

---

# Analisi e Specifica dei Requisiti

---

# Riferimenti

---

- Ian Sommerville, Ingegneria del Software, capitoli 6,7

# Obiettivi

---

- Introdurre i concetti di requisiti utente e di requisiti di sistema
- Descrivere i requisiti funzionali e non-funzionali
- Spiegare come i requisiti software possano essere organizzati in un documento di specifica dei requisiti

# Ingegneria dei requisiti

---

- Il processo condotto per stabilire i servizi che il cliente richiede al sistema, i vincoli operativi e quelli per lo sviluppo.
- I requisiti sono la descrizione di tali servizi e dei relativi vincoli.

# Che cos'è un Requisito?

---

- Può variare tra una **descrizione astratta**, ad alto livello di qualcosa che il sistema farà, o di un suo vincolo, ed una **dettagliata specifica funzionale** matematica
- Ciò è inevitabile perchè un requisito svolge una duplice funzione:
  - Può essere la base per una gara fra potenziali sviluppatori: quindi deve essere aperto ad interpretazioni diverse
  - Può essere la base del contratto stesso- quindi deve essere definito in dettaglio
  - Entrambi si possono chiamare requisiti

# Necessità di diversi livelli di astrazione nei requisiti (Davis)

---

“Se una compagnia vuole dare in appalto un grande progetto di sviluppo software, deve definire le sue esigenze in modo abbastanza astratto, da non predefinire alcuna soluzione.

I requisiti devono essere scritti in modo da consentire a vari appaltatori di interpretare le esigenze e proporre vari metodi per soddisfarle.

Quando l'appalto è stato assegnato, l'appaltatore deve scrivere per il cliente una descrizione del sistema molto dettagliata, in modo che il cliente possa capire e validare ciò che il sistema farà.

Entrambi i documenti si chiameranno *documenti dei requisiti* del sistema.”

# Analisi e Specifica dei Requisiti

---

- Analisi dei requisiti
  - Processo di valutazione delle necessità del committente del software, con redazione di un documento di analisi dei requisiti, di solito in linguaggio naturale
  - Viene redatto al termine di una campagna di interviste svolte dagli analisti presso il committente
- Specifica dei requisiti
  - Processo di schematizzazione dei requisiti di un sistema, con redazione di un documento formattato rispetto ad uno standard
  - E' la base del contratto di fornitura

# Tipi di Requisiti

---

- **Requisiti Utente**
  - Frasi in linguaggio naturale (e diagrammi) relativi ai servizi che il sistema fornisce e i suoi vincoli operativi. Scritti per i clienti.
- **Requisiti di Sistema**
  - Un documento strutturato che fornisce una descrizione dettagliata dei servizi del sistema. Può essere parte del contratto fra cliente e sviluppatore



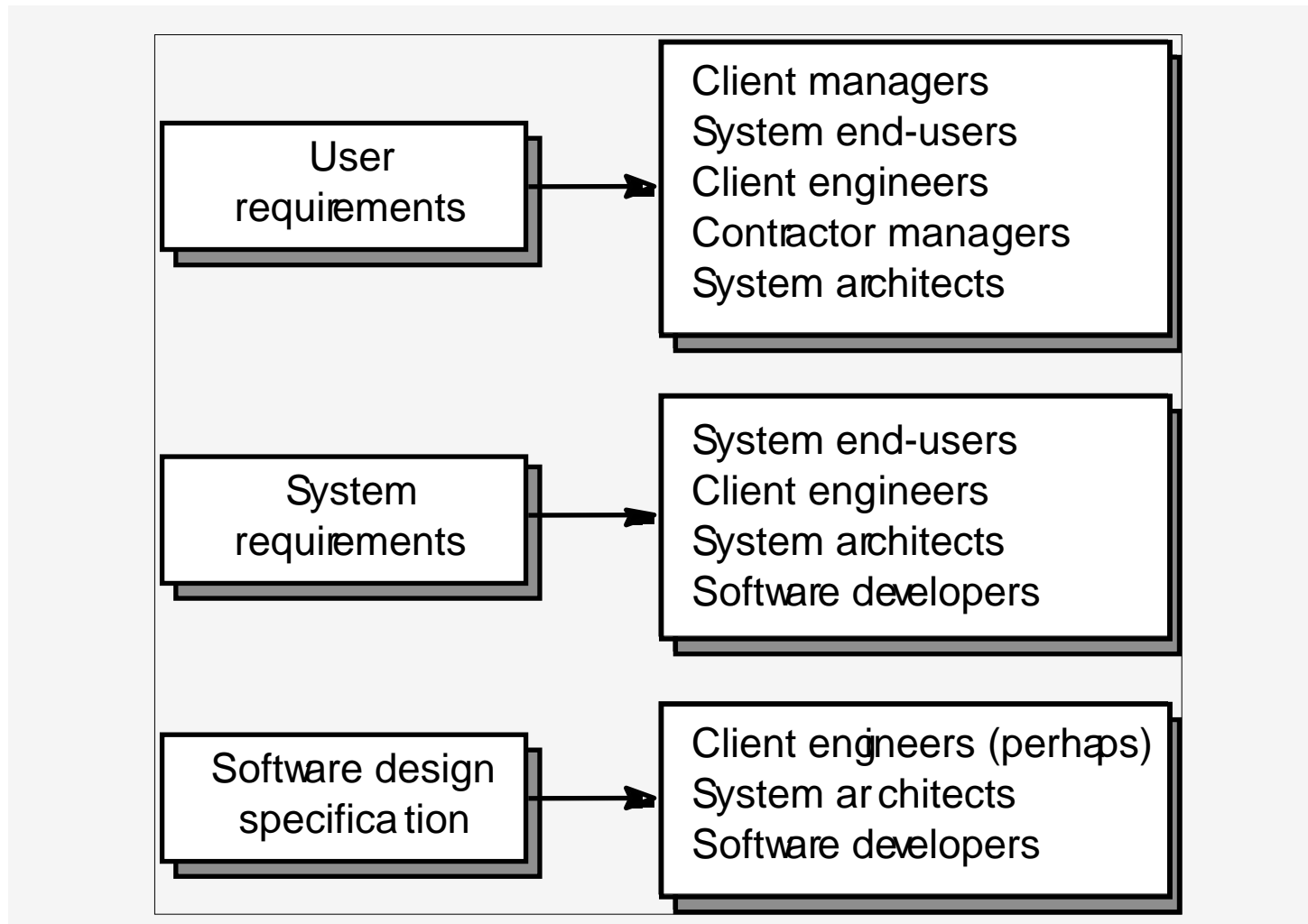
# Definizioni e specifiche dei requisiti

