

# INF-239 Bases de Datos

## Tarea 1: Consultas SQL

Profesores: Ricardo Salas (ricardo.salas@usm.cl)

Mauricio Figueroa (mauricio.figueroac@usm.cl)

Ayudante Coordinador: Gonzalo Alarcón(gonzalo.alarconc@usm.cl)

Agosto, 2025

### 1. Descripción del problema

El equipo de Software de la empresa ZeroPressure ha pensado en desarrollar un portal el cual permita administrar las solicitudes de nuevas funcionalidades y gestión de errores (bugs). Para esto se le ha solicitado a usted como un gran estudiante de la USM construir las bases para lo que será este portal.

En términos generales, uno de los principales problemáticas a resolver, radica en que el equipo encabezado por el Jefe de TI, recibe en promedio 300 solicitudes de gestión de errores y 200 solicitudes de funcionalidades nuevas, además cada error o funcionalidad esta asociado a un tópico distinto, lo cual hace muy complejo manejarlos y solucionarlos, por eso esta responsabilidad cae en 3 ingenieros especializados en cada área.

Para un mejor conocimiento del sistema la empresa ha entregado la siguiente información:

#### **Una solicitud de funcionalidad se compone de lo siguiente:**

- Título (Obligatorio) (Mínimo 20 caracteres)
- Ambiente de Desarrollo (Web/Movil) (Opcional)
- Resumen (Máximo 150 caracteres) (Obligatorio)
- Tópico (Obligatorio) (Categoría bajo la cual se clasifican las solicitudes por ej: Backend, Seguridad, UX/UI)
- Criterios de Aceptación (mínimo 3) (Obligatorio)
- Solicitante (Obligatorio)
- Estado (Abierto, En Progreso, Resuelto, Cerrado)

#### **Una solicitud de gestión de error se compone de lo siguiente:**

- Título (Obligatorio)
- Descripción (Máximo 200 caracteres) (Obligatorio)
- Fecha de publicación (Formato DD-MM-YYYY) (Obligatorio)
- Tópico (Obligatorio) (Categoría bajo la cual se clasifican las solicitudes por ej: Backend, Seguridad, UX/UI)
- Autor de creación de la incidencia (Obligatorio)
- Estado (Abierto, En Progreso, Resuelto, Cerrado)

#### **Los Usuarios solicitantes poseen los siguientes campos:**

- RUT (Obligatorio) (Formato XXXXXXXX-X)

- Nombre (Obligatorio)
- E-mail (Obligatorio)

**Los Ingenieros poseen los siguientes campos:**

- RUT (Obligatorio)
- Nombre (Obligatorio)
- E-mail (Obligatorio)
- Especialidad (Obligatorio) (Categoría en la cual se especializa un Ingeniero ej : Backend, Seguridad, UX/UI) (Máximo 2 por Ingeniero)

**Otros:**

- Cada funcionalidad/error debe estar asignado a 3 ingenieros, la asignación se hace mediante la especialidad/tópico. Pueden realizar las operaciones necesarias en la BD para dejar de forma explícita la asignación de los ingenieros.
- No se permiten títulos duplicados del mismo tipo de solicitud.
- Un Ingeniero puede estar asignado a un máximo de 20 solicitudes.
- Un Usuario solo puede crear 25 solicitudes de error y 25 solicitudes de funcionalidad por día

## 2. To Do List

- Para esta entrega la empresa le solicita lo siguiente:
  - Desarrollar una Base de Datos la cual posea las tablas necesarias para almacenar la información mencionada previamente
  - Diseñar el Modelo Lógico Relacional con sus respectivas entidades, atributos, PK, FK.
  - Deberán crear las tablas que se presentan en el modelo conceptual para representar toda la información requerida. Es requisito utilizar ID, claves primarias, foráneas y tipos de datos óptimos (VARCHAR, INT, DATE, etc) en caso que corresponda.
  - Deberán desarrollar el Diccionario de Datos.
  - Poblar la base de datos con solicitudes de funcionalidades, solicitud de gestión de errores y usuarios min 600 datos (50 Autores, 50 Ingenieros, 300 Solicitudes de Gestión de Errores, 200 Solicitudes de Nuevas Funcionalidades). (para esto esta permitido el uso de scripts) (En caso de no realizar la población de la BD habrá un descuento de 30 puntos)

