

# Desarrollo de Software

---

1. **¿Cuál de las siguientes características NO está asociada al modelo Waterfall?**
  - a) Hace énfasis en la documentación (*document-driven*).
  - b) Existe una secuencia ordenada de fases/actividades de desarrollo (por ejemplo, requerimientos, diseño, implementación, testing, etc.).
  - c) Se basa en realizar *planning up-front*.
  - d) Involucra al cliente en la mayoría de las fases/actividades del desarrollo.
  - e) Existe un *checkpoint* al final de cada fase.
2. **En un modelo iterativo con *timeboxing*, si para una iteración de 30 días ocurren inconvenientes que hacen que no se llegue con el *deliverable* a la fecha de entrega, ¿cuál de las siguientes acciones es más adecuada?**
  - a) Reprogramar la fecha de entrega para la iteración.
  - b) Al llegar a la fecha de entrega, recortar la funcionalidad a lo que se tiene en ese momento.
  - c) Reducir el *testing* y analizar opciones, a fin de cumplir con la mayor funcionalidad posible.
  - d) Re-planificar la funcionalidad a entregar, eliminando requerimientos de baja prioridad, pero manteniendo la fecha de entrega original.
  - e) Modificar el plan general de iteraciones.
3. **¿Cuáles considera que son las ventajas de contar con iteraciones (relativamente) cortas y con *timeboxing*, para un proyecto de desarrollo de software?**
4. **En el modelo iterativo-incremental (según C. Larman), cada iteración se ve como:**
  - a) Un mini-ciclo (o mini-proyecto) *waterfall*, donde se desarrolla para un conjunto de requerimientos, y se aprovecha el feedback para futuras iteraciones.
  - b) Un mini-ciclo (o mini-proyecto) *espiral*, donde se desarrolla para un conjunto de riesgos, y se aprovecha el feedback para futuras iteraciones.
  - c) Un mini-ciclo (o mini-proyecto) de 2 iteraciones, donde se desarrolla para la iteración actual, y se planea para la siguiente iteración (de forma de mitigar los riesgos).
  - d) Un prototipo, donde se desarrolla para todos los requerimientos, salvo para aquellos que involucran atributos de calidad.
5. **Para las siguientes duraciones “típicas” de iteración de desarrollo, mencionar el método que Ud. considera mejor se asocia a cada una (ej., Waterfall, Scrum, XP, SPL, etc.): 1 pto.**
  - Iteración de 2-12 semanas → Método ...
  - Iteración de 4 semanas → Método ...
  - Iteración de 2 semanas → Método ...
6. **Se tiene un proyecto que sigue un modelo iterativo-incremental, en el cuál se analizan y especifican todos los requerimientos antes de realizar el diseño, y/o luego se diseña toda la arquitectura antes de su implementación. ¿Cuál es el principal problema de este proyecto?**
7. **Explicar brevemente qué es el *backlog* y sus características, en el método Scrum.**
8. **Explicar brevemente en qué consiste la práctica de TDD (en métodos ágiles, como XP).**
  - a) En particular dentro de XP, ¿con cuáles otras prácticas debe combinarse TDD para obtener óptimos resultados?
9. **Criticar el Manifiesto Ágil (ver cuadro abajo) en el contexto de un proyecto con 200 desarrolladores, 5 millones de líneas de código, y una vida útil esperada del sistema de 20 años.**

# Desarrollo de Software

---

## **Manifiesto Agil**

We are uncovering better ways of developing software by doing it and helping others do it.  
Through this work we have come to value:

**Individuals and interactions** over processes and tools  
**Working software** over comprehensive documentation  
**Customer collaboration** over contract negotiation  
**Responding to change** over following a plan

That is, while there is value in the items on the right, we value the items on the left more.