ST10384670

Name: Zanenhlanhla. Tshenolo. Konjwayo

Module: CLOUD DEVELOPMENT A

Component Discussion

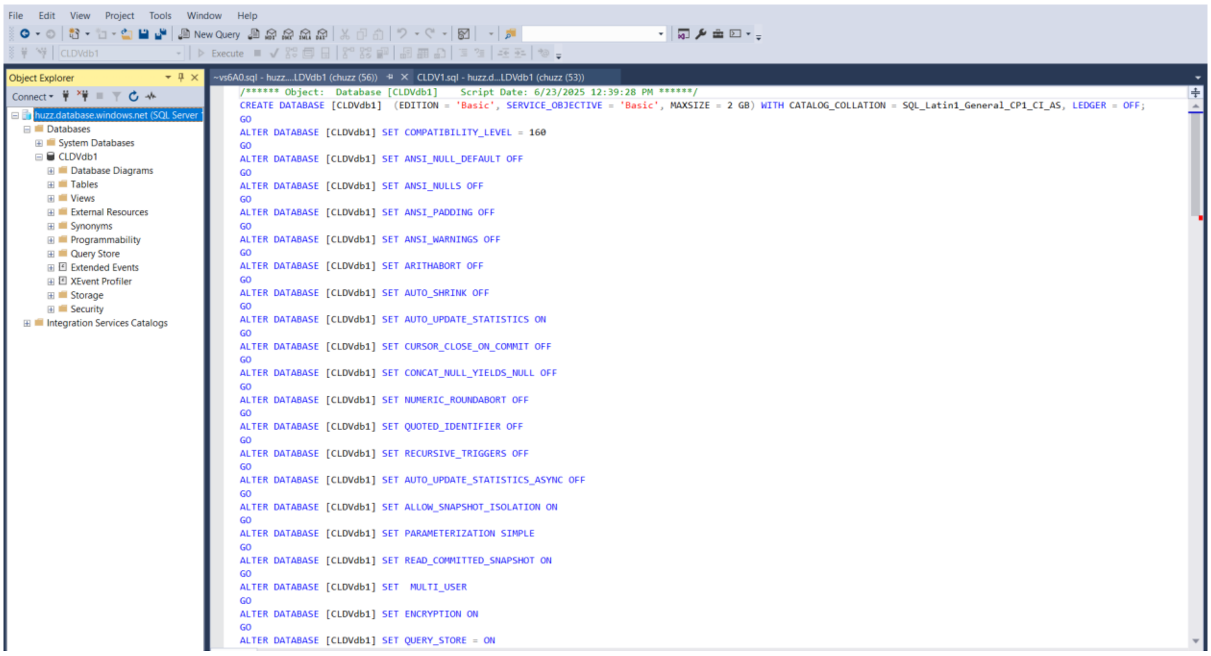
Technology Choices and Reasons

ASP.NET Core MVC

Because of its scalability, resilience, and close connection with Azure services, ASP.NET Core MVC was selected. It makes it possible for Models, Views, and Controllers to have their concerns clearly separated, which aids in logical system organization.

SQL Server

Because of its dependability, native cloud compatibility, and smooth Entity Framework integration, SQL Server was chosen. A scalable, cloud-hosted system that accommodates relational data and foreign key constraints was made possible by the use of Azure SQL Database.

