
20          }
21
22          // GET: Events
23          public async Task<IActionResult> Index()
24          {
25              return _context.Events != null ?
26                  View(await _context.Events.ToListAsync()) :
27                  Problem("Entity set 'ApplicationDbContext.Events' is null.");
28          }
29
30          // GET: Events/Details/5
31          public async Task<IActionResult> Details(int? id)
32          {
33              if (id == null || _context.Events == null)
34              {
35                  return NotFound();
36              }
37
38              var @event = await _context.Events
39                  .FirstOrDefaultAsync(m => m.EventId == id);
40              if (@event == null)
41              {
42                  return NotFound();
43              }
44
45              return View(@event);
46          }
47      }
48  }
```

The screenshot shows the Visual Studio IDE with the BookingsController.cs file open in the code editor. The code defines a controller for managing bookings. It includes methods for listing all bookings and retrieving a specific booking by ID. The controller uses an ApplicationDbContext context to interact with the database.

```
10  namespace EventEasePOE.Controllers
11  {
12      [ApiController]
13      public class BookingsController : Controller
14      {
15          private readonly ApplicationDbContext _context;
16
17          public BookingsController(ApplicationDbContext context)
18          {
19              _context = context;
20          }
21
22          // GET: Bookings
23          public async Task<IActionResult> Index()
24          {
25              return _context.Bookings != null ?
26                  View(await _context.Bookings.ToListAsync()) :
27                  Problem("Entity set 'ApplicationDbContext.Bookings' is null.");
28          }
29
30          // GET: Bookings/Details/5
31          public async Task<IActionResult> Details(int? id)
32          {
33              if (id == null || _context.Bookings == null)
34              {
35                  return NotFound();
36              }
37
38              var booking = await _context.Bookings
39                  .FirstOrDefaultAsync(m => m.BookingId == id);
40              if (booking == null)
41              {
42                  return NotFound();
43              }
44
45              return View(booking);
46          }
47      }
48  }
```

# Screenshots of ssms

The image consists of three vertically stacked screenshots of Microsoft SQL Server Management Studio (SSMS) showing the design of three tables: Bookings, Events, and Venue.

**Top Screenshot: Bookings Table**

Laptop-TKTPFF6\\$SQLEXPRESS.EventEasedata - dbo.Bookings - Microsoft SQL Server Management Studio

Object Explorer shows the database structure:

- DataCon
- DatCraft\_db
- EventEasedata
- Tables
  - System Tables
  - FileTables
  - External Tables
  - Graph Tables
  - dbo..EFMigrat
  - dbo.Bookings
    - Columns
    - Keys
    - Constraints
    - Triggers
    - Indexes
    - Statistics

Table Designer for Bookings:

Column Name	Data Type	Allow Nulls
BookingId	int	<input type="checkbox"/>
EventId	int	<input checked="" type="checkbox"/>
VenueId	int	<input checked="" type="checkbox"/>
BookingDate	datetime	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

**Middle Screenshot: Events Table**

Laptop-TKTPFF6\\$SQLEXPRESS.EventEasedata - dbo.Events - Microsoft SQL Server Management Studio

Object Explorer shows the database structure:

- DataCon
- DatCraft\_db
- EventEasedata
- Tables
  - System Tables
  - FileTables
  - External Tables
  - Graph Tables
  - dbo..EFMigrat
  - dbo.Booking
  - dbo.Events
    - Columns
    - Keys
    - Constraints
    - Triggers
    - Indexes
    - Statistics

Table Designer for Events:

Column Name	Data Type	Allow Nulls
EventId	int	<input type="checkbox"/>
EventName	nvarchar(100)	<input checked="" type="checkbox"/>
EventDate	datetime	<input checked="" type="checkbox"/>
Description	nvarchar(255)	<input checked="" type="checkbox"/>
VenueId	int	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

Column Properties for EventId:

- (General)
  - (Name) EventId
  - Allow Nulls No
  - Data Type int
- Table Designer
  - (General)

**Bottom Screenshot: Venue Table**

Laptop-TKTPFF6\\$SQLEXPRESS.EventEasedata - dbo.Venue - Microsoft SQL Server Management Studio

Object Explorer shows the database structure:

- DataCon
- DatCraft\_db
- EventEasedata
- Tables
  - System Tables
  - FileTables
  - External Tables
  - Graph Tables
  - dbo..EFMigrat
  - dbo.Booking
  - dbo.Events
  - dbo.Venue
    - Columns
    - Keys
    - Constraints
    - Triggers
    - Indexes
    - Statistics

Table Designer for Venue:

Column Name	Data Type	Allow Nulls
VenueId	int	<input type="checkbox"/>
VenueName	nvarchar(100)	<input checked="" type="checkbox"/>
Location	nvarchar(100)	<input checked="" type="checkbox"/>
Capacity	int	<input checked="" type="checkbox"/>
ImageUrl	nvarchar(255)	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

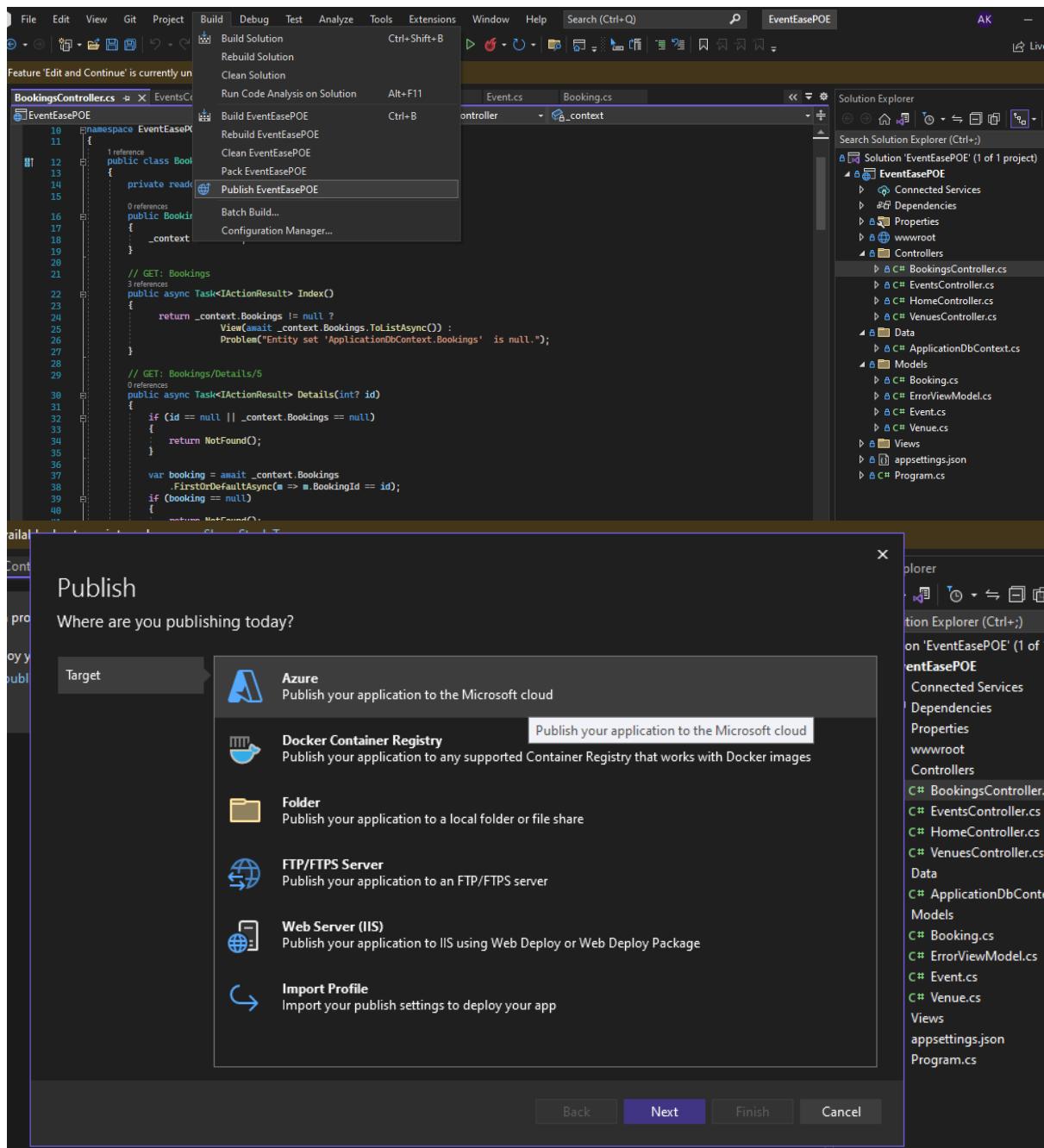
Column Properties for VenueId:

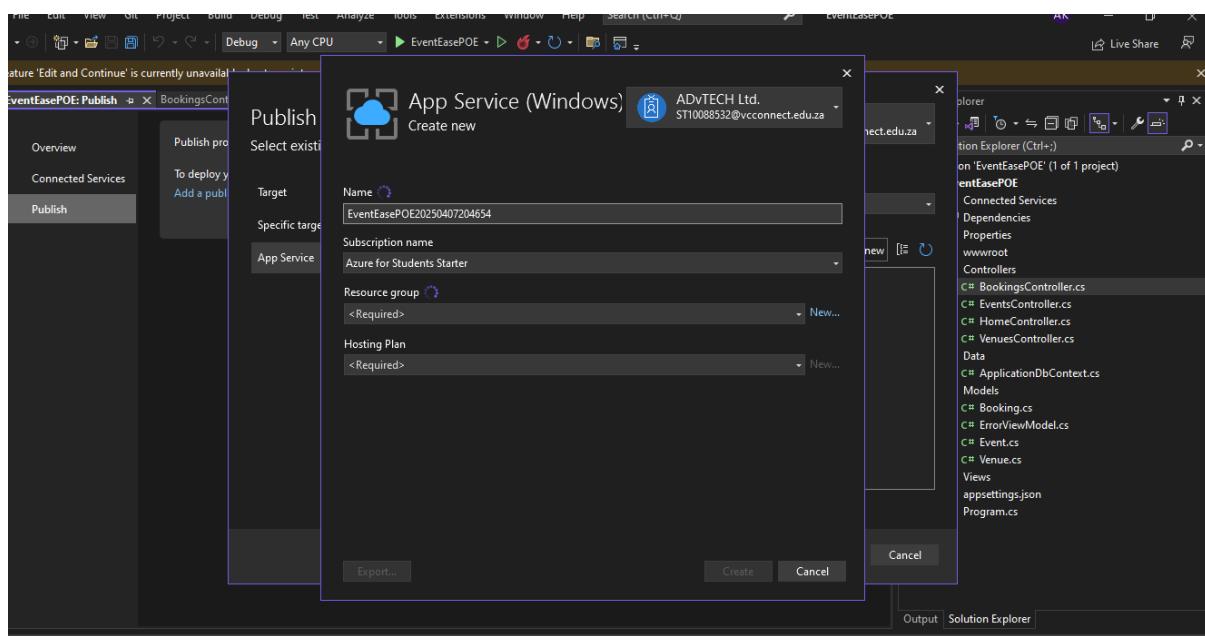
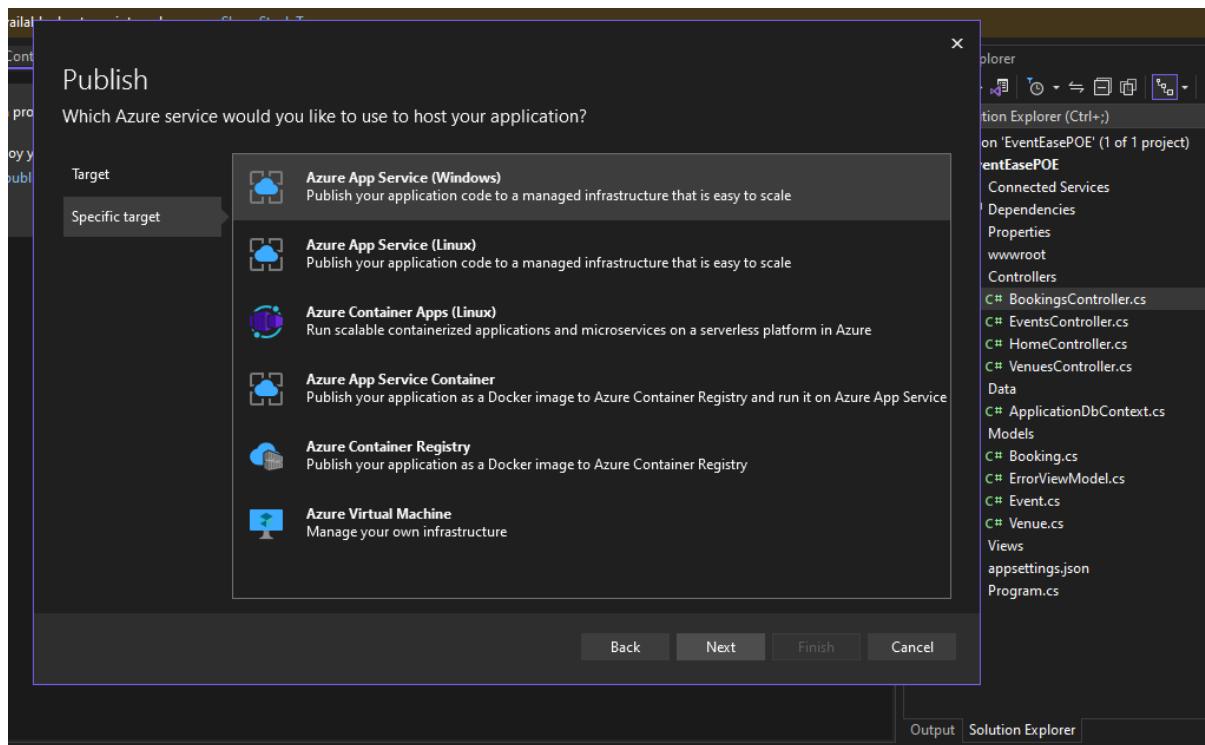
- (General)
  - (Name) VenueId
  - Allow Nulls No
  - Data Type int
- Table Designer
  - (General)

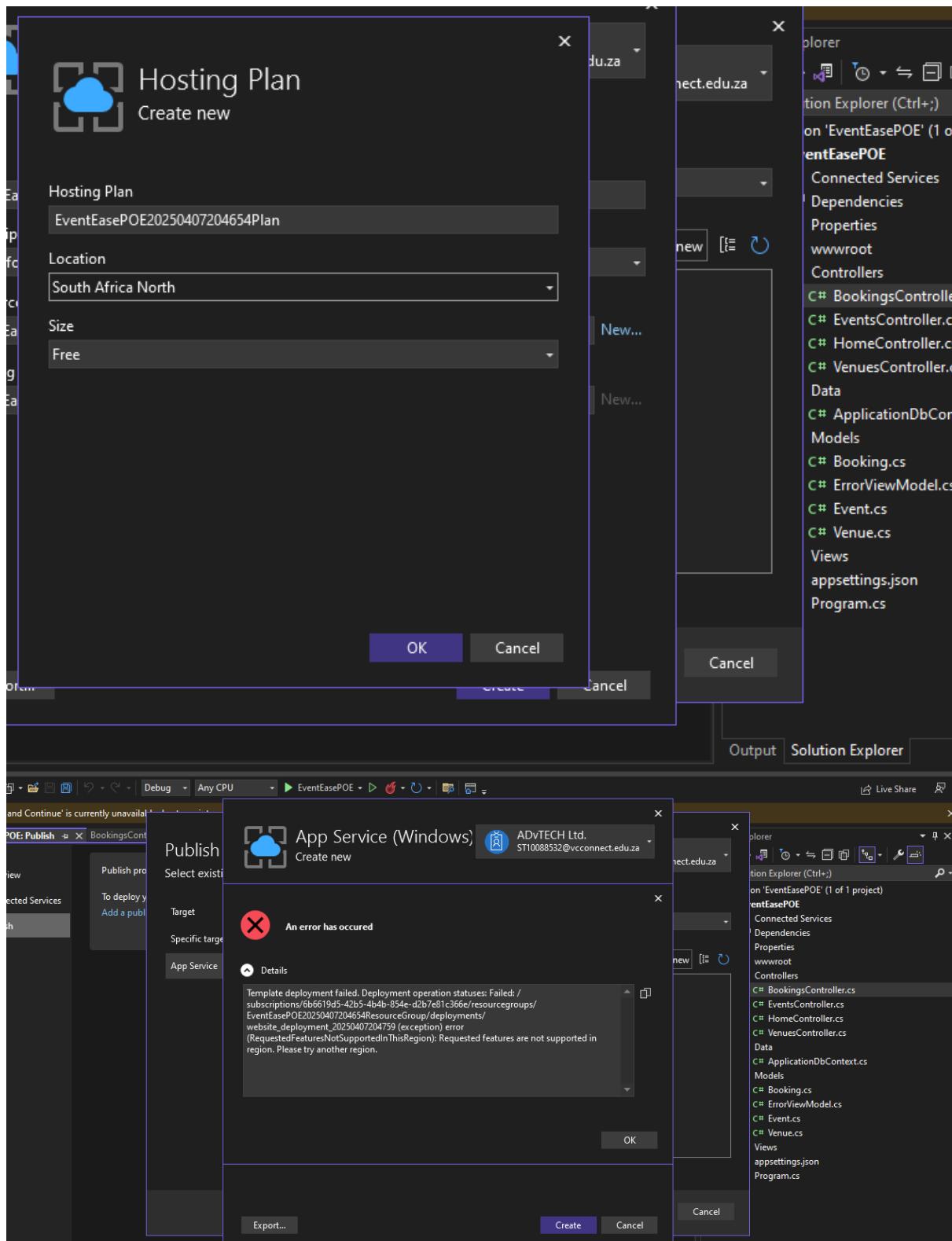
The screenshot shows the Microsoft SQL Server Management Studio interface. The title bar indicates the connection is to 'LAPTOP-TKTPFF6S\SQLEXPRESS.EventEasedata - dbo.Venues'. The main area displays the 'Table Designer' for the 'Venues' table. The 'Columns' pane lists five columns: 'VenueId' (int, Allow Nulls checked), 'VenueName' (nvarchar(100), Allow Nulls checked), 'Location' (nvarchar(100), Allow Nulls checked), 'Capacity' (int, Allow Nulls checked), and 'ImageUrl' (nvarchar(255), Allow Nulls checked). Below the table definition, the 'Column Properties' pane shows the properties for 'VenueId': Name is 'VenueId', Allow Nulls is 'No', Data Type is 'int', and Default Value or Binding is empty. The 'Table Designer' pane shows '(General)' under 'Table Designer'.

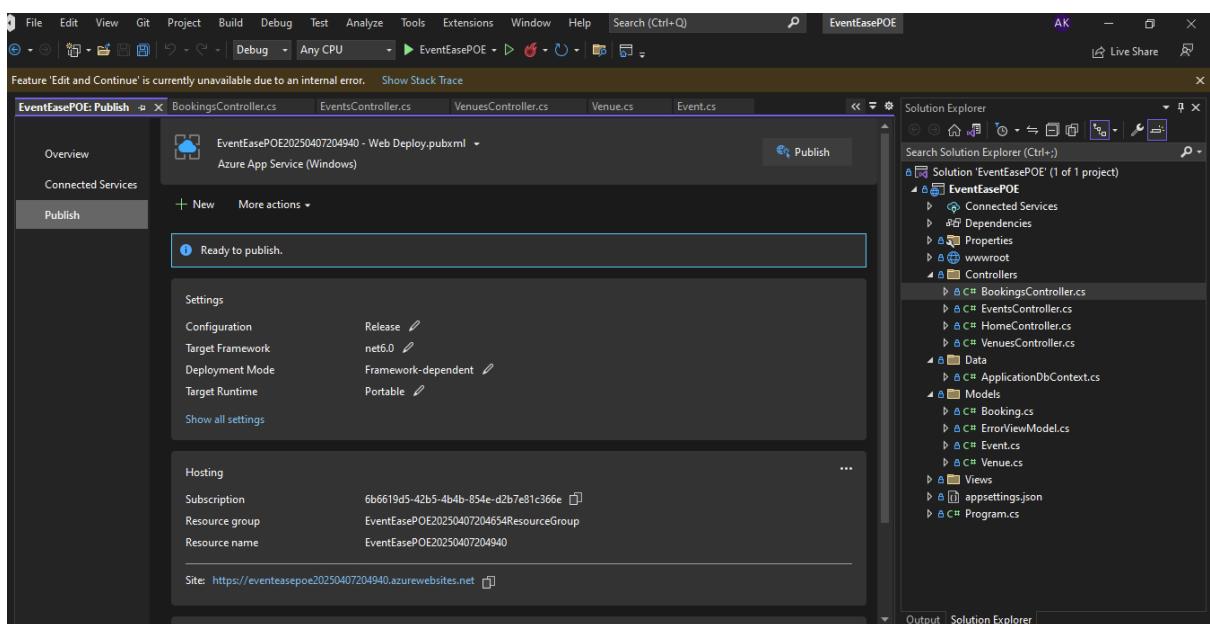
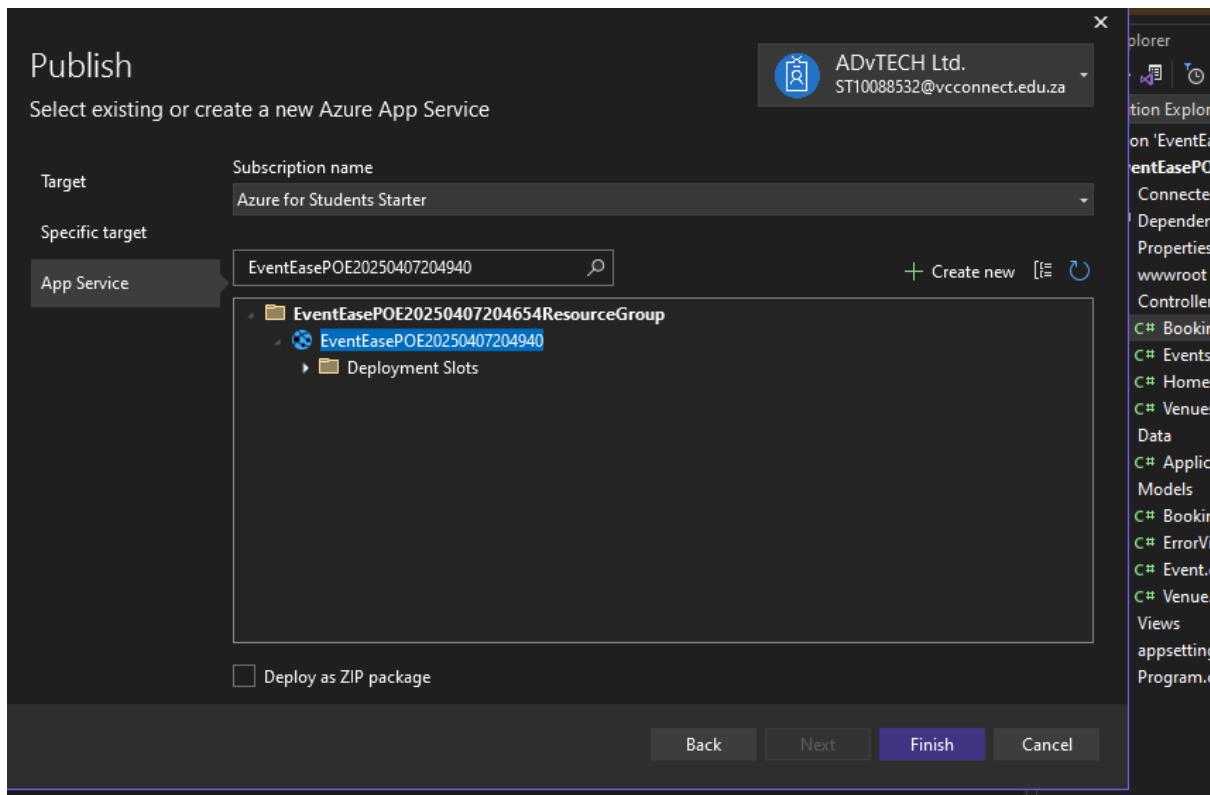
The screenshot shows the Microsoft SQL Server Management Studio interface. The title bar indicates the connection is to 'LAPTOP-TKTPFF6S\SQLEXPRESS'. The main area displays the 'Object Explorer' pane. Under the 'EventEasedata' database, the 'Tables' node is expanded, showing 'Tables (expanding...)'. Other nodes visible include 'Databases', 'Views', 'External Resources', 'Synonyms', 'Programmability', 'Service Broker', 'Storage', and 'Security'. The 'past paper 2021' and 'scholarship.db' databases are also listed at the bottom.

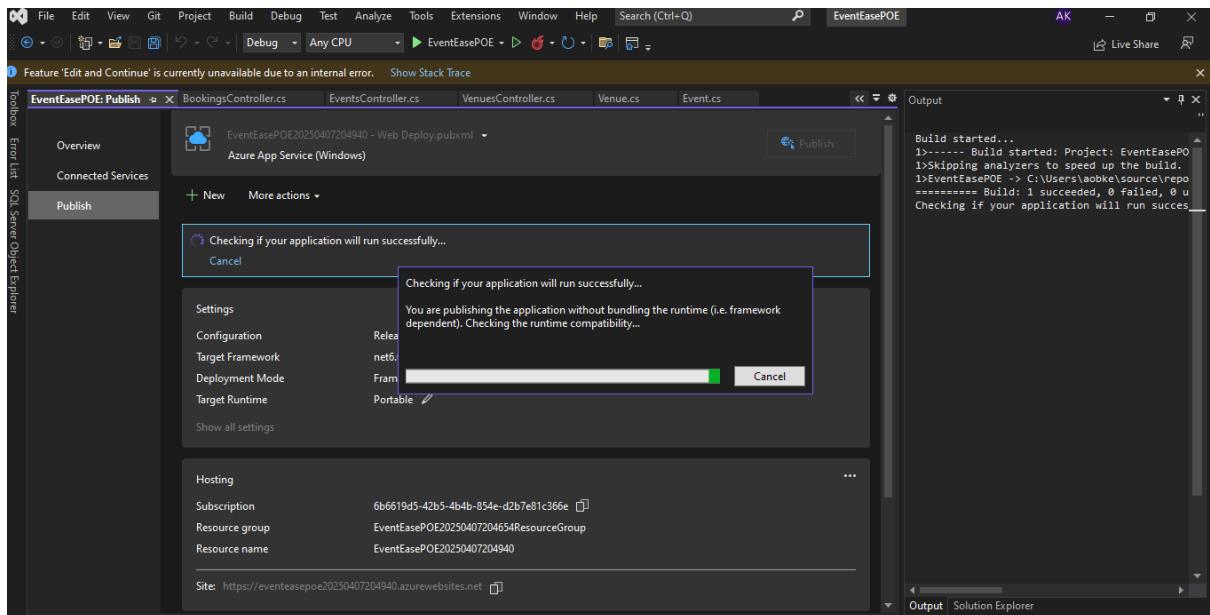
## Screenshots of deployment









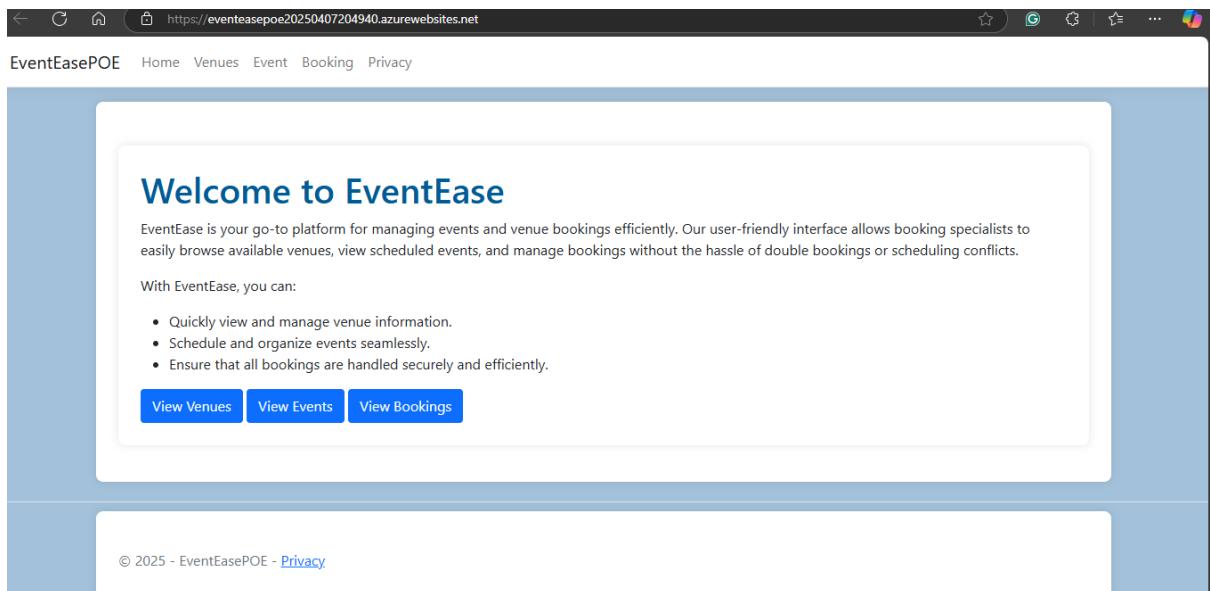


## GITHub repository link

<https://github.com/ST10088532/VCSANDTON-CLDV6211-ST10088532.git>

## Web app url

<https://eventeasepoe20250407204940.azurewebsites.net>



## Part D

1. On-premise software is installed on a company's network/infrastructure.  
Cloud software relies on a provider's servers and is accessed using the web.  
While cloud computing offers flexibility and cost-efficiency, on-premise

options offer dependability, security, and control. Mostly a blend of on-premise, cloud and SaaS applications is necessary for continued business growth.

Example, If EventEase decided to deploy on-premises they would need to hire cybersecurity employees/workers to setup firewalls, check the network and system for any threats and make sure the physical server secure.

2. IaaS is an on-demand access to cloud-hosted physical and virtual servers, storage and networking—the backend IT infrastructure for running applications and workloads in the cloud. PaaS is on-demand access to a complete, ready-to-use, cloud-hosted platform for developing, running, maintaining and managing applications.

