vico open modeling http://www.vico.org/ser.htm © Derechos reservados http://creativecommons.org/licenses/by-nc/3.0/deed.es http://aprendeuml.com

Manifiesto UML-TRAD

Decálogo para entender qué es una metodología abierta orientada al desarrollo de software

1	**La clave son las personas**	El riesgo número uno de un proyecto no es la tecnología ni la complejidad. La clave es el casting de un equipo de profesionales motivado, capaz de **cooperar y compartir conocimiento**. No necesitamos líderes de proyecto, sino un equipo responsable y autosuficiente, **capaz de tomar decisiones** y evaluar su impacto con **criterios de riesgo, calidad y coste**.
2	**La mejor metodología es la que no se ve**	Está interiorizada en los **hábitos de trabajo** de un equipo de proyecto y facilita su razonamiento y actuación en base a un muestrario de casos de referencia.
3	**Buscar la simplicidad consciente**	Hemos de orientar nuestros modelos de información hacia una **simplicidad consciente**. Evitemos la complejidad deliberada. Con un **número reducido de elementos** de la notación UML (Unified Modeling Language), y sus reglas de combinación (gramática de los elementos de un sistema de información), es posible construir y comunicar estructuras y funcionalidades muy complejas. Por ejemplo, con seis piezas de Lego® de ocho módulos podemos construir más de 915 millones de figuras distintas. Por ejemplo, cualquier forma de vida conocida es una combinación de 6 elementos de los 143 posibles de la tabla periódica.
4	**Aplicar un esquema de cooperación**	**Compartir conocimiento**, saber **compartir recursos** y evitar un esquema de competencia dentro de un equipo, son los factores clave en el éxito del proyecto.
5	**Usar un vocabulario controlado**	Un vocabulario controlado con una notación formal que facilite nuestro **razonamiento visual**: ▪ Evita ambigüedades y dispersión conceptual. ▪ Mejora en nuestro nivel de comunicación técnica. ▪ Desarrollamos productos complejos con una mayor fiabilidad, calidad, y eficiencia. ▪ Manejamos una documentación minimalista. ▪ Disponemos de trazabilidad entre la especificación y el producto.

 © Derechos reservados http://creativecommons.org/licenses/by-nc/3.0/deed.es

6	**Hay que triangular Espacio, Tiempo y Proceso**	**Espacio** define la **escala del proyecto** donde vamos a proyectar el tiempo y los procesos. ▪ A qué **categoría de proyecto** nos enfrentamos en función de su prioridad, complejidad y presupuesto. ▪ Cuáles son los **factores de riesgo** y cómo incidirán en el control de un buffer de ajuste de horas. ▪ Qué nivel de **exigencia de calidad** y qué criterios vamos a aplicar en las certificaciones de entregables. **Tiempo** define un esquema de **fases** con sus hitos de **transición**. ▪ Cuándo hay que entrar en una fase ▪ Cuántos ciclos tendrán una fase (iteraciones - esprints) ▪ Cuál será la duración de una iteración - esprint ▪ Cuándo hay que aplicar un determinado proceso **Proceso** define los **roles** participantes habilitados para realizar unas tareas orientadas a la elaboración de un **repertorio de entregables**, con un nivel de dedicación distinto en cada fase. ▪ Qué entregable hay que producir (obligado, recomendado, opcional) ▪ Qué competencia es necesaria (Roles) ▪ Cómo se realiza (Tareas - Actividades) ▪ Qué dependencia tiene cada Actividad (Flujo de Trabajo) ▪ Quién y con qué criterio verifica un entregable (Certificación)
7	**Realizar un casting de Actores**	Nuestra primera tarea es averiguar qué Agentes aportan o reciben valor del sistema en discusión. La **cadena de valor** puede surgir a través de una interfaz gráfica de usuario, a través de una interfaz de comunicación, o mediante un documento de negocio, dentro de un Punto de Actuación. Tenemos que ser muy restrictivos en el casting. Cada Actor puede acarrear una media de 6 Casos de Uso (Esquema de granularidad Bidwell10). Podemos establecer la misión y objetivos del proyecto a partir del análisis de los distintos **escenarios de usabilidad e interoperabilidad** que condicionan la cadena de valor de los Actores. Qué aporta y qué recibe cada Actor del sistema en discusión. El Actor principal es el que tiene su cadena de valor más alineada con la misión del proyecto.

