



**Corso Luigi Einaudi, 55 - Torino**

**Appunti universitari**

**Tesi di laurea**

**Cartoleria e cancelleria**

**Stampa file e fotocopie**

**Print on demand**

**Rilegature**

**NUMERO: 609**

**DATA: 0409/2013**

# **A P P U N T I**

**STUDENTE: Arlotta**

**MATERIA: Informatica Medica + Domande**

**Prof. Balestra**

Il presente lavoro nasce dall'impegno dell'autore ed è distribuito in accordo con il Centro Appunti.

Tutti i diritti sono riservati. È vietata qualsiasi riproduzione, copia totale o parziale, dei contenuti inseriti nel presente volume, ivi inclusa la memorizzazione, rielaborazione, diffusione o distribuzione dei contenuti stessi mediante qualunque supporto magnetico o cartaceo, piattaforma tecnologica o rete telematica, senza previa autorizzazione scritta dell'autore.

ATTENZIONE: QUESTI APPUNTI SONO FATTI DA STUDENTIE NON SONO STATI VISIONATI DAL DOCENTE.  
IL NOME DEL PROFESSORE, SERVE SOLO PER IDENTIFICARE IL CORSO.

L'informatica medica nasce negli anni '70 ma lo sviluppo è stato lento, è ancora in corso.

Il motivo per cui serve una figura professionale per l'informatica medica è la sicurezza del software per il paziente. È più virtuale e quindi più pericoloso.

Come per le biomagneti, l'evoluzione è partita con lo sviluppo dei microprocessori. Ha avuto un'evoluzione meno veloce delle altre aziende (banche, privati, ...)

Si è partiti dall'obiettivo di portare da cartaceo a computer, arrivati al "il computer decide la terapia" per puntare alla condivisione dei dati.

**eHEALTH**. Si fa riferimento alla possibilità di usare internet in attività ospedaliere per la trasmissione di dati.

Nel DL CROSCITA 2.0 si fa riferimento alla digitalizzazione in Italia del fascicolo sanitario che raccoglie la storia clinica e della ricetta (il medico la fa, il farmacista vede la ricetta e il farmacista vede la ricetta).

Lo scopo divide in:

- software legati alla strumentazione;
- software per l'organizzazione;
- software che permettono lo scambio di dati.

Secondo l'uso, si usano:

- per procedure organizzative (acquisti, pagamenti);
- per attività organizzative ma legate al paziente (prenotazione);
- direttamente per la cura del paziente (prescrizione della terapia). SII

## ANALISI dei FABBISOGNI

Il prodotto sviluppato deve supportare le attività del personale sanitario.

Sviluppare male le specifiche ed dire progettare male, nel senso che non risponde alle richieste.

Se voglio che il software serva davvero, deve integrarsi con il sistema. Devo capire in primis l'organizzazione e i processi che vi si portano avanti.

Elicitation = capire cosa dobbiamo informatizzare

Parto dal background, sapendo cosa succede con la telemedicina, cosa succede in un reparto ospedaliero. Mi può servire intervistare alcune persone. Devo avere gli strumenti per analizzare i processi. L'analisi dei processi mi serve a capire se funzionano.

I cambiamenti organizzativi devono **ADATTARSI** la situazione di cui vanno ad applicarsi.

Quando sappiamo rispetto a che processi andiamo a lavorare, parte la prima vera fase di progettazione del software.

Usiamo come esempio l'**INFORMATIZZAZIONE DELLA GESTIONE DELLA TERAPIA**. Siamo in un reparto di medicina

**BACKGROUND**: ci sono un certo numero di letti di degenza, qui medico ha un certo numero di letti. Non tutti i medici sono sempre presenti in reparto: tutti alternano, almeno uno al pomeriggio, uno reperibile per l'urgenza. Almeno uno resta al giorno cioè qualcuno che fa il punto della situazione.

La terapia viene stabilita subito e poi solo leggermente

Non tutti i modelli vanno bene per cosa dobbiamo fare.  
 Un sistema è un insieme di elementi collegati da relazioni.  
 Due capire cosa deve stare nel modello e cosa no, si parla  
 di modelli corretti e non corretti.

Gli strumenti sono schemi e diagrammi, quindi di tipo  
 grafico.

Un processo è un insieme di attività, ha un inizio e una fine.

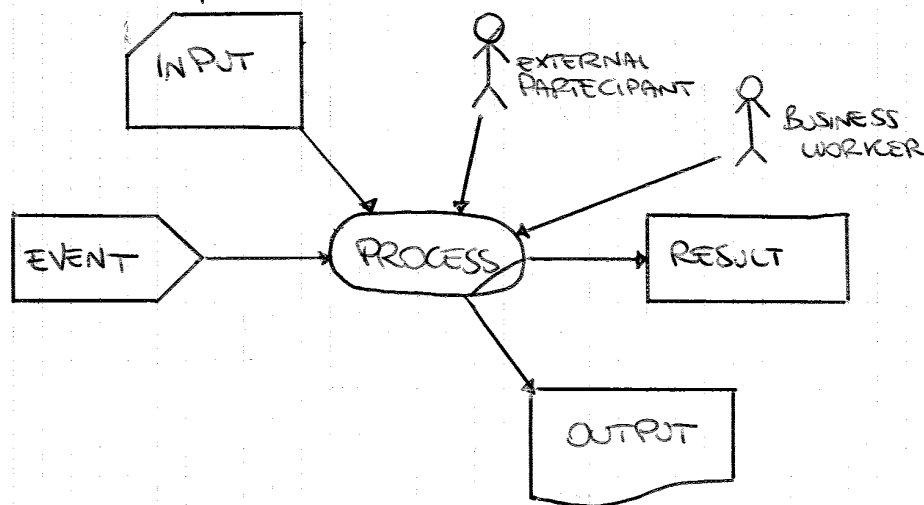
In informatica non l'azione in parallelo.

Il sistema comincia, passa per diversi stati e arriva a quello  
 finale; è importante definire lo stato finale. Oltre agli stati  
 sono importanti i dati e i partecipanti al processo (attivi o pas-  
 sivi).

Il nostro modello deve essere l'aiuto nella automazione (infor-  
 matizzazione) di un processo.

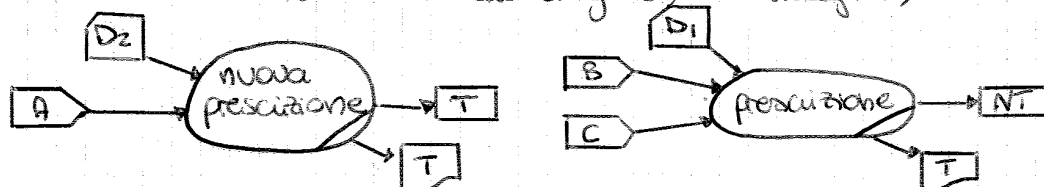
### SYNOPSIS DIAGRAM

È una visione compatta del processo, individua i partecipanti  
 al processo e li divide in interni ed esterni (interno = medico,  
 esterno = paziente)



Decido subito se tenere la prescrizione iniziale fatta sul paziente.  
 Gli eventi che fanno parte il processo:

- A → nuovo;
- B → modifica dello stato clinico;
- C → risultato esame diagnostico/radiologico;



Il sistema non è tutto il reparto. Gli elementi importanti sono persone, documenti, strumentazione, anche qualcosa di non fisico (come le conoscenze e le esperienze del medico).

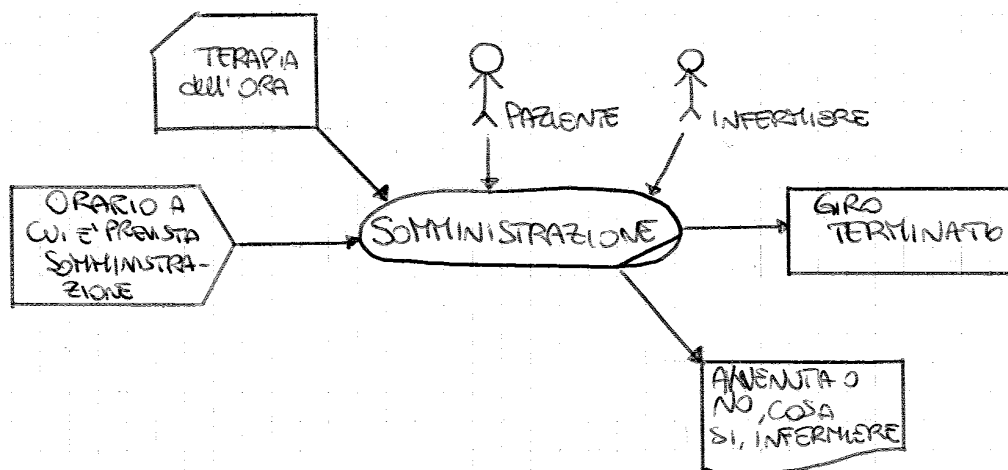
Ci servono:

- medici
- paziente
- terapia programmata
- dati clinici del paziente

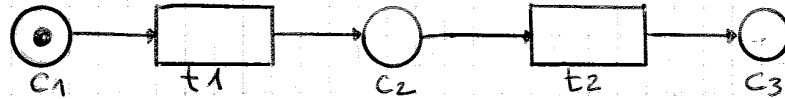
Gli input sono dati che ho ma che il paziente abbia o no una terapia:

- D1 → il paziente ha una terapia;
- D2 → il paziente non ha una terapia quindi non allegre, stato clinico.

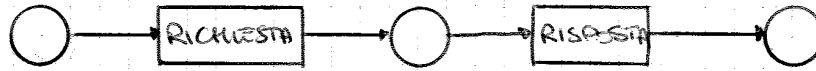
Nel caso della somministrazione



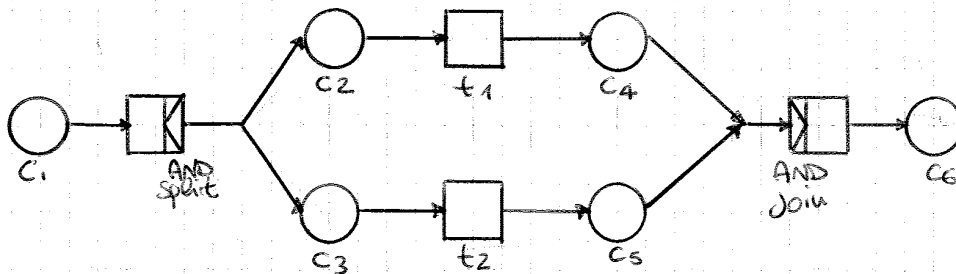
Il processo più semplice è una sequenza "lineare"



Le azioni vengono eseguite una dopo l'altra.



Si parte con un place e si termina con un place. Posso portare avanti delle azioni in parallelo (ad esempio se non ho un ordine rigido)

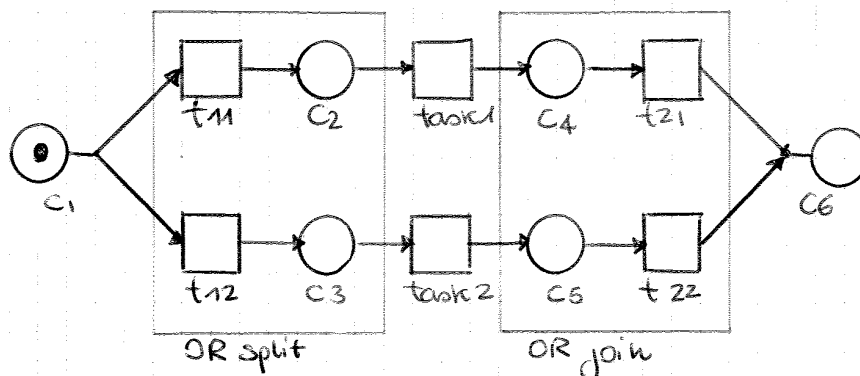


Quando avviene un ricambio può avvenire il parallelo e gli si assegna un posto. "Arrivo" e "Assegnazione letto" possono avvenire in parallelo.

AND



Mentre il parallelo è relativo al fatto che ci sono più attività, portare avanti (attività o sequenze di attività), la selezione indica che a seconda dei dati in quel momento sceglie una strada piuttosto che un'altra.



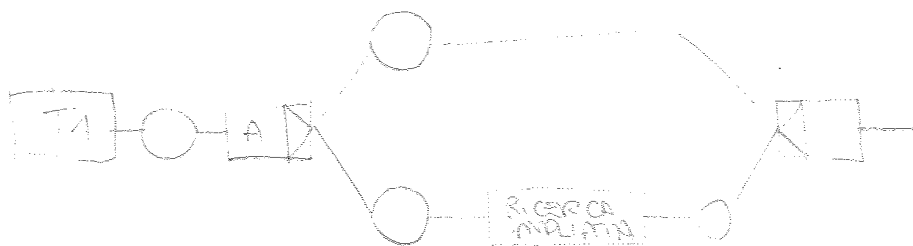
Immagino che il sistema parta dallo stato in cui il medico si appresta a modificare la terapia del paziente.



Supponiamo si usi una scheda unica di terapia. La prima T1 può essere RICERCA SCHEDA UNICA di TERAPIA del PAZIENTE. La ricerca può dare esito positivo, negativo (se il paziente non c'è) ma il medico dovrebbe saperlo a priori. Supponiamo che al ricambio si usi una cartella: la scheda c'è. Non considero che ma pensa. La T2 sono: COMPLICAZIONE SCHEDA UNICA di TERAPIA.

Posso considerare le T come macroprocessi ed aprirle in sotto processi.

La RICERCA diventa una vera ricerca tra le schede, una selezione, può dare esito negativo.



