Metodología de Sistemas I

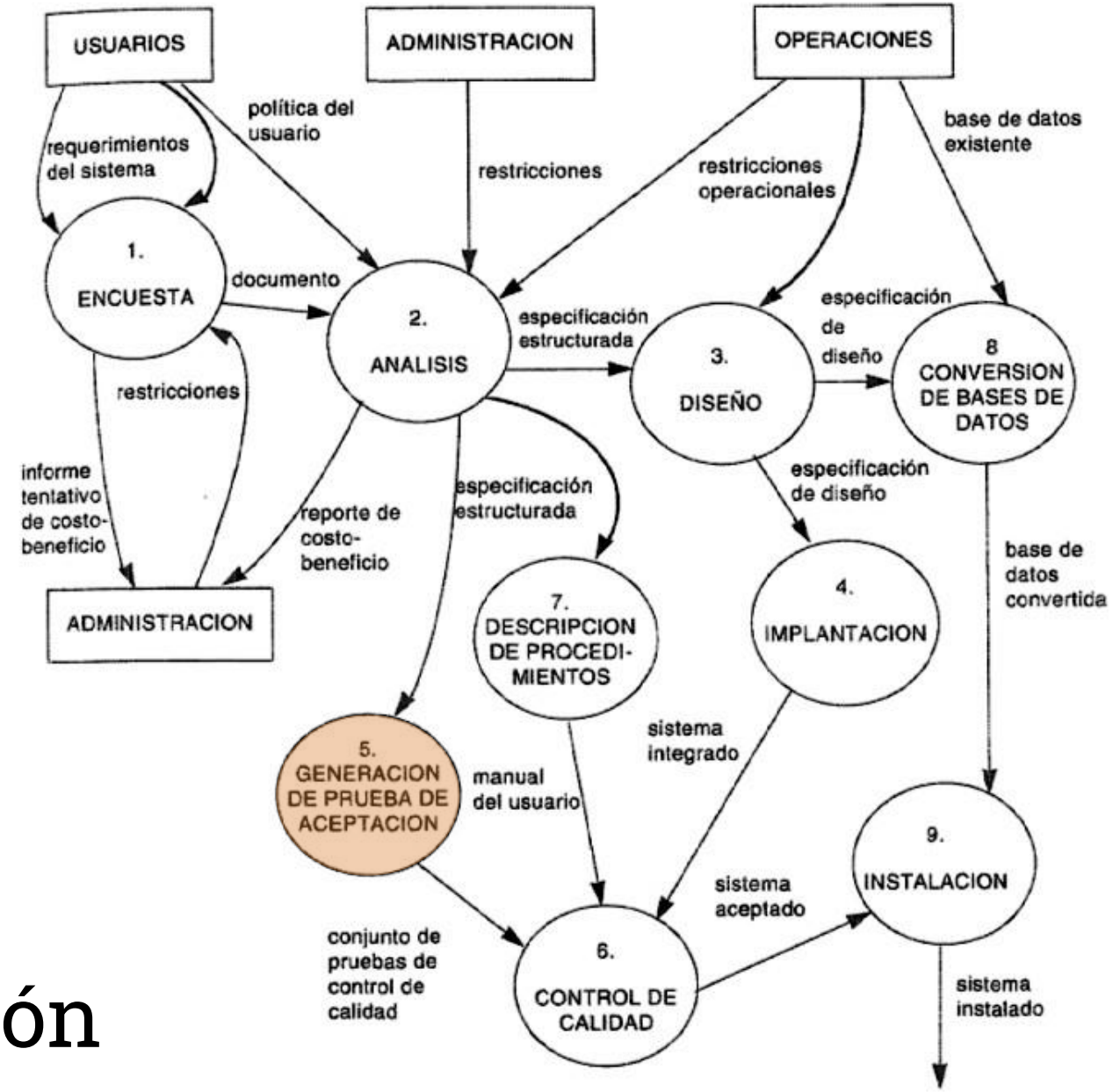
Año 2019
2º cuatrimestre

Ricardo Aiello
Germán Scarafilo

pruebas


We received your delivery on time, but we couldn't find the tests
Tests ?
What tests?
You know, the unit tests. The ones for testing the application layer, Quality Assurance and all
Ah, yes, the unit tests...
There aren't any unit tests
No tests ? But what about the guidelines...
How do you check that everything works correctly?
Um...
IT'S MAGIC
CommitStrip.com