 © Derechos reservados http://creativecommons.org/licenses/by-nc/3.0/deed.es

8	**Si el software es la respuesta, cuáles son las preguntas**	**1.- ¿Qué?** Identificar las tareas a realizar, o bien, el resultado que pretendemos en forma de un "Entregable" del proyecto. **2.- ¿Para qué?** Definir el propósito que persigue la tarea o el entregable objetivo. **3.- ¿Por qué?** Establecer la motivación que justifica el esfuerzo dedicado a dicho objetivo. **4.- ¿Quién?** Identificar el rol o roles de proyecto que han de participar en su elaboración con distintos niveles de responsabilidad. **5.- ¿Cuándo?** Señalar las condiciones espacio-tiempo-proceso del ciclo de vida: planificación, programación, realización y evaluación. **6.- ¿Cómo?** Determinar qué estándares, técnicas y procedimientos son los más adecuados para su realización. **7.- ¿Con qué?** Fijar los recursos habilitados para completar su propósito: personas, herramientas, tiempo y coste. **8.- ¿Sobre qué?** Indicar sobre qué esquema hay que realizar las tareas, o bien, elaborar los entregables: Plantillas de Procesos de Proyecto; Patrones de Análisis, Funcionalidad y Diseño; modelos de arquitectura; muestrario de un **repertorio de entregables de referencia**. **9.- ¿En base a qué?** Especificar los manuales de estilo que definen la organización y el nivel de detalle del contenido de los entregables; las normas de aplicación obligada y las alertas, recordatorios y orientaciones (ARO), a tener en cuenta en la toma de decisiones. **10.- ¿Qué hay que medir?** Qué criterios de granularidad hay que aplicar a una tarea o un entregable (su segmentación en unidades manejables), y cómo debemos medir su cumplimiento (Ej.- tiempo, personas y otros recursos).

 © Derechos reservados http://creativecommons.org/licenses/by-nc/3.0/deed.es

9	**Comprender visualmente el proceso y el producto**	Necesitamos combinar la notación visual UML con una metodología para saber: 1. Qué **aspectos esenciales** hay que modelar (desde un esbozo a un plano detallado). 2. Qué diagrama es el más **apropiado** para representar una vista del modelo (estructura y/o función). 3. En qué proceso de proyecto (Análisis, Diseño, Implementación, Testing, etc.), hay que realizar un determinado diagrama, y **quién participará** en su elaboración (Roles de proyecto). 4. Qué **escala de abstracción variable** y qué nivel de dedicación hay que aplicar a un diagrama en cada fase de proyecto (desde el estudio preliminar en adelante). 5. Cómo definimos un modelo a través de distintas **vistas de arquitectura**: estructura, procesos y Casos de Uso. 6. Cómo delimitamos **el alcance** y **la segmentación** de un proyecto en tiempo, coste, procesos y producto resultante.
10	**Transformar la realidad**	Con un proceso continuo de **micro-mejoras** en la calidad de nuestros procesos de desarrollo de software podemos conseguir: ▪ Documentación técnica más compacta, simple, usable y **re-usable**. ▪ Mayor **visibilidad** del impacto de nuestras decisiones. ▪ Coste-Beneficio del **esfuerzo de especificación/construcción** más eficaz. ▪ Funcionalidad **suficiente** y **alineada** con las necesidades de información reales de nuestros usuarios. ▪ Procedimientos de trabajo más **confortables**. ▪ Componentes de software que ofrecen servicios usables, seguros y fiables, **desacoplados** para ensamblar sin riesgo de colapso. ▪ **Tolerancia "cero"** a la ambigüedad conceptual. ▪ **Equilibrio** entre Riesgo, Calidad y Coste del proyecto.

 © Derechos reservados http://creativecommons.org/licenses/by-nc/3.0/deed.es

Manual de usuario del “Manifiesto”

No confundir este decálogo con los diez mandamientos. No se trata de ordenar sino de provocar una reflexión. Mirar hacia dentro, rebuscar en nuestra experiencia y, hacia fuera, examinar nuestra realidad.

