



Departamento de  
Lenguajes y Sistemas Informáticos



Universidad de Sevilla

# *Generación de pruebas de sistema a partir de la especificación funcional*

*Javier Jesús Gutiérrez Rodríguez*

[javierj@lsi.us.es](mailto:javierj@lsi.us.es)

[www.lsi.us.es/~javierj/](http://www.lsi.us.es/~javierj/)

Tutores: Doctor D. Manuel Mejías  
Doctora Dña. Maria José Escalona

# Índice.

---



1. Introducción.
2. Comparativa de propuestas.
3. Análisis y conclusiones.
4. Trabajos futuros.
5. Publicaciones.

# Índice.

---



1. Introducción.

2. Comparativa de propuestas.

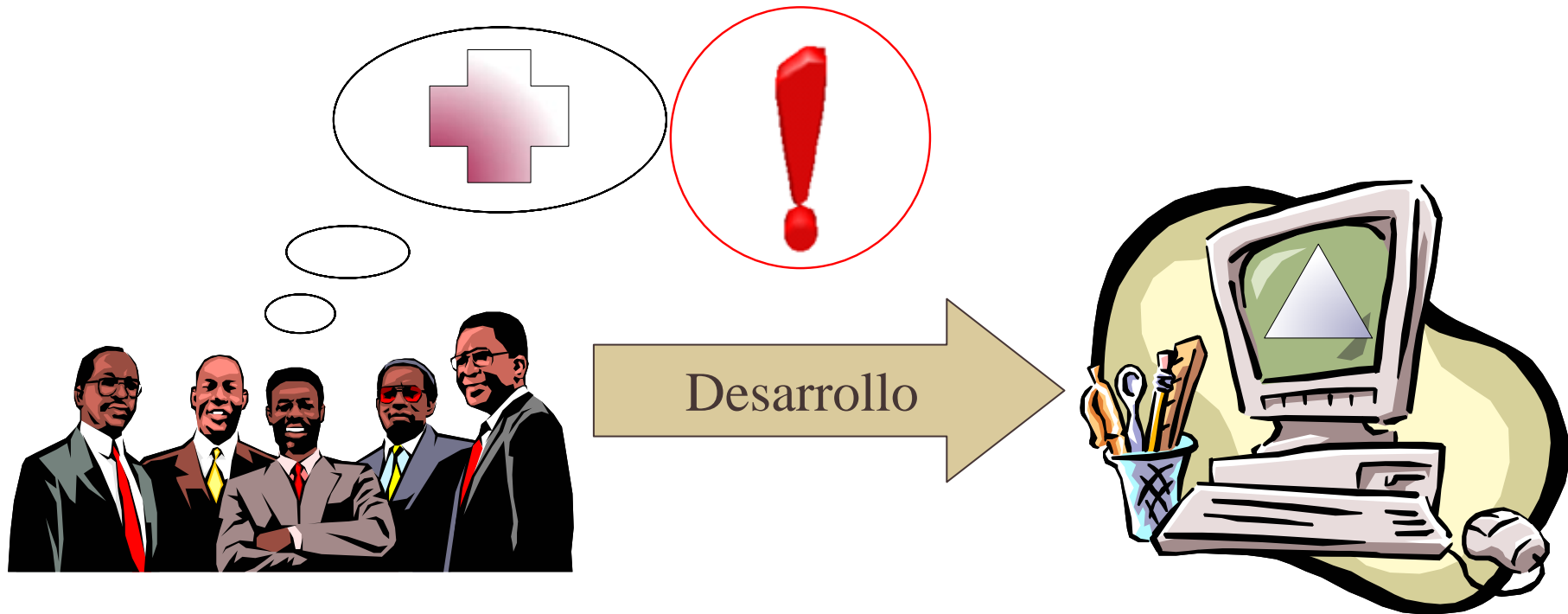
3. Análisis y conclusiones.

4. Trabajos futuros.

5. Publicaciones.

# *Problema.*

Asegurar la satisfacción de requisitos.

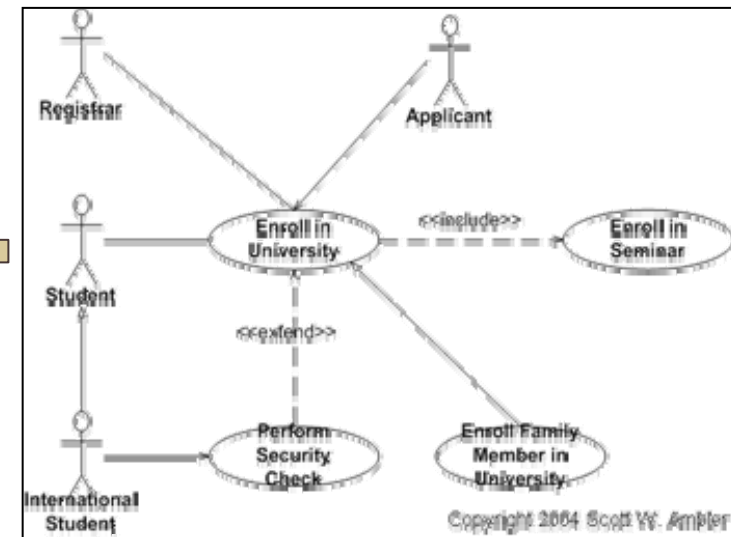
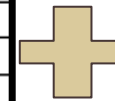


