

	ASD	AUP
Tiempo para cada iteración recomendado	4 a 8 semanas	Primeras iteraciones más tiempo que las demás.
Tamaño del equipo	Equipos pequeños 5 a 9 miembros	Todos los tamaños
Comunicación en el equipo	Informal. Cara a cara.	Por medio de comunicación concisa.
Tamaño del Proyecto	Proyectos pequeños	Todo tipo de proyectos
Involucración del cliente	En cada entrega el cliente dará su aportación	Participa activa del cliente en las 4 fases, aunque no necesario día a día.
	ASD	AUP
Documentación en el proyecto	Sólo documentación básica	Documentación de alto nivel, concisa y formación de vez en cuando.
Habilidades Especiales	Ciclo de Aprendizaje	Procesos Definidos

<p>Ventajas</p>	<ul style="list-style-type: none"> - Gracias a la colaboración y el aprendizaje, se mejoran mucho los procesos y se solucionan errores. - Muy tolerante al cambio. - Gestión del riesgo. 	<ul style="list-style-type: none"> - El personal sabe lo que esta haciendo: no obliga a conocer detalles (alto nivel) - Simplicidad: apuntes concisos. - Agilidad: procesos simplificados del RUP. - Centrarse en actividades de alto valor: esenciales para El desarrollo. - No fuerza a usar ninguna técnica o herramienta.
<p>Desventajas</p>	<ul style="list-style-type: none"> - Errores y cambios que no son detectados con anterioridad afectan la calidad del producto y su costo total. 	<ul style="list-style-type: none"> - El AUP es un producto muy pesado (simplificación de RUP), en relación a otros ágiles. - Como es un proceso simplificado, muchos desarrolladores eligen trabajar con RUP, por tener a disposición mas detalles en el proceso.

Crystal	DSDM	XP
Dependiendo del método de la familia.	80% solución en el 20% del tiempo.	1 a 6 semanas.
Dependiendo del método de la familia	2 a 6 personas. Pero puede haber varios equipos trabajando en distintos módulos.	Menos de 20 personas.
Informal. Cara a cara. Periodos de no interrupción. Osmótica: Misma sala.	Basado en documentación. Capacidad del equipo de tomar decisiones importantes sin esperar aprobación de alto nivel.	Informal. Cara a cara. Reuniones diarias de pie. Entre artefactos y personas(código autocomentado, pruebas unitarias)
Todo tipo de proyectos. Dependiendo del método de la familia	Todo tipo de proyectos. Pero en proyectos grandes deberá ser particionado.	Proyectos pequeños.
En cada entrega el cliente dará su aportación. No tiene porque estar accesible constantemente, pero si, al menos, semanalmente.	En cada entrega el cliente dará su aportación (entregas frecuentes)	Cliente involucrado. Comunicación fluída. El cliente decide prioridades y debe estar siempre disponible.
Crystal	DSDM	XP
Sólo documentación básica	Abundante	Sólo documentación básica. Código autodocumentado.
Familia compuesta por varios tipos que se adapta a todo tipo de proyectos y tamaño de equipos.	Prototipado	TDD, User Stories, Refactoring, Pair Programming

<p>Se ajusta a la criticidad o tipo de proyecto y al tamaño del equipo.</p>	<ul style="list-style-type: none"> - Permite alcanzar nivel 2 CMMI. - Uso de prototipos para obtener requisitos. - Todos los cambios durante el desarrollo son reversibles - Los requisitos están especificados a un alto nivel - El testing está Integrado a través del ciclo de vida. - Desarrollo rápido. 	<ul style="list-style-type: none"> - El cliente es parte del equipo. - Feedback constante. - Mejores prácticas bien definidas. - Simplicidad. - Potencia la mejora continua. - Da prioridad a la acción. - Propiedad colectiva del código
<p>Coordinación difícil de equipos grandes.</p>	<ul style="list-style-type: none"> - Documentación abundante. - Los equipos de DSDM deben tener el poder de tomar decisiones. 	<ul style="list-style-type: none"> - Documentación pobre. - Puede llevar a poca disciplina. - La reflexión puede quedar demasiado relegada frente a la acción. - Presencia obligatoria e importante del cliente. - Hacer las cosas simples requiere experiencia. - Requiere coraje para asumir ciertas prácticas.