A = scheda ~~presente~~ presente



La somministrazione avviene supponendo che il cancello dei farmaci  
sia stato recuperato in stadi precedenti. Inizierò con gli  
infermi che prendono il cancello e mi avviaio alla  
prima stanza. Posso immaginare il giro stanza dopo stanza.  
L'importante è il paziente singolo. Posso immaginare un  
processo che mi ripete e consiste in

- leggere i farmaci da somministrare;
- se ce ne sono → si prendono i farmaci;
  - mi somministro;
- il paziente può non essere;
- si fa la scheda;
- passa al successivo.

Da considerare il RISCHIO. È collegato alla tecnologia e ce ne sono di 3 tipologie:

→ INTRINSECO deriva da una cattiva progettazione ma ci sono le norme che, se seguite, evitano la cattiva progettazione in esito reale; ci sono poi i cattivi funzionamenti che non sono così evidenti (un sensore che funziona male). Con le nuove tecnologie, si sviluppa parecchi autotest che indicano il rischio di malfunzionamento;

→ GESTIONE;

→ USO CORRETTO.

Per quanto riguarda i software, possono essere

- gestionali
- medicali
- usati per un'apparecchiatura

I software che sono dispositivi medici sono strettamente legati usati in un'apparecchiatura, si discute sui medicali. Secondo alcuni sono gestionali perché consentono solo di memorizzare dei dati; ma poi verranno anche letti e usati per la terapia c'è un collegamento diretto con il paziente.

Una corretta gestione delle tecnologie sanitarie riduce i rischi ed è un compito dell'ingegneria clinica.

L'ingegneria clinica nasce negli anni '80 quando la strumentazione non era intrinsecamente sicura e si occupava quindi della sicurezza e delle prove. L'evoluzione della tecnologia ha fatto sì che ci si spostasse verso un concetto di gestione più allargato verso i processi e non solo la strumentazione.

Nel software, possono esserci manutenzioni e aggiornamenti (proposti con cadenza dalla ditta produttrice).

L'ultima attività è l'acquisizione, dove l'ingegnere clinico ha un grosso ruolo:

- costruire un capitale;
- valutare l'offerta;
- effettuare un collaudo prima di renderlo disponibile.

### RAFFORTO CE - IT

L'ingegneria clinica interagisce con due servizi nella struttura sanitaria; dal punto di vista del software, il servizio è quello della gestione dell'informatica.

Questo servizio gestisce le infrastrutture:

- pc
- rete dati
- 

Di fronte ad un malfunzionamento si può capire se la colpa è del software o della rete infrastrutturale.

Ci può essere una scansione di responsabilità tra CE e IT ma l'importante è arrivare ad una soluzione. In USA c'è stato un caso ed è stata fatta una richiesta di collaborazione, malvista da entrambi i lati. È stata poi accettata, migliorando il servizio agli operatori sanitari.

Un problema da risolvere in collaborazione è la GESTIONE DELLA RETE DATI. Collegate alle reti ci sono dispositivi medici e non il produttore garantisce il dispositivo medico senza pensare a questa connessione. Non va pregiudicato il DIT nel collegarlo alla rete. Nel 2010 è stato proposto uno standard per la sicurezza della rete. Per prima cosa, si individua un responsabile e delle figure professionali che devono garantire la gestione di questo rischio collaborazione CE - IT.

## ANALISI delle SPECIFICHE

C'è una fase di progetto, seguita da una costruzione.  
Il tutto completato da due fasi di testing:

- 1° il prodotto risponde alle specifiche
- 2° il prodotto è adeguato rispetto alle richieste dell'utente.

Il secondo testing ha esito tanto più positivo tanto più sono corrette le specifiche.

## OBJECT-ORIENTED PROGRAMMING

Il sistema operativo è l'insieme di programmi che fa funzionare il computer. Esiste una parte di software (il firmware).

Un aspetto critico è quello degli antivirus.

I primi software erano basati su 0 e 1. Con l'andare in meglio della tecnologia si è passati ad usare 3 elementi:

- sequenza;
- selezione;
- ciclo.

I dati arrivano per:  
◦ input utente  
◦ lettura parte del pc.

Nella programmazione procedurale per gestire la complessità è nato il concetto di SUBROUTINE:

- 1) c'è un programma principale;
- 2) ci sono delle subroutine richiamabili in caso d'uso.

Rimaneva il problema dei dati: il programma ha una istruzione di variabili che vengono usate la due fine.

Se modifichi una variabile, devi fare attenzione a che tutto funzioni ancora e può portare a degli errori.

La nascita dell'ool porta a un'analisi per la progettazione del software. C'è uno standard: UML (Unified Modeling Language). All'interno ci sono strumenti vari ed essendo un linguaggio è chi lo usa a decidere cosa utilizzare.

Puo' essere usato per mettere in rilievo gli aspetti più importanti, come "fotocopia" del sistema, oppure per aiutare a costruire (come linguaggio di programmazione).

Ce ne sono due versioni:

- 1) lo sviluppo del software avviene attraverso
  - analisi
  - costruzione
  - testing

è rappresentato il risultato come una sequenza di modelli.

I modelli-risultato sono gli input delle fasi successive.

La fase di analisi è pensata per essere collegata all'implementazione tra e propria;

- 2) è un'evoluzione abbastanza imprevista.

Si mette in evidenza che l'importante sono i casi d'uso e ci sono viste diverse dove ciascuna riprende il discorso precedente.

Si vede tutto non in sequenza ma incentrato sul caso d'uso.

L'approccio dell'UML 2 è mettere al centro il caso d'uso e poi le viste:

- la logica;
- process view, come viene usato il processo;
- development;
- physical.

→ estensione: caso che mi permette di dettagliare un particolare percorso la discussione



Problema ambiguo se c'è estensione o inclusione.

Bisogna descrivere i casi d'uso e si usa la USE CASE DETAILS

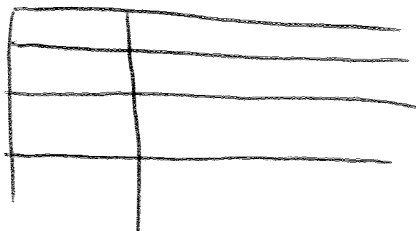
USE NAME	devo dare un nome che dia il senso generale
GOAL	cosa mi permette di fare posto caso d'uso
ACTORS	chi usa il caso d'uso
TRIGGER	cosa fa partire il caso d'uso (anche un altro caso) non è detto che esista
MAINFLOW	come il software arriva al goal scuola fa l'utente (rispetto al software) e cosa fa il software
ALTERNATIVE FLOW	possono esserci percorsi alternativi, ovvero quelli che vengono eseguiti meno. Di solito ce n'è almeno uno (per errore).
FORM	come il software si relaziona al caso d'uso (ma in genere non si trova). ↓ in genere mette l'immagine dell'interfaccia!

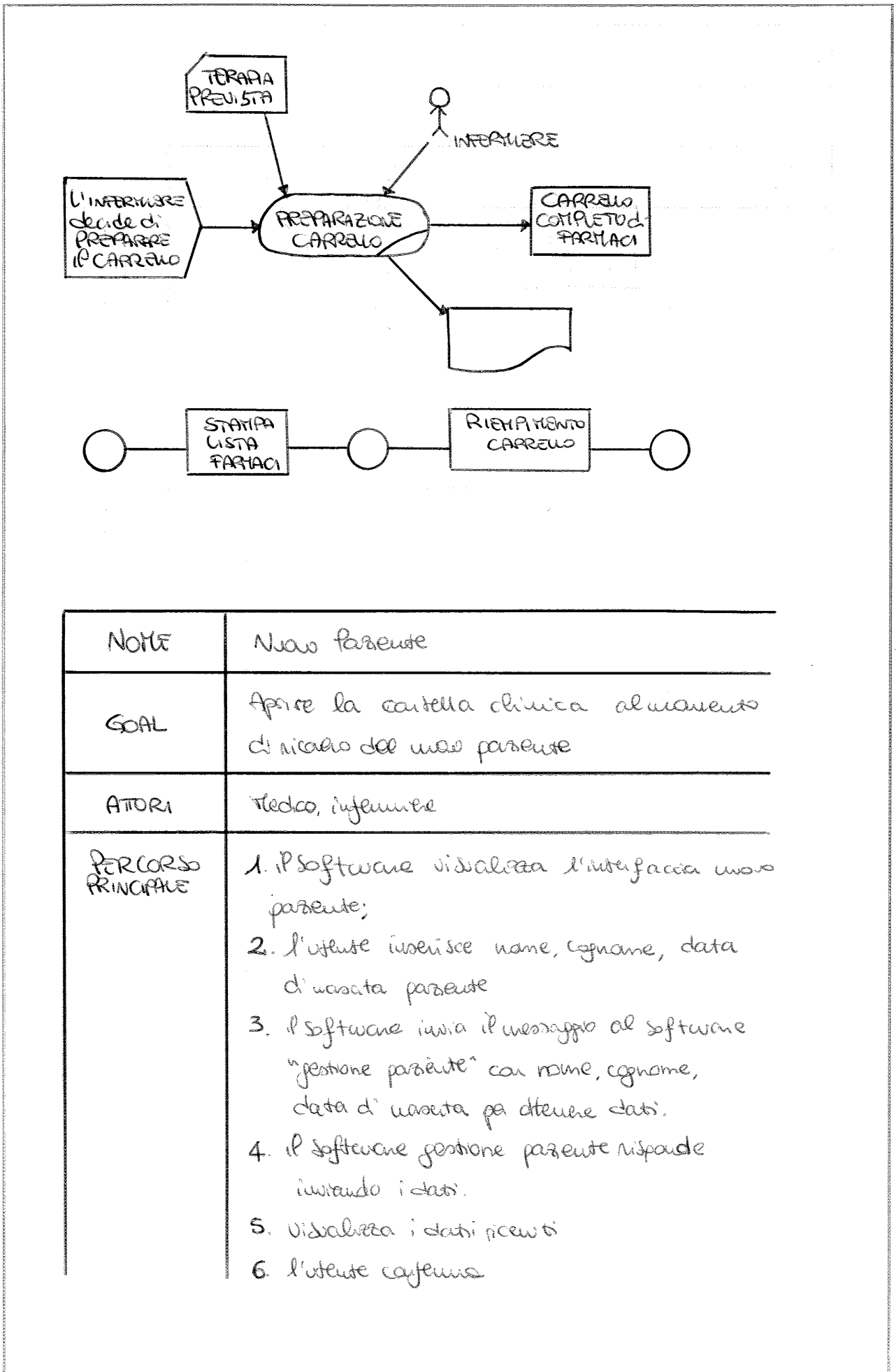
Si potrebbe aggiungere una riga per i dati e le precondizioni.

Ogni



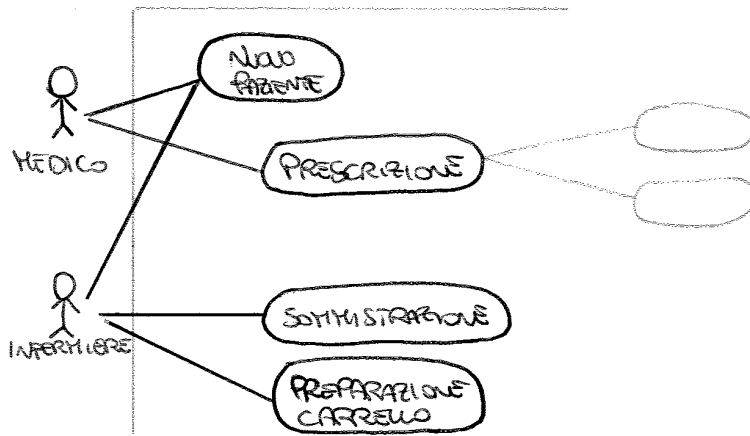
è uno use case details





NOTE	Nono paziente
GOAL	Aprire la cartella clinica al momento di nascita del mio paziente
ATTORI	Medico, infermiere
PERCORSO PRINCIPALE	<ol style="list-style-type: none"> <li>1. il software visualizza l'interfaccia uomo paziente;</li> <li>2. l'utente inserisce nome, cognome, data di nascita paziente</li> <li>3. il software invia il messaggio al software "gestione paziente" con nome, cognome, data di nascita per ottenere dati.</li> <li>4. il software gestione paziente risponde inviando i dati.</li> <li>5. visualizza i dati ricevuti</li> <li>6. l'utente conferma</li> </ol>

Nel caso delle PRESCRIZIONE, entro un caso d'uso per l'inserimento dei dati del paziente



Lo use case PRESCRIZIONE ne include alcuni e vi accede solo il medico.

Nella SOMMINISTRAZIONE vedo solo i farmaci da somministrazione al momento giusto e non vede la stessa interfaccia della prescrizione.

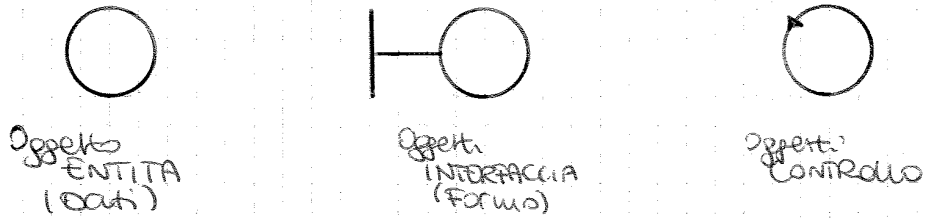
Ci sono un oggetto TERAPIA di cui prendono attributi diversi nel caso della PRESCRIZIONE o della SOMMINISTRAZIONE.

La prescrizione si fa dal medico e queste procedure hanno poi anche un risvolto in cause legali. Uno dei processi inclusi può essere la terapia al bisogno.