---

- Es. di definizione di un requisito utente:
  - Il software deve fornire un mezzo per rappresentare ed accedere a file esterni creati da altri tools.
- Es. di Specifiche dei requisiti di sistema
  1. L'utente deve poter definire il tipo dei file esterni
  2. Ciascun tipo di file esterno deve avere uno strumento associato
  3. Ciascun tipo di file esterno deve essere rappresentato con una sua icona sul display utente
  4. Quando un utente clicca sull'icona di un file, l'effetto sarà di attivare lo strumento associato al tipo di quel file
  5. ...

# Requirements readers

---



# Requisiti Funzionali e Non-funzionali

---

- **Requisiti Funzionali**
  - Frasi che descrivono ciò che il sistema dovrà fare, come reagirà agli input e si comporterà in varie situazioni.
- **Requisiti Non-funzionali**
  - Vincoli sui servizi offerti dal sistema (come vincoli sui tempi di risposta, ...) o a cui sottostare durante lo sviluppo → Requisiti di qualità
- **Requisiti di dominio**
  - Requisiti che derivano dal dominio di applicazione del sistema e ne riflettono le caratteristiche ed i limiti.

# Requisiti Funzionali

---

- Descrivono le funzionalità e i servizi offerti.
- Anche in questo caso, due livelli di astrazione:
  - Requisiti funzionali utente: frasi ad alto livello su ciò che il sistema farà
  - Requisiti funzionali di sistema: descrizioni dettagliate dei servizi.

# Un esempio: Il sistema LIBSYS

---

- É un sistema che fornisce un'interfaccia unica per navigare in un insieme di database di articoli.
- Gli utenti possono cercare, scaricare, ordinare, stampare tali articoli per motivi di studio.

# Esempi di requisiti funzionali utente

---

1. L'utente dovrà essere in grado di cercare un documento o in tutto il database o in un sottoinsieme di esso.
2. Il sistema dovrà fornire all'utente appropriati visualizzatori per leggere tutti i documenti presenti nel database.
3. Ad ogni ordine di articolo deve essere assegnato un identificatore univoco (ORDER\_ID) che l'utente deve poter copiare nella propria area di salvataggio.

*Diversi livelli di dettaglio (es. 1 e 3)*

# Imprecisioni nei requisiti

---

- Requisiti imprecisi o ambigui possono essere interpretati in modi diversi da sviluppatori ed utenti.
- Es.: il termine ‘appropriati visualizzatori’ nel req. 2
  - L’utente intendeva avere specifici visualizzatori per ogni tipo di documento;
  - Lo sviluppatore ha interpretato che dovrà fornire solo un generico visualizzatore di testi.

# Completezza e Consistenza dei requisiti

---

- I requisiti devono essere completi e consistenti
- Completezza
  - Tutti I requisiti richiesti devono essere presenti.
- Consistenza
  - Non ci dovrebbero essere definizioni contraddittorie.
- Nella realtà è facile commettere errori o omissioni, soprattutto per sistemi complessi!



