



Corso Luigi Einaudi, 55 - Torino

Appunti universitari

Tesi di laurea

Cartoleria e cancelleria

Stampa file e fotocopie

Print on demand

Rilegature

NUMERO: 1009

DATA: 14/07/2014

A P P U N T I

STUDENTE: Tortorici

MATERIA: Informatica Medica + Schemi

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.

INFORMATICA MEDICA

Professoressa G. Balestra

INDICE

informatica medica.....	2
eHealth.....	2
EHR.....	2
telemedicina.....	3
bioinformatica.....	3
processo di informatizzazione.....	3
analisi dei fabbisogni.....	4
process modeling.....	4
synopsis diagram.....	6
workflow diagram.....	9
flowchart.....	12
swim lane activity diagram.....	12
organigramma.....	13
stakeholder diagram.....	13
analisi delle specifiche.....	13
diagramma dei casi d'uso.....	14
use case details.....	15
oggetti.....	16
interfacce.....	16
control objects.....	19
requirements validation.....	20
entity objects.....	20
construction process.....	23
data base.....	26
testing.....	32
sistema sanitario nazionale.....	36
rischio.....	36
ingegneria clinica.....	36
standard.....	37
indicatori per la valutazione dell'informatizzazione dei processi clinici.....	38
dispositivo medico.....	45
HTA.....	45
POSSIBILI DOMANDE	

Anamnesi: storia clinica.

TELEMEDICINA

È l'area dell'informatica medica che include tutte le applicazioni usate per lo scambio di dati tra varie strutture.

BIOINFORMATICA

Utilizza computer e altri strumenti del genere per aiutare con osservazioni biologiche di base, come il genoma o le sequenze di proteine, e dirigere misure di processi molecolari e cellulari.

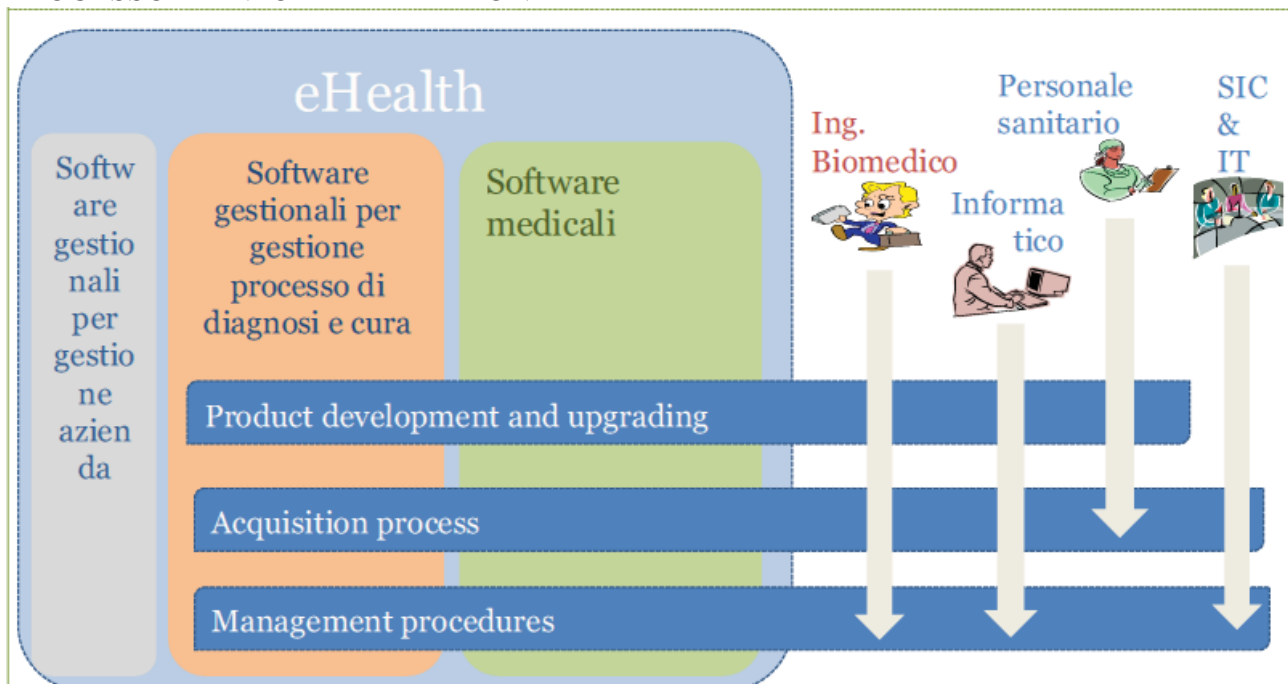
La bioinformatica è il concepire la biologia in termini di macromolecole e poi applicare tecniche informatiche (derivate da discipline quali matematica applicata, informatica e statistica) per comprendere e organizzare le informazioni associate con queste molecole.

Nel 2007 **Bansard** e altri hanno pubblicato uno studio in cui hanno riportato l'analisi della letteratura in ambito di bioinformatica e di informatica medica. Lo studio mostra che:

- genoma e proteoma non sono ancora inclusi nell'ambito medico;
- gli aspetti dell'informatica medica relativi all'organizzazione degli ospedali e alla gestione dei pazienti sono lontani dagli argomenti della bioinformatica;
- bioinformatica e informatica medica si sovrappongono nell'usare gli stessi metodi matematici e informatici.

Si è anche trovato che la tendenza della bioinformatica verso la medicina è maggiore di quella dell'informatica medica verso la bioinformatica.

PROCESSO DI INFORMATIZZAZIONE



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 validazione del software;
- certificazione;

mettere in evidenza.

I modelli servono a:

- documentare procedure;
- identificare debolezze nel processo;
- valutare miglioramenti;
- comunicare la logica del processo;
- aiutare nell'automatizzazione;
- acquisire consenso;
- facilitare la comprensione dei processi per i nuovi impiegati.

ESEMPIO: informatizzazione della gestione della terapia.

Background: conoscenza di come funziona un reparto ospedaliero e delle figure professionali che lavorano nel reparto. Il capo dei medici è il primario; il capo degli infermieri è il caposala.

La prescrizione avviene in 3 momenti:

- *prescrizione iniziale* → quando il paziente viene ricoverato;
- *revisione* → avviene giornalmente;
- *modifica parziale* → avviene in seguito a un evento (consultazione specialista, cambio delle condizioni, ecc.).

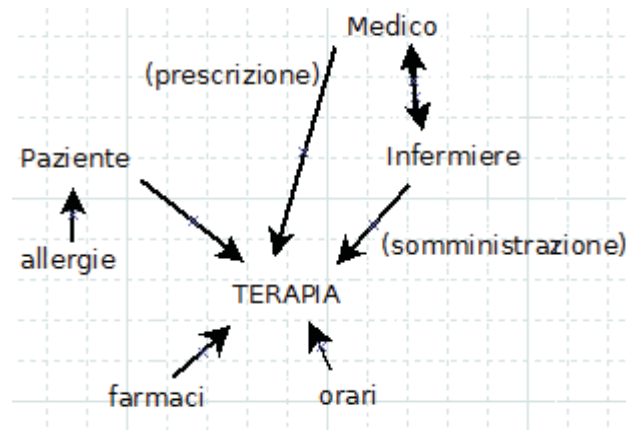
Vi è anche la *terapia al bisogno*: si tratta di farmaci che gestiscono problemi generali (emicrania, nausea, ecc.). Può essere gestita in due modi:

- in una sezione del foglio di terapia, il medico la scrive a priori;
- l'infermiere informa il medico, il quale decide sul momento.

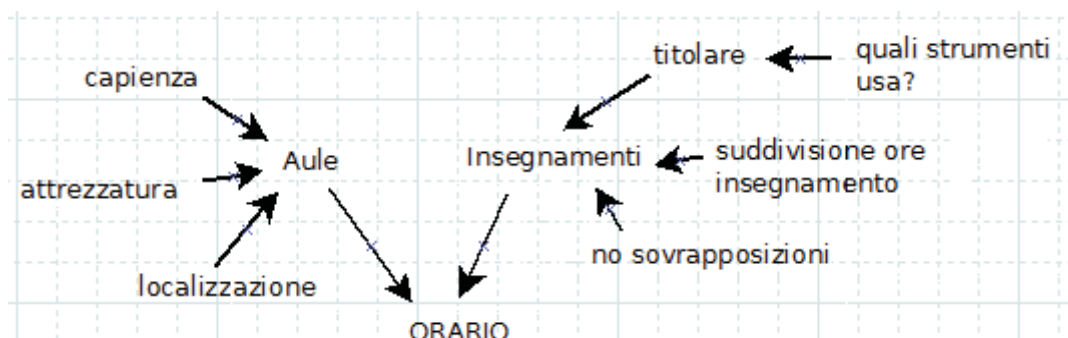
La *somministrazione* avviene per fasce orarie: gli infermieri devono confermare che sia avvenuta o spiegare perché non è stata effettuata.

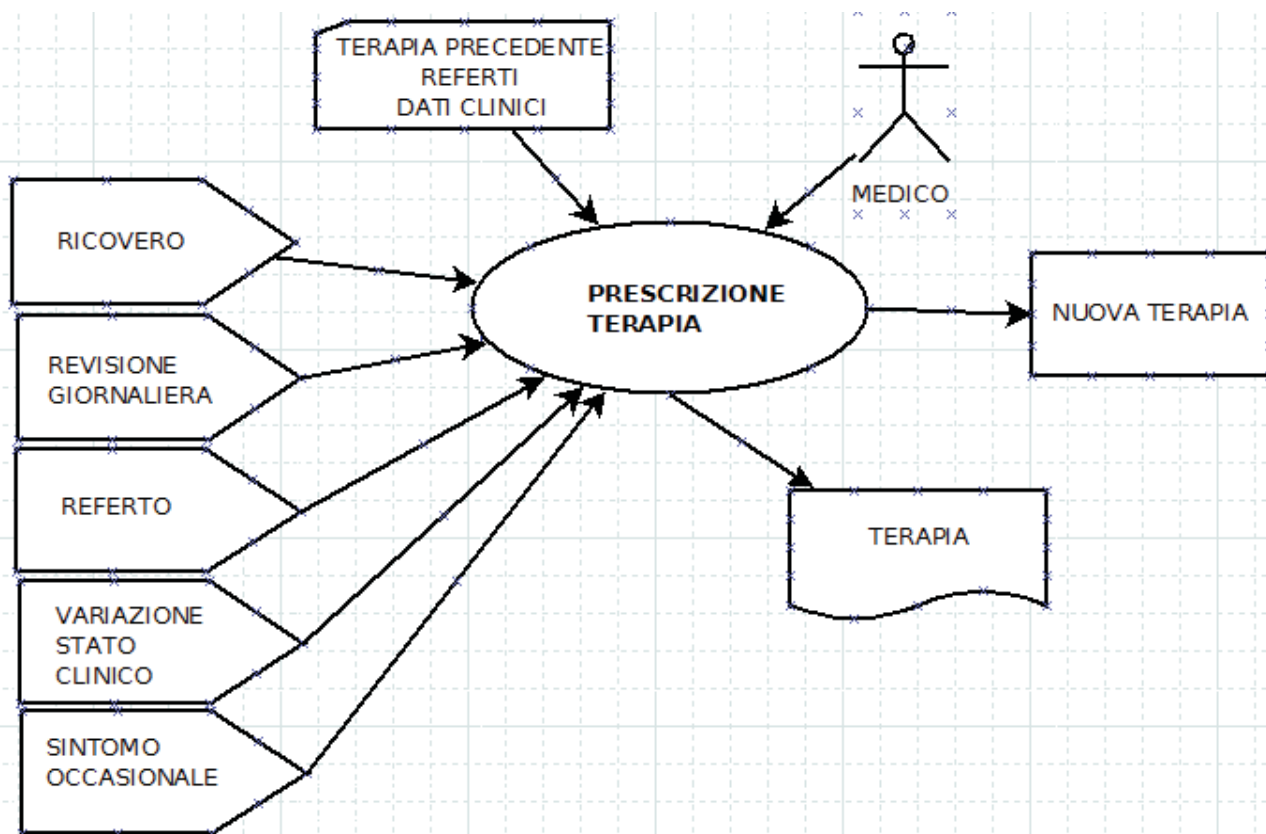
Le eventuali allergie del paziente sono segnalate nell'*anamnesi*.