	I <sub>1</sub>
	<ol style="list-style-type: none"> <li>1. il software visualizza I<sub>1</sub></li> <li>2. l'utente clicca sulle funzionalità</li> <li>3. il software riconosce                     <ol style="list-style-type: none"> <li>1) _____</li> <li>2) _____</li> <li>3) _____</li> </ol> </li> </ol>



## OBJECTS (UML v.1)



L'interfaccia è lo strumento con cui l'utente comunica con il sistema. Non deve essere troppo densa, deve avere un font leggibile, deve avere un adeguato uso dei colori e le codifiche colori:

verde ✓

rosso ✗

giallo ?

Si usa un design simile al cartaceo usato precedentemente. È bene evitare colori violenti poiché il software si usa a lungo.

Non vanno che scendo visivamente, i dati devono essere messi in maniera accesa e non casuale.

Vanno inseriti controlli per validare i controlli. Se inserisco un nato nel 1700, posso un dato non valido. Il controllo di validità si spingerebbe alla data di nascita nel 1900 o mi impedirebbe di scrivere 31-02. Ma se il paziente è nato il 20-04 e io scrivo 21-04 non ho controllo a meno che non vi sia controllo con il codice fiscale.

L'errore di cifra è facile:

invece di 37,8° posso 38,7° come temperatura corporea.

Restando troppe richieste di conferma, l'utente tende a confermare senza pensarci troppo. Vanno limitate a situazioni con effettiva necessità.

È un sistema gerarchico.

Nei CHAPTER ci sono le principali patologie, sintomi e un ultimo per specifiche applicazioni.

A ciascuna voce che ci interessa, si associa un codice alfanumerico:

L ##  
m n  
lettera da 0 a 99

Nel capitolo 9 (patologie del sistema circolatorio): hanno un codice da I00 a I99. C'è un'ulteriore suddivisione dei codici.

Ad esempio:

I13 ipertensione e problemi di ipotensione

→ ho la possibilità di un'ulteriore specificazione

I13.0, I13.1, ...

Va bene da usare nella SDO per i numeri: ma è meno utile se invece voglio descrivere la situazione del singolo paziente.

Vengono per lo più usati per studi epidemiologici.

Porre un sottogruppo ad una classe rigida non permette flessibilità di conoscenza.

Dal 1/1/09, in Italia, si è assunto come codice la versione 9.M dell'ICD (in pubblicazione parte SDO)

La SDO si basa su un concetto a il cui che riguardano aspetti diversi, non solo patologie.

Anche per c'è un codice legato al concetto, un identificatore univoco. Ai concetti sono associati delle descrizioni.

L'evoluzione è l'UMLS: è un sistema che include i precedenti.

È pensato per un uso in sistemi computerizzati ed è legato all'interpretazione del linguaggio naturale

## BASI di DATI

Abbiamo bisogno di salvare i dati ma sono di molte forme diverse:

→

→

→

→

Ho due strutture:

→ un INSIEME di FILE

→ BASE di DATI (più usato).

Il file un consente a conoscere la struttura e quindi usarlo in contesti diversi.

Le basi usano per staccare la memorizzazione dei dati e la lettura da parte delle applicazioni.

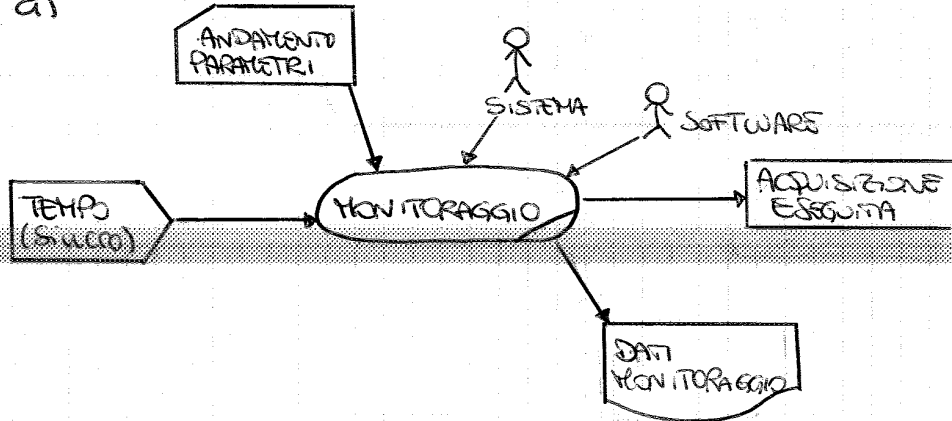
C'è un software dedicato: nel momento in cui devo modificare un certo file, lo modifico ed è dato che lo faccio tramite software, non ho più bisogno di modificare le singole applicazioni.

Ricevo la richiesta di un certo insieme di dati e lo prendo fisicamente dai file che lo contengono. Alcuni creano viste logiche a seconda dell'applicazione.

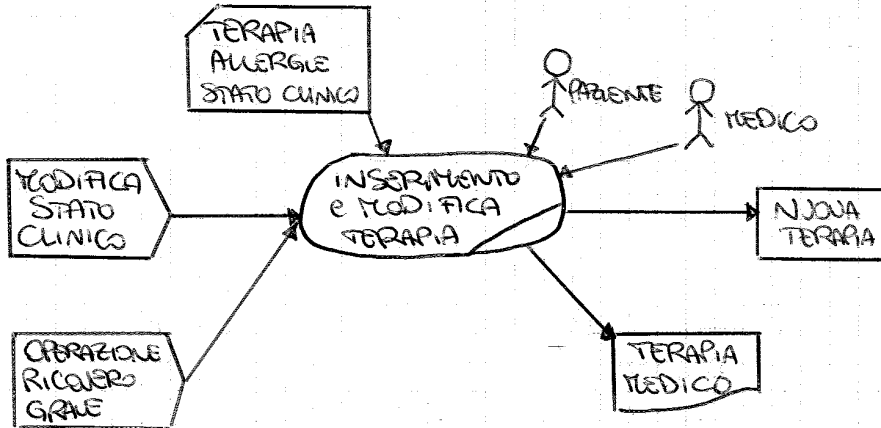
I file sono organizzati secondo una struttura a record, ovvero un insieme di activity.

### 1°) MONITORAGGIO

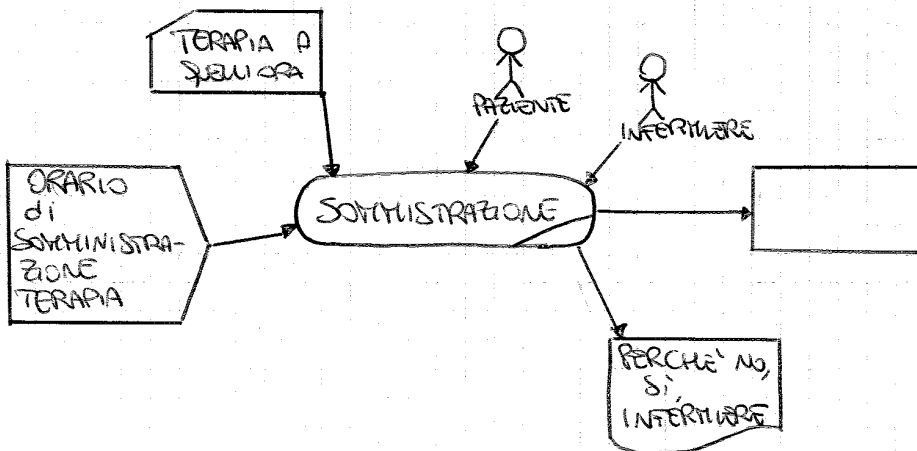
a)



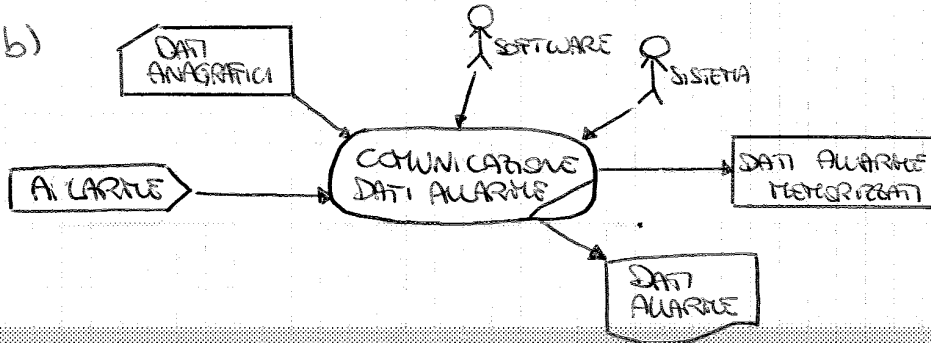
### 2°) INSERIMENTO e MODIFICA TERAPIA



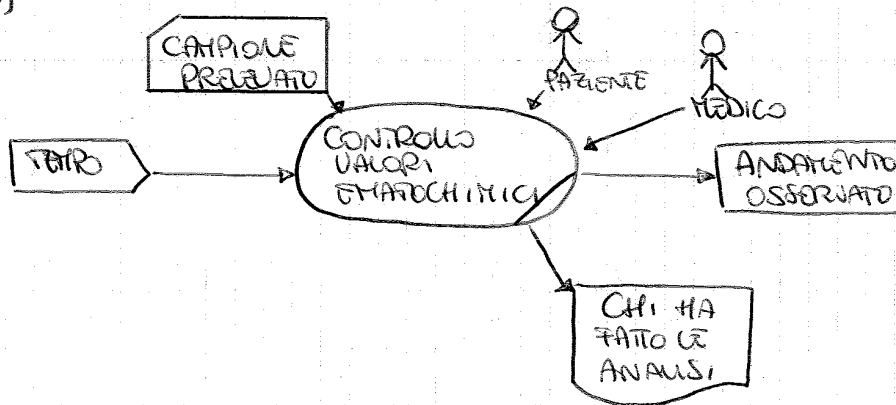
### 3°) SOMMINISTRAZIONE TERAPIA



1) b)



c)

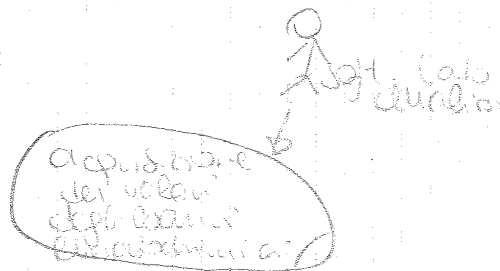


Si lavora su un signal paziente, quello per cui scatta l'allarme. Lo start è tipicamente ritardato dal momento in cui scatta l'allarme.

Come dati peggiori: molto pochi utili allo svolgimento del processo.

I dati significativi li usata non il tipo di evento d'allarme e il grado complessivo di momento d'allarme.

①



Ci sono due versioni:

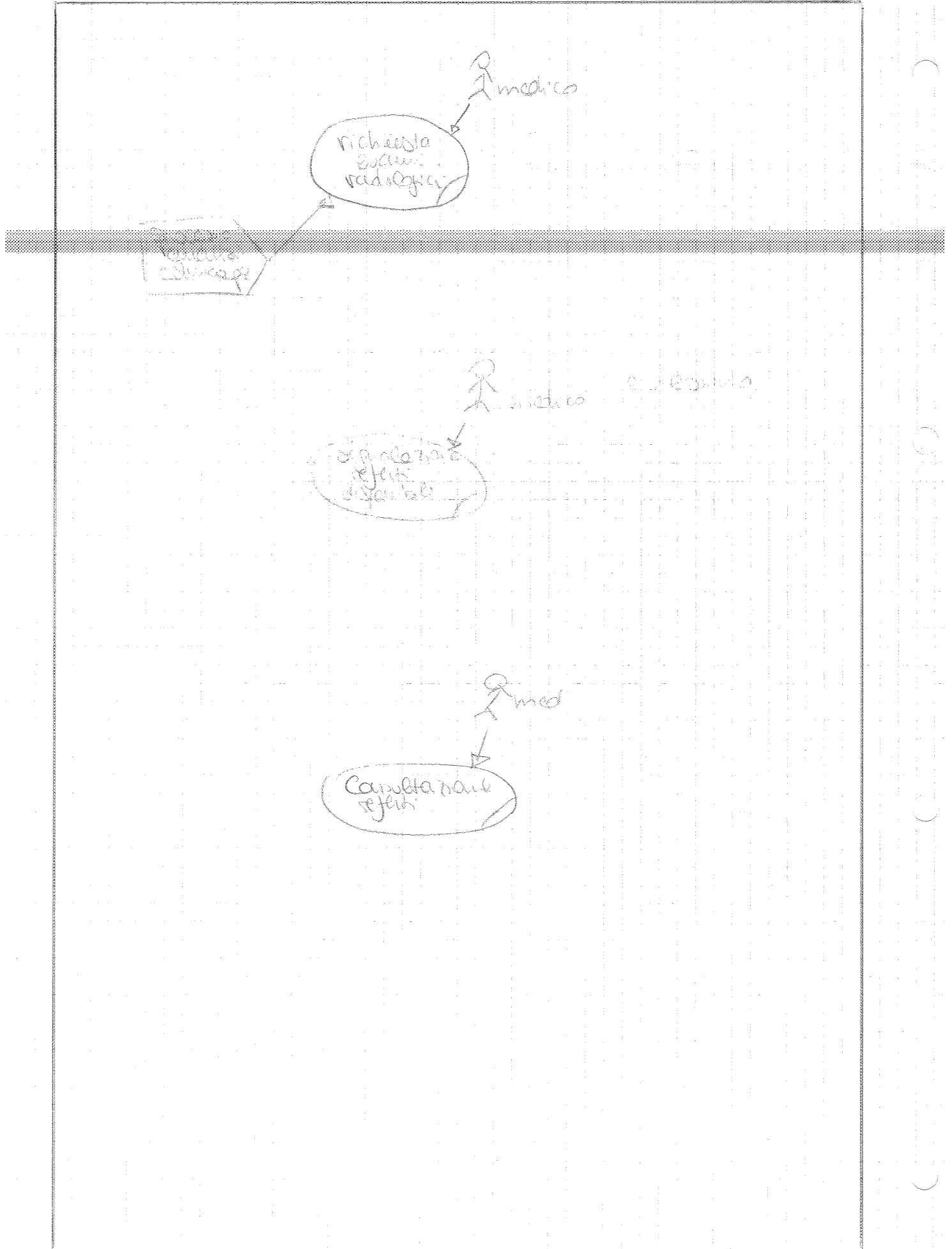
- 1) prelievo effettuato su tutti i pazienti;
- 2) sintire utile un prelievo una tantum su un paziente.

