

# Reflexiones Unidad 1 por César Alejandro Huerta Méndez

## Tema 1: Retrospectiva

La Ingeniería de Software es un área que, al tener su apogeo en la época actual, está llena de prejuicios y mitos respecto a las competencias y habilidades de un egresado en esta licenciatura, y, reflexionando de manera personal, puedo percatarme de que incluso yo escogí esta carrera partiendo de estos estereotipos no fundamentados.

Si bien mi decisión no fue incorrecta (algo que puedo incluso afirmar con total seguridad en este momento), las razones por las que la tomé quizás no fueron las adecuadas: antes de comenzar mis estudios en IS, consideraba que el plan de estudios formaba netamente *programadores*, personas que estarían frente a sus computadoras durante horas mientras intentan compilar un código que les asegura un sueldo alto, y era algo a lo que aspiraba hasta ese entonces, no obstante, mi perspectiva cambió drásticamente desde la primera plática con el coordinador del área en la universidad; IS es más que crear videojuegos o programar, y reducirlo a dichas actividades sería menospreciar las capacidades y potencial que los ingenieros egresados poseen para liderar proyectos de desarrollo, mantenimiento e innovación de software, donde programar se vuelve una actividad trivial. Además, la asignatura de fundamentos de IS a través de su proyecto me ha otorgado una visión más amplia de lo que se realiza dentro de la industria: la importancia de la planeación, gestión de recursos y tiempo, y, sobre todo, la parte de análisis de requerimientos, que, entre otras cosas, hacen la diferencia entre un programados y un ingeniero. Y esto cobra más sentido cuando en la actualidad cualquier persona puede ser un programador tomando un curso online en menos de medio año, pero no todos cuentan con las *soft skills* necesarias para ser un ingeniero de software.

## Tema: IS vs Otros Roles

Para este tema se reflexionará sobre las diferencias entre un ingeniero en IS y un Front End developer, dos profesiones o títulos relacionados, pero bastante distantes.

Es pertinente definir qué es un Front End developer: se trata de un especialista en el área “visual” (interfaz) de un software, al igual que es el encargado de que este programa luzca atractivo y sea fácil de usar para el usuario.

Partiendo de esta definición ya existen varios puntos a contrastar, comenzando con las funciones y naturalezas de cada rol: el término calificativo *I shaped person* hace referencia a que el conocimiento de una persona se orienta únicamente a un tópico particular, por ejemplo, el Front End, Back End, etc.; estas personas cuentan con facilidad para realizar tareas dentro de su área de trabajo, sin embargo, desconocen muchos aspectos de otras áreas diferentes a las que atienden, en resumen, son eficientes pero poco versátiles en el trabajo. Por su parte, a un ingeniero de software le es más ameno el término *T shaped person*, ya que es el tipo de persona que

posee conocimientos generales de diversas áreas de su carrera, y aunado a eso, tiende a especializarse en una sola, es decir, son tanto eficientes como versátiles, y si desconocen algo, pueden aprenderlo rápidamente gracias a que cuentan con las bases para ello.

Para ilustrar lo anterior, supongamos que se entrevista a un Front End developer certificado y a un recién egresado ingeniero de software y durante la entrevista se les plantea un problema ficticio del cuál tienen que definir los requerimientos del cliente y proponer al menos tres soluciones. Aquí es donde las diferencias se hacen notorias: el programador de Front End podría tener problemas para analizar las necesidades y determinar los requerimientos que su cliente le solicita, después de todo, no son tareas propias de desarrollo del front end; en cambio, el ingeniero cuenta con conocimientos generales para estas tareas.

En la actualidad, muchas compañías tienden a buscar personas T shaped ya que “existen muchos I shaped”, de ahí que estudiar una ingeniería otorgue más oportunidades tanto laborales como de crecimiento a sí se optara por una certificación o especialidad en su lugar.