# Requisiti Non-funzionali

---

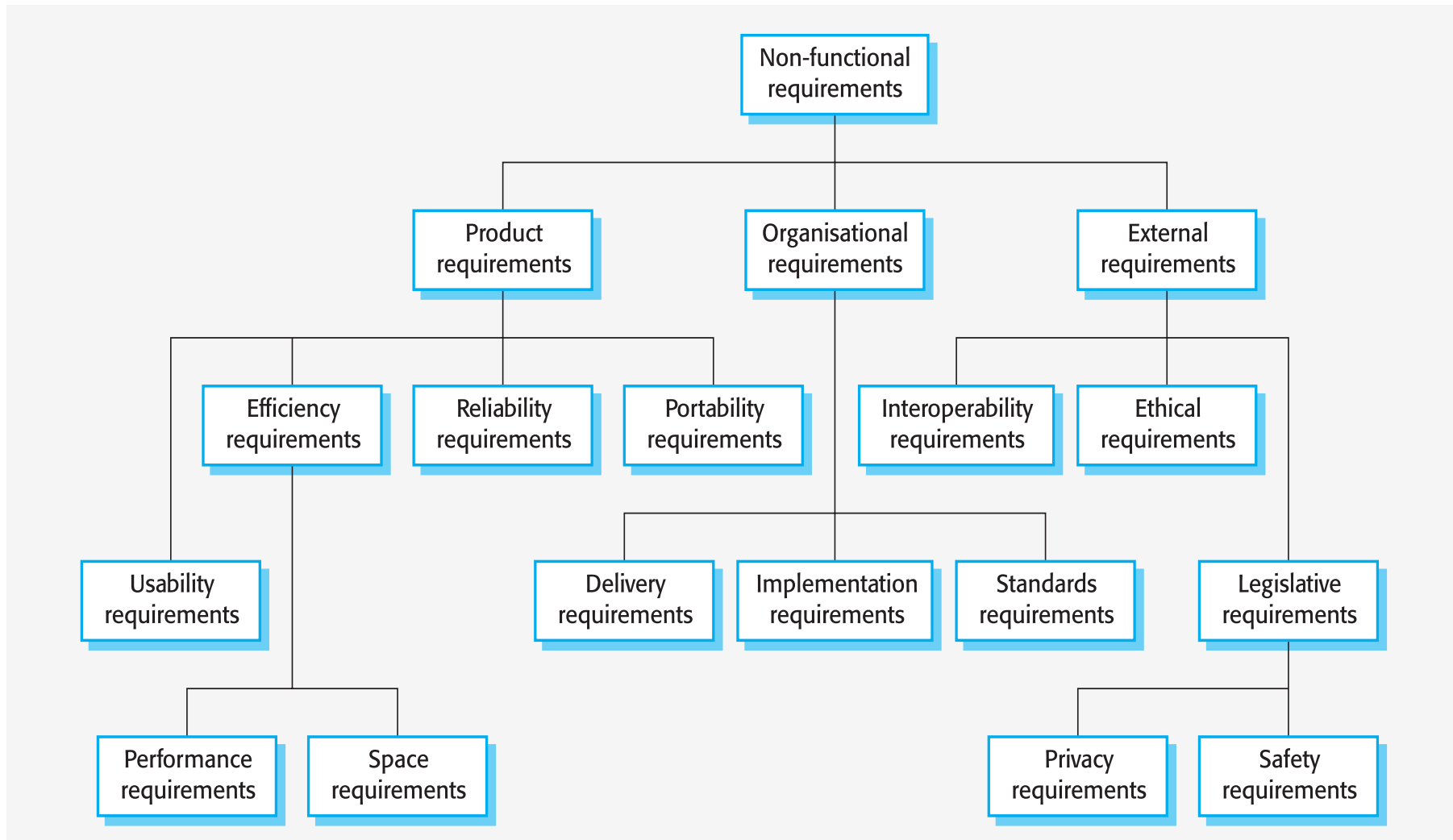
- Definiscono o limitano le proprietà del sistema (es. Affidabilità, tempi di risposta, memoria occupata...)
- Possono vincolare anche il processo di sviluppo da adottare (es. Uso di particolari standard per la documentazione, su sistemi CASE, linguaggi di codifica, metodi di sviluppo)
- Possono essere più critici dei req. funzionali: se non sono soddisfatti, il sistema è inutile.

# Tipi di requisiti non-funzionali

---

- **Requisiti del prodotto**
  - Specificano il comportamento del prodotto (usabilità, efficienza, affidabilità, portabilità).
- **Requisiti organizzativi**
  - Derivano dalle politiche e procedure dell'organizzazione del cliente e dello sviluppatore (es. Standard di processo da usare, piattaforme, requisiti di consegna, etc.)
- **Requisiti esterni**
  - Derivano da fattori esterni al sistema e al suo processo di sviluppo (come i requisiti di interoperabilità, legislativi, etici, etc.)

# Tipi di requisiti non-funzionali



# Esempi di requisiti non funzionali

---

- **Requisiti del Prodotto**
  - 8.1 L'interfaccia utente di LIBSYS deve essere implementata come semplice pagina HTML senza frame o Java applets (per garantirne l'accessibilità).
- **Requisiti organizzativi**
  - 9.3.2 Il processo di sviluppo e I documenti consegnati devono essere conformi alle norme XYZCo-SP-STAN-95.
- **Requisiti esterni**
  - 7.6.5 Il sistema non deve rivelare agli operatori del sistema alcuna informazione personale sui clienti, tranne il nome ed un numero di riferimento.