FDD	LSD	Kanban
2 días a 2 semanas.	Iteraciones cortas.	Flujo constante. Se liberan entregables en base a eventos o necesidades
Admite muchos miembros.	Todos los tamaños	Sin prescripción.
Basado en documentación.	Por medio de comunicación concisa.	Informal. Cara a cara. A través de tablero Kanban. Cada equipo define sus procesos.
Proyectos complejos y críticos.	Todo tipo de proyectos	Proyectos de corta-media duración.
En cada entrega el cliente dará su aportación.	En cada entrega el cliente dará su aportación (entregas frecuentes)	Sería conveniente una participación activa del cliente.
FDD	LSD	Kanban
Existe documentación. La mínima necesaria para que alguien nuevo entienda el sistema.	Sólo documentar lo estrictamente necesario.	Evita la documentación. Cada equipo define sus procesos.
Diagramas UML.	Muda, Mura y Muri	Tablero Kanban.

<ul style="list-style-type: none"> - Monitoreo constante del avance del proyecto. - Énfasis en la calidad. - Ofrece soluciones a situaciones de riesgo (cambios en requisitos, pérdidas de tiempo....) 	<ul style="list-style-type: none"> - Mejora la calidad. - Aumenta la productividad. - Elimina lo que no produzca valor para el cliente. 	<ul style="list-style-type: none"> - Facilita el que todo el mundo sepa que hacer, quien hace que y como va el proyecto. - Limita el número de tareas a hacer al mismo tiempo. Control del flujo de trabajo. - Fácil de aplicar. - Flexible ante cambios.
<ul style="list-style-type: none"> - Propiedad individual del código. - No cubre todo el ciclo de vida. Necesita de otras metodologías que la complementen. 	<ul style="list-style-type: none"> - Requiere equipo experimentado y responsable en su propio trabajo. - Requiere constantes esfuerzos de revisión y análisis de procesos. 	<ul style="list-style-type: none"> - Se deben definir las fases del ciclo de trabajo. - No define roles, ni fases, ni tampoco profundiza en el tablero Kanban. - Algunos lo ven más como una técnica que como metodología.

Scrum	Scrumban
2 a 4 semanas.	Flujo constante. Se liberan entregables en base a eventos o necesidades
Todos los tamaños (Scrum de Scrums)	No hay ScrumMaster. Equipos pequeños.
Informal. Reuniones diarias de pie. Reuniones de retrospectiva.	Informal. Reuniones diarias de pie. A través de tablero Kanban. Reuniones de retrospectiva.
Todo tipo de proyectos.	Proyectos de corta-media duración.
Cliente fuertemente involucrado a través del rol de ProductOwner.	Cliente involucrado. Comunicación fluida.
Scrum	Scrumban
Sólo documentación básica.	Sólo documentación básica.
Sprint, Product y Sprint BackLog, Scrum Board, Scrum Master, Planning Poker.	Lo mejor de Scrum y de Kanban

<ul style="list-style-type: none"> - Equipo autoorganizado. - El equipo sabe lo que tiene que hacer cada día. - El cliente sabe lo que se le entrega en cada sprint. - Flexible a cambios. - Desarrolladores tienen autonomía. - Minimiza el trabajo de gestión. - Minimiza el síndrome del estudiante. 	<p>Todas las de Kanban, más todo lo que aportan las reuniones de Scrum.</p>
<ul style="list-style-type: none"> - Puede producir stress al sentirse el equipo en un continuo sprint. - Requiere un equipo formado, motivado y con cierta experiencia. - Necesidad de involucración del cliente. - Problemas en equipos distribuidos geográficamente. 	<ul style="list-style-type: none"> - Algunos lo ven como Scrum sólo que usando un tablero Kanban. - No hay una clara definición de roles y fases.