

Especificación de Casos de Uso

MythQL

Fecha: 12/08/2025

Tabla de contenido

Historial de Versiones	2
Información del Proyecto	3
Aprobaciones	3
Resumen Ejecutivo	3
Diagrama de Casos de Uso	4
Descripción de Actores	5
[Nombre de Actor 1]	6
Especificación de Casos de Uso	6
[Nombre de Caso de Uso Nro. 1]	7

Historial de Versiones

Fecha	Versión	Autor	Organización	Descripción
12/08/2025	0.1	Martin Alejandro Blacha, Gabriel Eduardo Denuble, Franco Carnevali, Benjamin Moyano	Anonymous Mages A.S	Versión 0.1
26/08/2025	0.2	Martin Alejandro Blacha, Gabriel Eduardo Denuble, Franco Carnevali, Benjamin Moyano	Anonymous Mages A.S	Información actualizada y agregada, que refleja la 2da y 3era Iteración.

Información del Proyecto

Empresa / Organización	Esc. Tec. Maria Sanchez de Thompson
Proyecto	MythQL
Fecha de preparación	12/08/2025
Cliente	Pablo Linares
Patrocinadores	NeuroDoktor, Good Airs
Gerente / Líder de Proyecto	Denuble Gabriel
Gerente / Líder de Desarrollo de Software	Denuble Gabriel

Aprobaciones

Nombre y Apellido	Cargo	Departamento u Organización	Fecha	Firma

Resumen Ejecutivo

El MythQL es un gestor de base de datos que tendrá su propia sintaxis pero basado en los modelos de base de datos relacionales. Es un proyecto educativo.

Se tomará como referencia al enunciado de la tarea final “Trabajo Grupal - Programación Sobre Redes - 2025.pdf” y se tomará como base la sintaxis que tiene el MySQL WORKBENCH, pero la forma de usar los comandos serán distintas, aparte de cambiar el nombre de los comandos de cada comando del MySQL se cambiará la forma de la gramática.

También se basará en aparte en Oracle, por que tiene que iniciar sesión, la persona que se crea la cuenta para usar el gestor de base de datos podrán crear otros usuarios internos en el gestor de base de datos.

Estos Usuarios creados de forma interna podrá iniciar sesion como si fuera la cuenta principal, tendrá privilegios delimitados o establecidos por la cuenta principal(usted)

Su sintaxis es la siguiente:

Select = **Bring**
Delete = **Burn**
Create = **Summon**
Insert = **File**
Alter = **Morph**
Update = **Rewrite**
Join = **Stitch**
Truncate = **SWEEP**
Count = **Enum**

Between = **Span**
Like = **Resemble**
Exists = **Echo**

Show = **Manifest**
Describe = **Depict**

Begin = **Start**

Procedure = **COMMAND**

Commit = **SEAL**

Rollback = **Undo**

Where = **CONDITION**

Group by = **Cluster**

Order by = **Align**

Limit = **Cap**

As = **AKA**

DESC = **DECLINE**

ASC = **INCREASE**

Auto increment = **SELF STACKABLE**

Grant = **Empower**

Revoke = **Disarm**

Role = **admin, scribe (editor), seer (lector), guard (auditor).**

La semántica sería así (Junto con MySQL para referencia):

```
SELECT (name, lastname, role)
FROM employee;
BRING employee{name, lastname, role};
```

```
SELECT (name,lastname, role)
FROM employee
WHERE lastname = Perez;
BRING employee{name, lastname, role} CONDITION lastname = Perez;
```

