

**EXAMEN DEL PARCIAL I**  
**Carrera: Tecnologías de la información**

**Asignatura:** Aplicaciones Web II  
**Fecha:** Noviembre/2021  
**Estudiante:** DANNY JAVIER MENDOZA MACIAS

**Docente:** John Cevallos  
**Nivel:** Sexto Nivel "B"  
**Cédula:** 1315906618 **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 funcionabilidad, 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:	Contrato de TV Pagada	%	% Calif.												
Página estática	<ul style="list-style-type: none"><li>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:</li></ul> <table><tr><th>Fecha Inicio</th><th>Tipo de Servicio</th><th>Meses contratados</th><th>Total contrato</th><th>Meses transcurridos</th><th>Total abono</th></tr><tr><td>01-01-2021</td><td>Plata</td><td>12</td><td>120</td><td>7</td><td>70</td></tr></table>	Fecha Inicio	Tipo de Servicio	Meses contratados	Total contrato	Meses transcurridos	Total abono	01-01-2021	Plata	12	120	7	70	10%	
Fecha Inicio	Tipo de Servicio	Meses contratados	Total contrato	Meses transcurridos	Total abono										
01-01-2021	Plata	12	120	7	70										
Base de datos y modelo con mongoDB	<p>Crear un modelo con Mongoose con la estructura:</p> <ul style="list-style-type: none"><li>✓ <b>FechaInicio</b></li><li>✓ <b>TipoServicio</b></li><li>✓ <b>MesesContratados</b></li><li>✓ <b>TotalContrato</b></li><li>✓ <b>MesesTranscurridos</b></li><li>✓ <b>TotalAbonado</b></li><li>✓ <b>Tipo de error</b></li></ul> <p>Y la funcionabilidad para insertar y consultar las colecciones de MongoDB Atlas, utilizando este modelo.</p>	10%													
Cron Job	<ul style="list-style-type: none"><li>✓ Cada 2 minutos captar información de la página web estática utilizando librerías externas (node-cron y axios) y ubicarla en un</li></ul>	20%													

	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).		
Servicio REST	Definir un servicio REST que presente: <ul style="list-style-type: none"> <li>✓ Método POST que almacene información utilizando el modelo previamente definido.</li> <li>✓ Métodos GET que devuelva las colecciones que presenten algún error <b>en el cálculo de los meses transcurridos (Debe considerar la fecha actual menos la fecha de inicio para esta validación).</b></li> </ul>	20%	
Interoperabilidad e integración	<ul style="list-style-type: none"> <li>✓ El cron job debe almacenar la información en la base de datos de MongoDB Atlas, utilizando el servicio REST y el modelo en mongoose.</li> <li>✓ Utilizando Postman conectarse al servicio REST y consultar los movimientos con error.</li> </ul>	30%	
Repositorio GIT	<ul style="list-style-type: none"> <li>✓ Subir código en repositorio GIT con acceso a <a href="mailto:joancemac@gmail.com">joancemac@gmail.com</a>.</li> </ul>	10%	
<b><u>TOTAL</u></b>	<b><u>Suma de todos los porcentajes.</u></b>	<b><u>130%</u></b>	

#### 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