# Problema.

## Pruebas del sistema

Las pruebas del sistema verifican su funcionalidad a través de sus interfaces externas comprobando que sea la esperada según los **requisitos del sistema** [2004 SWEBOK. Guide to the Software Engineering BoK IEEE Computer Society]

RF-1	Entrar en el sistema	
Objetivos	OBJ-01: El sistema debe controlar la entrada en el sistema	
Descripción	El sistema deberá comportarse tal y como se describe en el siguiente caso de uso cuando el usuario quiera conectarse al sistema	
Secuencia normal	Paso	Acción
	1	El usuario conecta con la página de login del sistema.
	2	El sistema solicita autenticación
	3	El sistema recibe la conexión y muestra una página donde pide nombre y contraseña
	4	El usuario rellana el formulario introduciendo su nombre y pulsa el botón entrar
Excepción	Paso	Acción
	1	[1] Si el servidor no está activo o la página no carga adecuadamente se muestra un mensaje de error y termina el caso de uso.
	4	[2] Si la contraseña o el nombre no se introdujeron el sistema vuelve a solicitarlos indicando un error.
	4	[3] Si el nombre no se encuentra en la lista de nombres registrados el sistema vuelve a solicitarlo indicando un error
	4	[4] Si la contraseña suministrada no es válida, el sistema lo indica con un error y la pide de nuevo.



# *Problema*

---

**¿Existe algún proceso para guiar y automatizar la obtención de pruebas a partir de los requisitos?**



# *Trabajos anteriores.*

---

## Test Case Derived from Requirements Specifications

- 24 de abril de **2003**.
- 12 propuestas.
- Descripciones muy sistemáticas y ordenadas.
- **Ningún** análisis.
- No incluye casos prácticos.
- Propuesta más antigua: **1988**
- Propuesta más moderna: **2002**

# *Aportaciones originales*

---

- ❖ **Primer** análisis comparativo.
- ❖ Criterios de comparación.
- ❖ Casos **prácticos**.
- ❖ Propuesta de metodología **completa**.
- ❖ Compilación de referencias.

# *Aportaciones originales*

---

## *Test Case Derived from Requirements Specifications*

- 24 de abril de **2003**.
- 12 propuestas.
- Descripciones muy sistemáticas y ordenadas.
- Ningún análisis y conclusiones escuetas.
- No incluye casos prácticos.
- Propuesta más antigua: **1988**
- Propuesta más moderna: **2002**

## *Generación de pruebas del sistema a partir de la especificación funcional*

- 24 de mayo de **2005**.
- 12 propuestas.
- Descripciones sistemáticas y ordenadas.
- Análisis y conclusiones detallados
- Caso práctico.
- Propuesta más antigua: **1998**.
- Propuesta más moderna: **2004**.

# Índice.

---



1. Introducción.

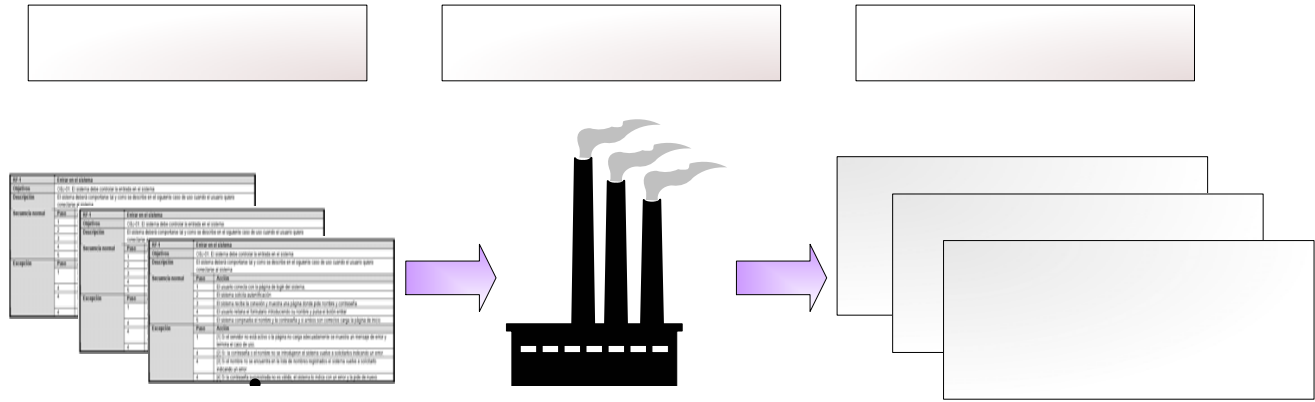
2. Comparativa de propuestas.

3. Análisis y conclusiones.

4. Trabajos futuros.

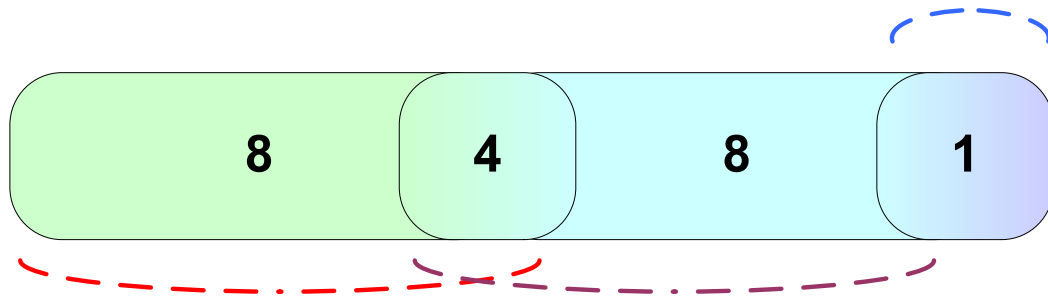
5. Publicaciones.

