



**TECNOLOGICO
NACIONAL DE MÉXICO**



INSTITUTO TECNOLÓGICO DE TLAXIACO

INGENIERIA EN SISTEMAS COMPUTACIONALES FUNDAMENTOS DE INGENIERIA DE SOFTWARE (5US) ENTREVISTA

NOMBRES INTEGRANTES:

- **ALVARO FROYLAN PACHECO OSORIO**
- **ANTONIA MARTINEZ SANDOVAL**
- **Yael de Jesús Santiago Ortiz**
- **ISAAC CORONEL SANTIAGO**

PROFESOR: ING. ROMAN CRUZ JOSE ALFREDO



SEPTIEMBRE DEL 2023

Índice

INTRODUCCIÓN.....	3
ENTREVISTA.....	4
CONCLUSIÓN	8

INTRODUCCIÓN

Mediante el uso de una serie de preguntas que buscan ampliar la comprensión teórica acerca del desarrollo de software enfocándose en una perspectiva personal basado en el desempeño laboral.

El principal objetivo de entrevistar a un desarrollador de software es conocer la experiencia práctica, conocimientos y perspectivas personales en el campo, buscamos entender su enfoque orientado hacia el desarrollo, la capacidad para abordar desafíos técnicos y las habilidades necesarias para trabajar en equipo. Además, deseamos explorar las opiniones con base en experiencias sobre las mejores prácticas y las tendencias que surgen en este campo.

A continuación, se presenta un resumen de las respuestas y la información recopilada durante la entrevista.

DESARROLLO DE LA ENTREVISTA

Nombre de Ingeniero entrevistado: Dalia silva Martínez

1. ¿Qué es ser un desarrollador de software?

Un desarrollador es una persona con conocimientos amplios de programación, metodologías de desarrollo de software, ingeniería de software y que se dedica a desarrollar, así a crear software para resolver problemas.

Pues bueno básicamente es ser un desarrollador de software es ser alguien que tiene la capacidad, las habilidades y el conocimiento crear situaciones, y demás a partir de un código fuente que es de lo que está en lo que un software como tal es una serie de códigos y palabras, entonces ser un desarrollador de software es básicamente ser alguien que puede manipular todo este tipo de información para lo un cometido.

2. ¿Qué es lo que más le gusta de su profesión?

Lo que más me gusta de mi profesión es la libertad con la que puedes trabajar esos aspectos por lo que muchas veces puedes dedicarte a un proyecto en específico o a varios.

3. ¿Cuál fue uno de sus retos más difíciles que ha pasado por tu carrera?

En una ocasión que me estaban pidiendo que hiciera un trabajo de un lenguaje que no conocía y lo me tuve que aprender, pues a marchas forzadas porque no lo había visto en la escuela. Pero lo importante, es que todas las bases del lenguaje las tenía ya que es la base de programación, eso sucede cuando aprendes a programar, puedes tú rápidamente aprender otro lenguaje de programación u otro tipo de entorno como entorno o framework lo que necesitas para trabajo. Eso es lo importante.

4. ¿Cuánto tiempo lleva haciendo desarrollador?

Bueno, yo estuve en desarrollo de software con ocho años y ahora me dedico prácticamente a dar clases de programación. Ahorita ya no estoy desarrollando me he dedicado a dar clases porque ya tengo tiempo completo en la institución y entonces ya no propiamente en desarrollo de software, pero si imparto todas las materias que se implica en el área de la programación, en ocasiones y últimamente con los chicos en cuanto a los proyectos participan en experiencias en lo que antes era en IT y en lo que ahora se le conoce IBM y en eso he estado trabajando con ellos.

5. ¿Cómo es el trabajo en equipo entre su área de desarrollo?

Hay un poco de complicación, a veces y a veces no. Porque hay unos que se atrasan mucho y en algunos casos los tienes que ayudar o por lo menos explicarles y decirles que es exactamente lo que tienen que hacer.

6. Problemas al trabajar en su área

- Podría ser porque la gente no trabaja a la misma velocidad de uno y se llega a atrasar el proyecto.
- Las matemáticas y física, algo común que se verá en esta área.
- Por el tema de la comunicación, ya que puede llegar a ver con función en lo que quieres que hagan y no hagan.
- Por el tema de cuándo se va entregar el proyecto
- También podría ser por que la función o el método te dé es de diferente tipo de valor y tú quieras que te dé un valor en específico y no te lo de.

