

requisiti e use cases

a.a. 2003-2004

1

furps+

- sistema di categorizzazione dei requisiti
- utile come checklist
 - per essere sicuri di aver analizzato tutti i risvolti del problema
- significato dell'acronimo:
 - Funcional
 - Usability
 - Reliability
 - Performace
 - Supportability
 - più altro...

2

furps+: F come “functional”

- i requisiti funzionali descrivono cosa il sistema dovrà fare
 - sono di gran lunga i più importanti rispetto a tutti gli altri
 - sono sempre presenti
 - esistono metodologie che aiutano a individuarli

3

furps+: “URPS”

- Usability
 - quanto l’utente gradisce l’interfaccia verso il sistema?
 - difficile da misurare ma molto importante
- Reliability
 - frequency of failure, recoverability, predictability
- Performace
 - quality of service
- Supportability
 - adaptability, maintainability, internationalization, configurability

4

furps+: “+”

- Implementation
 - risorse necessarie, linguaggi, tools, hw, ecc.
- Interface
 - interfacce con sistemi già esistenti
- Operations
 - system management
- Packaging
 - programmi di installazione automatica
- Legal
 - licensing, leggi da rispettare, etc.

5

casi d'uso

(use cases)

- sono un modo per esprimere requisiti funzionali
- descrivono (idealmente tutti) i possibili modi di interazione (o scenari) tra utente e sistema
 - una sequenza di azioni dell'utente e/o del sistema
- gli use cases si documentano con dei testi strutturati
- in UP: requirements discipline

- **ATTENZIONE:** per quanto voi siate bravi e la metodologia usata rigorosa gli use cases non saranno perfetti!
 - sono comunque molto importanti
 - vengono raffinati nel tempo (durante le prime iterazioni)

6

casi d'uso: concetti base

- actor
- scenario
- use case instance
- vari formati e livelli di dettaglio
 - brief, casual, fully dressed
- granularità
 - elementary business process: “a task performed by **one person** at **one time** in response to **a business event** which **adds measurable business value** and leaves the **data in a consistent state**” (da C. Larman)
 - possibili violazioni: sub use cases

7

casi d'uso: concetti base

- utente-sistema
 - non necessariamente l'utente è una persona
 - non necessariamente il sistema è software o solo software
 - il confine utente e sistema tra viene messo dove è conveniente
- un success scenario e molte extensions
 - uno è lo scenario principale
 - molte sono le varianti
 - esempio importante: condizioni di errore
 - recuperabile
 - non recuperabile
- precondizioni e postcondizioni
- frequenza
- altri requisiti associati:
 - vedi **furps+**

8

use case stili

- brief
 - informale, solo lo scenario principale
- casual
 - informale, vari scenari
- fully dressed
 - tutti i dettagli

9

use case: fully dressed template (esempio)

- name
- stakeholders interests:
 - specifica *chi* ha interesse a certe caratteristiche di questo use case è *quali interessi* ha

- | | |
|--|---------------|
| <ul style="list-style-type: none">• contract<ul style="list-style-type: none">– preconditions– postconditions• main success scenario (basic flow)<ul style="list-style-type: none">– uno solo, evita passi condizionali e ripetitivi• extensions (alternative flows)<ul style="list-style-type: none">– tutte le varianti | furps+ |
|--|---------------|

- special requirements
 - vedi **furps+**
- technology and data variation list
 - aspetti tecnici e tecnologici come dispositivi di input, formati dati di input
- frequency of occurrence
- open issues

10

system sequence diagram

può essere utile associare agli use case una rappresentazione diagrammatica degli scenari principi tramite sequence diagram attore-systema