Interviste: di solito si intervistano caposala e primario: se loro lo indicano, si può parlare anche con altri medici o infermieri.

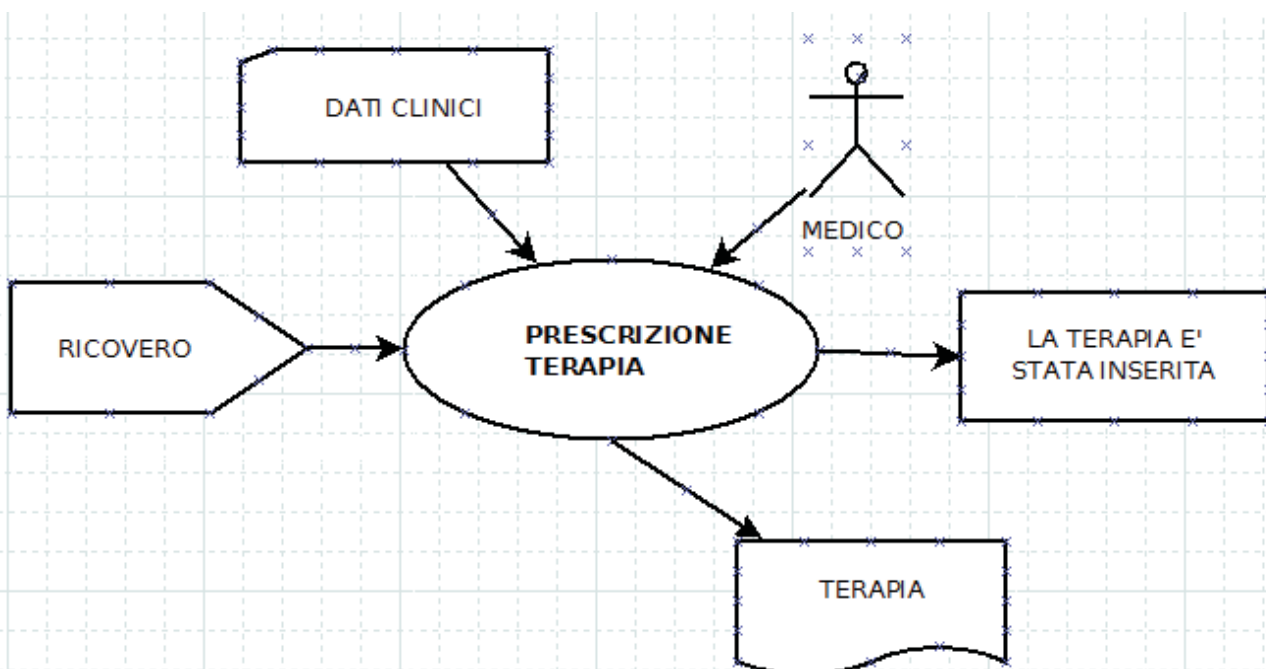


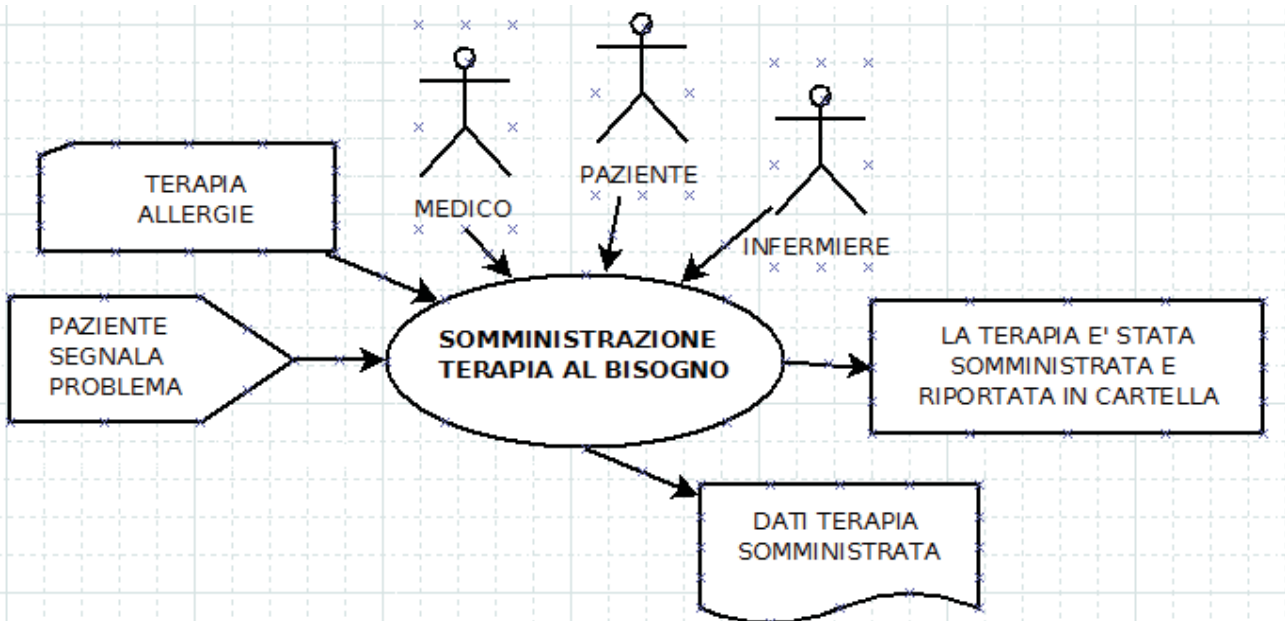
ESEMPIO: definizione degli orari al Politecnico





Prescrizione terapia: seconda possibile modellizzazione.



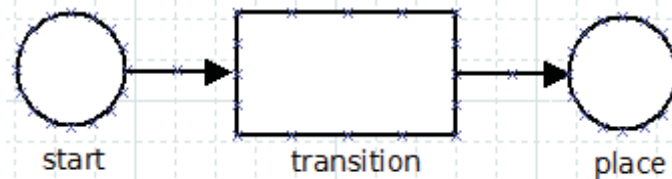


WORKFLOW DIAGRAM

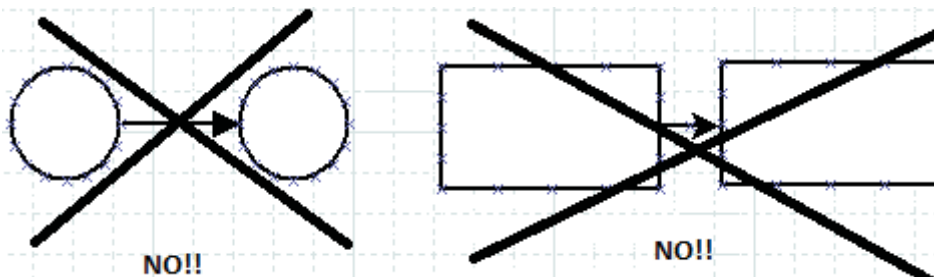
Descrive le azioni svolte durante un processo. Si basa sul formalismo delle *reti di Petri*, basato su due elementi:

place → stato → ○
 transition → attività → □

I place vanno collegati alle transition e le transition vanno collegate ai place. La connessione avviene tramite archi direzionali. Un workflow inizia e termina con un place.



NON si possono collegare due transition o due place!



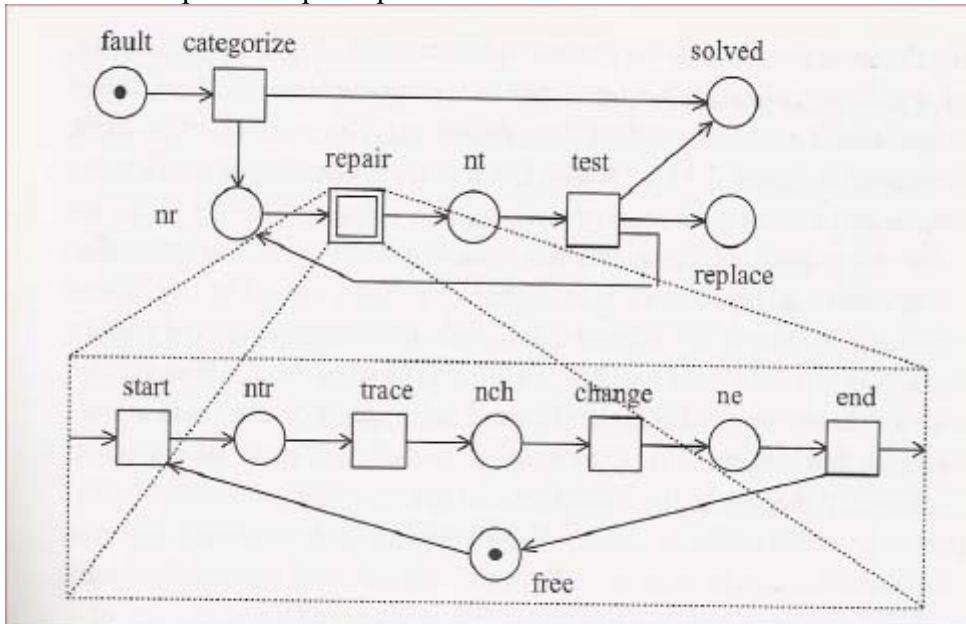
Place e transition possono formare 4 costruzioni di base:

- *sequenza*: i compiti vanno svolti uno dopo l'altro;



- *parallelo*: le attività vanno svolte tutte, ma il loro ordine non è importante;

quanto essi si trovano nel processo principale.