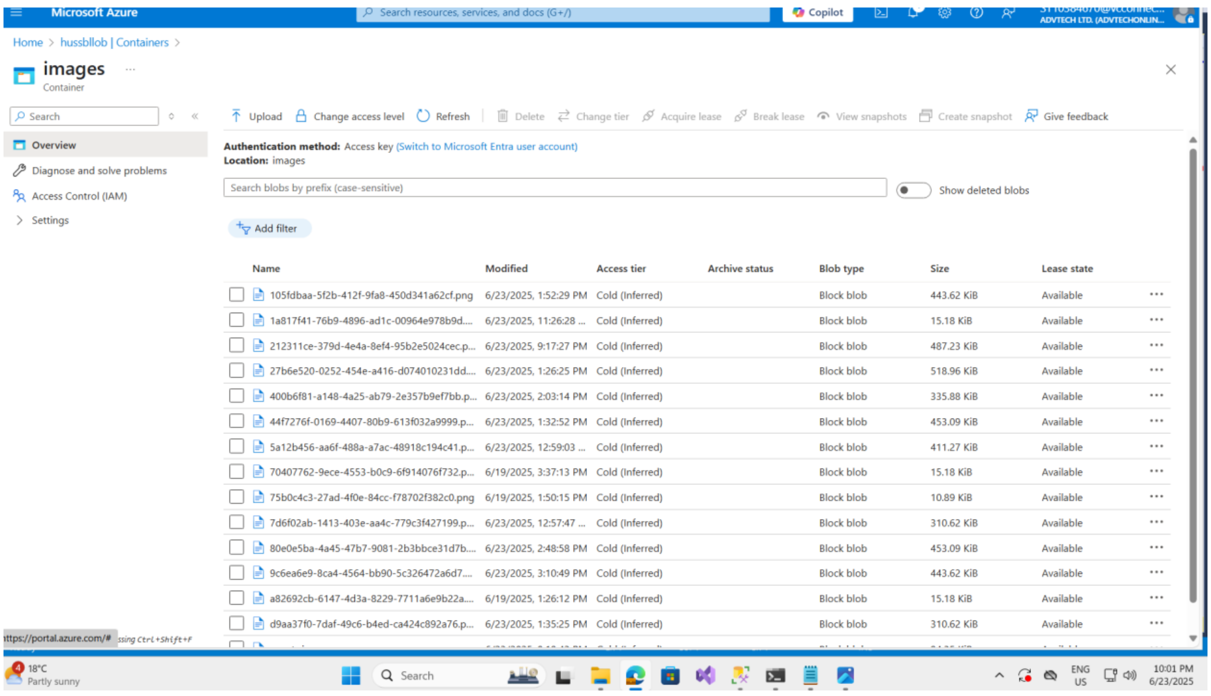


```sql
/****** Object:  Database [CLDVdb1]    Script Date: 6/23/2025 12:39:28 PM ******/
CREATE DATABASE [CLDVdb1]  (EDITION = 'Basic', SERVICE_OBJECTIVE = 'Basic', MAXSIZE = 2 GB) WITH CATALOG_COLLATION = SQL_Latin1_General_CP1_CI_AS, LEDGER = OFF;
GO
ALTER DATABASE [CLDVdb1] SET COMPATIBILITY_LEVEL = 160
GO
ALTER DATABASE [CLDVdb1] SET ANSI_NULL_DEFAULT OFF
GO
ALTER DATABASE [CLDVdb1] SET ANSI_NULLS OFF
GO
ALTER DATABASE [CLDVdb1] SET ANSI_PADDING OFF
GO
ALTER DATABASE [CLDVdb1] SET ANSI_WARNINGS OFF
GO
ALTER DATABASE [CLDVdb1] SET ARITHABORT OFF
GO
ALTER DATABASE [CLDVdb1] SET AUTO_SHRINK OFF
GO
ALTER DATABASE [CLDVdb1] SET AUTO_UPDATE_STATISTICS ON
GO
ALTER DATABASE [CLDVdb1] SET CURSOR_CLOSE_ON_COMMIT OFF
GO
ALTER DATABASE [CLDVdb1] SET CONCAT_NULL_YIELDS_NULL OFF
GO
ALTER DATABASE [CLDVdb1] SET NUMERIC_ROUNDABORT OFF
GO
ALTER DATABASE [CLDVdb1] SET QUOTED_IDENTIFIER OFF
GO
ALTER DATABASE [CLDVdb1] SET RECURSIVE_TRIGGERS OFF
GO
ALTER DATABASE [CLDVdb1] SET AUTO_UPDATE_STATISTICS_ASYNC OFF
GO
ALTER DATABASE [CLDVdb1] SET ALLOW_SNAPSHOT_ISOLATION ON
GO
ALTER DATABASE [CLDVdb1] SET PARAMETERIZATION SIMPLE
GO
ALTER DATABASE [CLDVdb1] SET READ_COMMITTED_SNAPSHOT ON
GO
ALTER DATABASE [CLDVdb1] SET  MULTI_USER
GO
ALTER DATABASE [CLDVdb1] SET ENCRYPTION ON
GO
ALTER DATABASE [CLDVdb1] SET QUERY_STORE = ON
```

Sql data base

Azure Blob Storage

Because Azure Blob Storage is very scalable and perfect for storing huge binary things like venue and event photographs, it was used to manage images. It also facilitates safe storage and speedy retrieval.



