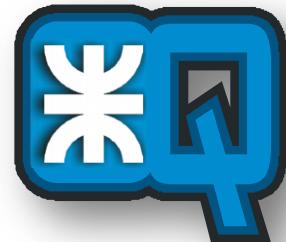


Sprint 0: Inicio

- Mgtr. Diego Rubio
- Mgtr. Natalia Andriano
- Ing. Juan Pablo Bruno
- Ing. Mauricio Silclir

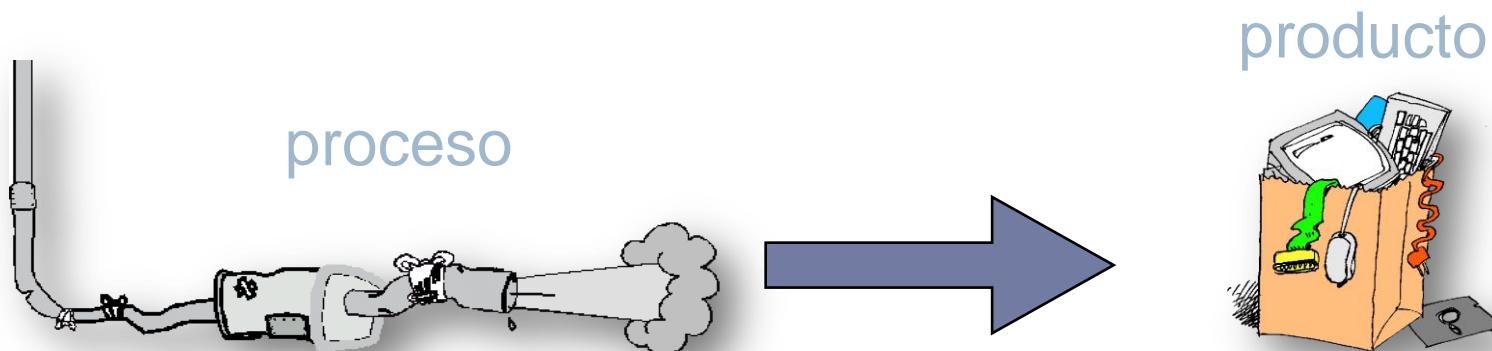


Objetivos específicos

- Desarrollar en el estudiante una comprensión de la importancia de las metodologías ágiles en la ingeniería de software.
- Brindar herramientas al estudiante para que ayuden a la toma de decisiones sobre cuándo es factible aplicar éstas metodologías
- Proveer al estudiante una herramienta que le permita seleccionar el método ágil apropiado para su organización o proyecto.
- Brindar al estudiante conocimiento relacionado a los diferentes métodos ágiles existentes (SCRUM, Lean, XP).

Proceso

- La calidad del software está fuertemente determinada por la calidad del proceso usado para desarrollarlo y mantenerlo.



Proceso

- Conjunto de herramientas, métodos, y prácticas que usamos para construir un producto *[Humphrey, Managing the Software Process]*
- Secuencia de pasos encadenados con el fin de construir un producto u ofrecer un servicio
- (Del lat. *processus*). *[RAE]*
 - 1. m. Acción de ir hacia adelante.
 - 2. m. Transcurso del tiempo.
 - 3. m. Conjunto de las fases sucesivas de un fenómeno natural o de una operación artificial.

¿Qué es el Proceso de Desarrollo de Software?

Es la aplicación de un método para el desarrollo de software por parte de las **personas** con el **entrenamiento** adecuado.

Proceso sin definir



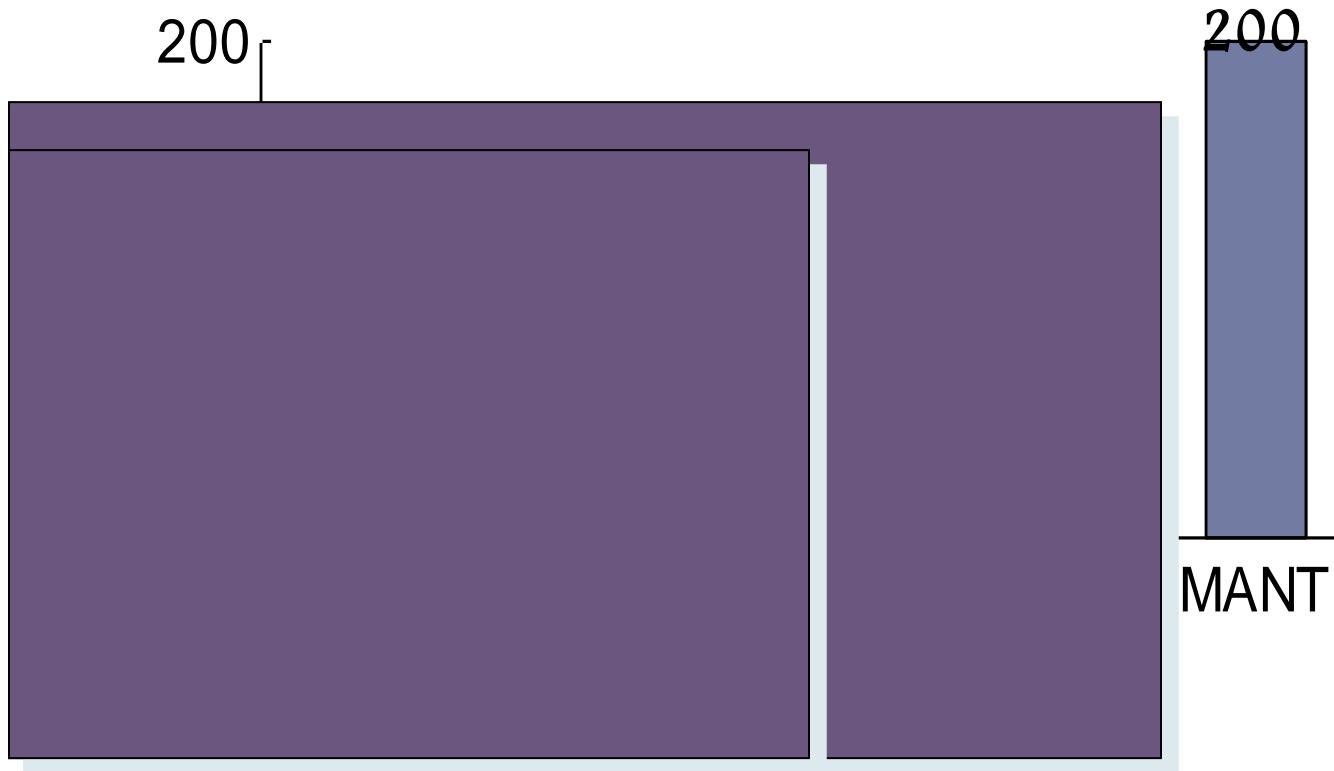
¿Qué ocurre cuando el proceso no está definido?



‘Los participantes harán lo que puedan para resolverlo’

Qué pasa si el proceso no está definido

Costo de detección de errores



¿Qué es el Proceso de Desarrollo de Software?

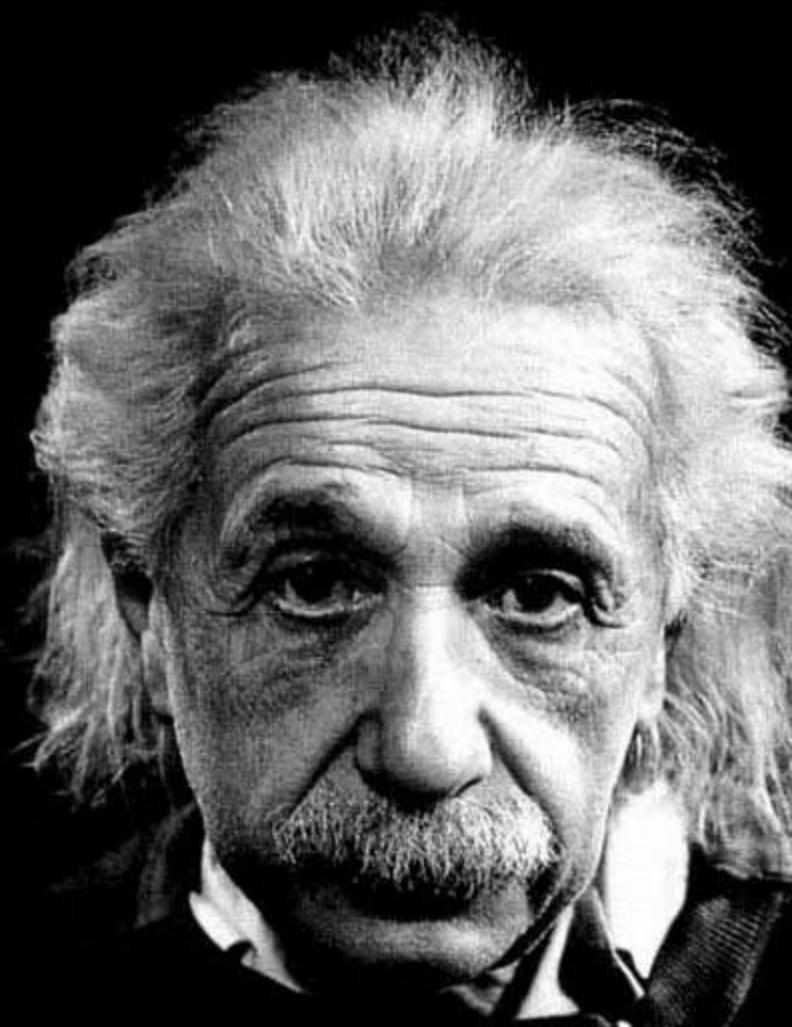
- No debemos confundir la definición de un proceso con el proceso real:
 - La definición del proceso es la expresión de como deseamos trabajar
 - El proceso real es el modo en que efectivamente hacemos el trabajo

El proceso real es la ejecución del proceso definido

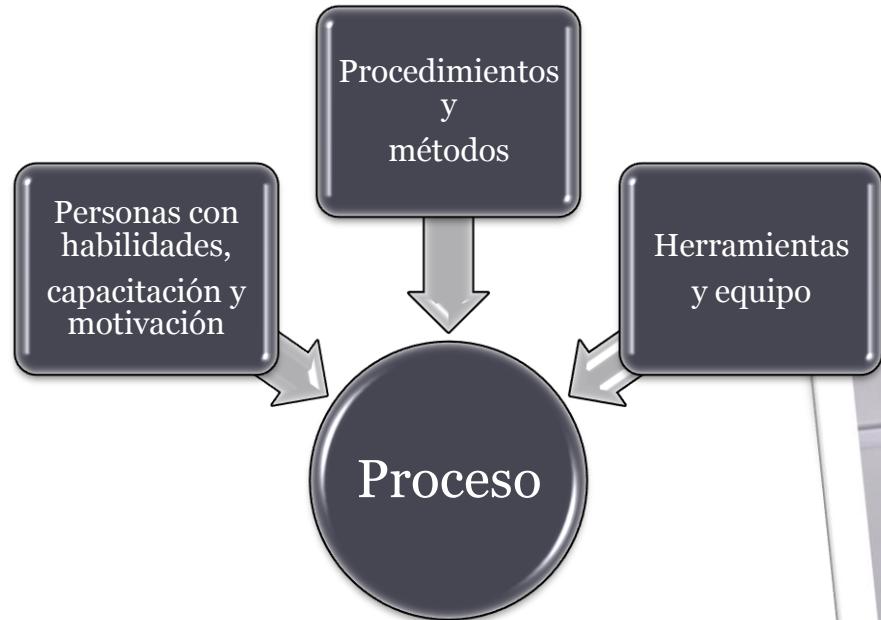
¿Cómo lo definimos?

“Everything should be made
as simple as possible,
but not simpler.”

