Southern New Hampshire University

Ilyass Hmamou

ilyass.hmamou@snhu.edu

CS-499: Computer Science Capstone

5-2 Milestone Four: Enhancement Three: Databases

1- Artifact description:

For the improvement in this category of Databases, I continued working with the "Salvare Search For Rescue App." artifact. As a reminder, It was developed during the period between September 2023 and October 2023 as a component of the final project for CS-340: Client/Server development course.

The focus in this enhancement is not placed on the code itself but rather on the data. While the application offers a user interface for visualizing data records through interactive features like a datatable, map, and pie chart, it currently lacks the capability to edit the data. The primary goal of this enhancement is to empower authenticated users with the **security clearance** to modify the data using a third-party UI. This integration involves connecting with the dataset. Regarding the dataset, it was imperative to host the data on the cloud to enable seamless integration with the third-party UI. This aspect involves **utilizing skills in DevOps and database management**, making it a pivotal component of the overall enhancement.

2- Reason of choosing this artifact:

This artifact offers notable prospects for progress, providing an opportunity for me to demonstrate my skills in analyzing ongoing projects, identifying deficiencies, and executing effective solutions. Within this particular refinement, my emphasis was on elevating the user experience during data interactions. The execution of this improvement involved establishing a cloud database host to manage our working dataset, generating authenticated user profiles, and integrating with a third-party user interface. The goal was to augment the user experience by enhancing accessibility and user-friendliness, readability and availability of data, data security, streamlining updates and maintenance processes, promoting collaboration and data sharing, and ultimately ensuring scalability without requiring user intervention.

3- Enhancement technical details:

As previously stated, this improvement centered on transferring the data to a cloud server and establishing a straightforward method to link to the server and engage with the data through a third-party integrated UI tool. In this segment of the documentation, I will outline a step-by-step configuration process pertaining to configuring the database on the server, generating user profiles for database access, and finally establishing a connection to the database using the third-party UI tool.

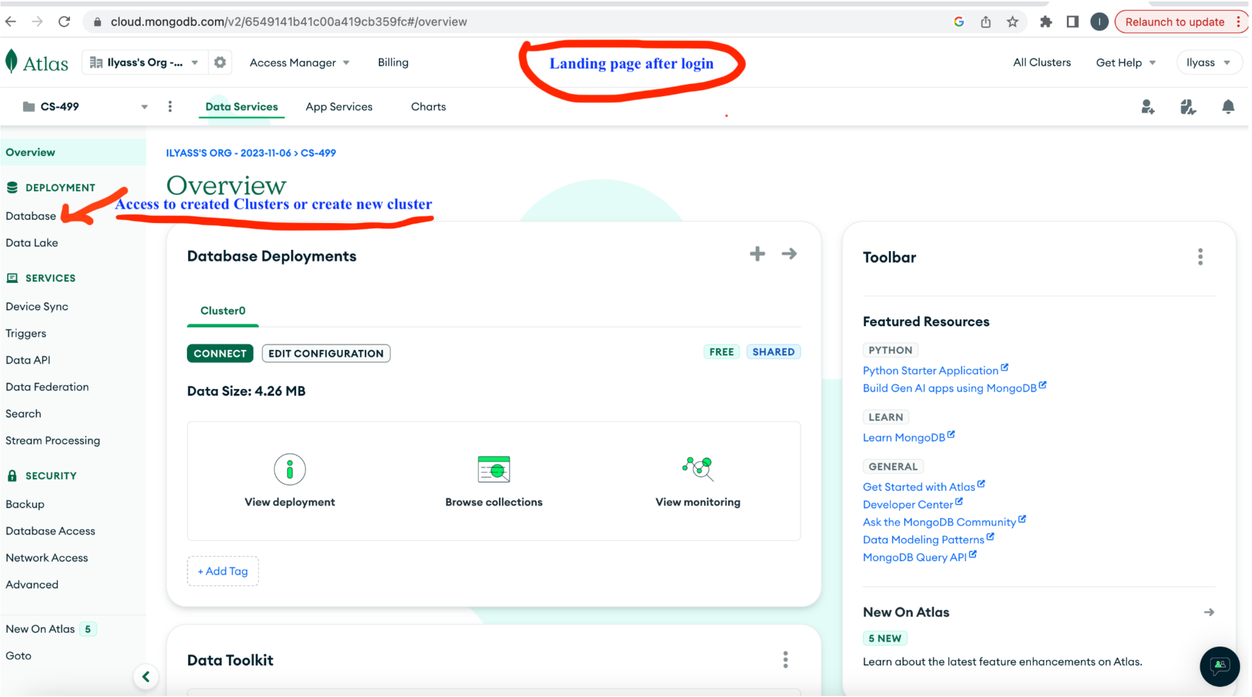
3.1 Cloud database details:

The server used is MongoDB Atlas which is a fully-managed cloud database that handles all the complexity of deploying, managing, and healing your deployments on the cloud service provider of your choice (AWS , Azure, and GCP).

Link: https://www.mongodb.com/

3.2 Database creation within Atlas:

Once you login to Mongodb Atlas, you will land on the following page:



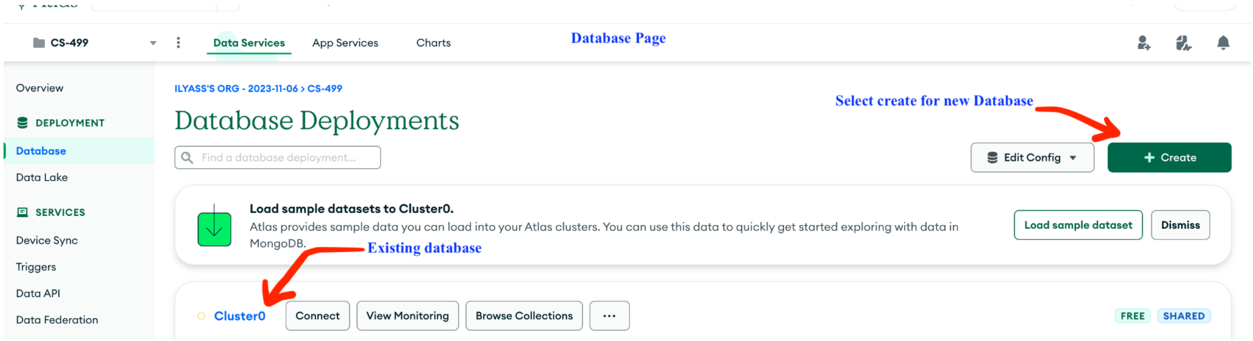
Landing page after login

Access to created Clusters or create new cluster

Data Size: 4.26 MB

Learn about the latest feature enhancements on Atlas.

