

# INF-239 Bases de Datos

## Tarea 2: PHP + MySQL/PostgreSQL

José Luis Martí Lara - Guido González Naranjo  
Ayudante de Tarea: Isidora Ubilla Zavala

27 de octubre de 2021

### 1. Objetivo

- **Primera parte:** Analizar y comprender un problema real, modelando correctamente su contexto en un modelo de datos consistente, en base a los métodos estudiados en clases.
- **Segunda parte:** Investigar y aplicar conceptos básicos de creación de aplicaciones web con bases de datos integradas, utilizando herramientas como PHP, MySQL y PostgreSQL.

### 2. Especificaciones y reglas

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

#### 2.1. Primera parte

- Deben conformar **equipos de 4 personas**. Cada uno de los equipos deberá registrarse en el anuncio disponible en la sección de Tareas **Inscripción equipos T2 parte 1**, disponible en Aula, **con fecha límite hasta las 23:55 del 8 de Noviembre**.
- Se debe entregar un informe vía Aula, **en formato PDF** y debe llevar el nombre: **MODELO-ROL1-ROL2-ROL3-ROL4.pdf**, en la sección Tareas, a más tardar el día **8 Noviembre, hasta las 23:55 horas**. Existirá un descuento de 10 puntos por la primera hora de atraso, luego se descuentan 20 puntos por día de atraso (incluyendo los 10 puntos iniciales).
- El informe será evaluado y a partir de este se realizará una **breve presentación en clases**, el profesor les dará feedback correspondiente a la presentación.
- El informe no tiene un formato en específico, aunque este debe incluir lo siguiente:
  1. Descripción breve del problema, incluyendo los supuestos que realizó el equipo de trabajo.
  2. Modelo de datos conceptual, expresado en un diagrama de clases UML, utilizando una herramienta CASE, como starUML o Draw.io.
  3. Modelo de datos relacional obtenido utilizando el proceso de 8 pasos vistos en clases. También expresado en un diagrama de clases UML.

4. Diccionario de datos de cada una de las tablas del modelo relacional, ordenadas de acuerdo al nombre de estas, usando el formato visto en clases.
5. Una regla de negocio dinámica que represente un futuro trigger de la aplicación web.

## 2.2. Segunda parte

- Debe realizarse en equipos de dos personas y sólo entre integrantes del mismo equipo que llevó a cabo la parte 1. Los equipos deberán registrarse en el anuncio disponible en la sección de Tareas **Inscripción equipos T2 parte 2**, disponible en Aula, **con fecha límite hasta el 8 de Noviembre, hasta las 23:55 horas**.
- Se debe desarrollar una aplicación web básica, teniendo libertad en elegir como implementar lo que se les solicita, utilizando PHP como herramienta de desarrollo.
- La parte 2 debe ser entregada el **6 de Diciembre hasta las 23:55 horas**, posterior a esto no se podrán realizar más entregas.
- Por medio de Aula se les informará el motor de bases de datos que cada equipo debe utilizar.

### 2.2.1. Sobre la entrega

La tarea debe ser entregada como un archivo comprimido de la forma T2-ROL1-ROL2.zip y debe contener los siguientes archivos:

1. Una carpeta **PHP**: Todos los archivos **.php** deben estar contenidos aquí. Puede crear los archivos que sean necesarios para resolver su tarea.
2. Una carpeta **BD**: Esta carpeta contiene la base de datos que se debe exportar en formato **.sql**, el nombre de este archivo debe ser **SansApp**.
3. Si necesita tener más archivos, por ejemplo, imágenes que adornen el sitio o archivos **.js** o **.css**, estos deben estar ordenados dentro de carpetas que contengan archivos del mismo estilo.
4. Se debe incluir un archivo **README**, puede ser **.txt** o **.md**, el cual debe contener los nombres, rol de los alumnos y las instrucciones para la correcta ejecución de su programa. Ante cualquier duda sobre algo que no aparezca especificado en la tarea **y no afecte las reglas de esta**, puede asumir lo que estime conveniente, pero debe ser especificado en el README, **en caso de asumir eventos y no especificar habrá descuento**.

## 2.3. Reglas

Las siguientes reglas corren para ambas partes de la tarea.