SaaS, or software as a service, is on-demand access to ready-to-use cloud-hosted application software. (Mrzyglod, 2022) (Ron Karjian, 2024) (S., n.d.)

Example EventEase use IaaS so they need to configure virtual machines, install web servers and manage Operating system patches/updates. With SaaS they would have to adapt their business processes to fit available software. With PaaS comes the perfect balance by using Azure App Service and Azure SQL Database, EventEase can deploy their event booking app, while maintaining control over their application's functionality and user experience. (Mrzyglod, 2022) (Ron Karjian, 2024) (S., n.d.)

## Part 2

Azure database creation:

**Create SQL Database Server**

Microsoft

⚠️ Changing Basic options may reset selections you have made. Review all options prior to creating the resource.

Azure Active Directory (Azure AD) is now Microsoft Entra ID. [Learn more](#)

Select your preferred authentication methods for accessing this server. Create a server admin login and password to access your server with SQL authentication, select only Microsoft Entra authentication [Learn more](#) using an existing Microsoft Entra user, group, or application as Microsoft Entra admin [Learn more](#), or select both SQL and Microsoft Entra authentication.

Authentication method:  Use SQL authentication

Server admin login \*: easeevent\_data

Password \*: .....  
Confirm password \*: .....

**Review + create** **Next : Networking >**

**Create SQL Database Server**

Microsoft

SQL Database Server by Microsoft

Estimated cost per month: No additional charges

Terms: By clicking "Create", I (a) agree to the legal terms and privacy statement(s) associated with the Marketplace offering(s) listed above; (b) authorize Microsoft to bill my current payment method for the fees associated with the offering(s), with the same billing frequency as my Azure subscription; and (c) agree that Microsoft may share my contact, usage and transactional information with the provider(s) of the offering(s) for support, billing and other transactional activities. Microsoft does not provide rights for third-party offerings. For additional details see [Azure Marketplace Terms](#).

**Basics**

Subscription	ADVTECH-Tertiary Vega School
Resource group	AZ-JHB-RSG-VCGPSD-ST10088532-TER
Server name	easeevents
Authentication method	SQL authentication
Server admin login	easeevent_data
Location	South Africa North

**Deployment**

Microsoft.SQLServer.createServer\_a8171998f07748fcaaf39955b340a0c | Overview

Deployment is in progress

Deployment name : Microsoft.SQLServer.createServer\_a81... Start time : 5/13/2025, 12:36:32 AM

Subscription : ADVTECH-Tertiary Vega School Correlation ID : f62ac2d1-0f31-434a-897d-0cd596eb7...

Resource group : AZ-JHB-RSG-VCGPSD-ST10088532-TER

**Deployment details**

Resource	Type	Status	Operation details
There are no resources to display.			

**Microsoft Defender for Cloud**  
Secure your apps and infrastructure  
[Go to Microsoft Defender for Cloud](#)

**Free Microsoft tutorials**  
[Start learning today](#)

**Work with an expert**  
Azure experts are service provider partners who can help manage your assets on Azure and be your first line of support.  
[Find an Azure expert](#)

Microsoft Azure easeevent\_data Copilot Home > SQL databases Create SQL Database Microsoft

Basics Networking Security Additional settings Tags Review + create

**Product details**

SQL database by Microsoft Terms of use | Privacy policy

**Estimated cost**

Storage cost 0.00 USD / month + Compute cost 0.000000 USD / vCore second

**Terms**

By clicking "Create", I (a) agree to the legal terms and privacy statement(s) associated with the Marketplace offering(s) listed above; (b) authorize Microsoft to bill my current payment method for the fees associated with the offering(s), with the same billing frequency as my Azure subscription; and (c) agree that Microsoft may share my contact, usage and transactional information with the provider(s) of the offering(s) for support, billing and other transactional activities. Microsoft does not provide rights for third-party offerings. For additional details see [Azure Marketplace Terms](#).

**Cost summary**

General Purpose (GP_S_Gen5_2)	0.00
Cost per GB (in USD)	0.00
Max storage selected (in GB)	x 41.6
First 32 GB storage free	
First 100,000 vCore seconds free	
Overage billing <sup>1</sup>	Disabled
ESTIMATED STORAGE COST / MONTH	0.00 USD
COMPUTE COST / VCORE SECOND <sup>2</sup>	0.000000 USD

**Basics**

Subscription ADvTECH-Tertiary Vega School

Create < Previous Download a template for automation

ST10088532@vcconnect... ADVTECH LTD. (ADVTECHONLINE)

Microsoft Azure easeevent\_data Copilot Home > SQL databases Create SQL Database Microsoft

Basics Networking Security Additional settings Tags Review + create

**Product details**

SQL database by Microsoft Terms of use | Privacy policy

**Estimated cost per month**

7.51 USD

**Terms**

By clicking "Create", I (a) agree to the legal terms and privacy statement(s) associated with the Marketplace offering(s) listed above; (b) authorize Microsoft to bill my current payment method for the fees associated with the offering(s), with the same billing frequency as my Azure subscription; and (c) agree that Microsoft may share my contact, usage and transactional information with the provider(s) of the offering(s) for support, billing and other transactional activities. Microsoft does not provide rights for third-party offerings. For additional details see [Azure Marketplace Terms](#).

**Cost summary**

Basic (Basic)	1.50
Cost per DTU (in USD)	1.50
DTUs selected	x 5
ESTIMATED COST / MONTH	7.51 USD

**Basics**

Subscription ADvTECH-Tertiary Vega School

Resource group AZ-JHB-RSG-VCGPSD-ST10088532-TER

Create < Previous Download a template for automation

ST10088532@vcconnect... ADVTECH LTD. (ADVTECHONLINE)

**Microsoft Azure - Microsoft.SQLDatabase.newDatabaseNewServer\_4fb38a5c134d49aca2491 | Overview**

**Deployment**

**Deployment is in progress**

Deployment name : Microsoft.SQLDatabase.newDatabase... Start time : 5/13/2025, 12:50:24 AM

Subscription : ADVTECH-Tertiary Vega School Correlation ID : 4dc9b480-fa66-4eee-8e7c-a385095c3...

Resource group : AZ-JHB-RSG-VCGPSD-ST10088532-TER

**Deployment details**

Resource	Type	Status	Operation details
eventeasedat	Microsoft.Sql/servers	Accepted	<a href="#">Operation details</a>

**Microsoft Defender for Cloud**  
Secure your apps and infrastructure  
[Go to Microsoft Defender for Cloud >](#)

**Free Microsoft tutorials**  
[Start learning today >](#)

**Work with an expert**  
Azure experts are service provider partners who can help manage your assets on Azure and be your first line of support.  
[Find an Azure expert >](#)

**Microsoft Azure - Microsoft.SQLDatabase.newDatabaseExistingServer\_beda964bd6894d9d | Overview**

**Deployment**

**Your deployment is complete**

Deployment name : Microsoft.SQLDatabase.newDatabase... Start time : 5/13/2025, 12:53:39 AM

Subscription : Azure for Students Starter Correlation ID : 467796d1-87fc-4301-ad4a-9f96b0a5d...

Resource group : EventEasePOE20250407204654Resour...

**Deployment details**

**Next steps**

[Go to resource](#)

**Cost management**  
Get notified to stay within your budget and prevent unexpected charges on your bill.  
[Set up cost alerts >](#)

**Microsoft Defender for Cloud**  
Secure your apps and infrastructure  
[Go to Microsoft Defender for Cloud >](#)

**Free Microsoft tutorials**  
[Start learning today >](#)

## Code Corrections:

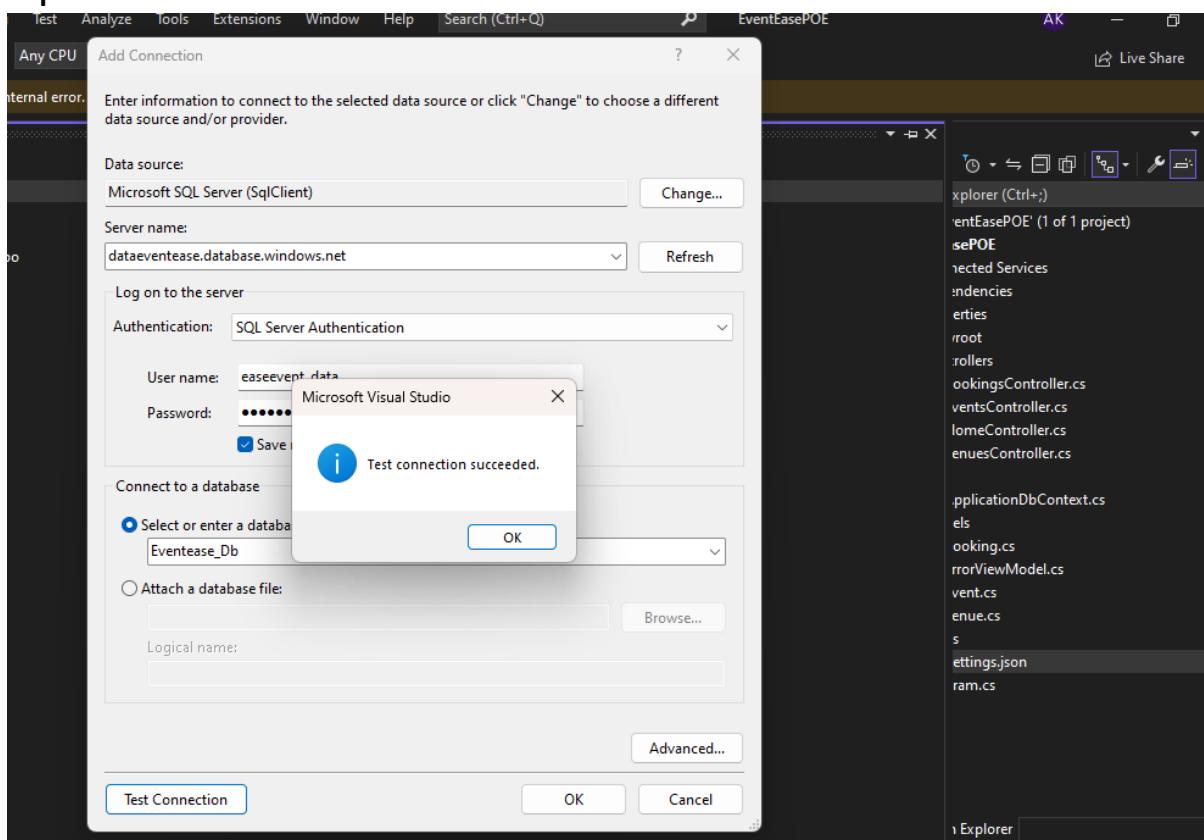
Updated json file with azure connection string.