Click on the Database option on the right side menu



Database Page

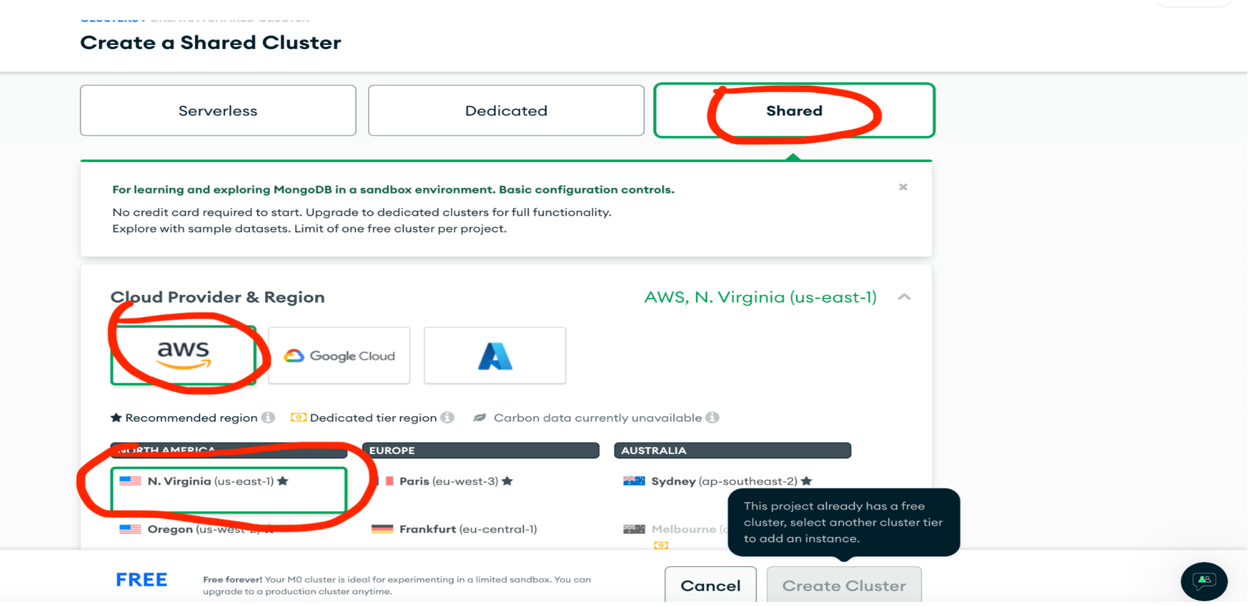
Select create for new Database

Existing database

Load sample datasets to Cluster0.

Atlas provides sample data you can load into your Atlas clusters. You can use this data to quickly get started exploring with data in MongoDB.

Once you select Create, you get the following page where you can choose your host cloud provider and your preferred location/timezone



Create a Shared Cluster

For learning and exploring MongoDB in a sandbox environment. Basic configuration controls.

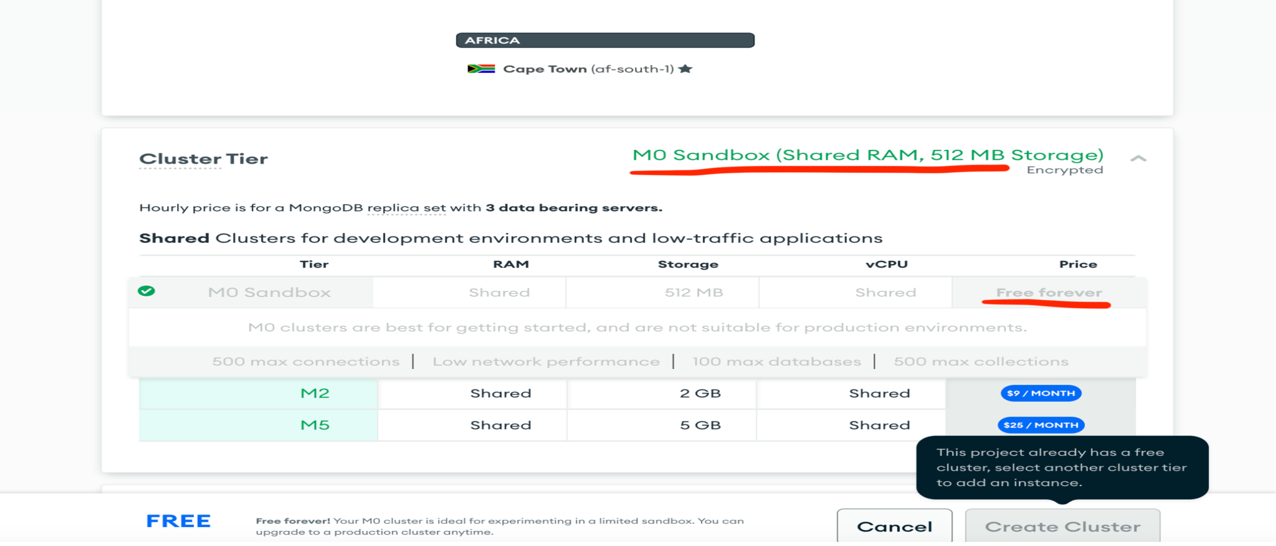
No credit card required to start. Upgrade to dedicated clusters for full functionality.
Explore with sample datasets. Limit of one free cluster per project.

Cloud Provider & Region — AWS, N. Virginia (us-east-1)

This project already has a free cluster, select another cluster tier to add an instance.

FREE — **Free forever!** Your M0 cluster is ideal for experimenting in a limited sandbox. You can upgrade to a production cluster anytime.

For this course, I'm using the Shared Cluster (has a free trial option), with AWS as the cloud provider and N.Virginia as the region.



Cluster Tier — M0 Sandbox (Shared RAM, 512 MB Storage) Encrypted

Hourly price is for a MongoDB replica set with **3 data bearing servers.**

Shared Clusters for development environments and low-traffic applications

Tier	RAM	Storage	vCPU	Price
✓ M0 Sandbox	Shared	512 MB	Shared	Free forever
M2	Shared	2 GB	Shared	$9 / MONTH
M5	Shared	5 GB	Shared	$25 / MONTH

M0 clusters are best for getting started, and are not suitable for production environments.

500 max connections | Low network performance | 100 max databases | 500 max collections

This project already has a free cluster, select another cluster tier to add an instance.

