



**Dirección de
Tecnologías de la Información
-Senacyt-**

Evaluación

Desarrollador de Sistemas de Información Junior

Confidencial

¡Hola!

Queremos agradecerte por el tiempo invertido dentro del proceso de Selección.

Hoy te plantaremos un proyecto donde pondrás en práctica tus conocimientos y la experiencia que has obtenido en el transcurso de tu trayectoria laboral. Esta prueba está enfocada en tus conocimientos de desarrollo de software backend pero principalmente en el desarrollo de frontend

Nombre: Jorge Mario López Monterroso

Correo electrónico: jmario.lopez91@gmail.com

Github: <https://github.com/JorgeDev91>

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

1. ¿Qué es Scrum?

- a. Una metodología de desarrollo de software
- b. Una metodología ágil de proyectos
- c. Un marco de trabajo
- d. Otro

Justifique su respuesta:

Scrum es el framework más popular dentro de la metodología Agile, se basa en el aprendizaje y ajustes continuos. No es una metodología debido a que se adapta a cada proyecto en el que se desea utilizar y se pueden o no utilizar algunas de sus características según sea necesario.

2. ¿Cuál de los siguientes actores no deben de existir en Scrum?

- a. Scrum Master
- b. Product Owner
- c. Project Manager
- d. Developer team
- e. Stakeholder
- f. Ninguna de las anteriores

Justifique su respuesta:

El Project manager no forma parte de scrum.

3. 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?

- a. Sí, se puede hablar en el 2, es el Sprint Refinement
- b. No, eso le corresponde al Sprint Planning 3**
- c. Lo decide el Scrum Master
- d. Lo decide el Product Owner

Justifique su respuesta:

Debido a que en el sprint 2 se tienen objetivos ya establecidos, lo mejor es anotar todas las posibles mejoras y abordarlas en el sprint planning 3 que es donde corresponde.

4. El objetivo principal de scrum es:

- a. Entregas ágiles y continuas**
- b. Desarrollo de software
- c. Hablar más con el cliente o interesado
- d. Generar valor vía el desarrollo de un producto
- e. Otro

Justifique su respuesta:

Entregas ágiles, ya que con cada sprint se va entregando avances del producto de los cuales se puede obtener feedback del cliente, lo cual, genera mejoras constantes en el producto a entregar.

Parte 2: Desarrollo de software y base de datos

5. 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.

6. Una versión publicada en producción presenta inestabilidad, ¿qué acciones toma usted?

Revisar si todos los test de prueba fueron exitosos.

Notificar al equipo sobre los bugs o errores encontrados en la aplicación y formular un plan de acción para resolver los inconvenientes a la brevedad.

Si no es posible resolver los problemas en un tiempo corto, proponer dejar en función solo el MVP (Mínimo Producto Viable) y resolver los problemas con calma para que los errores sean arreglado de forma permanente.

7. Defina con sus propias palabras qué es un modelo maestro-detalle.

8. ¿Cuándo usarías Node.js en lugar de otras tecnologías en el lado del servidor?

9. Explica cómo funciona el sistema de módulos en Node.js.

Entendiendo que un módulo es un fragmento de código que realiza una función en específico, por lo tanto, en Nodejs se tienen dos sistemas de módulos que se pueden utilizar.

El primero es utilizando la palabra reservada "require" o requerir que permite importar un módulo desde el sistema de archivos local. Para poder utilizarlo, la extensión del modulo debe de ser ".js"

El segundo es en el que se utiliza el ES6, donde se utilizan las palabras reservadas "export" e "import" para exportar o importar módulos. Para poder utilizarlo, la extensión del módulo debe de ser mjs.

10. ¿Cómo manejas la comunicación entre componentes en React?

Se puede realizar de varias maneras dependiendo del tamaño del proyecto que se esta realizando. Si es un proyecto pequeño, se puede hacer por medio de los props de los componentes. Si es un proyecto grande, se puede utilizar redux para manejar estados en el proyecto y tener un punto focal donde se manejan las variables que se utilizan en varios componentes. En caso de que no se quiera utilizar redux, se puede utilizar 'context' que es nativo de react.

Evaluación práctica -50pts-

Instrucciones

Esta sección tiene como objetivo medir tu nivel de conocimiento técnico en el área de desarrollo de sistemas de información. A continuación se te brinda una serie de requerimientos funcionales con los cuales debes construir un sistema básico de registro de usuarios en un sistema.

1. Se debe crear una base de datos que maneje usuarios de un sistema. Esto debe incluir nombre, correo electrónico, contraseña como mínimo.
2. Realizar el registro y login del sistema para usuarios.
3. Debe crearse una tabla de proyectos, los cuales deben incluir como mínimo: Nombre del proyecto, Fecha inicio, Fecha fin, Id del proyecto, descripción, objetivos, departamento de ejecución, monto de ejecución.
4. Cada usuario registrado en el sistema puede tener asociado 1 o más proyectos, y cada proyecto puede estar asociado a uno o mas usuarios.
5. El usuario puede modificar sus datos y/o actualizar su contraseña.
6. El administrador puede ver una lista general de todos los proyectos con su detalle.
7. El administrador puede ver una lista general de los usuarios registrados en el sistema.
8. Al seleccionar el usuario el administrador puede ver una lista de los proyectos asociados con su detalle.
9. El administrador puede asociar un proyecto a uno o más usuarios.
10. El administrador puede dar de baja un usuario y deshabilitar un proyecto.

Para la realización de esta tarea deberá realizar:

1. Base de datos de las tablas necesarias, incluir esquema entidad relación
2. Api de servicios para manejo de datos y backend
3. Front end para la visualización de lo solicitado.

Aspectos que evaluar

- Diseño general
- Manejo de errores
- Usabilidad
- Interactividad

Lenguajes:

- Node.JS
- Api Rest
- React
- MySQL/PostgreSQL