# Comparativa de propuestas



# *Comparativa de propuestas*

## **Propuestas estudiadas.**



### Selección de propuestas:

- Todas las encontradas (salvo Quasar).
- Coincidencias fortuitas.

# Propuestas Quasar

Año	Autores	Título
1988	Ostrand T. J. Balcer M. J.	The Category Partition Method for Specifying and Generating Functional Tests.
1998	Hsia P. Kung D.	Software Requirements and Acceptance Testing.
1998	Meyer S. Sandfoss R.	Applying Use-Case Methodology to SRE and System Testing.
1999	Binder R. V.	Extended Use Case Test Design Pattern.
1999	Carpenter P. B.	Verification of Requirements for Safety-Critical software.
1999	Collard R.	Test Design: Developing Test Cases from Use Cases.
1999	Cunning S. J. Rozenblit J. W.	Test Scenario Generation From a Structured Requirement Specification.
2000	Hindel, B. Hehn U.	Constructing Test Cases from Derived Requirements.
2002	Blacburn M. Busser R., Nauman A.	Interface-Driven Model-Based Test Automation.
2002	Ahlowalia N. Testing	Testing From Use Cases Using Path Analysis Technique.
2002	Pudipeddi H. V.	Understanding, Designing and Testing Use Cases.
2002	Wood D. Reis J.	Use Case Derived Test Cases.

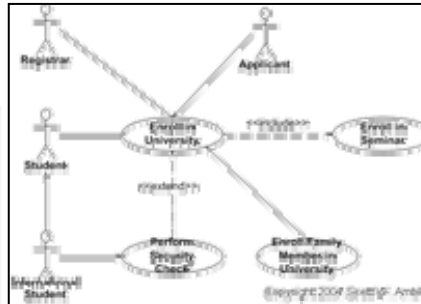
# *Propuestas I.I.*

<b>Año</b>	<b>Título</b>
2004	Requirements to Testing in a Natural Way
2003 / 2004	Requirements by Contract
2003 / 2004	PLUTO y Category Partition Method
2002	A UML-Based Approach to System Testing
2004	Derivation of Domain Test Scenarios from Activity Diagrams (TDE)
2000	Automated Test Case Generation from Dynamic Models
2002	Test Cases from Use Cases
1999	Scenario-Based Validation and Test of Software (SCENT)
2001	Requirement Base Testing