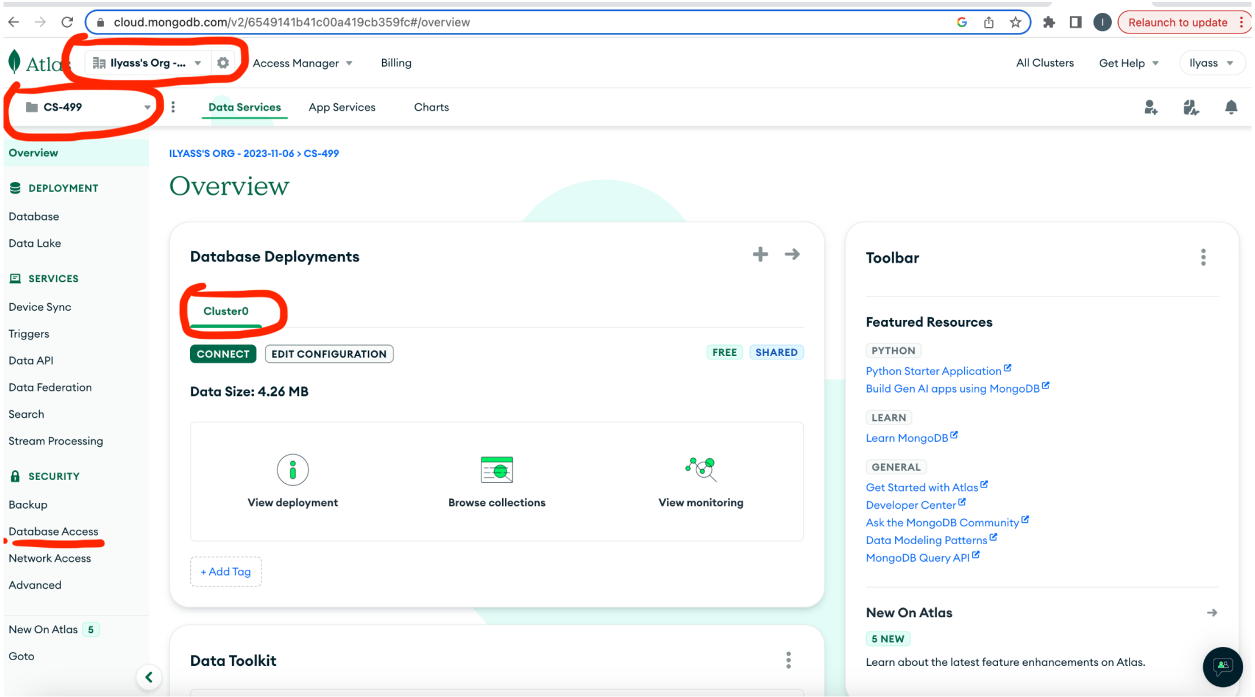
FREE — **Free forever!** Your M0 cluster is ideal for experimenting in a limited sandbox. You can upgrade to a production cluster anytime.

Once you select the cluster Tier you want to have, select Create Cluster.

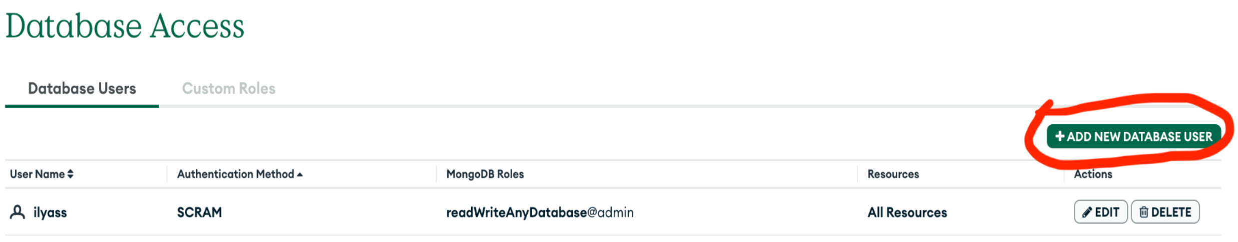
Note: The host permits only one complimentary sandbox cluster per account. The "Create Cluster" option is deactivated in this screenshot since I had already established a cluster a few weeks ago for the purpose of this course.

3.3 New user creation

After creating the database cluster, the next step is to configure a user. Ensure that you are on the landing page within the desired project and cluster. Then, choose "Database access" from the menu on the right side.



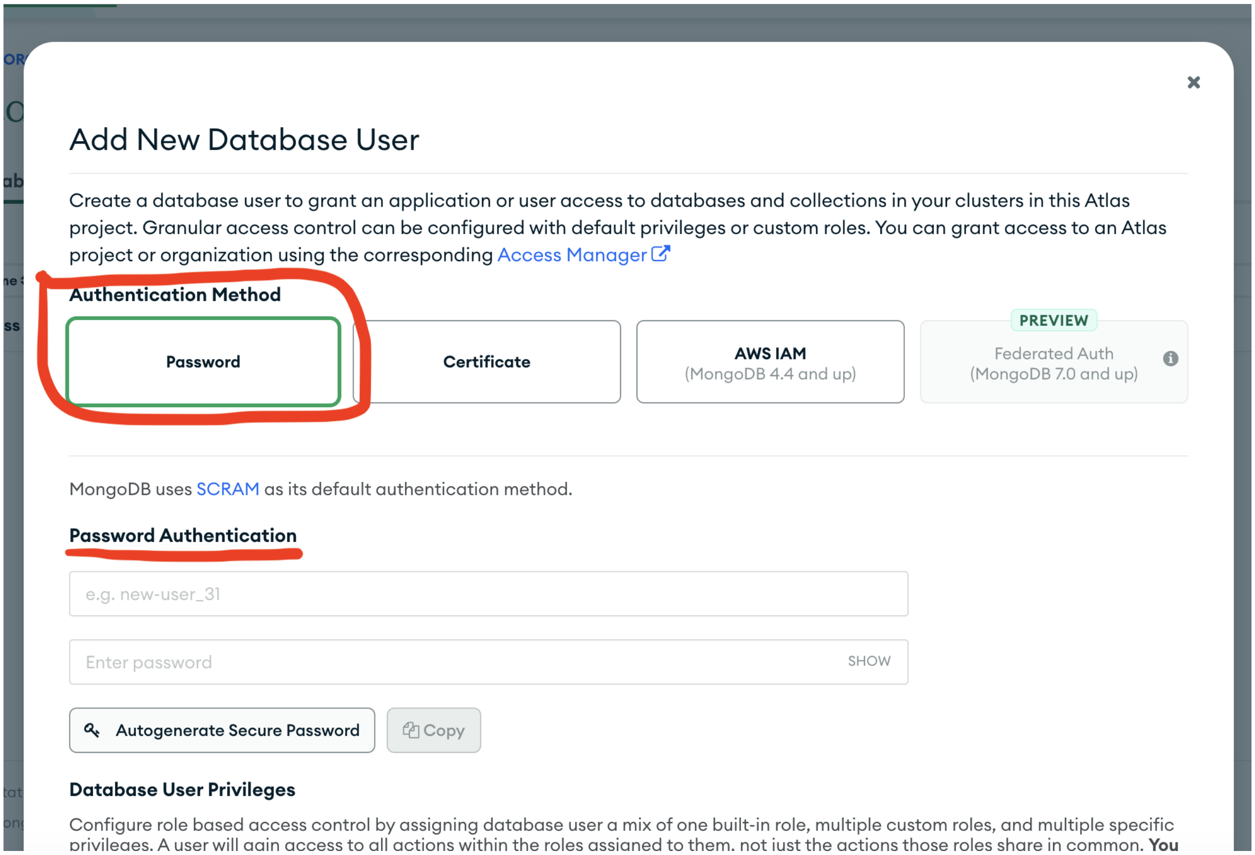
On the next page, select "Add new Database user"