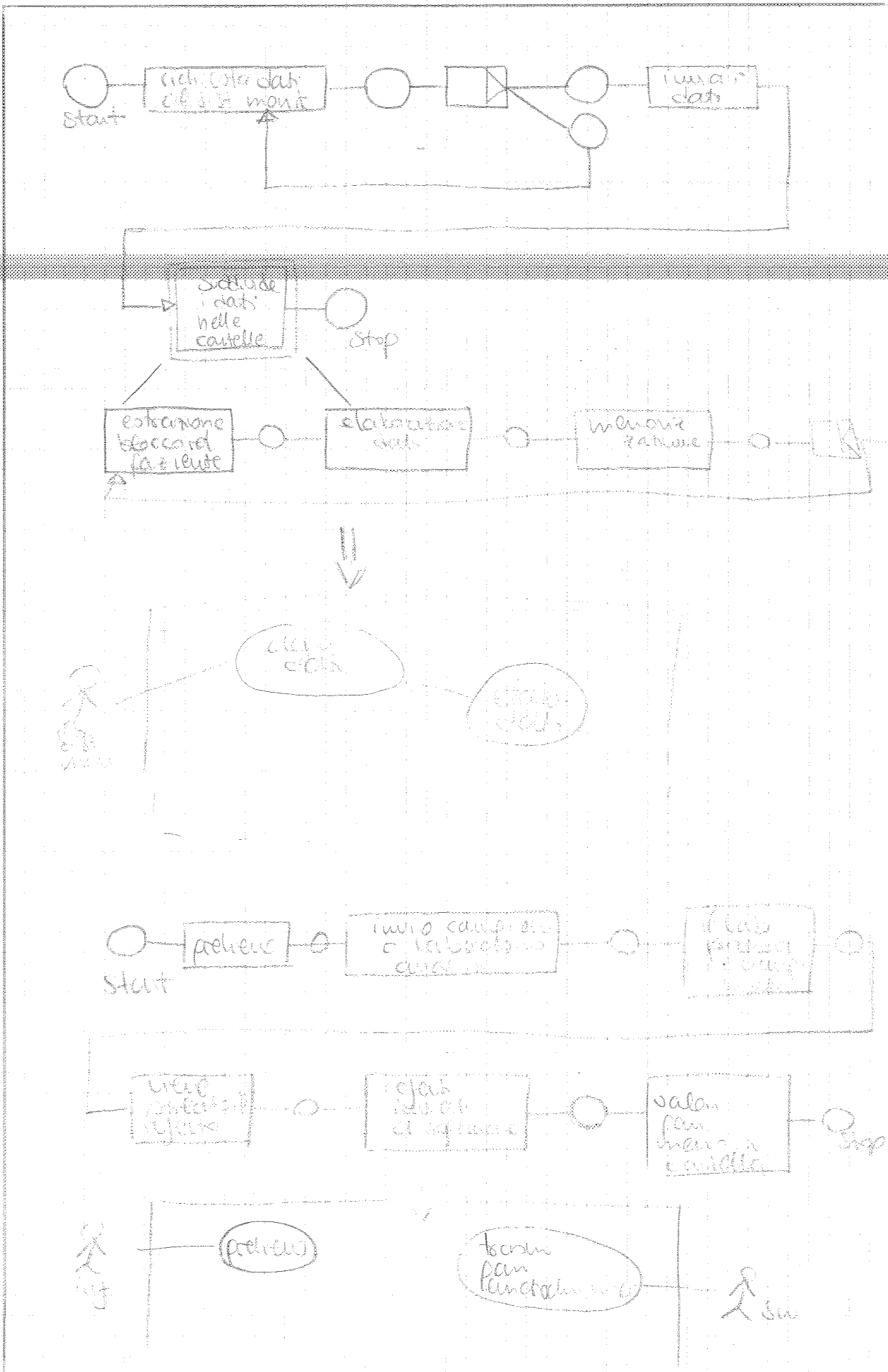
Dipendono da che momento faccio partire il processo. Se ho l'input anche dell'effettuazione di un prelievo, controllo se mi mancano dei reati. Se ne ho uno, devo controllare in un un - errore del software.

Devo segnalare se ho effettuato i prelievi o no.

Nella prima versione:

- ⊕ posso controllare quanti me ne trassero;
- ⊖ devo segnalare punt prelievi fatti.







Lo scopo è partire dai processi, arrivare ai requisiti e validarli per scrivere il software

Attraverso l'uso integrato di diagrammi si arriva ai requisiti

Ogni synopsis ha un workflow, a cui corrisponde uno swimlane.

Se ci sono attori è importante avere uno swimlane.

I requisiti vengono fuori dall'analisi del workflow

L'OBIETTIVO È VALIDARE I REQUISITI!

Come guida si possono usare due domande:

→ Quando e per fare cosa il Software interagirà con utenti "umani"?

→ Quando e per fare cosa il Software interagirà con utenti software?

Vanno in primo luogo identificate le transizioni che rispondono a questa domanda.

I tipi di diagrammi più utilizzati sono:

- 1) SYNOPSIS D. descrizione di massima di un processo
- 2) FLOWCHARTS D. sequenze di attività
- 3) WORKFLOWS D. fluc, possibili casi di un processo
- 4) SWIM LANE ACTIVITY D. sequenza di attività con chiara definizione del ruolo

Vengono sviluppati per:

- o documentare un processo;
- o valutare miglioramenti e carenze nel processo;
- o aiutare nell'automazione;
- o soddisfare le esigenze in conformità alla normativa.

5) UML: COS'È E A COSA SERVE?

UML Unified Modeling Language è un linguaggio standard di modellazione per lo sviluppo di software e sistemi.

La modellazione è importante perché serve a gestire la complessità del sistema in sviluppo, a monitorarne le evoluzioni e a documentarne le attività.

L'UML incorpora pratiche del design della modellazione del linguaggio, della programmazione per oggetti orientata alla scrittura della descrizione del linguaggio.

Ne sono due versioni (UML v 1.x, v 2.x)

6) UML COSA SONO I VARI STRUMENTI DI DESCRIZIONE DEI PROGETTI? SU CHI SONO?

Vengono fatte modellazioni tramite diagrammi quali:

- 1) USE CASE DIAGRAM individua chi si occupa di cosa;
- 2) USE CASE DETAILS spiega ogni "cosa" e nel dettaglio;
- 3) OBJECTS individua categorie e dati da memorizzare;
- 4) ACTIVITY DIAGRAM come il sistema risolve il suo scopo;

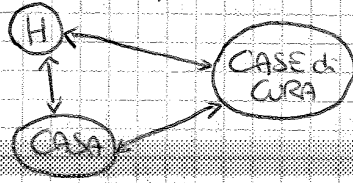
7) COS'È IL EHR?

EHR sta per electronic health record ed è un concetto ancora in evoluzione. Si tratta di una raccolta di dati che integra i dati del paziente (generati in cura) e permette, sotto automazione, l'accesso a chi di dovere come medici, infermieri...

Ha notevoli vantaggi come favorire la ricerca a livello biomedico e permettere la definizione di un protocollo.

## o eHEALTH

Si riferisce alla pratica della cura del paziente attraverso il supporto di strumenti informatici e tecniche di comunicazione medico-paziente. Comprende concetti come la cartella clinica elettronica e la telemedicina, l'interazione tra cittadini/pazienti e servizi medici.



Le applicazioni si dividono per:

- scopo
  - scu legati alla documentazione
  - scu legati all'organizzazione
  - scu legati allo scambio di dati
- uso
  - organizzazione di procedure
  - attività legate al paziente
  - legate alla cura

## o RISCHIO

- 1) intrinseco: legati ad una scorretta progettazione o malfunzionamento
- 2) gestionale: da mancata o non corretta gestione
- 3) uso: derivanti da un uso non corretto

## o SERVIZIO DI INGEGNERIA CLINICA

Interfaccia la direzione e l'operatore.

Fanno piani (mensili) che seguono molti: anzitutto compatibile mente con il budget. Acquistare un software significa non solo decidere il acquisto una programmazione le attività necessarie per l'uso.

Ci sono attività pianificate a breve termine.

Esempio un istituto tecnologico. Si occupa della acquisizione, della costruzione di un capitolato, valuta l'offerta, effettua il collaudo.

3

## Definizione di Informatica medica

Since the end of the 70s, researchers used to call the application of informatics to medicine *medical informatics*.

**Morris F. Collen, 1977 - *Medical informatics is the application of computer technology to all fields of medicine - medical care, medical teaching, and medical research.***

Preliminary announcement for the **Third World Conference on Medical Informatics**, MEDINFO , 1977.

The first examples of computerized management of patients data were described in the 80s.

## Definizione Informatica medica

### The evolution of Medical Informatics

From .... computerized management of patients data .....

**M.S. Blois and Edward H. Shortliffe, 1990 - "Medical informatics is the rapidly developing scientific field that deals with the storage, retrieval, and optimal use of biomedical information, data, and knowledge for problem solving and decision making."**

**British Medical Informatics Society - "...the understanding, skills, and tools that enable the sharing and use of information to deliver healthcare and promote health"**

to .... eHealth, mHealth ....

In the last 20 years, industry and academia have invested into applying ICT (Information and Communication Technologies: electronics , informatics and telecommunication) for health.

4

## DIGITAL AGENDA FOR EUROPE

A Europe 2020 Initiative

Agenda for Europe > Our Goals > Pillar VII: ICT-enabled benefits for EU society > Taxonomy Table

Life Business & Science and Telecoms & Creativity  
Funding Technology the Internet media

**Action 75: Give Europeans secure online access to their medical health data and achieve widespread telemedicine deployment**

**Undertake pilot actions to equip Europeans with secure online access to their medical health data by 2015 and to achieve by 2020 widespread deployment of telemedicine services**

**Action 76: Propose a recommendation to define a minimum common set of patient data**

**The Commission will work with Member States competent authorities and all interested stakeholders to: Propose a recommendation defining a minimum common set of patient data for interoperability of patient records to be accessed or exchanged electronically across Member States by 2012; Implementation of this action is also expected to contribute to Action 77.**

**Action 77: Foster EU-wide standards, interoperability testing and certification of eHealth**

**Foster EU-wide standards, interoperability testing and certification of eHealth systems by 2015 through stakeholder dialogue.**

7

## L'Italia e l'agenda digitale

[http://www.governo.it/Presidente/Comunicati/testo\\_int.asp?d=69362](http://www.governo.it/Presidente/Comunicati/testo_int.asp?d=69362) (8 febbraio 2013)

4 Ottobre 2012 - Dl Crescita 2.0

1.4 Misure per la Sanità digitale

• Fascicolo sanitario elettronico, cartella e prescrizione medica digitali (artt. 12,13).

Al via il fascicolo sanitario elettronico (FSE), che conterrà tutti i dati digitali di tipo sanitario e sociosanitario del cittadino, raccogliendone di fatto l'intera storia clinica. Il fascicolo verrà aggiornato da diversi soggetti che, nell'ambito del servizio sanitario pubblico, prendono in cura gli assistiti.

In questo senso, le strutture sanitarie pubbliche e quelle private accreditate potranno conservare le cartelle cliniche solo in forma digitale, realizzando così significativi risparmi e semplificazioni. Viene accelerato anche il processo di digitalizzazione delle prescrizioni mediche, definendo tempi certi e uguali su tutto il territorio nazionale.

8

## eHealth applications

According to their scope, applications can be divided into three main groups:

1. Software linked to a medical device
2. Software that manage data inside the organization
3. Software that allow data exchange among different organizations.

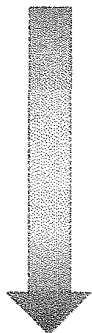
According to their use, applications can be divided into three main groups:

1. Software used for the management procedures
2. Software used for the management procedures that have an indirect link with patient care
3. **Software used directly for patient care (medical device software)**

## Applicazioni

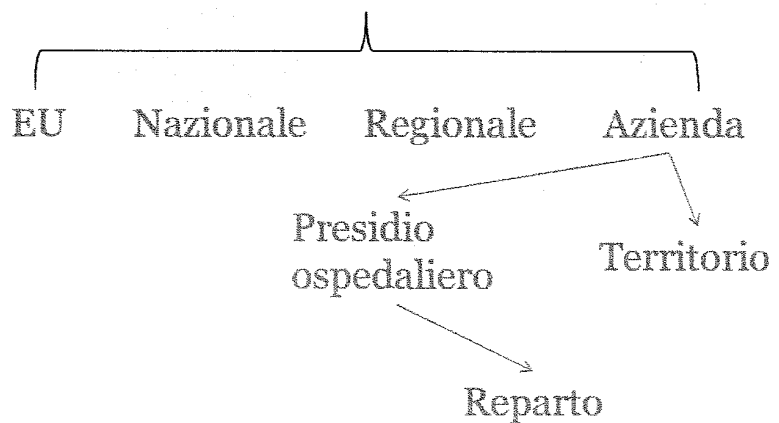
Le applicazioni possono variare

asestate

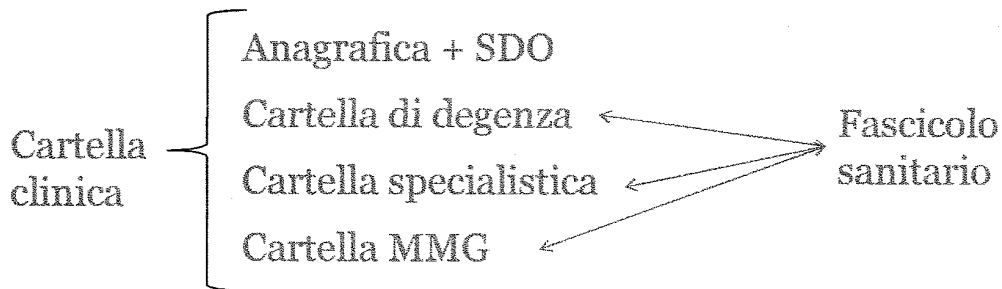
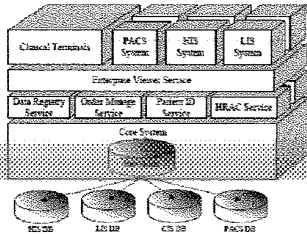


ricerca

Livello



EHR (electronic health record) that integrates patient data generated in the care processes and enables those that need it and have the proper authorization to access it.