Home > hussbllob | Containers >

images
Container

Authentication method: Access key (Switch to Microsoft Entra user account)
Location: images

Name	Modified	Access tier	Archive status	Blob type	Size	Lease state
105fdbaa-5f2b-412f-9fa8-450d341a62cf.png	6/23/2025, 1:52:29 PM	Cold (Inferred)		Block blob	443.62 KiB	Available
1a817f41-76b9-4896-ad1c-00964e978b9d....	6/23/2025, 11:26:28 ...	Cold (Inferred)		Block blob	15.18 KiB	Available
212311ce-379d-4e4a-8ef4-95b2e5024cec.p...	6/23/2025, 9:17:27 PM	Cold (Inferred)		Block blob	487.23 KiB	Available
27b6e520-0252-454e-a416-d074010231dd....	6/23/2025, 1:26:25 PM	Cold (Inferred)		Block blob	518.96 KiB	Available
400b6f81-a148-4a25-ab79-2e357b9ef7bb.p...	6/23/2025, 2:03:14 PM	Cold (Inferred)		Block blob	335.88 KiB	Available
44f7276f-0169-4407-80b9-613f032a9999.p...	6/23/2025, 1:32:52 PM	Cold (Inferred)		Block blob	453.09 KiB	Available
5a12b456-aa6f-488a-a7ac-48918c194c41.p...	6/23/2025, 12:59:03 ...	Cold (Inferred)		Block blob	411.27 KiB	Available
70407762-9ece-4553-b0c9-6f914076f732.p...	6/19/2025, 3:37:13 PM	Cold (Inferred)		Block blob	15.18 KiB	Available
75b0c4c3-27ad-4f0e-84cc-f78702f382c0.png	6/19/2025, 1:50:15 PM	Cold (Inferred)		Block blob	10.89 KiB	Available
7d6f02ab-1413-403e-aa4c-779c3f427199.p...	6/23/2025, 12:57:47 ...	Cold (Inferred)		Block blob	310.62 KiB	Available
80e0e5ba-4a45-47b7-9081-2b3bbce31d7b....	6/23/2025, 2:48:58 PM	Cold (Inferred)		Block blob	453.09 KiB	Available
9c6ea6e9-8ca4-4564-bb90-5c326472a6d7....	6/23/2025, 3:10:49 PM	Cold (Inferred)		Block blob	443.62 KiB	Available
a82692cb-6147-4d3a-8229-7711a6e9b22a....	6/19/2025, 1:26:12 PM	Cold (Inferred)		Block blob	15.18 KiB	Available
d9aa37f0-7daf-49c6-b4ed-ca424c892a76.p...	6/23/2025, 1:35:25 PM	Cold (Inferred)		Block blob	310.62 KiB	Available

EF Core

ORM mapping was done with EF Core, which allowed for effective communication with the Azure SQL Database using LINQ queries and strongly typed C# classes. It made CRUD and database administration easier.

The Azure App Service

The web application was deployed in the cloud using the Azure App Service, which enables seamless scaling, continuous deployment, and direct GitHub connection for modifications in the future.

HTML and Bootstrap

A responsive and aesthetically pleasing front-end was developed using HTML and Bootstrap, while ASP.NET MVC's Razor Views enabled dynamic content rendering.

Alternative Elements (Taken into Account)
Database storage might have been done with Firebase or MongoDB Atlas, but unlike Azure SQL, they do not natively enforce relational fidelity.

Although Azure Blob Storage offered superior interoperability with the rest of the Azure ecosystem, Amazon S3 could have been utilized for picture storage.

Although they would have required more API levels and would have overcomplicated the project scope, Angular or React might have been utilized for the front end.

Reflection on the Project

Throughout my academic career, this cloud development project has been one of the most rewarding and challenging experiences I've had.
I lost the complete submission in Part 1 since I had trouble integrating GitHub and was unable to push my code in time. This was really discouraging, but I used it as a lesson rather than abandoning up.