Albert Einstein



El diseño del proceso...una analogía

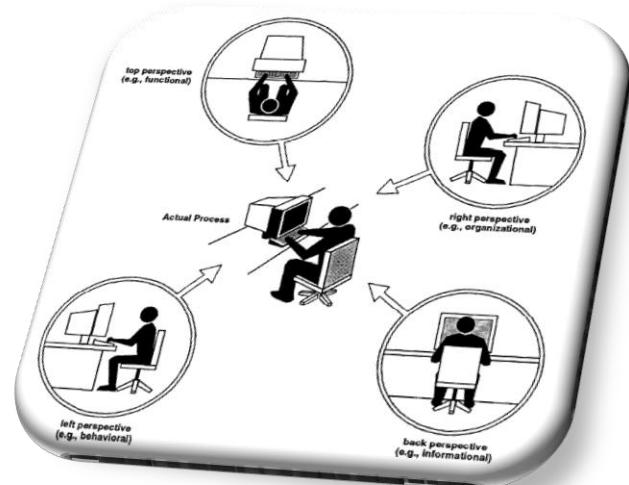


Ref: Don Norman, Ph.D. Author of "The Design of Everyday Things"

Cómo describimos el proceso (1/2)

Perspectivas

- Funcional
 - Actividades y flujo de información
- De comportamiento
 - Cuándo y cómo
 - Secuencias, iteraciones, etc.
- Organizacional
 - Roles
- De información
 - Productos de trabajo



Cómo describimos el proceso (2/2)

Lenguajes

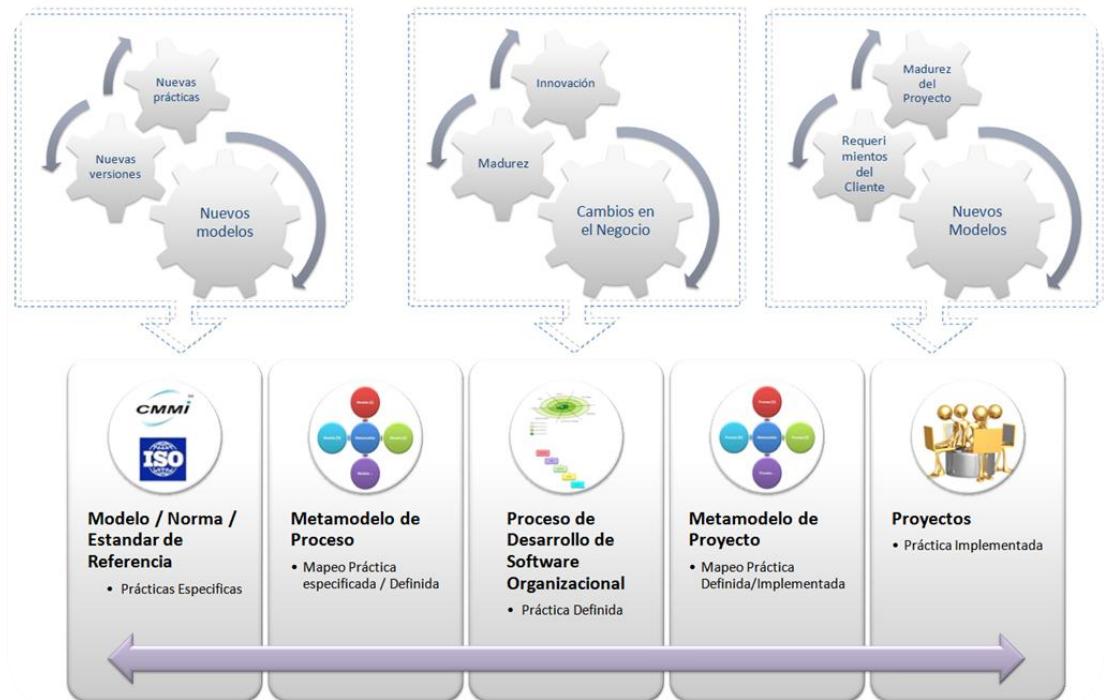
- SPEM (meta)
- UML
- ETVX
- BPMN

Herramientas

- EPF
- RTC
- TFS
- Word, excel
- Xml
- Otras...

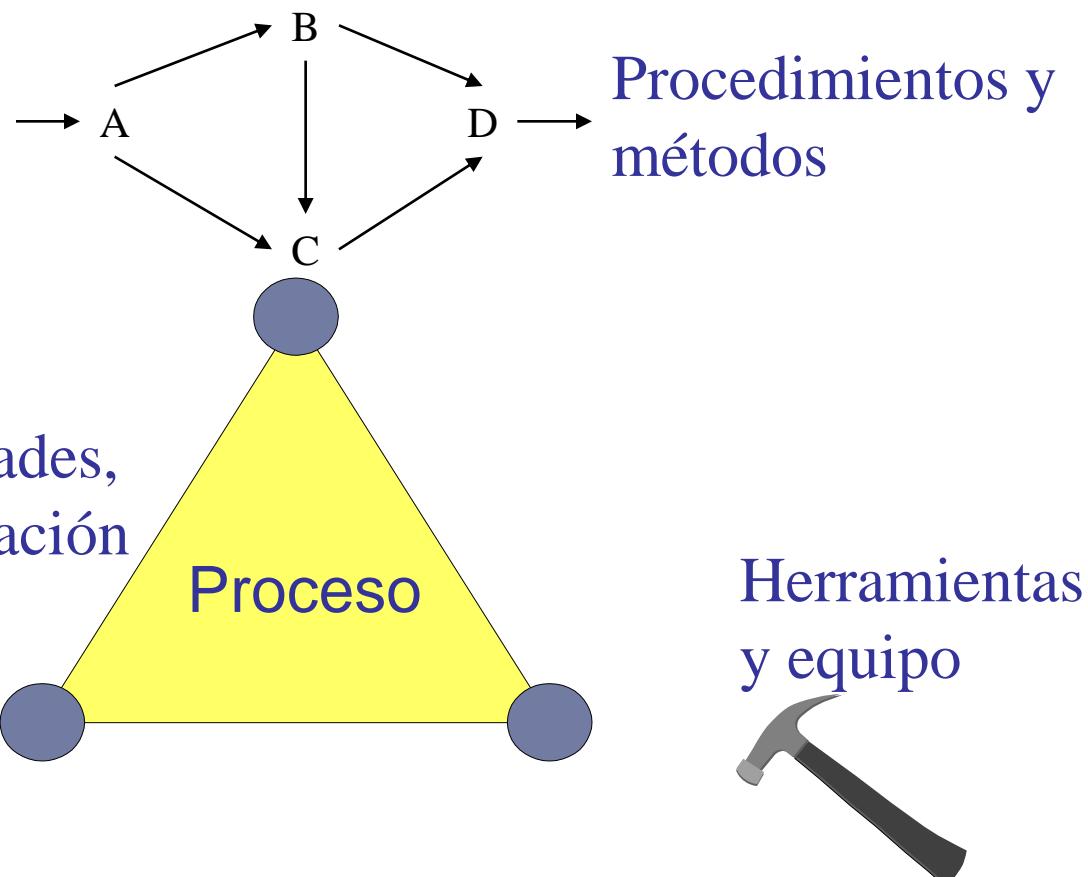
¿Cuál usamos?

- Depende! (Ref: Silclir)

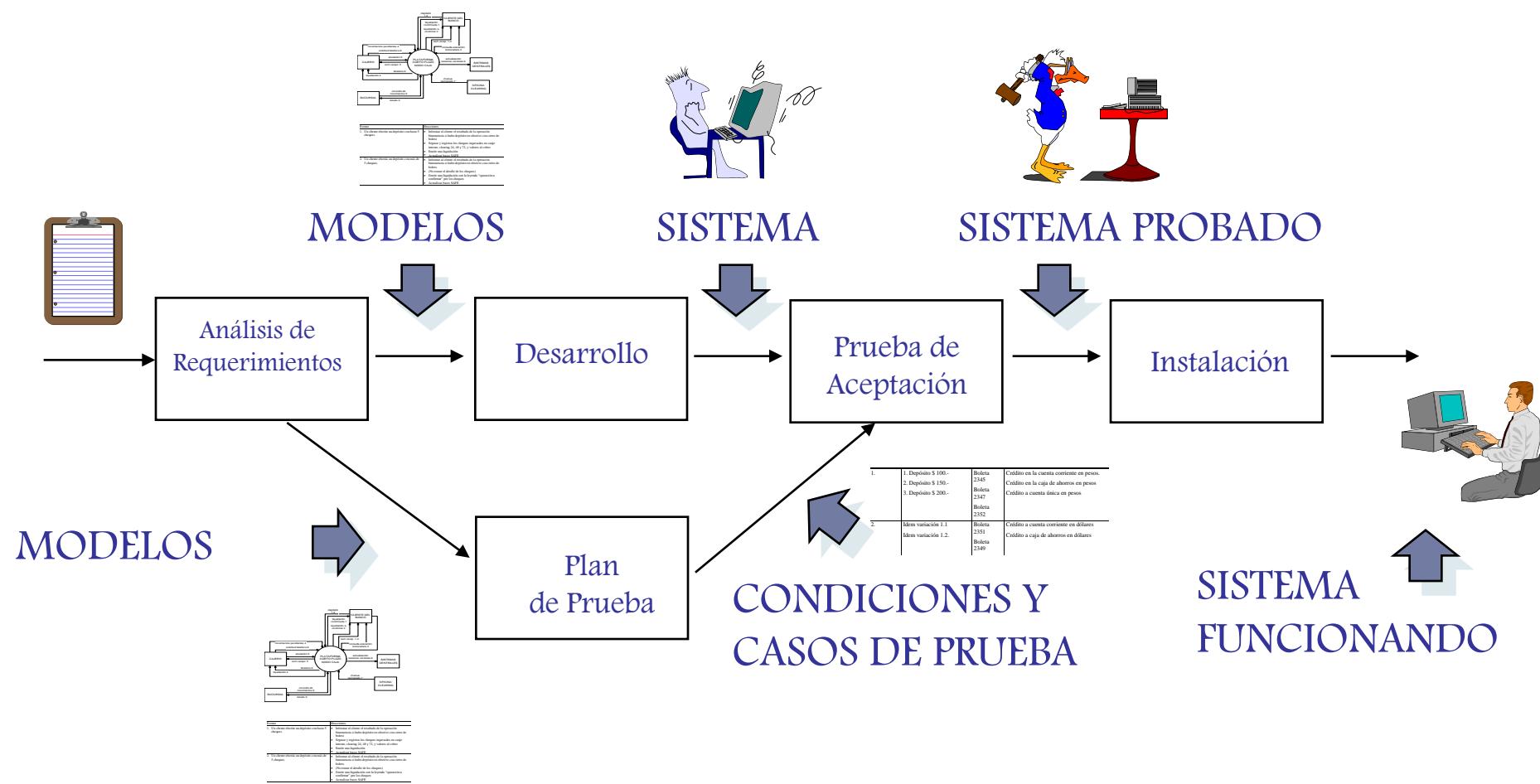


Proceso: Uniendo las piezas

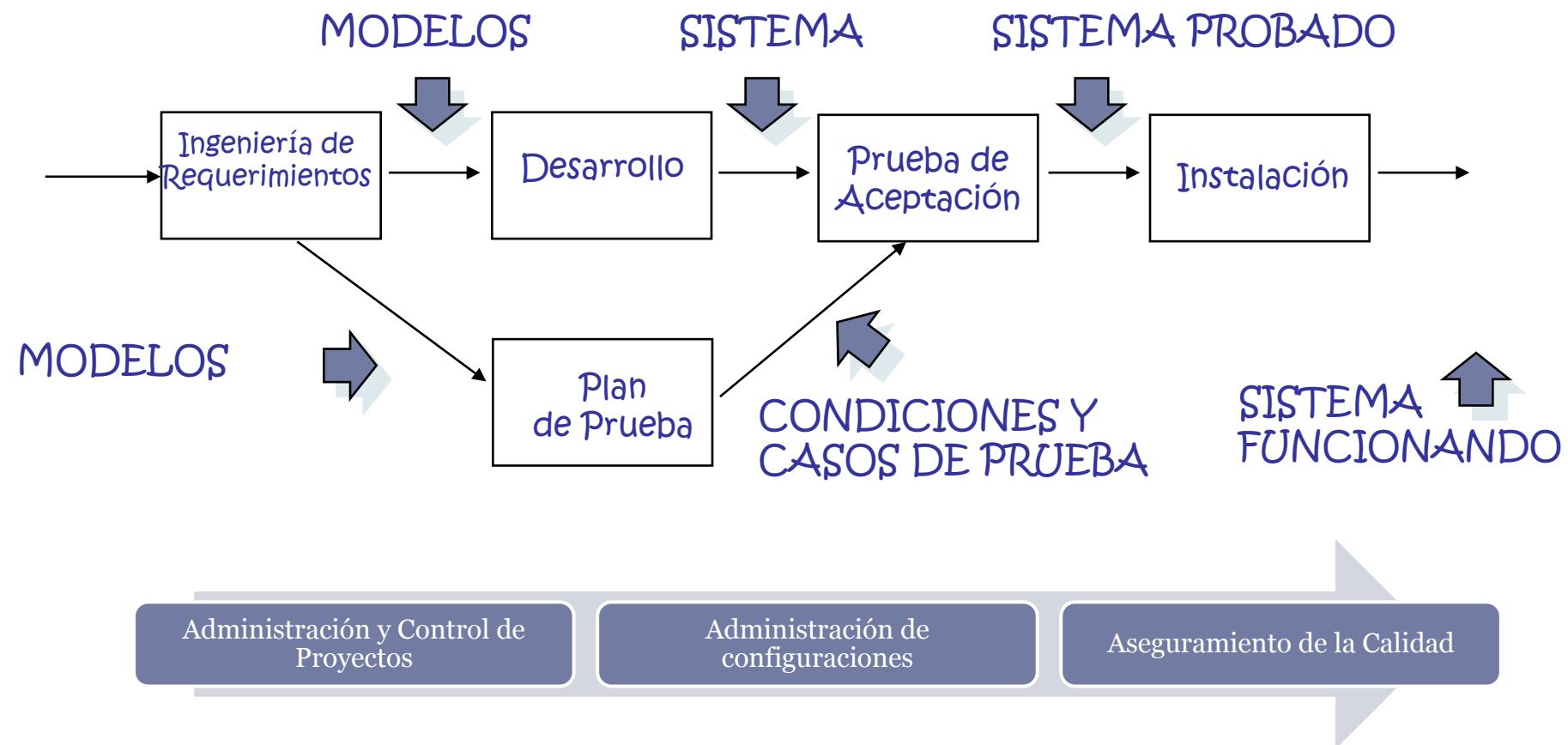
El proceso real es la ejecución del proceso definido



¿Cómo es un proceso para un proyecto?