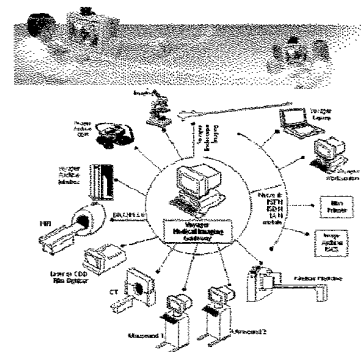
### eHealth Applications

Software that allow data exchange among different organizations

It can be:

1. Software used for the management procedures
2. Software used for the management procedures that have an indirect link with patient care
3. Software used directly for patient care (medical device software)

Telemedicine is the area of medical informatics that includes all the applications used to exchange data among different health facilities.



## L'eHealth nel contesto italiano



23

## Esempi

14-20 giugno 2011

24 ORE  
**Sanità**

aziende/territorio **19**

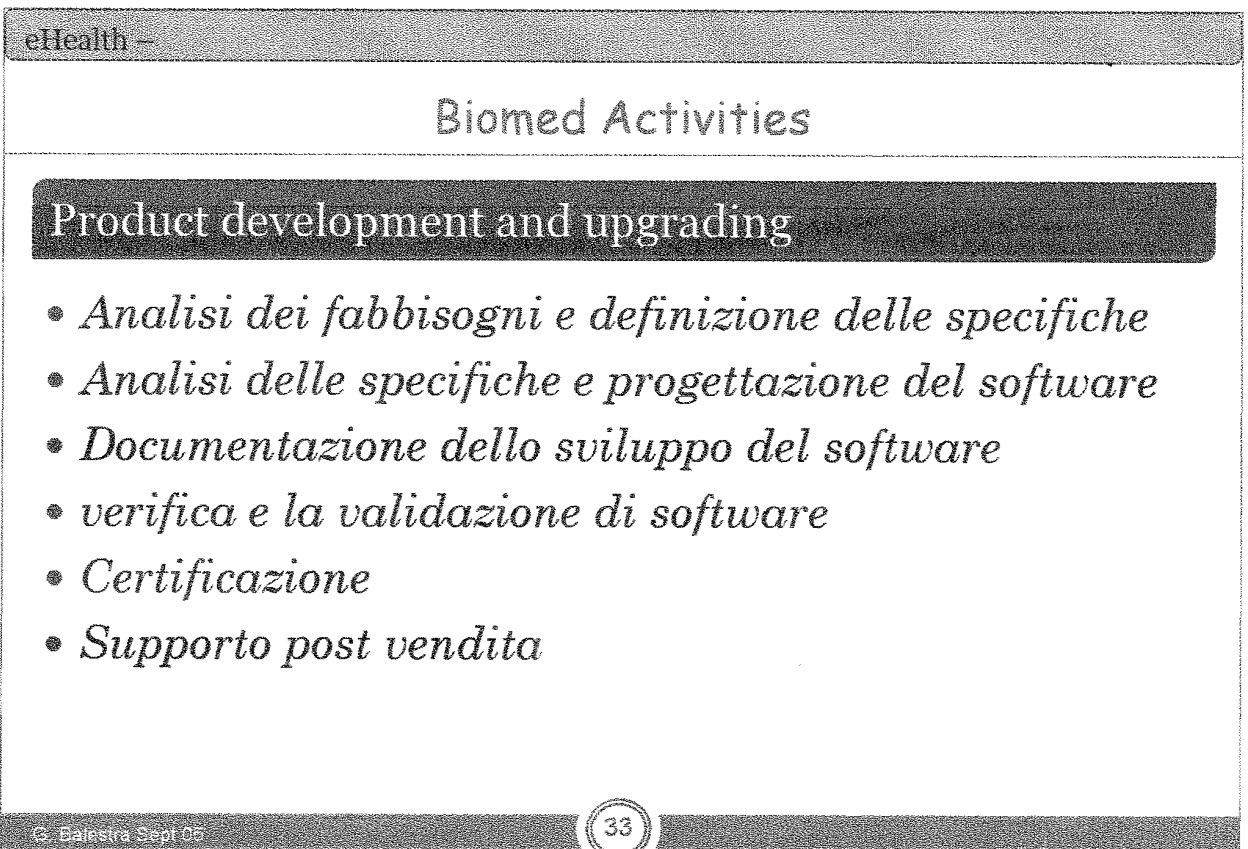
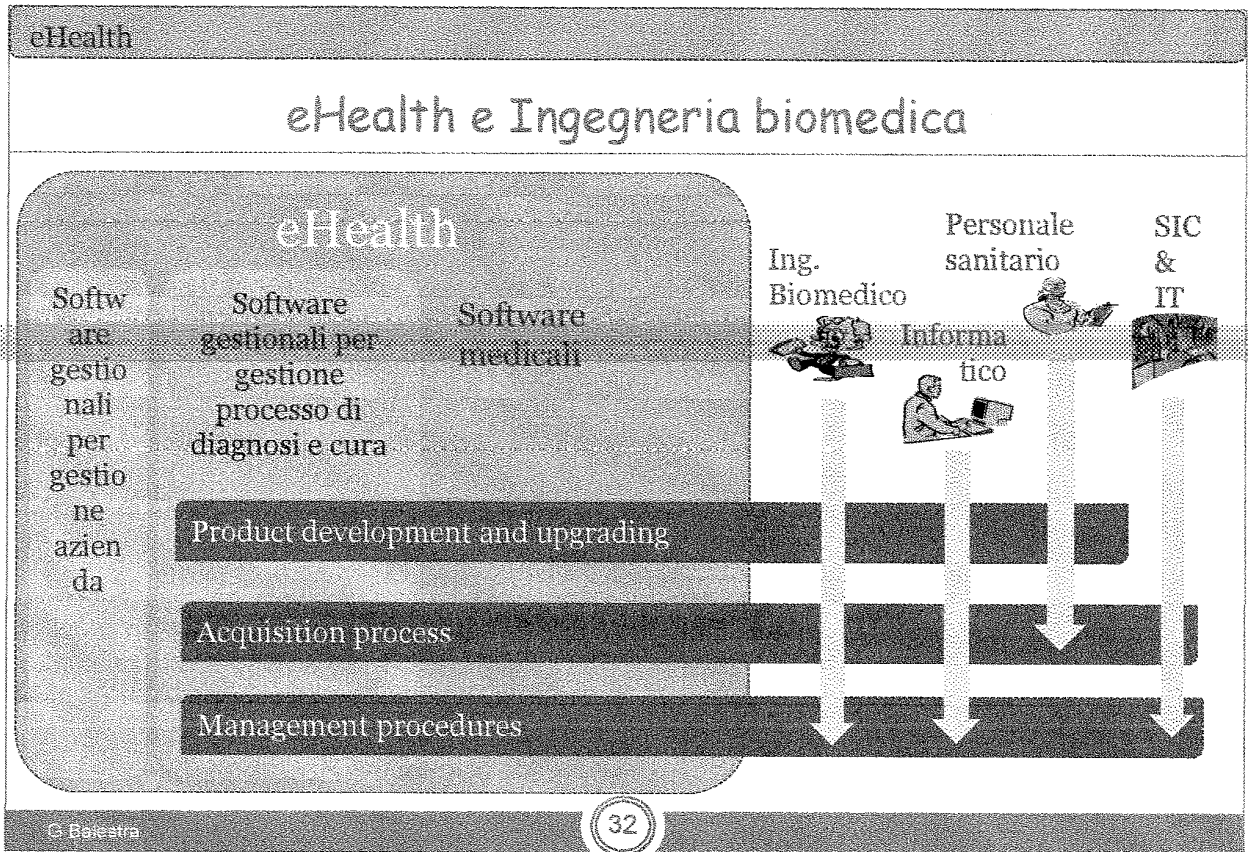
PIEMONTE/ Al S. Giovanni Bosco di Torino un sistema informatizzato per le terapie

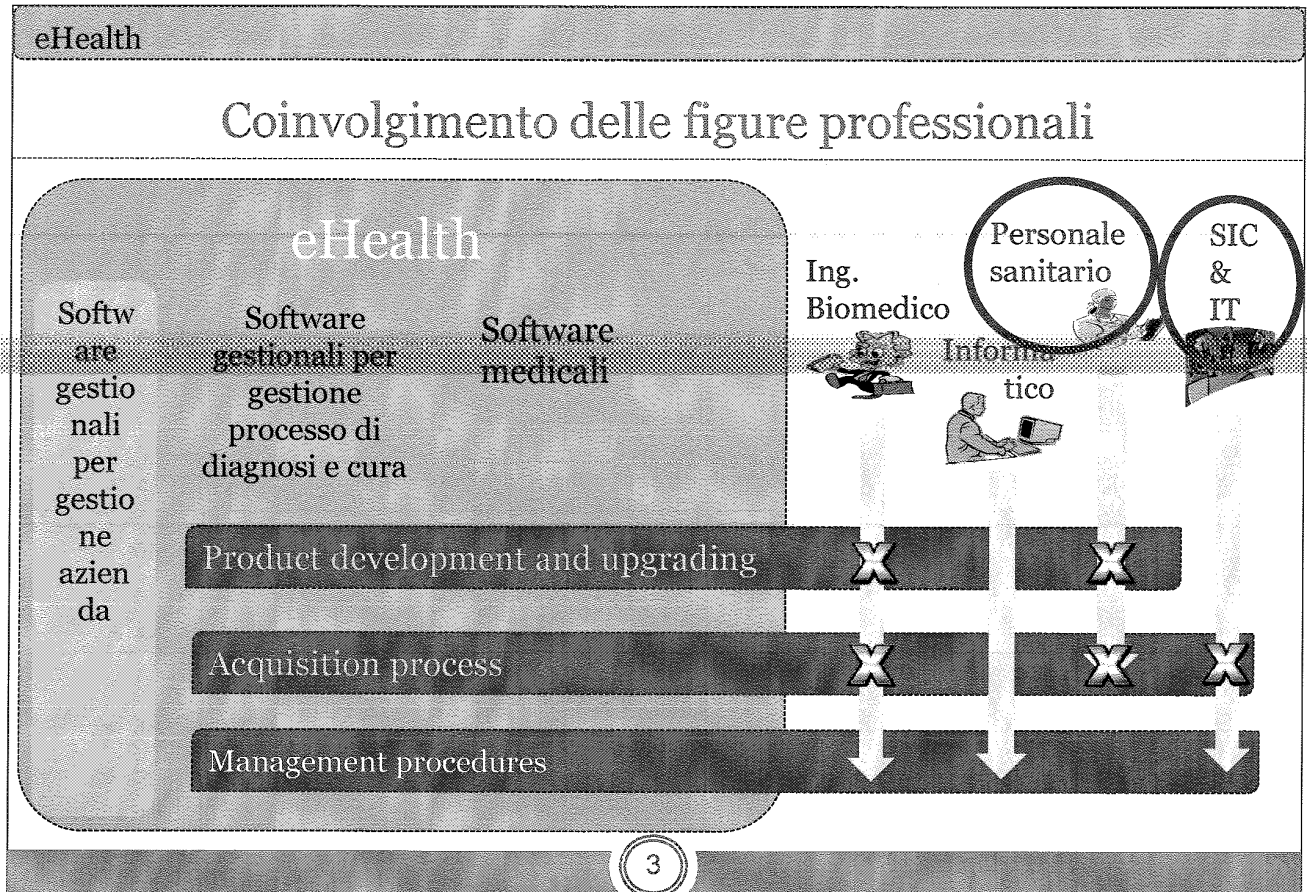
## Un software gestisce i farmaci

A disposizione degli operatori i dati clinici del paziente e il Prontuario









eHealth -

## Biomed Activities

### Product development and upgrading

- *Analisi dei fabbisogni e definizione delle specifiche*
- *Analisi delle specifiche e progettazione del software*
- *Documentazione dello sviluppo del software*
- *verifica e la validazione di software*
- *Certificazione*
- *Supporto post vendita*

