

**Dirección de**

**Tecnologías de la Información**

**-Senacyt-**

**Evaluación**

## Desarrollador de Sistemas de Información

**Confidencial**

Evaluación de conocimientos teóricos y prácticos

# Nombre: Edwar Everaldo Zacarias

# Correo electrónico: cuentaroja11@gmail.com

# Github: https://github.com/Cuentaroja11/Prueba\_FullSatack.git

# Dirección de Tecnologías

*“La Dirección de Tecnologías de la información de la Senacyt es el órgano responsable de planificar, organizar, dirigir, supervisar y controlar los recursos y sistemas informáticos de la Secretaría, estableciendo normas, estándares, políticas y metodologías relacionadas con las tecnologías de la información, comunicaciones e infraestructura, así como dar apoyo tecnológico para el desarrollo de las Tecnologías de la Información y Comunicación a Nivel Nacional. “*

Evaluación teórica -50pts-

# Instrucciones

A continuación, se presenta una serie de enunciados los cuales tiene como objetivo medir su capacidad de toma de decisión, así como conocer su punto de vista en el desarrollo de sistemas informáticos. Todos los enunciados de selección múltiple deben justificar su respuesta.

## Parte 1: Scrum -2pts cada una-

1. **¿Qué es Scrum?**
2. Una metodología de desarrollo de software
3. **Una metodología ágil de proyectos**
4. Un marco de trabajo
5. Otro

Justifique su respuesta: Scrum es na rutina de trabajo para garantizar calidad y eficiencia en el desarrollo de proyectos informáticos.

1. **Analice y responda: Se planificaron 6 sprints, pero en el transcurso del desarrollo del producto, nos damos cuenta de que el sprint número 5 ya no genera valor, ¿qué propone usted?**
2. Cancelar el sprint
3. Realizar el sprint porque ya fue planificado
4. **Consultar a la parte interesada.**
5. Consultar al Project Manager
6. Otro

 Justifique su respuesta: Esto debido a que la decisión de agregar o quitar funcionalidades (historias) dentro de la planificación del proyecto en el desarrollo del sistema, es propia de la parte interesada.

1. **¿Cuál de los siguientes actores no deben de existir en Scrum?**
2. Scrum Master
3. Product Owner
4. **Project Manager**
5. Developer team
6. Stakeholder
7. Ninguna de las anteriores

 Justifique su respuesta: Project Manager no es un rol de la metodología Scrum.

1. **Estamos realizando el sprint 2, pero nos damos cuenta de mejoras que podemos incluir en el sprint 3, ¿lo podemos hablar en medio del sprint 2 o hasta el Sprint Planning 3?**
2. Sí, se puede hablar en el 2, es el Sprint Refinement
3. **No, eso le corresponde al Sprint Planning 3**
4. Lo decide el Scrum Master
5. Lo decide el Product Owner

 Justifique su respuesta: para que la metodología ágil funcione efectivamente y produzca los resultados esperados, debe respetarse la secuencia de sus actividades. Al delegar este tipo de situaciones a la metodología, esta misma nos va a garantizar que en su debido tiempo todos los alcances serán cumplidos.

1. **El objetivo principal de scrum es:**
2. **Entregas ágiles y continuas**
3. Desarrollo de software
4. Hablar más con el cliente o interesado
5. Generar valor vía el desarrollo de un producto
6. Otro

Justifique su respuesta: esta metodología permite realizar avances continuos en un proyecto dejando las historias funcionales de acuerdo al Sprint Planning. Permitiendo realizar entregas frecuentes de los avances en el ciclo de vida del desarrollo de software.

## Parte 2: Desarrollo de software -4pts cada una-

1. **Es el momento de subir el código a control de cambios, usted está en la rama de desarrollo local y debe ser trasladado a producción, pasando por ambiente de pruebas. Indique cuál es la secuencia de instrucciones ideal para validar la integridad del sistema y el control de versiones. Seguir el gitflow establecido para el proyecto en cuestión.**

1. **Se tenía un sistema antiguo en un servidor, dicho servidor dejó de funcionar y con ello, por el modelo monolítico que tenía, hubo perdida de datos, ¿cómo se hubiera evitado este fallo? Implementando una arquitectura de microservicios.**

1. **Se tiene un API que debe de traer 1 millón de datos aproximadamente, la consulta puede tardar 1 minuto, usted propone una llamada**:
2. Sincrónica
3. **Asincrónica**

Justifique su respuesta: debido a que el tiempo de respuesta es demasiado largo, la mayoría de navegadores la tomaría como error de conexión al no especificar la llamada como Asincrónica, provocando que el resto de componentes de la API no carguen hasta obtener la respuesta.

1. **Una versión publicada en producción presenta inestabilidad, ¿qué acciones toma usted? Hacer un merge de la rama con la de HotFix, para solucionar por aparte lo más rápido posible y posteriormente volver a sacar a producción.**

1. **¿Tiene algún beneficio una tipificación de un lenguaje o es indiferente? Por lo general son los lenguajes compilados donde se denota el beneficio de conocer los tipos de variable en tiempo de compilación para así reservar adecuadamente la memoria stack o heap.**

## Parte 3: Bases de datos -4pts cada una-

1. **Defina con sus propias palabras qué es un modelo maestro-detalle. Es una tabla que sirve para relacionar 2 o más tablas a través de sus llaves primarias o con llaves compuestas.**
2. **Un sistema necesita la carga de archivos, tales como expedientes, fotografías, etc, ¿cuál es la mejor opción para guardar los archivos?**
3. En el sistema de archivos del servidor
4. En una base de datos relacional
5. En una base de datos documental
6. En otro servidor
7. **Otro**

 Justifique su respuesta: es más sencillo administrar este tipo de datos en servicios que se orientan a dicha función tal como los bucket de aws.