```

{
  "Logging": {
    "LogLevel": {
      "Default": "Information",
      "Microsoft.AspNetCore": "Warning"
    }
  },
  "AllowedHosts": "*",
  "ConnectionStrings": {
    "DefaultConnection": "Server=tcp:database.windows.net,1433;Initial Catalog=EventEase_Db;Persist Security Info=False;User ID=EventEase;Password=EventEase@123;MultipleActiveResultSets=true"
  }
}

```

## Updated connection to azure:



## Adding Migrations and Updating Database:

The screenshot shows the Visual Studio interface with the Package Manager Console and Solution Explorer open.

**Package Manager Console:**

```
PM> Add-Migration SecondCreate
Build started...
Build succeeded.

PM> Update-Database
Build started...
Build succeeded.

Microsoft.EntityFrameworkCore.Infrastructure[10483]
  Entity Framework Core 6.0.36 initialized 'ApplicationDbContext' using provider 'Microsoft.EntityFrameworkCore.SqlServer:6.0.36' with options: None
Microsoft.EntityFrameworkCore.Database.Command[20101]
  Executed DbCommand (26ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    SELECT ...
Microsoft.EntityFrameworkCore.Database.Command[20101]
  Executed DbCommand (2ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    SELECT OBJECT_ID(N'[__EFMigrationsHistory']");
Microsoft.EntityFrameworkCore.Database.Command[20101]
  Executed DbCommand (1ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    SELECT ...
Microsoft.EntityFrameworkCore.Database.Command[20101]
  Executed DbCommand (17ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    CREATE TABLE [__EFMigrationsHistory] (
      [MigrationId] nvarchar(190) NOT NULL,
      [ProductVersion] nvarchar(32) NOT NULL,
      CONSTRAINT [PK__EFMigrationsHistory] PRIMARY KEY ([MigrationId])
    );
Microsoft.EntityFrameworkCore.Database.Command[20101]
  Executed DbCommand (0ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    SELECT 1
Microsoft.EntityFrameworkCore.Database.Command[20101]
  Executed DbCommand (1ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    SELECT OBJECT_ID(N'[__EFMigrationsHistory']");
Microsoft.EntityFrameworkCore.Database.Command[20101]
  Executed DbCommand (2ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    SELECT [MigrationId], [ProductVersion]
    ORDER BY [MigrationId];
Microsoft.EntityFrameworkCore.Migrations[20402]
  Applying migration '20230513081203_SecondCreate'.
  Applying migration '20230513081203_SecondCreate'.
Microsoft.EntityFrameworkCore.Database.Command[20101]
  Executed DbCommand (0ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    CREATE TABLE [Bookings] (
      [BookingId] int NOT NULL IDENTITY,
      [EventId] int NOT NULL,
      [VenueId] int NOT NULL,
      [BookingDate] datetime2 NOT NULL,
      CONSTRAINT [PK_Bookings] PRIMARY KEY ([BookingId])
    );
Microsoft.EntityFrameworkCore.Database.Command[20101]
  Executed DbCommand (0ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    CREATE TABLE [Events] (
      [EventId] int NOT NULL IDENTITY,
      [EventName] nvarchar(max) NOT NULL,
```

**Solution Explorer:**

- on EventEasePOE (1 of 1 project)
  - EventEasePOE
    - Connected Services
    - Dependencies
    - Properties
    - wwwroot
    - Controllers
      - C# BookingsController.cs
      - C# EventsController.cs
      - C# HomeController.cs
      - C# VenuesController.cs
    - Data
      - C# ApplicationDbContext.cs
    - Models
      - C# Booking.cs
      - C# ErrorViewModel.cs
    - C# Event.cs
    - Views
      - A C# Venue.cs
  - appsettings.json

## A. Creating Azure Storage Account

[https://portal.azure.com/#view/Microsoft\\_Azure\\_Marketplace/PlusNew.ReactView/package/hub/additionalConfig~/%7B%7Dselected...](https://portal.azure.com/#view/Microsoft_Azure_Marketplace/PlusNew.ReactView/package/hub/additionalConfig~/%7B%7Dselected...)

**Create a resource**

Popular Azure services See more in All services

- Storage account Create
- Azure Cosmos DB Create
- Azure Cosmos DB for MongoDB Create
- Azure File Sync Create
- Storage task - Azure Storage Actions Create
- Azure NetApp Files Create

Popular Marketplace products See more in Marketplace

- SUSE Linux Enterprise Server for SAP Applications 12 SP5 - BYOS Create | Learn more
- Standard - Full Access Create | Learn more
- Veeam Backup for Microsoft Azure Free Edition v6 Create | Learn more
- Flex 1-50 users — Annual Subscription Set up + subscribe | Learn more
- ARM - Azure Storage Account Create | Learn more
- Standard Create | Learn more

[https://portal.azure.com/#view/Microsoft\\_Azure\\_Marketplace/PlusNew.ReactView/package/hub/additionalConfig~/%7B%7Dselected...](https://portal.azure.com/#view/Microsoft_Azure_Marketplace/PlusNew.ReactView/package/hub/additionalConfig~/%7B%7Dselected...)

**Create a storage account**

Review + create

Basics Advanced Networking Data protection Encryption Tags

View automation template

**Basics**

Subscription	ADvTECH-Tertiary Vega School
Resource group	AZ-JHB-RSG-VCGPSD-ST10088532-TER
Location	Central US
Storage account name	easeimages
Primary service	
Performance	Standard
Replication	Locally-redundant storage (LRS)

**Advanced**

Enable hierarchical namespace	Disabled
Enable SFTP	Disabled
Enable network file system v3	Disabled

<https://portal.azure.com/#view/HubsExtension/DeploymentDetailsBlade/~/overview/id/%2Fsubscriptions%2Fd1788afb-2254-4f5f-882...>

**Deployment**

easeimages\_1747128964596 | Overview

Deployment succeeded

Deployment 'easeimages\_1747128964596' to resource group 'AZ-JHB-RSG-VCGPSD-ST10088532-TER' was successful.

Go to resource Pin to dashboard

**Overview**

Your deployment is complete

Deployment name: easeimages\_1747128964596  
Subscription: ADvTECH-Tertiary Vega School  
Resource group: AZ-JHB-RSG-VCGPSD-ST10088532-...

Start time: 5/13/2025, 11:36:31 AM  
Correlation ID: 0283a95c-2dfd-4a26-be93-fe35156fb42c

Deployment details  
Next steps

Give feedback  
Tell us about your experience with deployment

Cost Management  
Get notified to stay within your budget and prevent unexpected charges on your bill.  
Set up cost alerts >

Microsoft Defender for Cloud  
Secure your apps and infrastructure  
Go to Microsoft Defender for Cloud >

Free Microsoft tutorials  
Start learning today >

Work with an expert

# Integration of Azure Storage:

Process: [21500] EventEasePOE.exe

Feature 'Symbol search' is currently unavailable due to an internal error. Show Stack Trace

Server Explorer

Solution Explorer

Search Solution Explorer (Ctrl+.)

Create.cshtml Blobstorage.cs Layout.cshtml Index.cshtml Index.cshtml appsettings.json

Schema: <http://json.schemastore.org/appsettings.json>

