Administración de proyectos de ingeniería de software

APIS - Semestre i

MA. Ing. Jaime Omar Tiznado Carranza - 2015 - 2018

Tecnológico de Monterrey

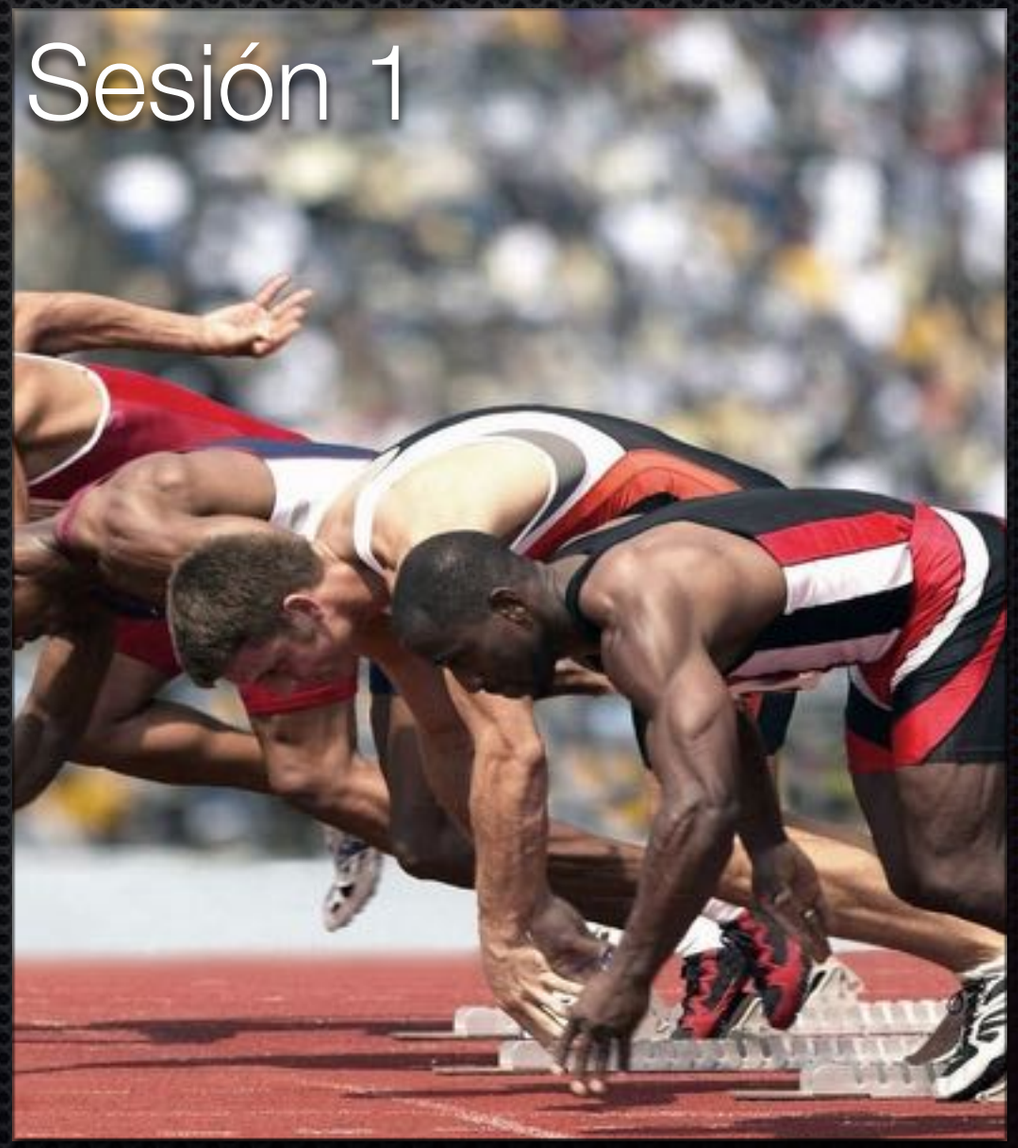
Presentación

- Nombre
- Carrera / Estudios
- Experiencia
- Rol preferido en proyecto de TI
- Fortalezas / Debilidades

Tecnológico de Monterrey

Agenda

- Introducción
  - Estructura de las sesiones de trabajo
  - Temario
  - Área de Confort
  - Evaluación
  - Proyecto
  - Material de apoyo
- Temas de la sesión
  - La crisis del software
  - Modelos de procesos de software
    - Capability Maturity Model (CMMi)
  - Modelos de ciclo de vida de proyectos



Primera parte (50 Minutos)

Introducción y teoría de los temas

10 min descanso

Segunda parte (55 Minutos)

Practica / Teoría

15 min descanso

Tercera parte (40 Minutos)

Trabajar sobre el proyecto

Temario

TC3002 : Administración de proyectos de ingeniería de software

Temas y Subtemas	Objetivos	Duración	Unidad
1. Importancia de los procesos en el marco de la Ingeniería de Software.	1. Importancia de los procesos en el marco de la Ingeniería de Software.	3	horas
2. Análisis de alcance y requerimientos de un proyecto.	2. Análisis de alcance y requerimientos de un proyecto.	3	horas
3. Estimación de tamaño, esfuerzo, tiempo y costo en proyectos de software.	3. Estimación de esfuerzo en proyectos de software.	12	horas
4. Técnicas avanzadas y herramientas automatizadas para estimación de esfuerzo, en proyectos de software.	4. Técnicas avanzadas y herramientas automatizadas para estimación de esfuerzo, en proyectos de software.	3	horas
5. Planeación y calendarización de un proyecto.	5. Planeación y calendarización de un proyecto.	3	horas
6. Ejecución y control del proyecto.	6. Ejecución y control del proyecto.	3	horas
7. Gestión de riesgos.	7. Gestión de riesgos.	3	horas
8. Gestión de la configuración.	8. Gestión de la configuración.	3	horas
9. Calidad en el software.	9. Calidad en el software.	6	horas
10. Gestión del aprovisionamiento.	10. Gestión del aprovisionamiento.	3	horas
11. Cierre del proyecto y lecciones aprendidas.	11. Cierre del proyecto y lecciones aprendidas.	3	horas
HORAS		**45**	
Exámenes		3	
HORAS		**48**	