# Verificabilità dei requisiti

---

- I Requisiti non-funzionali possono essere difficili da definire precisamente, e quindi difficili da verificare.
- L'utente li specifica come obiettivi
  - Es. La facilità d'uso
- Requisito verificabile
  - Una frase che contiene qualche misura che potrà essere oggettivamente verificata.
-

# Esempi

---

- **Un obiettivo di sistema**
  - Il sistema dovrebbe essere usato facilmente da controllori esperti e dovrebbe minimizzare gli errori.
- **Un requisito non-funzionale verificabile**
  - I controllori esperti dovranno essere capaci di usare tutte le sue funzioni dopo due ore di addestramento. Dopo tale addestramento, il numero medio di errori fatti dagli utenti non dovrà essere superiore a due errori al giorno.

# Misure per requisiti non-funzionali

---

Property	Measure
Speed	Processed transactions/second User/Event response time Screen refresh time
Size	K Bytes Number of RAM chips
Ease of use	Training time Number of help frames
Reliability	Mean time to failure Probability of unavailability Rate of failure occurrence Availability
Robustness	Time to restart after failure Percentage of events causing failure Probability of data corruption on failure
Portability	Percentage of target dependent statements Number of target systems

# Contraddizioni fra i Requisiti

---

- I diversi requisiti non-funzionali possono essere in contraddizione fra loro, soprattutto in sistemi complessi (es. Tempi di risposta e portabilità)
- Es.: Sistemi aerospaziali
  - Per minimizzare il peso, il numero di chips a bordo dovrebbe essere minimo e la memoria complessiva non superiore a 4Mbyte
  - Un vincolo potrebbe richiedere di usare il linguaggio ADA (adatto a sistemi critici real time)
  - Purtroppo può accadere che non si possano avere tutte le funzioni richieste in 4 Mbyte. Quale requisito dovrà essere soddisfatto fra i due?



# Requisiti di Dominio

---

- Derivano dal dominio di applicazione, piuttosto che da necessità degli utenti.
- I requisiti di dominio possono essere nuovi requisiti funzionali, vincoli su altri requisiti funzionali, o possono delineare calcoli da effettuarsi.
- Se non sono soddisfatti, il sistema potrebbe essere inutilizzabile.

# Requisiti di dominio per il sistema LibSys

---

- Deve esserci una interfaccia utente uniforme per tutti i database basata sullo standard Z39.50.
- A causa delle restrizioni sul copyright, alcuni documenti devono essere cancellati immediatamente dopo l'arrivo. A seconda delle richieste dell'utente, questi documenti potranno o essere stampati localmente sul server del sistema per l'invio manuale all'utente, oppure devono essere inoltrati presso una stampante di rete.

# Problemi dei requisiti di dominio

---

- **Comprensibilità**
  - I requisiti sono espressi nel linguaggio del dominio;
  - Tale linguaggio potrebbe non essere immediatamente comprensibile per gli ingegneri del software.
- **Implicitezza**
  - Gli specialisti del dominio conoscono così bene l'area, da lasciare fuori dai requisiti informazioni che potrebbero sembrare ovvie (e invece non lo sono...).

# Requisiti Utente

---

- Dovrebbero descrivere requisiti funzionali e non-funzionali in modo da essere comprensibili ad utenti del sistema privi di dettagliate conoscenze tecniche.
- Non devono descrivere caratteristiche di progettazione, ma solo il comportamento esterno.
- Vengono definiti usando il Linguaggio Naturale, tabelle e diagrammi che sono comprensibili a tutti gli utenti.

# Problemi del linguaggio naturale

---

- Mancanza di chiarezza
  - È difficile usare il linguaggio in modo preciso e non ambiguo, senza rendere il documento lungo e difficile da leggere.
- Confusione dei requisiti
  - Le varie tipologie di requisiti potrebbero non essere facilmente distinguibili.
- Mescolanza dei requisiti
  - Diversi requisiti potrebbero essere espressi insieme come un singolo requisito.

# Un requisito di LIBSYS

---

**4..5** LIBSYS deve fornire un sistema di contabilità finanziaria che memorizzi tutti i pagamenti fatti dagli utenti del sistema. I gestori del sistema possono configurarlo in modo che gli utenti abituali possano ricevere prezzi scontati.

*Confonde due tipi di requisiti:  
(un requisito funzionale concettuale e  
un requisito di sistema di dettaglio)!*

# Un requisito per una griglia su un editor grafico

---

**2.6 Funzionalità della griglia.** Per facilitare il posizionamento delle entità in un diagramma, l'utente può abilitare una griglia, in centimetri o pollici, attraverso una opzione del pannello di controllo. La griglia può essere abilitata e disabilitata durante le sessioni di lavoro, variando anche fra centimetri e pollici. Il requisito della griglia sarà presente anche quando si è scelta la visione ridotta dell'immagine (adattata alla finestra) ma, in tal caso, il numero di righe della griglia sarà ridotto.