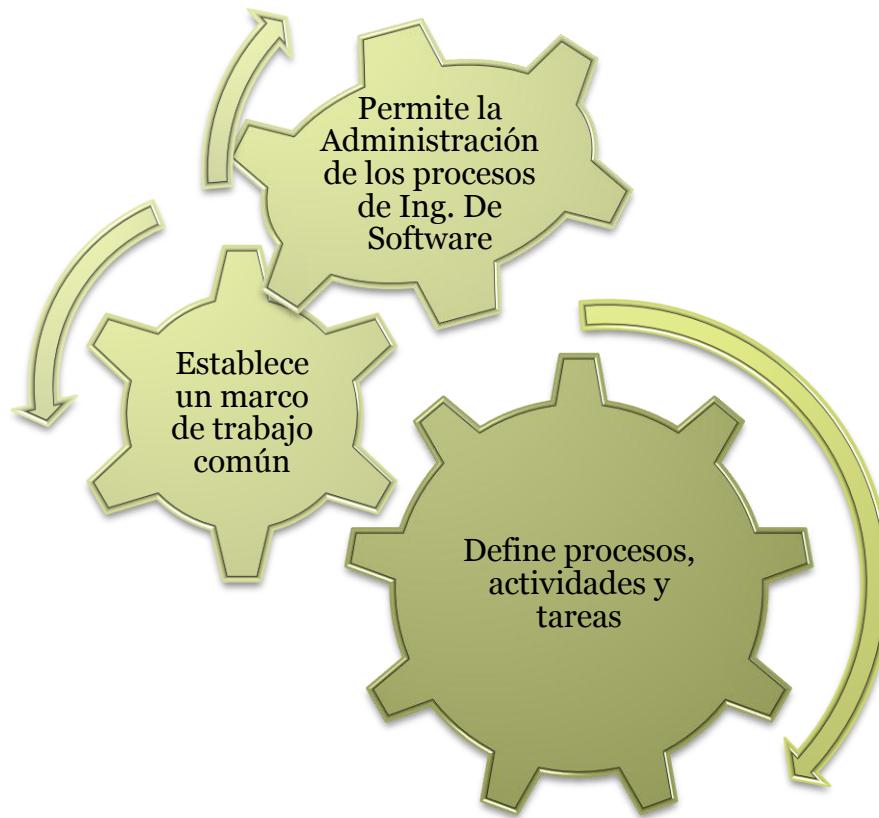
Soporte al Proceso



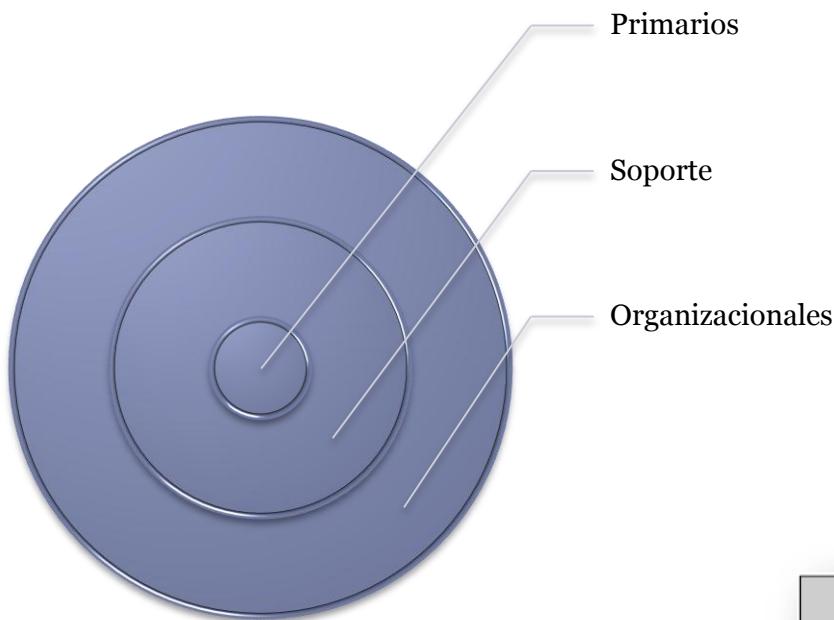
¿Cómo empezamos?



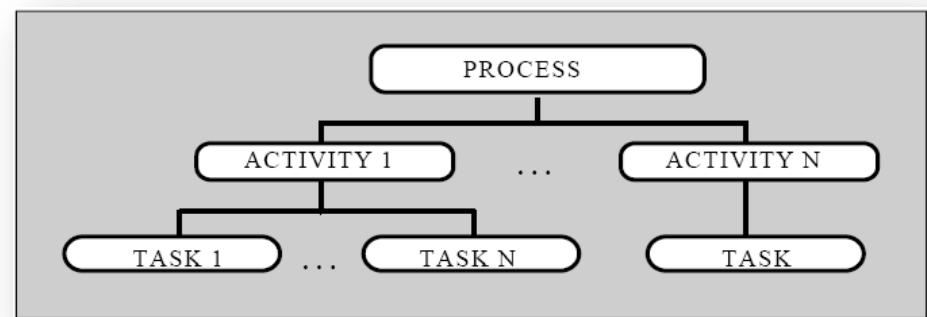
ISO 12207: Propósito



ISO 12207: Conceptos principales



- Estructura de los procesos

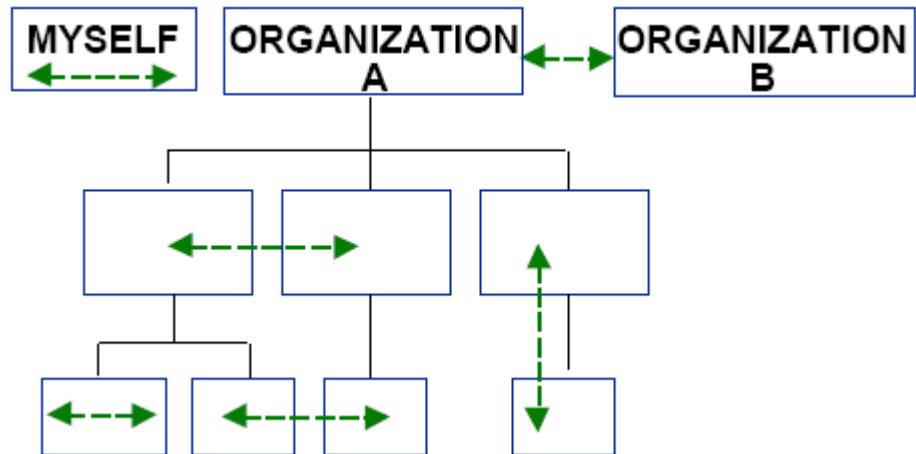


ISO 12207: Aplicación

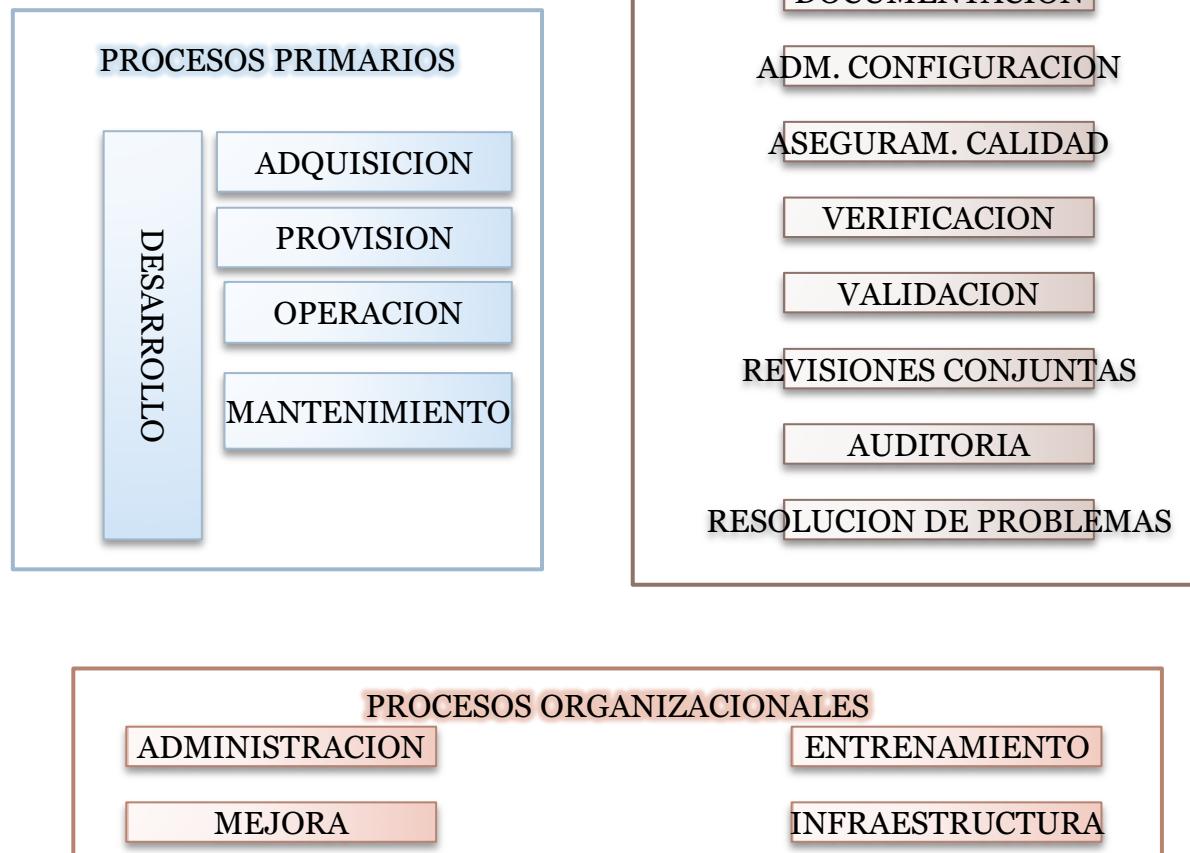
- **Niveles de Aplicación:**

- Por una persona, como un estándar auto impuesto.
- Por una organización internamente.
- Entre personas dentro de una misma organización.
- Entre dos organizaciones.

- Las actividades y tareas descriptas en la norma expresan el “QUE” y no el “COMO”
 - Puede ser utilizado con cualquier modelo de ciclo de vida (cascada, incremental, evolutivo, espiral, etc).
 - Cualquier metodología de ingeniería de software (diseño orientado a objetos, estructurado, etc)
 - Cualquier lenguaje de programación (Ada, assembly, Java, etc)



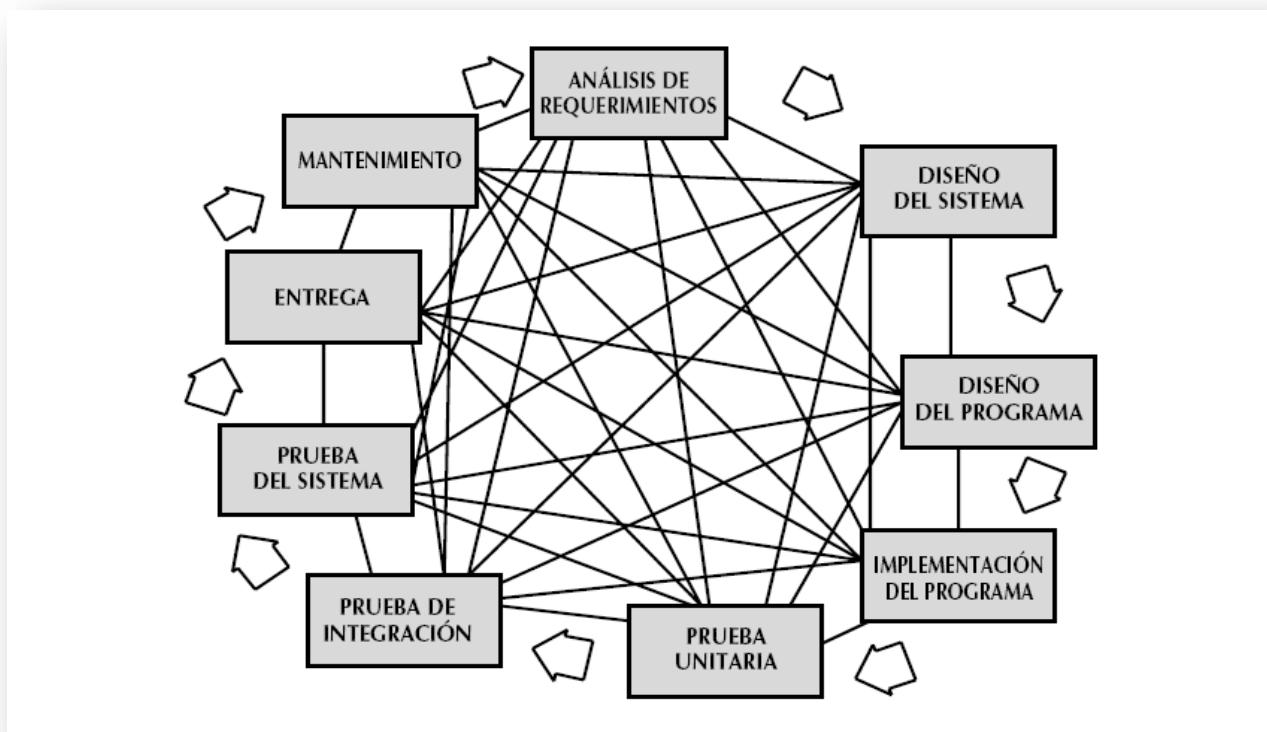
ISO 12207: Procesos del Ciclo de Vida del Software



