

**Instituto Tecnológico de Costa Rica (ITCR)**  
**Bases de datos I**  
**II Semestre 2025**  
**Profesora: María Auxiliadora Mora**  
**Proyecto #1**  
**Aplicación Web con persistencia en una base de datos relacional**

**Resumen:**

El proyecto consiste en desarrollar una aplicación web que utilice bases de datos como mecanismo de persistencia. La aplicación a desarrollar será definida por el grupo de estudiantes y presentada por medio de un ante-proyecto que será evaluado, realimentado y aprobado antes de su implementación.

El proyecto permitirá a los estudiantes poner en práctica metodologías de análisis y diseño de aplicaciones en un ejemplo asociado a un tema de su interés e implementar este ejemplo utilizando tecnología que forma parte de las tendencias actuales en bases de datos. La aplicación deberá persistir los datos en una base de datos relacional (ej. PostgreSQL, MariaDB, Oracle, MySQL, entre otras). Los únicos requisitos del motor de base de datos es que permita desarrollar procedimientos y funciones almacenadas y que permita la integración de Hibernate (una herramienta de mapeo objeto-relacional para la plataforma Java que facilita el mapeo de atributos entre una base de datos relacional tradicional y el modelo de objetos de una aplicación).

El objetivo del proyecto es diseñar e implementar una aplicación web que solviente requerimientos de un sistema de información de la vida real utilizando Java, Apache Tomcat, Hibernate y una BD relacional. Este objetivo está estrechamente relacionado con el objetivo específico del curso de utilizar y evaluar las tendencias actuales en el campo de la tecnología de base de datos.

**Es requisito fundamental que el sistema esté desarrollado usando Spring Boot (No se revisará el proyecto si no cumple este requisito).**

**Instrucciones:**

- **Forma de trabajo:** El proyecto se puede realizar de forma individual o en grupos de máximo dos personas.
- **Anteproyecto:** El sistema a desarrollar será propuesto por el grupo estudiante y **aprobado** antes de su implementación. Contenido del documento:
  - Introducción (descripción del problema y propuesta de solución)

- Descripción del sistema
- Objetivos y alcance del sistema
- Conceptualización del sistema:
  - Casos de uso a implementar (especificación completa de cada caso de uso que incluya: nombre del caso de uso, actores, precondiciones, post-condiciones, pasos de ejecución flujo básico y flujos alternos, resultado, entre otras secciones).
- Diseño del sistema
  - Describa la arquitectura del sistema propuesto.
  - Modelo de clases (al menos 3 clases que capturen datos de los objetos de la vida real).
  - A partir del modelo de clases y los casos de uso, genere el diagrama físico de la base de datos (al menos en quinta forma normal).
  - Diccionario de datos.
  - El proyecto debe incluir el uso de una función, un procedimiento almacenado y un trigger en la base de datos.
- **Tecnología** que se debe utilizar: Java por medio del framework Spring Boot, Apache Tomcat, una base de datos relacional y la herramienta Hibernate.
- **Interfaz:** El sistema debe contar con una **interfaz web** y estar organizado utilizando el patrón de arquitectura **modelo-vista-controlador** (MVC). La aplicación debe realizar operaciones CRUD (incluir, consultar, modificar y borrar) en la base de datos relacional por medio de Hibernate.
- En la **base de datos relacional**:
  - o El esquema de la base de datos debe tener un nombre representativo (ej. inventario, contabilidad, rec-humanos) es decir no se pueden almacenar los componentes (i.e. tablas, funciones, índices) de la base de datos en un esquema que tenga el nombre de un usuario. El esquema con nombre representativo debe contener todas las **tablas con las restricciones** que apliquen, ejemplo, llaves primarias, foráneas, restricciones de no nulidad,

secuencias automáticas para las llaves numéricas y otras restricciones. Los nombres de todos los objetos deben ser representativos.

- o El esquema debe incluir un **trigger**, un **procedimiento** y una **función**.
  - El trigger debe respaldar todos los cambios realizados en el contenido de las tablas de la base de datos. Es decir si se inserta, actualiza o borra un reuna herramienta de mapeo objeto-relacional para la plataforma Java que facilita el mapeo de atributos entre una base de datos relacional tradicional y el modelo de objetos de una aplicacióngistro en una relación eso debe quedar reflejado. Se debe guardar la operación (insert, update o delete) que se realiza, la fecha y hora, el usuario, el nombre de la relación (o tabla), el ID del registro, los datos anteriores del registro (concatenados en una hilera) y los datos nuevos (concatenados en una hilera).
  - Un procedimiento almacenado que tome un conjunto de registros no normalizados y los inserte en las tablas de su modelo. Se debe utilizar al menos un cursor dentro del procedimiento almacenado.
  - Una función propuesta por los estudiantes.
- **Documentación final:** Se debe entregar **documentación** del sistema en formato electrónico que incluya las secciones del anteproyecto actualizadas.
- Cada grupo de estudiantes deberá estar preparado para demostrar y explicar el trabajo realizado.
- El grupo de estudiantes que no presente el proyecto en la fecha definida en el TecDigital podrá realizarlo hasta una semana después con nota máxima de 60%.

### Rúbrica para evaluar anteproyecto (15 puntos)

Redacción y ortografía	2 puntos si el documento completo está bien redactado y se prestó especial cuidado a la ortografía.
Todas las secciones completas	2 puntos si el documento contiene todas las secciones requeridas.
Al menos 3 clases en el diagrama de clases	2 puntos si el modelo de clases del problema incluye al menos 3 clases, sus atributos y todas las relaciones.
Diagrama de clases	2 puntos si el modelo de clases captura los elementos necesarios, sus atributos, comportamiento y relaciones.
Modelo físico	3 puntos si el modelo físico está al menos en quinta forma normal.
Diccionario de datos	2 punto si el diccionario de datos está bien especificado, es decir contiene una descripción por tabla, una descripción por campo, el tipo de los campos, y las restricciones (ie. PK, FK, NN, entre otras)
Casos de uso	2 puntos si los casos de uso están completos y siguen el formato especificado.

### Rúbrica detallada para evaluar el proyecto completo

	Puntos
<b>1- Documentación</b>	
Anteproyecto (15 puntos).	15
Documentación final del sistema	3
<b>2- Desarrollo del sistema</b>	

¿La interfaz web es agradable y funciona bien?	4
¿Implementa MVC?	7
¿El sistema utiliza Hibernate para realizar la comunicación con la base de datos?	5
<b>Operaciones CRUD en la base de datos</b>	
¿Permite incluir registros en las tablas definidas en el diagrama físico de la base de datos?	4
¿Realiza reportes o consultas?	6
¿Permite borrar?	4
¿Permite modificar?	4
<b>3- Esquema en la base de datos relacional</b>	
Tiene nombre representativo del sistema	1
¿Tiene definidas PK?	1
¿Tiene definidas FK y restricciones de no nulidad?	1
¿Genera automáticamente las llaves primarias numéricas?	1
¿El esquema contiene todas las tablas del modelo?	1
<b>4-Funciones, procedimientos y triggers</b>	
¿El trigger funciona bien?	3
¿El procedimiento importa bien los datos?	5
¿La función retorna el resultado apropiado de acuerdo a lo propuesto?	3

<b>5- Base de datos no relacional (CRUD desde la interfaz)</b>	
¿Permite incluir?	3
¿Permite realizar consultas?	2
¿Permite borrar?	3
¿Permite modificar?	3