Part 2 presented yet another set of challenges. Although I was able to get the system to function, my solution did not adhere to the rubric's requirements. Important aspects like adequate image storage and thorough validation were absent from my implementation. I came to the realization that in order to create a workable solution, I needed to focus more on the details and fully comprehend the rubric's requirements.

I made a self-promise that this would be my comeback when I got to Part 3. I pushed myself to create a completely working, cohesive solution by taking all of the criticism, errors, and frustrations from the preceding sections. I concentrated on:

.being aware of the connections between the models.
.enhancing my ability to handle errors.
.Understanding Azure deployment procedures to guarantee a functional system.

Along the process, I ran into a number of technical blunders, including SQL migration issues, mismatched model relationships, and foreign key concerns. However, I persevered this time. I investigated, debugged, and improved the application until it performed as I had anticipated.

Knowledge Acquired

.Making a plan is important. I learned to always test my deployment processes before committing to them after missing out on Part 1's GitHub submission.

.Take note of the rubric. It's important to design the system they requested, not simply a system.

.Talent is surpassed by perseverance. Even though I made a lot of mistakes along the road, I managed to get everything together for Part 3 by persevering.

.Full-stack thinking is necessary while using cloud technologies. Getting the web application to function locally is insufficient; data integrity, scalability, and cloud deployment are equally crucial.

Present Knowledge about Cloud Applications

I now understand the design and development of cloud-based applications on a deeper level. Coding is only one aspect of cloud development; another is comprehension:

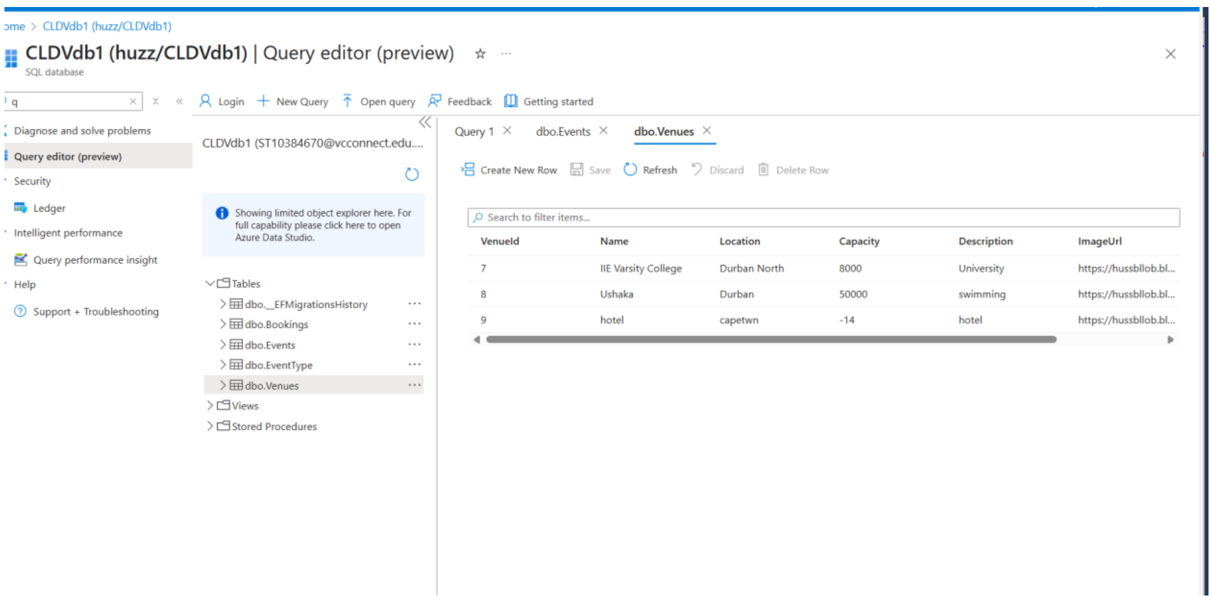
- Deployment pipelines
- Cloud storage solutions
- Data management at scale
- Error handling in a live environment
- Building for real users

Additionally, I now know how to manage cloud-hosted databases, use Azure services efficiently, create a system that is both practical and easy to use, and organize projects according to best practices.

This project served as my atonement, and I am pleased with the results of this last section. Despite my past mistakes, I've learned a lot and can claim with confidence that I now know what it takes to create, implement, and maintain a cloud-based web service.