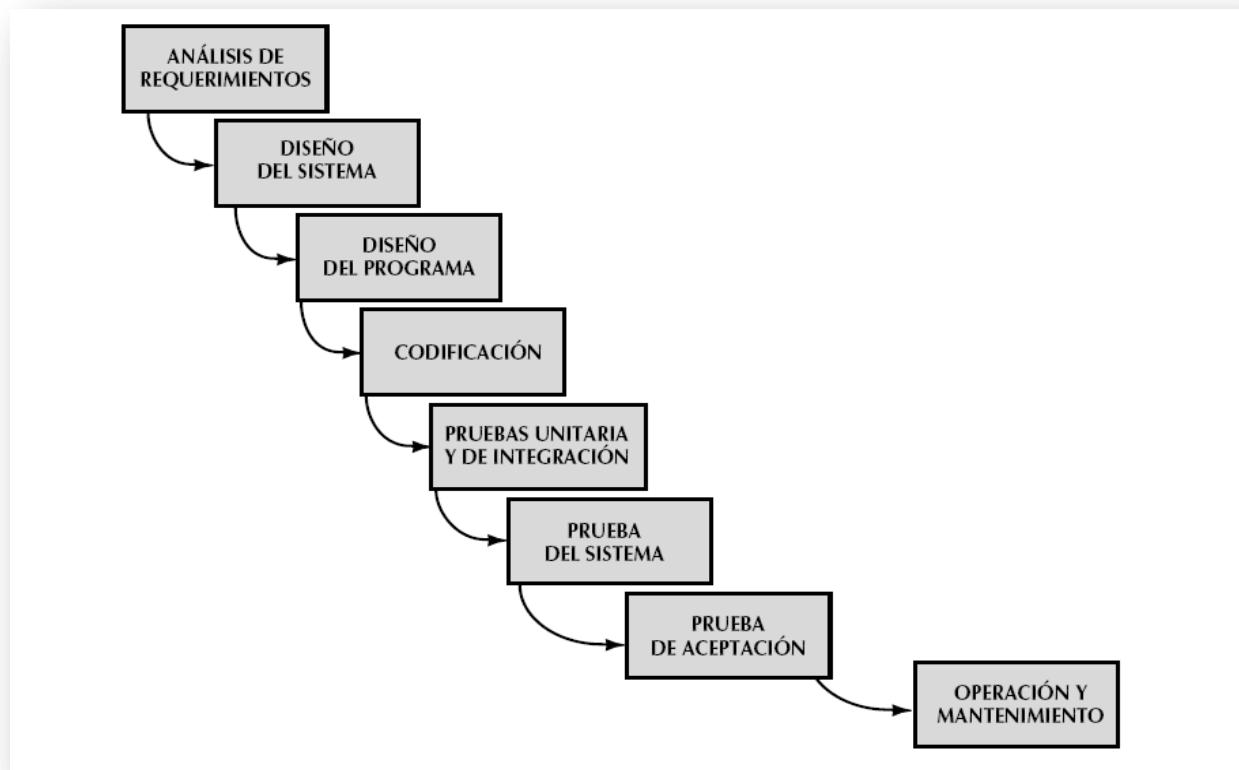
Ciclos de Vida

Distintos Modelos de procesos

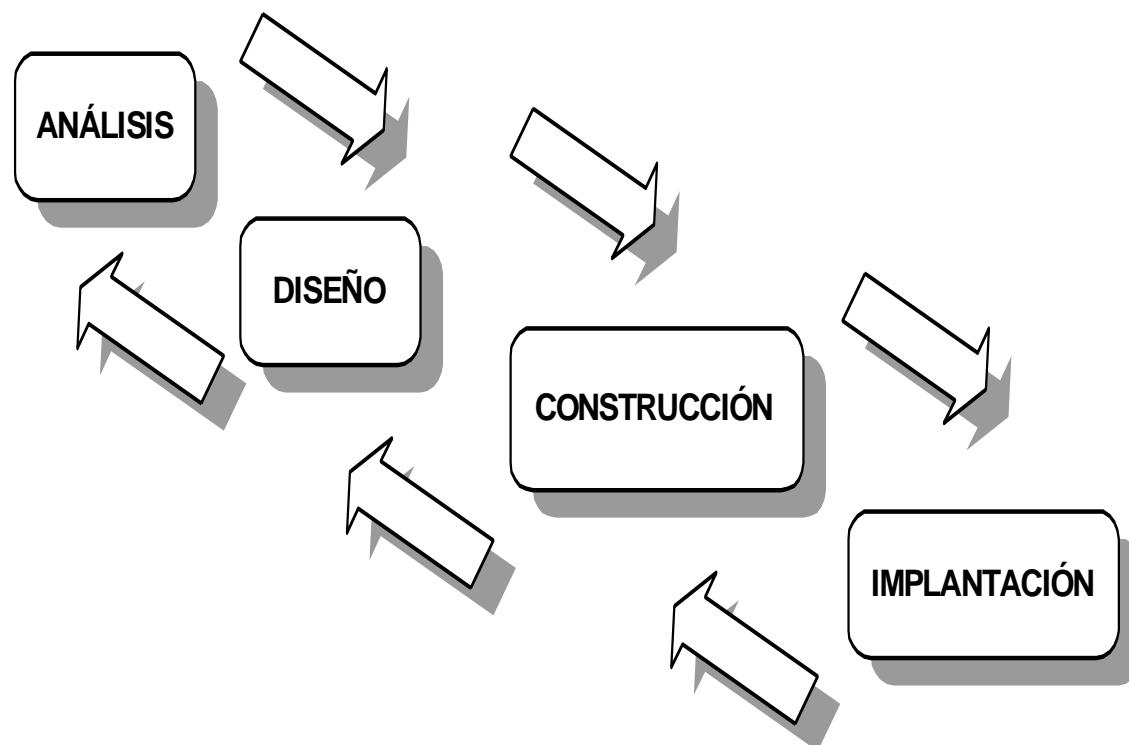
Modelo ‘Real’



