

## Proyecto de Desarrollo / Proyecto de Software - Entrega 2 -

La finalidad de esta entrega es extender el trabajo de la entrega anterior con parte de la funcionalidad del sistema desarrollado.

### Objetivo del Trabajo

Se desea realizar un sistema de gestión para un banco de alimentos. Se van a manejar datos de entidades receptoras, donantes y alimentos, y se controlará el manejo de pedidos y entregas de alimentos a las entidades.

### Destinatarios

Estará destinado al personal de la asociación, los cuales tendrán diferentes roles (administración, gestión, consultas).

### Especificación

El sistema debe contar con ciertos módulos que permitan dicha gestión.

En esta **primera etapa**, será necesaria la implementación de los siguientes módulos:

- ❖ Módulo de Entidades Receptoras
- ❖ Módulo de Donantes
- ❖ Módulo de Alimentos

### Módulo de Entidades Receptoras

Este módulo permitirá realizar las Altas, Bajas y Modificaciones (en adelante ABMs) de Entidades Receptoras. La información que se debe registrar es:

Razón Social, Teléfono, Domicilio, Servicio Prestado, Necesidad (máxima, mediana, mínima), Estado (alta, en trámite, suspendida, baja).

Los Servicios Prestados se refieren a los distintos servicios que brinda la entidad, como por ejemplo si es un Hogar de Día, un Comedor Infantil, un Hogar de Adolescentes, un Jardín Maternal, etc.

## **Módulo de Donantes**

Este módulo permitirá realizar los ABMs de Donantes. La información que se debe registrar es: Razón Social e información de contacto (Apellido, Nombre, Domicilio, Teléfono y Mail).

## **Módulo de Alimentos**

Este módulo permitirá realizar los ABMs de Alimentos y consultar su stock. La información que se debe registrar es:

Código, Descripción, Fecha de Vencimiento, Contenido, Peso del Paquete, Stock, Reservado, Donante.

Donde el Contenido se refiere al detalle del producto en base a la cantidad y el peso o volumen, por ejemplo si se trata de aceite podría ser “4 x 5 lt.” y si se trata de azúcar sería “10 x 1 kg.” Tanto el campo Stock como Reservado se tratan de las cantidades del producto en stock (paquetes, no unidades) y reservada por alguna entidad respectivamente. Por ejemplo, si dice 20, indicará que hay 20 paquetes de “10 x 1Kg”. En el caso de que un mismo producto se reciba en paquetes de distintas unidades (fideos 10 x 500gr y fideos 20 x 500gr) se consideran dos alimentos distintos.

Además se deberá almacenar la cantidad que fue dada por cada donante.

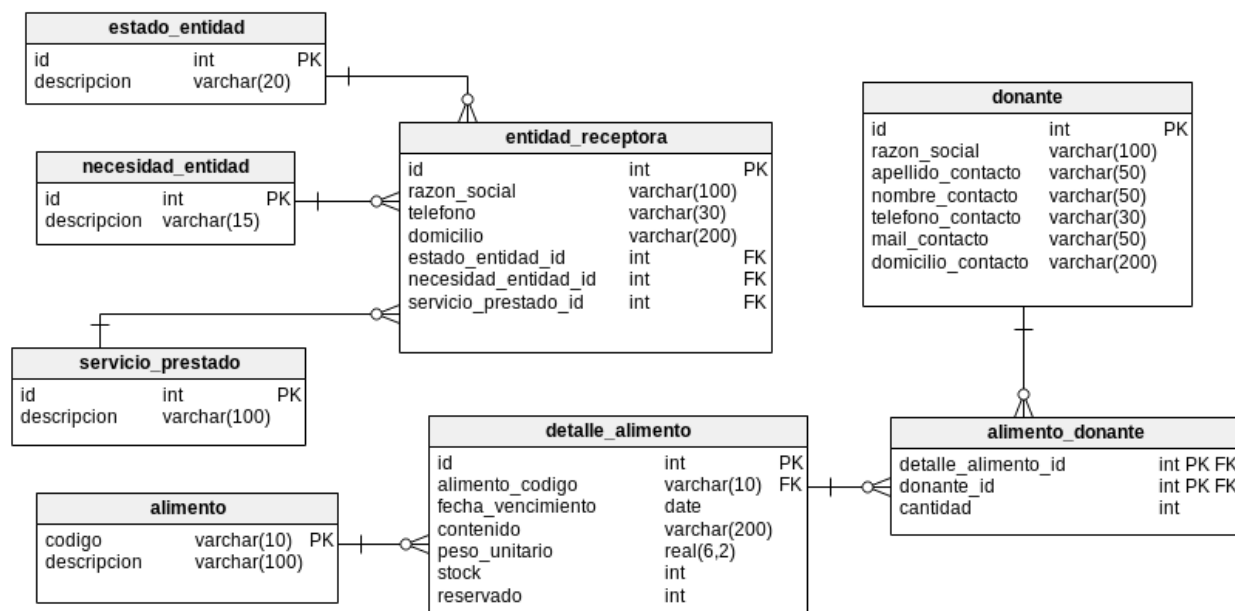
Para el stock de alimentos se mostrará un listado aparte con aquellos productos que se encuentren en stock junto con la cantidad reservada de cada uno (si es que lo están).

## **Alcance**

Como alcance de esta segunda entrega, se deberá disponer del **Frontend** y **Backend** realizados en la primer entrega. El frontend se refiere a la parte pública, en donde se mostrarán listados de entidades receptoras y donantes, y el backend es la parte de administración, la cual va a requerir que el **login** se encuentre implementado, y dependiendo del tipo de usuario, serán las operaciones que estarán disponibles. El Administrador podrá realizar todas las funciones de los módulos del sistema, el de Gestión sólo podrá confeccionar pedidos y entregas (en esta entrega no se debe implementar) y el de Consulta sólo podrá listar los alimentos en stock (dichos usuarios ya se encontrarían previamente en la base de datos).

## Sugerencia

Para simplificar el desarrollo del trabajo, la cátedra sugiere la utilización del siguiente modelo de base de datos:



El diseño del manejo de los usuarios y los roles queda a criterio del alumno.

## Consideraciones generales

- Debe ser desarrollado utilizando PHP, HTML5, CSS3 y MySQL, y de ser posible respetando el modelo en capas MVC. Debe utilizar PDO como mecanismo de abstracción de bases de datos. Debe tener en cuenta los conceptos de Semántica Web proporcionada por HTML5 siempre y cuando sea posible con una correcta utilización de las etiquetas del lenguaje.
- El trabajo será evaluado desde el servidor de la cátedra que cada grupo deberá gestionar mediante Git. **NO se aceptarán entregas que no estén realizadas en tiempo y forma en el servidor provisto por la cátedra.**
- El ayudante a cargo evaluará el progreso y la participación de cada integrante mediante las consultas presenciales y el seguimiento mediante GitLab.
- Toda vista (**HTML5** y **CSS3**) debe validar contra las especificaciones corrientes de la W3C (<http://validador.w3.org/>). **No se puede** utilizar ningún framework o herramienta de diseño para la vista, como por ejemplo Bootstrap. Se puede utilizar Twig como motor de

plantillas, aunque en esta entrega no se evaluará su uso.

- Tener en cuenta que el servidor de la cátedra utiliza los siguientes servicios y versiones:
  - Servidor Web: nginx 1.6.1-3
  - Servidor BD: mariadb 10.0.13-1
  - Intérprete Php: php 5.5.16-2

**Fecha límite de Entrega: Lunes 29 de septiembre a las 8:00 (cierre del servidor)**