Los puntos del “Manifiesto” representan patrones básicos de cambio y transformación en los dominios de la ingeniería del software y los procesos de negocio. Su orientación nos proporciona un esquema conceptual para evaluar nuestras limitaciones y oportunidades como equipo desde una perspectiva empírica.

Es importante compartir lo que consideramos más relevante o irrelevante de cada punto, si están todos los que son, falta o sobra alguien en la lista, su alcance y comprensibilidad, etc. Todas las metodologías buscan la mejor estrategia de acción para comprender la lógica de los acontecimientos, adaptarse a los cambios, anticiparse en el tiempo y en definitiva, transformar la realidad.

No hay que confundir un método o una técnica con una metodología. La diferencia está en su alcance y en sus objetivos. Un método/técnica, es un conjunto de reglas o de principios orientado a obtener un resultado. Nos describe cómo realizar una actuación, con qué propósito y con que herramientas. En cambio una metodología define qué hay que hacer, quién es quién, cuándo y cómo tiene que actuar y con quien interactuar dentro de un dominio espacio-tiempo-proceso.

En cualquier ámbito de ingeniería hay una fractura real entre los responsables de analizar y definir los problemas (necesidades), los expertos en proveer soluciones (tecnología) y los administradores del presupuesto (financiación). Las metodologías nacen como resultado de este conflicto. Su propósito es conseguir la solución más eficaz con los recursos disponibles y su motivación es establecer un tipo de “contrato social” entre todos los participantes en un proyecto para conciliar su riesgo, calidad y coste.

Aunque parezca una contradicción, la mejor metodología es la que no se ve. Cuando se logra este fenómeno, es porque un equipo de proyecto ha logrado interiorizar de tal manera sus hábitos de trabajo que es capaz de actuar sin estar pendiente de manuales ni perder el tiempo averiguando cuál es la dirección correcta. Sabe interpretar lo que sucede en su contexto y actúa en consecuencia como un equipo bien cohesionado.

 © Derechos reservados http://creativecommons.org/licenses/by-nc/3.0/deed.es

La inflación actual de propuestas metodológicas hace difícil distinguir qué es esencial y qué es accesorio en una metodología. Quizás lo más urgente cuando nos planteamos su utilización es evaluar si el esfuerzo requerido será "sostenible" y saber cuál será su impacto real en nuestra organización.

Toda metodología madura ofrece un "vocabulario controlado" bien definido y suministra unos recursos de ayuda a la elaboración de los entregables de un proyecto. Aporta a los profesionales una guía de buena práctica, basada en plantillas y muestrarios de entregables de proyecto, que permiten ampliar su experiencia, compartir su conocimiento y evolucionar dentro de un proceso continuado de micro-mejoras progresivas.

En la órbita de la notación UML encontramos múltiples metodologías: Unified Process, ICONIX, eXtreme Programming, Crystal, Agile Alliance, Agile & Iterative Development, Enterprise Architecture, Executable UML, Problem Frames, Analysis & Data Model Patterns, entre otras muchas más como UML-TRAD, que no pretende ser nada original, sólo ser de utilidad.

Todas las metodologías coinciden en priorizar las tareas de identificación de requerimientos y difieren en como manejar la transición de dichos requerimientos a modelos dinámicos (procesos y reglas de negocio), y modelos estáticos (Clases); hasta llegar a la definición y desarrollo de componentes de software dentro de una arquitectura bien desacoplada. Su esquema de referencia está centrado invariablemente en los "Casos de Uso"; un artefacto de modelado que sirve para empaquetar escenarios de usabilidad e interoperabilidad que dan valor a los "Actores" de las distintas formas de vida organizativa. Las diferencias más significativas entre las metodologías residen en su nivel de formalización en referencia a las pautas de dirección y control de un proyecto.

Planteamos el "Manifiesto" de una metodología abierta para el desarrollo de software desde una perspectiva ecléctica y empírica. Nos interesa profundizar en todo aquello que forma parte del tronco común y es aceptado por cualquier metodología. También queremos compartir nuestra experiencia para señalar cuándo y dónde hay que poner el mayor esfuerzo de modelado y cómo no perderse dentro de las múltiples fases y procesos, utilizando algo parecido a un GPS conceptual (sistema de posicionamiento global) en el desarrollo de un proyecto y en la administración de una cartera de proyectos.