

¿Por qué es esencial que los ingenieros de software prioricen el bienestar público?

Los ingenieros de software tienen la responsabilidad de desarrollar productos que impactan directamente la vida de las personas, desde aplicaciones móviles hasta sistemas críticos como los que controlan hospitales o infraestructuras públicas. Priorizar el bienestar público significa considerar cómo su trabajo puede mejorar la calidad de vida, proteger la privacidad y garantizar la seguridad de los usuarios. Ignorar este enfoque puede llevar a consecuencias graves, como violaciones de datos, fallas en sistemas esenciales o desigualdades en el acceso a la tecnología. Es un recordatorio de que detrás de cada línea de código hay personas cuyas vidas pueden verse afectadas positiva o negativamente.

¿Qué significa mantener la calidad del producto en software?

Aunque la mayoría pensamos que se trata de asegurarse de que el software funcione sin errores, y pues, aunque obviamente es fundamental. También se trata de desarrollar un programa o aplicación que cumpla con las expectativas de los usuarios y clientes. Un programa o producto puede no tener ningún bug, pero si es difícil de usar (mal UX) o no cumple con las necesidades del cliente, no se puede considerar un producto de calidad por obvias razones. Se trata de iterar en mejorar el producto continuamente.

¿Qué implica el compromiso con el aprendizaje continuo?

Pienso que nuestro campo de estudio es de los que más requieren aprender continuamente, la tecnología evoluciona constantemente, y lo que hoy es super innovador, en el futuro con toda probabilidad estará obsoleto, por ejemplo, el caso de GPT 2, que en su momento parecía ser el techo técnico de modelos de lenguaje y pues ahora comparado con los modelos actuales se ve muy pobre. Además, no solo se trata de desarrollar nuestras habilidades profesionales (o duras), sino que también tenemos que mejorar constantemente las habilidades blandas. El mantenerse actualizado nos permite enfrentar desafíos de manera más efectiva y proponer soluciones más creativas, y obviamente, innovadoras que alguien que no se mantenga en constante aprendizaje. Siempre hay algo nuevo que aprender, y de nuevo reitero que en nuestro campo es de las ramas del conocimiento en donde más aplica

¿Qué prácticas promueven la transparencia y la integridad en el desarrollo de software?

La transparencia y la integridad son pilares éticos fundamentales. Algunas prácticas esenciales incluyen documentar de manera clara y detallada cada etapa del desarrollo, comunicar honestamente las limitaciones y posibles fallos del software, y

asegurarse de que el código sea accesible para auditorías internas o externas cuando sea necesario. También es importante evitar prácticas engañosas, como ocultar errores críticos para cumplir con plazos o prometer funcionalidades que no pueden entregarse. Estas acciones no solo construyen confianza con los usuarios y clientes, sino que también fomentan un entorno laboral más ético y respetuoso.

¿Por qué es importante el respeto y apoyo entre colegas en el campo de la ingeniería de software?

En la ingeniería de software (y en muchos ámbitos) es prácticamente imposible desarrollar un proyecto completamente solo; la mayoría de las veces se trabaja en equipos multidisciplinarios donde cada persona aporta sus fortalezas, por ejemplo cosas como el frontend, backend, UX, etc. Respetar a los compañeros no solo es tratar a todos con respeto, sino también pues valorar sus ideas, reconocer sus contribuciones y estar dispuesto a ayudar cuando no puedan resolver un problema solos. Este apoyo mutuo crea un mejor ambiente laboral, donde todos podemos crecer profesional y personalmente, haciendo referencia nuevamente a que tenemos que estar desarrollando constantemente todo nuestro espectro de habilidades (blandas y duras). Al final del día, el respeto y el apoyo en el equipo de trabajo fortalecen el sentido de comunidad dentro del campo y eventualmente elevan la calidad del producto que se entrega.