# Problemi dei requisiti

---

- Il requisito della griglia combina tre tipi di requisiti:
  - Un requisito funzionale concettuale (il bisogno di una griglia);
  - Un requisito non-funzionale dettagliato (le unità di misura);
  - Un requisito non funzionale dell'Interfaccia Grafica definisce come la griglia può essere abilitata o disabilitata in vari casi.
- *I requisiti utente dovrebbero dare una descrizione generale del requisito, per essere compresi più facilmente e non confondere il lettore.*



# Esempio: Analisi dei requisiti per un Ristorante (1/4)

---

- Si vuole realizzare un sistema software che sia in grado di supportare la gestione di un ristorante. Esso deve consentire la gestione dell'anagrafica dei clienti, dando la possibilità di mantenere una scheda per ognuno di essi, in particolare per coloro che abbiano prenotato o organizzato cerimonie presso il ristorante.
- Dati anagrafici rilevanti sono considerati il Nome, il Cognome, la Data di Nascita, l'Indirizzo, la Città, il Codice Fiscale, i Numeri di telefono (eventuali) e gli indirizzi e-mail (eventuali).

# Esempio: Analisi dei requisiti per un Ristorante (2/4)

---

- Il sistema deve consentire l'inserimento, la visualizzazione, la modifica, la ricerca delle schede dei clienti. Inoltre, per ogni cliente, devono essere mantenuti dati statistici sulle operazioni da esso effettuate in passato, quali il numero di cerimonie effettuate, il numero medio annuo di cerimonie, l'incasso medio per cerimonia, l'incasso massimo, l'incasso minimo, il numero d'invitati medio, il numero di cerimonie disdette.
- Tali operazioni devono essere effettuate unicamente dal titolare, per cui sarà necessaria la realizzazione di un meccanismo per il controllo dell'identità dell'operatore, basato su di una login ed una password costanti.

# Esempio: Analisi dei requisiti per un Ristorante (3/4)

---

- Il sistema deve gestire le operazioni relative alla prenotazione del ristorante per cerimonie. La sala del ristorante può essere prenotata per una cerimonia solo da un cliente registrato, il quale deve indicare il tipo di cerimonia, la data e l'ora, il numero di invitati (distinguendo tra persone adulte e bambini), il numero di animatori, scegliere il menù (tra quelli prefissati oppure indicando un insieme di portate), definire tipologia e prezzo di eventuali servizi accessori e l'ammontare dell'eventuale acconto. La modifica dei dati della prenotazione è possibile solo per quello che riguarda il numero di invitati e il numero di animatori.
- E' possibile, inoltre, disdire la prenotazione della cerimonia. In tal caso, però, l'importo dell'eventuale acconto, deciso e pagato in sede di prenotazione della cerimonia, non viene restituito al cliente. Si noti che non è possibile prenotare cerimonie con meno di una settimana di preavviso e che in caso di prenotazione per un matrimonio, è necessario prenotare l'intera sala.

# Esempio: Analisi dei requisiti per un Ristorante (4/4)

---

- Il sistema deve poi gestire il pagamento ad avvenuta cerimonia lasciando libero il gestore di apportare arbitrariamente uno sconto sul prezzo inizialmente pattuito. Il sistema deve inoltre produrre una ricevuta fiscale o una fattura comprovante l'avvenuto pagamento.
- Il sistema deve presentare un'interfaccia utente basata su finestre e deve funzionare su calcolatori dotati di sistemi operativi Microsoft (Windows 98 o superiori). Inoltre si richiede che la memorizzazione dei dati sia effettuata su database Microsoft Access.

# Alcune Linee Guida

---

- Adottare un formato standard per i requisiti
- Usare il linguaggio in maniera coerente. Es. distinguere i requisiti **obbligatori** (indicati con “deve”) da quelli **desiderati** (indicati con “dovrebbe”)
- Usare diversi stili di testo per evidenziare i punti chiave del requisito
- Evitare il gergo informatico

# Esempio di formato per la descrizione

---

- *Nome requisito:*
- *Descrizione funzionalità:*
- *Motivazione:*
- *Specifica dettagliata:*
- *Sorgente:*

# Esempio di descrizione

---

- *Nome requisito:* **Funzionalità della griglia**
- *Descrizione funzionalità:*
  - L'editor deve fornire una funzione griglia dove una matrice di linee orizzontali e verticali fornisce uno sfondo alla finestra dell'editor.
- *Motivazione:*
  - La griglia aiuta l'utente a costruire diagrammi più precisi
- *Specifica:*
  - ECLIPSE/WS/Tools/FS Sezione 5.6
- *Sorgente:*
  - Mario Rossi, Ufficio di Roma

