

Universidad San Carlos de Guatemala  
Facultad de ingeniería.  
Ingeniería en ciencias y sistemas



# Título del Proyecto:

## Proyecto - Fase 2

**PONDERACIÓN: 18**

**Horas Aproximadas: 20**

# Índice

<b>1. Resumen Ejecutivo</b>	<b>3</b>
<b>2. Competencia que desarrollaremos</b>	<b>3</b>
<b>3. Objetivos del Aprendizaje</b>	<b>3</b>
3.1 Objetivo General	3
3.2 Objetivos Específicos	3
<b>4. Enunciado del Proyecto</b>	<b>4</b>
4.1 Descripción del problema a resolver	4
4.2 Alcance del proyecto	4
4.2.1 Funcionalidades	4
4.4 Entregables	4
<b>6. Desarrollo de Habilidades Blandas</b>	<b>4</b>
6.1 Proyectos en Grupo	5
<b>7. Cronograma</b>	<b>5</b>
<b>8. Rúbrica de Calificación</b>	<b>5</b>
8.1 Requisitos para optar a la calificación	5
8.2 Valores	6

# 1. Resumen Ejecutivo

Este proyecto tiene como finalidad implementar una arquitectura de base de datos de alta disponibilidad y rendimiento, aplicando esquemas de replicación y una capa de caché para optimizar la velocidad y la resiliencia del sistema. Para ello, se utilizará una topología de replicación maestro-esclavo, se establecerá una política de copias de seguridad y se integrará Redis como sistema de caché en memoria.

Tomando como referencia la gestión de grandes volúmenes de datos de sistemas como la base de datos de IMDB, este proyecto aborda la necesidad crítica de las plataformas de alto tráfico de garantizar la continuidad del negocio y un rendimiento óptimo bajo altas cargas de trabajo. La falta de una estrategia de replicación de datos, de un plan de recuperación ante desastres y de un sistema de caché eficiente expone a las organizaciones a lentitud en el servicio, tiempos de inactividad y pérdidas de información valiosa de clientes y transacciones.

# 2. Competencia que desarrollaremos

Los estudiantes desarrollarán competencias en la implementación de esquemas de replicación en bases de datos bajo los modelos maestro-esclavo y maestro-maestro, en el diseño y ejecución de estrategias de respaldo y recuperación de información, así como en el uso de Redis como tecnología de almacenamiento en memoria para optimizar el rendimiento y la disponibilidad. Estas habilidades les permitirán fortalecer la resiliencia de los sistemas de información, garantizar la continuidad operativa y aplicar soluciones tecnológicas avanzadas en entornos empresariales críticos.

# 3. Objetivos del Aprendizaje

## 3.1 Objetivo General

Al finalizar el proyecto, los estudiantes serán capaces de implementar esquemas de replicación en bases de datos bajo los modelos maestro-esclavo y maestro-maestro, garantizando la continuidad y disponibilidad de la información. Asimismo, desarrollarán competencias en el uso y manejo de respaldos para la recuperación ante fallos y en la integración de Redis como sistema de almacenamiento en memoria orientado al alto rendimiento.

## 3.2 Objetivos Específicos

Al finalizar el proyecto, los estudiantes deberán ser capaces de:

1. **Diseñar y configurar un entorno de replicación en bases de datos:** utilizando esquemas maestro-esclavo o maestro-maestro, evaluando sus diferencias, ventajas y limitaciones en escenarios empresariales.
2. **Implementar mecanismos de respaldo y recuperación de información:** Que aseguren la continuidad del servicio y minimicen la pérdida de datos ante fallos o incidentes críticos.

3. **Integrar Redis como sistema de almacenamiento en memoria:** para optimizar consultas, mejorar la latencia de acceso a datos y aumentar el rendimiento de aplicaciones distribuidas.
4. **Simular escenarios de falla y recuperación:** a fin de comprobar la efectividad de la replicación y de los planes de respaldo implementados.
5. **Documentar la arquitectura, configuraciones y procesos utilizados:** En la replicación, backups y uso de Redis, garantizando buenas prácticas de gestión y mantenimiento de bases de datos empresariales.

## 4. Enunciado del Proyecto

### 4.1 Descripción del problema a resolver

Después de observar los resultados obtenidos en la implementación de la base de datos, la empresa IMDB ha solicitado el desarrollo de un sistema de alta disponibilidad, capaz de aplicar failback cuando sea necesario y acompañado de la gestión de respaldos automáticos. Además, se requiere incorporar un registro diario en Redis, el cual deberá ser posteriormente consolidado y escrito en la base de datos principal, garantizando así la continuidad operativa y la integridad de la información.