# Resumen de las propuestas



RF 1		Entrar en el sistema																														
Objetivo	Obj-C1. El sistema debe controlar si el usuario se ha registrado en el sistema	Objetivo	Obj-C1. El sistema debe controlar si el usuario se ha registrado en el sistema																													
Descripción	El sistema deberá comprobar la y cómo se describe en el siguiente caso de uso cuando el usuario quiere conectarse al sistema	Descripción	El sistema deberá comprobar la y cómo se describe en el siguiente caso de uso cuando el usuario quiere conectarse al sistema																													
Secuencia normal	<table border="1"> <thead> <tr> <th>Fase</th> <th>RF 1</th> <th>Entrar en el sistema</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Objetivo</td> <td>Obj-C1. El sistema debe controlar si el usuario se ha registrado en el sistema</td> </tr> <tr> <td>2</td> <td>Descripción</td> <td>El sistema deberá comprobar la y cómo se describe en el siguiente caso de uso cuando el usuario quiere conectarse al sistema</td> </tr> <tr> <td>3</td> <td>Objetivo</td> <td>Obj-C1. El sistema debe controlar si el usuario se ha registrado en el sistema</td> </tr> <tr> <td>4</td> <td>Descripción</td> <td>El sistema deberá comprobar la y cómo se describe en el siguiente caso de uso cuando el usuario quiere conectarse al sistema</td> </tr> </tbody> </table>	Fase	RF 1	Entrar en el sistema	1	Objetivo	Obj-C1. El sistema debe controlar si el usuario se ha registrado en el sistema	2	Descripción	El sistema deberá comprobar la y cómo se describe en el siguiente caso de uso cuando el usuario quiere conectarse al sistema	3	Objetivo	Obj-C1. El sistema debe controlar si el usuario se ha registrado en el sistema	4	Descripción	El sistema deberá comprobar la y cómo se describe en el siguiente caso de uso cuando el usuario quiere conectarse al sistema	<table border="1"> <thead> <tr> <th>Fase</th> <th>RF 1</th> <th>Entrar en el sistema</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Objetivo</td> <td>Obj-C1. El sistema debe controlar si el usuario se ha registrado en el sistema</td> </tr> <tr> <td>2</td> <td>Descripción</td> <td>El sistema deberá comprobar la y cómo se describe en el siguiente caso de uso cuando el usuario quiere conectarse al sistema</td> </tr> <tr> <td>3</td> <td>Objetivo</td> <td>Obj-C1. El sistema debe controlar si el usuario se ha registrado en el sistema</td> </tr> <tr> <td>4</td> <td>Descripción</td> <td>El sistema deberá comprobar la y cómo se describe en el siguiente caso de uso cuando el usuario quiere conectarse al sistema</td> </tr> </tbody> </table>	Fase	RF 1	Entrar en el sistema	1	Objetivo	Obj-C1. El sistema debe controlar si el usuario se ha registrado en el sistema	2	Descripción	El sistema deberá comprobar la y cómo se describe en el siguiente caso de uso cuando el usuario quiere conectarse al sistema	3	Objetivo	Obj-C1. El sistema debe controlar si el usuario se ha registrado en el sistema	4	Descripción	El sistema deberá comprobar la y cómo se describe en el siguiente caso de uso cuando el usuario quiere conectarse al sistema
Fase	RF 1	Entrar en el sistema																														
1	Objetivo	Obj-C1. El sistema debe controlar si el usuario se ha registrado en el sistema																														
2	Descripción	El sistema deberá comprobar la y cómo se describe en el siguiente caso de uso cuando el usuario quiere conectarse al sistema																														
3	Objetivo	Obj-C1. El sistema debe controlar si el usuario se ha registrado en el sistema																														
4	Descripción	El sistema deberá comprobar la y cómo se describe en el siguiente caso de uso cuando el usuario quiere conectarse al sistema																														
Fase	RF 1	Entrar en el sistema																														
1	Objetivo	Obj-C1. El sistema debe controlar si el usuario se ha registrado en el sistema																														
2	Descripción	El sistema deberá comprobar la y cómo se describe en el siguiente caso de uso cuando el usuario quiere conectarse al sistema																														
3	Objetivo	Obj-C1. El sistema debe controlar si el usuario se ha registrado en el sistema																														
4	Descripción	El sistema deberá comprobar la y cómo se describe en el siguiente caso de uso cuando el usuario quiere conectarse al sistema																														
Excepción	<table border="1"> <thead> <tr> <th>Fase</th> <th>RF 1</th> <th>Entrar en el sistema</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Objetivo</td> <td>Obj-C1. El sistema debe controlar si el usuario se ha registrado en el sistema</td> </tr> <tr> <td>2</td> <td>Descripción</td> <td>El sistema deberá comprobar la y cómo se describe en el siguiente caso de uso cuando el usuario quiere conectarse al sistema</td> </tr> <tr> <td>3</td> <td>Objetivo</td> <td>Obj-C1. El sistema debe controlar si el usuario se ha registrado en el sistema</td> </tr> <tr> <td>4</td> <td>Descripción</td> <td>El sistema deberá comprobar la y cómo se describe en el siguiente caso de uso cuando el usuario quiere conectarse al sistema</td> </tr> </tbody> </table>	Fase	RF 1	Entrar en el sistema	1	Objetivo	Obj-C1. El sistema debe controlar si el usuario se ha registrado en el sistema	2	Descripción	El sistema deberá comprobar la y cómo se describe en el siguiente caso de uso cuando el usuario quiere conectarse al sistema	3	Objetivo	Obj-C1. El sistema debe controlar si el usuario se ha registrado en el sistema	4	Descripción	El sistema deberá comprobar la y cómo se describe en el siguiente caso de uso cuando el usuario quiere conectarse al sistema	<table border="1"> <thead> <tr> <th>Fase</th> <th>RF 1</th> <th>Entrar en el sistema</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Objetivo</td> <td>Obj-C1. El sistema debe controlar si el usuario se ha registrado en el sistema</td> </tr> <tr> <td>2</td> <td>Descripción</td> <td>El sistema deberá comprobar la y cómo se describe en el siguiente caso de uso cuando el usuario quiere conectarse al sistema</td> </tr> <tr> <td>3</td> <td>Objetivo</td> <td>Obj-C1. El sistema debe controlar si el usuario se ha registrado en el sistema</td> </tr> <tr> <td>4</td> <td>Descripción</td> <td>El sistema deberá comprobar la y cómo se describe en el siguiente caso de uso cuando el usuario quiere conectarse al sistema</td> </tr> </tbody> </table>	Fase	RF 1	Entrar en el sistema	1	Objetivo	Obj-C1. El sistema debe controlar si el usuario se ha registrado en el sistema	2	Descripción	El sistema deberá comprobar la y cómo se describe en el siguiente caso de uso cuando el usuario quiere conectarse al sistema	3	Objetivo	Obj-C1. El sistema debe controlar si el usuario se ha registrado en el sistema	4	Descripción	El sistema deberá comprobar la y cómo se describe en el siguiente caso de uso cuando el usuario quiere conectarse al sistema
Fase	RF 1	Entrar en el sistema																														
1	Objetivo	Obj-C1. El sistema debe controlar si el usuario se ha registrado en el sistema																														
2	Descripción	El sistema deberá comprobar la y cómo se describe en el siguiente caso de uso cuando el usuario quiere conectarse al sistema																														
3	Objetivo	Obj-C1. El sistema debe controlar si el usuario se ha registrado en el sistema																														
4	Descripción	El sistema deberá comprobar la y cómo se describe en el siguiente caso de uso cuando el usuario quiere conectarse al sistema																														
Fase	RF 1	Entrar en el sistema																														
1	Objetivo	Obj-C1. El sistema debe controlar si el usuario se ha registrado en el sistema																														
2	Descripción	El sistema deberá comprobar la y cómo se describe en el siguiente caso de uso cuando el usuario quiere conectarse al sistema																														
3	Objetivo	Obj-C1. El sistema debe controlar si el usuario se ha registrado en el sistema																														
4	Descripción	El sistema deberá comprobar la y cómo se describe en el siguiente caso de uso cuando el usuario quiere conectarse al sistema																														



