

<div style="display: flex; justify-content: space-between; align-items: center;"> <div style="width: 10%; background-color: #cccccc; height: 30px;"></div> <div style="text-align: center;"> <b>Universidad Tecnológica Nacional</b>  <b>Facultad Regional Avellaneda</b> </div> </div>											
Técnico Superior en Programación - Técnico Superior en Sistemas Informáticos											
<b>Materia: PROGRAMACIÓN III</b>											
<b>Apellido:</b>						<b>Fecha:</b>					
<b>Nombre:</b>						<b>Docente<sup>(2)</sup>:</b>					
<b>División:</b>	<b>3C</b>					<b>Nota<sup>(2)</sup>:</b>					
<b>Legajo:</b>						<b>Firma<sup>(2)</sup>:</b>					
<b>Instancia<sup>(1)</sup>:</b>	<b>PP</b>		<b>RPP</b>		<b>SP</b>	<b>X</b>	<b>RSP</b>		<b>FIN</b>		

**A partir del punto 3 todas las rutas deben estar autenticadas a través de un token.**  
**Se deben usar todas las columnas de las tablas y no se puede agregar ninguna columna ni cambiar el tipo.**

1. **(POST) usuario.** Recibe email, nombre, clave, tipo (alumno, profesor, admin) y legajo (entre 1000 y 2000) y guarda el usuario en la base de datos. Validar que el email y el legajo no sean únicos.
2. **(POST) login:** Recibe email y clave y si son correctos devuelve un JWT, de lo contrario informar lo sucedido.
3. **(POST) materias:** (Solo para admin). Recibe materia, cuatrimestre, vacantes y profesor y los guarda en la tabla materias.
4. **(GET) materias/{id}:** Para alumno muestra los datos de la materia, para profesor y admin muestra los datos de la materia y la lista de inscriptos (email, nombre y legajo).
5. **(PUT) materias/{id}/{profesor}:** (Solo admin) Asigna un profesor a una materia.
6. **(PUT) materias/{id}:** Solo alumno. Se inscribe a una materia si hay vacantes. Restar una vacante si la inscripción es exitosa.
7. **(GET) materias:** Muestra nombre de materia, vacantes, cuatrimestre, nombre y email del profesor y cantidad de inscriptos.
8. **(GET) materias:** Desarrollar un middleware que ponga en mayúsculas la tipografía de las materias sin vacantes.