### 4.2 Alcance del proyecto

#### 4.2.1 Funcionalidades

- Sistema maestro-maestro o maestro-esclavo
  - Failover
  - Failback
- Backups automáticos
  - Completos
  - Diferenciales
  - Incrementales
- Implementación de redis para registro del momento en que se hacen los backups y la dirección de cada uno.
- API

### 4.4 Entregables

- Documentación completa
  - Manual de usuario
  - Manual técnico
  - Explicación de cómo realizaron el proyecto
  - Bitácora de trabajo
- Código fuente del funcionamiento de lo solicitado
- Archivos de configuración utilizados (docker, json, yml).

## 6. Desarrollo de Habilidades Blandas

### 6.1 Proyectos en Grupo

El trabajo en equipo es clave para el éxito de cualquier proyecto. Los estudiantes deben colaborar, asignando roles específicos (líder de proyecto, desarrollador, diseñador, etc.) para organizar y distribuir tareas de manera eficiente. A través de herramientas como Trello, Google Docs o GitHub, los equipos gestionarán sus avances y compartirán información de manera eficaz.

Además, se fomenta la toma de decisiones conjunta y la resolución de problemas en grupo, desarrollando habilidades como la comunicación, el liderazgo y la negociación. Los grupos recibirán retroalimentación constante para mejorar sus soluciones y adaptarse a los desafíos que surjan.

#### 6.1.1 Trabajo en Equipo

Cada grupo debe trabajar en colaboración, asignando roles específicos a cada miembro (líder de proyecto, desarrollador, diseñador, etc.). Los estudiantes deben coordinarse mediante herramientas de gestión como Trello, Google Docs o GitHub.

#### 6.1.2 Comunicación Efectiva

Los grupos deben realizar presentaciones periódicas ante sus compañeros y el profesor, compartiendo los avances y problemas encontrados, y recibiendo retroalimentación para mejorar sus soluciones.

#### 6.1.3 Resolución de Conflictos

A lo largo del proyecto, los estudiantes pueden enfrentar desafíos o desacuerdos. Deberán aplicar técnicas de resolución de conflictos para mantener la armonía y la productividad en el equipo.

## 7. Cronograma

Tipo	Fecha
Asignación de Proyecto	17 de septiembre
Entrega	8 de octubre
Calificación	11 de octubre

## 8. Rúbrica de Calificación

### 8.1 Requisitos para optar a la calificación

- Se debe hacer uso de un repositorio en la nube para realizar la entrega de su proyecto con el nombre de SDB2\_PF2\_<Grupo#>. Deberán agregar al auxiliar al repositorio:
  - feribarra-usac
- Para tener derecho a calificación, el estudiante deberá cumplir con los siguientes requerimientos mínimos:
  - Documentación completa
  - Sistema maestro-esclavo o maestro-maestro
  - Script de backups automáticos
- Se trabajará en grupos de 3 integrantes.
- Las copias completas/parciales serán merecedoras de una nota de 0 puntos, los responsables serán reportados al catedrático de la sección y a la Escuela de Ciencias y Sistemas

## 8.2 Valores

En el desarrollo del proyecto, se espera que cada estudiante demuestre honestidad académica y profesionalismo. Por lo tanto, se establecen los siguientes principios:

1. **Originalidad del Trabajo**
  - Cada estudiante o equipo debe desarrollar su propio código y/o documentación, aplicando los conocimientos adquiridos en el curso.
2. **Prohibición de Copias y Plagio**
  - Si se detecta la copia total o parcial del código, documentación o cualquier otro entregable, la calificación será de **0 puntos**.
  - Esto incluye la reproducción de código entre compañeros, la reutilización de proyectos de semestres anteriores o el uso de código externo sin la debida referencia.
3. **Uso Responsable de Recursos Externos**
  - El uso de bibliotecas, frameworks y ejemplos de código externos está permitido, siempre y cuando se referencian correctamente y se comprendan plenamente. ( Consultar con el catedrático su política)
4. **Revisión y Detección de Plagio**
  - Se podrán utilizar herramientas automatizadas y revisiones manuales para identificar similitudes en los proyectos.
  - En caso de sospecha, el estudiante deberá justificar su código y demostrar su desarrollo individual o en equipo. Si este extremo no es comprobable la calificación será de **0 puntos**.

Al detectarse estos aspectos se informará al catedrático del curso quien realizará las acciones que considere oportunas.