

Materia: PROGRAMACIÓN III

Apellido:		Fecha:	
Nombre:		Docente⁽²⁾:	Rampi
División:	3C	Nota⁽²⁾:	
Legajo:		Firma⁽²⁾:	
Instancia⁽¹⁾:	<div style="display: flex; justify-content: space-around;"> <div>PP</div> <div></div> <div>RPP</div> <div></div> <div>SP</div> <div>x</div> <div>RSP</div> <div></div> <div>FIN</div> </div>		

Desaprueban el parcial los que no respeten los siguientes requerimientos:

- a- La aplicación se deberá realizar en un skeleton de slim framework.
- b- Todas las en clases de PHP , correspondiente a una entidad.
- c- Las consultas a la BD se deberán hacer con un ORM.
- d- No debe haber código obsoleto o que no corresponda a las entidades del parcial.
- e- A partir del punto 4 todas las rutas deben estar autenticadas.
- f- Se deben respetar las rutas y parámetros indicados.

Se consideran usuarios administradores a aquellos cuyo legajo sea menor a 100.

- 1-(1pt.) Se deben guardar los siguientes datos de cada petición a la API: hora, IP, ruta, metodo, usuario(si es posible).
 - 2-(3 pts.) **users** (POST): se recibe email, clave, legajo(1 entre 1 y 1000) y dos imágenes (guardarlas en la carpeta images/users y cambiarles el nombre para que sea único). No se podrá guardar la clave en texto plano. Todos los datos deberán ser validados antes de guardarlos en la BD.
 - 3- (2 pts.) **login**: (POST): Recibe clave, email y legajo, si estos datos existen en la BD, retornar un JWT, de lo contrario informar lo ocurrido. La consulta debe ser *case insensitive*.
- A PARTIR DE AQUÍ TODAS LAS RUTAS DEBEN ESTAR AUTENTICADAS.
- 4- (2 pt.) **ingreso** (PUT): Se debera fichar el ingreso del usuario guardando la hora y el usuario. Si el usuario ya había ingresado con anterioridad informar la situación.
 - 5-(2 pts.) **egreso** (PUT): Se guardaran los datos de egreso del usuario. Si no estaba ingresado informar la situación.