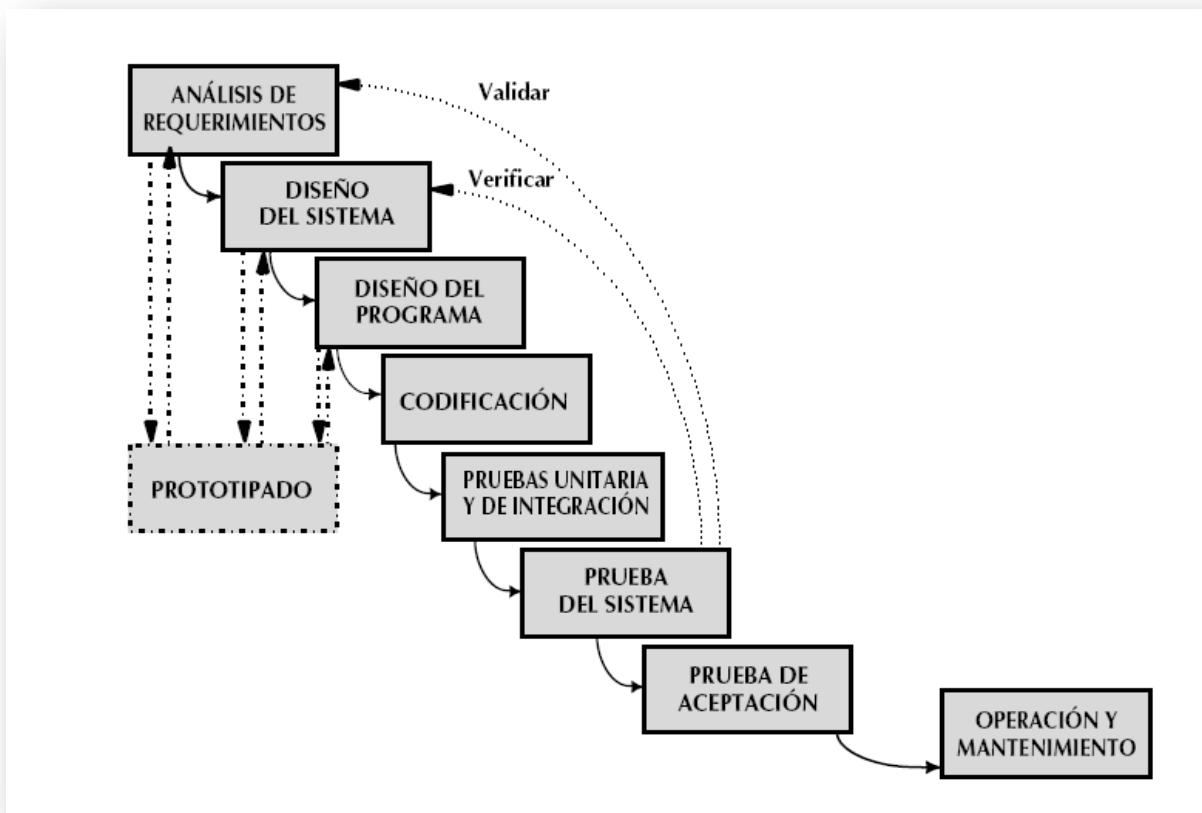
Modelo en Cascada



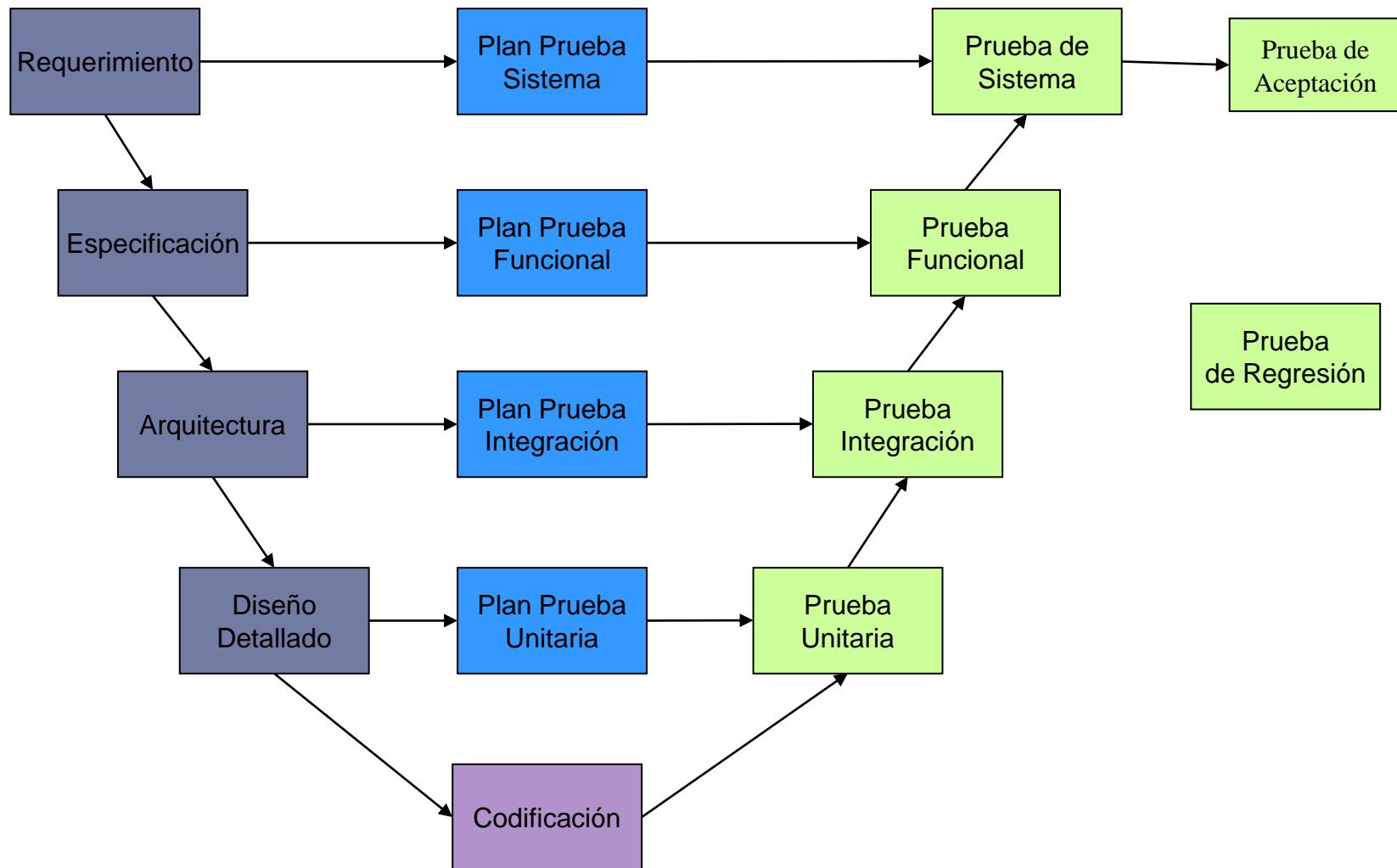
Modelo en Cascada



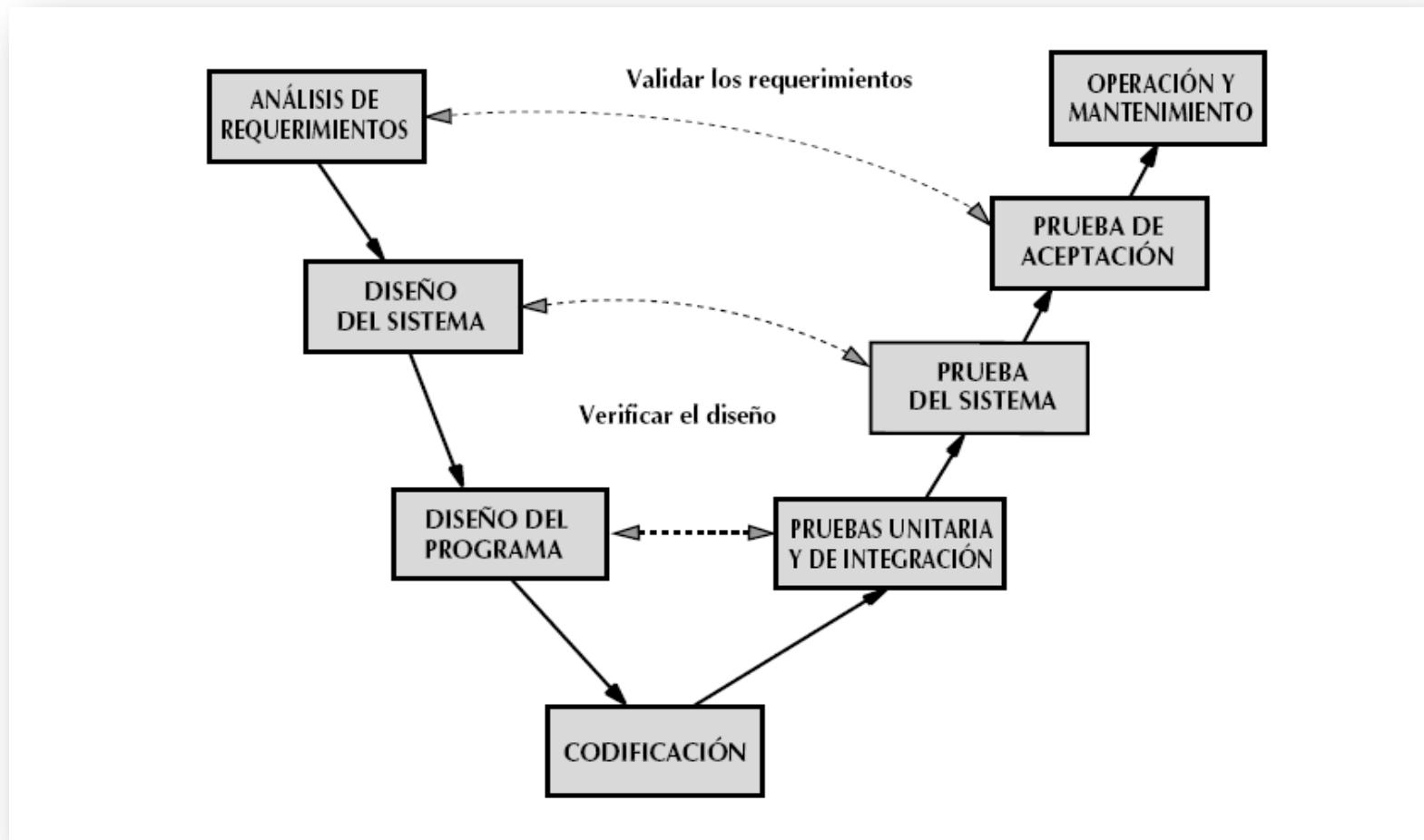
Modelo en Cascada con Prototipo



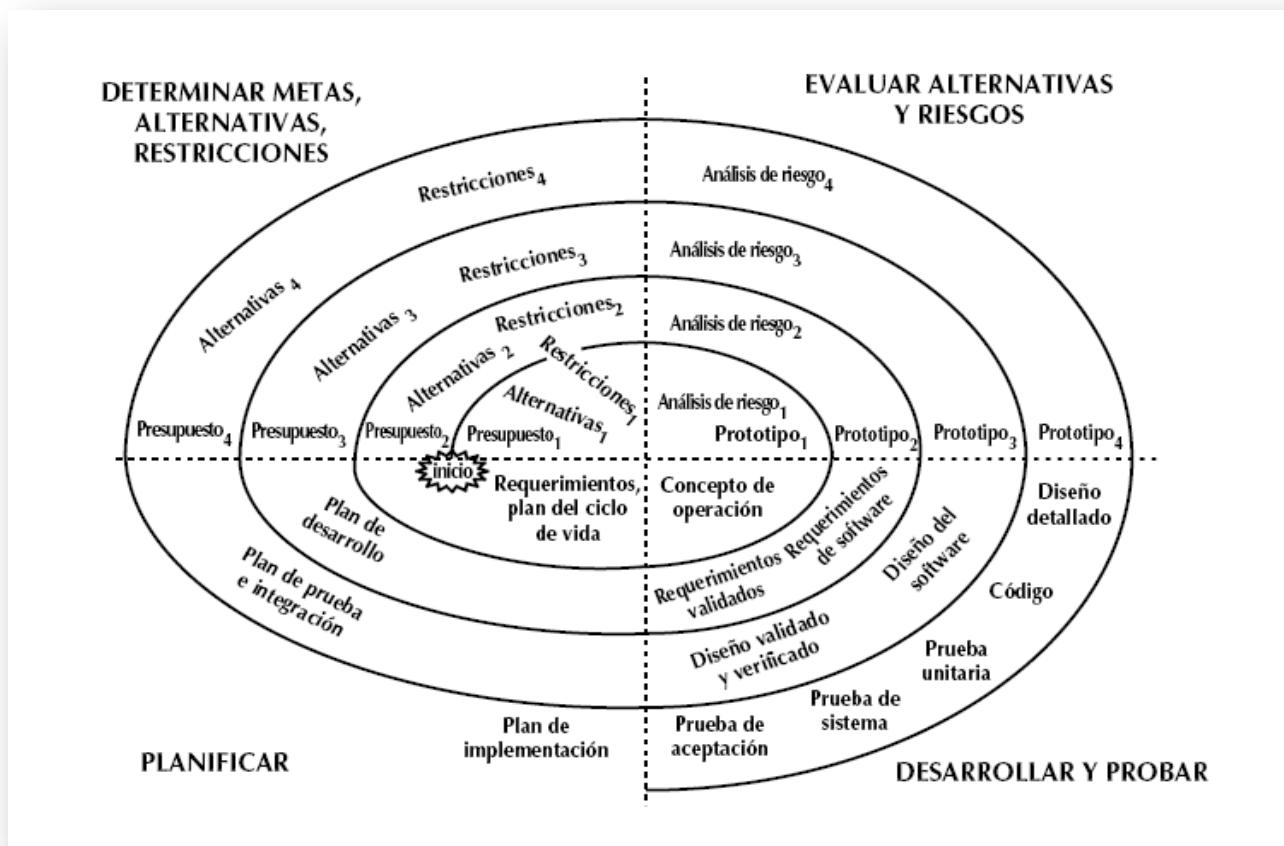
Modelo en V



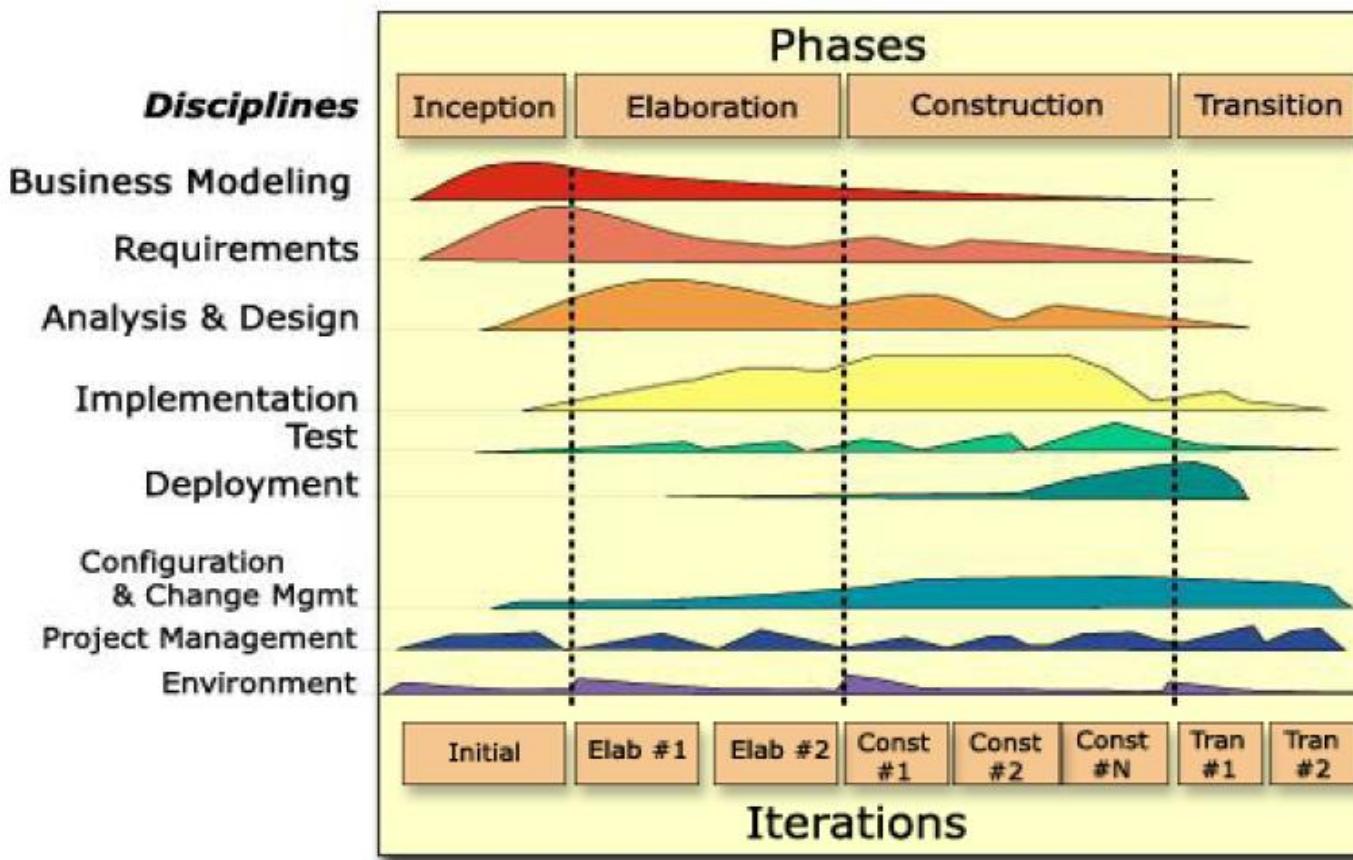
Modelo en V



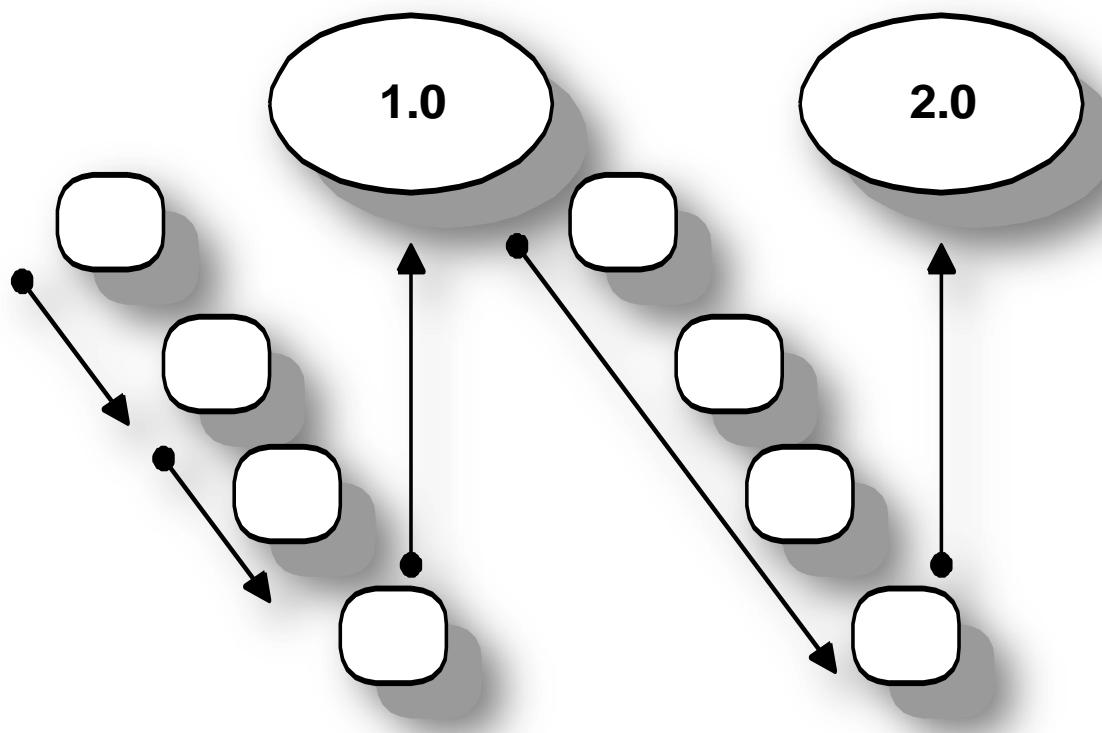
Espiral; Boehm



Rational Unified Process

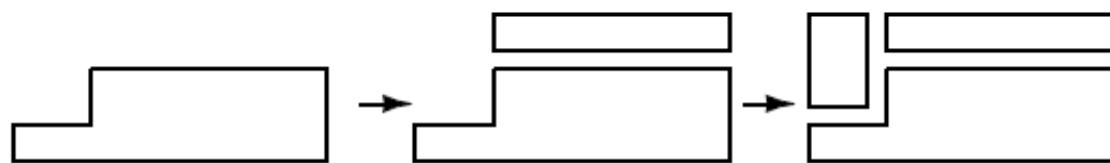


¿Incremental o Iterativo?