Área de confort

SIMON SINEK

Y LOS

MILLENIALS

@anuoraguilar

MA. Ing. Jaime Omar Tiznado Carranza - 2015 - 2018

Tecnológico de Monterrey

Evaluación

Evaluaciones	Ponderación
Evaluación parcial	
Examen Parcial	70 %
Avance del proyecto y Actividades	30 %
Total	**100.00 %**
Evaluación Final	
Examen Parcial	30.00 %
Examen Final	30 %
Proyecto - calificación del cliente	35 %
Semana i	5.00 %
Total	**100.00 %**

Proyecto - metodologia clásica

1) Rational Unified Process (RUP) developed by Scott Ambler

- Modelo
- Implementación
- Pruebas
- Deployment

Administración del proyecto

Mejora continua

- Análisis
- Diseño
- Construcción
- Pruebas
- Estabilización / Soporte

- Análisis de requerimientos
- Dimensionamiento
- Caso de Uso
- Prototipo
- ER
- Casos de pruebas
- Código documentado
- Scripts BD
- Pruebas de integración
- Pruebas funcionales
- Pruebas de usurario
- Piloto con cliente

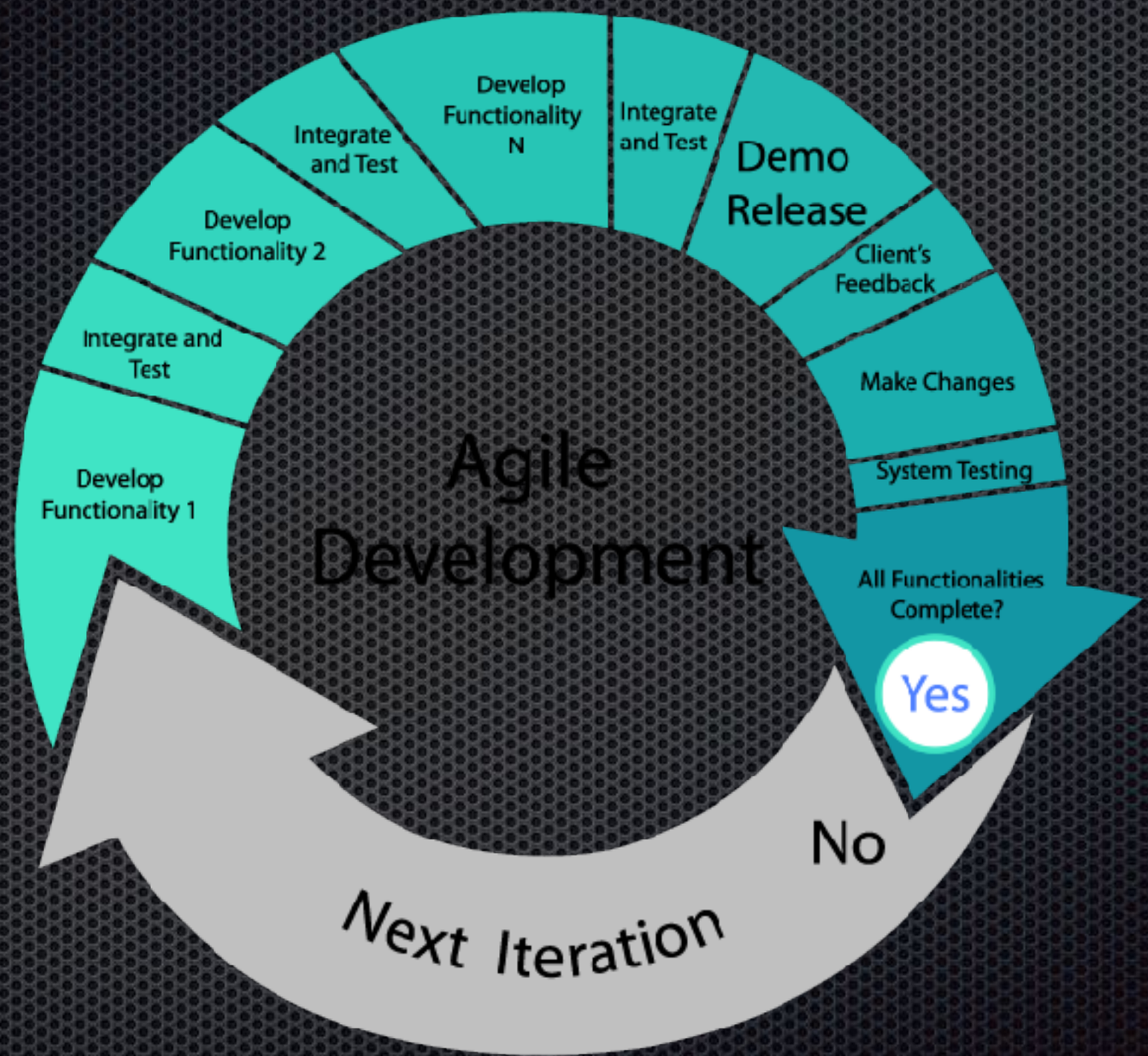
Manual de instalación y Usuario

Administración de riesgos

Reportes de seguimiento / Lineas base

Environment / Ambiente

Proyecto - Metodologia ágil



Tecnológico de Monterrey

Organizar equipos

1. Cada equipo tiene que tener un experto de:
   - Administración.
   - Análisis / Diseño
   - Desarrollo
   - QA´s / Ing. de pruebas
   - Documentación
2. Conformar
   - 2 equipos de 6 personas
3. Metodologia
   - Libre

Material de apoyo

Bibliografía

LIBROS DE TEXTO:

*** Pandian, C. Ravindranath, Software metrics : a guide to planning, analysis, and application, , Boca Raton, FL : Auerbach Publications, flu, 2003, eng, [0849316618 (papel alcalino)]**

* Humphrey, Watts S., Managing the software process, , Reading, Mass. : Addison-Wesley, , c1989, , [0201180952]