È possibile che all'interno di un processo ci siano punti in cui la sequenza delle attività non sia lineare nel tempo, o si debba verificare qualcosa affinché tale sequenza avvenga: queste situazioni sono indicate nel workflow da elementi detti **trigger**. Ve ne sono di 3 tipi:

- necessità di una risorsa:



- elemento proveniente dall'esterno dell'organizzazione:



- evento che dipende dal tempo:

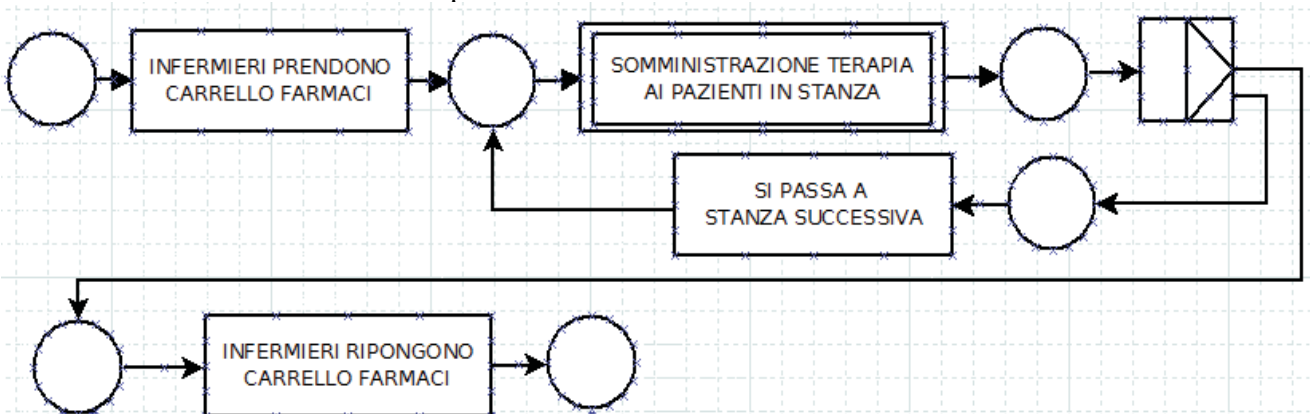


Nei workflow è possibile indicare gli attori o mettere in evidenza attività importanti, per esempio evidenziandole.

È importante che il workflow sia coerente al synopsis diagram corrispondente!

1 synopsis → 1 workflow

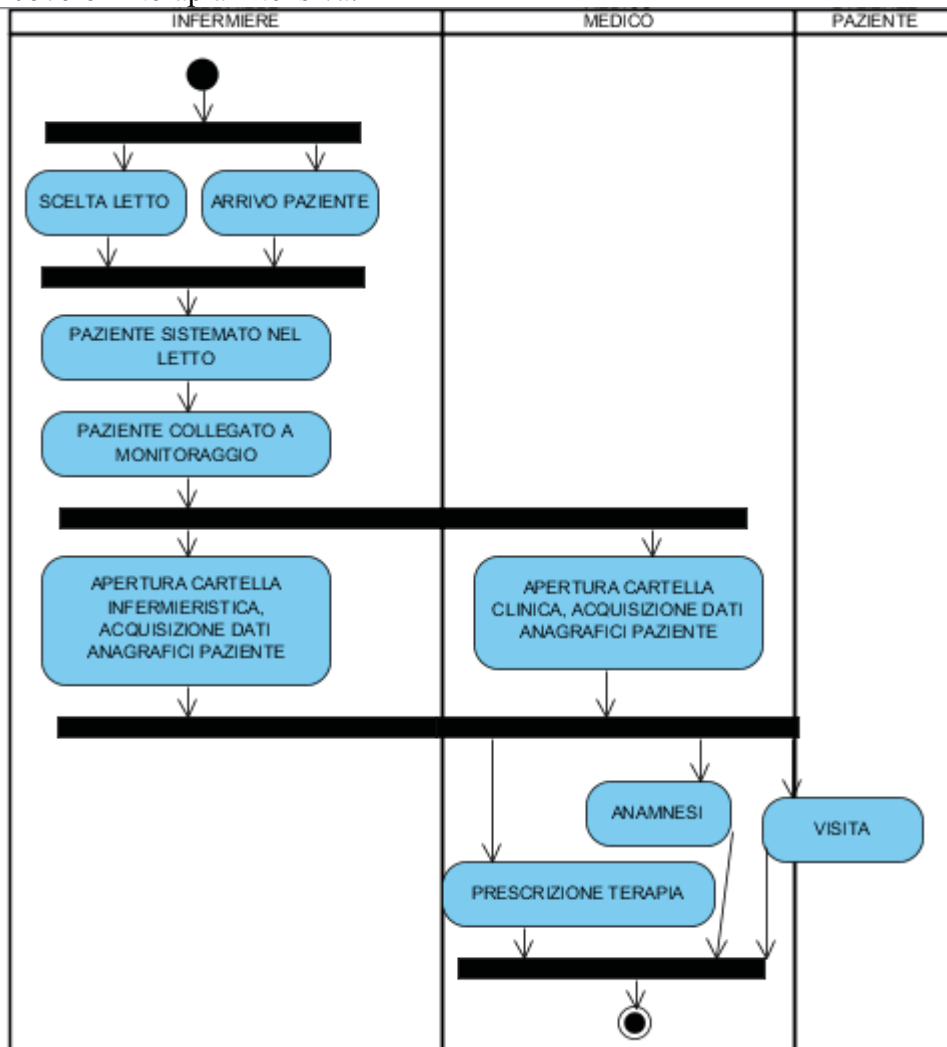
ESEMPIO: somministrazione terapia.



Anche in questo diagramma si possono indicare attività dipendenti dal tempo tramite il simbolo dell'orologio. Inoltre, si possono inserire note per aggiungere dettagli. Se a un'azione partecipa più di un attore, il rettangolo può essere messo a cavallo delle corsie.

1 workflow → 1 swim lane

ESEMPIO: ricovero in terapia intensiva.



ORGANIGRAMMA (organizational chart)

Si tratta di una struttura a forma di albero che illustra la relazione gerarchica tra enti. È utile quando si devono richiedere autorizzazioni.

STAKEHOLDER DIAGRAM

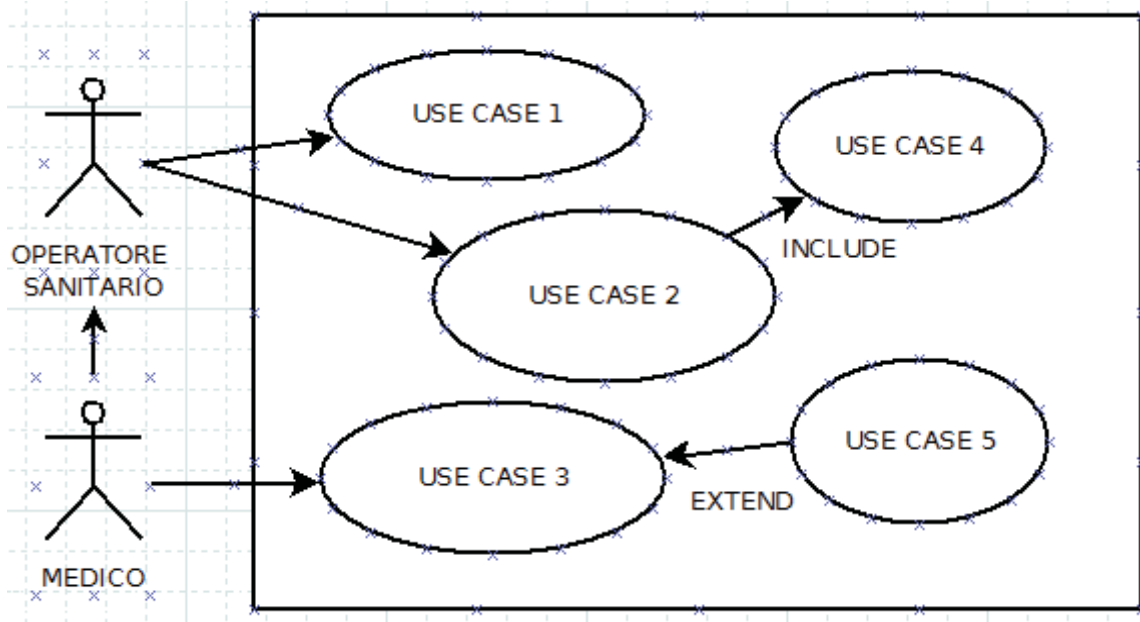
Mostra come le persone coinvolte nel processo siano strutturate gerarchicamente.

ANALISI DELLE SPECIFICHE

Serve a sviluppare o a modificare un software, più raramente per la reverse engineering (→ dato un sistema esistente, se ne ricavano le funzionalità).

Partendo dall'analisi delle specifiche, lo sviluppo di un software è diviso in tre momenti: **progettazione, costruzione, testing.**

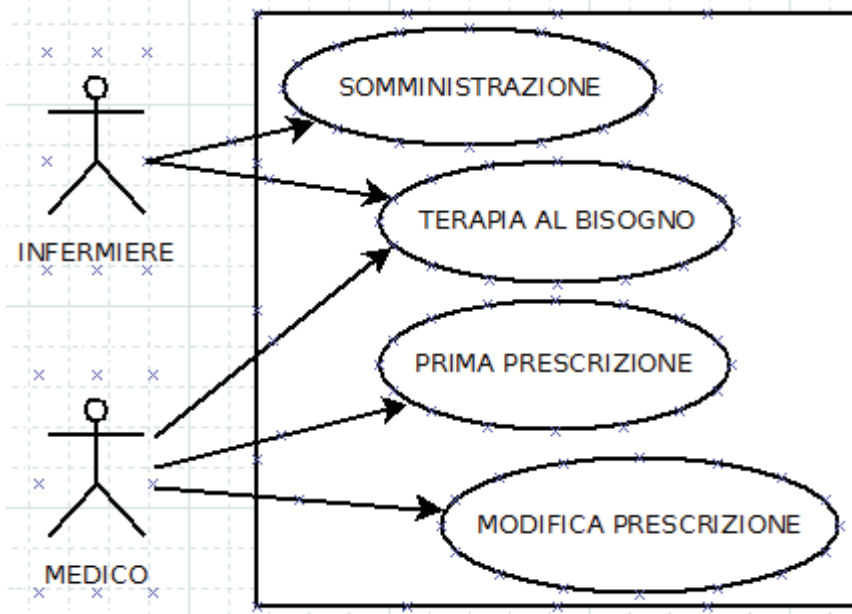
All'inizio la programmazione era svolta in modo procedurale: si avevano una struttura dati e delle istruzioni. In questa modalità la manutenzione del software era complicata, soprattutto per il fatto



Esistono due relazioni tra casi d'uso:

- include: una funzionalità appartiene anche ad altri casi d'uso. La notazione è una freccia dallo use case che include a quello che è incluso;
- extend: un caso d'uso estende quello di partenza, cioè il comportamento dello use case estensione può essere inserito nello use case esteso in certe condizioni. La notazione è una freccia dallo use case estensione allo use case esteso.