Database Access

User Name	Authentication Method	MongoDB Roles	Resources	Actions
ilyass	SCRAM	readWriteAnyDatabase@admin	All Resources	EDIT DELETE

Next, pick your authentication method. In this course, I opted for the Username and Password authentication method.



Add New Database User

Create a database user to grant an application or user access to databases and collections in your clusters in this Atlas project. Granular access control can be configured with default privileges or custom roles. You can grant access to an Atlas project or organization using the corresponding Access Manager

Authentication Method

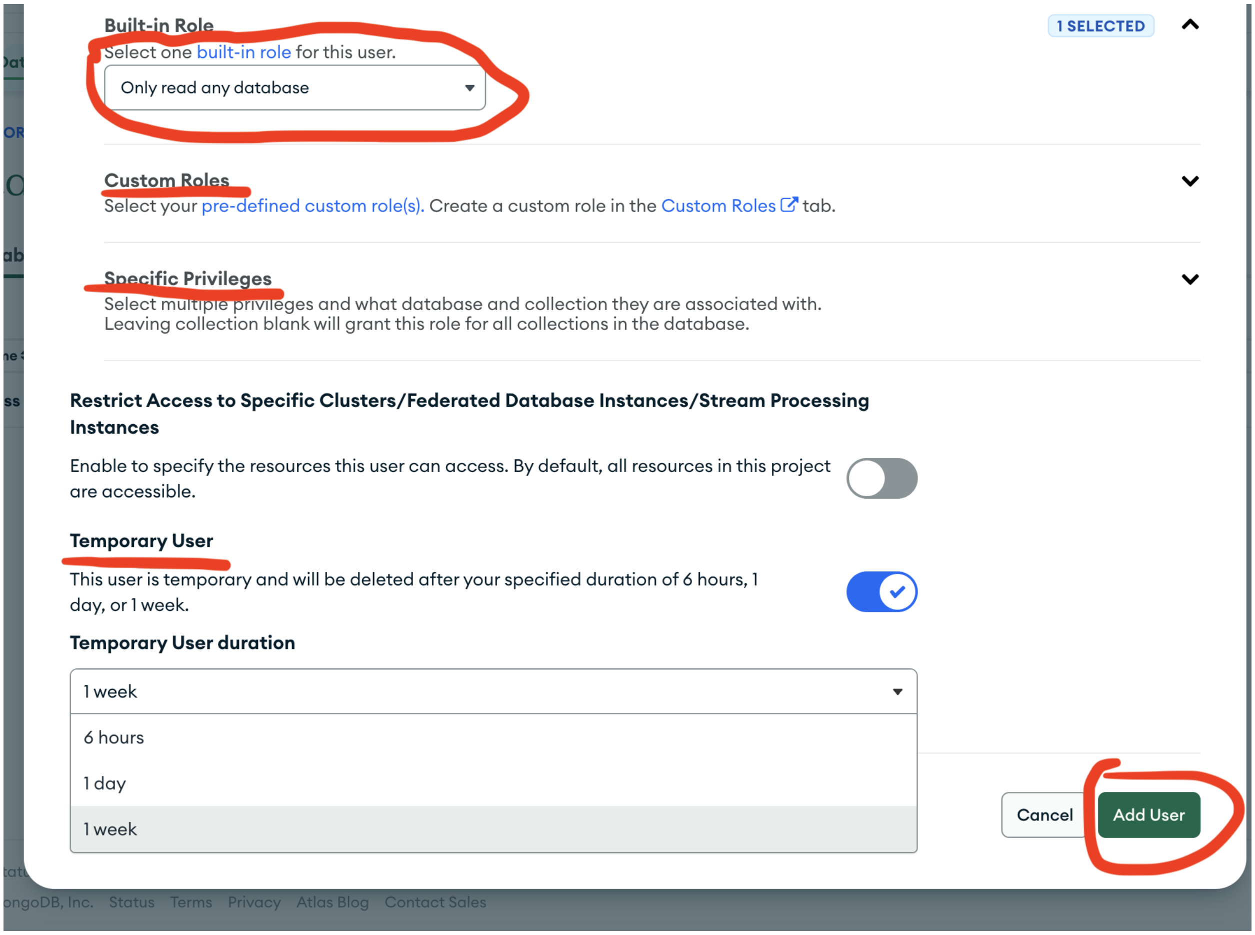
MongoDB uses SCRAM as its default authentication method.

Password Authentication

Database User Privileges

Configure role based access control by assigning database user a mix of one built-in role, multiple custom roles, and multiple specific privileges. A user will gain access to all actions within the roles assigned to them, not just the actions those roles share in common. You

In the upcoming section, configure the user's permissions. You have the option to select Built-in permissions, providing choices like read-only, read and write, or admin. Alternatively, you can opt for a more advanced option offering detailed permission settings. For the simplicity of this course, I've chosen one of the Built-in permissions to streamline the process.



Built-in Role

Select one built-in role for this user.

Only read any database

Custom Roles

Select your pre-defined custom role(s). Create a custom role in the Custom Roles tab.

Specific Privileges

Select multiple privileges and what database and collection they are associated with.
Leaving collection blank will grant this role for all collections in the database.