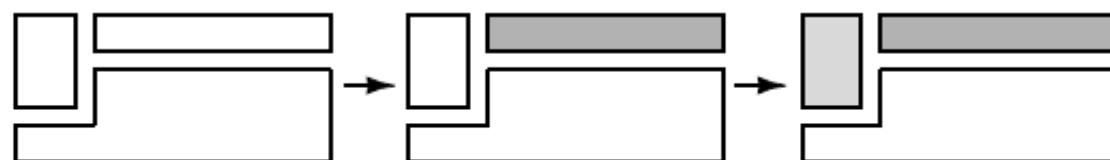


Incremental vs Iterativo

DESARROLLO INCREMENTAL



DESARROLLO ITERATIVO



Actividad en clase - Discusión:

¿En qué casos se considera más apropiado la utilización de cada una de estas metodologías de las metodologías “no-ágiles”?

Entendiendo el Agile Manifesto

Estamos descubriendo formas mejores de desarrollar software tanto por nuestra propia experiencia como ayudando a terceros. A través de este trabajo hemos aprendido a valorar:

Individuos e interacciones sobre procesos y herramientas
Software funcionando sobre documentación extensiva
Colaboración con el cliente sobre negociación contractual
Respuesta ante el cambio sobre seguir un plan

Esto es, aunque valoramos los elementos de la derecha, valoramos más los de la izquierda.

Kent Beck
Mike Beedle
Arie van Bennekum
Alistair Cockburn
Ward Cunningham
Martin Fowler

James Grenning
Jim Highsmith
Andrew Hunt
Ron Jeffries
Jon Kern
Brian Marick

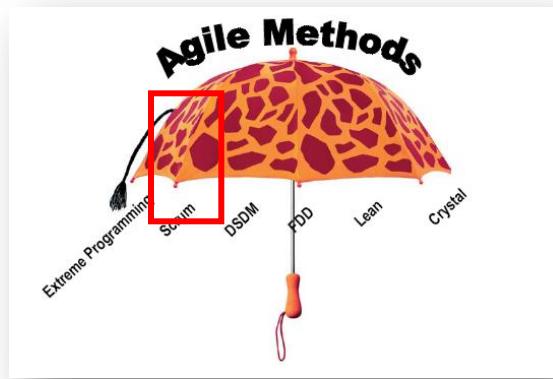
Robert C. Martin
Steve Mellor
Ken Schwaber
Jeff Sutherland
Dave Thomas

Acerca de (Agile)



Métodos Ágiles

- Agile describe una **aproximación a un método**, no el método en sí mismo
- Debajo del paraguas Agile, existen varios **métodos más específicos**, como Extreme Programming, Scrum, Lean Development, etc.



- Cada una de estas metodologías persigue **los mismos principios genéricos** (comunicación mejorada, software funcional, colaboración y participación del cliente, feedback, simplicidad, respuesta a cambios)

Actividad en clase - Discusión:

Encontrar similitudes y diferencias entre las metodologías ágiles y otros modelos de ciclo de vida. ¿Agrega valor este tipo de discusiones?

Relación con modelos y estándares de calidad existentes (CMMI/ISO).

Actividad en clase - Discusión:

¿Es factible integrar estas metodologías en organizaciones que necesiten certificar estándares (ISO9000, TL9000) o modelos internacionalmente reconocidos (CMMI)?

Scrum

¿Qué es Scrum?

- Es una enfoque **AGIL** para la gestión de un proyecto. Más que una metodología o proceso, es un Marco de Trabajo
- Utiliza procesos **ITERATIVO/INCREMENTALES**
- Orientado a **RESULTADOS Y COMPROMISOS**
- No está restringido a proyectos de software solamente
- Tiene una implementación **SIMPLE**, pero no es una “bala de plata”
(silver bullet)
- Su visión es opuesta a la propuesta por la metodología en cascada ...

Opuesta a la metodología en Cascada...

Requerimientos

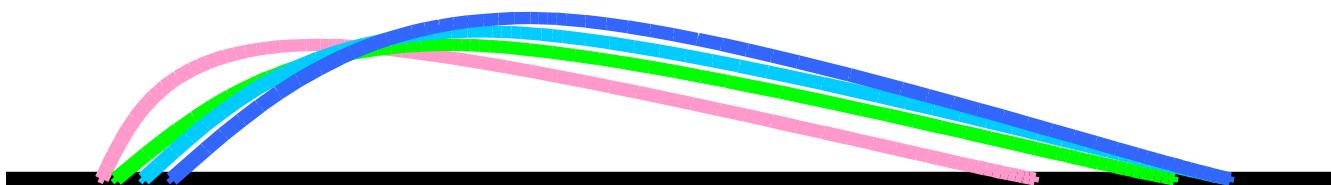
Diseño

Código

Prueba

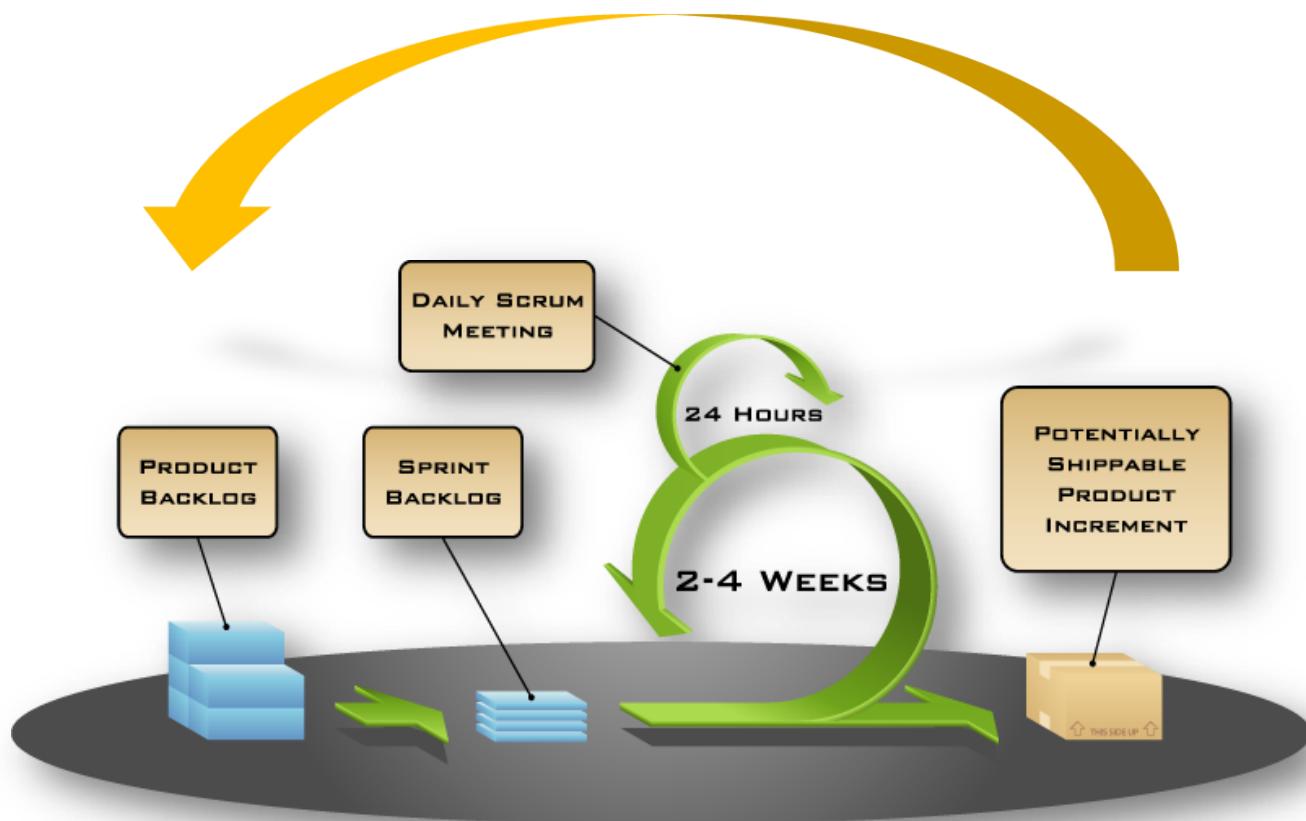
En vez de hacer una sola cosa a la vez...

...los equipos Scrum hacen un poco de todo, todo el tiempo



Ref: "The New New Product Development Game" by Takeuchi and Nonaka. *Harvard Business Review*, January 1986.

Ciclo de vida Scrum



Criterios Básicos de Scrum



Equipos pequeños, de entre 5 y 8 personas



Ubicados en el mismo lugar *físico* de trabajo

Dueño del Producto disponible (en casi cualquier momento!)



El Dueño del Producto NO ES el Scrum Master



Scrum Master preferentemente ubicado en el mismo lugar que el equipo de trabajo

Tres criterios básicos: **disciplina, disciplina y disciplina**



Prácticas Fuertemente Recomendadas

- Automatización de pruebas unitarias
- Automatización de pruebas de sistema
- Herramientas integradas (por ejemplo, para mantener trazabilidad entre ítems)
- Integración Continua (builds automatizados, ejecución de pruebas, análisis estático de código, colección de métricas, etc)
- Aprendizaje implícito
- Documentación implícita



Marco de Trabajo Scrum



Roles y responsabilidades en Scrum



Dueño del Producto

- Define las funcionalidades, decide la fecha de release y su contenido
- Prioriza el Backlog de Producto
- Puede cambiar las funcionalidades y prioridades en cada sprint
- Acepta o rechaza los resultados del sprint
- Participa en las reuniones de Planificación de Sprint y revisión



Scrum Master

- Asegura que el equipo sea completamente funcional y productivo
- Promueve la cooperación entre todos los roles y remueve barreras
- Protege al equipo de interferencias externas
- Asegura que el proceso sea cumplimentado
- Participa en las reuniones diarias, de planificación de sprint y revisión del mismo



Equipo

- Típicamente de 5-9 personas
- Selecciona el Backlog de Sprint
- Tiene el derecho de hacer lo que sea dentro de los límites de las guías del proyecto para alcanzar el objetivo del sprint
- Se organiza a sí mismo y a su trabajo
- Demuestra los resultados del trabajo al Dueño del Producto