```

1  {
2     "Logging": {
3         "LogLevel": {
4             "Default": "Information",
5             "Microsoft.AspNetCore": "Warning"
6         }
7     },
8     "AllowedHosts": "*",
9     "ConnectionStrings": {
10        "DefaultConnection": "Server=tcp:dataeventease.database.windows.net,1433;Initial Catalog=EventEase_DB;Persist Security I"
11    },
12    "AzureStorage": {
13        "ConnectionString": "DefaultEndpointsProtocol=https;AccountName=easeimages;AccountKey=g5J4Jg3D+ApVQloZM1LekTkFbVg3oyd",
14        "ContainerName": "venue-images"
15    }
16 }
17

```

Error List

Entire Solution

No issues found

Code Description Project File Line Suppres

Non-nullable property 'Venues' must contain a non-null value when exiting

Process: [21500] EventEasePOE.exe

Feature 'Symbol search' is currently unavailable due to an internal error. Show Stack Trace

Server Explorer

Solution Explorer

Search Solution Explorer (Ctrl+.)

Create.cshtml Blobstorage.cs Layout.cshtml Index.cshtml Index.cshtml appsettings.json

EventEasePOE

```

1  using Azure.Storage.Blobs;
2  using Azure.Storage.Blobs.Models;
3  using Microsoft.AspNetCore.Http;
4
5  namespace EventEasePOE.Functions
6  {
7      public class Blobstorage
8      {
9          private readonly BlobServiceClient _blobServiceClient;
10         private readonly string _containerName;
11
12         public Blobstorage(string connectionString, string containerName)
13         {
14             _blobServiceClient = new BlobServiceClient(connectionString);
15             _containerName = containerName;
16         }
17
18         public async Task<string> UploadImageAsync(IFormFile file)
19         {
20             if (file == null || file.Length == 0)
21                 return null;
22
23             var containerClient = _blobServiceClient.GetBlobContainerClient(_containerName);
24             await containerClient.CreateIfNotExistsAsync();
25
26             // Create a unique name for the blob
27             var blobName = $"{Guid.NewGuid()}{Path.GetExtension(file.FileName)}";
28             var blobClient = containerClient.GetBlobClient(blobName);
29
30             // Upload the file
31             using (var stream = file.OpenReadStream())
32             {
33                 await blobClient.UploadAsync(stream, new BlobHttpHeaders { ContentType = file.ContentType });
34             }
35
36             return blobClient.Uri.ToString();
37         }
38
39         public async Task DeleteImageAsync(string blobUrl)
40         {
41             if (string.IsNullOrEmpty(blobUrl))
42                 return;
43
44             var containerClient = _blobServiceClient.GetBlobContainerClient(_containerName);
45             var blobName = Path.GetFileName(new Uri(blobUrl).LocalPath);
46             var blobClient = containerClient.GetBlobClient(blobName);
47
48             await blobClient.DeleteIfExistsAsync();
49         }
50     }
51 }

```

Error List

Entire Solution

No issues found

Code Description Project File Line Suppres

Non-nullable property 'Venues' must contain a non-null value when exiting

Process: [21500] EventEasePOE.exe

Feature 'Symbol search' is currently unavailable due to an internal error. Show Stack Trace

Server Explorer

Solution Explorer

Search Solution Explorer (Ctrl+.)

Create.cshtml Blobstorage.cs Layout.cshtml Index.cshtml Index.cshtml appsettings.json

EventEasePOE

```

28         if (file == null || file.Length == 0)
29             return null;
30
31         var containerClient = _blobServiceClient.GetBlobContainerClient(_containerName);
32         await containerClient.CreateIfNotExistsAsync();
33
34         // Create a unique name for the blob
35         var blobName = $"{Guid.NewGuid()}{Path.GetExtension(file.FileName)}";
36         var blobClient = containerClient.GetBlobClient(blobName);
37
38         // Upload the file
39         using (var stream = file.OpenReadStream())
40         {
41             await blobClient.UploadAsync(stream, new BlobHttpHeaders { ContentType = file.ContentType });
42         }
43
44         return blobClient.Uri.ToString();
45     }
46
47     public async Task DeleteImageAsync(string blobUrl)
48     {
49         if (string.IsNullOrEmpty(blobUrl))
50             return;
51
52         var containerClient = _blobServiceClient.GetBlobContainerClient(_containerName);
53         var blobName = Path.GetFileName(new Uri(blobUrl).LocalPath);
54         var blobClient = containerClient.GetBlobClient(blobName);
55
56         await blobClient.DeleteIfExistsAsync();
57     }
58 }

```

Error List

Entire Solution

No issues found

Code Description Project File Line Suppres

Non-nullable property 'Venues' must contain a non-null value when exiting

## B. Implementing error handling and validations

```
using System.ComponentModel.DataAnnotations;
using System.ComponentModel.DataAnnotations.Schema;

namespace EventEasePOE.Models
{
    public class Venue
    {
        [Required(ErrorMessage = "Venue Name is required.")]
        public int VenueId { get; set; }

        [Required(ErrorMessage = "Location is required.")]
        public string VenueName { get; set; }

        [Required(ErrorMessage = "Capacity is required.")]
        public int Capacity { get; set; }

        [NotMapped]
        public IFormFile ImageFile { get; set; }
    }
}
```

```
using System.ComponentModel.DataAnnotations;
using System.ComponentModel.DataAnnotations.Schema;

namespace EventEasePOE.Models
{
    public class Booking
    {
        [Required(ErrorMessage = "Booking Date is required.")]
        public DateTime BookingDate { get; set; }
    }
}
```

Feature 'Symbol search' is currently unavailable due to an internal error. Show Stack Trace

Server Explorer    Solution Explorer

Process: [12152] EventEasePOE.exe    Lifecycle Events - Thread: [41436].NET ThreadPool Worker    Stack Frame: Program.<Main>\$AnonymousMethod\_1

VenuesController.cs

```

154     return View(venue);
155 }
156
157 // POST: Venues/Delete/9
158 [HttpPost, ActionName("Delete")]
159 [ValidateAntiForgeryToken]
160 public async Task<ActionResult> DeleteConfirmed(int id)
161 {
162     if (_context.Venues == null)
163     {
164         return Problem("Entity set 'ApplicationDbContext.Venues' is null.");
165     }
166     var venue = await _context.Venues.FindAsync(id);
167     if (venue == null)
168     {
169         // Delete image from blob storage if it exists
170         if (!string.IsNullOrEmpty(venue.ImageUrl))
171         {
172             await _blobStorage.DeleteImageAsync(venue.ImageUrl);
173         }
174         _context.Venues.Remove(venue);
175     }
176     await _context.SaveChangesAsync();
177     return RedirectToAction(nameof(Index));
178 }
179
180 [Reference]
181 private bool VenueExists(int id)
182 {
183     return false;
184 }

```

Booking.cs

Venue.cs

EventEasePOE.Controllers.VenuesController

VenuesController(ApplicationDbContext context)

Live Share

Server Explorer    Solution Explorer

Process: [12152] EventEasePOE.exe    Lifecycle Events - Thread: [41436].NET ThreadPool Worker    Stack Frame: Program.<Main>\$AnonymousMethod\_1

VenuesController.cs

```

122     await _context.SaveChangesAsync();
123     catch (DbUpdateConcurrencyException)
124     {
125         if (!VenueExists(venue.VenueId))
126         {
127             return NotFound();
128         }
129         else
130         {
131             throw;
132         }
133     }
134     return RedirectToAction(nameof(Index));
135 }
136
137 return View(venue); // Return the view with validation errors
138 }
139
140 // GET: Venues/Delete/5
141 [HttpGet]
142 public async Task<ActionResult> Delete(int? id)
143 {
144     if (id == null || _context.Venues == null)
145     {
146         return NotFound();
147     }
148     var venue = await _context.Venues.FirstOrDefaultAsync(m => m.VenueId == id);
149     if (venue == null)
150     {
151         return NotFound();
152     }
153     return View(venue);
154 }

```

Booking.cs

Venue.cs

EventEasePOE.Controllers.VenuesController

VenuesController(ApplicationDbContext context)

Live Share

The screenshot shows the Visual Studio IDE with the following details:

- Process:** [12152] EventEasePOE.exe
- Lifecycle Events:** Thread: [41436].NET ThreadPool Worker
- Stack Frame:** Program.<Main>\$.AnonymousMethod\_

The code editor displays the `VenuesController.cs` file, specifically the `Edit` action method. The code is as follows:

```
// POST: Venues/Edit/5
[HttpPost]
[ValidateAntiForgeryToken]
public async Task Edit(int id, [Bind("VenueId,VenueName,Location,Capacity,ImageUrl")] Venue venue, IFor
{
    if (id != venue.VenueId)
    {
        return NotFound();
    }

    if (ModelState.IsValid)
    {
        try
        {
            if (imageFile != null && imageFile.Length > 0)
            {
                // Delete old image if it exists
                if (String.IsNullOrEmpty(venue.ImageUrl))
                {
                    await _blobStorage.DeleteImageAsync(venue.ImageUrl);
                }
                // Upload new image
                venue.ImageUrl = await _blobStorage.UploadImageAsync(imageFile);
            }

            _context.Update(venue);
            await _context.SaveChangesAsync();
        }
        catch (DbUpdateConcurrencyException)
        {
            if (!VenueExists(venue.VenueId))
            {
                return NotFound();
            }
        }
    }
}
```

The code editor highlights the line `return NotFound();` with a red squiggle under "NotFound". The status bar at the bottom left shows "81%". The status bar at the bottom right shows "Ln: 187 Ch: 2 SPC CRLF".

The Solution Explorer on the right side lists the project structure, including files like `Booking.cs`, `Venue.cs`, `EventEasePOE.Controllers.VenuesController`, and `VenuesController(ApplicationDbContext context)`.

feature 'Symbol search' is currently unavailable due to an internal error. Show Stack Trace

```
public IActionResult Create()
{
    return View();
}

// POST: Venues/Create
[HttpPost]
[ValidateAntiForgeryToken]
public async Task Create([Bind("VenueId,VenueName,Location,Capacity")] Venue venue, IFormFile imageFile)
{
    if (ModelState.IsValid)
    {
        try
        {
            if (imageFile != null && imageFile.Length > 0)
            {
                venue.ImageUrl = await _blobStorage.UploadImageAsync(imageFile);
            }

            _context.Add(venue);
            await _context.SaveChangesAsync();
            return RedirectToAction(nameof(Index));
        }
        catch (Exception ex)
        {
            ModelState.AddModelError("", "Error uploading image: " + ex.Message);
        }
    }
    return View(venue); // Return the view with validation errors
}
```

Feature 'Symbol search' is currently unavailable due to an internal error. Show Stack Trace

VenuesController.cs BookingController.cs Venue.cs EventEasePOE

```
// POST: Bookings/Create
[HttpPost]
[ValidateAntiForgeryToken]
public async Task<IActionResult> Create([Bind("BookingId,EventId,VenueId,BookingDate")] Booking booking)
{
    if (ModelState.IsValid)
    {
        // Check for double booking
        var existingBooking = await _context.Bookings
            .AnyAsync(b => b.VenueId == booking.VenueId && b.BookingDate == booking.BookingDate);

        if (existingBooking)
        {
            ModelState.AddModelError("", "This venue is already booked for the selected date and time.");
            return View(booking);
        }

        _context.Add(booking);
        await _context.SaveChangesAsync();
        return RedirectToAction(nameof(Index));
    }
}

return View(booking); // Return the view with validation errors
```

// GET: Bookings/Edit/5

```
public async Task<IActionResult> Edit(int? id)
{
    if (id == null || _context.Bookings == null)
    {
        return NotFound();
    }

    var booking = await _context.Bookings.FindAsync(id);
    if (booking == null)
```

The thread 0x8e8 has exited with code 0  
The thread 0x3d7c has exited with code 0  
The thread 0x5d29 has exited with code 0  
The thread 0x4d46 has exited with code 0  
The thread 0x7d4 has exited with code 0  
The thread 0x7780 has exited with code 0  
The thread 0x84d4 has exited with code 0  
The thread 0x9990 has exited with code 0  
The thread 0x84d4 has exited with code 0  
The thread 0x518 has exited with code 0  
The thread 0x518 has exited with code 0  
The thread 0x1170 has exited with code 0  
The thread 0x1170 has exited with code 0  
The thread 0x712c has exited with code 0  
The thread 0x91d4 has exited with code 0  
The thread 0x77c9 has exited with code 0  
The thread 0x82d4 has exited with code 0  
The thread 0x79cc has exited with code 0  
The thread 0x5f64 has exited with code 0  
The thread 0x6338 has exited with code 0  
The thread 0x205c has exited with code 0  
The thread 0x9c2c has exited with code 0  
The thread 0x58dc has exited with code 0  
The thread 0x449b has exited with code 0  
The thread 0x9930 has exited with code 0  
The thread 0x7124 has exited with code 0  
The thread 0x91d4 has exited with code 0  
The thread 0x97c4 has exited with code 0  
The thread 0x59dc has exited with code 0  
The thread 0x449c has exited with code 0  
The program '[12152] EventEasePOE.exe' has terminated with exit code 0 (0x0).

Feature 'Symbol search' is currently unavailable due to an internal error. Show Stack Trace

VenuesController.cs BookingsController.cs Booking.cs Venue.cs

EventEasePOE EventEasePOE.Controllers.BookingsController Edit(int id, Booking booking)