- Las copias serán evaluadas con nota 0 y se informará a las respectivas autoridades.
- Consultas vía la plataforma oficial AULA. Se creará un hilo en consultas donde se subirán las consultas del otro paralelo.
- La tarea fue subida con tiempo para que pudieran trabajar adecuadamente, y la entrega esta programada para finales de semestre, por lo que **es imposible negociar un cambio de fecha**.

### 3. Descripción del Problema

SansApp es una aplicación web que usted debe desarrollar para ayudar a la comunidad sana a comprar y vender de forma confiable, libre y segura. Para poder utilizar SansApp se debe tener una cuenta, es decir, la aplicación debe contar con un sistema de usuarios. Los usuarios deben registrarse con su nombre, correo, rol y fecha de nacimiento, información importante para lograr una compra-venta segura. Los usuarios podrán optar entre comprar o vender productos, por lo que debe existir la opción de ingresar un producto para ser vendido, además se debe poder navegar para escoger productos para comprar. En el caso de la venta se necesita tener un nombre para el producto, una descripción, una etiqueta de categoría, el precio, unidades disponibles a la venta. Para una mayor transparencia, SansApp permite a sus usuarios comentar el producto, además de dar una calificación de 1 a 5 estrellas. Otra funcionalidad importante es el carrito de compra, en este carrito de compra se agregan los productos, sin embargo, se debe ser tolerante a cambios, si el comprador se arrepiente sobre el producto en cuestión entonces debe poder quitar el producto del carrito, lo mismo con la cantidad de productos, se debe poder cambiar la cantidad de productos, en el caso de agregar cantidad debe ser posible siempre y cuando existan más unidades a la venta. Una vez realizada la compra, todos los usuarios tienen acceso a un historial de compra y venta, donde se puede ver toda la información histórica, como el producto, la fecha, el precio, cantidad de productos y nombre del comprador o vendedor dependiendo de quien se trate.

Para ayudarles a organizar todo el trabajo por hacer, se le deja la siguiente To-Do list :

#### ■ General

- ☐ Se debe hacer uso de mínimo dos **triggers**.
- ☐ Se debe hacer uso de mínimo dos **vistas**.
- ☐ El sistema debe tener navegación entre secciones (recargar la página o retroceder con el navegador no es válido).
- ☐ Implementar sistema de usuarios.
- ☐ El sistema debe mostrar el nombre de usuario activo en todo momento.

#### ■ Página principal

- ☐ Se debe implementar una barra de búsqueda, esta permite buscar productos, categorías y a otros usuarios.
- ☐ Se debe mostrar el Top de los 5 productos con mejor promedio en calificaciones con estrellas.
- ☐ Se debe mostrar el Top de los 5 productos más vendidos.
- ☐ Se debe mostrar el Top de los 5 usuarios con más ventas en SansApp.

#### ■ Perfil

- ☐ Se puede editar datos del perfil y eliminar la cuenta si se desea.
- ☐ Se debe mostrar la información básica del perfil, como la cantidad de productos vendidos, cantidad de productos comprados, el nombre de usuario y el correo.
- ☐ Se debe poder ver el historial de compras y ventas.

#### ■ Carrito de compra

- ☐ Se puede editar los productos agregados al carrito de compra.

- ☐ Se debe mostrar la información de todos los productos en el carrito, como la cantidad de productos, el precio de cada uno, el nombre, etc.
- ☐ Debe tener un botón de compra que simule la transacción, de tal forma que se logre ver la compra en el historial.

#### ■ Productos en venta

- ☐ CRUD para productos en venta.
- ☐ CRUD para los comentarios de los productos en venta.
- ☐ Se debe mostrar toda la información relacionada al producto mencionada en el enunciado, además de cantidad total vendida del producto y total de unidades disponible para la venta.
- ☐ Se debe mostrar la cantidad de comentarios, promedio de calificación del producto y nombre del vendedor del producto como enlace al perfil de este.

## 4. Historias de usuario HDU

Las HDU se definen como *"la descripción de una funcionalidad que debe incorporar un sistema de software, y cuya implementación aporta valor al cliente/usuario"*. Para simplificar su uso dentro del contexto de la asignatura, sólo se trabajará con la estructura más básica de éstas: **descripción y criterios de aceptación**.