Restrict Access to Specific Clusters/Federated Database Instances/Stream Processing Instances

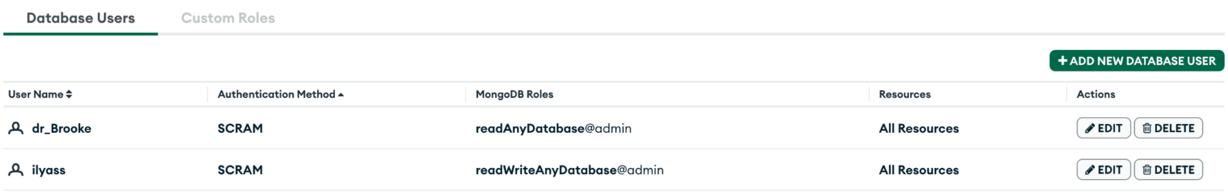
Enable to specify the resources this user can access. By default, all resources in this project are accessible.

Temporary User

This user is temporary and will be deleted after your specified duration of 6 hours, 1 day, or 1 week.

Temporary User duration

1 week

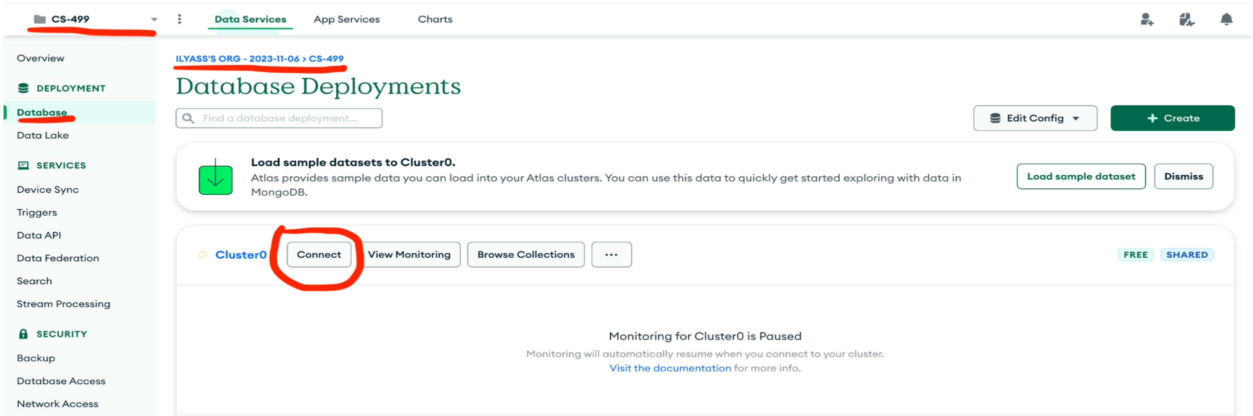


User Name	Authentication Method	MongoDB Roles	Resources	Actions
dr_Brooke	SCRAM	readAnyDatabase@admin	All Resources	EDIT DELETE
ilyass	SCRAM	readWriteAnyDatabase@admin	All Resources	EDIT DELETE

3.4 Remote Connection to Database using Compass

To establish a remote connection, we utilize MongoDB Compass—an effective graphical user interface for querying, aggregating, and analyzing MongoDB data visually. Compass is freely available and open-source, compatible with macOS, Windows, and Linux. Download Compass and refer to the installation instructions for further details.

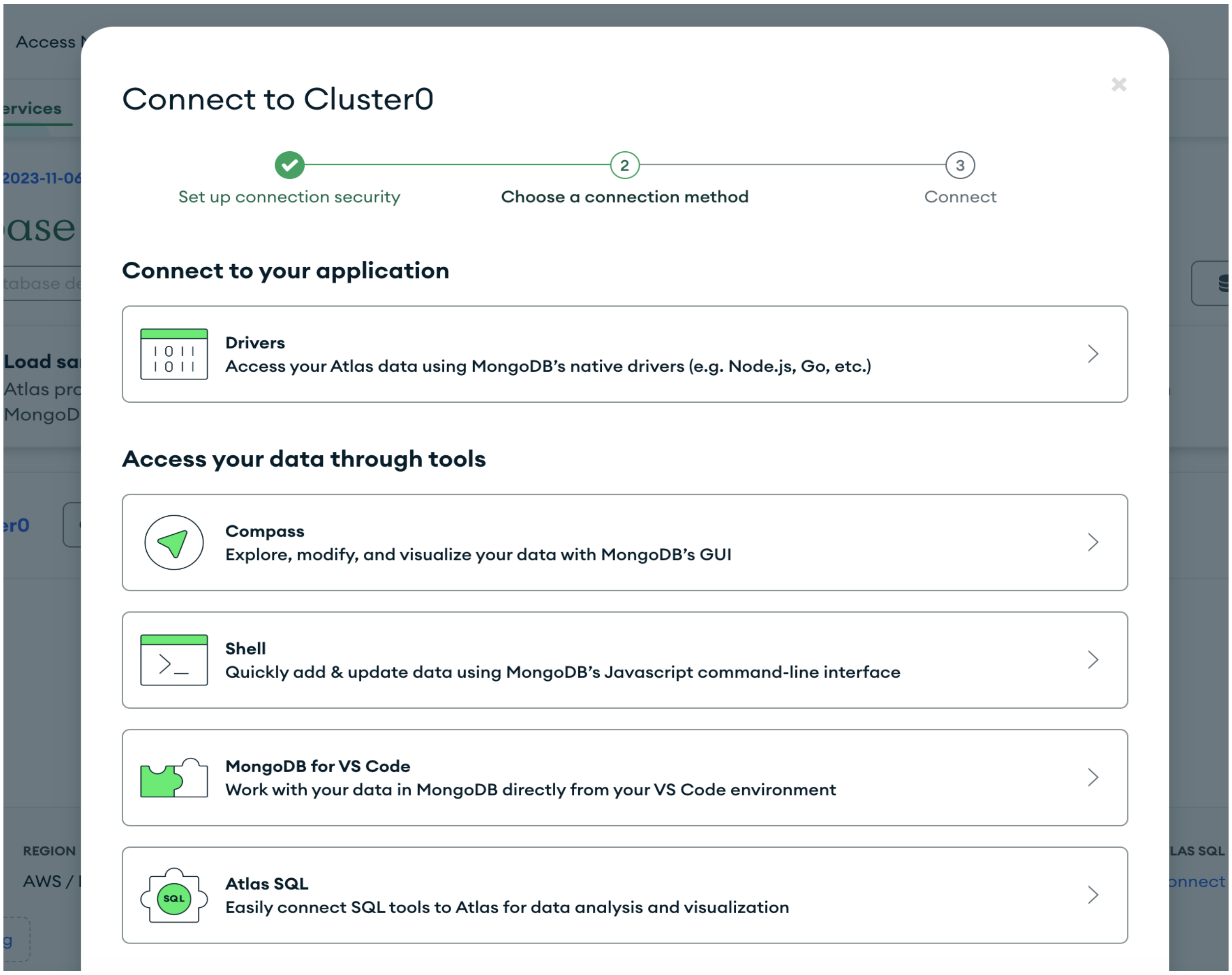
While on the Atlas MongoDB site, ensure you are within the desired project and cluster. Navigate to the right-side menu, select "Database," and then proceed to click on "Connect."