* Hughes, Bob., Software project management , 5th ed., London : McGraw-Hill Higher Education, , 2009, , [9780077122799], [0077122798]

LIBROS DE CONSULTA:

* De Marco, T; Lister, T., Peopleware: Productive Projects and Teams, Segunda edición, Dorset House Publishing, , 1999, , [0932633439]

* Jalote, P., CMM in practice : processes for executing software projects at Infosys, , Reading, Mass : Addison-Wesley, Massachusetts, 2000, eng, [0201616262]

* Phillips, Joseph, IT project management : on track from start to finish, 2nd ed, Emeryville, Calif. : McGraw-Hill/Osborne ; London : McGraw-Hill, California, 2004, eng, [0072232021]

LIBROS RECOMENDADOS

A Guide to the Project Management Body of Knowledge (PMBOK® Guide)-Fifth Edition, A Guide to the Project Management Body of Knowledge (PMBOK® Guide)-Fifth Edition

Material de apoyo

Software
MS Project
Hoja de Calculo
P. Point o similar
IDE y lenguaje predilecto
Herramienta para diagramas
Procesador de textos
Repositorio de código e integración (SVN y TortoiseSVN)
Seguimiento de hallazgos (BUG Tracker)

1. Importancia de los procesos en el marco de la Ingeniería de Software.

1. Definir: Características de pan
2. Cualidades de ambos productos
3. Apego a diseño

Calidad

- Según Deming (1989) la calidad es “un grado predecible de uniformidad y fiabilidad a bajo coste, adecuado a las necesidades del mercado”.

- Juran (Juran y Gryna 1993) la calidad se define como adecuación al uso, esta definición implica una adecuación del diseño del producto o servicio (calidad de diseño) y la medición del grado en que el producto es conforme con dicho diseño (calidad de fabricación o conformidad).