## Requisitos

1

# *Propuestas no consideradas.*

<b>Año</b>	<b>Título</b>
2003	Generating Tests from UML Specifications
2004	Towards Automated Support for Deriving Test Data from UML Statecharts
1999	Testing Object-Oriented Software Using the Category-Partition Method
2003	UML-Based Statistical Test Case Generation.
2001	Analysis and Testing of Web Applications.
2004	AGEDIS

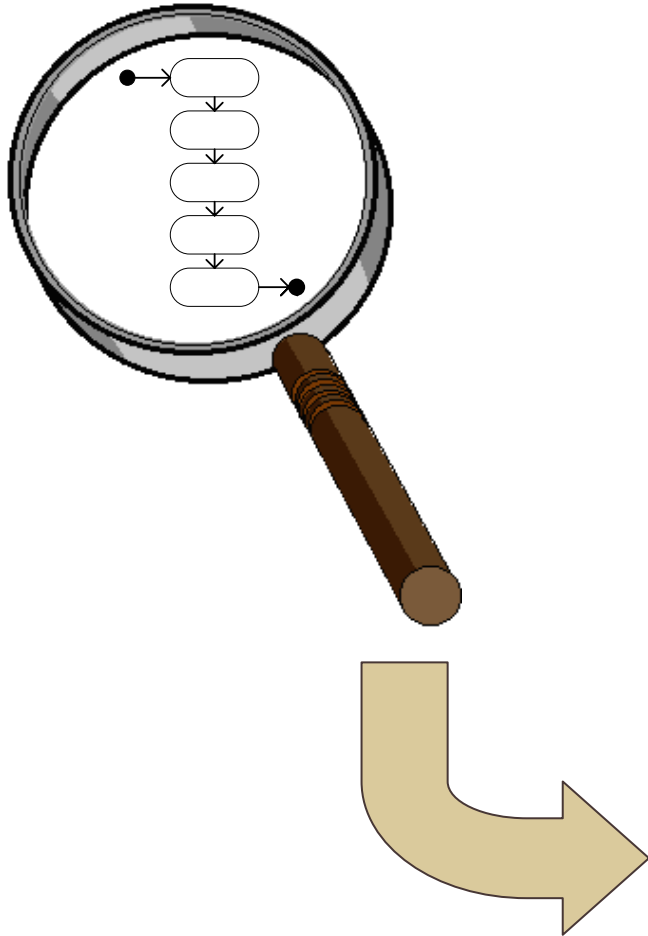
# *Comparativa de propuestas*

---

¿Qué evaluamos?.

1. Sistematización y **automatización**.
2. Herramientas de soporte.
3. Calidad de la documentación.
4. Respeto por los **estándares**.
5. Resultado **ejecutable**.

# Análisis y conclusiones



**Cinco** grupos de características.

1. Información necesaria	3
2. Modelo de comportamiento	4
3. Descripción del proceso	9
4. Aspectos complementarios	3
5. Casos prácticos	2
<b>Total:</b>	<b>21</b>

Extensión de los  
casos de uso con  
contratos

Construcción  
del modelo de  
ejecución

Selección del  
criterio de  
cobertura

# Características I

- ❖ Origen. *Dominio: Set{Natural language, Use cases, Several}*
- ❖ Nueva notación. *Dominio: Boolean*
- ❖ Notación gráfica. *Dominio: Boolean*
- ❖ Casos prácticos. *Dominio: Boolean*
- ❖ Uso de estándares. *Dominio: Boolean*
- ❖ Criterio de cobertura.  
*Dominio: Set{Enum{Path analysis, Category-Partition}}*
- ❖ Valores de prueba. *Dominio: Boolean*
- ❖ Optimización de pruebas. *Dominio: Boolean*
- ❖ Dependencias de casos de uso. *Dominio: Boolean*

# Características II

- ❖ Modelo de comportamiento.

*Dominio: Enum{Finite State Machines, Flow Diagrams, Use cases execution model, Activity Diagram, Category choices and restrictions}*

- ❖ Momento de comienzo. *Dominio: Enum {Requirements Elicitation, System Analysis}*

- ❖ Resultados. *Dominio: Set{Enum{ Test scripts, Sequence of interactions among use cases, Instances of use cases}}*

- ❖ Calidad de la documentación. *Dominio: Enum{High. Medium, Low}*

- ❖ Implementación de las pruebas. *Dominio: Boolean*

# Categorías III

---

❖ Herramientas de soporte.

*Dominio: Enum{No, Process supporting, Test case generation}*

❖ Ejemplos de aplicación. *Dominio: Boolean*

❖ Experiencia necesaria. *Dominio: Enum{High, Medium, Low}*

❖ Prioridad de casos de uso. *Dominio: Boolean*

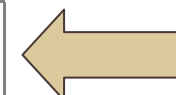
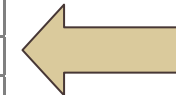
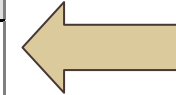
❖ Propuesta activa. *Dominio: Enum{Yes, No, ?}*

❖ Formato de casos de uso. *Dominio: Enum{Own format, Reference, Not indicated}*

❖ Año de publicación.

