# HACKDAY

## Instrucciones

* Registrarse en Discord App mediante el enlace: <https://discord.gg/4GAkVuG> ahí pueden realizar sus dudas y consultas, solo se responderá en el canal #examen.
* Al inicio del reto es necesario que indiquen los repositorios de su código fuente en GitHub para que podamos ir monitoreando el avance; deben enviar sus repositorios mediante mensaje directo a cualquiera de los moderadores del chat.
* Los commits que realicen tienen que ser identificados con su nombre y correo electrónico.
* Solo se tomará en cuenta los commits que se realicen durante el horario del desarrollo del reto en los repositorios indicados, **verificar que se subieron todos sus cambios a sus repositorios antes de finalizar el reto**.
* En un archivo README.md detallar la lista de funcionalidades solucionadas y los pasos para la instalación tanto para el backend y frontend.

## Herramientas

Todo tiene que ser desarrollado con herramientas, librerías, frameworks open source:

* NodeJS
* ExpressJS, SailsJS, Restify
* AngularJS, Angular, VueJS, React
* Base de Datos: PostgreSQL, MongoDB, MariaDB

## Reto.- Creación de horarios (100 pts.)

**Backend**

* (5) Registro y listado de docentes
* (5) Registro y listado de aulas
* (10) Registro y listado de materias (una materia siempre es impartida por el mismo docente)
* (15) Registro y listado de horarios (un horario pertenece a una materia y se pasa en un aula)
* (10) Validación para evitar cruce de horarios por aula (un horario no puede ser ocupado por dos materias a la vez)
* (5) Opción para poder registrar varios horarios de una materia al mismo tiempo

**Frontend**

* (5) Registro y listado de docentes
* (5) Registro y listado de aulas
* (10) Registro y listado de materias
* (15) Registro y listado de horarios
* (10) Lista semanal de horarios por docente
* (5) Opción para poder registrar varios horarios de una materia al mismo tiempo

## Actividades opcionales (25 pts.)

* (5) Documentación del api-rest con apidoc
* (5) Validación de datos en el backend
* (5) Validación de datos y despliegue de errores por cada campo en el frontend
* (5) Diseño Responsive del frontend
* (5) Uso de un pre o post procesador