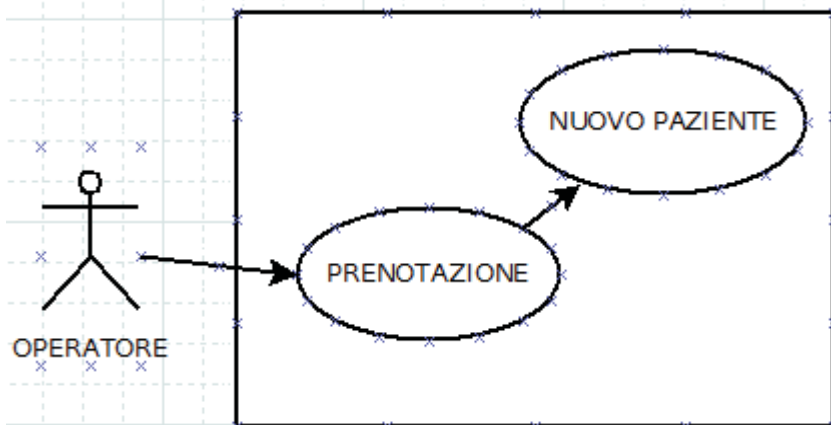
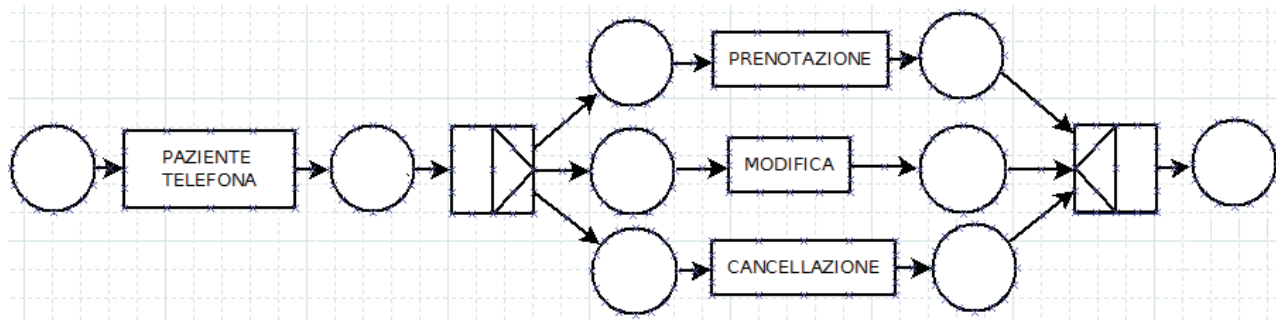
ESEMPIO: gestione della terapia.



USE CASE DETAILS

Tabella contenente:

- *use case name* → identificazione del caso d'uso;
- *goal* → obiettivo da raggiungere;
- *actors* → lista degli utenti abilitati a usare il caso d'uso;
- *triggers* → casi d'uso che mandano in esecuzione quello preso in considerazione (per esempio: include);



TITOLO	Prenotazione Visita	
GOAL	Gestire le prenotazioni	
ATTORI	Operatore Sanitario	
TRIGGERS		
MAIN FLOW	UTENTE	SISTEMA
		Visualizzazione form PV 1
	Clicca su casella vuota (*1)	
		Visualizzazione form PV 2a
	Inserisce dati paziente	
	Clicca RICERCA	
		Trova il paziente (*2)
		Visualizzazione form PV 2b
	Clicca CONFERMA (*3)	
	Memorizza prenotazione	
	Visualizza form PV 1 aggiornata	

Form PV 1:

*3: – utente clicca ANNULLA

Attenzione! È importante che ci sia l'opzione ANNULLA!

P. A. 1	UTENTE	SISTEMA
		Visualizza form PV 1
	Clicca casella occupata	
		Visualizza form PV 3
	Clicca CANCELLA (**1)	
		Chiede conferma cancellazione
	Clicca CONFERMA (**2)	
		Cancella prenotazione
	Visualizza form PV 1 aggiornata	

Anche nel percorso alternativo ci sono altri percorsi alternativi:

**1: - utente clicca MODIFICA
 - utente clicca ANNULLA

**2: - utente clicca ANNULLA

P. A. 2	UTENTE	SISTEMA
		Visualizza forma PV 1
	Clicca HOME	
	Torna a form iniziale	

Form PV 3:

MARIO ROSSI
Nato il: 20-10-1975 A: Torino
Indirizzo: via Bla Bla 15
 ⋮
PRENOTAZIONE: 18-04.14 ore 18.30

CANCELLA
MODIFICA
ANNULLA

E così via.

CONTROL OBJECTS

Controllano il flusso delle attività: sono le istruzioni che portano avanti i casi d'uso (NON controllano che tutto vada bene!). Si usano gli **activity diagrams**, che specificano come il sistema raggiunge i suoi obiettivi: possono essere usati per descrivere il percorso principale e i percorsi

analogica, ma lo è se l'archiviazione è computerizzata.

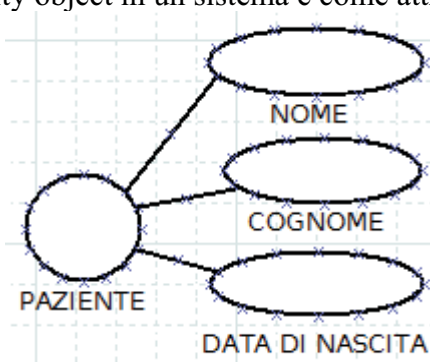
Un singolo dato è caratterizzato da 4 elementi:

- il *paziente* in questione;
- il *parametro* osservato;
- il *valore* del parametro;
- il *tempo* dell'osservazione.

Se il dato va inserito in una interfaccia, esso deve essere valido: ci devono essere sistemi di verifica.

Gli entity objects sono usati per modellizzare i dati che il sistema dovrà maneggiare. Gli entity objects usano attributi per memorizzare i dati: a ciascun entity object si possono legare molti attributi; gli attributi si sviluppano man mano che i casi d'uso vengono analizzati.

A volte non è facile decidere se un dato debba essere modellizzato come entity object o come attributo: per decidere, bisogna vedere come verranno usati i dati. Una serie di dati che vengono usati separatamente vanno modellizzati come entity objects, mentre una serie di dati che sono fortemente collegati ad altri dati vanno modellizzati come attributi. Una certa serie di dati può essere modellizzata come un entity object in un sistema e come attributi in un altro.



Il **sistema di codificazione** trasforma le descrizioni di malattie, ferite o procedure mediche in codici numerici. Questo permette l'accesso a diagnosi e procedure da usare per cura, ricerca o istruzione. Gli usi più comuni dei codici medici includono:

- l'identificazione di sintomi che vanno valutati;
- la descrizione dei servizi offerti per i rimborsi;
- aiuto nelle funzioni amministrative;
- la comparazione di strutture e la pianificazione di nuovi servizi nelle aree meno servite.

I sistemi di codificazione possono essere *generali* o *specifici*. Ne esistono 3 principali, tutti generali:

- ICD → usato soprattutto per codificare patologie e prestazioni per il rimborso (usato nelle SDO);
- SNOMED → usato dagli operatori sanitari per descrivere una situazione clinica con un buon livello di dettaglio;
- UMLS → non è un sistema di codifica singola, ma un ambiente in cui convivono i sistemi di codifica esistenti.

Con i sistemi di codificazione si perde il dettaglio sul paziente, ma si acquisisce la capacità di individuare situazioni simili.

Bisogna capire a che livello usare un sistema di codifica oppure il linguaggio naturale: i due non si escludono a vicenda e in certi casi possono essere mescolati.

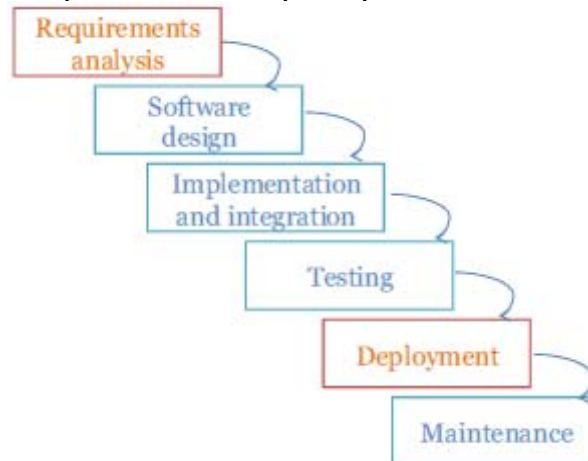
I sistemi di codifica devono rispondere a 2 necessità discordanti: la necessità di un sistema di codifica abbastanza generale da coprire molti pazienti diversi e la necessità di termini precisi e unici che si applichino accuratamente a uno specifico paziente.

UMLS: è un ambiente molto complesso. È composto da un *metathesaurus*, che contiene le altre codifiche, e da una *rete semantica* di collegamenti che permette di associare concetti e codifiche nel metathesaurus. Con l'UMLS si possono codificare testi scritti in linguaggio naturale.

CONSTRUCTION PROCESS

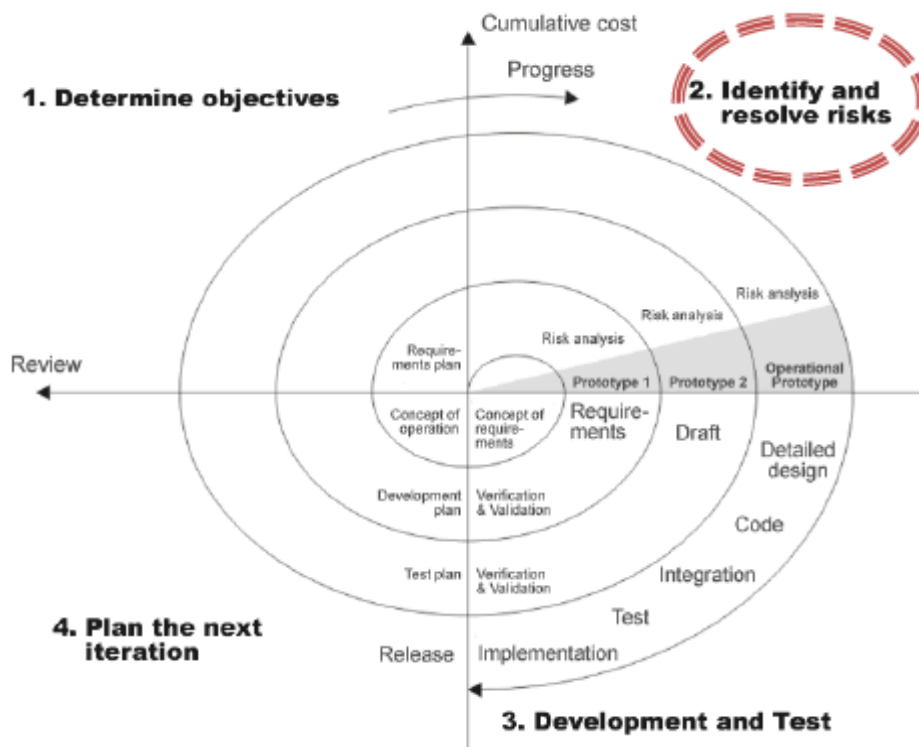
Bisogna documentare lo sviluppo del software: i possibili modelli sono diversi e per norma bisogna specificare quale viene usato.