Load sample datasets to Cluster0.
Atlas provides sample data you can load into your Atlas clusters. You can use this data to quickly get started exploring with data in MongoDB.

Monitoring for Cluster0 is Paused
Monitoring will automatically resume when you connect to your cluster.
Visit the documentation for more info.

On the landing page for "Connect," you have the option to choose your preferred connection method. As previously detailed in this documentation, we will be using Compass.



Connect to Cluster0

Connect to your application

Drivers
Access your Atlas data using MongoDB's native drivers (e.g. Node.js, Go, etc.)

Access your data through tools

Compass
Explore, modify, and visualize your data with MongoDB's GUI

Shell
Quickly add & update data using MongoDB's Javascript command-line interface

MongoDB for VS Code
Work with your data in MongoDB directly from your VS Code environment

Atlas SQL
Easily connect SQL tools to Atlas for data analysis and visualization

Upon reaching the landing page for the "Compass" option, you'll find various download links for Compass. Below, there are links available for MacOS 64-bit (10.14+) and Windows 64-bit (10+).

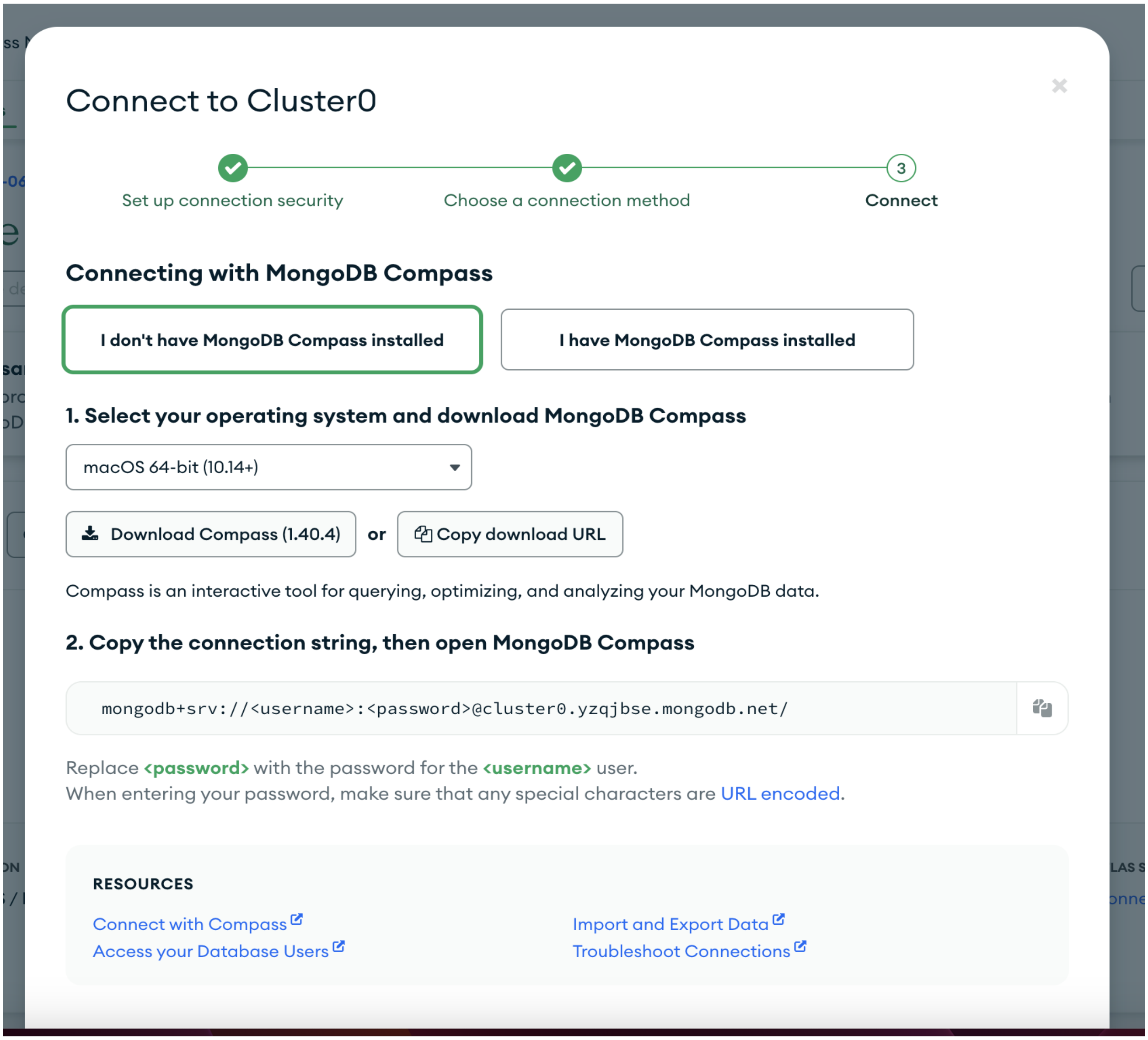
Compass download links:

MacOs 64 bit (10.14+):

https://downloads.mongodb.com/compass/mongodb-compass-1.40.4-darwin-x64.dmg

Windows 64 bit (10+):

https://downloads.mongodb.com/compass/mongodb-compass-1.40.4-win32-x64.exe



Connect to Cluster0

Set up connection security
Choose a connection method
3
Connect

Connecting with MongoDB Compass

I don't have MongoDB Compass installed
I have MongoDB Compass installed

1. Select your operating system and download MongoDB Compass

macOS 64-bit (10.14+)

Download Compass (1.40.4)
or
Copy download URL

Compass is an interactive tool for querying, optimizing, and analyzing your MongoDB data.