```

103     {
104         try
105         {
106             // Check for double booking
107             var existingBooking = await _context.Bookings
108                 .AnyAsync(b => b.VenueId == booking.VenueId && b.BookingDate == booking.BookingDate && b.BookingId != booking.BookingId);
109
110             if (existingBooking)
111             {
112                 ModelState.AddModelError("", "This venue is already booked for the selected date and time.");
113                 return View(booking);
114             }
115
116             _context.Update(booking);
117             await _context.SaveChangesAsync();
118         }
119         catch (DbUpdateConcurrencyException)
120         {
121             if (!BookingExists(booking.BookingId))
122             {
123                 return NotFound();
124             }
125             else
126             {
127                 throw;
128             }
129         }
130         return RedirectToAction(nameof(Index));
131     }
132
133     return View(booking);
134 }
135
136 // GET: Bookings/Delete/5
137 public async Task<ActionResult> Delete(int? id)
138 {
139     if (id == null || _context.Bookings == null)
140     {
141         return NotFound();
142     }

```

Feature 'Symbol search' is currently unavailable due to an internal error. Show Stack Trace

VenuesController.cs BookingsController.cs Booking.cs Venue.cs

EventEasePOE EventEasePOE.Controllers.BookingsController Edit(int id, Booking booking)

```

153     // POST: Bookings/Delete/5
154     [HttpPost, ActionName("Delete")]
155     [ValidateAntiForgeryToken]
156     public async Task<ActionResult> DeleteConfirmed(int id)
157     {
158         if (_context.Bookings == null)
159         {
160             return Problem("Entity set 'ApplicationDbContext.Bookings' is null.");
161         }
162         var booking = await _context.Bookings.FindAsync(id);
163         if (booking != null)
164         {
165             _context.Bookings.Remove(booking);
166         }
167
168         await _context.SaveChangesAsync();
169         return RedirectToAction(nameof(Index));
170     }
171
172     private bool BookingExists(int id)
173     {
174         return (_context.Bookings?.Any(e => e.BookingId == id)).GetValueOrDefault();
175     }
176 }
177

```

The thread 0x1e8 has exited with code 0  
The thread 0x3d7c has exited with code 0  
The thread 0xb520 has exited with code 0  
The thread 0x8a4c has exited with code 0  
The thread 0x47c4 has exited with code 0  
The thread 0x7780 has exited with code 0  
The thread 0x8464 has exited with code 0  
The thread 0x9990 has exited with code 0  
The thread 0x5a18 has exited with code 0  
The thread 0x1170 has exited with code 0  
The thread 0x712c has exited with code 0  
The thread 0x91a4 has exited with code 0  
The thread 0x77c8 has exited with code 0  
The thread 0x824 has exited with code 0  
The thread 0x79cc has exited with code 0  
The thread 0x5f64 has exited with code 0  
The thread 0x6338 has exited with code 0  
The thread 0x205c has exited with code 0  
The thread 0x9c2c has exited with code 0  
The thread 0x58dc has exited with code 0  
The thread 0x4a99 has exited with code 0  
The thread 0x9030 has exited with code 0  
The thread 0x7124 has exited with code 0  
The thread 0x97c4 has exited with code 0  
The thread 0x4f8c has exited with code 0  
The thread 0x5b0c has exited with code 0  
The thread 0x75a4 has exited with code 0  
The thread 0x4d74 has exited with code 0  
The thread 0x6a24 has exited with code 0  
The program '[12152] EventEasePOE.exe' has exited with code 0 (0x0)  
The thread 0x0 has exited with code 0 (0x0)  
The program 'JavaScript debug adapter: lc' has exited with code 4294967295  
The program '' has exited with code 4294967295

## C. Enhanced Display and Search: Booking view class:

Feature 'Symbol search' is currently unavailable due to an internal error. Show Stack Trace

VenuesController.cs BookingsController.cs Booking.cs Venue.cs

EventEasePOE EventEasePOE.Models.BookingViewing

```

1  namespace EventEasePOE.Models
2  {
3      public class BookingViewing
4      {
5          public int BookingId { get; set; }
6          public s readonly struct System.Int32
7              Represents a 32-bit signed integer.
8          public s
9          public DateTime BookingDate { get; set; }
10     }
11 }

```

Solution Explorer

- Controllers
  - BookingsController.cs
  - EventsController.cs
  - HomeController.cs
  - VenuesController.cs
- Data
- Functions
- Migrations
  - 20230513081203\_SecondCreate.cs
  - 20230513174235\_InitialMig.cs
  - ApplicationDbContextModelSnapshot.cs
- Models
  - Booking.cs
  - BookingViewing.cs
  - ErrorViewModel.cs

## Improved index class/cshtml:

```

1 <?*>
2     For more information on enabling MVC for empty projects, visit https://go.microsoft.com/fwlink/?LinkID=39860
3     @{
4         ViewData["Title"] = "Improved Bookings";
5     }
6     <h1>Improved Bookings</h1>
7     <form asp-action="ImprovedIndex" method="get">
8         <p>
9             <input type="text" name="searchString" placeholder="Search by Booking ID or Event Name" />
10            <input type="submit" value="Search" />
11        </p>
12    </form>
13    <table class="table">
14        <thead>
15            <tr>
16                <th>Booking ID</th>
17                <th>Venue Name</th>
18                <th>Event Name</th>
19                <th>Booking Date</th>
20            </tr>
21        </thead>
22        <tbody>
23            <@foreach (var booking in Model)>
24            <tr>
25                <td>@booking.BookingId</td>
26                <td>@booking.VenueName</td>
27                <td>@booking.EventName</td>
28                <td>@booking.BookingDate.ToShortDateString()</td>
29            </tr>
30        </tbody>
31    </table>
32 
```

## Improved index added to bookings controller:

```

1 <?*>
2     _context.Bookings.RemoveBooking();
3     await _context.SaveChangesAsync();
4     return RedirectToAction(nameof(Index));
5 
6     // GET: Bookings/ImprovedIndex
7     public async Task<ActionResult> ImprovedIndex(string searchString)
8     {
9         var bookings = from b in _context.Bookings
10                     join v in _context.Venues on b.VenueId equals v.VenueId
11                     join e in _context.Events on b.EventId equals e.EventId
12                     select new BookingViewing
13                     {
14                         BookingId = b.BookingId,
15                         VenueName = v.VenueName,
16                         EventName = e.EventName,
17                         BookingDate = b.BookingDate
18                     };
19 
20         if (!string.IsNullOrEmpty(searchString))
21         {
22             bookings = bookings.Where(b => b.BookingId.ToString().Contains(searchString) ||
23                                  b.EventName.Contains(searchString));
24         }
25         return View(await bookings.ToListAsync());
26     }
27 
28     private bool BookingExists(int id)
29     {
30         return (_context.Bookings?.Any(e => e.BookingId == id)).GetValueOrDefault();
31     }
32 
```

(BillWagner, 2024)

## Database Update:

```

Done.
PM> Update-Database
Build started...
Build succeeded.
Microsoft.EntityFrameworkCore.Infrastructure[10400]
  Entity Framework Core 6.0.36 initialized 'ApplicationDbContext' using provider 'Microsoft.EntityFrameworkCore.SqlServer:6.0.36' with options: None
Microsoft.EntityFrameworkCore.Database.Command[20101]
  Executed DbCommand (26ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
  SELECT 1
Microsoft.EntityFrameworkCore.Database.Command[20101]
  Executed DbCommand (0ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
  SELECT OBJECT_ID('[_EFMigrationsHistory]');
Microsoft.EntityFrameworkCore.Database.Command[20101]
  Executed DbCommand (0ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
  SELECT [MigrationId], [ProductVersion]
  FROM [_EFMigrationsHistory]
  ORDER BY [MigrationId];
Microsoft.EntityFrameworkCore.Migrations[20405]
  No migrations were applied. The database is already up to date.
  No migrations were applied. The database is already up to date.
Done.
PM> 
```

The screenshot shows the Azure portal interface for the `Eventease_Db` database. The left sidebar lists various management options like Overview, Activity log, Tags, and Query editor (preview), which is currently selected. The main area displays the database details and a query editor window titled "Query 1". The query is:

```
1 SELECT * FROM [dbo].[Bookings];
```

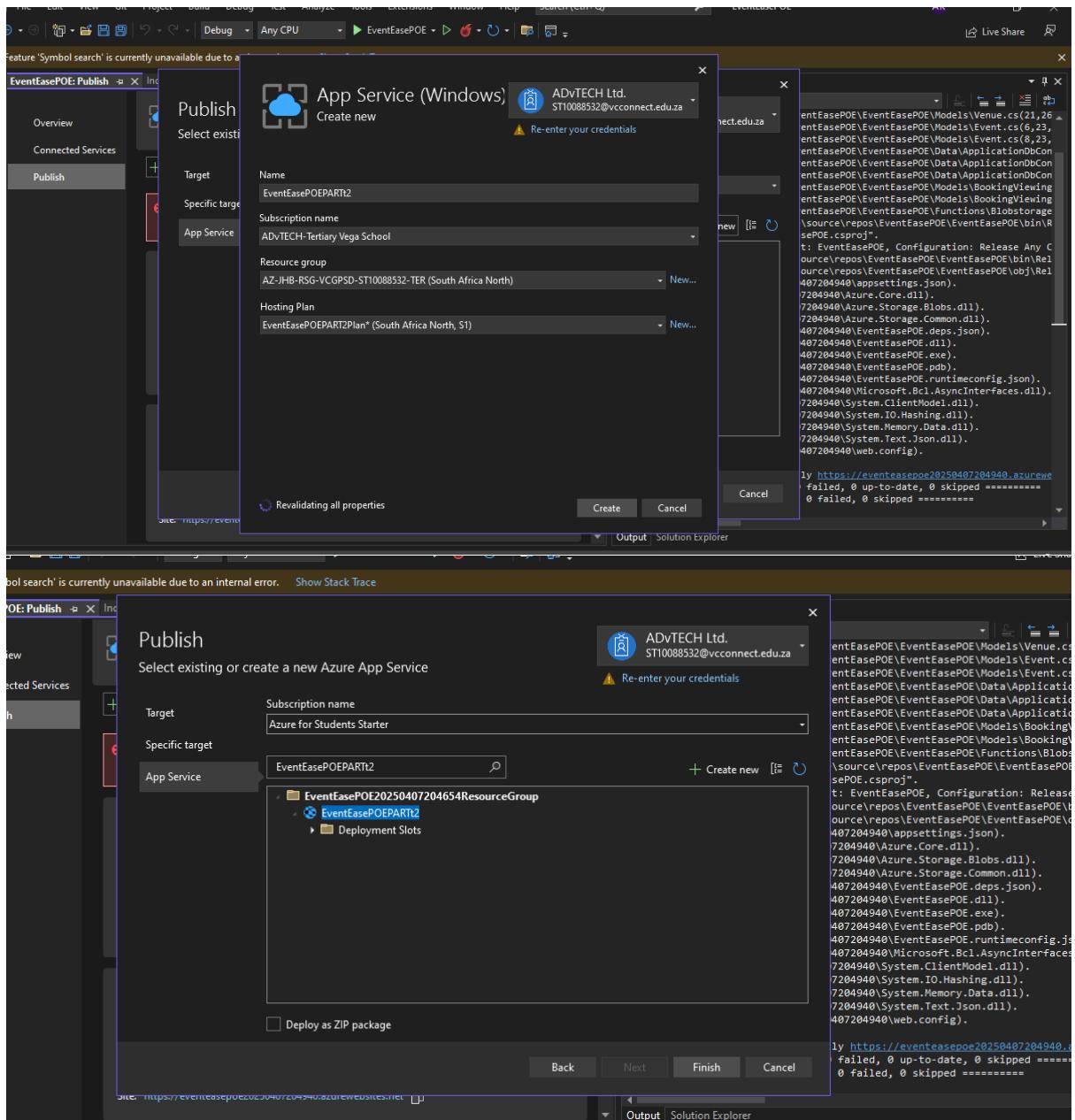
The results pane shows the following table data:

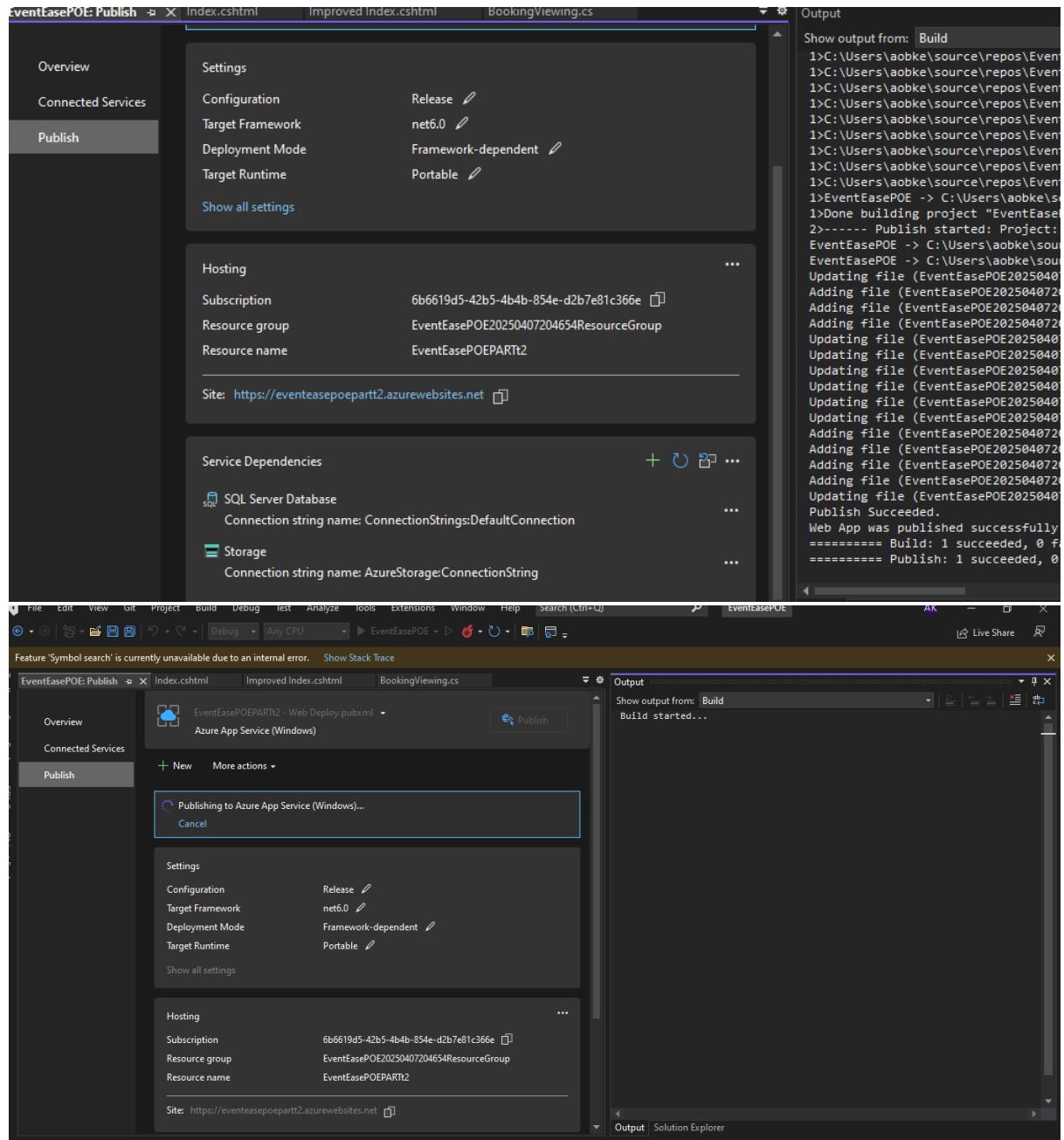
BookingId	EventId	VenueId	BookingDate
1	1	1	2025-08-20T13:13:00.000000...

## D. Azure deployment Updates:

The screenshot shows the Microsoft Visual Studio interface with the following details:

- Top Bar:** File, Edit, View, Git, Project, Build, Debug, Test, Analyze, Tools, Extensions, Window, Help, Search (Ctrl+Q), Live Share.
- Status Bar:** Feature 'Symbol search' is currently unavailable due to an internal error. Show Stack Trace.
- Solution Explorer:** Shows the project structure with files like EventEasePOE.csproj, Improved.Index.cshtml, BookingViewing.cs, and EventEasePOE.cs.
- Task List:** Shows the status of the build process: Checking if your application will run successfully... 0 up-to-date, 0 skipped, 0, 0 skipped.
- Central Area:** Publish dialog for EventEasePOE Publish.
  - Overview:** Shows the Azure App Service (Windows) configuration with subscription ID 6b6619d5-42b5-4b4b-854e-d2b7e01c366e, resource group EventEasePOE20250407204654ResourceGroup, and resource name EventEasePOE20250407204940. Site URL: https://eventeasepoe20250407204940.azurewebsites.net
  - Publish:** Selected tab. Sub-options include New, Publish (highlighted in red), and See logs.
  - Settings:** Publishing to Azure App Service (Windows)...
  - Configuration:** Relaxed (net6.0).
  - Target Framework:** Framework 4.8.
  - Deployment Mode:** Development.
  - Target Runtime:** Portable.
  - Show all settings:** Link to full settings.
- Bottom Area:** Publish dialog for EventEasePOE Publish.
  - Overview:** Shows the Azure service selection.
  - Publish:** Selected tab. Sub-options include New, Publish (highlighted in red), and See logs.
  - Target:** Specific target (selected). Options include:
    - Azure App Service (Windows): Publish your application code to a managed infrastructure that is easy to scale.
    - Azure App Service (Linux): Publish your application code to a managed infrastructure that is easy to scale.
    - Azure Container Apps (Linux): Run scalable containerized applications and microservices on a serverless platform in Azure.
    - Azure App Service Container: Publish your application as a Docker image to Azure Container Registry and run it on Azure App Service.
    - Azure Container Registry: Publish your application as a Docker image to Azure Container Registry.
    - Azure Virtual Machine: Manage your own infrastructure.
  - Back, Next, Finish, Cancel:** Navigation buttons.





The screenshot shows the Azure App Service (Windows) blade for the project "EventEasePOEPARTt2". At the top, it displays "EventEasePOEPARTt2 - Web Deploy.pubxml" and "Azure App Service (Windows)". A "Publish" button is visible. Below the header, there's a message box stating "Publish succeeded on 2025/05/13 at 23:12." with a "Open site" link. The main area contains sections for "Settings" and "Hosting".

**Settings**

- Configuration: Release (Edit)
- Target Framework: net6.0 (Edit)
- Deployment Mode: Framework-dependent (Edit)
- Target Runtime: Portable (Edit)
- Show all settings

**Hosting**

- Subscription: 6b6619d5-42b5-4b4b-854e-d2b7e81c366e (Edit)
- Resource group: EventEasePOE20250407204654ResourceGroup
- Resource name: EventEasePOEPART2

Site: <https://eventeasepoepart2.azurewebsites.net>

The screenshot shows the Azure Data Studio Query editor for the database "Eventease\_Db". The sidebar menu includes "Overview", "Activity log", "Tags", "Diagnose and solve problems", "Query editor (preview)" (which is selected), "Mirror database in Fabric (preview)", "Resource visualizer", "Settings", "Data management", "Integrations", "Power Platform", and "Security".

The main area shows a query titled "Query 1" with the following content:

```
1 SELECT * FROM [dbo].[Events]
```

The results pane displays a table with the following data:

EventId	EventName	EventDate	Description	VenueId
1	The Sundae Chronicles	2025-04-09T03:00:00....	Sunday vibes	2

## E. Database Design, Cognitive search

1. Azure cognitive search is a cloud-based search service from Microsoft that helps developers build and manage powerful search experiences. It allows users to index and query content from various sources, including websites, databases, and documents. (HeidiSteen, 2025) (Brush, 2023) Traditional search engines, like Google, Bing, and Yahoo, rely on indexing and ranking algorithms to deliver search results. They crawl the web, index content, and then use algorithms to match user queries with relevant information. These engines excel at finding specific web pages or documents based on keyword matching and offer features like advanced query syntax and filters. (Kerner, 2024) The system strongly excels in understanding the sophisticated queries, contextually recommending, and ranking results based on true relevance against the actual query instead of inflating counts of keyword matches.

2. “This process makes your database simpler to navigate while also enhancing the precision and integrity of your data. Data normalization ensures that all of the records in the database of customers have the same appearance, readability, and usability.” By reducing data redundancy, minimizing cloud storage costs and improving elastic scaling capabilities.

The impact on normalised structures, it is organizing your data so you don't repeat the same information everywhere. Instead of having customer names in every order record, you store them once and reference them. It is important because normalized data is much easier to manage across multiple servers when it comes to scaling issues and It keeps cloud data consistent.

The impact on denormalised structures, Combining of data from multiple different tables is done for optimizing data retrieval into one single table or document. It reads faster because joining is less necessary.

Data that is readily available in just a single table simplifies any queries. Updates become more complex with potentially wasting storage space also data redundancy increases. It can compromise data, if one does not manage it carefully. Data's careful management is important for its integrity. (Kordos, 2024)

## URL Links:

GITHUB: <https://github.com/ST10088532/VCSANDTON-CLDV6211-POE-PART2.git>

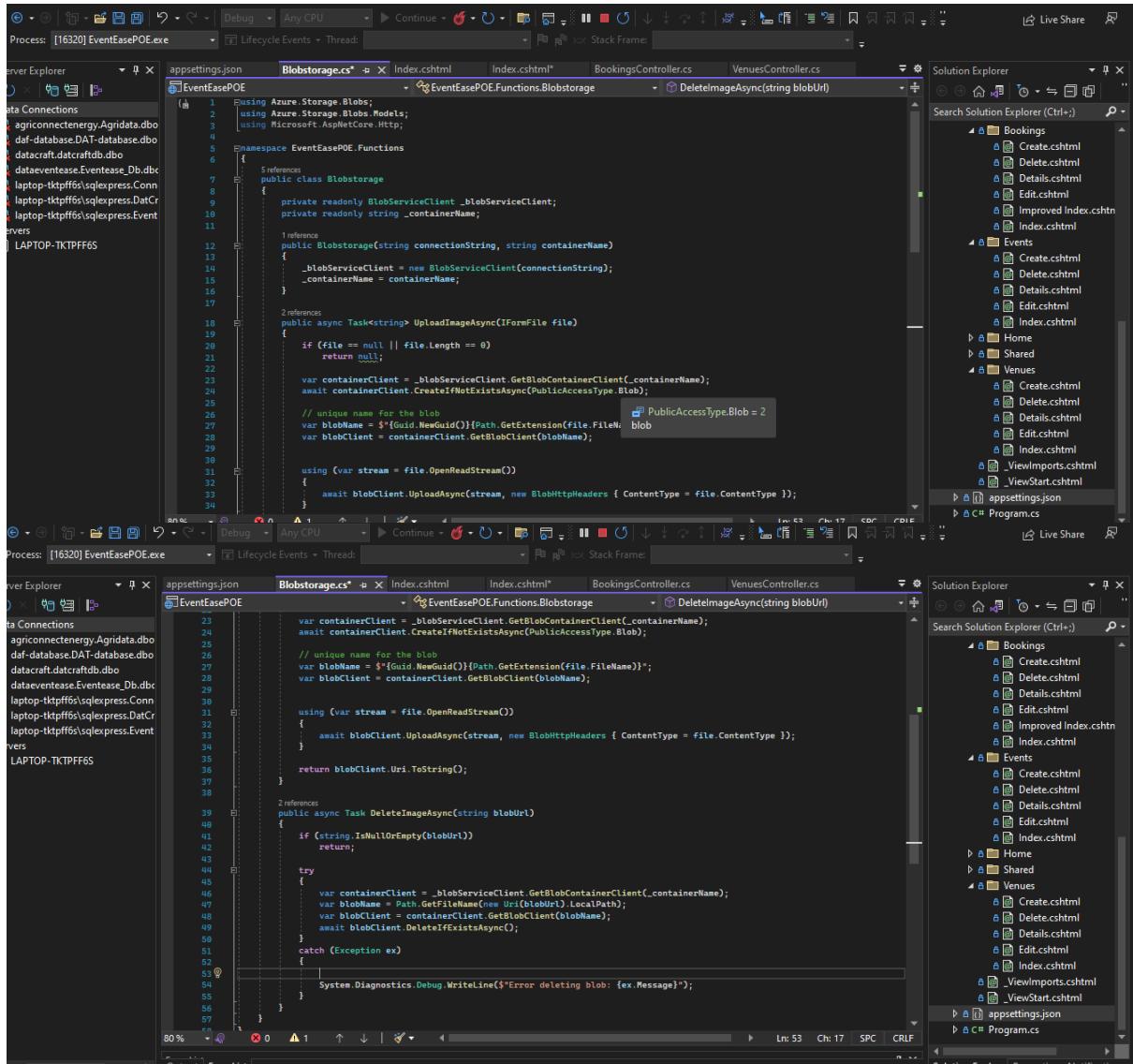
Azure published link:

<https://eventeasepoepart2.azurewebsites.net>

# Part 3

Updated codebases and files/functionality

## Blob storage:



The screenshot shows two instances of the Blobstorage.cs file from a .NET Core application named EventEasePOE. Both instances are displayed in the code editor, with the top one being the original version and the bottom one being an updated version.

**Original Blobstorage.cs (Top):**

```
1  using Azure.Storage.Blobs;
2  using Azure.Storage.Blobs.Models;
3  using Microsoft.AspNetCore.Http;
4
5  namespace EventEasePOE.Factories
6  {
7      public class Blobstorage
8      {
9          private readonly BlobServiceClient _blobServiceClient;
10         private readonly string _containerName;
11
12         public Blobstorage(string connectionString, string containerName)
13         {
14             _blobServiceClient = new BlobServiceClient(connectionString);
15             _containerName = containerName;
16         }
17
18         public async Task<string> UploadImageAsync(IFormFile file)
19         {
20             if (file == null || file.Length == 0)
21                 return null;
22
23             var containerClient = _blobServiceClient.GetBlobContainerClient(_containerName);
24             await containerClient.CreateIfNotExistsAsync(PublicAccessType.Blob);
25
26             // unique name for the blob
27             var blobName = $"{Guid.NewGuid()}{Path.GetExtension(file.FileName)}";
28             var blobClient = containerClient.GetBlobClient(blobName);
29
30             using (var stream = file.OpenReadStream())
31             {
32                 await blobClient.UploadAsync(stream, new BlobHttpHeaders { ContentType = file.ContentType });
33             }
34         }
35     }
36 }
```

**Updated Blobstorage.cs (Bottom):**

```
23     var containerClient = _blobServiceClient.GetBlobContainerClient(_containerName);
24     await containerClient.CreateIfNotExistsAsync(PublicAccessType.Blob);
25
26     // unique name for the blob
27     var blobName = $"{Guid.NewGuid()}{Path.GetExtension(file.FileName)}";
28     var blobClient = containerClient.GetBlobClient(blobName);
29
30     using (var stream = file.OpenReadStream())
31     {
32         await blobClient.UploadAsync(stream, new BlobHttpHeaders { ContentType = file.ContentType });
33     }
34
35     return blobClient.Uri.ToString();
36 }
37
38 public async Task DeleteImageAsync(string blobUrl)
39 {
40     if (string.IsNullOrEmpty(blobUrl))
41         return;
42
43     try
44     {
45         var containerClient = _blobServiceClient.GetBlobContainerClient(_containerName);
46         var blobName = Path.GetFileName(new Uri(blobUrl).LocalPath);
47         var blobClient = containerClient.GetBlobClient(blobName);
48         await blobClient.DeleteIfExistsAsync();
49     }
50     catch (Exception ex)
51     {
52         System.Diagnostics.Debug.WriteLine($"Error deleting blob: {ex.Message}");
53     }
54 }
```

The screenshot shows two side-by-side browser windows. The left window displays the 'Create a storage account' configuration page, showing various data protection and encryption settings. The right window shows the 'easeimages\_1750597159392 | Overview' page, indicating a deployment is in progress.

**Create a storage account**

**Data protection**

Point-in-time restore	Disabled
Blob soft delete	Enabled
Blob retention period in days	7
Container soft delete	Enabled
Container retention period in days	7
File share soft delete	Enabled
File share retention period in days	7
Versioning	Disabled
Blob change feed	Disabled
Version-level immutability support	Disabled

**Encryption**

Encryption type	Microsoft-managed keys (MMK)
Enable support for customer-managed keys	Blobs and files only
Enable infrastructure encryption	Disabled

**Deployment Overview**

Deployment name: easeimages\_1750597159392  
Subscription: ADVTECH-Tertiary Vega School  
Resource group: AZ-JHB-RSG-VCGPSD-ST10088532-...

Start time: 6/22/2025, 2:59:44 PM  
Correlation ID: 0e23226e-4a07-4dac-b960-5e034a1d14b8

**Deployment is in progress**

**Deployment details**

Resource	Type	Status	Operation details
No results.			

[Give feedback](#)  
[Tell us about your experience with deployment](#)

**Microsoft Defender for Cloud**  
Secure your apps and infrastructure  
[Go to Microsoft Defender for Cloud >](#)

**Free Microsoft tutorials**  
[Start learning today >](#)

**Work with an expert**  
Azure experts are service provider partners who can help manage your assets on Azure and be your first line of support.  
[Find an Azure expert >](#)

The image displays two screenshots of the Microsoft Azure portal.

**Screenshot 1: Deployment Overview**

This screenshot shows the deployment details for a deployment named "easeimages\_1750597159392". The deployment is marked as complete. Key information includes:

- Deployment name: easeimages\_1750597159392
- Subscription: ADVTECH-Tertiary Vega School
- Resource group: AZ-JHB-RSG-VCGPSD-ST10088532...
- Start time: 6/22/2025, 2:59:44 PM
- Correlation ID: 0e23226e-4a07-4dac-b960-5e034a1d14b8

Next steps include "Go to resource" and "Give feedback". A "Tell us about your experience with deployment" link is also present.

**Screenshot 2: Container Access Level Change**

This screenshot shows the "Change access level" dialog for a container named "venue-images". The current access level is set to "Blob (anonymous read access for blobs only)". A note states: "Blobs within the container can be read by anonymous request, but container data is not available. Anonymous clients cannot enumerate the blobs within the container." The dialog has "Ok" and "Cancel" buttons.

## Booking View and Search? Enhanced Display and Search

The image shows two side-by-side screenshots of a Microsoft Visual Studio IDE interface, likely version 2019, displaying code for a .NET Core application.

**Screenshot 1 (Top):**

- Code Editor:** Shows the `BookingViewing.cs` file in the `EventEasePOE.Models` namespace. The class definition is as follows:
 

```
1  namespace EventEasePOE.Models
2  {
3      public class BookingViewing
4      {
5          public int BookingId { get; set; }
6          public string VenueName { get; set; }
7          public string EventName { get; set; }
8          public DateTime BookingDate { get; set; }
9      }
10 }
11 
```
- Solution Explorer:** Shows the project structure. Under the `Models` folder, there are `Booking.cs` and `BookingViewing.cs` files.
- Toolbars and Status Bar:** Standard Visual Studio toolbars and status bar at the bottom.

**Screenshot 2 (Bottom):**

- Code Editor:** Shows the `BookingViewing.cshtml` file. The code is a Razor view for displaying bookings. It includes a search bar and a table to list bookings. The table columns are `Booking ID`, `Venue Name`, `Event Name`, and `Booking Date`.
 