1.1 La crisis del software

IT Infrastructure Library

1.2. Modelos de procesos de software.

CMMI® Institute

CMMIDEV/3℠
Exp. 2017-05-09 / Appraisal #22278

NMX-I-059/02 / MOPROSOFT
NYCE
REGISTRO

Scrum.org
Improving the Profession of Software Development

PMI®
Project Management Institute

THE Open GROUP
™ TOGAF® 9

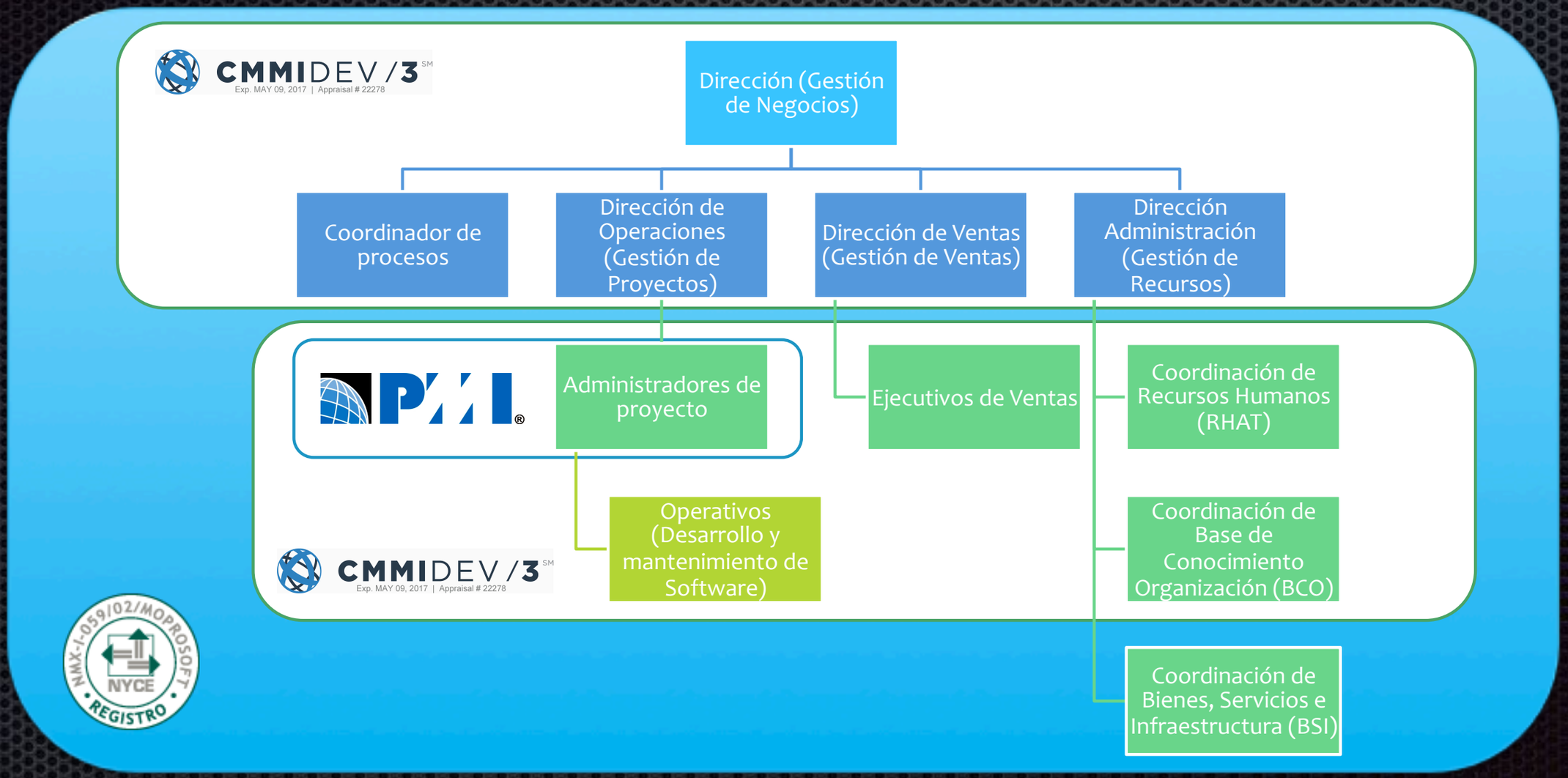
ITIL®

Information Security Management System
ISO
27001
Certified

CobiT®
GOVERNANCE, CONTROL and ASSURANCE for INFORMATION and RELATED TECHNOLOGY

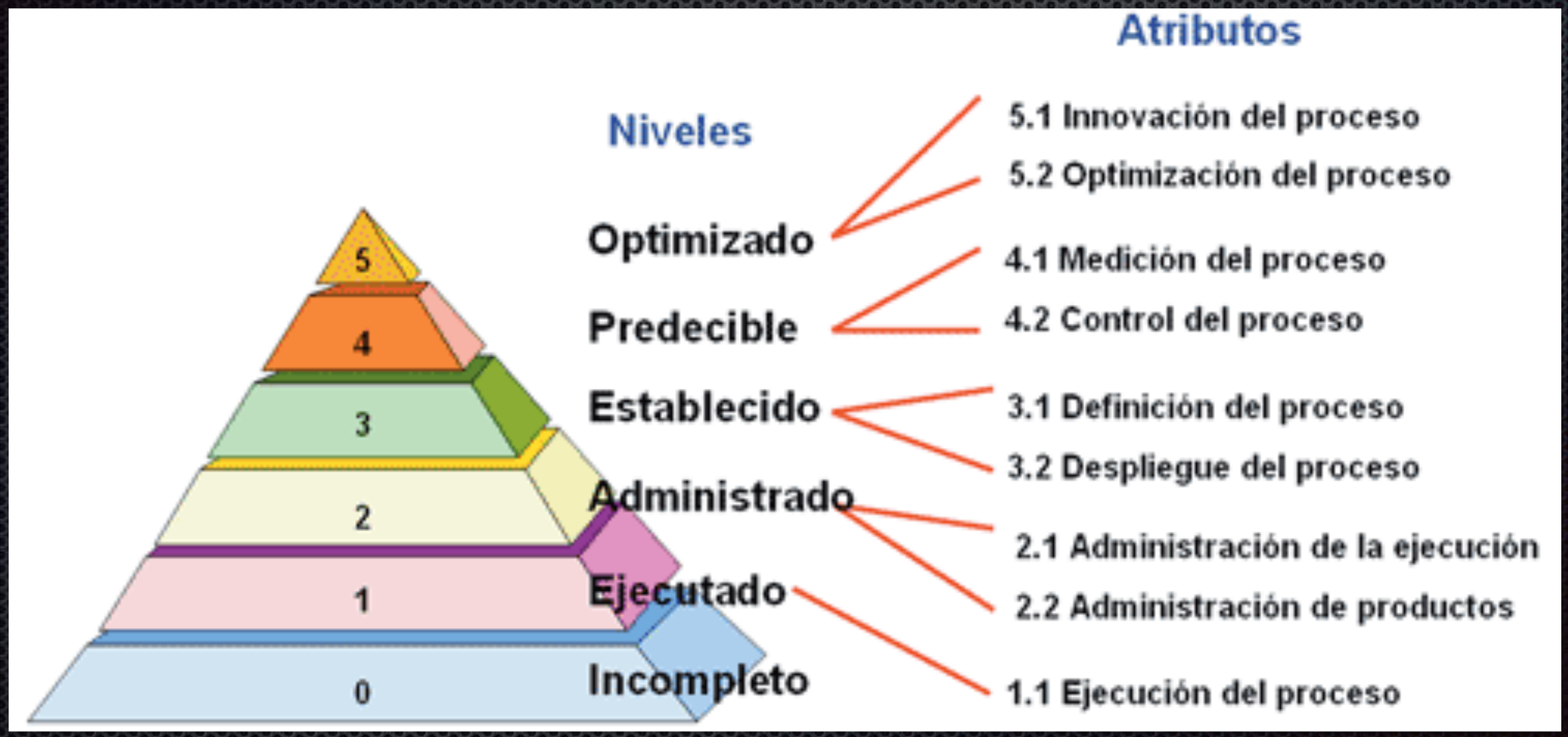
Tecnológico de Monterrey

1.2. Modelos de procesos de software.



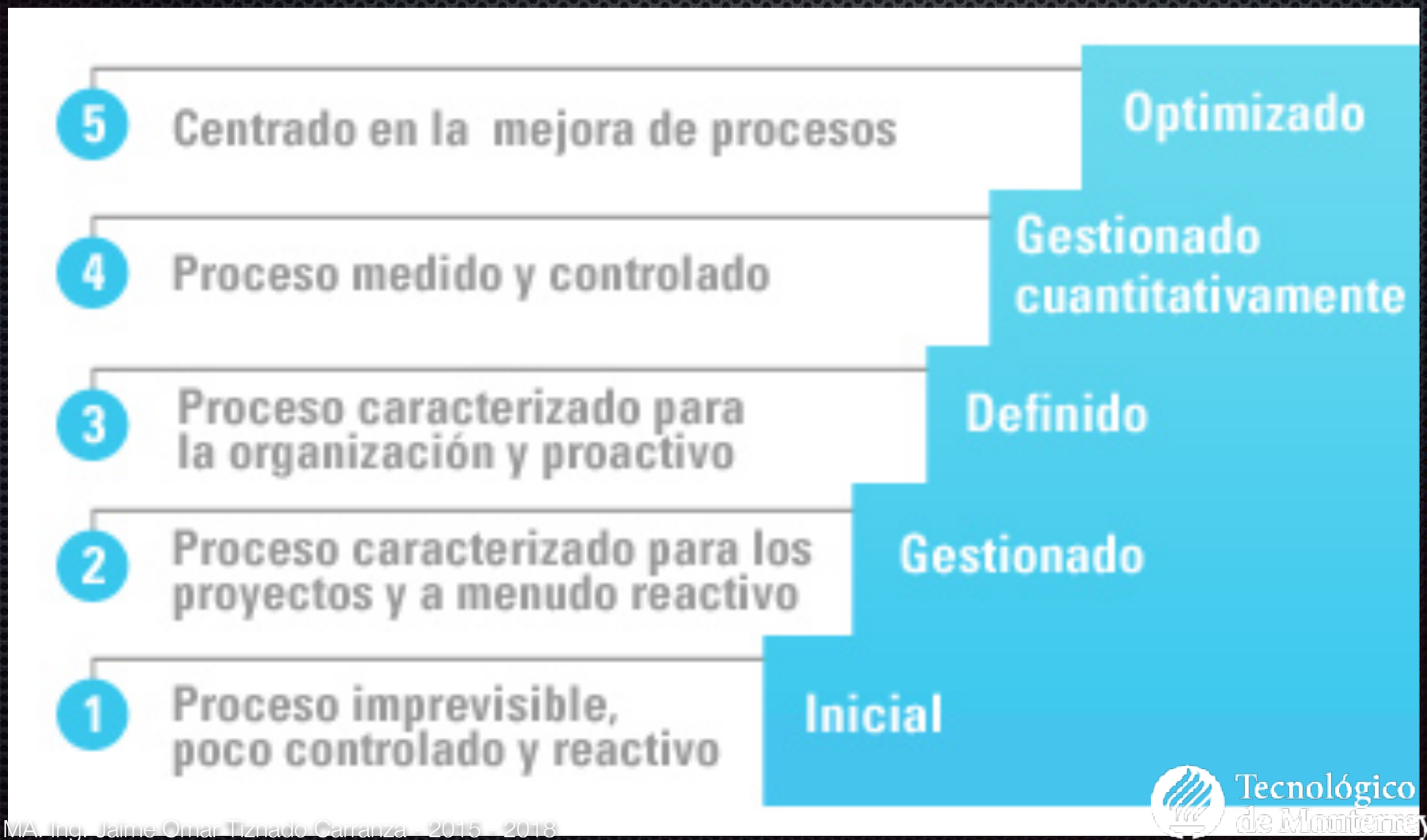
Tecnológico de Monterrey

NMX-I-059-NYCE (MoProSoft) Niveles de madurez



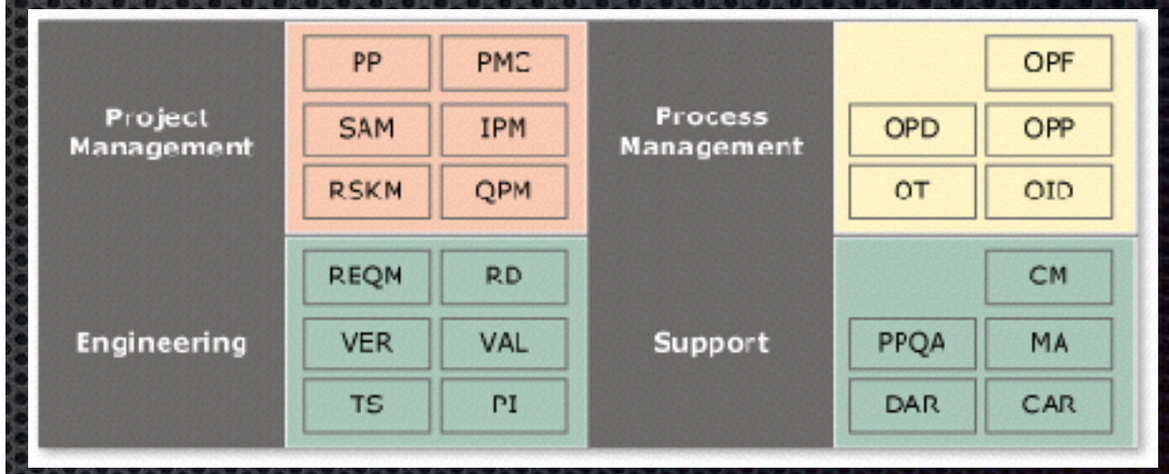
NMX-I-059-NYCE (MoProSoft) Áreas de proceso

CMMI / DEV
Niveles de madurez



CMMI - DEV
Areas de proceso

Configuration Management	CM	2	Target Profile 2
Measurement and Analysis	MA	2	
Project Monitoring and Control	PMC	2	
Project Planning	PP	2	
Process and Product Quality Assurance	PPQA	2	
Requirements Management	REQM	2	
Supplier Agreement Management	SAM	2	
Decision Analysis and Resolution	DAR	3	Target Profile 3
Integrated Project Management	IPM	3	
Organizational Process Definition	OPD	3	
Organizational Process Focus	OPF	3	
Organizational Training	OT	3	
Product Integration	PI	3	
Requirements Development	RD	3	
Risk Management	RSKM	3	
Technical Solution	TS	3	
Validation	VAL	3	
Verification	VER	3	
Organizational Process Performance	OPP	4	Target Profile 4
Quantitative Project Management	QPM	4	
Causal Analysis and Resolution	CAR	5	Target Profile 5
Organizational Performance Management	OPM	5	



Tecnológico de Monterrey

Manifiesto Agil - 12 principios

1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
4. Business people and developers must work together daily throughout the project.
5. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

http://agilemanifesto.org/

Manifiesto Agil - 12 principios

7. Working software is the primary measure of progress.
8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
9. Continuous attention to technical excellence and good design enhances agility.
10. Simplicity--the art of maximizing the amount of work not done--is essential.
11. The best architectures, requirements, and designs emerge from self-organizing teams.
12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