## **Reflexión x2**

Una reflexión es un tipo de texto que se caracteriza por el análisis de un tema y la expresión de un punto de vista o comentario argumentado sobre él; son texto de gran valor literario ya que dan pauta al desarrollo de ensayos y tesis. Existen distintas metodologías (no formales) para realizarlas de manera correcta, y si bien algunas pueden contener más o menos pasos, la mayoría de estas coinciden en los siguientes:

- 1- Se debe escoger un tema del que se posea conocimientos previos, los suficientes para realizar inferencias y deducciones de él. El tema es de libre elección, desde la trama de una película hasta el estado actual de la conciencia colectiva.
- 2- Posteriormente se debe realizar una investigación exhaustiva sobre el tema en cuestión, cuestionar las concepciones que tenemos y depurarlas para obtener información útil de ellas.
- 3- Una vez se considere que se cuentan con los conocimientos suficientes, se realiza la redacción: puede ser directa o comenzar con ideas principales que serán unidas posteriormente. La extensión de la reflexión es a elección del autor, lo importante es que la calidad del producto sea la esperada.
- 4- Finalmente se revisa y corrige la reflexión, abarcando desde el apartado gramatical hasta el lógico.

Utilizando como ejemplo la anterior reflexión sobre *IS vs otros roles*, podemos destacar:

- Se abarca un tema específico se delimita correctamente: se exponen las diferencias entre un área y otra, así como se ejemplifica el punto de vista de manera concreta.
- Antes y durante su realización consulté distintas fuentes de información, y contrasté lo recabado con lo aprendido durante este semestre en LIS, así como mis conocimientos e ideales previos a este.
- Revisé constantemente que exista cohesión y coherencia en lo escrito, particularmente que la reflexión no se desviara del tema.

Por lo anterior, considero que mi reflexión cumple con las características que hacen válida y valiosa una reflexión.

**Tema libre: ¿Cuáles son las competencias que un ingeniero en software debe dominar y cómo medirlas?**

En la actualidad, muchos modelos educativos y filtros de contratación laboral analizan y priorizan la evaluación de competencias a la evaluación de conocimientos. Sobre este tema es posible reflexionar: ¿qué competencias se espera que un ingeniero en software deba poseer al egresar de sus estudios? El CONAIC responde a dicha pregunta con un repertorio de 12 competencias específicas para el área de IS (de utilidad resaltar que, además, estas coinciden en un gran nivel con las del plan de estudio de la UADY), por nombrar algunas: “diseña software, realiza mantenimiento de software, asegura la calidad del software, usa herramientas para creación de software”.

Considerando lo anterior, es válido preguntarse, ¿son estas competencias las apropiadas?, ¿bajo que criterios se establecen como correctas?, ¿se podrían agregar o eliminar competencias? Dar una respuesta a estos cuestionamientos requeriría de un alto nivel de conocimientos y documentación, sin embargo, lo cierto es que el CONAIC es un organismo de alto prestigio, conformado por ingenieros que constantemente se hace estas mismas preguntas para llegar a conclusiones más concretas.

Respecto a cómo se miden estas competencias también es posible reflexionar, ya que requieren de métodos más abstractos que cuando se miden conocimientos, después de todo, las competencias son de carácter cualitativo mientras que los aprendizajes son más cuantitativos. Es por ello que los exámenes contrarreloj o pruebas prácticas pueden no dar resultados útiles respecto al potencial de una persona y su capacidad de desarrollarse en el área laboral y social, entonces, ¿qué propuesta se podría dar para medirlas? Una solución podría ser tomar a varios candidatos a evaluar (por ejemplo, un grupo de recién egresados) y asignarles una tarea de alto nivel a cumplir en un tiempo determinado, entonces evaluar aspectos como la organización, capacidad de innovación, etc., en un periodo de tiempo adecuado (7 días, 15 días, etc.); el problema con esta metodología es que requiere de muchos gastos comparado a las tradicionales entrevistas con exámenes

cuantitativos, y es justamente por ello que la mayoría de empresas siguen reclutando a través del sistema clásico, pese a que cada vez se torna más retrógrado.

### **Bibliografía:**

[1] CONAIC. 2017. COMPETENCIAS TRANSVERSALES Y ESPECÍFICAS DE LOS PERFILES QUE EVALÚA CONAIC DEFINIDAS POR ANIEI Y CONAIC. Recuperado el 15 de octubre de 2022 de: <https://www.conaic.net/publicaciones/7BIS%20COMPETENCIAS%20DE%20PERFILES%20DE%20INFORMATICA%20Y%20COMPUTACION-UV2018.pdf>