```
1  @model IEnumerable<EventEasePOE.Models.BookingViewing>
2
3  @{
4      ViewData["Title"] = "Booking Viewing";
5  }
6
7  <h1>Booking Viewing</h1>
8
9  <form asp-action="BookingViewing" method="get">
10     <p>
11         <input type="text" name="searchString" placeholder="Search by Booking ID or Event Name" value="@{ViewContext.H
12         <input type="submit" value="Search" />
13     </p>
14 </form>
15
16 <table class="table">
17     <thead>
18         <tr>
19             <th>Booking ID</th>
20             <th>Venue Name</th>
21             <th>Event Name</th>
22             <th>Booking Date</th>
23         </tr>
24     <thead>
25     <tbody>
26         @foreach (var booking in Model)
27         {
28             <tr>
29                 <td>@booking.BookingId</td>
30                 <td>@booking.VenueName</td>
31                 <td>@booking.EventName</td>
32                 <td>@booking.BookingDate.ToString("MM dd, yyyy")</td>
33             </tr>
34         }
35     </tbody>
36 </table>
```
- Solution Explorer:** Shows the project structure. Under the `Views` folder, there is a `Bookings` folder containing `BookingViewing.cshtml` and other view files like `Create.cshtml`, `Delete.cshtml`, etc.
- Toolbars and Status Bar:** Standard Visual Studio toolbars and status bar at the bottom.

**A.**

The screenshot displays three separate instances of the Microsoft Visual Studio IDE, each showing a different class from the `EventEasePOE.Models` namespace with associated code analysis errors. The Solution Explorer on the right shows the project structure for `EventEasePOE`, which includes files for Booking, BookingViewing, ErrorViewModel, Event, EventType, Venue, Views, and Program.

**EventEasePOE.Models.EventType.cs:**

```

1  namespace EventEasePOE.Models
2  {
3      public class EventType
4      {
5          public int EventTypeID { get; set; }
6          public string Name { get; set; }
7      }
8  }

```

**EventEasePOE.Models.Event.cs:**

```

1  using System.ComponentModel.DataAnnotations;
2
3  namespace EventEasePOE.Models
4  {
5      public class Event
6      {
7          public int EventId { get; set; }
8          [Required(ErrorMessage = "Event Name is required.")]
9          public string? EventName { get; set; }
10         [Required(ErrorMessage = "Event Date is required.")]
11         public DateTime EventDate { get; set; }
12         [Required(ErrorMessage = "Description is required.")]
13         public string? Description { get; set; }
14         [Required(ErrorMessage = "Venue is required.")]
15         public int VenueId { get; set; }
16
17         public int EventTypeID { get; set; }
18         public virtual EventType EventType { get; set; }
19     }
20 }

```

**EventEasePOE.Models.Venue.cs:**

```

1  using System.ComponentModel.DataAnnotations;
2
3  namespace EventEasePOE.Models
4  {
5      public class Venue
6      {
7          public int VenueId { get; set; }
8
9          [Required(ErrorMessage = "Venue Name is required.")]
10         public string? VenueName { get; set; }
11
12         [Required(ErrorMessage = "Location is required.")]
13         public string? Location { get; set; }
14
15         [Required(ErrorMessage = "Capacity is required.")]
16         public int Capacity { get; set; }
17
18         public string? ImageUrl { get; set; }
19
20         [NotMapped]
21         public IFormFile? ImageFile { get; set; }
22
23         public bool Available { get; set; }
24     }
25 }

```

**Code Analysis Errors (CS8601, CS8618):**

- CS8601:** Possible null reference assignment. Non-nullable property 'VenueName' must contain a non-null value when exiting constructor. Consider declaring the property as nullable.
- CS8618:** Non-nullable property 'EventName' must contain a non-null value when exiting constructor. Consider declaring the property as nullable.

The screenshot shows three instances of the Visual Studio IDE interface, each displaying a different file from a .NET Core project named "EventEasePOE".

**Top Window:**

- Code Editor:** Displays `ApplicationDbContext.cs` with three CS8618 warnings.
- Solution Explorer:** Shows the project structure with files like `EventEasePOE.csproj`, `Controllers`, `Data`, `Models`, `Views`, and `appsettings.json`.

```

using EventEasePOE.Models;
using Microsoft.EntityFrameworkCore;
using System.Collections.Generic;

12 references
13 public class ApplicationDbContext : DbContext
14 {
15     0 references
16     public DbSet<Venue> Venues { get; set; }
17     14 references
18     public DbSet<Event> Events { get; set; }
19     17 references
20     public DbSet<Booking> Bookings { get; set; }
21     0 references
22     public DbSet<EventType> EventTypes { get; set; }
23 }

```

**Middle Window:**

- Code Editor:** Displays `BookingViewing.cs` with one CS8618 warning.
- Solution Explorer:** Shows the project structure with files like `EventEasePOE.csproj`, `Controllers`, `Data`, `Models`, `Views`, and `appsettings.json`.

```

1 namespaces EventEasePOE.Models
2 {
3     0 references
4     public class BookingViewing
5     {
6         3 references
7         public int BookingId { get; set; }
8         2 references
9         public string VenueName { get; set; }
10        3 references
11        public string EventName { get; set; }
12        2 references
13        public DateTime BookingDate { get; set; }
14        0 references
15         public string EventType { get; set; }
16         0 references
17         public bool VenueAvailable { get; set; }
18     }
19 }

```

**Bottom Window:**

- Code Editor:** Displays `BookingViewing.cs` with one CS8618 warning.
- Solution Explorer:** Shows the project structure with files like `EventEasePOE.csproj`, `Controllers`, `Data`, `Models`, `Views`, and `appsettings.json`.

```

1 namespaces EventEasePOE.Models
2 {
3     0 references
4     public class BookingViewing
5     {
6         3 references
7         public int BookingId { get; set; }
8         2 references
9         public string VenueName { get; set; }
10        3 references
11        public string EventName { get; set; }
12        2 references
13        public DateTime BookingDate { get; set; }
14        0 references
15         public string EventType { get; set; }
16         0 references
17         public bool VenueAvailable { get; set; }
18     }
19 }

```

```
Microsoft.EntityFrameworkCore.Database.Command[2018]
    Executed DbCommand (3ms) [Parameters=@1, CommandType='Text', CommandTimeout='30']
        CREATE INDEX IX_Events_EventTypeId ON [Events] ([EventTypeId]);
Fail: Microsoft.EntityFrameworkCore.Database.Command[2018]
    Failed executing DbaCommand (47ms) [Parameters=@1, CommandType='Text', CommandTimeout='30']
        ALTER TABLE [Events] ADD CONSTRAINT [FK_Events_EventTypes_EventId] FOREIGN KEY ([EventTypeId]) REFERENCES [EventTypes] ([EventTypeId]) ON DELETE CASCADE;
    Failed executing DbaCommand (47ms) [Parameters=@1, CommandType='Text', CommandTimeout='30']
ALTER TABLE [Events] ADD CONSTRAINT [FK_Events_EventTypes_EventId] FOREIGN KEY ([EventTypeId]) ON DELETE CASCADE;
The conflict occurred in database "Eventsease_Db", table "dbo.EventTypes", column 'Event
at Microsoft.Data.SqlClient.SqlException (0x80131984): The ALTER TABLE statement conflicted with the FOREIGN KEY constraint "FK_Events_EventTypes_EventTypeId". The conflict occurred in database "Event
at Microsoft.Data.SqlClient.SqlConnection.OnError(SqlException exception, Boolean breakConnection, Action`1 wrapCloseInAction)
at Microsoft.Data.SqlClient.SqlInternalConnection.OnError(SqlException exception, Boolean breakConnection, Action`1 wrapCloseInAction)
at Microsoft.Data.SqlClient.TdsParser.ThrowExceptionAndWarning(TdsParserStateObject stateObj, Boolean callerHasConnectionLock, Boolean asyncClose)
at Microsoft.Data.SqlClient.TdsParser.TryRun(RunBehavior runBehavior, SqlCommand cmdHandler, SqlDataReader dataStream, BulkCopySimpleResultSet bulkCopyHandler, TdsParserStateObject stateObj, Boolean& dataReady)
at Microsoft.Data.SqlClient.SqlCommand.RunExecuteNonQueryTds(String methodName, Boolean isAsync, Int32 timeout, Boolean asyncWrite)
at Microsoft.Data.SqlClient.SqlCommand.InternalExecuteNonQuery(TaskCompletionSource`1 completion, Boolean sendToPipe, Int32 timeout, Boolean usedCache, Boolean asyncWrite, Boolean inRetry, String methodName)
at Microsoft.Data.SqlClient.SqlCommand.ExecuteNonQuery()
at Microsoft.EntityFrameworkCore.Relational.CreateCommand(ICommandParameterObject parameterObject)
at Microsoft.EntityFrameworkCore.Relational.CreateCommand(ExecutionNonQuery<IRelationalConnection> parameterObject)
at Microsoft.EntityFrameworkCore.Migrations.Internal.MigrationCommand.ExecuteNonQuery(IRelationalConnection connection, IReadOnlyDictionary<T> parameterValues)
at Microsoft.EntityFrameworkCore.Migrations.Internal.MigrationCommandExecutor.ExecuteNonQuery(IEnumerable<IMigrationCommand>, IRelationalConnection connection)
at Microsoft.EntityFrameworkCore.Migrations.Internal.Migrator.Migrate(String targetMigration)
at Microsoft.EntityFrameworkCore.Design.Internal.MigrationsOperations.UpdateDatabase(String targetMigration, String connectionString, String contextType)
at Microsoft.EntityFrameworkCore.Design.Internal.MigrationsOperations.UpdateDatabase(String targetMigration, String connectionString, String contextType)
at Microsoft.EntityFrameworkCore.Design.Internal.MigrationsExecutor.UpdateDatabase(String targetMigration, String connectionString, String contextType)
at Microsoft.EntityFrameworkCore.Design.OperationExecutor.UpdateDatabase(Action action)
ClientConnectionId:3be3f3af-0922-4064-b18d-ca705d71b5fc
Error Number:547, State:0, Class:16
```

Eventease\_Db (dataeventease/Eventease\_Db) | Query editor (preview)

Query 1

1 `DELETE FROM Bookings;`

2 `DELETE FROM Events;`

Results

Messages

```
Package source: All
Default project: EventEasePOE
Build started...
Build succeeded.
Microsoft.EntityFrameworkCore.Infrastructure[10403]
  Entity Framework Core 6.0.3 initialized 'ApplicationDbContext' using provider 'Microsoft.EntityFrameworkCore.SqlServer:6.0.36' with options: None
Microsoft.EntityFrameworkCore.Database.Command[20801]
  Executed DbCommand (35ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    SELECT 1
Microsoft.EntityFrameworkCore.Database.Command[20801]
  Executed DbCommand (44ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    SELECT OBJECT_ID('[_EFMigrationsHistory]');
Microsoft.EntityFrameworkCore.Database.Command[20801]
  Executed DbCommand (23ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    SELECT 1
Microsoft.EntityFrameworkCore.Database.Command[20801]
  Executed DbCommand (27ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    SELECT OBJECT_ID('[_EFMigrationsHistory]');
Microsoft.EntityFrameworkCore.Database.Command[20801]
  Executed DbCommand (0ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    SELECT [MigrationId], [ProductVersion]
    FROM [_EFMigrationsHistory]
    ORDER BY [MigrationId];
Microsoft.EntityFrameworkCore.Migrations[20802]
  Applying migration '20258622222256_AddAdvancedFiltering'.
  Applying migration '20258622222256_AddAdvancedFiltering'.
Microsoft.EntityFrameworkCore.Database.Command[20801]
  Executed DbCommand (32ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    DECLARE @var0 sysname;
    SELECT @var0 = [name]
    FROM [sys].[default_constraints] [d]
    INNER JOIN [sys].[columns] [c] ON [d].[parent_column_id] = [c].[column_id] AND [d].[parent_object_id] = [c].[object_id]
    WHERE (([d].[parent_object_id] = OBJECT_ID('N'[Venues])) AND ([c].[name] = N'ImageURL'));
    IF (@var0 IS NOT NULL EXECUTE ALTER TABLE [Venues] DROP CONSTRAINT [r_1];
    ALTER TABLE [Venues] ADD [Available] bit NOT NULL DEFAULT CAST(0 AS bit);
Microsoft.EntityFrameworkCore.Database.Command[20801]
  Executed DbCommand (38ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    ALTER TABLE [Events] ADD [EventTypeId] int NOT NULL DEFAULT 0;
Microsoft.EntityFrameworkCore.Database.Command[20801]
  Executed DbCommand (31ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    CREATE TABLE [EventTypes] (
      [EventTypeId] int NOT NULL,
      [Name] nvarchar(max) NOT NULL,
      CONSTRAINT [PK_EventTypes] PRIMARY KEY ([EventTypeId])
    );
Microsoft.EntityFrameworkCore.Database.Command[20801]
  Executed DbCommand (30ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    CREATE INDEX [IX_EventTypeEventId] ON [Events] ([EventTypeId]);
Microsoft.EntityFrameworkCore.Database.Command[20801]
  Executed DbCommand (35ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    ALTER TABLE [Events] ADD CONSTRAINT FK_EventEventTypes ON [Events] ([EventTypeId]) REFERENCES [EventTypes] ([EventTypeId]);
Microsoft.EntityFrameworkCore.Database.Command[20801]
  Executed DbCommand (35ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    ALTER TABLE [Events] ADD CONSTRAINT FK_EventEventTunstall ON [Events] ([EventTunstall]) REFERENCES [EventTunstall] ([EventTunstall]);
Microsoft.EntityFrameworkCore.Database.Command[20801]
  Executed DbCommand (35ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    ALTER TABLE [Events] ADD CONSTRAINT FK_EventEventTunstall ON [Events] ([EventTunstall]) REFERENCES [EventTunstall] ([EventTunstall]);
```

## Venues page:

EventEasePOE Home Venues Event Booking Booking Search Privacy

### Venues

Create New

VenueName	Location	Capacity	Image	Actions
Sundaechronicles	Soweto	7		<a href="#">Edit</a> <a href="#">Details</a> <a href="#">Delete</a>
Sundae chronicles	Soweto	100		<a href="#">Edit</a> <a href="#">Details</a> <a href="#">Delete</a>

localhost:5035/Venues/Create

EventEasePOE Home Venues Event Booking Booking Search Privacy

### Create

#### Venue

VenueName

Location

Capacity

Venue Image  
   
Please select an image file

[Create](#) [Back to List](#)

localhost:5035/Venues/Details/1

EventEasePOE Home Venues Event Booking Booking Search Privacy

### Details

#### Venue

<b>VenueName</b>	Sundaechronicles
<b>Location</b>	Soweto
<b>Capacity</b>	7
<b>ImageUrl</b>	<a href="https://easeimages.blob.core.windows.net/venue-images/051bee8f-5ae5-40ba-9550-a949fe584912.png">https://easeimages.blob.core.windows.net/venue-images/051bee8f-5ae5-40ba-9550-a949fe584912.png</a>

[Edit](#) | [Back to List](#)

localhost:5035/Venues/Edit/1

## Edit

### Venue

VenueName  
Sundaechronicles

Location  
Soweto

Capacity  
7

Current Image



Choose File Screenshot 20...1 185040.png

localhost:5035/Venues/Delete/4

EventEasePOE Home Venues Event Booking Booking Search Privacy

## Delete

Are you sure you want to delete this?

### Venue

VenueName	njalos
Location	Soweto
Capacity	203
ImageUrl	<a href="https://easeimages.blob.core.windows.net/venue-images/a43bfe18-8694-48ee-975c-2569623c5afd.png">https://easeimages.blob.core.windows.net/venue-images/a43bfe18-8694-48ee-975c-2569623c5afd.png</a>

[Delete](#) | [Back to List](#)

Events page

localhost:5035/Events/Create

## Create

### Event

EventName  
The Sundae Chronicles

EventDate  
2025/06/24 01:51

Description  
Sunday vibes

VenueId  
Sundae chronicles

[Create](#)

[Back to List](#)

## Advanced search/Filter Search

localhost:5035/Bookings/BookingViewing

EventEasePOE Home Venues Event Booking Booking Search Privacy

### Advanced Booking Search

Search	Event Type	Start Date	End Date	Venue Availability	
1	All	2025/06/24	2025/06/24	All	<a href="#">Apply Filters</a>
Booking ID	Event Name	Event Type	Venue Name	Venue Is Available	Booking Date

[6680] EventEasePOE.exe Lifecycle Events Thread: Stack Frame:

'pExtractClassCodeRefactoringProvider' encountered an error and has been disabled. Show Stack Trace Enable Enable and ignore future errors

File Explorer EventEasePOE.cs \_Layout.cshtml EventsController.cs Create.cshtml Edit.cshtml BookingViewing.cs