### Acquisition process

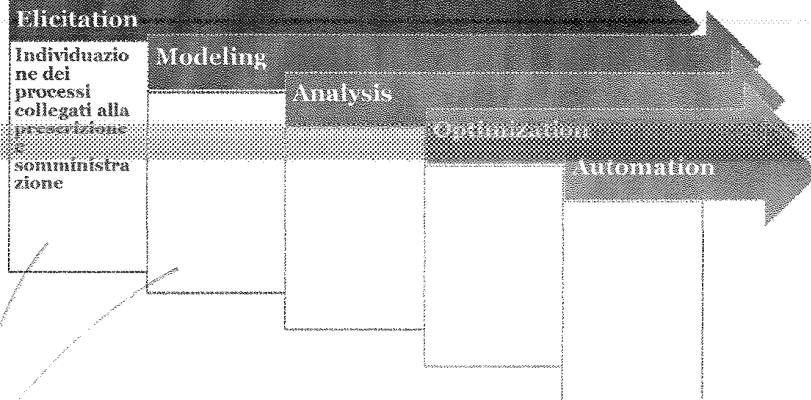
- *Analisi dei fabbisogni e definizione delle specifiche*
- *Valutazione offerte*
- *Collaudo*

4

7

## ESEMPIO

### Informatizzazione della gestione della terapia



**BACKGROUND:** conoscenza di come funziona un reparto ospedaliero

**INTERVIEWS:** bisogna individuare chi

**OBSERVATION:** per verificare che quello che è emerso dalle interviste corrisponda al reale svolgimento dei processi

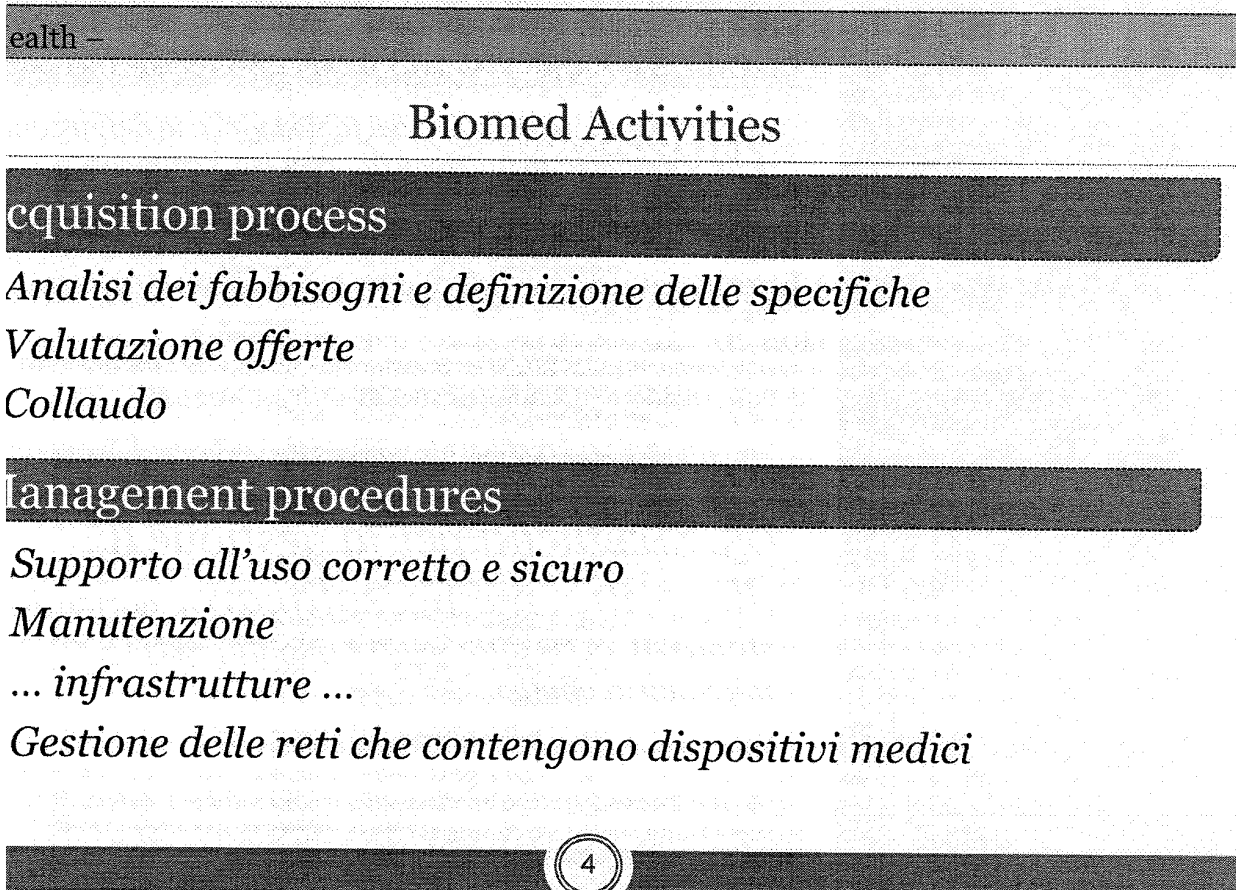
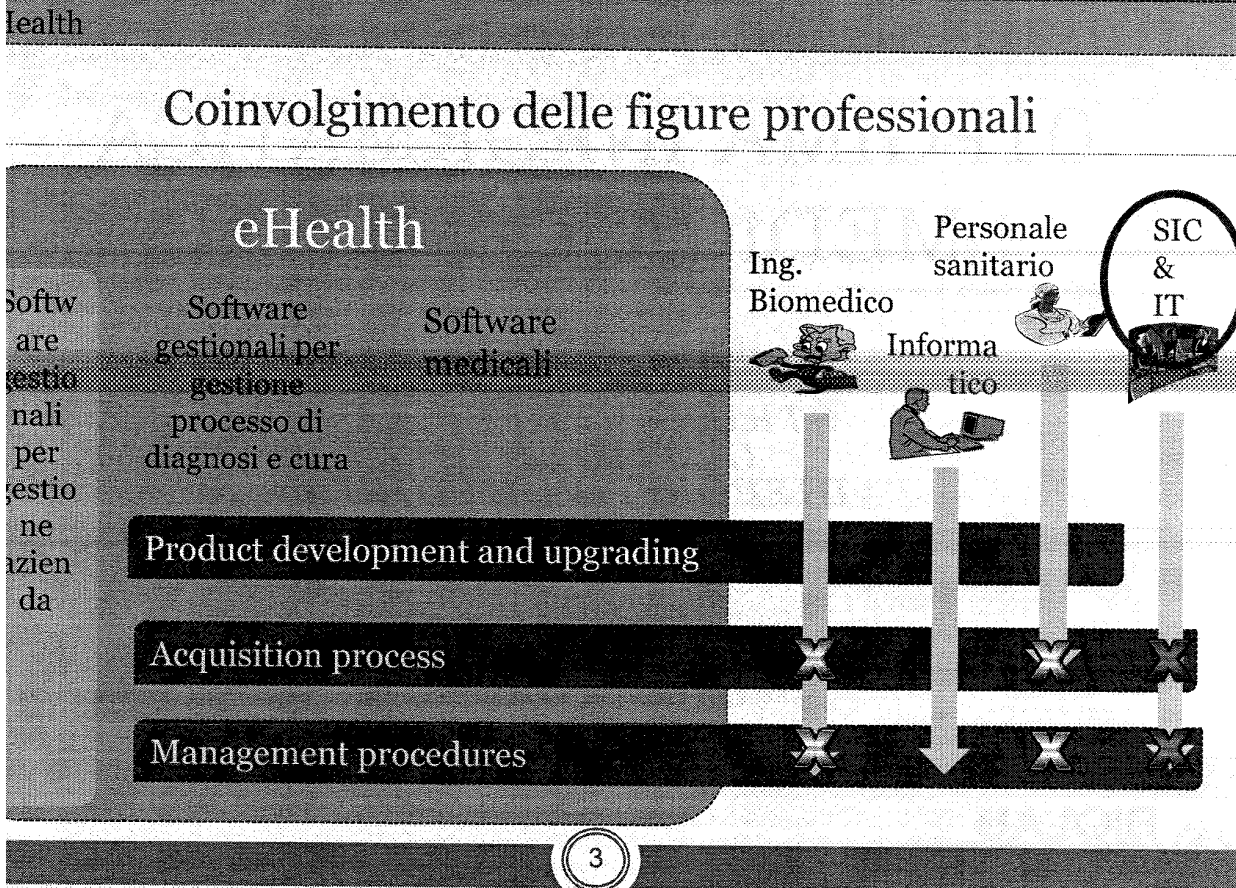
8

## MODELING

### Esistenza di un sistema

Il modello dipende dall'obiettivo e da chi lo costruisce

Individuare quali componenti del sistema considerare



## Rischio

Dal punto di vista della analisi del rischio le tecnologie presentano tre tipologie di rischi:

- INTRINSECO: rischi legati ad una scorretta progettazione o ad un malfunzionamento
- GESTIONALE: i rischi che derivano da una mancata o non corretta gestione
- USO: rischi derivanti da un uso non corretto.

## Rischio intrinseco

La normativa relativa ai dispositivi medici e la certificazione consentono di ridurre sensibilmente i rischi legati ad una progettazione non corretta.

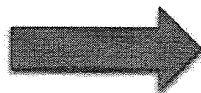
La possibilità di usare controlli inseriti nel firmware del dispositivo riduce la necessità di controlli da parte degli operatori ed aumenta la probabilità di detezione dei malfunzionamenti.

- diagnosis, prevention, monitoring, treatment or alleviation of disease.
- diagnosis, monitoring, treatment, alleviation of or compensation for an injury or handicap.
- investigation, replacement or modification of the anatomy or of a physiological process,

and which does not achieve its principal intended action in or on the human body by pharmacological, immunological or metabolic means, but which may be assisted in its function by such means;

## Medical Device Softwares (MDS)

It is necessary to clarify that software in its own right, when specifically intended by the manufacturer to be used for one or more of the medical purposes set out in the definition of a medical device, is a medical device. Software for general purposes when used in a healthcare setting is not a medical device.



When a software stand alone is a medical device?

## *Clinical Engineering - the origins*

✓ **1971:** Ralph Nader shocked the nation by reporting in the Ladies Home Journal that thousands died in hospitals yearly from electrocution. At the beginning, the problem of technology management was mostly concerned with electrical safety.

✓ **1972:** United States Department of Veteran's Affairs prepared VA Biomedical Engineering Program to create a consolidated program for instrumentation management and biomedical engineering.

✓ **80s:** technology became more sophisticated and safer, but the number of devices increased dramatically. The problem became how to correctly manage the devices maintenance, how to purchase the most suitable instrument and so on.

## *Clinical Engineering - the evolution*

Inizialmente l'attività dei SIC (Servizi di Ingegneria Clinica) era basata principalmente sugli aspetti di sicurezza elettrica. Negli ultimi vent'anni si è passati a parlare di Healthcare Technology Management (HTM) cioè di gestione delle tecnologie biomediche puntando l'attenzione sui processi (manutenzione, acquisizione, ...).

Questo cambiamento è ampiamente giustificato dall'evoluzione della tecnologia che ha di molto ridotto i problemi di sicurezza elettrica aumentando la sicurezza intrinseca delle singole apparecchiature, mentre il numero di dispositivi è aumentato pesantemente richiedendo sempre di più lo sviluppo di procedure efficaci ed efficienti per garantirne la disponibilità oltre che l'uso sicuro.

## Le figure professionali

### Ingegnere clinico

- ✓ direttore
- ✓ assistente
- ✓ specialista di settore

*specializzazione orientata alla funzione*

### Tecnico biomedico

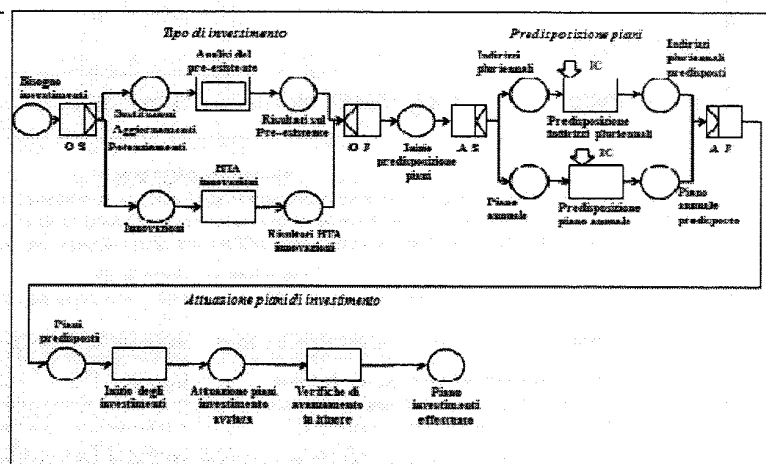
- ✓ coordinatore
- ✓ addetto alla gestione manutentiva
- ✓ addetto alla gestione ricambi, accessori, consumabili
- ✓ addetto ai collaudi, ai fuori uso e all'aggiornamento dell'inventario tecnologico
- ✓ addetto al call-center tecnico
- ✓ addetto ai controlli funzionali e di sicurezza

19

## Attività di un SIC

ATTIVITÀ: COORDINAMENTO E ORGANIZZAZIONE

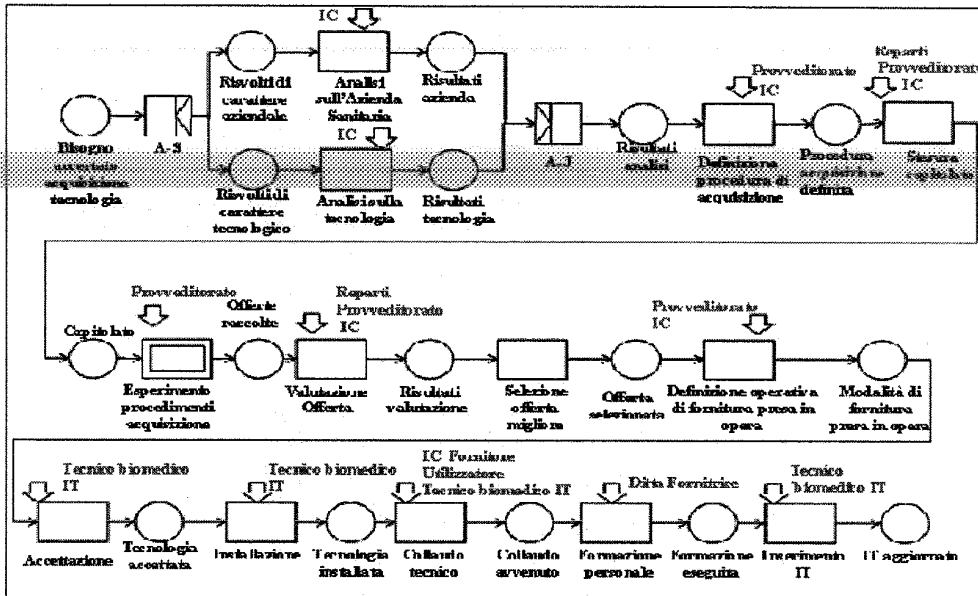
ATTIVITÀ: PARTECIPAZIONE AL PROCESSO DI PROGRAMMAZIONE DEGLI INVESTIMENTI TECNOLOGICI



20



## ATTIVITÀ: ACQUISIZIONE



23

## Clinical engineering in Italy

<http://www.aiic.it>



La situazione dei Servizi di Ingegneria Clinica (SIC) in Italia è piuttosto variegata e disomogenea: si va dalla struttura complessa al global service passando per tutti i possibili tipi di organizzazione possibili



Clinical engineering in Regione Piemonte (ITALY) in the last twenty years grew in a very diversified and fragmented way.