Il **waterfall model** parte dall'analisi dei requisiti e arriva fino alla fine del processo attraverso step, ciascuno dei quali è ritenuto la prosecuzione di quello precedente.



Il problema del waterfall model è che se avviene un cambiamento nei requisiti, l'impatto può essere molto forte.

Lo **spiral model** parte dai requisiti. A ogni giro della spirale si amplia il prototipo. Si ritrovano le stesse fasi del waterfall, ma non definite nello stesso modo lineare. La spirale è suddivisa in 4 quadranti: tra questi, ve ne è uno dedicato all'identificazione dei rischi. La validazione è svolta contemporaneamente alla fase di sviluppo.



Il **V-model** è simile al waterfall, ma mette in evidenza le fasi di test. In particolare mostra come le varie fasi forniscano le informazioni per pianificare i test.

- *package* (simbolo ~): l'elemento è accessibile solo alle procedure di classi appartenenti allo stesso package (sottogruppo) di quella di partenza;
- *private* (simbolo -): l'elemento è accessibile solo alle procedure della classe di appartenenza.

La visibilità è determinata dagli sviluppatori del software.

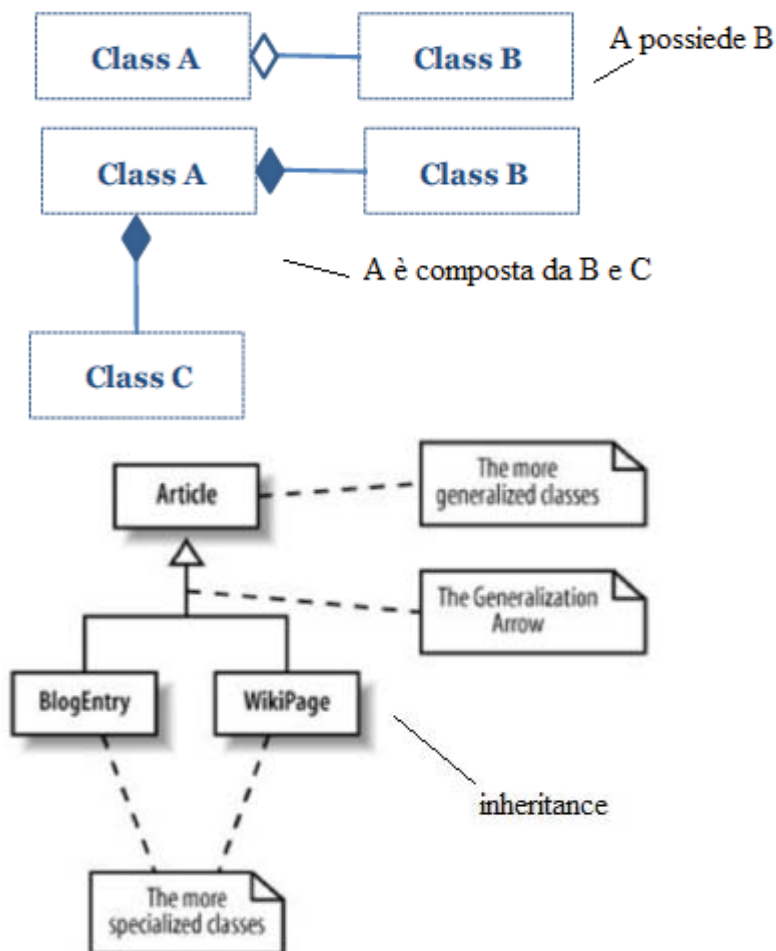
Le classi possono lavorare insieme usando 5 diversi tipi di relazioni:

- *dipendenza*: gli oggetti di una classe lavorano brevemente con quelli di un'altra classe;
- *associazione*: gli oggetti di una classe lavorano con quelli di un'altra classe per un periodo di tempo prolungato;
- *aggregazione*: le due classi condividono degli elementi, ma una possiede l'altra;
- *composizione*: una classe contiene oggetti di un'altra classe;
- *inheritance*: una classe è un tipo particolare di un'altra classe.

Simboli:



Alcuni esempi:



utilizzano;

- i DBMS garantiscono l'affidabilità della base dati: esistono meccanismi automatici di salvataggio (*backup*) o ripristino (*recovery*) dei dati. Il *mirroring* è un backup istantaneo, ottenuto tramite due basi dati fisicamente distinte: ogni modifica avviene contemporaneamente su entrambe;
- i DBMS garantiscono la privatezza dei dati: ciascun utente viene abilitato a vedere solo i dati di sua competenza e a svolgere solo determinate azioni su di essi. Questo è ottenuto tramite un meccanismo di autorizzazioni.

Un *modello dei dati* è un insieme di concetti utilizzati per organizzare i dati di interesse e descriverne la struttura in modo che essa risulti comprensibile a un elaboratore. Ne esistono 4:

- **modello gerarchico**: i files del data base dipendono l'uno dall'altro in modo gerarchico. È stato il primo modello utilizzato, ma crea difficoltà nell'andare a rileggere i dati;
- **modello reticolare**: ha un'applicazione molto ristretta;
- **modello relazionale**: è il modello più utilizzato. I files sono collegati dalle relazioni necessarie e la scrittura e la riletture dei dati sono più semplici. Una relazione viene solitamente rappresentata da una tabella in cui le righe corrispondono ai records e le colonne ai campi dei records;
- **modello a oggetti**: in pratica, quelle che nel modello relazionale erano tabelle in questo modello diventano oggetti.

I modelli appena descritti sono disponibili su DBMS commerciali; vengono detti *modelli logici* per sottolineare il fatto che le strutture utilizzate, pur essendo astratte, riflettono una particolare organizzazione:

alberi → gerarchico
 (grafi → reticolare)
 tabelle → relazionale
 oggetti → a oggetti

I *modelli concettuali* sono quelli che vengono utilizzati per descrivere i dati in modo indipendente dal modello logico. Essi tendono a descrivere i concetti del sistema reale, piuttosto che i dati utili a rappresentarli. Il modello concettuale più diffuso è il **diagramma entità-relazione (E-R)**.

Il diagramma E-R consente di descrivere l'organizzazione dei dati contenuti nel data base. I costrutti su cui si basa sono:

- entità: consente di rappresentare gli oggetti che hanno proprietà comuni;



- relazioni: rappresentano i legami logici tra 2 o più entità;



- attributi: descrivono le proprietà elementari di una entità o di una relazione. Possono essere semplici o composti.



Per esempio:

...
-----	-----	-----

Relazione

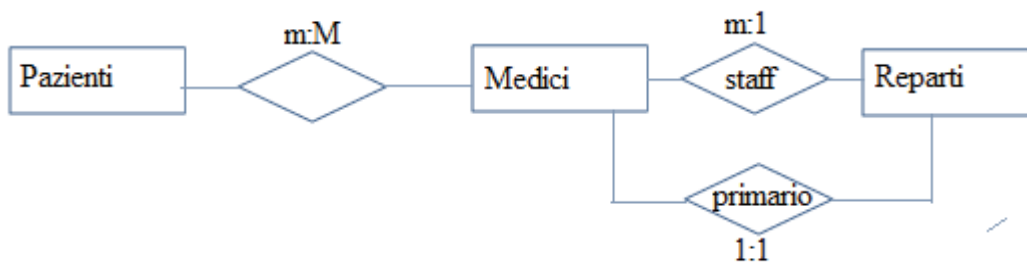
Matricola	Codice	ID
...

Il primo esempio di modellizzazione è migliore dal punto di vista della velocità. D'altro canto, non ha senso creare una relazione con troppi attributi. In definitiva, se alla relazione devono essere collegati pochi attributi le due modellizzazioni sono equivalenti; se alla relazione devono essere collegati molti attributi, è meglio trasformarla in un'entità come nel secondo esempio.

Dunque, le entità vengono rappresentate mediante tabelle che contengono un codice e una serie di attributi. Le relazioni vengono rappresentate mediante tabelle con gli identificativi delle entità, ma possono avere anche degli attributi.

La **cardinalità delle relazioni** indica quante volte una relazione tra entità mette in corrispondenza un record della prima tabella con uno della seconda tabella. La corrispondenza può essere 1:1, 1:M (uno a molti), m:1 (molti a uno) o m:M (molti a molti).

Per esempio:



Una **chiave** è un attributo che identifica univocamente il record. Significa che nella tabella, nel campo dell'attributo chiave, in ogni riga si trova un valore diverso, per esempio un numero progressivo. Si può anche usare un attributo proprio del record, del quale si è sicuri dell'univocità: per esempio, il numero di matricola. In ambito medico si usava come chiave il codice fiscale, ma presentava due problemi: in certe situazioni poteva non essere fisicamente disponibile (per esempio al pronto soccorso); inoltre i dati relativi al paziente, per problemi di privacy, non devono essere riconoscibili.

Una chiave può anche essere creata unendo due attributi: l'importante è che il codice risultante sia univoco.



ESEMPIO: gestione computerizzata della terapia.

Oggetti entità (UML): si individuano nella fase di analisi dei requisiti.

Lo schema è un prototipo che viene implementato. I dati reali vengono memorizzati secondo lo schema di prototipo. Per esempio due ospedali diversi potrebbero avere lo stesso schema, ma istanze diverse (i dati sono diversi).

Lo schema *interno* riguarda come i dati vengono distribuiti nei files. Si ha anche uno schema *esterno*, in quanto le tabelle possono essere riorganizzate se due applicazioni distinte le usano in modo diverso. Per esempio la gestione della terapia rispetto al paziente e la gestione della terapia rispetto alla farmacia fanno capo allo stesso data base, ma non è detto che una sola organizzazione di tabelle vada bene per entrambe. Per questo i DBMS consentono di avere uno schema esterno relativo a una singola applicazione.

In totale ci sono 3 tipi di schema:

- *logico*: descrive il DB per mezzo di un modello logico;
- *interno*: rappresenta lo schema logico per mezzo di strutture fisiche di memorizzazione;
- *esterno*: descrive una porzione di DB per mezzo del modello logico per rappresentare il punto di vista di un particolare insieme di utenti.

Per quanto riguarda i linguaggi, nella fase di creazione del data base si fa riferimento a DDL, che permette di descrivere le tabelle. In seguito DML permette di modificare le tabelle.

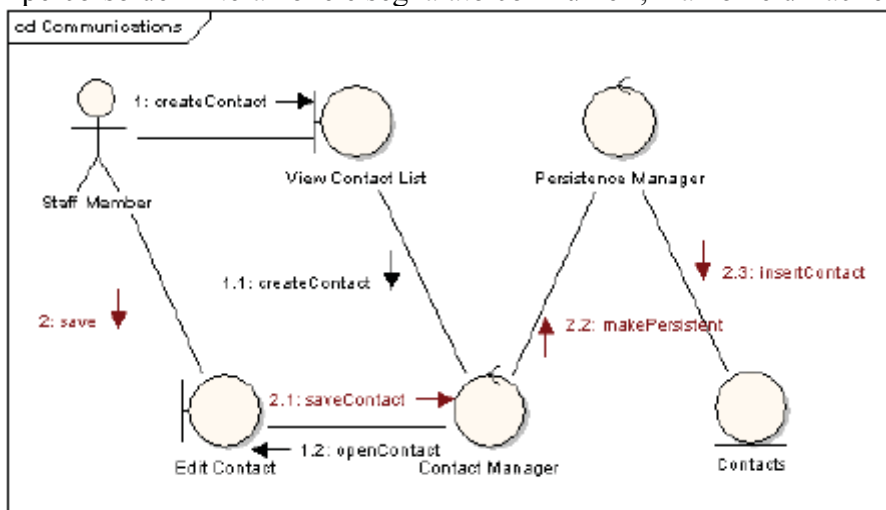
Il linguaggio più usato è SQL. La maggior differenza tra ambienti è la modalità di accesso ai dati.