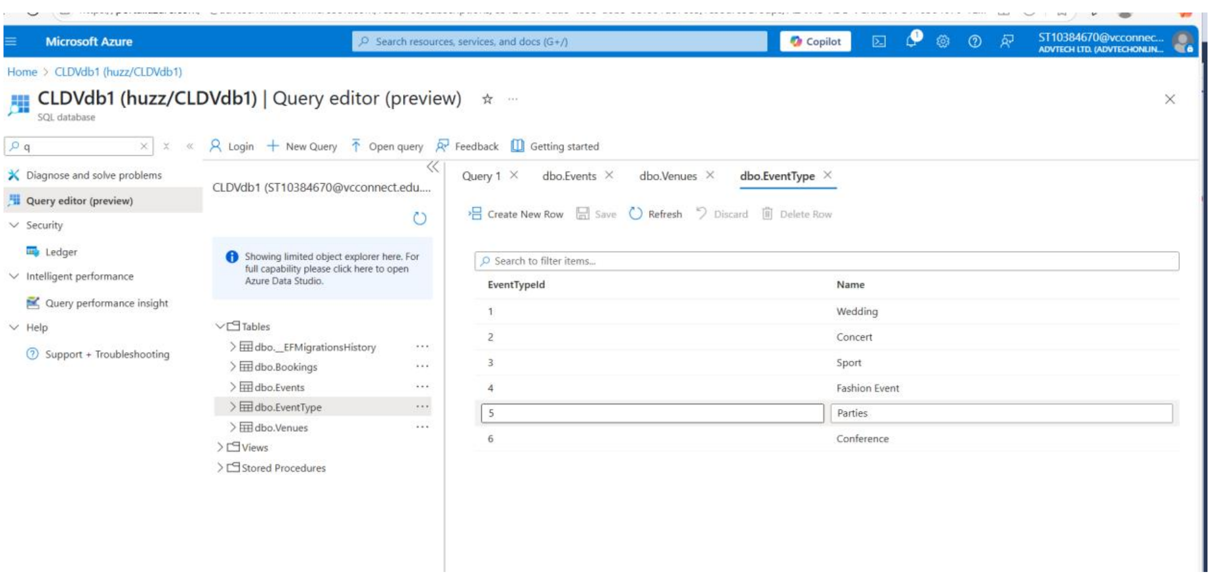
Screenshots of azure labs

This is my new Venue table in the query editor


ome > CLDVdb1 (huzz/CLDVdb1)
CLDVdb1 (huzz/CLDVdb1) | Query editor (preview)
SQL database
Diagnose and solve problems
Query editor (preview)
Security
Ledger
Intelligent performance
Query performance insight
Help
Support + Troubleshooting
Login
New Query
Open query
Feedback
Getting started
CLDVdb1 (ST10384670@vcconnect.edu....
Showing limited object explorer here. For full capability please click here to open Azure Data Studio.
Tables
dbo.__EFMigrationsHistory
dbo.Bookings
dbo.Events
dbo.EventType
dbo.Venues
Views
Stored Procedures
Query 1
dbo.Events
dbo.Venues
Create New Row
Save
Refresh
Discard
Delete Row
Search to filter items...

VenueId	Name	Location	Capacity	Description	ImageUrl
7	IIE Varsity College	Durban North	8000	University	https://hussbllob.bl...
8	Ushaka	Durban	50000	swimming	https://hussbllob.bl...
9	hotel	capetwn	-14	hotel	https://hussbllob.bl...