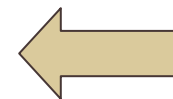
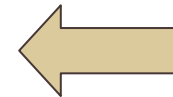
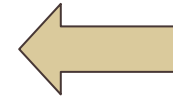
# Comparativa I

	ATCGDM	SCENT	TCUC	UCPA	PLUTO	RBC
<b>Origen</b>	Requisitos	Necesidades	Requisitos	Requisitos	Casos de uso (17)	Diagrama de casos de uso
<b>Nueva notación</b>	No	Sí	No	Sí	No	Sí (11)
<b>Notación gráfica</b>	Sí	Sí	No	Sí	No	Sí
<b>Casos practicos</b>	No	Sí	No	No	No	No
<b>Uso de estándares</b>	Sí	No	No	No	No	Sí
<b>Herramientas de soporte</b>	Sí (14)	No	No	No	Sí (15)	Sí
<b>Ejemplos de aplicación</b>	Sí	Sí	Sí	Sí	Sí	Sí
<b>Criterio de cobertura</b>	Varios (6)	Análisis de caminos	Análisis de caminos	Análisis de caminos	Particiones de categories	Varios
<b>Valores de prueba</b>	No	No	No	No	Sí	No
<b>Optimización de pruebas</b>	No	No	No	Sí	Sí	Sí
<b>Dependencias de casos de uso</b>	Sí	Sí	No	No	Sí	Sí
<b>Construcción del modelo</b>	Sí	Sí	No	Sí	No	Sí
<b>Prioridad de casos de uso</b>	No	No	No	Sí	Sí	No
<b>Propuesta activa (5)</b>	?	No	?	?	?	?
<b>Momento de comienzo</b>	Elicitación de requisitos	Creación de scenarios	Elicitación de requisitos	Elicitación de requisitos	Elicitación de requisitos	Elicitación de requisitos
<b>Resultados</b>	Secuencias de transiciones	Pruebas en lenguaje natural	Pruebas en lenguaje natural	Pruebas en lenguaje natural	Pruebas en lenguaje natural	Secuencias de casos de uso
<b>Calidad de la documentación</b>	Media	Media	Media	Media	Media	Media
<b>Pasos</b>	7	15+3 (1)	3	5	8	4
<b>Año publicación</b>	2000	1999 / 2000	2002	2002	2004	2000 / 2001



# Comparativa II

	TO TEM	RBT	EUC	AT	UCDTC	IDEUML
<b>Origen</b>	Varios (13)	Requisitos	Casos de uso	Necesidades	Casos de uso	Casos de uso
<b>Nueva notación</b>	Sí (12)	No	No	Sí (2)	No	Sí (3)
<b>Notación gráfica</b>	Sí	Sí	No	Sí	No	Sí
<b>Casos prácticos</b>	No	No	No	Sí	No	No
<b>Uso de estándares</b>	Sí	Sí (8)	No	No	Sí	Sí
<b>Herramientas de soporte</b>	No	Sí (9)	No	No	No	Sí (9)
<b>Ejemplos de aplicación</b>	Sí	No	Sí	Sí	Sí	Sí
<b>Criterio de cobertura</b>	Varios	No (10)	Análisis de caminos	Varios	Análisis de caminos	Varios
<b>Valores de prueba</b>	No	No	Sí	No	No	Sí
<b>Optimización de pruebas</b>	No	No	IEEE	No	No	No
<b>Dependencias de casos de uso</b>	Sí	No	No	Sí	No	No
<b>Construcción del modelo</b>	Sí	Sí	No	Sí	No	Sí
<b>Prioridad de casos de uso</b>	Sí	No	No	No	No	Sí
<b>Propuesta activa (5)</b>	No	?	?	?	?	Sí
<b>Momento de comienzo</b>	Análisis del sistema	Elicitación de requisitos	Requisitos estables	Después del análisis de escenarios (16)	Elicitación de requisitos	Elicitación de requisitos
<b>Resultados</b>	Modelo de prueba y Secuencias de transiciones	Pruebas en lenguaje natural	Pruebas en lenguaje natural	Modelo de prueba	Pruebas en lenguaje natural	Scripts de prueba ejecutables
<b>Calidad de la documentación</b>	Alta(7)	Baja	Media	Baja	Baja	Media (4)
<b>Pasos</b>	8	12	4	6	2	3
<b>Año publicación</b>	2002	2001	1999	1997	2002	2004



# Índice.

---

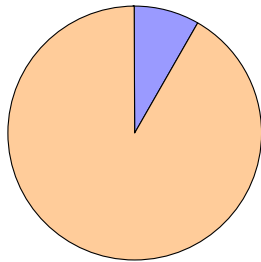


1. Introducción.
2. Comparativa de propuestas.
3. Análisis y conclusiones.
4. Trabajos futuros.
5. Publicaciones.

# Análisis y conclusiones

---

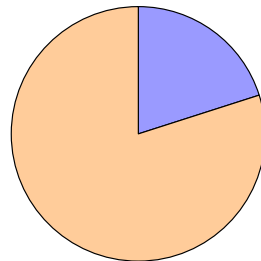
## Resultados I.



1/12 tiene documentación completa.

1/12 no hace referencia a la fase de elicitación.