```
namespace EventEasePOE.Models
{
    public class EventType
    {
        public int EventTypeID { get; set; }
        public string Name { get; set; }
    }
}
```

Solution Explorer

B.

## Web App URL:

<https://st10088532eventeasepoe.azurewebsites.net>

GITHUB link: <https://github.com/ST10088532/VCSTDN.git>

The screenshot shows two instances of the Microsoft Azure SQL Database Query Editor. Both instances display the same query results for the `EventTypes` table.

**Query 1 (Top Window):**

```
1 | SELECT * FROM EventTypes;
```

**Results (Top Window):**

	Name
1	Conference
2	Wedding
3	Concert
4	Party

**Query 2 (Bottom Window):**

```
1 | SELECT * FROM EventTypes;
```

**Results (Bottom Window):**

	Name
1	Conference
2	Wedding
3	Concert
4	Party

**Object Explorer (Left Side):**

- Tables
  - dbo.Booking
  - dbo.Bookings
  - dbo.Event
  - dbo.Events
  - dbo.EventTypes
  - dbo.Venue
  - dbo.Venues
- Views

**Properties (Bottom Left):**

dbo.EventTypes  
EventId (PK, int, not null)  
Name (nvarchar, not null)

# Final Deployment:

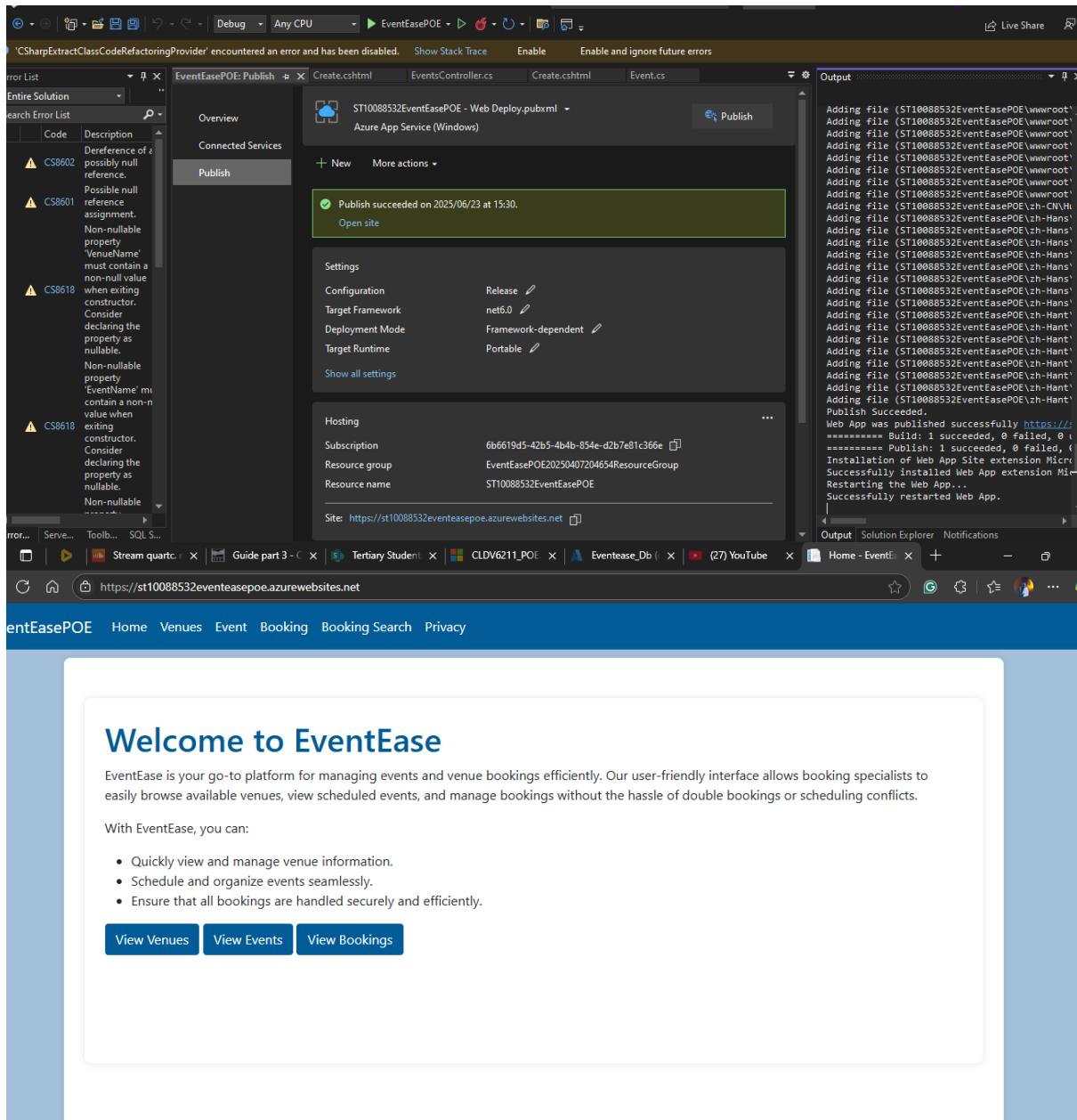
The screenshot shows the final deployment steps for an application named "EventEasePOE".

**Visual Studio Publish Dialog:**

- Subscription name:** Azure for Students Starter
- Resource Group:** EventEasePOE20250407204654ResourceGroup
- Deployment Options:** Deploy as ZIP package
- Buttons:** Back, Next, Finish, Cancel

**Azure Portal Deployment Blade:**

- Site:** https://eventeasenewpart2.azurewebsites.net
- Subscription:** ST10088532EventEasePOE - Web Deploy.pubxml
- Resource group:** EventEasePOE20250407204654ResourceGroup
- Resource name:** ST10088532EventEasePOE
- Status:** Ready to publish.
- Settings:** Configuration (Release), Target Framework (net6.0), Deployment Mode (Framework-dependent), Target Runtime (Portable)
- Output:** Shows deployment logs.



C.

## Reflective Technical Report

### Introduction

This report will demonstrate and describe my development path during the POE which has been/was a web application for managing venues, events, and bookings. The web application takes advantage of Azure services, such as: Azure Web App, Azure SQL Database, and Azure Blob Storage. This report is a brief account of the time I spent on these services, the technical and

architectural/UI decisions that I took and the lessons learnt in designing, developing and deploying a cloud based application.

## System feature list:

### Comprehensive List of Features and Requirements

Azure Web App - The EventEasePOE ASP. NET MVC application in an elastic, managed web environment.

Azure SQL Database - Contains all tabular data such as venues, events, bookings, and event type data with integrity, security, and high availability.

Azure Blob Storage - Twitter metadata in JSON format processed and served; unstructured data stored at scale in an efficient and secure manner.

Venue Management - CRUD operations for Venues with Image upload and delete into Blob Storage not allowing a delete when there are active bookings on the Venue.

Event Management – CRUD operations for events which are related with venues and event types, also doesn't allow double booking of venues for same date time.

Manage Booking - CRUD for bookings (with double-booking protection, and heavy validation)

Search and Filter - Search Bookings by Booking ID or Event Name and filter by event type, venue availability and date range.

Validation and Exception Management- Performed server side and client-side validation using data annotations, error message and validation summaries.

Optimized Display – Provides an all-in-one view of booking, event, and venue details using intuitive navigation tabs.

### Brief Explanation of How Each Feature/Requirement Works

#### Azure Web App:

The app is deployed to Azure App Service and is publicly accessible via a URL. This service takes care of scaling, patching and load balancing out of the box.

#### Azure SQL Database:

Azure SQL is used for all of the application data. ORM, migrations, and data access done using Entity Framework Core. The database ensures referential integrity and enables concurrent access.

#### Azure Blob Storage:

Venue images are uploaded from web app and saved in Azure Blob Storage. Images are loaded using URLs in the app, and also removed when venues are unlisted.

#### Local event hosting and coordination.

Users can manage the venues, events, and bookings (CRUD). Business rules (e.g. no double-bookings or no-deletion with active booking) are enforced in the controllers.

#### Filtering and Search:

Tribe members can search and narrow down bookings and events by way of a variety of filters, making it all the more user-friendly and convenient!

#### Validation/Error Handling:

Data annotations make certain that the required fields are filled in. The validation errors show up in the user interface and server side checks keep invalid operations out of the application.

#### Enhanced Display:

Views are used to display linked data (Bookings display event and venue details), to enhance user interaction.

## Technologies Used and Why

### 1. ASP.NET Core MVC

- Why: Offers a good architecture to design web applications which are scalable as well as maintainable with clear separation of concerns.
- Alternative: Node.js/Express, Django or Ruby on Rails.

### 2. Entity Framework Core

- Why: Eases the data access and migration tasks, supports LINQ queries, and has a good integration with Azure SQL.
- Alternative: Dapper, ADO.NET or NHibernate.

### 3. Bootstrap

- Why: With very minimal effort, allows the creation of modern user interfaces that are responsive and align with contemporary design standards.
- Alternative: Tailwind CSS, Materialize or custom made CSS.

## **Alternative Components Discussion**

Web Hosting:

Instead of Azure App Service, AWS Elastic Beanstalk or Google App Engine could provide similar managed hosting. On-premises IIS is an option but lacks cloud scalability.

Database:

Azure Table Storage or Cosmos DB could be used for NoSQL scenarios, but lack relational features needed for this app. AWS RDS or Google Cloud SQL are also alternatives.

Storage:

Amazon S3 or Google Cloud Storage offer similar blob storage capabilities. Local file storage is possible but not scalable or cloud-friendly. (Anon., n.d.)

## **Self-reflection:**

The PoE went okay in my view, it had a consistent flow with the questions asked and the tasks that were needed to do in order to complete the project. Part 1 was good, with the mixture of practical work and theory it set a good starting point for the projects needs and understanding, testing our knowledge and understanding in both ways.

Part 2 of the POE was a bit more challenging, as it need more functionality for the web app. Personally I found the azure database connection and configuration trickier than the coding because my azure account kept giving me issues which caused my web app to not function the way it is supposed to which also caused some delay in finalising my submission. The final part is a bit more easier in terms of work load, but needs concentration and time in order to complete which I like. What I learnt form the project is time management and how to build a web application and use azure cloud services. Designing for the cloud means considering security, cost, and performance from the start. I understand the value of using managed services. This project has prepared me to architect, develop, and deploy robust cloud solutions using Azure and modern web technologies.

## Bibliography

- Anon., n.d. *ChatGpt*. [Online]  
Available at: <https://chatgpt.com/>  
[Accessed 20 June 2025].
- BillWagner, 2024. *Resolve errors and warnings related to assembly references*. [Online]  
Available at: [https://learn.microsoft.com/en-us/dotnet/csharp/language-reference/compiler-messages/assembly-references?f1url=%3FappId%3Droslyn%26k%3Dk\(CS0234\)](https://learn.microsoft.com/en-us/dotnet/csharp/language-reference/compiler-messages/assembly-references?f1url=%3FappId%3Droslyn%26k%3Dk(CS0234))  
[Accessed 10 May 2025].
- Brush, K., 2023. *cognitive search*. [Online]  
Available at: <https://www.techtarget.com/searchenterpriseai/definition/cognitive-search>  
[Accessed 10 May 2025].
- HeidiSteen, e.-u. s.-a., 2025. *What's Azure AI Search?*. [Online]  
Available at: <https://learn.microsoft.com/en-us/azure/search/search-what-is-azure-search>  
[Accessed 10 May 2025].
- Kerner, S. M., 2024. <https://www.techtarget.com/whatis/feature/GenAI-search-vs-traditional-search-engines-How-they-differ#:~:text=Traditional%20search%20engines%20help%20users,ands%20brief%20description%20or%20snippet..> [Online]  
Available at: <https://www.techtarget.com/whatis/feature/GenAI-search-vs-traditional-search-engines-How-they-differ#:~:text=Traditional%20search%20engines%20help%20users,ands%20brief%20description%20or%20snippet..>  
[Accessed 10 May 2025].
- Kordos, A., 2024. *Normalization vs. Denormalization in databases*. [Online]  
Available at: <https://codilime.com/blog/normalization-vs-denormalization-in-databases/>  
[Accessed 10 May 2025].
- Mrzyglod, K., 2022. *Azure for developers , Implement rich Azure Paas Ecosystems using containers, serverless services, and storage solutions*. 2nd ed. Birmingham: Packt Publishing Ltd..
- Ron Karjian, Z. F., 2024. *On premises vs. cloud pros and cons, key differences*. [Online]  
Available at: <https://www.techtarget.com/searchcloudcomputing/tip/Evaluate-on-premises-vs-cloud-computing-pros-and-cons>  
[Accessed 01 April 2025].
- S., N., n.d. *On-Premises vs Cloud: Key Differences, Pros & Cons*. [Online]  
Available at: <https://www.cloudpanel.io/blog/on-premises-vs-cloud-computing/>  
[Accessed 01 April 2025].