

EXAMEN DEL PARCIAL I
Carrera: Tecnologías de la información

Asignatura: Aplicaciones Web II
Fecha: Noviembre/2021

Docente: John Cevallos
Nivel: Sexto Nivel "A"

Estudiante: NESTOR ALEX Y ANCHUNDIA ANCHUNDIA **Cédula:** 1314892470 **Firma:**

CALIFICACIÓN EXAMEN: 10 PUNTOS - TIEMPO PARA RESOLVER: 120 MIN

Micro Proyecto de Integración aplicando Node y Express. (10 puntos)

Considere lo siguiente para desarrollar su ejercicio:

- El código debe ser subido en algún repositorio GIT,
- Publicar un video explicativo de los proyectos ejecutados y funcionando (entre 2 y 4 minutos de duración), mostrando su funcionalidad, y explicando generalidades en el código del mismo.
- Definir los siguientes proyectos: (El número de proyectos puede variar según su análisis)
 - Proyecto que publica sitio web estático.
 - Proyecto Cron Job.
 - Proyecto que publica servicio REST.

Tema:	Planificación docente	%	% Calif.												
Página estática	<ul style="list-style-type: none">Definir un proyecto aplicando Node y Express que publique un sitio web estático que muestre una tabla que contenga por lo menos 10 filas, con la siguiente estructura: <table><tr><th>Docente</th><th>Periodo</th><th>Horas Docencia</th><th>Horas planificación</th><th>Horas tutorías</th><th>Horas gestión</th></tr><tr><td>Ing. Juan Carlos Sendón</td><td>2021-1</td><td>18</td><td>6</td><td>4</td><td>12</td></tr></table>	Docente	Periodo	Horas Docencia	Horas planificación	Horas tutorías	Horas gestión	Ing. Juan Carlos Sendón	2021-1	18	6	4	12	10%	
Docente	Periodo	Horas Docencia	Horas planificación	Horas tutorías	Horas gestión										
Ing. Juan Carlos Sendón	2021-1	18	6	4	12										
Base de datos y modelo con mongoDB	Crear un modelo con Mongoose con la estructura: <ul style="list-style-type: none">✓ Docente✓ Periodo✓ Horas docencia✓ Horas planificación✓ Horas tutorías✓ Horas gestión✓ Tipo de <u>error</u>	10%													

	Y la funcionabilidad para insertar y consultar las colecciones de MongoDB Atlas, utilizando este modelo.		
Cron Job	<ul style="list-style-type: none"> ✓ Cada 2 minutos captar información de la página web estática utilizando librerías externas (node-cron y axios) y ubicarla en un arreglo listo para ser insertado en cualquier tipo de persistencia. (para facilitar esta tarea puede utilizar cualquier clase o marca sobre el HTML de la tabla). 	20%	
Servicio REST	Definir un servicio REST que presente: <ul style="list-style-type: none"> ✓ Método POST que almacene información utilizando el modelo previamente definido. ✓ Métodos GET que devuelva las colecciones que presenten algún error en el cálculo de horas de trabajo semanal (Todas sus actividades deben sumar 40 horas). 	20%	
Interoperabilidad e integración	<ul style="list-style-type: none"> ✓ El cron job debe almacenar la información en la base de datos de MongoDB Atlas, utilizando el servicio REST y el modelo en mongoose. ✓ Utilizando Postman conectarse al servicio REST y consultar los movimientos con error. 	30%	
Repositorio GIT	<ul style="list-style-type: none"> ✓ Subir código en repositorio GIT con acceso a joancemac@gmail.com. 	10%	
TOTAL	<u>Suma de todos los porcentajes.</u>	<u>100%</u>	

Forma de entrega:

El URL del proyecto en GIT y el URL de video explicativo del producto final, deberán ser ubicados en un documento formato PDF en el que consten sus datos personales y subidos al aula virtual Moodle (La actividad estará ubicada en la Semana 8).

El video y el repositorio GIT será la evidencia de la entrega en cuanto a su examen.

NOTA: De presentar problemas de conexión al internet se acepta y recomienda trabajar con una versión local de MongoDB, así como llevar GIT en su computadora y tener como evidencia el repositorio local (carpeta oculta. git)

Elaborado por	Revisado y aprobado en área	Recepción Comisión Académica
Ing. John Cevallos Macías Docente	Ing. Edgardo Panchana Flores Colectivo disciplinario Software	Ing. John Cevallos Macías Comisión Académica