2. Copy the connection string, then open MongoDB Compass

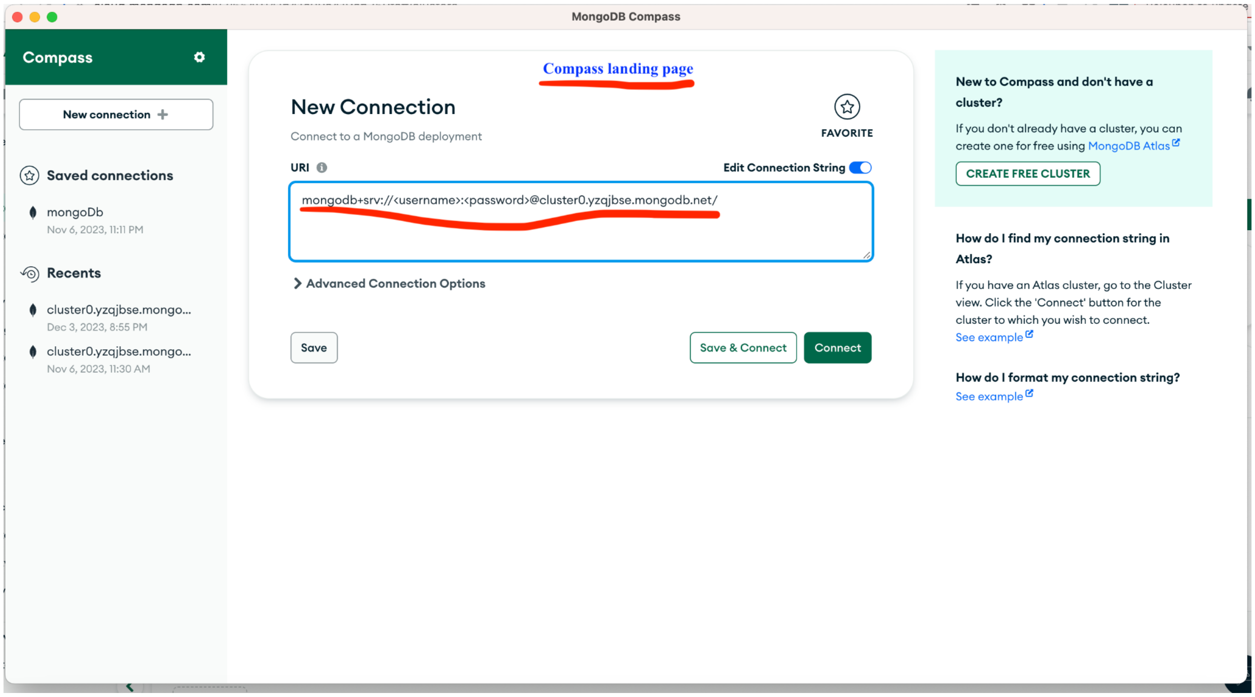
```
mongodb+srv://<username>:<password>@cluster0.yzqjbse.mongodb.net/
```

Replace <password> with the password for the <username> user.
When entering your password, make sure that any special characters are URL encoded.

Once Compass is downloaded and installed, open Compass and the following command to connect:

mongodb+srv://<username>:<password>@cluster0.yzqjbse.mongodb.net/

Replace <username> and <password> with your login information (I sent the login information on email)



Compass landing page

MongoDB Compass

New Connection

Connect to a MongoDB deployment

URI

mongodb+srv://<username>:<password>@cluster0.yzqjbse.mongodb.net/

Edit Connection String

New to Compass and don't have a cluster?

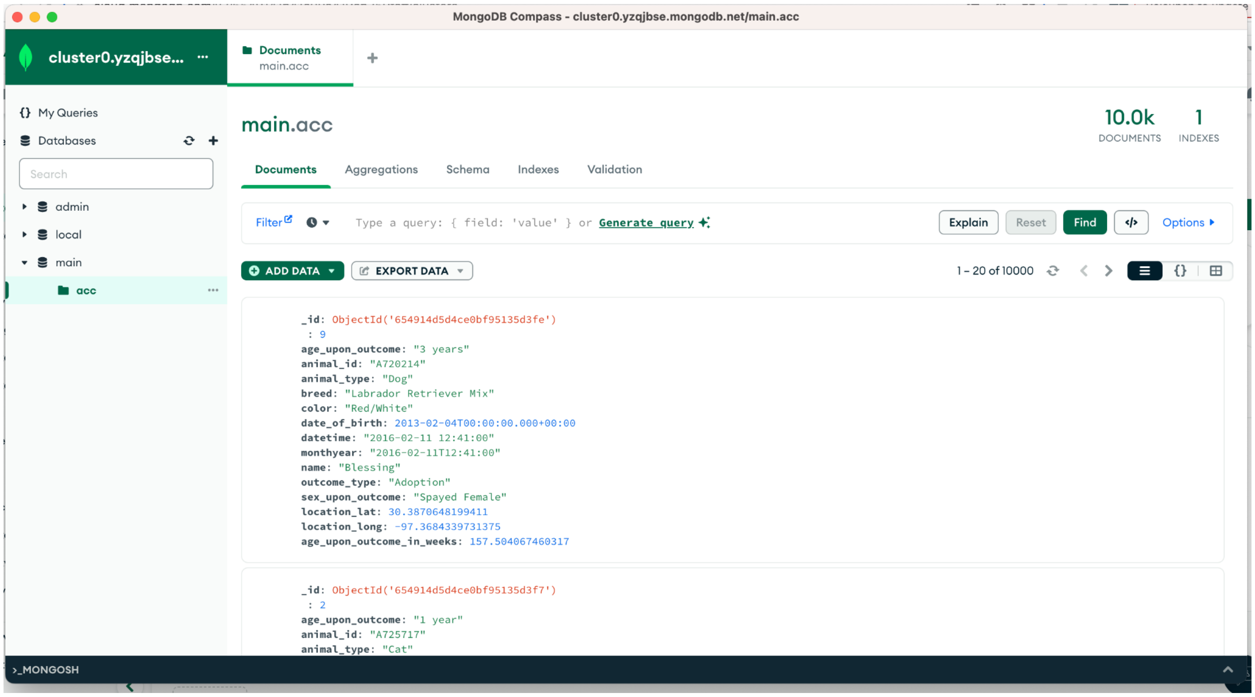
If you don't already have a cluster, you can create one for free using MongoDB Atlas

How do I find my connection string in Atlas?