# Requisiti di Sistema

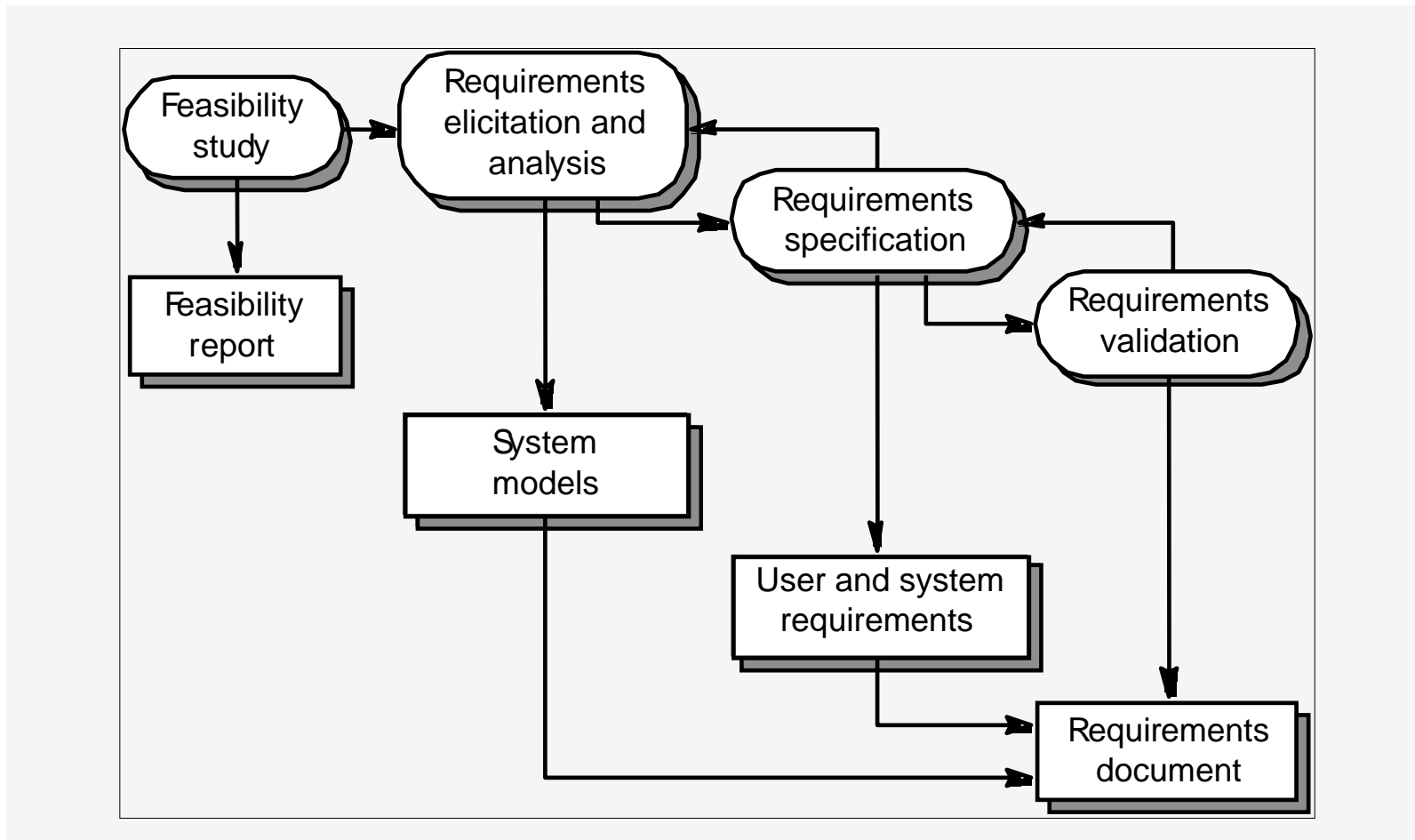
---

- Sono specifiche più dettagliate dei requisiti utente che serviranno come base per la progettazione, ma non sono le specifiche di come verrà sviluppato.
- Spesso sono inclusi nel contratto fra cliente e sviluppatore.
- In alcuni casi potrebbero però includere già riferimenti all'architettura che il sistema avrà
  - per poter strutturare meglio i requisiti relativamente ai sotto-sistemi,
  - per soddisfare specifici requisiti non-funzionali,
  - per esigenze di inter-operabilità.



# Un processo di analisi e specifica dei requisiti

---



# Linguaggi per la specifica

---

- Spesso i requisiti di sistema vengono espressi in linguaggio naturale (NL).
- Alcune alternative al NL:
  - Linguaggio naturale strutturato e stilizzato (con uso di Moduli/Template standard)
  - Modelli grafici (come gli use case e sequence diagrams)
  - Specifiche matematiche formali (con macchine a stati, notazioni insiemistiche, o altro)

# Specifiche in linguaggio strutturato

---

- La libertà di chi scrive i requisiti è limitata usando un 'modello' predefinito e standard per tutti i requisiti.
- Viene limitata anche la terminologia usata per descrivere il requisito.
- Il vantaggio è che si conserva l'espressività del linguaggio naturale, ma al tempo stesso si impone un grado di uniformità per descrivere le specifiche.

# Specifiche basate su un Modello

---

- Definizione della funzione o entità.
- Descrizione degli input e relative sorgenti.
- Descrizione degli output e loro destinazione.
- Altre entità richieste.
- *Pre-condizioni e Post-condizioni*
- Eventuali effetti collaterali della funzione.

# Esempio

*Insulin Pump/Control Software/SRS/3.3.2*

**Function** Compute insulin dose: Safe sugar level

**Description** Computes the dose of insulin to be delivered when the current measured sugar level is in the safe zone between 3 and 7 units.