Generación de pruebas de aceptación

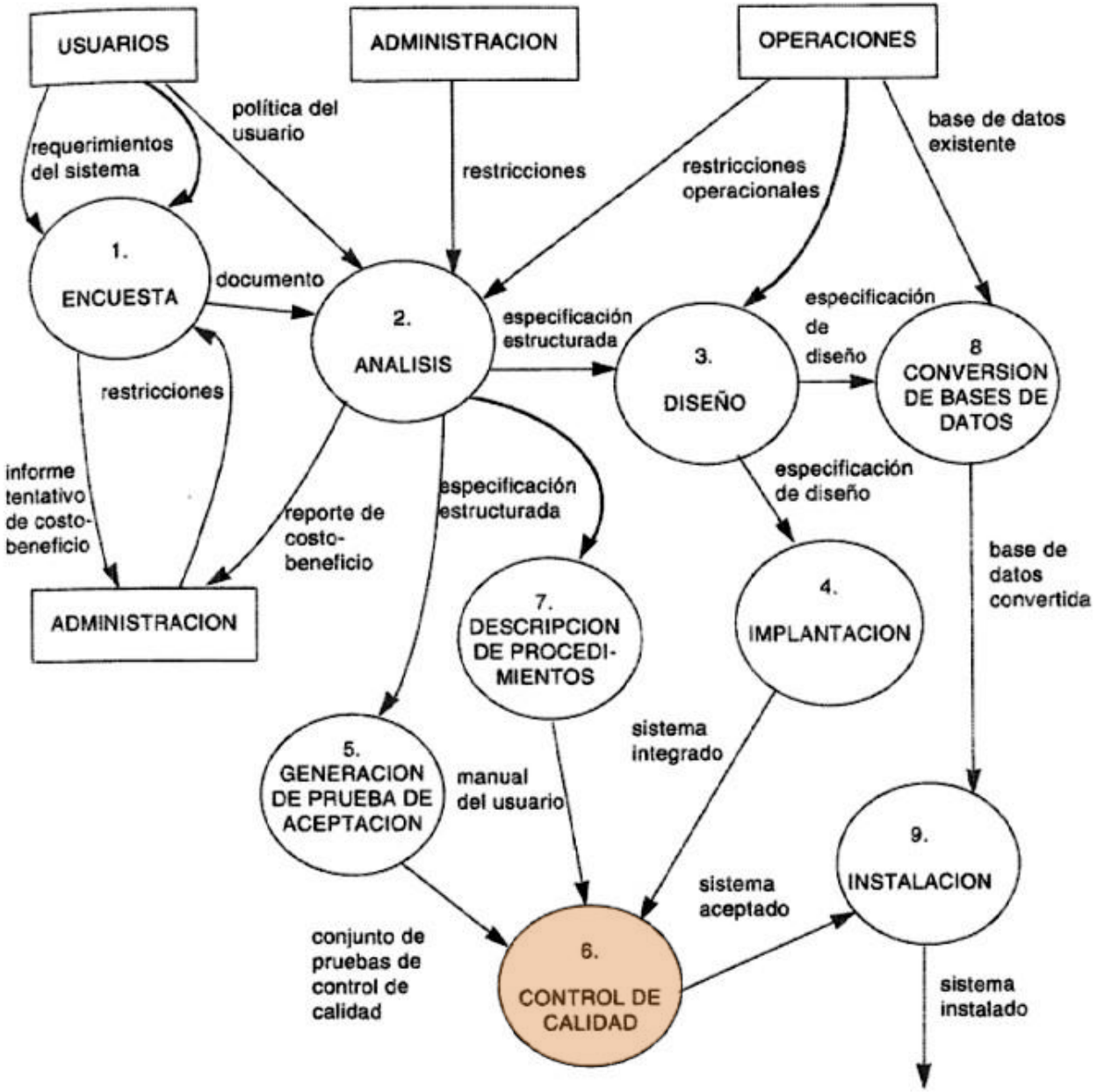


generación de pruebas de aceptación

preparación de evaluación funcional en conjunto con los usuarios

porque, como sabemos, puede fallar

Control de calidad



control de calidad

ejecución de pruebas de aceptación para garantizar la calidad mínima exigible del sistema

pruebas

- se crean
  en la actividad de Generación de pruebas de aceptación
- se ejecutan
  en la actividad de Control de calidad
- toman tiempo
  probablemente la mitad del destinado al desarrollo
- constituyen un proceso iterativo
  cada tanda verifica lo identificado en la anterior
- pueden ser manuales o automáticas

plan de prueba

documento organizado, más o menos formal, según el caso

- propósito de las pruebas
  su objetivo, y la parte del sistema que se estará probando
- localización, horario y responsables
  dónde y cuándo se harán, y quién(es) estará(n) a cargo
- descripciones
  entradas a proveer, salidas y resultados esperados
- procedimientos
  cómo ejecutarla, cómo capturar y analizar los resultados
- resultados
  se agregan luego de ejecutarla

estrategias

enfoque ascendente

- comienza con módulos individuales y los prueba de manera separada
  las famosas pruebas de unidad, de módulos o de programas
- va combinando los módulos probados en unidades cada vez más grandes
  todavía de manera separada
- finalmente, prueba el sistema completo
  pruebas de sistema, seguidas de las pruebas de aceptación que el usuario ejecute con sus propios casos de prueba

enfoque descendente

- comienza con un esqueleto del sistema
  ejercitando principalmente las interfaces entre subsistemas
- continúa con aspectos más detallados
  verificando el funcionamiento interior de cada módulo,
  y bajando de nivel hasta alcanzar los más pequeños

tipos de prueba

según momento del proyecto

caja negra

se conocen las entradas y salidas,
pero no el algoritmo

puede realizarse al finalizar el análisis

caja blanca
o “de vidrio”

basada en la lógica
de un programa existente

requiere diseño e implantación

tipos de prueba

según aspecto a verificar

funcional

asegurar que el sistema
realice sus funciones normales [1]
de manera correcta

1- los requisitos funcionales

de recuperación

asegurar que el sistema
pueda recuperarse adecuadamente
de diversos tipos de fallas

de desempeño

asegurar que el sistema
cumpla con las especificaciones [1]
de volumen de datos/transacciones
y tiempo de respuesta

1- volcadas en el modelo de implantación del usuario

exhaustiva

asegurar un comportamiento perfecto cubriendo todas las situaciones posibles que podría enfrentar el sistema

claramente, esto no es viable en sistemas complejos

bibliografía

análisis estructurado moderno

- Cap. 5: El ciclo de vida del proyecto
- Cap. 23: Programación y prueba


UTNFra