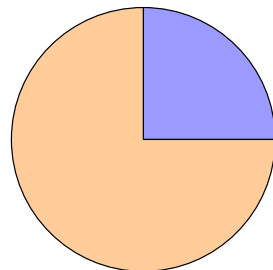
---



3/12 incluyen generación de valores de prueba

3/12 generan pruebas ejecutables (con herramienta propietaria).

---



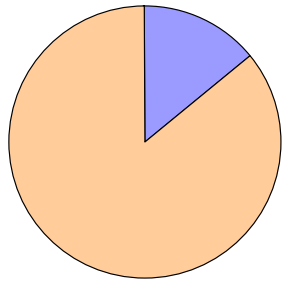
4/12 cuentan con herramientas de soporte.

4/12 indican como reducir el número de pruebas obtenidas

# *Análisis y conclusiones*

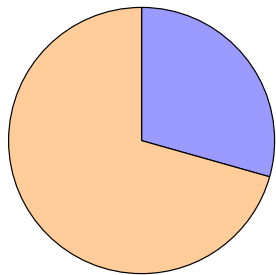
---

## **Resultados II.**



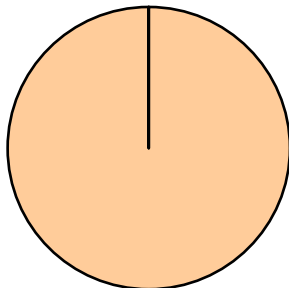
2/12 nos han informado de que siguen en activo

---



5/12 trabajan con diagramas UML

---



0/12 indican cómo medir la calidad de las pruebas generadas.

0/12 ofrece estudios empíricos de su eficacia.

# *Análisis y conclusiones*

---

## Conclusiones.

- **No** existe la propuesta **completa**.
- La documentación es **escasa** y los trabajos son **aislados**.
- Falta de **sistematización** y de automatización.
- Falta de **herramientas** que soporten las propuestas.
- Falta de **estudios** empíricos.

# Índice.

---

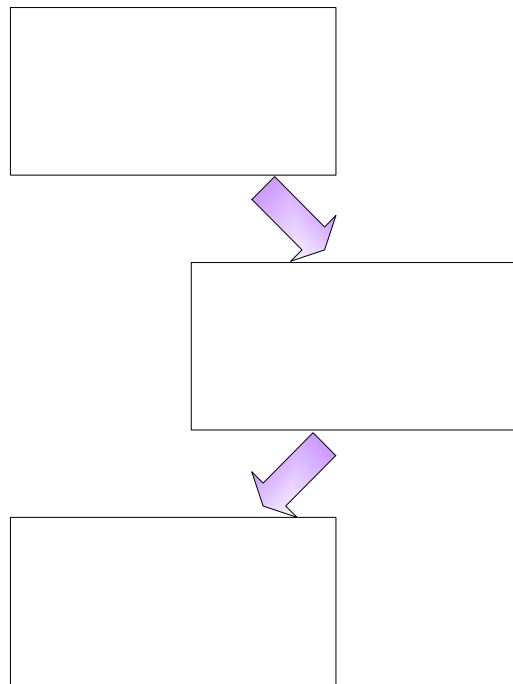


1. Introducción.
2. Comparativa de propuestas.
3. Análisis y conclusiones.
4. Trabajos futuros.
5. Publicaciones.

# Trabajo futuro.

---

Estudio de las características de una propuesta completa.