Normalmente c'è un *utente amministratore* che crea e modifica le tabelle. Gli utenti *progettisti e programmatori* implementano le relazioni. Gli utenti veri e propri utilizzano il data base tramite applicazioni e si distinguono in utenti *finali* (per es il medico) e utenti *casuali* (per es un tecnico).

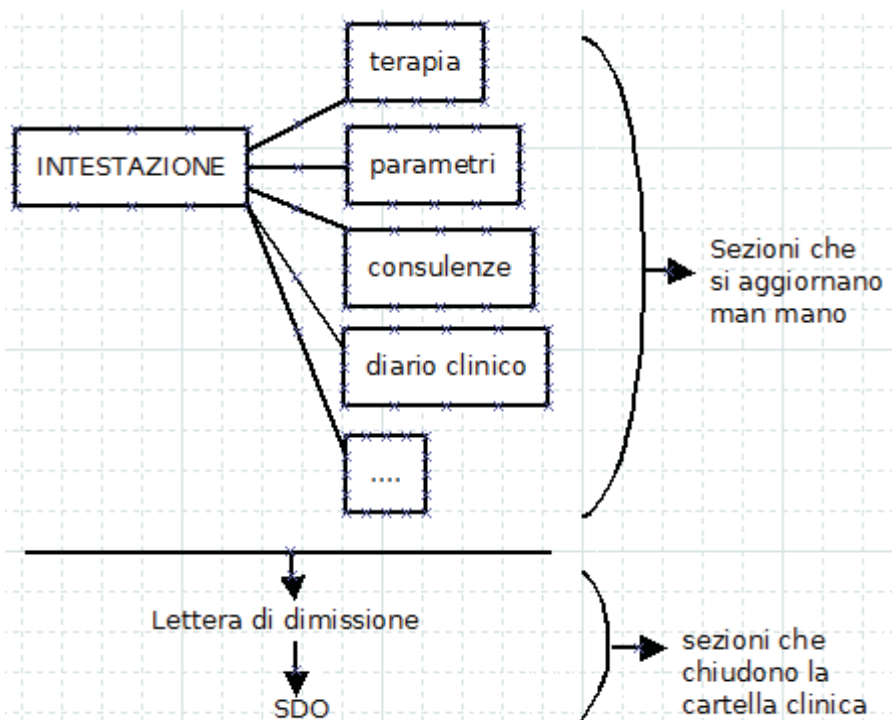
Un *data warehouse* è un tipo di data base con uno scopo diverso. Un data base, infatti, è un'organizzazione di dati specifica per applicazioni che svolgono varie attività. Un data warehouse, invece, contiene dati organizzati per essere usati in ricerche o statistiche. L'organizzazione dei dati è diversa.

La descrizione della parte procedurale del software avviene mediante diversi diagrammi.

Nel **communication diagram** vengono rappresentati gli oggetti, come sono connessi tra loro e come interagisce l'utente. L'utente comunica con l'interfaccia, che è collegata ad oggetti controllo. Questi ultimi possono essere collegati tra loro, ma interagiscono anche con i dati e cioè con gli oggetti entità. Il percorso dell'interazione è segnalato con numeri, ma non è di facile interpretazione.

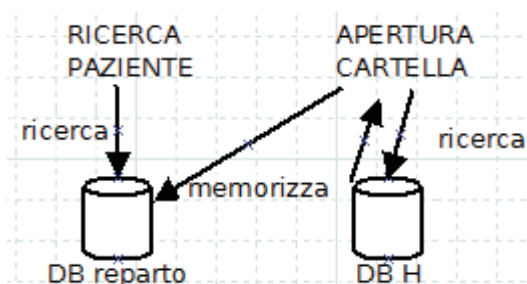


Anche il **sequence diagram** permette di inserire gli oggetti e gli attori. È composto da una linea, che rappresenta il tempo, e da rettangoli, che rappresentano i range temporali in cui un dato elemento è coinvolto. Delle frecce mostrano come si collegano gli elementi. Con questo diagramma



Aprire la cartella clinica significa creare l'intestazione: ci si collega al DB dell'ospedale, si copiano i dati del paziente e si crea l'anagrafica.

Cercare un paziente è diverso dall'aprire la cartella, in quanto il paziente viene cercato nel DB locale.



Il **referto** ha una serie di parti. Contiene la descrizione di ciò che è avvenuto (esame, visita) e riferimenti ad avvenimenti precedenti. Vi è inserita una descrizione e si basa su dati (giudizi, misure, valutazioni). Contiene anche un commento e le conclusioni che si possono trarre. Viene firmato dal medico.

Lo **use case diagram** è un solo diagramma, NON un diagramma per ciascun caso d'uso! Li contiene tutti. Se la situazione da descrivere è molto complessa, al più viene suddiviso in package. Deve esserci un rettangolo intero con all'esterno gli attori (a destra o a sinistra). Gli attori si ottengono dal synopsis: se per caso uno non si ritrova nello use case diagram bisogna specificare il perché in una nota (per esempio: il paziente non compare perché non interagisce con il software).

I **casì d'uso** vengono derivati dal workflow: bisogna capire quali attività comportano un caso d'uso e quali no (per esempio sistemazione paziente nel letto non lo comporta, la somministrazione della terapia sì).

Non ha senso che ci siano un workflow e uno swim lane diagram uguali: ci possono essere processi semplici per i quali non è necessario lo swim lane. Se si fa lo swim lane, esso deve essere più dettagliato del workflow.

Gli **use case details** si ricavano dagli swim lane. Non è detto che i due siano identici perché possono esserci attività che non ha senso inserire nel software.

vertiginosamente, richiedendo sempre di più lo sviluppo di procedure efficaci ed efficienti per garantirne la disponibilità, oltre che l'uso sicuro.

Le figure professionali che fanno parte del SIC sono l'ingegnere clinico (IT) e il tecnico delle apparecchiature biomediche (BMET). L'ingegnere clinico è definito come una persona che, in modo sistematico, faccia in modo che le tecnologie più efficaci, sicure, appropriate ed economiche siano disponibili in risposta alle necessità del paziente. Il tecnico delle apparecchiature biomediche è responsabile di collaudi e riparazioni dell'equipaggiamento.

Le attività del SIC comprendono:

- coordinamento e organizzazione;
- partecipazione al processo di programmazione degli investimenti tecnologici;
- sicurezza di esercizio delle apparecchiature biomediche in uso;
- continuità di esercizio delle apparecchiature biomediche individuate come critiche;
- gestione dell'inventario tecnologico;
- manutenzione delle attrezzature sanitarie;
- acquisizione.

La richiesta di nuova tecnologia viene dal personale sanitario, mentre l'obsolescenza viene indicata dagli ingegneri clinici.

Il SIC partecipa alla gestione dell'*inventario tecnologico*, un data base separato dall'*inventario patrimoniale*. Ogni dispositivo, strumento o pezzo di mobilio (ogni oggetto non consumabile) viene catalogato e inserito nell'archivio patrimoniale insieme ad alcuni dati (descrizione, luogo in cui si trova, ecc): serve dal punto di vista contabile e organizzativo, ma non è funzionale alla gestione delle tecnologie in quanto non c'è attenzione alla manutenzione. Per questo c'è bisogno di un archivio parallelo, nel quale ogni strumento ha un codice; qualunque intervento o spostamento relativo a una tecnologia viene segnalato nell'inventario tecnologico.

Quali sono i problemi che possono essere collegati alla gestione di un software?

Il software ha bisogno di un'infrastruttura. Gli apparecchi, infatti, sono collegati all'impianto elettrico, che deve essere a norma rispetto allo strumento. È necessaria anche un'infrastruttura di rete, gestita dall'area informatica (IT). Il software gira su calcolatori: in certi casi si trovano nell'hardware del dispositivo stesso, in altri casi si trovano su computer collegati. In passato software e computer venivano gestiti in modo separato, ma oggi non si può più fare così: c'è bisogno di collaborazione tra SIC e IT, ma spesso non è facile. Per i software medicali il problema è ancora più grande perché non è chiaro quando sono dispositivi medici: non è un problema solo di definizione, ma di gestione (il dispositivo medico ha bisogno di collaudi e controlli diversi). I software stand alone vanno incontro a una fase di adattamento dopo l'acquisto. Inoltre, di solito sono piattaforme di cui si possono comprare moduli diversi. Bisogna stare attenti alla manutenzione e ai possibili malfunzionamenti.

STANDARD

La normativa si divide in:

- *legge* → deve essere seguita;
- *norme* → proposte che supportano fabbricanti e utenti per seguire la legge.

Uno standard consiste in una serie di specifiche accettabili da fabbricanti e utenti che vengono definite e aggiornate. Per esempio, l'UML e i sistemi di codifica sono standards: non sono obbligatori, sono scelte di utilizzo. Lo standard spesso è un riferimento.

In inglese la parola "standard" si riferisce sia alla legge che alle norme: per distinguerli, si parla di standard *de iure* (→ legge) e di standard *de facto* (→ norme).

I benefici prodotti dagli standard sono quelli di migliorare l'interoperabilità, ridurre la duplicazione e rendere possibile la competitività sul mercato.

I principali enti che producono standard a livello internazionale sono ISO (International

- test del software.

Infine si è validato il set di indicatori.

Gli indicatori clinici permettono di:

- monitorare l'evoluzione di una patologia;
- comparare pazienti diversi;
- misurare e monitorare la qualità delle cure mediche.

Per esempio, il software Prisma è un software per la gestione informatizzata della terapia farmacologica. Il suo obiettivo è quello di evitare gli errori di somministrazione e prescrizione dei farmaci e di semplificare la complessità di alcuni processi clinici.

Un **errore** è un fallimento nella pianificazione e/o nell'esecuzione di una sequenza di azioni che determina il mancato raggiungimento, non attribuibile al caso, dell'obiettivo desiderato. Un errore può trasformarsi in un evento avverso.

Un **evento avverso** da farmaco è un qualsiasi evento indesiderato che si verifica durante una terapia farmacologica per effetto dell'uso o del non uso di un farmaco, ma non strettamente correlato all'azione del farmaco stesso.

Gli errori più frequenti nella gestione della terapia sono:

- errore di prescrizione;
- errore di trascrizione/interpretazione;
- errore di etichettatura/confezionamento;
- errore di allestimento/preparazione;
- errore di distribuzione;
- errore di somministrazione.