Marco de Trabajo Scrum



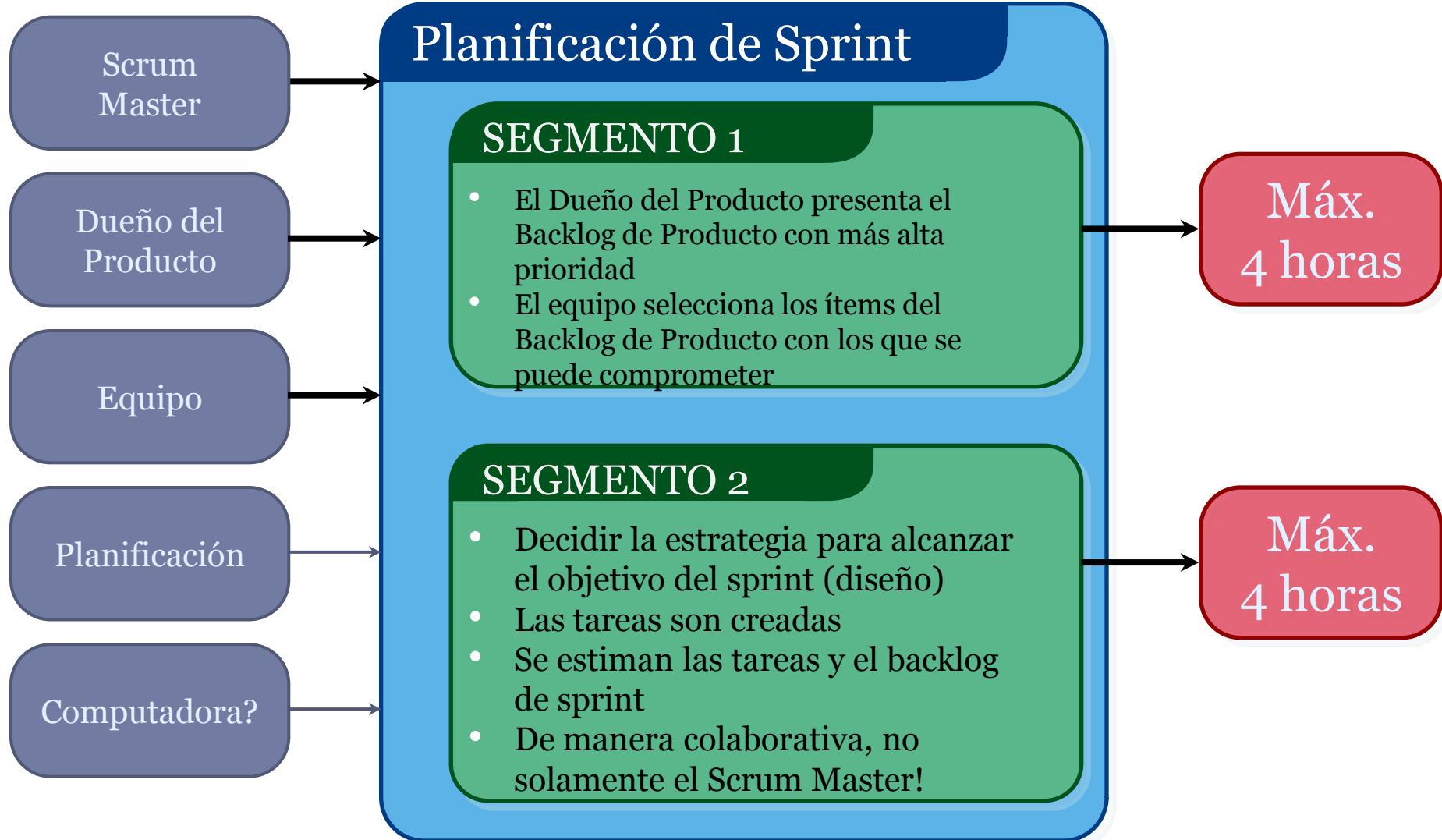
Ceremonias

- Planificación del Sprint
- Reunión Diaria
- Revisión del Sprint
- Retrospectiva del Sprint

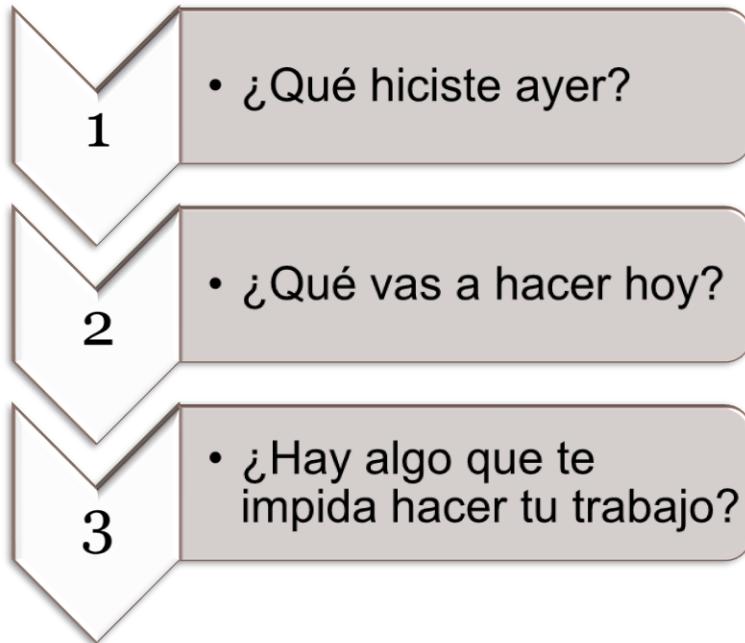
Artefactos

- Backlog de Producto
- Backlog de Sprint
- Métricas
- Planes de Proyecto y de Release
- Etc.

Reunión de Planificación de Sprint



Reunión Diaria Scrum (Máx. 15min)



- **NO** son reuniones de reporte de estado al Scrum Master
- Son **compromisos** frente a los demás miembros del equipo



Reglas

- Diaria
- En el mismo lugar
- A la misma hora (sea puntual!)
- 15 minutos
- Participantes (Equipo)
- El Scrum Master comienza la reunión al horario convenido (sin importar quién está presente y quién no!)

Revisión del Sprint / Retrospectiva del Sprint

Revisión del Sprint	Retrospectiva del Sprint
<ul style="list-style-type: none">• El equipo presenta la funcionalidad terminada (done) al Dueño del Producto y demás stakeholders• Los miembros del equipo responden las preguntas de los stakeholders en relación a la demostración, y toman nota de los cambios propuestos• Al finalizar la presentación, los stakeholders dan su impresión acerca del producto, cambios deseados y la prioridad de esos cambios	<ul style="list-style-type: none">• Provee una visión periódica de qué está funcionando y qué no está funcionando• Típicamente de 1 hora• Se realiza al final de cada sprint• Todo el equipo participa (Scrum Master, Dueño del Producto, Equipo)• Se pueden utilizar varias técnicas

- Reunión informal
- 1h de preparación



Marco de Trabajo Scrum



Ceremonias

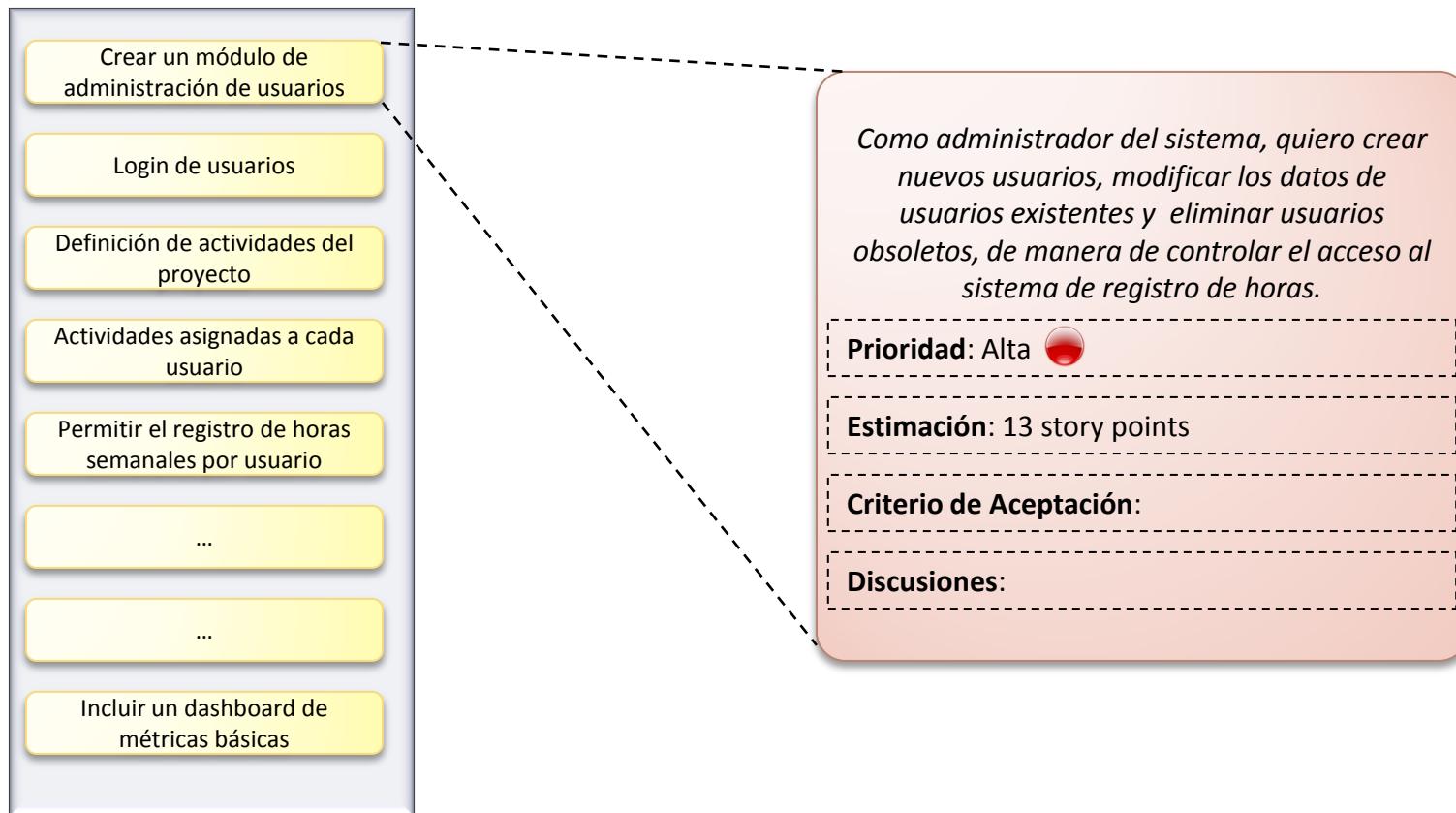
- Planificación del Sprint
- Reunión Diaria
- Revisión del Sprint
- Retrospectiva del Sprint

Artefactos

- Backlog de Producto
- Backlog de Sprint
- Métricas
- Planes de Proyecto y Release
- Etc.

Backlog de Producto

Es la lista maestra de toda la funcionalidad deseada del producto. Esta lista expresa los requerimientos del dueño del producto **en su lenguaje**, está estimada y se encuentra ordenada de acuerdo a la **prioridad** de cada ítem.



Backlog de Sprint

Como administrador del sistema, quiero crear nuevos usuarios, modificar los datos de usuarios existentes y eliminar usuarios obsoletos, de manera de controlar el acceso al sistema de registro de horas.

Prioridad: Alta

Estimación: 13 sp

Criterio de Aceptación:

Discusiones:

Crear estructura de BD

Crear web service para conexión LDAP

Crear pantalla de login

Crear casos de prueba automáticos para LDAP

Crear casos de prueba manuales de la pantalla

...

...

Actualizar el manual de usuario

Crear las tablas de usuario, permisos y permisos/usuario en la BD, generando los scripts de las mismas.

Crear los correspondientes mappings de Hibernate, y las clases DAO de Usuario y Permisos.

Generar las pruebas unitarias para las DAO.

Responsable: Mauricio

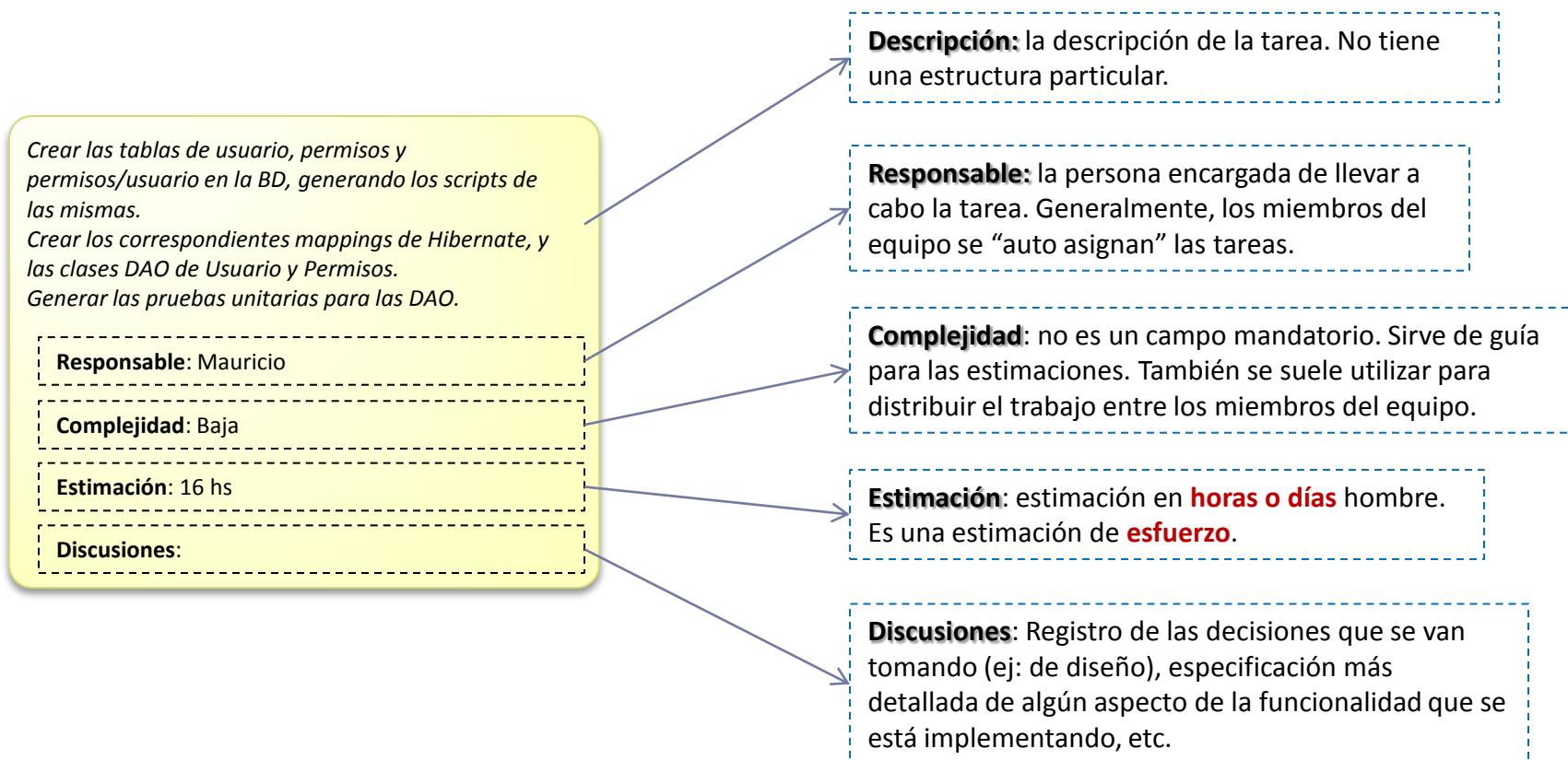
Complejidad: Baja

Estimación: 16 hs

Discusiones:

Ítems de Backlog de Sprint

Un ítem de backlog de sprint es una descripción de una **tarea** necesaria para completar una historia de usuario dada. El backlog de sprint está compuesto por todas las tareas necesarias para completar todas las historias comprometidas.