Por otro lado para verificar el correcto funcionamiento de su BD se les solicita realizar las siguientes consultas sql. Estas deberán hacerlas en SQL Server, PostgreSQL o Oracle:

1. Muestra los nombres de los ingenieros que tienen más de 5 solicitudes (de cualquier tipo) asignadas.
2. Identificar los 10 errores más antiguos que se han reportado. Muestra el título del error, su fecha de publicación y el nombre del autor.
3. Obtener una lista de todas las nuevas funcionalidades solicitadas para el ambiente "Móvil", mostrando el título, el nombre del tópico asociado y el nombre del usuario que la solicitó.
4. Mostrar los nombres de los tópicos que son los más problemáticos, definidos como aquellos tópicos con más de 10 reportes de error.
5. Encontrar todas las solicitudes de funcionalidad para las cuales el solicitante ha reportado al menos un error en el mismo tópico previamente.

6. Actualizar el estado de todas las funcionalidades que tengan más de 3 años a "Archivado" **Para la defensa traer datos necesarios para validar esta consulta**
7. Obtener una lista de todos los ingenieros que son especialistas en un tópico específico, por ejemplo, 'Seguridad'.
8. Obtener la cantidad total de solicitudes (errores y funcionalidades) creadas por cada usuario
9. Obtener la cantidad de Ingenieros que son especialistas en cada tema (el resultado debe estar separado por ",", como por ej: Backend: 5, Seguridad: 2, UI/UX: 4)
10. Elimina de la Base de Datos todas las solicitudes de gestión de error que tengan más de 5 años de antigüedad **Para la defensa traer la BD completa para verificar la eliminación**

### 3. Especificaciones y reglas

El desarrollo de esta tarea debe cumplir las siguientes especificaciones, de lo contrario podría existir un descuento en la nota final:

- **Debe** realizarse en parejas, no se aceptarán tareas individuales.
- **Se aplicarán descuentos por no cumplir las reglas, los cuales estarán detallados en la pauta.**
- En el foro de consultas podrán buscar pareja quienes no tengan, esto es exclusiva responsabilidad del estudiante.
- En caso de problemas con su pareja deberán contactar al profesor explicando su situación, (hacerlo con anticipación).
- La tarea debe ser entregada como un archivo .ZIP comprimido de la forma T1\_ROL1.ROL2, este debe contener los archivos necesarios para el funcionamiento de su tarea (CREATE e INSERT/Script, además del archivo con las consultas, y un archivo PDF el cual contenga el diccionario de datos y el modelo conceptual de la BD), además de un archivo README.txt el cual debe contener nombre, rol de los alumnos y las instrucciones para la correcta ejecución de su programa.
- Solo un alumno debe realizar la entrega.
- La entrega será vía AULA y el plazo máximo de entrega es hasta el **5 de Septiembre a las 23:59**.
- Las entregas posterior a las 00:00 tendrán un descuento de 10 puntos por cada hora o fracción.
- Las copias serán evaluadas con nota 0 y se informarán a las respectivas autoridades.
- En caso que falle la ejecución de algún comando, no se asignará puntaje a éste.
- **Las consultas se deberán hacer mediante el foro de Aula el cual se encontrara en la sección Tareas.**
- Esta evaluación debe ser defendida a través del servidor de Discord (el link se compartirá por aula los días previos a las defensas). Quienes no defiendan tendrán nota 0 en su tarea 1.
- Existe la posibilidad de que a su defensa, asista su profesor y realice preguntas.
- Cada grupo tendrá un horario definido para su defensa, en caso de atraso contarán con un tiempo menor para presentar su trabajo.
- La información respecto a la defensa será publicada previamente a estas en Aula, esto considera el detalle sobre los descuentos. Es su obligación estar atento a esta información y cumplir con lo establecido allí.
- Durante las defensas se les preguntará sobre el código entregado, debiendo responder correctamente para poder optar al puntaje completo. Se le pedirá que explique su razonamiento detrás de cada consulta SQL para verificar su comprensión de la tarea.
- El incumplimiento de cualquier punto expuesto aquí, podría implicar un descuento en la nota final de la tarea 1.