Only a few healthcare facilities have a Clinical Engineering (CE) department

## Elementi che compongono il modello proposto

Attività

Figure professionali

Risorse umane

Organizzazione

Modello dimensionamento Staff

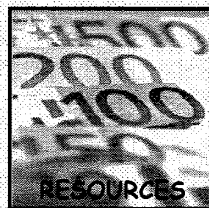
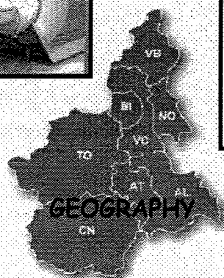
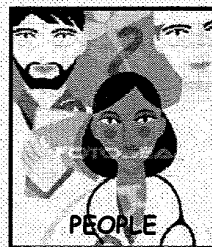
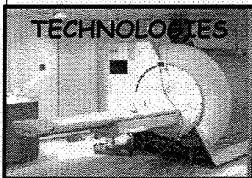
27

## The model

AIM

To create a sustainable model of regional network of CE services

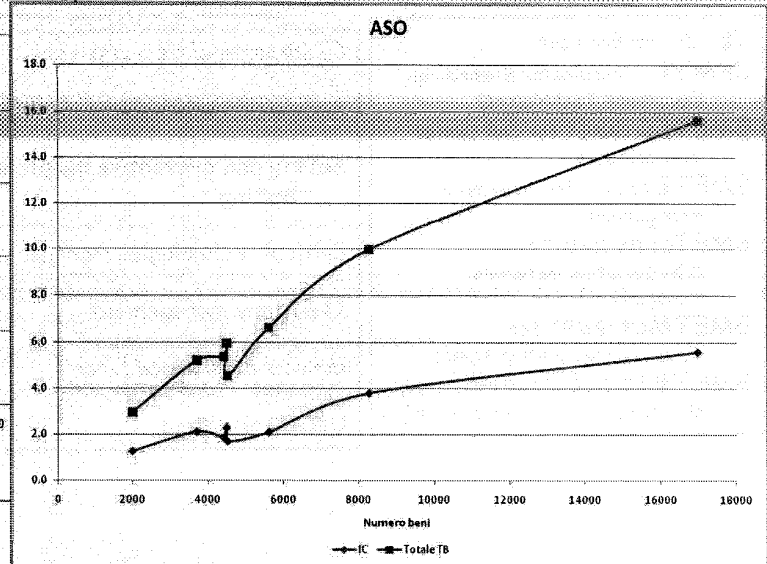
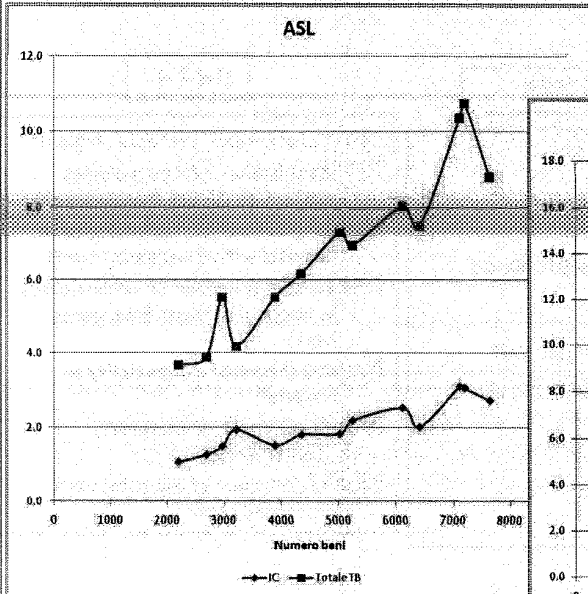
- ✓ Consistent with available resources
- ✓ Able to generate consensus among people involved in the process (like physicians, managers, etc.)



Number of CE Departments  
Staff composition

28

## Risultati della applicazione del modello





31

## Linea guida regionale

**Gennaio 2010**

### INDICE

- Scopo
- Funzioni
- Aree tecnologiche di competenza
- Profili professionali
- Organizzazione del servizio
- Dimensionamento
- Modello "misto"
- Sovrazionalità
- Richiamo ai documenti A.Re.S.S.

 	
<b>LINEA GUIDA PER L'APPLICAZIONE DI UN MODELLO ORGANIZZATIVO REGIONALE DI SERVIZI DI INGEGNERIA CLINICA (S.I.C.)</b>	
Indice	
1. Scopo	2
2. Funzioni	3
3. Aree tecnologiche di competenza	3
4. Profili professionali	4
5. Organizzazione del servizio	5
6. Dimensionamento	5
7. Modello "misto"	7
8. Sovrazionalità	8
9. Richiamo ai documenti A.Re.S.S.	9
<small>                     Referenza a cura di:                      Agenzia Regionale per i Servizi Sanitari del Piemonte (A.Re.S.S.)                      Area Organizzativa e Programmatoria - Progetto Health Technology Management                 </small>	

FEB 08 GB

32

## WHY A STANDARD FOR RISK MANAGEMENT OF MEDICAL NETWORKS?

Individual medical devices are designed and validated to be safe for their intended use. When they are incorporated into heterogeneous networks that include other medical devices and IT components, a new system is created, and the medical device has not been validated with that new system. The safety requirements and constraints identified by the manufacturer as necessary to ensure the safety of the medical device may not adequately control hazards in this new system. Additionally, new hazards may emerge that involve interactions among the network components that were not considered when the medical device was designed, and new uses of the medical device may emerge that were not part of the medical device validation.

The goal of 80001 is to prevent patient harm. But 80001 extends the meaning of harm beyond the idea of injury to the patient. In addition to physical injury or damage to the health of a patient, harm includes a reduction of effectiveness or a breach of data and system security.

80001 covers the entire life cycle of the medical IT-network. After risk management for a new or modified network has been completed and put into use in the patient environment, monitoring will be necessary to ensure that the key properties are not impacted by the dynamic characteristics of that environment.

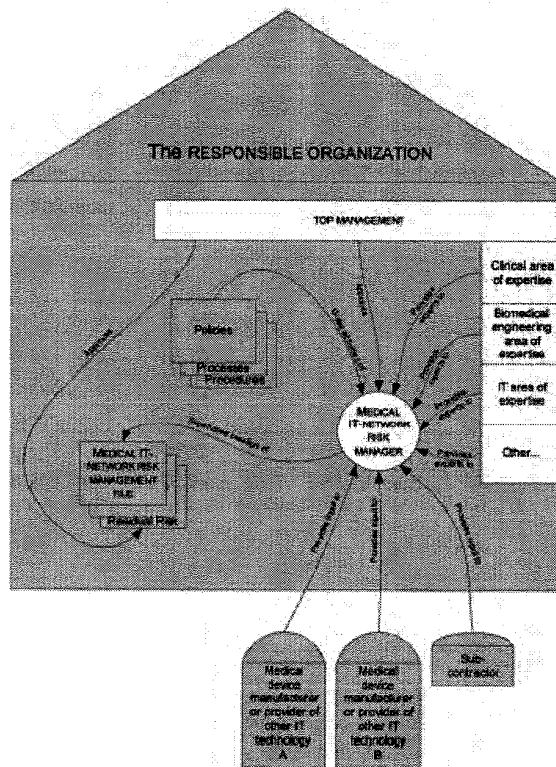


Figure 1. Overview of Roles and Relationships from 80001  
[IEC 80001-1:2010, Figure B.1]



# 01PGIMA INFORMATICA MEDICA / Bioimmagini

## REQUIREMENTS ANALYSIS

**Gabriella Balestra**



<http://socrate.polito.it/biolab>

2

### 00. INTRODUCTION

*01. Strumenti per l'analisi dei fabbisogni e la definizione delle specifiche: definizione di processo, strumenti per l'analisi e la descrizione dei processi*

*02. Strumenti per l'analisi delle specifiche e la progettazione di software: UML (Use Case Diagram, Use Case Details, Oggetti, Activity Diagrams), Sistemi di codifica*

*03. Strumenti per la documentazione dello sviluppo di software: UML (...), definizione e principali caratteristiche di una base dati*

*04. Strumenti per la verifica e la validazione di software: differenza tra verifica, validazione e collaudo: UML (piano delle prove e check list)*

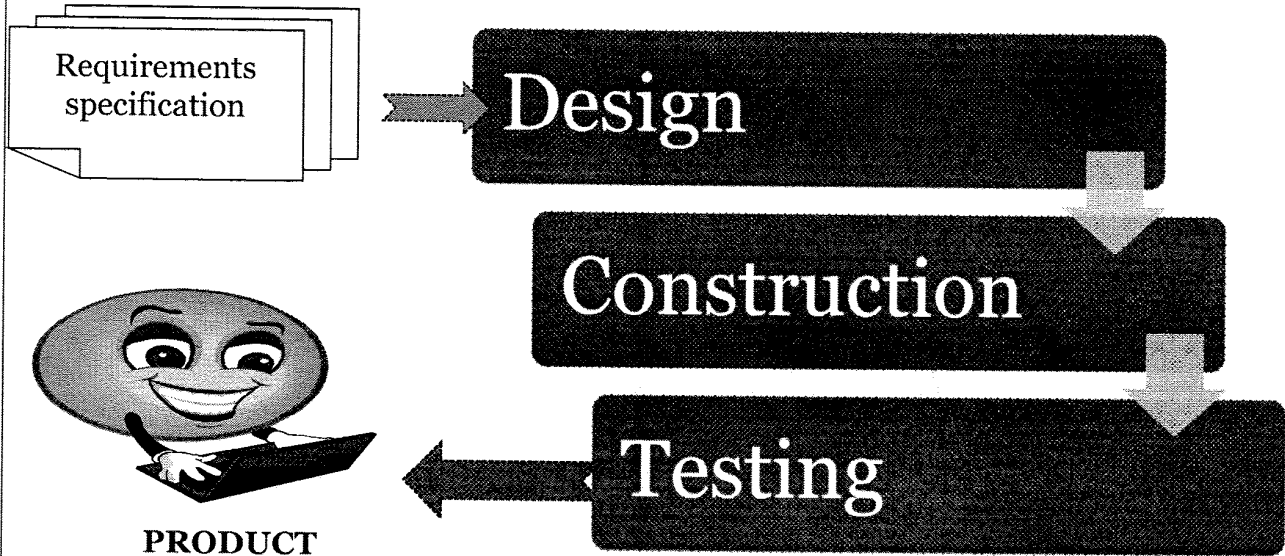
*05. Strumenti per la gestione di dispositivi medici software: normativa e standard, accreditamento e qualità, servizi di ingegneria clinica*

*06. Principali applicazioni e problematiche aperte*

*07. Innovazione e futuro dei dispositivi medici software: sistemi di supporto alla decisione clinica, HTA*

## System development

System development is a complex task that can be view as a process with three main steps.



5

## Object programming

The intent of object-oriented programming (OOP) is to improve the development of computer programs.

Before OOP, computer programs were based on the concept of a *function* or *procedure* (or subroutine). Computer languages that support this concept are referred to as *procedural languages*.

A function allows a programmer to break a large list of instructions down into smaller units that can be executed repeatedly. Ideally, each function has a clearly defined purpose and interface to other functions in the program.

The relation between data structures and functions is not visible in procedural programming languages.

Since data structures can be accessed by different functions in a program, it is important that changes to the data structure be reflected in all functions that access the data structure.