If you have an Atlas cluster, go to the Cluster view. Click the 'Connect' button for the cluster to which you wish to connect. See example

How do I format my connection string? See example

Once you connect, you should be able to interact with the data



MongoDB Compass - cluster0.yzqjbse.mongodb.net/main.acc

main.acc

10.0k DOCUMENTS

1 INDEXES

Filter: Type a query: { field: 'value' } or Generate query

1 – 20 of 10000

```
_id: ObjectId('654914d5d4ce0bf95135d3fe')
 : 9
age_upon_outcome: "3 years"
animal_id: "A720214"
animal_type: "Dog"
breed: "Labrador Retriever Mix"
color: "Red/White"
date_of_birth: 2013-02-04T00:00:00.000+00:00
datetime: "2016-02-11 12:41:00"
monthyear: "2016-02-11T12:41:00"
name: "Blessing"
outcome_type: "Adoption"
sex_upon_outcome: "Spayed Female"
location_lat: 30.3870648199411
location_long: -97.3684339731375
age_upon_outcome_in_weeks: 157.504067460317

_id: ObjectId('654914d5d4ce0bf95135d3f7')
 : 2
age_upon_outcome: "1 year"
animal_id: "A725717"
animal_type: "Cat"
```

3.5 Benefits of using cloud solution for the database

In this segment, I'll elaborate on the significance of this enhancement and the benefits it offers to end-users, whether they are technical individuals (such as software engineers, architects, or database administrators) or non-technical users (regular users).

3.5.1: Benefits for technical individuals:

Hosting a dataset on a cloud database offers several benefits:

1. Scalability: Cloud databases provide scalable solutions, allowing you to easily scale your dataset storage as your data needs grow. This ensures that you can accommodate increasing volumes of data without worrying about infrastructure limitations.

2. Accessibility: Cloud databases enable remote access to your dataset from anywhere with an internet connection. This accessibility is particularly beneficial for teams spread across different locations or for applications that need to serve users globally.

3. Cost Efficiency: Cloud databases often follow a pay-as-you-go model, allowing you to pay only for the resources you use. This can be more cost-effective than maintaining and managing physical servers, especially for smaller projects or startups.

4. Reliability and High Availability: Cloud providers typically offer robust infrastructure with redundancy and backups, ensuring high availability and data durability. This reduces the risk of data loss and minimizes downtime.

5. Security Measures: Cloud providers invest heavily in security measures, including encryption, identity management, and compliance certifications. This helps enhance the overall security of your dataset, protecting it from unauthorized access or data breaches.

6. Automated Maintenance: Cloud databases often come with automated maintenance tasks, such as backups, software updates, and scaling operations. This reduces the administrative burden on your team and ensures that your database is well-maintained.

7. Integration with Other Cloud Services: Cloud databases can easily integrate with other cloud services, such as analytics tools, machine learning platforms, and serverless computing. This facilitates a more comprehensive and efficient data ecosystem.

8. Faster Deployment: Setting up a database on the cloud is typically quicker than procuring and configuring physical hardware. This allows for faster deployment of applications and reduces time-to-market for projects.

In summary, hosting a dataset on a cloud database provides advantages in terms of scalability, accessibility, cost efficiency, reliability, security, automated maintenance, integration capabilities, and faster deployment, making it a popular choice for many organizations and projects.

3.5.1: Benefits for non technical users:

1. Accessibility and Ease of Use: Cloud databases enable applications to run seamlessly without requiring end users to manage or understand the underlying technical infrastructure. This ensures that users can interact with applications and access data without the need for technical expertise.

2. Reliability and Availability: Cloud databases often provide high reliability and availability, which means end users can access their data whenever they need it. This reliability ensures a consistent and uninterrupted user experience.

3. Data Security: Cloud providers implement robust security measures to protect data. End users benefit from this by having their data stored in a secure environment without needing to actively manage security protocols.

4. Automatic Updates and Maintenance: Cloud databases often handle updates and maintenance tasks automatically. End users don't need to worry about manually updating software or managing server maintenance, ensuring that the application remains up-to-date and available.

5. Collaboration and Sharing: Cloud databases facilitate easy collaboration and data sharing among users. Whether it's a collaborative project or sharing data with colleagues, the cloud infrastructure simplifies these processes, making it user-friendly for non-technical individuals.

6. Scalability without User Intervention: If an application grows, cloud databases can scale resources automatically. This scalability happens behind the scenes, without requiring end users to understand or manage the technical aspects of scaling.

In essence, cloud databases contribute to a user-friendly experience for individuals who may not possess technical knowledge, offering them reliable, secure, and accessible access to applications and data.

4- Course objective achieved:

This enhancement showcases two course outcomes:

1- ‘Develop a security mindset that anticipates adversarial exploits in software architecture and designs to expose potential vulnerabilities, mitigate design flaws, and ensure privacy and enhanced security of data and resources’

2- ‘Employ strategies for building collaborative environments that enable diverse audiences to support organizational decision making in the field of computer science’

Using a cloud database host aligns with the stated goals in the following ways:

1. Develop a security mindset:

- **Anticipating Adversarial Exploits:** Hosting a database on a cloud platform involves leveraging the security features provided by the cloud service provider. Cloud databases often come with built-in security measures, such as encryption, access controls, and monitoring, which can help anticipate and defend against adversarial exploits.

- **Mitigating Design Flaws:** Cloud providers invest heavily in ensuring the security of their infrastructure. By utilizing a cloud database host, one can benefit from the provider's security protocols, helping to mitigate potential design flaws that might arise in a self-hosted environment.

- **Ensuring Privacy and Enhanced Security:** Cloud databases are designed to prioritize data privacy and security. They offer features like data encryption in transit and at rest, compliance certifications, and regular security updates, contributing to enhanced security and privacy of data and resources.

2. Employ strategies for building collaborative environments:

- **Building Collaborative Environments:** Cloud database hosting facilitates collaboration by providing a centralized and accessible repository for data. Multiple users, regardless of their physical location, can access and work with the data concurrently. This collaborative environment supports organizational decision-making by ensuring that diverse audiences can contribute to and utilize data in real-time.

- **Enabling Diverse Audiences:** Cloud databases typically offer user management features that allow administrators to define roles and permissions, ensuring that diverse audiences can engage with the data according to their responsibilities. This flexibility contributes to building a collaborative environment where individuals with varying levels of technical expertise can participate in decision-making processes.

In summary, the technical skills required involve a combination of security-focused knowledge, cloud computing expertise, database management proficiency, and a strong foundation in secure software development practices. These skills collectively empower individuals to design and implement secure, collaborative, and efficient systems in the field of computer science.

5- Reflection:

Throughout this enhancement process, I tested my proficiency in database management and gained insights into DevOps practices. Reflecting on my professional journey, I haven't previously been involved in setting up databases on the cloud or working with cloud solutions in general. Typically, companies have dedicated teams for these functions, and as a result, I hadn't acquired skills in this area. Following this enhancement, I independently explored cloud hosting, made informed selections, exported databases, established admin user profiles, and integrated with a third-party GUI tool for data access. Although the complexity of this process doesn't match that of larger companies, even at a smaller scale, navigating through these tasks is immensely beneficial for me as a software architect aspiring to transition into management roles.