Cálculo de la capacidad del equipo

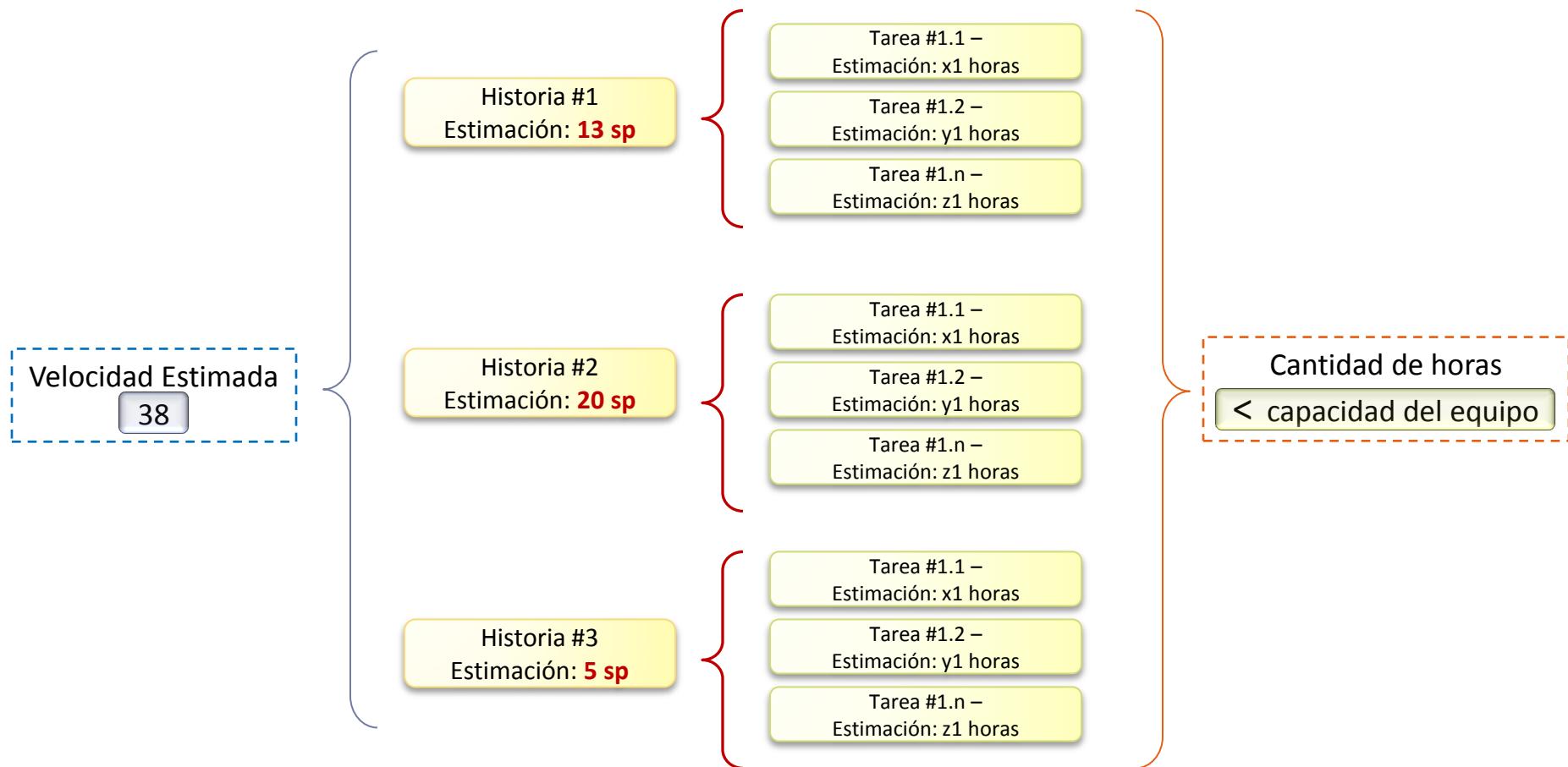
La capacidad del equipo es la cantidad de horas efectivas que dicho equipo tiene disponibles, para este sprint.

	Días disponibles	Horas diarias disponibles	% asignación al proyecto	Capacidad en el sprint
	20	7	100	140
	20	5	100	100
	18	7	100	126
	20	7	50	70
	Capacidad total del equipo			436

Capacidad individual = dias_disp * horas_diarias_disp * porcentaje_asignacion

Capacidad equipo = \sum (capacidad individual)

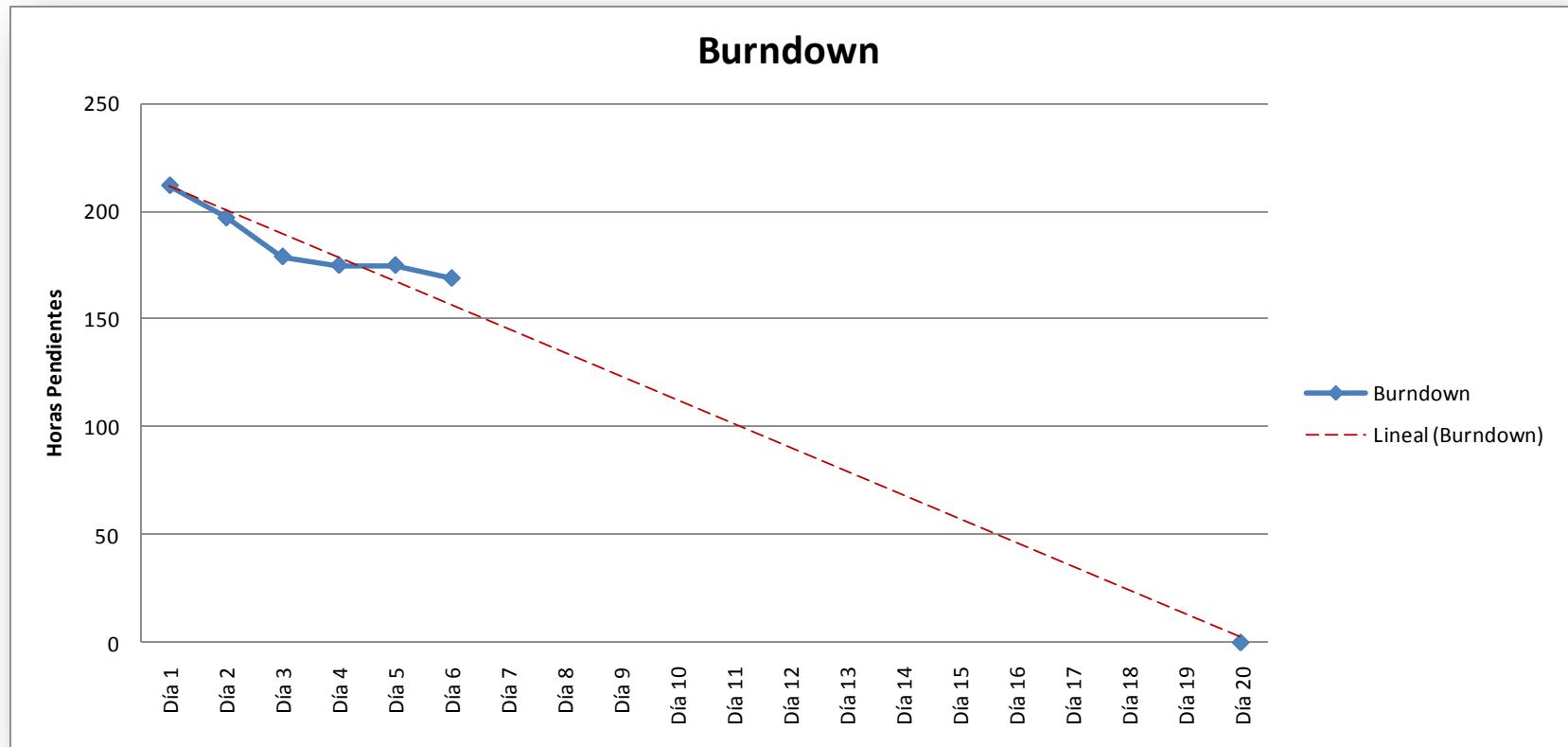
Backlog de Sprint



Sprint Burndown Chart

Nombre de la métrica	Propósito/Objetivo	Formula	Responsable de Colectarla	Período de Reporte
Sprint BurnDown Chart (SBDC)	Provee un indicador diario de la velocidad (velocity) del equipo y el progreso respecto del trabajo comprometido para el sprint actual.	Tendencia de Regresión Lineal Trabajo Pendiente + Tendencia = Trabajo pendiente del día anterior a ayer + Trabajo pendiente de ayer + Trabajo pendiente de hoy) / 3	Scrum Master	Se colecta una vez por día. Es la métrica principal de seguimiento del Sprint.

Sprint Burndown Chart



Lecturas Obligatorias

Autor	Título	Editor	Referencia
Ken Schwaber	Scrum Development Process; 1995		http://www.jeffsutherland.org/oopsla/schwarz.pdf
Mike Cohn	Mountain Goat Software	Web	http://www.mountaingoatsoftware.com/topics/scrum

Lecturas Recomendadas

Autor	Título	Editor	Referencia
Curtis, B., Kellner, M., Over, J.	Process Modelling	ACM	Comm. ACM. 35(9); pp. 75-90 (1992)
Hamid Shojaee	SCRUM in Under 10 Minutes	Youtube	http://www.youtube.com/watch?v=Q5k7a9YEoUI&feature=related

Bibliografía

Autor	Título	Editor	Referencia
Pfleeger	Ingeniería de Software	Prentice Hall	
Jacobson, Booch, Rumbaugh	El Proceso Unificado de Desarrollo de Software	Addison Wesley, 1999	
Weinberg	Quality Software Management, vol 1	DorsetHouse, 1992	
Tom Gilb	Principles of Software Engineering Management	Addison Wesley	
Florac, Park, Carleton	Practical Software Measurement	SEI	
B. Curtis, M. Kellner, J. Over	Process Modelling	ACM	Comm. ACM. 35(9); pp. 75-90 (1992)
Mauricio Silclr, Pablo Szyrko, Álvaro Ruiz de Mendarozqueta, Diego Rubio	Un Método Heurístico para el Análisis y Selección de Herramientas de Modelado de Procesos de Desarrollo de Software	LIDICALSO – UTN - FRC	http://www.institucional.frc.utn.edu.ar/sistemas/lidicalso/
Barry Boehm	A spiral model of Software development and enhancement	IEEE	0018-9162 (1988)

Bibliografía

- Ken Schwaber and Jeff Sutherland; Scrum Guide; Scrum Alliance; 2010
- Kent Beck; Embracing Change with Extreme Programming; IEEE; 1999
- Brent Barton et al.; Reporting Scrum Project Progress to Executive Management through Metrics; Scrum Alliance; 2005
- Victory Szalvay et al; Agile Transformation Strategy; Danube; 2005
- Jeff Sutherland et al.; Scrum and CMMI Level 5: The Magic Potion for Code Warriors; 2007;
- Mike Cohn; Agile Estimating and Planning ; Prentice Hall; 2006; 0-13-147941-5
- Mary and Tom Poppendieck; Lean Software Development: An Agile Toolkit; Addison-Wesley; 2003; 0-321-15078-3

Versión

Versión	Fecha	Comentarios	Autor
1.0.0_Draft_A	24-Jul-2012	Versión inicial	Natalia Andriano
1.0.0_Draft_B	10-Aug-2012	Cambios varios	Diego Rubio
1.0.0	10-Aug-2012	Baseline 2012.	Diego Rubio