Gli strumenti usati per modellizzare i processi clinici sono stati synopsis diagram, wrkflow diagram, swimlane diagram. I processi analizzati sono stati:

- nuovo ricovero;
- giro delle visite;
- trascrizione schede di terapia;
- somministrazione terapia;
- preparazione carrello farmaci;
- preparazione della terapia parenterale.

Cosa cambia con l'introduzione di un software per la gestione della terapia informatizzata?

Attività svolte dai medici:

ATTIVITÀ	SENZA SW	CON SW
Ricovero	Scrive la terapia sul diario clinico, la trascrive sulla scheda di terapia e la consegna agli infermieri	Inserisce la terapia in Prisma. Viene generato automaticamente il foglio di terapia a cui gli infermieri possono accedere subito da qualsiasi pc
Giro delle visite	Appunta le modifiche da apportare alla terapia in agenda, al termine del giro trascrive le modifiche sul diario clinico e	Il medico modifica la terapia accedendo a Prisma tramite il totem su cui è installato un pc, che porta con sé in reparto

	Numero di attività che costituiscono un processo clinico	quantitativo
Riduzione dei tempi legati alle trascrizioni multiple e impiegati per la preparazione delle etichette per la terapia parentale	Durata delle attività organizzative/gestionali svolte nell'arco di una giornata dagli infermieri del reparto	quantitativo
Necessità di una quantità minore di documentazione cartacea per lo svolgimento dei processi clinici	Numero di documenti necessari per lo svolgimento di un processo clinico	quantitativo
	Numero di documenti presenti in reparto in relazione al numero di pazienti ricoverati	quantitativo
Diminuzione dei tempi di accesso ai dati clinici relativi alla terapia	Tempo di accesso ad un dato clinico relativo alla terapia	quantitativo
Ottimizzazione della gestione delle scorte	Percentuale di scorte di farmaci presenti in reparto	quantitativo
	Numero di informazioni ridondanti (duplicazione dei dati clinici)	quantitativo
Aumento della disponibilità dei dati clinici relativi alla terapia		<i>qualitativo</i>
Altro	Controllo sulle autorizzazioni e gli accessi al sistema	<i>qualitativo</i>
	Accessi differenti in funzione dei diversi tipi di utenti	<i>qualitativo</i>
	Tracciabilità dei dati clinici e del personale sanitario che esegue un'attività	<i>qualitativo</i>
	Inserimento di un paziente con i propri dati anagrafici all'interno di un database	<i>qualitativo</i>
	Aggiornamento automatico dei farmaci presenti nel prontuario	<i>qualitativo</i>
	Archiviazione dei dati clinici	<i>qualitativo</i>
	Aggiornamento dei dati clinici	<i>qualitativo</i>

Indicatori: gestione del rischio clinico:

Cosa ci si aspetta con l'informatizzazione della terapia?	Indicatori	Tipo
------------------------------------------------------------------	-------------------	-------------

Interfaccia ergonomica ed uniforme del SW	
Interfaccia del SW user friendly	
Continuità dei processi supportati dal SW	
Soddisfazione dei diversi utenti verso il sistema informatizzato	
Il contributo del SW per i successi ospedalieri	
Il contributo del SW per gli obiettivi dei medici e degli infermieri	
Il contributo del SW per l'assistenza del paziente	
Compatibilità del sistema con le esigenze dei diversi utenti	

Affinché il SW possa essere considerato come un SW di qualità, dovrà rispondere a degli specifici requisiti di:

- funzionalità;
- sicurezza;
- affidabilità;
- usabilità;
- efficienza;
- facilità d'uso.

In particolare, la facilità d'uso determina l'atteggiamento nei confronti della tecnologia da parte degli utenti e il suo grado di utilizzo.

Cosa ci si aspetta con l'informatizzazione della terapia? La diminuzione degli eventi avversi da farmaco, difficilmente rilevabili a posteriori.

Gli indicatori relativi alla sicurezza del paziente sono:

- degenza media per i pazienti ricoverati;
- numero di complicanze ospedaliere;
- numero di riospedalizzazioni (nell'arco massimo di 2 settimane);
- numero di eventi sentinella;
- numero di contenziosi.

Come ogni dispositivo medico, il SW deve essere periodicamente sottoposto a procedure di manutenzione preventiva per garantire la conformità ai requisiti. L'attività mantenitiva ha l'obiettivo di garantire nel tempo il mantenimento delle caratteristiche iniziali, il mantenimento della sicurezza informatica, il mantenimento in efficienza e l'aggiornamento dell'infrastruttura hardware e SW. Inoltre, nel momento in cui si presenta un evento avverso durante l'utilizzo del SW dovrà essere eseguita la manutenzione correttiva.

Indicatori	Tipo
Numero di chiamate dovute ad un malfunzionamento o guasto del SW	quantitativo
Dopo quanto tempo da una segnalazione arriva il supporto tecnico in reparto	

efficiente nel rispondere alle richieste degli utilizzatori?

- Qualità e appropriatezza del sistema → quanto il sistema soddisfa le aspettative degli utilizzatori?

Anche in questo caso si sono individuati numerosi indicatori relativi a diversi contesti.

Gli indicatori individuati si sono dimostrati utili al fine della valutazione. Gli indicatori misurati hanno permesso di dimostrare il miglioramento ottenuto in termini di:

- riduzione della complessità dei processi e del tempo richiesto per lo svolgimento;
- ottimizzazione delle risorse.

DISPOSITIVO MEDICO

Definizione della direttiva dispositivi medici 93-42:

qualsiasi strumento, apparato, applicazione, software, materiale o altro articolo, usato da solo o in combinazione, insieme a qualsiasi accessorio, incluso il software da essere usato specificatamente (nelle intenzioni del costruttore) per scopi diagnostici e/o terapeutici necessario per il loro corretto utilizzo, pensati dal costruttore per essere usati sugli esseri umani allo scopo di:

- diagnosi, prevenzione, monitoraggio, trattamento, alleviazione sintomi di malattie;
- diagnosi, monitoraggio, trattamento, alleviazione di o compensazione per una ferita o un handicap;
- investigazione, sostituzione o modifica dell'anatomia o di un processo fisiologico;

e che non raggiunge il suo scopo principale nel o sul corpo umano per mezzi farmacologici, immunologici o metabolici, ma che può essere assistito nella sua funzione da essi.

HTA

L'HTA (Health Technology Assessment) nasce negli anni '60 dalla necessità di avere dati sulle possibili innovazioni. Può essere micro o macro, a seconda che sia fatto a livello locale o governativo. L'HTA è uno strumento interdisciplinare (cioè portato avanti da team di persone con competenze diverse) e tiene conto di aspetti molto eterogenei (tecnici, sociali, etici, organizzativi). L'HTA è portata avanti tramite comitati o agenzie, a seconda che sia micro o macro: la risposta è un report in cui si descrive cosa fa la tecnologia, quali sono i suoi vantaggi e svantaggi. La decisione finale sull'adozione della tecnologia viene presa da chi ha richiesto l'analisi. L'HTA è un'**analisi strutturata**, cioè segue un iter procedurale con 8 step:

1. richiesta esterna di valutare una tecnologia;
2. scelta domanda di assessment;
3. scelta del team in funzione della tecnologia sotto esame;
4. indagini per ottenere le informazioni volute;
5. mettere insieme i dati;
6. compilare il report;
7. rivedere report;
8. divulgare report.

L'analisi evidenzia diversi scenari: situazione presente (senza tecnologia) vs possibili situazioni future (con tecnologia). Il concetto di tecnologia è molto esteso: è tutto ciò che riguarda l'applicazione di conoscenza (procedure, farmaci, ecc).

POSSIBILI DOMANDE

- differenza sistema-modello?
- Metodi di modellizzazione processi?
- UML?

INFORMATICA MEDICA: è l'applicazione della tecnologia dei computer a tutti i campi della medicina
(Collen, MEDINFO, 1977)

E-HEALTH: applicazione delle moderne ICT (Information and Communication Technologies) in supporto della salute e dei campi ad essa correlati. Consente la prestazione sanitaria ovunque

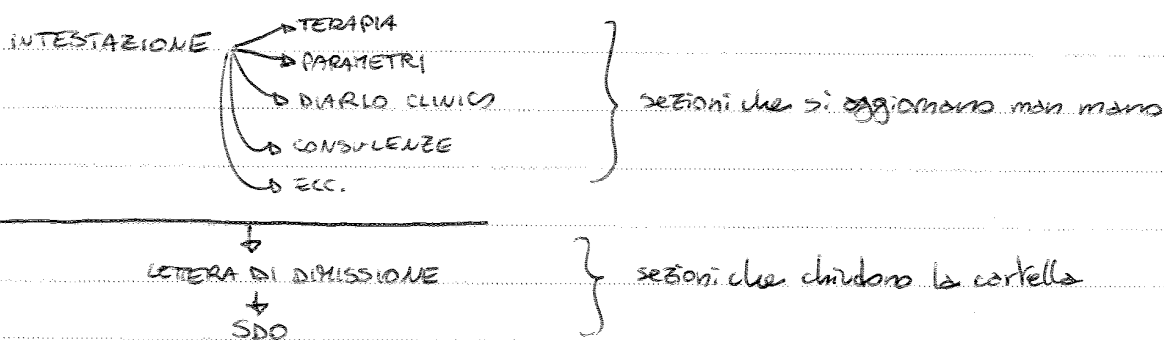
Applicazioni eHealth in base a scopo:

- SW collegati a dispositivo medico
- SW che gestiscono dati nell'organizzazione
- SW che scambiano dati tra organizzazioni diverse

Applicazioni eHealth in base a uso:

- SW per procedure di gestione
- SW per gestione con connessione indiretta in cura paziente
- SW per cura paziente (MEDICAL DEVICE SW)

EHR (Electronic Health Record): cartella clinica informatizzata. Integra i dati del paziente generati nel processo di cura e consente di visualizzarli a chi ne ha bisogno e ha autorizzazione. È un insieme di documenti:



Apertura cartella → estrazione dati paziente da DB ospedale → creazione intestazione
 Contiene l'anamnesi (storia clinica) → remota / recente
 SDO → scheda di dimissione

TELEMEDICINA: area dell'informatica medica che include tutte le applicazioni usate per lo scambio di dati tra varie strutture.

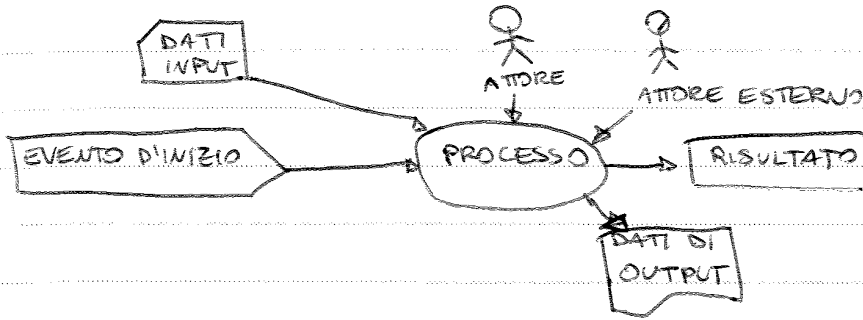
BIOINFORMATICA: utilizza computer e strumenti simili per aiutare con osservazioni biologiche di base, come il genoma o le sequenze di proteine, e dirigere misure di processi molecolari o cellulari. Applica le tecniche informatiche per comprendere e organizzare le informazioni associate alle molecole.