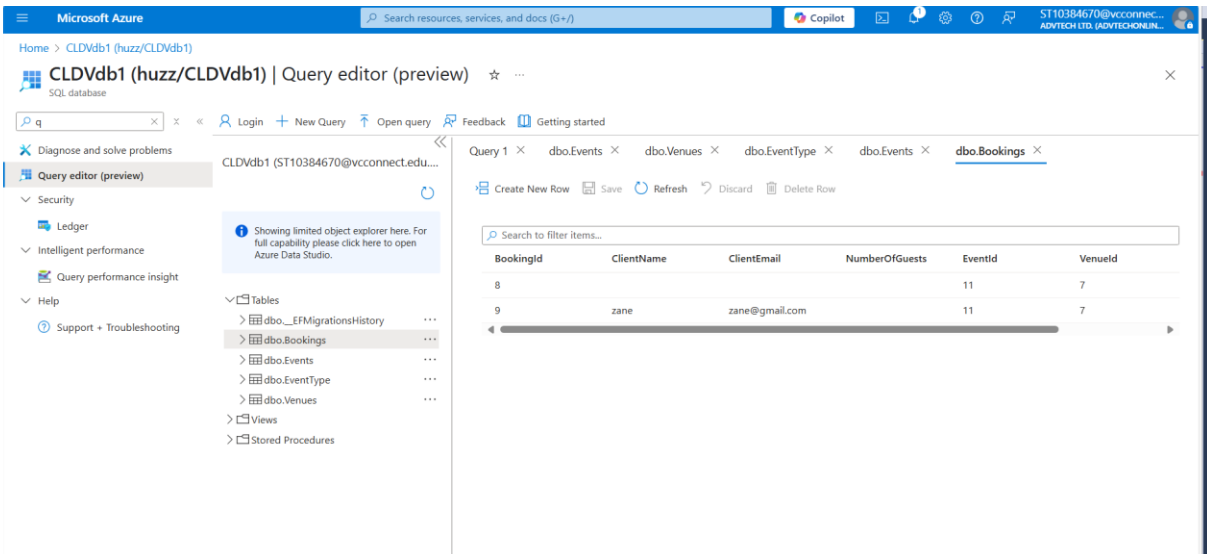
This is my my new Eventype table



Showing limited object explorer here. For full capability please click here to open Azure Data Studio.

EventTypeId	Name
1	Wedding
2	Concert
3	Sport
4	Fashion Event
5	Parties
6	Conference

This is my new Bookings table with filtering


Microsoft Azure
Search resources, services, and docs (G+/)
Copilot
ST10384670@vcconnec...
ADVTECH LTD. (ADVTECHONLIN...
Home > CLDVdb1 (huzz/CLDVdb1)
CLDVdb1 (huzz/CLDVdb1) | Query editor (preview)
SQL database
Diagnose and solve problems
Query editor (preview)
Security
Ledger
Intelligent performance
Query performance insight
Help
Support + Troubleshooting
Login
New Query
Open query
Feedback
Getting started
CLDVdb1 (ST10384670@vcconnect.edu....
Showing limited object explorer here. For full capability please click here to open Azure Data Studio.
Tables
dbo.__EFMigrationsHistory
dbo.Bookings
dbo.Events
dbo.EventType
dbo.Venues
Views
Stored Procedures
Query 1
dbo.Events
dbo.Venues
dbo.EventType
dbo.Events
dbo.Bookings
Create New Row
Save
Refresh
Discard
Delete Row
Search to filter items...

BookingId	ClientName	ClientEmail	NumberOfGuests	EventId	VenueId
8				11	7
9	zane	zane@gmail.com		11	7

References

- Microsoft. (n.d.). *ASP.NET Core MVC overview*. Microsoft Learn. Retrieved June 23, 2025, from https://learn.microsoft.com/en-us/aspnet/core/mvc/overview
- Microsoft. (n.d.). *What is Azure SQL Database?*. Microsoft Learn. Retrieved June 23, 2025, from https://learn.microsoft.com/en-us/azure/azure-sql/database/sql-database-paas-overview
- Microsoft. (n.d.). *Introduction to Azure Blob Storage*. Microsoft Learn. Retrieved June 23, 2025, from https://learn.microsoft.com/en-us/azure/storage/blobs/storage-blobs-introduction
- Microsoft. (n.d.). *Entity Framework Core Overview*. Microsoft Learn. Retrieved June 23, 2025, from https://learn.microsoft.com/en-us/ef/core/
- Microsoft. (n.d.). *What is Azure App Service?*. Microsoft Learn. Retrieved June 23, 2025, from https://learn.microsoft.com/en-us/azure/app-service/overview
- W3Schools. (n.d.). *Bootstrap 4 Tutorial*. Retrieved June 23, 2025, from https://www.w3schools.com/bootstrap4/
- Microsoft. (n.d.). *Razor syntax for ASP.NET Core*. Microsoft Learn. Retrieved June 23, 2025, from https://learn.microsoft.com/en-us/aspnet/core/mvc/views/razor