**Inputs** Current sugar reading (r2), the previous two readings (r0 and r1)

**Source** Current sugar reading from sensor. Other readings from memory.

**Outputs** CompDose – the dose in insulin to be delivered

**Destination** Main control loop

**Action:** CompDose is zero if the sugar level is stable or falling or if the level is increasing but the rate of increase is decreasing. If the level is increasing and the rate of increase is increasing, then CompDose is computed by dividing the difference between the current sugar level and the previous level by 4 and rounding the result. If the result, is rounded to zero then CompDose is set to the minimum dose that can be delivered.

**Requires** Two previous readings so that the rate of change of sugar level can be computed.

**Pre-condition** The insulin reservoir contains at least the maximum allowed single dose of insulin..

**Post-condition** r0 is replaced by r1 then r1 is replaced by r2

**Side-effects** None

# Il Documento di Specifica dei Requisiti (SRS)

---

- È una dichiarazione ufficiale di ciò che gli sviluppatori dovrebbero implementare (attenzione: il COSA ma non il COME)
- Deve includere sia i requisiti utente (RU), sia una specifica dettagliata dei requisiti di sistema (RS).
- Talora RU e RS sono contenuti nello stesso documento, talora in documenti separati.

# Il Documento di Specifica dei Requisiti (SRS)

---

- Un SRS costituisce il punto di convergenza di tre diversi punti di vista: cliente, utente, sviluppatore
- Un SRS fornisce un punto di riferimento per la *validazione* del prodotto finale
- Un SRS di qualità è il pre-requisito per un software di alta qualità
  - un errore nell'SRS produrrà errori nel sistema finale
- Un SRS di qualità riduce i costi di sviluppo
  - correggere un errore dell'SRS dopo lo sviluppo costa 100 volte più che correggerlo durante la fase di Analisi

FASE	COSTO (in ore uomo)
Analisi dei Requisiti	2
Progettazione	5
Codifica	15
Test di accettazione	50
Esercizio e manutenzione	150

# Lo standard IEEE 830 -1998

---

- Definisce una generica struttura per un documento dei requisiti che deve essere istanziata per ogni specifico sistema.

Contiene più capitoli:

- Introduzione.
- Descrizione Generale.
- Requisiti Specifici.
- Appendici.
- Indice.



# Standard *IEEE per l'SRS*

---

Table of contents

## **1.Introduction**

- 1.1. Purpose (Scopo del documento)
- 1.2. Scope (Scopo del prodotto)
- 1.3. Definitions, Acronyms and Abbreviations
- 1.4. References
- 1.5. Overview (Descrizione generale del resto del documento)

## **2.General Description**

- 2.1. Product Perspective
- 2.2. Product Functions
- 2.3. User Characteristics
- 2.4. General Constraints
- 2.5. Assumptions and Dependencies

....

# Standard *IEEE per l'SRS*

---

## 3. Specific Requirements

### 3.1. Functional Requirements

#### 3.1.1. Functional Requirement 1

3.1.1.1. Introduction

3.1.1.2. Inputs

3.1.1.3. Processing

3.1.1.4. Output

#### 3.1.2. Functional Requirement 2

...

#### 3.1.3. Functional Requirement n

...

---

### **3.2. External Interface Requirements**

- 3.2.1. User Interfaces
- 3.2.2. Hardware Interfaces
- 3.2.3. Software Interfaces
- 3.2.4. Communications Interfaces

### **3.3. Performances Requirements**

### **3.4. Design Constraints**

- 3.4.1 Standard Compliance
- 3.4.2. Hardware Limitation

...

### **3.5. Software System Attributes**

- 3.5.1. Security
- 3.5.2. Maintainability

...

### **3.6. Other Requirements**

- 3.6.1. Logical Database Requirements
- 3.6.2. Operations
- 3.6.3. Site Adaptation requirements

...

# Esempio di Specifica di Funzionalità

---

Ciascuna Funzionalità deve essere specificata indicando gli input, l'output e l'azione elaborativa. Ad esempio:

## 3.2.1 Inserisci articolo

*Introduzione:* La funzionalità consente l'immissione dei dati relativi ad un nuovo articolo ed alla loro registrazione in un archivio

*Input:* codice articolo, descrizione, unità di misura

*Elaborazione:* visualizzazione di un form per l'immissione dei dati di input; lettura dei dati di input; verifica che l'unità di misura sia uno dei tre valori possibili e registrazione dei dati immessi nell'archivio Articoli

*Output:* dati di input registrati nell'archivio Articoli o messaggio di errore 'Unità di misura non ammessa'

# Altro esempio (1/3)

---

## **Elenco Fornitori**

- **Utenti della funzionalità:** Titolare
- **Introduzione**
  - La funzionalità visualizza la lista dei fornitori con i debiti verso questi e con possibilità di stampa. Consente, se richiesto, di selezionare un fornitore, di visualizzarne la scheda personale ed eventualmente di stamparla.