# *Aspectos adecuada e inadecuadamente tratados*

## **Aspectos adecuadamente tratados:**

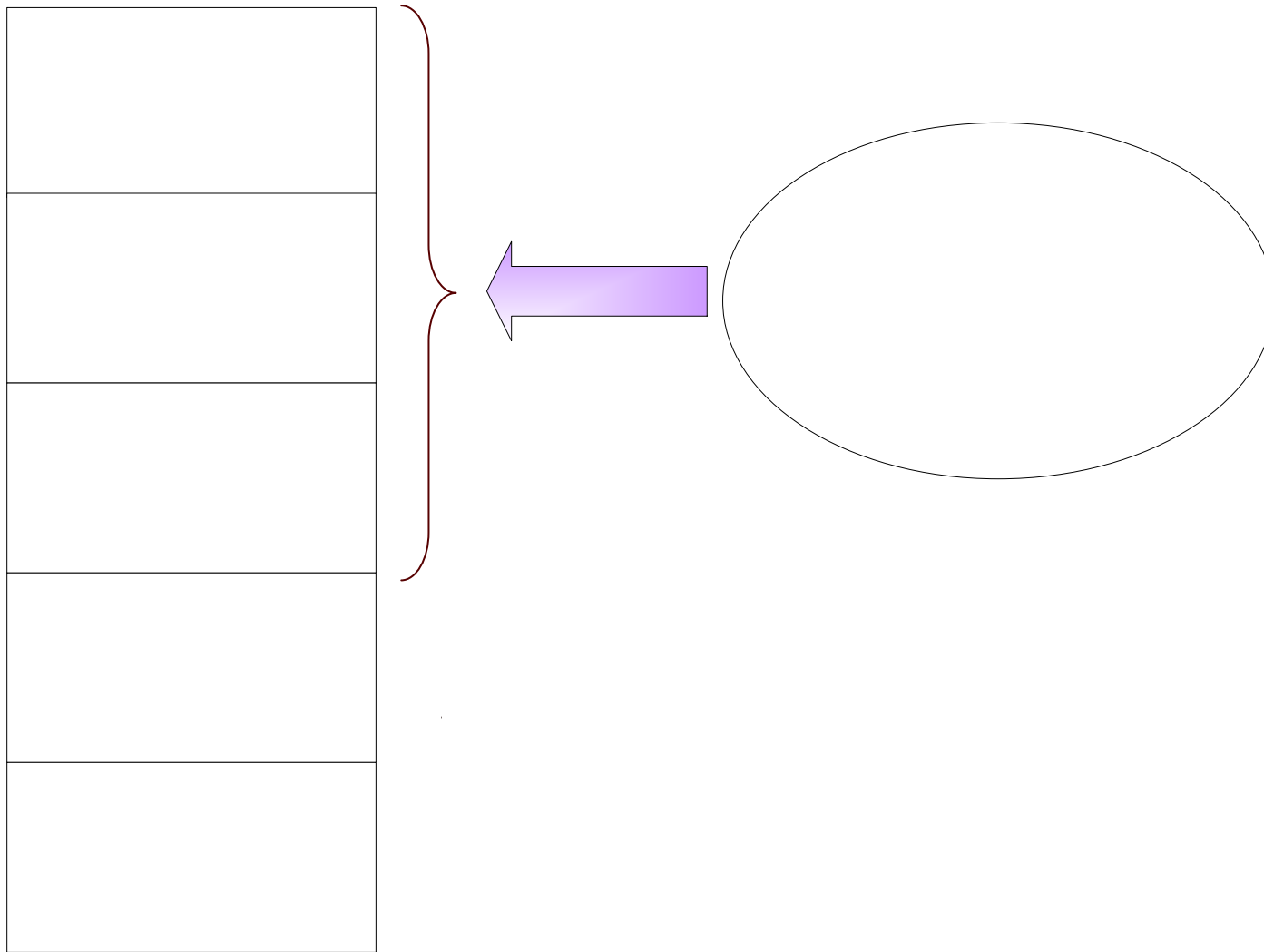
- Construcción de un modelo de comportamiento a partir de los requisitos.
- Generación de secuencias de casos de uso
- Aplicaciones del análisis
- Generación de valores

## **Aspectos inadecuadamente tratados:**

- Herramientas de soporte.
- Generación de pruebas ejecutables.
- Cálculo de cobertura.
- Estudios empíricos.
- Oracles de prueba.
- Obtención del conjunto mínimo.

# *Trabajo futuro*

---



# Índice.

---



1. Introducción.
2. Comparativa de propuestas.
3. Análisis y conclusiones.
4. Futuros trabajos.
5. Publicaciones.

# Publicaciones

Desde mayo de 2.004 : **26** trabajos / **19** publicados

Trabajos más relevantes (1/2):

<i>Año</i>	<i>Título</i>	<i>Reunión</i>
<b>2004</b>	Generación de casos de prueba a partir de requisitos funcionales. Descripción y análisis de propuestas.	<i>ICSTEST®-E Bilbao</i>
<b>2004</b>	Aplicando técnicas de testing en sistemas para la difusión patrimonial	<i>Turitec Málaga</i>
<b>2004</b>	Análisis de propuestas para la generación de casos de prueba para el control de calidad.	<i>Avances en gestión de proyectos y calidad del software. Salamanca</i>
<b>2004</b>	Métodos de Testing Sobre la Ingeniería de Requisitos Web de NDT.	<i>Conferencia Ibero-Americana IADIS WWW/Internet. Madrid.</i>
<b>2004</b>	Comparative analysis of methodological proposes to systematic generation of system test cases from system requirements.	<i>Third Workshop on System Testing and Validation. Paris</i>

# Publicaciones

## Trabajos más relevantes (2/2):

<i>Año</i>	<i>Título</i>	<i>Reunión</i>
<b>2004</b>	Aplicación de la programación orientada a aspectos en el diseño e implementación de pruebas funcionales.	<i>JISBD. Workshop de aspectos. Málaga</i>
<b>2005</b>	Pruebas Funcionales y de Carga en Aplicaciones Web.	<i>MundoInternet. Madrid (seleccionado para exposición)</i>
<b>2005</b>	Analysis of proposals for the generation of system test cases from system requirements	<i>CAiSE Forum. Porto</i>
<b>2005</b>	A practical approach to system testing in web systems	<i>14 International Conference on Information Systems Development. Suecia</i>
<b>2005</b>	Testing web applications in practice	<i>WWV. Valencia</i>

# *Publicaciones*

---

## Difusión de los trabajos.

- Sao Paulo, Unicam (*Brasil*).
- Departamento de Informática (*Oviedo*).
- ISEE (*Skövde University - Suecia*).
- Institut für Informatik Programmierung und Softwaretechnik (*Manchen University – Alemania*).

# *Epílogo.*

---



I've seen several papers on deriving tests from use cases - they've all been pretty obvious, derivative, and repetitious [*A. Jefferson Offutt*].



Departamento de  
Lenguajes y Sistemas Informáticos



Universidad de Sevilla

# *Generación de pruebas de sistema a partir de la especificación funcional*

*Javier Jesús Gutiérrez Rodríguez*

Tutores: Doctor D. Manuel Mejías  
Doctora Dña. Maria José Escalona