



Trabajo Final PySW (2020)

Objetivo: Aplicar todas las tecnologías vistas en la materia: HTML, CSS, BOOTSTRAP, ANGULAR, TYPESCRIPT, EXPRESS, etc. para el desarrollo de un sistema en base al siguiente enunciado.

Se espera y es requisito que el proyecto aplique todos los componentes vistos en clase: datatable, validaciones (personalizadas), filtros, diferentes templates de Bootstrap: menues responsivos, modals, tarjetas.

Enunciado: El portal de una obra social lleva un registro de todos sus afiliados. La obra social ofrece servicios variados a los afiliados, los servicios pueden ser "Taller de Guitarra básico", "Taller de Computación inicial", etc. Los afiliados pueden inscribirse gratuitamente a los servicios que ofrece la obra social. Se lleva un registro de los pagos mensuales de las cuotas sociales los afiliados.

A todos los afiliados se les crea una cuenta de usuario de tipo "socio" para que hagan uso de un formulario para ponerse formalmente en contacto con la obra social dejando novedades.

El portal tiene páginas que son públicas para lo cual no es necesario loguearse para acceder a ellas como ser: inicio (aquí se publican las noticias), contáctenos, nosotros, etc. Las noticias son redactadas por los usuarios de tipo "administrativo".

Preparar el entorno para que al Sistema acceden al menos 5 usuarios socios, 3 administrativos y 2 dueños (perfiles).

Modelo: A continuación, se muestran las clases para el ejercicio planteado. El diseño de las clases/entidades puede tener leves modificaciones como ser nuevos atributos o cambio de tipos de datos, Se debe respetar las relaciones de composición/agregación indicadas o que se infieren del enunciado.

<u>Afiliado:</u> _id: string apellido: string nombres: string dni: number email: string imagen: string teléfono: number	<u>Servicios:</u> _id: string nombre: string imagen: string activo: boolean afiliadosInsc: Array<Afiliado>	<u>Pagos:</u> _id: string afiliado: Afiliado fecha: fecha monto: number año: number mes: number/string
<u>Usuario:</u> _id: string usuario: string password: string activo: boolean perfil: string (administrativo, propietario)	<u>Novedades:</u> _id: string Usuario: Usuario Texto: string estado: string (pendiente-procesado)	<u>Noticias:</u> _id: string titulo: string descripcion: string fecha: date usuario: Usuario vigente: boolean

Aclaración de algunos campos:

Usuario->usuario: el administrador setea este campo con email de un propietario.
Usuario->activo: el valor es true-false e indica si el usuario está bloqueado o no.
Novedades->estado: cuando un usuario "administrativo" lee la novedad la pasa a procesado.
Noticia->vigente: si es true esa noticia saldrá en el portal de inicio.
Servicio->activo: si un servicio tiene este campo en false no debería salir como servicio ofrecido.



Puntos mínimos para regularizar la materia:

- Gestión de Cuentas de usuarios que accederán al sitio, los perfiles de usuario disponibles son (socio, administrativo, administrador, gerente), el menú será adaptable al tipo de usuario que ingresa, en el caso de los Administradores pueden acceder a todas las posibilidades del mismo. Controlar también el acceso por rutas.
- El usuario de tipo "administrador" estará habilitado para la gestión (crud) de usuarios.
 - o El usuario de tipo "administrativo" es el encargado de realizar la gestión (crud) de afiliados, servicios, pagos.
 - o El usuario de tipo "socio" que se asigna a los afiliados es el encargado de hacer uso del formulario de novedades.
- La página de inicio que es de carácter público (diferente a las páginas de gestión) mostrará al público en forma agradable (no utilizar datatable, sí por ej. cards) las noticias que postean los empleados "administrativos" de la obra social.

Puntos para poder acceder a la promoción:

- Aplicar una librería de npm en angular que permita imprimir en pdf los reportes de las entidades gestionadas por los administrativos.
- Aplicar Notificación en redes sociales para que las noticias que se redacten puedan ser publicadas en una página de la Red Social.

Forma de Evaluación:

- se evalúa al alumno para determinar el grado de participación en los proyectos basándose en la participación de su proyecto en github.
- También se evalúa a los alumnos particular y grupalmente en la defensa final del mismo (por meet).
- Grupos de 4 o 3 integrantes, no se permiten integrantes que estén fuera de los requerimientos de la materia (ver planilla en drive para seguimiento de situación particular).

Fecha Limite de defensa final: Martes 07 de Julio/2020. El grupo puede hacer su presentación con anticipación en clases.