Studio di Bonsard (2007):

- genoma e proteoma non ancora inclusi in ambito medico;
- gli aspetti dell'info medica relativi all'organizzazione degli ospedali sono lontani dalla bioinformatica;
- bioinformatica e info medica si sovrappongono nell'usare gli stessi metodi matematici e informatici.

→ QUESTO GLIELLO POTEVO DIRE ANCHE IO!

• SYNOPSIS DIAGRAM:



• WORKFLOW DIAGRAM: si basa sul formalismo delle reti di Petri (place, transition)

- istruzioni di base: sequenza
- parallelo
- selezione
- ciclo

• sottoprocessi ⇒ ESTENSIONE GERARCHICA

↳ NO place a inizio e fine!

• TRIGGER: • necessita di una risorsa

- elemento esterno
- dipendenza dal tempo

• SYNOPSIS ⇒ WORKFLOW ⇒ devono essere coerenti

• FLOW CHARTS: descrivono una sequenza di attività, ma non c'è il parallelo. Alternativa di workflow.

• SWIM LANE ACTIVITY DIAGRAM: mostra sequenza di attività con chiara definizione ruoli.

• WORKFLOW ⇒ SWIM LANE ⇒ se swim lane ha maggior approfondimento.



• ORGANIGRAMMA (Organizational Chart): mostra relazione gerarchica tra enti.

• STAKE HOLDER DIAGRAM: mostra relazione gerarchica tra persone coinvolte nel processo.

ANALISI DELLE SPECIFICHE: serve a sviluppare o modificare sw, a volte anche per reverse engineering
 progettazione → costruzione → testing

programmazione ad oggetti: OOP (Object-Oriented Programming). Tutte le routine che usano una stessa struttura dati formano un oggetto (entità logica del sw).

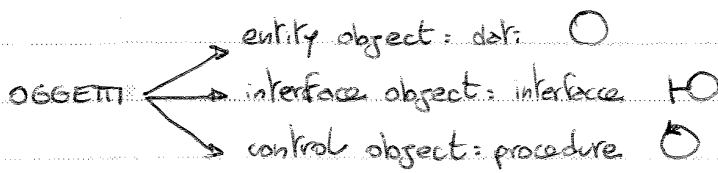
oggetto { struttura dati
 funzioni

La manutenzione è locale ⇒ sul singolo oggetto

prima: programmazione procedurale (struttura dati + istruzioni)

USE CASE DETAILS:

USE CASE NAME:	identificazione del caso d'uso
GOAL:	obiettivo da raggiungere
ACTORS:	lista degli utenti abilitati a usare il caso d'uso
TRIGGERS:	casi d'uso che mandano in esecuzione quello in esame
MAIN FLOW:	svolgimento del caso d'uso che avviene più frequentemente
ALTERNATIVE FLOWS:	evoluzioni diverse del caso d'uso
FORM:	interfacce



INTERFACCE: sono usate dagli utenti per comunicare con il sistema
 il design deve tener conto di:

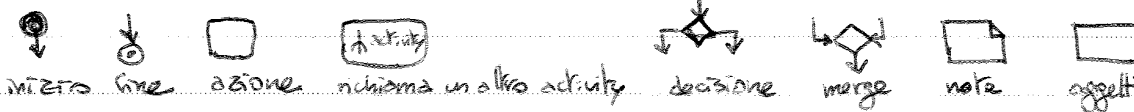
- usabilità

- senso comune
- analisi dei rischi

dal punto di vista dell'utente bisogna tener conto di:

- successione attività
- rischi
- valide
- possibilità di personalizzazione

CONTROL OBJECTS: sono le istruzioni che portano avanti i casi d'uso ⇒ controllano flusso attività
 per specificare come il sistema raggiunga i suoi obiettivi si usano activity diagrams



1 USE CASE DETAILS ⇒ 1 ACTIVITY DIAGRAM

ENTITY OBJECTS: sono usati per modellizzare i dati che il sistema dovrà gestire
 usano attributi per memorizzare i dati

dati usati separatamente ⇒ diversi entity object
 dati fortemente collegati ⇒ attributi



Dato: qualunque osservazione o misura che riguardi il paziente. Può essere un valore singolo o un insieme di valori (per es. un'immagine)

I dati sono usati per prendere decisioni. La decisione viene presa in base all'informazione, cioè all'applicazione della conoscenza ai dati

↳ a priori
 ↳ derivata dai dati

È basato su concetti e relazioni; ogni concetto è legato a descrizioni

concetto → ha un identificativo numerico univoco (concept ID)

↳ è rappresentabile da un unico nome in linguaggio naturale (Fully Specified Name)

↳ è formalmente definito in funzione delle sue relazioni con altri concetti

descrizioni → sono nomi e sinonimi

↳ hanno un identificativo numerico univoco (description ID)

↳ 1 solo concetto è collegato a tante description

relazioni → collegano concetti

↳ tipi: defining, qualifying, historical, additional

UMLS (Unified Medical Language System): è un ambiente composto da un set di files e SW.

È composto da: metathesaurus → contiene le altre codifiche

- rete semantica → permette di associare concetti e codifiche nel metathesaurus
- specialistico: lexicon & tools → strumenti di traduzione

CONSTRUCTION PROCESS: bisogna documentare lo sviluppo del SW.

Possibili modelli:

- WATERFALL MODEL: descrive il processo attraverso step, ciascuno dei quali è ritenuto la prosecuzione di quello precedente.

Problema: modifica nei requisiti → forte impatto

- SPIRAL MODEL: spirale suddivisa in 4 quadranti: (determinazione obiettivi, identificazione e risoluzione rischi, sviluppo e test, pianificazione nuova iterazione)

A ogni giro della spirale si amplia il prototipo.

Validazione svolta contemporaneamente a sviluppo

- V-MODEL: mostra come le varie fasi forniscano informazioni per pianificare i test

In UML: vari diagrammi mostrano aspetti diversi del SW:

- ASPETTO STATICO (o strutturale): struttura statica del sistema → class diagrams

- ASPETTO DINAMICO (o comportamentale): mostra collaborazione tra oggetti e cambiamenti: situazione interna agli oggetti → activity diagrams

CLASS DIAGRAMS: descrivono le classi e le loro relazioni

classe: insieme di dati e procedure. Descrivono i diversi tipi di oggetti del sistema.

I class diagrams sono forniti direttamente dall'ambiente di sviluppo.

attributi → descrivono stato oggetto

procedure → descrivono comportamento oggetto

Modello dei dati: insieme di concetti usati per organizzare i dati di interesse e descriverne la struttura in modo che risulti comprensibile a un elaboratore.

- modelli logici
- ▷ MODELLO GERARCHICO
 - ▷ MODELLO RETICOLARE (proposizionale) (applicazione ristretta)
 - ▷ MODELLO RELAZIONALE: relazione = tabella
 - ↳ righe = records
 - ↳ colonne = campi dei records
 - ▷ MODELLO A OGGETTI

Le strutture utilizzate (anche se astratte) riflettono una particolare organizzazione:

- gerarchico → alberi
- (reticolare → grafi)
- relazionale → tabelle
- a oggetti → oggetti

Modelli concettuali: descrivono i dati in modo indipendente dal modello logico. Descrivono i concetti del sistema reale piuttosto che i dati utili a rappresentarli.

Più usato: diagramma entità-relazione

- entità → rappresenta gli oggetti con proprietà comuni Entità
- relazione → legame logico tra 2 o più entità ◇
- attributi → descrivono proprietà elementari di entità e relazioni ∞
- ENTITÀ = tabelle con codice + attributi
- RELAZIONI = tabelle con ID entità + pochi attributi

cardinalità-relazioni: quante volte una relazione mette in corrispondenza un record della prima tabella con uno della seconda tabella.

1: 1 molti: 1 1: molti molti: molti

chiave: attributo che identifica univocamente il record

schema: descrizione del DB. Per ogni tabella descrive ogni attributo del record in termini di campo.

Può essere:

- LOGICO: descrive DB per mezzo di uno dei modelli logici

- INTERNO: rappresenta schema logico per mezzo di strutture fisiche di memorizzazione (per es. file binario)

- ESTERNO: (viste) descrizione di una porzione di DB tramite modello logico per rappresentare il punto di vista di un particolare insieme di utenti

istanza: tabella completa di dati.

La rappresentazione con schemi permette l'indipendenza dei dati:

- ↳ FISICA: si possono modificare strutture DB senza influire sui programmi che usano i dati
- ↳ LOGICA: si può interagire con schema esterno indipendentemente da schema logico

RISCHIO: La strumentazione biomedica presenta 3 tipi di rischio:

- intrinseco: legato al malfunzionamento dello strumento. Ridotto grazie a presenza calcolatori
- gestionale: bisogna fare continuamente manutenzione preventiva
- uso: dovuto a uso non corretto; si agisce su formazione del personale

INGEGNERIA CLINICA: HTM (Health Technology Management) o SIC (Servizi Ing. Clinica)

prima si occupava di sicurezza, adesso di tutti gli aspetti relativi a gestione tecnologia.

Figure professionali:

- ingegnere clinico (IT): persona che, in modo sistematico, fa in modo che le tecnologie più efficaci, sicure, appropriate ed economiche siano disponibili in risposta alle necessità del paziente
- tecnico delle apparecchiature biomediche (BMET): responsabile di collaudi e riparazioni dell'equipaggiamento

STANDARD: consiste in una serie di specifiche accettabili da fabbricanti e utenti che vengono definite e aggiornate.

standard de iure → legge → deve essere rispettata

standard de facto → norma → proposte per seguire la legge

Principali enti che producono standard:

- ISO (International Organisation for Standardisation)
- CEN (Comité Européen de Normalisation)
- HL7 (Health Level Seven)

HL7: è un'organizzazione no profit che fornisce framework (strutture logiche di supporto) e i relativi standard per scambiare, integrare, condividere e recuperare informazioni elettroniche relative alla salute che supportino pratica clinica, gestione, inizio e valutazione di servizi sanitari.

HL7 message: è in formato ASCII e consiste di uno o più segmenti, separati dal carattere \r. Ogni segmento è mostrato su una riga diversa di testo. Ogni segmento consiste di uno o più campi, separati da "|". Eventuali sotto campi sono separati da "~".

DICOM (Digital Imaging and Communications in Medicine) è lo standard internazionale per immagini mediche e informazioni correlate (ISO **12052**). Definisce i formati delle immagini che possono essere scambiate con i dati e la qualità necessari per l'uso clinico. Raggruppa le informazioni in set di dati: IOD (Image Information Objects Definitions) → le immagini contengono anche informazioni chiave (per es ID paziente).