1. **¿Una hoja de cálculo es una base de datos?**
2. Sí
3. **No**

Justifique su respuesta: las hojas de cálculo como por ejemplo las de Excel son tablas hash. Y para que una colección de información sea una base de datos, debe garantizar la integridad de los datos, así como el acceso a los mismos, características que no cumple una hoja de cálculo.

1. **¿En qué casos utilizaría usted una vista? Para realizar la carga masiva de datos de archivos csv, xml, json, con datos que aun no has sido normalizados o que están pendientes de un proceso ETL.**
2. **¿Qué tipos de bases de datos conoce o ha trabajado, describa con sus propias palabras una breve reseña y mencione algunas tecnologías?**
   * **MySQL base de datos relacional que usado en ambientes Linux.**
   * **SQLServer base de datos propia de Microsoft, la prefiero por su facilidad al hacer backups y para el manejo de replicas.**
   * **ORACLE la use en su versión 11g muy buena para hacer migraciones de datos y muy buen soporte técnico.**
   * **PostgrSQL una base de datos muy liviana**
   * **MongoDB: es una base de datos no relacional y de tipo documental.**
   * **Redis: la he usado para base de datos de archivos de tipo temporal.**

1. **Mencione 5 motores de bases de datos NoSQL y en qué casos los utilizaría.**
   * **MongoDB, para guardar información descentralizada con una lógica de negocios no relacionada.**
   * **Redis, para almacenar datos de cache, temporales de las transacciones.**
   * **MongoDB Atlas, con uso igual que MongoDB, pero con la ventaja de estar en línea.**
   * **Amazon DINAMODB, lo usaría en una arquitectura cuyo ecosistema este muy relacionado con los otros servicios de AWS con SaaS y PaaS.**
   * **Azure Cosmos DB, la usaría si hay restricciones de licencias bajo la marca Microsoft y para aprovechar su soporte técnico.**

1. **Explique la diferencia entre INNER JOIN, LEFT JOIN y RIGHT JOIN : con base en la teoría de conjuntos el INNER JOIN obtiene los datos que se interceptan entre ambos grupo de datos; LEFT JOIN obtiene los datos del primer grupo que no están en el segundo y los que se interceptan entre ambos; RIGTH JOIN obtiene los datos de la intersección entre los grupos de datos y los datos en el segundo grupo.**

Evaluación práctica -50pts-

# Instrucciones

A continuación, se presenta una serie de enunciados los cuales tiene como objetivo medir su nivel de conocimiento técnico y de desarrollo de software. Para lo cual se le solicita crear un ambiente de desarrollo local basado en PHP, MySQL/MariaDB, y otras tecnologías que se indican en el enunciado para resolver los escenarios planteados.

Así también, todo el código utilizado deberá subirlo a una carpeta GitHub para su evaluación en la fecha y horario indicado.

Deberá incluir este archivo, así como cualquier documentación que considere oportuna, no olvide notificar a Recursos Humanos el enlace a su carpeta en GitHub para programar una reunión de 20 dónde se le pedirá presentar la solución desarrollada.

# Serie I – 10 pts.

Se tiene un registro de personas y se necesita el almacenamiento de sus datos personales como nombres, apellidos, teléfono, dirección, fecha de nacimiento y otro dato personal de relevancia, estas personas pueden tener o no uno o varios logros académicos de los cuales pueden ser licenciatura, maestría, doctorado o también diplomados, certificaciones entre otros tipos de logros académicos, el logro académico debe de tener registrado año, título del programa, institución de la que lo obtuvo y otra información que sea necesaria y pertinente. De igual manera se desea que se registre sus áreas de desempeño de las personas, por ejemplo: ciencias de la computación, psicología, ciencias económicas, entre otros, recordando que pueden tener varias áreas de desempeño. Estas personas registradas deben de contar con usuario y contraseña del sistema.

Se desea que se construya una base de datos relacional del anterior enunciado. No olvide incluir el diagrama ER.

# Serie II – 15 pts.

Se necesita que realice un *frontend* en una librería o framework basada en JQuery que tenga lo siguiente:

* Login para las personas registradas
  + Actualización de datos personales
  + Actualización de datos académicos
  + Actualizar área de desempeño
  + Agregar o quitar logros académicos
  + Agregar o quitar área de desempeño
* Una vista (**no** se necesita un login para acceder) en la cual se pueda visualizar las personas registradas, así como el último grado académico que han conseguido con su respectiva información de dicho grado académico.
* Una vista (**no** se necesita un login para acceder) en la cual se pueda visualizar las áreas de desempeño registradas, así como el número de personas que tienen esa área de desempeño.

# Serie III – 25 pts.

Se necesita que realice un *backend* de todo el manejo de base de datos (inserciones, actualizaciones, consultas, entre otros) y lógica del negocio del escenario planteado en los incisos anteriores por medio de APIs Rest en Node JS.

**Aspectos que evaluar**

* Modelo relacional creado con sentencias SQL
* Uso adecuado de llaves primarias, foráneas e índices (este último si es necesario)
* Uso adecuado de framework o librería JQuery, si lo necesita puede apoyarse de PHP.
* El acceso a datos, manejo de estos, lógica de negocio y demás debe de ser desarrollado **netamente** en Node JS y API Rest.
* Es libre utilizar los módulos de Node JS que necesite para crear las rutas de las APIs Rest o demás módulos.
* Para consumir las APIs Rest puede hacerlo desde JavaScript o PHP.