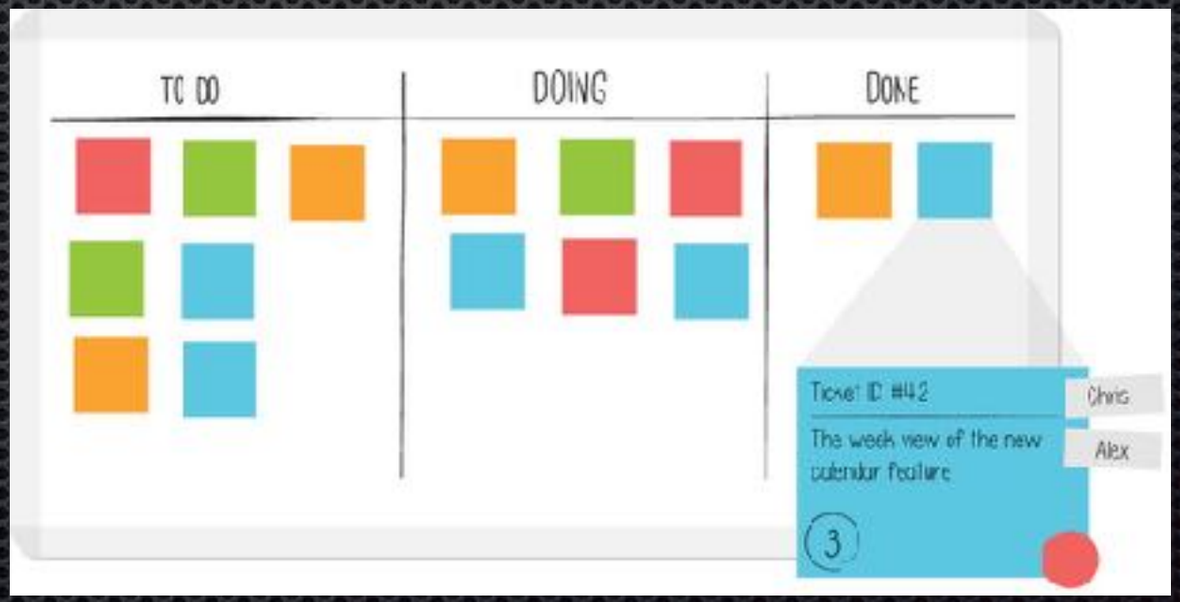
http://agilemanifesto.org/

Scrum (metodologias ágiles)

- Conjunto de procesos bien definidos para estructurar e trabajo de desarrollo de software con las siguientes características:
  - La base son iteraciones (Sprints).
  - Construir equipos multi - disciplinarios.
  - Definir un dueño del producto (Stakeholders).
  - Scrum Master.
  - Definir juntas periódicas y frecuentes (cortas).
  - Daily Status
  - Sprint Reviews.

Kanban (metodologias ágiles)

- Menos estructurado que Scrum:
  - Introduce el concepto de mejora incremental.
  - Pizarra / Tablero
  - swim lanes – horizontal “pipelines” for different types of work
  - “Work In Progress (WIP)”
  - The WIP limit tells you how much work items are allowed to be in a certain state at any given point in time