# Altro esempio (2/3)

---

- **Precondizione**
  - L'utente si è precedentemente autenticato come titolare del ristorante
- **Input**
  - Ragione sociale – Facoltativo
  - Nome titolare – Facoltativo
  - Cognome titolare – Facoltativo
  - Indirizzo – Facoltativo
  - Città – Facoltativo
  - P. I.V.A. – Facoltativo
  - Telefono – Un recapito telefonico. Facoltativo
  - E-Mail – Un solo indirizzo. Facoltativo.
  - Almeno uno tra le precedenti informazioni deve essere obbligatoriamente inserito.

# Altro esempio (3/3)

---

- **Elaborazione**

- Ristò presenta una lista con l'elenco dei fornitori
- Selezionando un fornitore dalla lista è possibile richiedere la visualizzazione o la stampa della sua scheda riportante tutti i dati anagrafici e contabili del fornitore.

- **Output**

- La lista con l'elenco dei fornitori, visualizzata a schermo o stampata, riporta per ciascuno di essi:
- Ragione Sociale - Nome Fornitore - Cognome Fornitore - Data d'inizio periodo Corrente - Saldo fornitore

# Le caratteristiche di un buon SRS ...

---

- Corretto/ Valido
  - se ogni requisito presente nell'SRS è realmente richiesto/necessario per gli utenti del sistema finale
- Completo
  - se ogni funzione richiesta al software ed il suo comportamento rispetto ad ogni possibile input è specificato
- Non Ambiguo
  - ogni requisito ha una sola interpretazione (formale vs. informale)
- Verificabile
  - se è possibile verificare che il sistema realizzi ogni requisito (richiede la non ambiguità dei requisiti)



# ... le caratteristiche di un buon SRS

---

- Consistente
  - se nessun requisito è in contraddizione con gli altri
- Modificabile
  - se la struttura e lo stile dell'SRS sono tali da consentire facili modifiche, preservando consistenza e completezza (un SRS con ridondanze non si modifica facilmente)
- Tracciabile
  - se l'origine di ciascun requisito è chiara e può essere referenziata nello sviluppo futuro (def. IEEE)
  - *forward traceability*: requisito collegabile a qualche elemento del progetto e del codice
  - *backward traceability*: dal progetto e dal codice è possibile risalire al requisito corrispondente

# La Validazione delle Specifiche

---

- L'obiettivo è quello di assicurare che l'SRS rifletta accuratamente e con chiarezza i requisiti effettivamente richiesti al software
- Tipi di errori riscontrabili in un SRS
  - *Omissione* (mancata presenza di un requisito)
  - *Inconsistenza* (contraddizione fra i vari requisiti o dei requisiti rispetto all'ambiente operativo)
  - *Incorrettezza* (fatti non corretti nell'SRS)
  - *Ambiguità* (requisiti con significati multipli)



How the customer explained it



How the Project Leader understood it



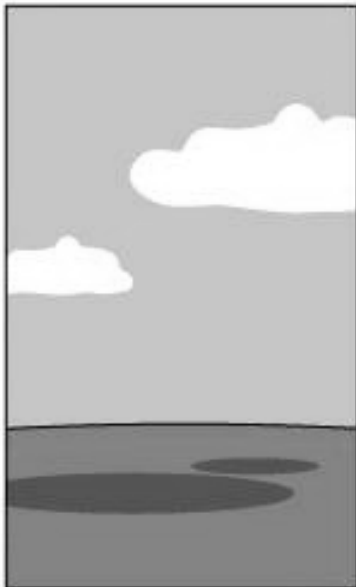
How the Analyst designed it



How the Programmer wrote it



How the Business Consultant described it



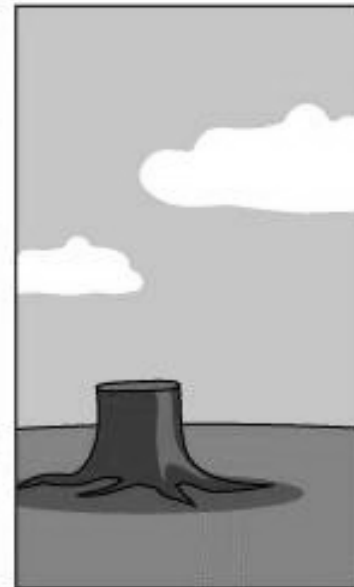
How the project was documented



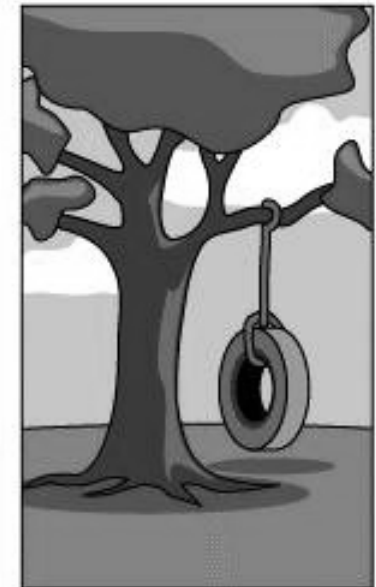
What operations installed



How the customer was billed



How it was supported



What the customer really needed