Entonces pienso que en esas partes es un poquito complicado.

7. ¿Es mejor trabajar solo o en equipo?

No digo que sea mejor trabajar solo ya que por el tiempo hay cosas que no se pueden acabar solo. Y además no solo necesitas del programador, necesitas de otras personas que te ayuden como en el del diseñador. A veces uno dice para qué quiera uno el diseñador, no, si se necesita. ¿Quién va a hacer que se vea bonito tu software? Además, en software necesitas al que trabaja detrás o de fuera. Así que por más que quieras no puedes trabajar solo.

8. ¿Cómo gestionas la comunicación y la colaboración en estos casos?

En mi caso para poder saber qué es lo que se necesita realizar cuáles son los requerimientos del proyecto hacer y basado en esto pues se hace una lluvia de ideas para cuántos puntos o qué tanto es lo que se requiere hacer, qué tanto es lo que se requiere realizar y con en mente encuentren allí este repartir cada una de las actividades basados en las aptitudes y los de cada uno todo el proyecto en específico o con alguien más hagan esos equipos dentro del equipo.

9. ¿Cuál es el lenguaje de programación preferido?

Pues bueno más allá de preferido a no conocí muchos realmente conocí Python conocí Pascal, pero Pascal es muy viejo entonces hasta donde yo sé ya no existe que es uno de los que siguen hasta ahorita vigentes ah el único con el que he trabajado fácil de entender, aprender no hay mayor problema con C++.

10. ¿Has trabajado en proyectos que involucren tecnologías emergentes como inteligencia artificial o blockchain?

Realmente no he profundizado en esa área, en esos proyectos, en inteligencia artificial, trabajé un poco cuando estaba en la escuela, pero fue muy muy poco fue con redes neuronales, pero ya tiene muchísimo tiempo de eso.

11. Estrategias empleadas para que un código sea limpio y fácil de mantener o sea eficiente.

En un código puedas poner tipos de señalización para poder ubicar algo como tal en todo tu código, otra de las maneras es hacer un índice conforme vas avanzando en tu código ir creando un índice aparte para ir sabiendo en qué situación están las partes de tu código un espacio entre bloques de código para que no te confundas con otro, si es que necesitas agregar información etcétera, y así sea mucho más fácil.

12. Mensajes de motivación para futuros desarrolladores de software

Pues un mensaje tal vez un tanto trillado puede ser que no desistan si es lo que les gusta, continúen adelante es un camino largo y difícil, pero al final si es lo que realmente quieren van a obtener por lo difícil o por lo complicado que sea la situación para todo hay una salida ya sea anotándolo para así que puedan o quieran recordarse lo cada día. No dejen de aprender lo mejor que puedan y mucho menos dejen que esos conocimientos se les olviden.

Escuchen y aprendan todo lo que quieran realizar, haganlo por y para ustedes; es un pensamiento tanto egoísta pero la verdad es que funciona bastante, hagan las cosas que les gustan y no piensen en negativo para lograr un buen aprendizaje.

CONCLUSIÓN

Durante la entrevista, el candidato proporcionó información valiosa sobre su experiencia y enfoque en el desarrollo de software. Demostró un sólido conocimiento de metodologías de desarrollo, destacó su preferencia por la programación en (escribir lenguaje) debido a su versatilidad. El candidato también discutió su participación en proyectos anteriores, resaltando su capacidad para adaptarse a plazos ajustados y su énfasis en la escritura de código limpio y eficiente.

Además, el candidato mencionó su experiencia en la gestión de equipos remotos y la importancia de la comunicación en estos contextos. Habló sobre desafíos técnicos pasados y cómo los abordó, destacando su capacidad para resolver problemas de manera efectiva. También subrayó la importancia de las pruebas automatizadas y la seguridad en el desarrollo de software.

En resumen, el candidato demostró ser un desarrollador de software con una sólida base técnica, habilidades de resolución de problemas y un enfoque orientado a la calidad.