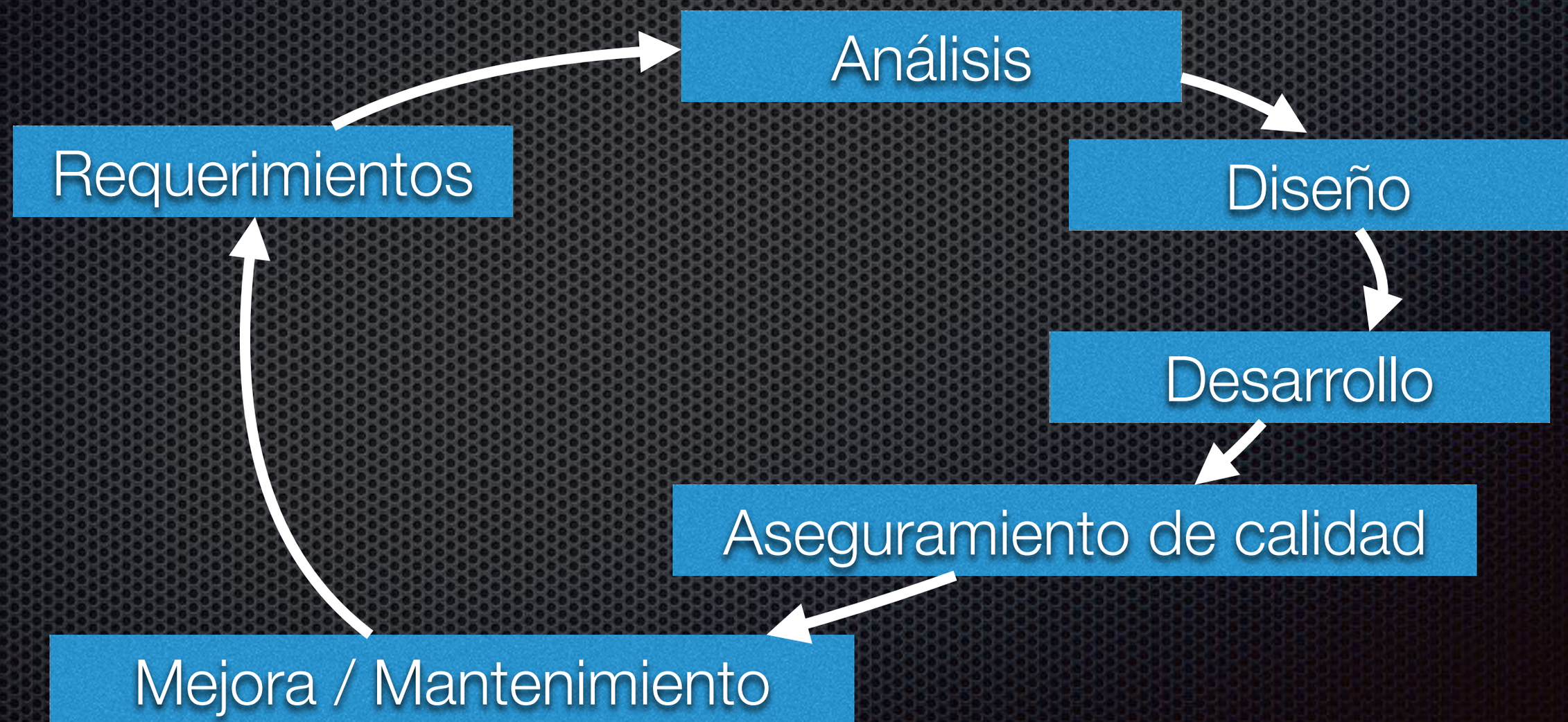


1.3. Modelos de ciclo de vida de proyectos de software.

- Modelo de cascada

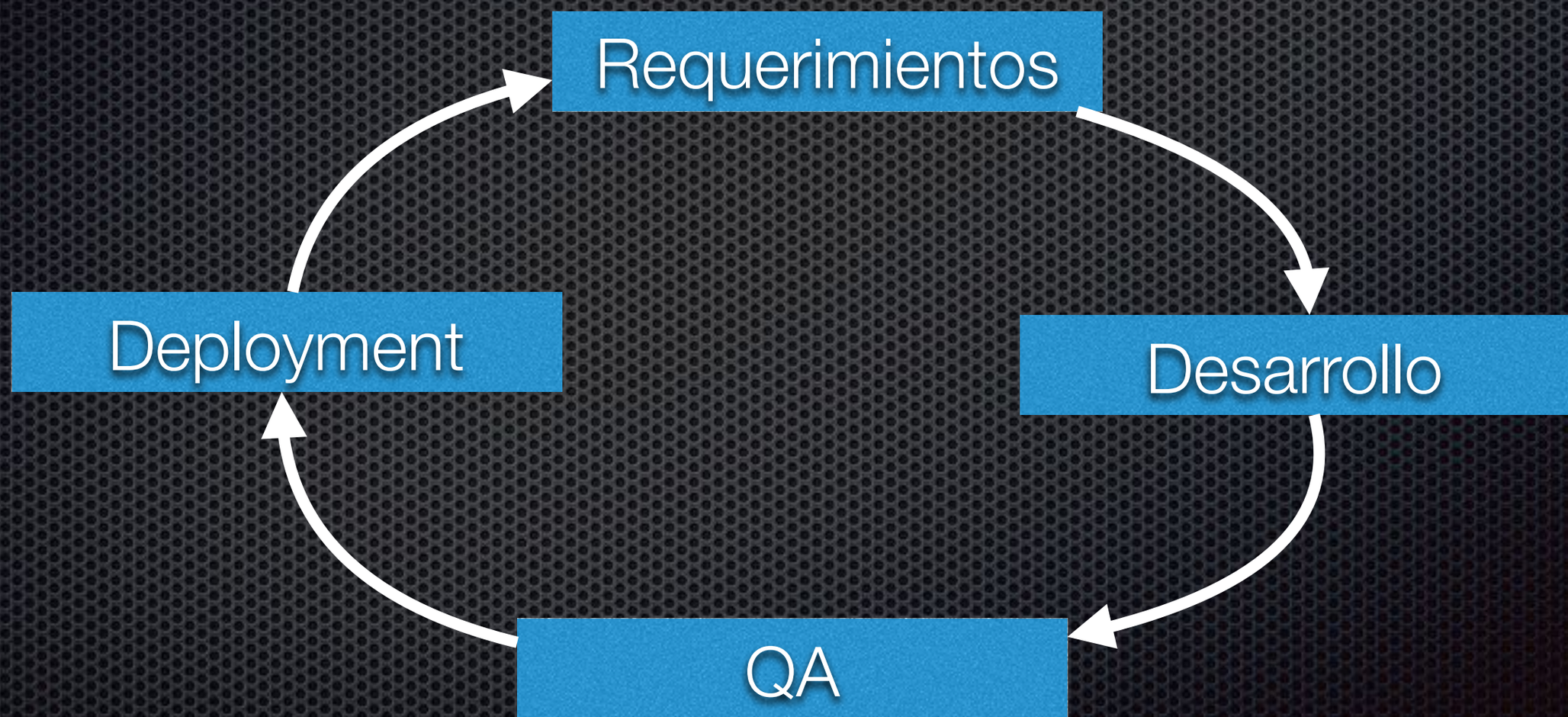
1.3. Modelos de ciclo de vida de proyectos de software.

- Modelo iterativo



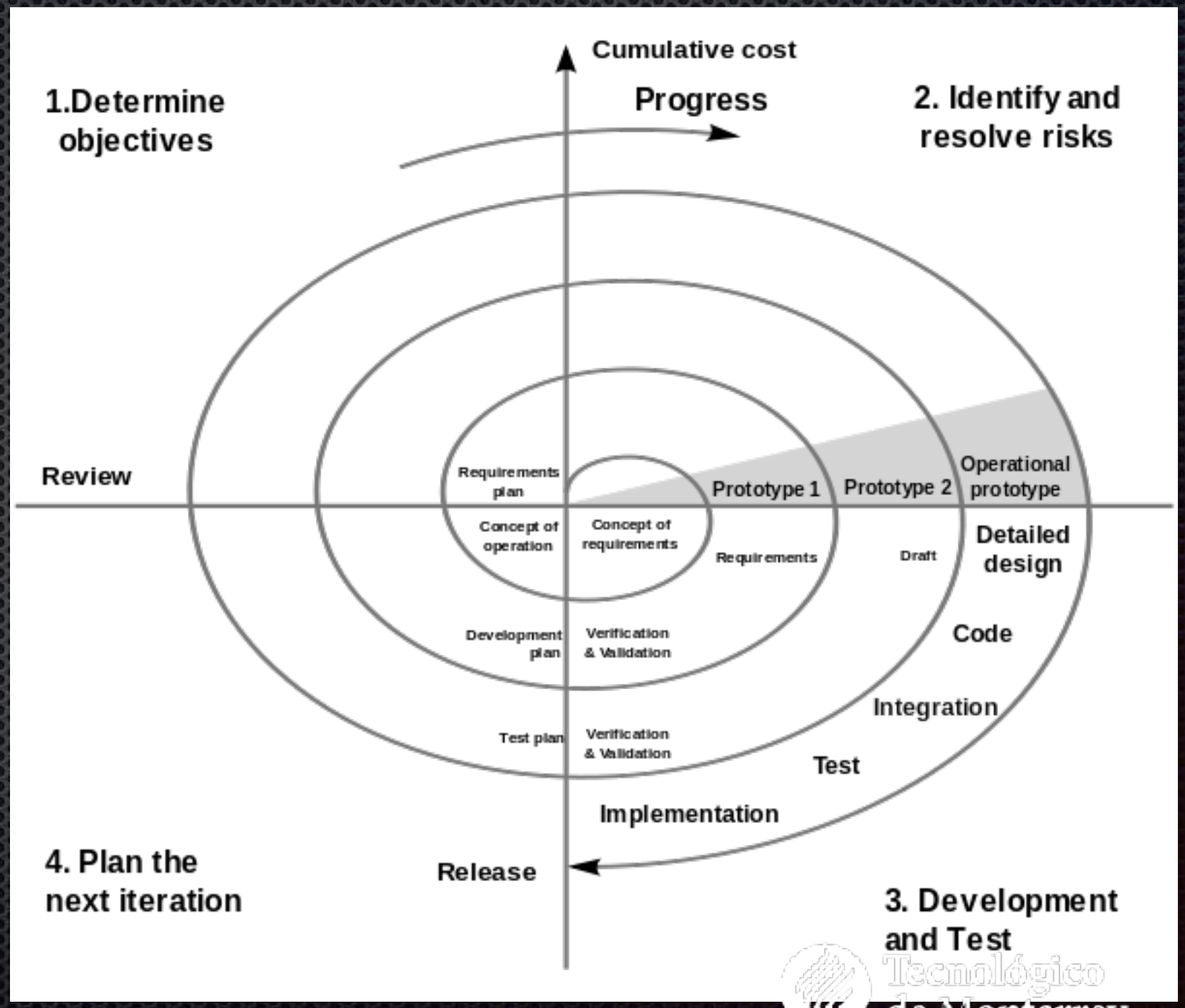
1.3. Modelos de ciclo de vida de proyectos de software.

