

**Universidad Tecnológica Nacional**

**Facultad Regional Rosario**

**Tecnicatura Universitaria de la Programación**



**Programación III**

**Trabajo Práctico Integrador**

**Enunciado**

## Abstract

El **trabajo práctico integrador** (de ahora en más, TPI) es un trabajo grupal de un mínimo de 3 integrantes y un máximo de 4, donde los alumnos tendrán como objetivo principal la creación de una aplicación web basándose en los contenidos dados a través del cuatrimestre por la cátedra de Programación III, de manera que los mismos logren reforzar conceptos y los pongan en práctica.

## Objetivos del TPI

- Demostrar y aplicar los conceptos acordes a lo aprendido sobre la librería React y la librería Node con Express server, mediante la implementación de ellos en un caso de uso concreto referido a un negocio real o ficticio propuesto por el grupo.
- Defender el TPI en un coloquio de modalidad presencial y/o virtual frente al cuerpo docente. Aprobar el examen oral individual correspondiente posterior a dicho coloquio.

## Requerimientos necesarios para la aprobación del TPI

1. El sistema debe poseer **como mínimo 3 ABM**(Alta, baja y modificación), una por cada entidad. Cada entidad, a su vez, debe tener su tabla correspondiente en base de datos (por ejemplo: **Usuario - Producto - Orden de compra**)
2. Server / backend en Node / express para la gestión de los datos.
3. No se permite la utilización de librerías externas que resuelvan funcionalidades por fuera de las dadas en la cátedra (consultar al personal docente la lista permitida al momento). Como por ejemplo:
  - a. redux (o algún otro manejador de estados),
  - b. react-hook-form (o similares para el manejo y validación de formulario)Si está permitido la utilización de librerías de estilizado externas.
4. Poseer funcionalidades de autenticación de usuarios, incluyendo manejo de permisos de los mismos (**como mínimo 3** roles de usuarios distintos: super-admin, admin y usuario común).
5. Validación en todos los formularios con su correspondiente aviso al usuario en caso de error.
6. Se tomará en cuenta un **mínimo criterio de calidad** en cuanto al diseño y la experiencia de usuario. A tener en cuenta:
  - a. Estilos.
  - b. Ortografía.
  - c. Acompañamiento al usuario.

7. Enrutado dinámico establecido mediante la librería “react-router” para aprovechamiento e implementación de conceptos de SPA.
8. Proteger las rutas que requieran acceso de usuarios específicos.
9. Utilizar Context en al menos un caso de uso.
10. Desarrollar un custom hook para al menos un caso de uso.
11. Cada integrante del grupo deberá al menos haber aportado un 25% del código al proyecto del TPI.

## Enunciado de ejemplo

El local de comida rápida “TanBurger” nos ha solicitado armar un sistema web para gestionar su negocio. Ellos desean que la web nos permita

- Para un usuario cliente
  - Ver la carta de productos. Confeccionar un carrito en el navegador y luego efectuar / crear un pedido. El usuario a su vez puede ver todos los pedidos realizados en un historial de pedidos.
- Para un usuario admin (el dueño de la hamburguesería)
  - Ver la carta de productos.
  - Ver los pedidos de los clientes. Poder filtrar los pedidos por fecha y por cliente.
  - Poder hacer Alta, baja y modificación de los productos de la carta.
- Para un usuario sysadmin o super admin (el equipo nuestro de desarrollo):
  - Todas las funcionalidades de los otros usuarios deben estar disponibles
  - Poder hacer Alta, baja y modificación de usuarios (incluyendo cambio de roles a los usuarios si así se lo desea)

En el ejemplo anterior vemos cómo cumplimos los requisitos principales del TPI, teniendo como mínimo 3 roles de usuario (sysadmin, admin y cliente) y 3 entidades (usuario, producto, pedido).

<b>Fecha</b>	<b>Versionado actual</b>	<b>Autor</b>	<b>Observaciones</b>
27/04/2022	1.0.0	Gabriel Golzman / Pablo Botta	Primera versión
04/08/2022	2.0.0	Gabriel Golzman / Pablo Botta	Diferenciamos condiciones para aprobar y regularizar
02/04/2025	3.0.0	Cátedra	Agregado requisitos nuevos