TECNOLÓGICA DEL ORIENTE
INSTITUCIÓN DE EDUCACIÓN SUPERIOR

INGENIERÍA DE SOFTWARE ÁGIL

PRINCIPIOS

PRINCIPIOS

1. Satisfacer al cliente mediante entregas tempranas y continuas, de software con valor.

Este principio enfatiza que el objetivo principal de cualquier proyecto ágil es generar valor real desde etapas tempranas. A través de entregas frecuentes, los usuarios pueden probar el producto, dar retroalimentación y ajustar sus necesidades, lo cual reduce el riesgo de crear soluciones que no se alineen con sus expectativas.

Ejemplo. Una empresa de e-commerce que lanza primero una versión básica de su carrito de compras, puede verificar su funcionalidad y mejorarla antes de desarrollar todo el sistema de pagos.

2. Aceptar cambios en los requisitos, incluso en etapas tardías del desarrollo.

A diferencia del enfoque tradicional, donde los cambios son vistos como fallas de planificación, en el enfoque ágil se reconoce que el entorno de negocios es dinámico. Incorporar cambios en fases avanzadas, se interpreta como una mejora continua del producto, y no como un obstáculo.

Ejemplo. Si una nueva regulación afecta la privacidad de los usuarios, el equipo debe estar en capacidad de modificar el sistema, incluso si ya se encuentra en producción.

3. Entregar software funcional con frecuencia, entre dos semanas y dos meses, con preferencia hacia los períodos más cortos.

Este principio promueve ciclos breves de desarrollo, llamados iteraciones. Cada ciclo debe producir un incremento funcional del software. Esto permite evaluar avances concretos y no solo promesas o documentación.

Reflexión. La entrega frecuente genera confianza con el cliente y permite detectar errores o mejoras rápidamente, haciendo más eficiente el proceso de desarrollo.

4. Colaboración diaria entre personas de negocio y desarrolladores.

Los equipos ágiles promueven la participación activa del cliente o de sus representantes, durante todo el proceso. De esta manera, los requisitos se ajustan constantemente a las prioridades del negocio.

Ejemplo. En una startup financiera, los desarrolladores trabajan de la mano con analistas de crédito, para construir herramientas más alineadas a las decisiones reales.

5. Construir proyectos alrededor de individuos motivados. Darles el entorno y el apoyo que necesitan, y confiar en que harán bien su trabajo.

Este principio reconoce que el talento y la motivación del equipo son el activo más valioso. Los entornos ágiles fomentan la autonomía, la confianza y el liderazgo distribuido, lo que resulta en equipos más comprometidos y productivos.

Reflexión. En lugar de controlar al equipo mediante microgestión, se promueve su empoderamiento y la toma responsable de decisiones.

6. El método más eficiente y efectivo de comunicar información dentro del equipo de desarrollo, es la conversación cara a cara.

Aunque los equipos remotos requieren adaptaciones, el principio subraya la importancia de la comunicación directa y continua, sin intermediarios innecesarios. Las herramientas digitales actuales, permiten emular este principio en entornos distribuidos mediante videollamadas, chats activos y pizarras colaborativas.

Ejemplo. Usar herramientas como Zoom o Microsoft Teams, permite reuniones diarias (dailys) donde los miembros del equipo pueden sincronizar su trabajo, de forma ágil.

7. El software funcional es la principal medida de progreso.

En lugar de evaluar el avance por cantidad de documentos generados o tareas completadas, lo que realmente cuenta es que el producto funcione. Esto evita falsas sensaciones de avance y orienta al equipo a entregar resultados tangibles.

Ejemplo. Un equipo puede haber completado el 80 % de las tareas, pero si no hay ninguna funcionalidad usable, el proyecto no ha avanzado realmente.

8. Los procesos ágiles promueven el desarrollo sostenible. Los promotores, desarrolladores y usuarios, deben ser capaces de mantener un ritmo constante de forma indefinida.

Este principio protege la salud mental y física del equipo. Se rechazan las jornadas extenuantes o la cultura del “crunch”, promoviendo en su lugar, ritmos equilibrados que permitan sostener la productividad a largo plazo.

Reflexión crítica. En muchas organizaciones, se confunde agilidad con rapidez a cualquier costo, cuando en realidad se busca estabilidad y mejora continua.

9. La atención continua a la excelencia técnica y al buen diseño, mejora la agilidad.

Sin calidad técnica, el producto se vuelve difícil de modificar o escalar. Por ello, se incentiva el uso de buenas prácticas como la programación limpia, las pruebas automatizadas y el diseño modular. Esto permite adaptarse más fácilmente a cambios futuros.

Ejemplo. Un equipo que documenta bien su código y mantiene una arquitectura limpia, puede incluir nuevas funciones sin poner en riesgo el sistema existente.

10. La simplicidad, el arte de maximizar la cantidad de trabajo no realizado, es esencial.

Este principio invita a eliminar el trabajo innecesario y a enfocarse solo en lo que aporta valor. Evitar sobre ingeniería, funcionalidades innecesarias o burocracia interna, mejora la eficiencia del equipo.

Ejemplo. En lugar de desarrollar tres módulos para exportar datos, se prioriza uno solo que cubra el 80 % de los casos de uso.

11. Las mejores arquitecturas, requisitos y diseños, emergen de equipos auto organizados.

En lugar de imponer estructuras desde arriba, se confía en que los equipos con experiencia y conocimiento del problema, generen las mejores soluciones. La auto organización fomenta la responsabilidad compartida y la innovación.

Reflexión. Un equipo que decide cómo organizar sus tareas y cómo colaborar, tiene más motivación que uno que solo sigue órdenes externas.

12. A intervalos regulares, el equipo reflexiona sobre cómo ser más efectivo, y ajusta su comportamiento en consecuencia

Este principio introduce la práctica de la retrospectiva, donde el equipo evalúa lo que funcionó, lo que puede mejorarse y cómo ajustar el proceso. Así, la agilidad no es solo externa, sino también interna, en la forma de trabajar.

Ejemplo. Tras notar que hay demoras en la revisión de código, el equipo decide establecer nuevos horarios de revisión conjunta, para reducir los cuellos de botella.