- RAD Rapid Application Development



1.3. Modelos de ciclo de vida de proyectos de software.

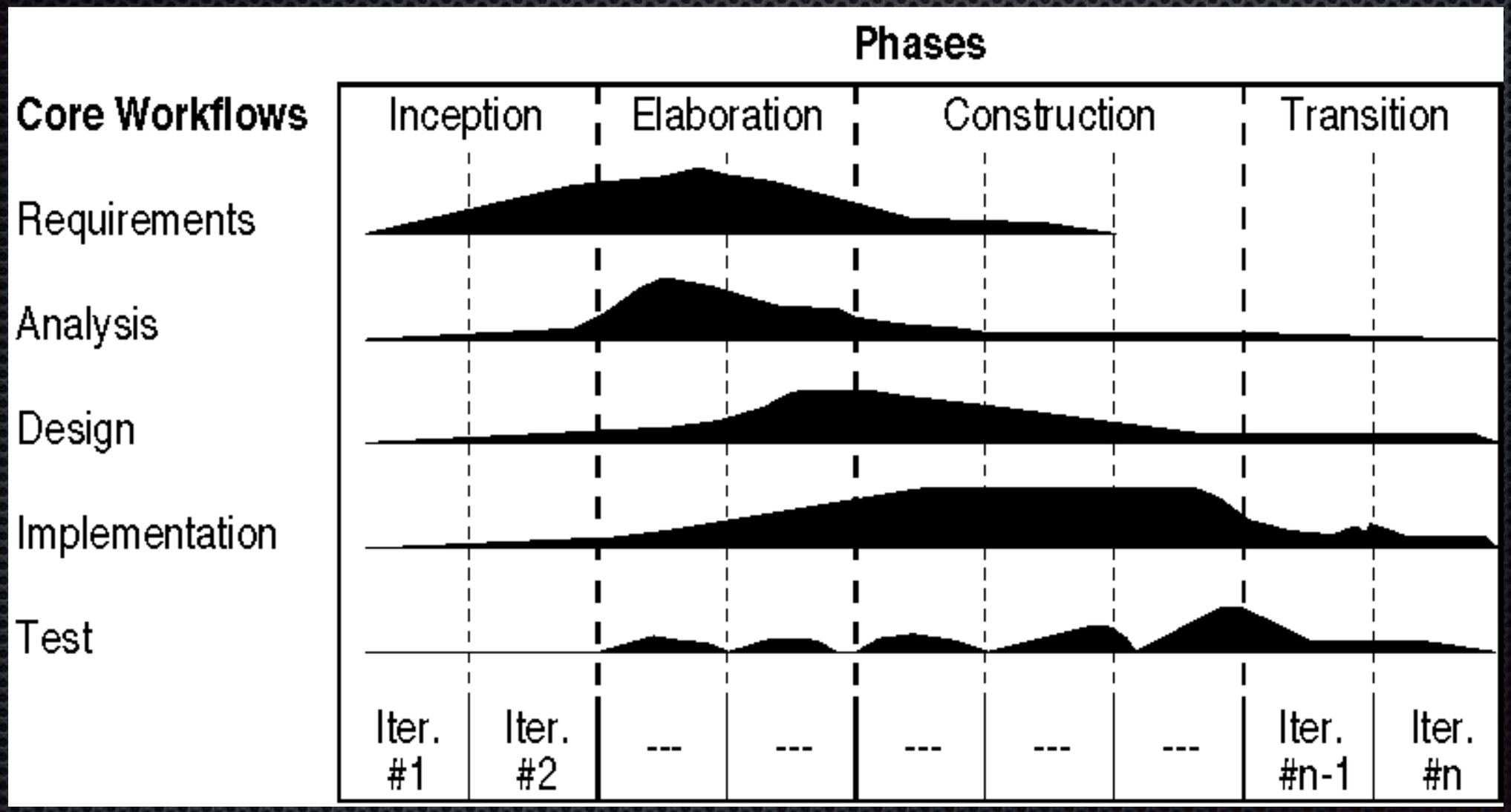
- Modelo de espira



1988, Barry Boehm

Tecnológico de Monterrey

1.3. Modelos de ciclo de vida de proyectos de software.



The Unified Software Development Process by Ivar Jacobson

Proyecto Actividad 1

- Suponer que son emprendedores y montan su fabrica de software, sobre este supuesto:
- Tomar de base los modelos de procesos antes explicados.
- Hacer los diagramas necesarios y redactar como implementarían los modelos de procesos y mejores practicas en su empresa.