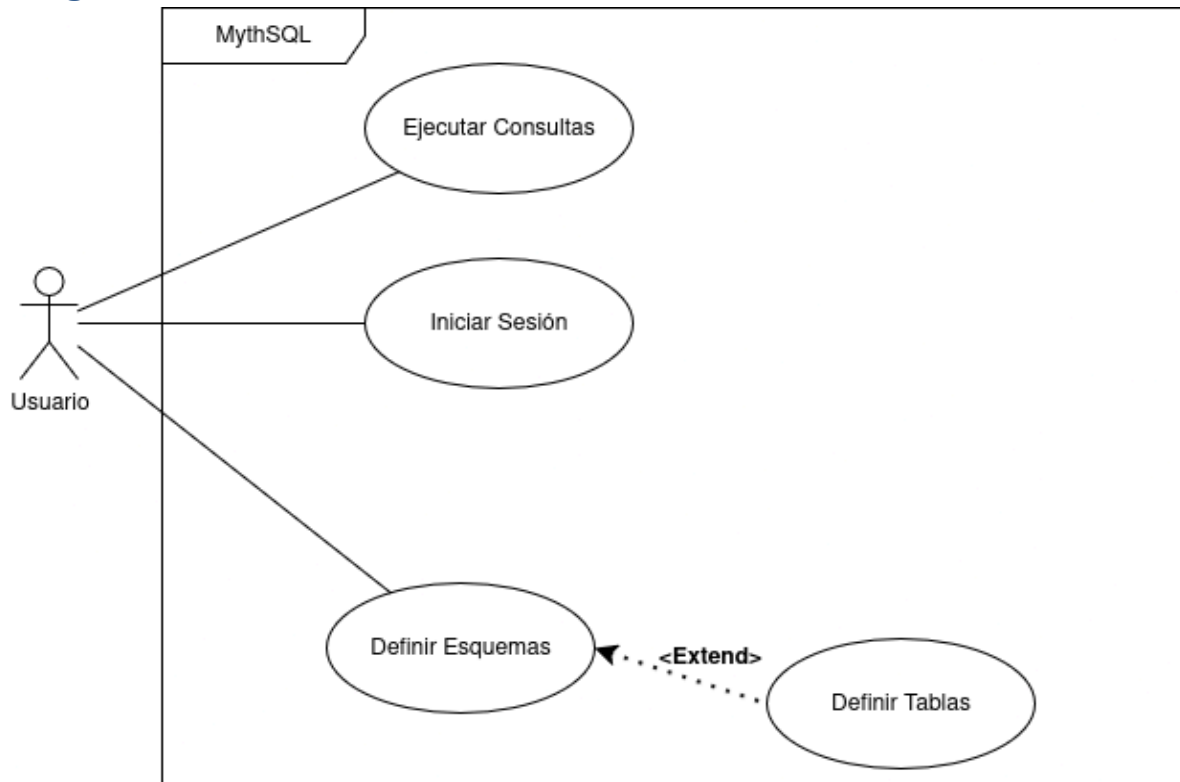
```
SELECT department, COUNT(*) AS total_employees
FROM employees
WHERE status = 'active' AND hire_date >= '2020-01-01'
GROUP BY department
ORDER BY total_employees DESC
LIMIT 10;
```

```
BRING employees{department, ENUM(*) AKA total_employees} CONDITION  
status = 'active' && hire_date >= '2020-01-01' CLUSTER department ALIGN  
total_employees DECLINE CAP 10;
```

```
INSERT INTO products (product_id, name, category, price, created_at)  
VALUES (101, 'Laptop', 'Electronics', 1200.00, CURRENT_TIMESTAMP);
```

```
FILE products{product_id, name, category, price, created_at}  
  [1, 'Laptop', 'Electronic', '1200.00', 12/08/2025],  
  [2],  
  [3];
```

Diagrama de Casos de Uso



Descripción de Actores

Usuario

Actor	Usuario	Identificador: id01
Descripción	Es ser humano que ingresara al sistema de gestor de base de datos y podrá gestionar y administrar sus bases de datos.	
Características		
Relación		
Referencias	RF-01,	

Atributos		
Nombre	Descripción	Tipo
Nombre		
Rol		

Contraseña		
Comentarios		

Especificación de Casos de Uso

Iniciar Sesión

Caso de Uso	Iniciar Sesión	Identificador: RF-01
Actores	Usuario	
Tipo	Primario	
Referencias		
Precondición	-	
Postcondición	El usuario queda logueado en el sistema.	
Descripción	El usuario entrega el nombre/Ip de la base de datos a la que quiere ingresar junto con su usuario y contraseña.	
Resumen	El usuario ingresa un identificador para la base de datos a la que quiere conectarse e ingresa la contraseña.	

Curso Normal

Nro.	Ejecutor	Paso o Actividad
1	Sistema	Muestra pantalla de login al usuario
2	Usuario	ingresa sus credenciales y base de datos e intenta iniciar sesión
3	Sistema	válida que no haya caracteres invalidos
4	Sistema	verifica que la base de datos exista
5	Sistema	verifica que la cuenta exista en la base de datos que se está intentando ingresar y las credenciales sean validas
6	Sistema	brinda acceso al usuario a la base de datos con el rol asignado para su usuario

Cursos Alternos

Nro.	Descripción de acciones alternas
3.1	Hay caracteres inválidos
3.2	Muestra mensaje de error
3.3	Vuelve al curso normal Paso 1

Nro.	Descripción de acciones alternas
4.1	La base de datos a la que quiso entrar no fue encontrada.
4.2	Muestra mensaje de error
4.3	Vuelve al curso normal paso 1

Nro.	Descripción de acciones alternas
5.1	La cuenta no fue encontrada en la base de datos
5.2	Muestra mensaje de error
5.3	Vuelve al curso normal paso 1