Es importante que tenga esto en consideración para que tengan una idea clara de como debería funcionar la aplicación y como puede ser revisada la tarea para su evaluación.

A continuación, se lista una parte de las historias de usuario del sistema a desarrollar (*las restantes deben inferirse del enunciado*):

### 4.1. Usuario no registrados

Como persona que no se ha registrado en la aplicación, yo quiero poder registrarme en el sistema, para poder acceder a mi perfil.

#### 4.1.1. Criterios de aceptación

1. Dado un usuario que no ha sido registrado en el sistema, cuando intente buscar un producto con la barra, entonces se le muestran los resultados de búsqueda y en consecuencia, puede acceder a ver el producto.
2. Dado un usuario que no ha sido registrado en el sistema, cuando haga click en el botón de registrarse, entonces se le muestra un módulo para que se registre o inicie sesión.
3. Dado un usuario que no ha sido registrado en el sistema, cuando intente dar una calificación al producto y/o comentar el producto, entonces se le redirige para que se registre o inicie sesión.

### 4.2. Barra de búsqueda

Como usuario, yo quiero poder usar la barra de búsqueda, para encontrar productos, usuarios y categorías.

#### 4.2.1. Criterios de aceptación

1. Dada una persona logueada en el sistema, cuando intente encontrar un producto o usuarios con la barra de búsqueda, entonces se le muestran todos los productos o usuarios con el mismo nombre o nombres similares al buscado. *Pista:* SELECT LIKE.
2. Dada una persona logueada, cuando quiera buscar un producto por categoría, entonces se le muestra todos los productos con dicha categorización.
3. Dada una persona logueada, cuando intente buscar algo que no exista en la base de datos, entonces el sistema muestra un mensaje de que no se encuentra lo buscado.
4. Dada una persona logueada, cuando busque a otros usuarios, entonces se le muestra un recuadro con la información básica del usuario buscado (si es que se realiza una búsqueda exitosa).

### 4.3. Gestión de Ventas

Como persona que está logueada en el sistema, yo quiero poder subir productos, para ser vendidos a otros usuarios.

#### 4.3.1. Criterios de aceptación

1. Dada una persona que quiera subir un producto, cuando presione el botón que le permita subir la venta, entonces se le muestra el formulario para registrar el nombre del producto, la descripción, la categoría, cantidad y la información mencionada en el enunciado.
2. Dado un usuario logueado, cuando quiera editar información del producto en venta, entonces el sistema muestra la opción de edición y la nueva información se updatea.
3. Dado un creador de contenido logueado, cuando quiera retirar un producto en venta, entonces el sistema le permite retirar el producto en venta.

### 4.4. Gestión del perfil

Como persona que está registrada en el sistema, yo quiero gestionar mi perfil, para actualizar información o editar contenido.

#### 4.4.1. Criterios de aceptación

1. Dada una persona logueada, cuando quiera editar su nombre de usuario o tipo de usuario, entonces el sistema le permite cambiar los datos.
2. Dada una persona logueada, cuando quiera eliminar su cuenta, entonces se borra todo registro asociada a esta.
3. Dada una persona logueada, cuando quiera revisar el historial de compra-venta, entonces se muestra una tabla con toda la información histórica de compras y ventas.

## 5. Aclaraciones

1. La tarea busca evaluar la interacción de la base de datos, esto quiere decir que puede ser una página en base a texto plano y html básico.
2. Puede hacer uso de Bootstrap o JS, sin embargo, no afecta su nota final.

3. Implementar **READ** significa que se debe poder ver las entidades mencionadas.
4. Toda persona que quiera dar una calificación, comentar o comprar debe iniciar sesión, solo se puede buscar productos sin necesidad de estar registrado en el sistema.
5. Una persona puede hacer cuantos comentarios quiera a un producto.
6. No se puede dar calificación a un producto más de una vez, si desea cambiar la cantidad de estrellas otorgadas, entonces se debe realizar un update en la base.
7. Los usuarios pueden cancelar un carrito de compras eliminando todos los productos.
8. Un producto puede tener más de una categoría.