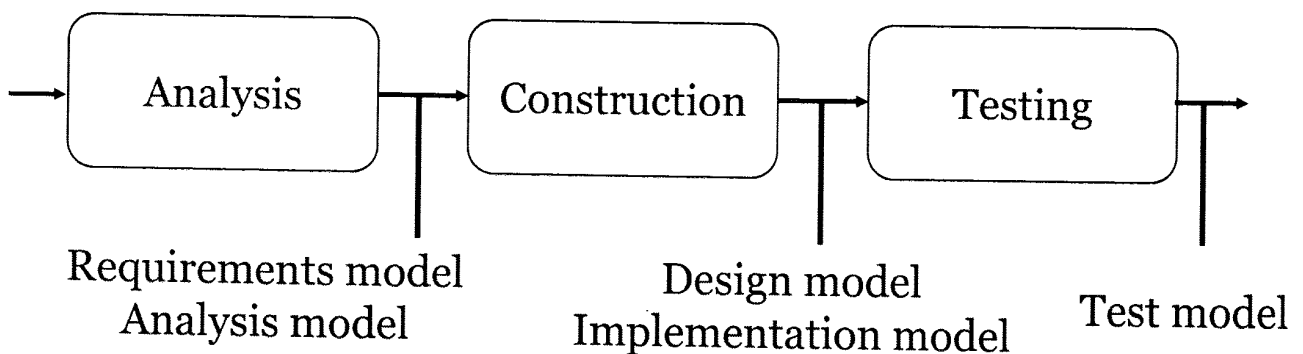
6

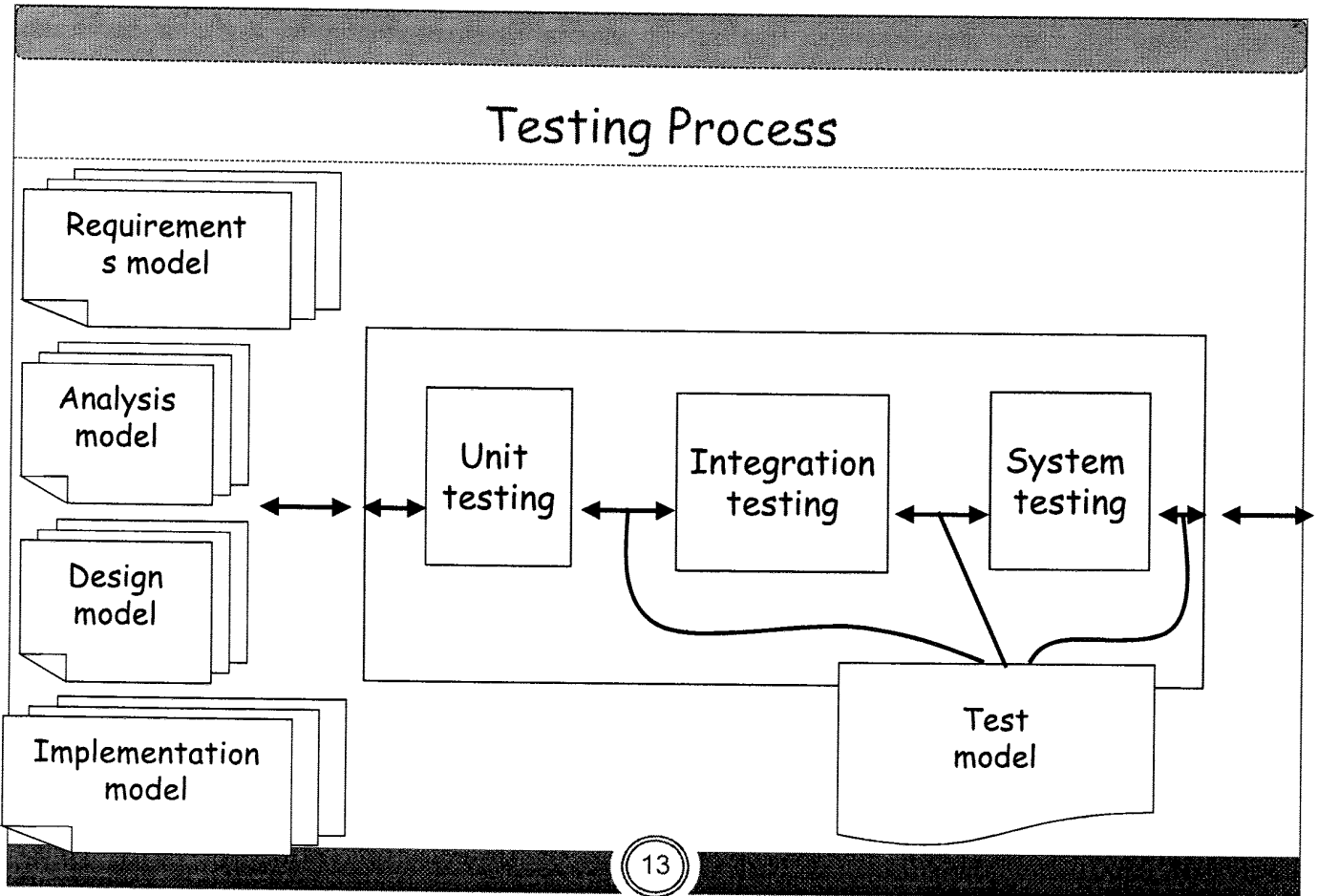
## UML can be used as ...

- ... *a sketch*: UML is used to make brief sketches to convey key points.
- ... *a blueprint*: provide detailed specification of a system with UML diagrams. These diagrams would be generated with a UML tool. This approach usually involves using forward and reverse engineering to keep the model synchronized with the code.
- ... *a programming language*: this goes directly from a UML model to executable code (not just portions of the code as with forward engineering), meaning that every aspect of the system is modeled. Theoretically, you can keep your model indefinitely and use transformations and code generation to deploy to different environments.

## UML v 1.x

**System development** is a complex task that can be view as a process of model building. The result of the process is a **sequence of models** that allow for a better management of both **development and maintenance**.





## UML v 2.x

Many of the original diagrams and associated notations have been retained and extended in UML 2.0.

New diagrams types has been introduced to extend the language just enough so that it can support the latest best practices.

**Release date: August 2011**

**Specification Name:** Unified Modeling Language™ (UML®)

**Description:** A specification defining a graphical language for visualizing, specifying, constructing, and documenting the artifacts of distributed object systems.

**Keywords:** abstraction, action sequence, action state, activity graph, architecture, association, class diagram, collaboration diagram, component diagram, control flow, data flow, deployment diagram, execution, implementation, pins, procedure.

**Latest / past specifications:** Current version (Infrastructure and Superstructure): 2.4.1 Past versions

**NOTE:** Version 2.0 does not have XSD or XML associated files due to structural problems with the UML metamodel.

**Practitioner Certification:** OCUP (OMG-Certified UML Professional) <http://www.omg.org/ocup/>

**Related OMG Specifications:** Diagram Interchange, OCL, MOF, XMI

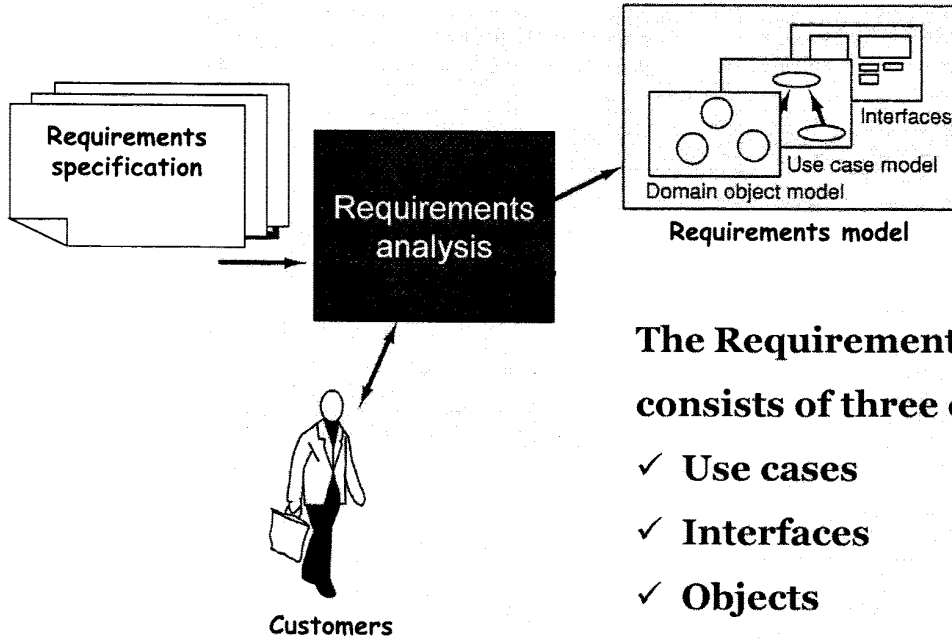
**Related Industry Standards:** UML 1.4.2 is available as ISO/IEC 19501; 19505 assigned to UML 2.1.2; ITU-T Recommendations Z.100 (SDL) and Z.109 (SDL UML profile).

<b>Most recent IPR and Implementation questionnaire responses:</b>	<b>BCQ</b> Oracle	<b>Borland Softeam</b>	<b>IBM Sparx Systems</b>	<b>Kabira THALES</b>
--	----------------------	------------------------	--------------------------	----------------------

14



# Results of the Requirements Analysis Process



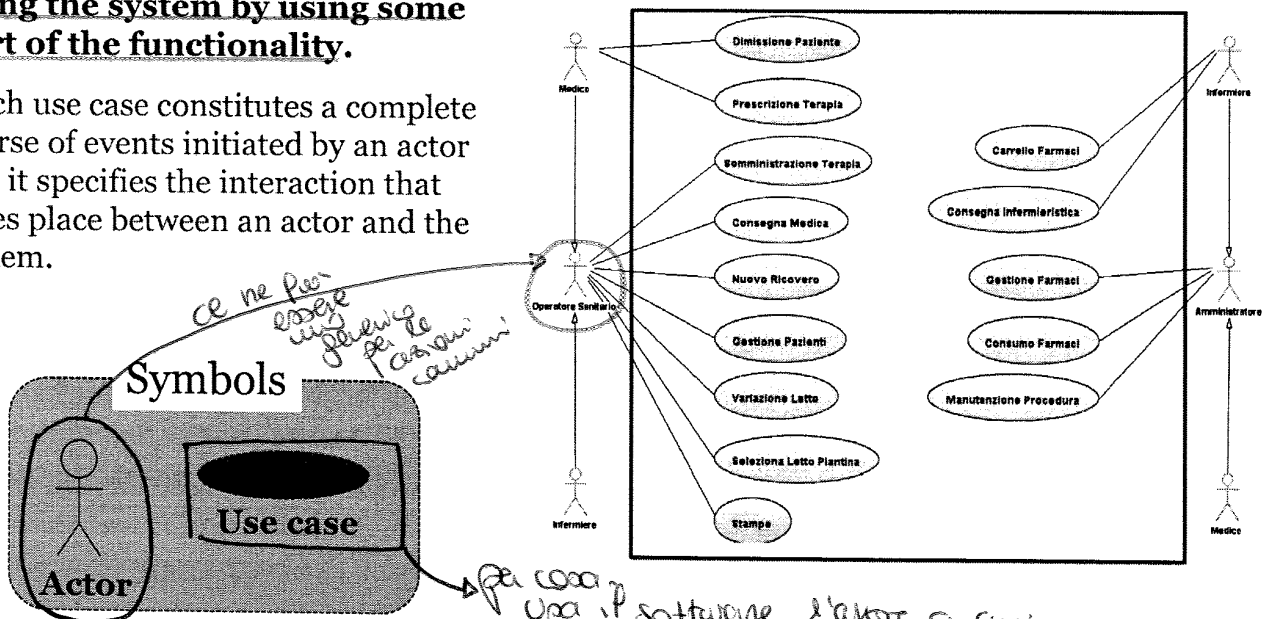
The Requirements model consists of three components:

- ✓ Use cases
- ✓ Interfaces
- ✓ Objects

## Use Cases

**A use case is a specific way of using the system by using some part of the functionality.**

Each use case constitutes a complete course of events initiated by an actor and it specifies the interaction that takes place between an actor and the system.



## Use case details

<b>Use Case Name:</b>	<i>Use case identifier and reference number and modification history</i>
<b>Goal:</b>	<i>Goal to be achieved by use case and sources for requirement</i>
<b>Actor(s):</b>	<i>List of actors involved in use case</i>
<b>Trigger:</b>	.....
<b>Main Flow:</b>	<p><i>Interactions between actors and system that are necessary to achieve goal</i></p> <p>The sequence of interactions necessary to successfully meet the goal. The interactions between the system and actors are structured into one or more steps which are expressed in natural language. A step has the form &lt;sequence number&gt;&lt;interaction&gt;</p>
<b>Alternative Flows:</b>	<p><i>Any variations in the steps of a use case</i></p> <p>Further detail about a step may be given by listing any variations on the manner or mode in which it may happen.</p>
<b>Form</b>	<i>The software interfaces related to the use case</i>

<b>TITOLO</b>	<b>GESTIONE PAZIENTI</b>
<b>Obiettivo</b>	Consentire di attivare le varie funzionalità relative ai pazienti
<b>Precondizioni</b>	
<b>Trigger</b>	Da un qualunque caso d'uso viene cliccato il pulsante Pazienti
<b>Attori</b>	Medico
<b>Percorso principale</b>	<ol style="list-style-type: none"> <li>1. Viene visualizzata la form <i>Pazienti</i></li> <li>2. Vengono visualizzati i nomi dei pazienti attivi prendendo i dati dall'oggetto entità LISTA PAZIENTI</li> <li>3. A seconda di quale pulsante si clicca si passa al corrispondente caso d'uso                      NUOVO: passa il controllo al caso d'uso NUOVO PAZIENTE                      NOME PAZIENTE: passa il controllo al caso d'uso PAZIENTI ATTIVI                      ARCHIVIO: passa il controllo al caso d'uso ARCHIVIO PAZIENTI                      Protocolli: passa il controllo al caso d'uso GESTIONE PROTOCOLLI                      Export: passa il controllo al caso d'uso EXPORT</li> </ol>
<b>Percorso Alternativo 1</b>	<ol style="list-style-type: none"> <li>1. Viene visualizzata la form <i>Pazienti</i></li> <li>2. Viene cliccato il pulsante ESCI: il programma termina</li> </ol>
<b>FORM</b>	<p><b>Form <i>Pazienti</i></b></p> 