

Marco Casu

∞ Ingegneria del Software ∞



SAPIENZA
UNIVERSITÀ DI ROMA

Facoltà di Ingegneria dell'Informazione, Informatica e Statistica
Dipartimento di Informatica



Questo documento è distribuito sotto la licenza [GNU](#), è un resoconto degli appunti (eventualmente integrati con libri di testo) tratti dalle lezioni del corso di Ingegneria del Software per la laurea triennale in Informatica. Se dovessi notare errori, ti prego di segnalarmeli.



INDICE

1	Introduzione	3
1.1	Modellazione	3

CAPITOLO

1

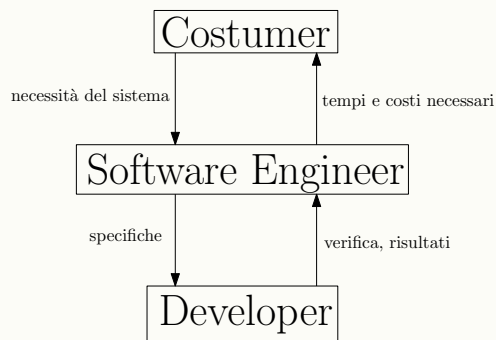
INTRODUZIONE

Quando si vuole produrre software complesso, non terminante e di grandi dimensioni, è necessario ingegnerizzare il metodo di sviluppo ed astrarre il modello del software. L'obiettivo dell'ingegneria del software non riguarda il codice stesso, ma la definizione dell'architettura e delle specifiche del sistema. Vanno definiti i processi di sviluppo.

È cruciale l'analisi dei requisiti, e la modellizzazione delle specifiche di sistema, UML è un linguaggio che permette di descrivere la *dinamica* ed il comportamento del modello. Una volta definiti i requisiti, si affronta la fase di *planning*.

Un modello computazionale che risulterà utile prende il nome di *digital twin*, ossia una copia del software sulla quale è possibile testare i vari *scenari operativi*, tramite la generazione di test automatici. Essendo il sistema discreto, a stati non necessariamente finiti, è possibile descriverlo tramite una catena di Markov.

Nella produzione software prendono parte 3 principali attori



1.1 Modellazione

Il linguaggio UML è utilizzato per modellare sia i requisiti che il sistema stesso, esso è composto da differenti diagrammi

- activity
- use case
- sequence

- class
- state
- context

Si consideri il seguente esempio di un sistema di gestione di una clinica psichiatrica, lo schema in figura 1.1, è il context diagram, e rappresenta l'insieme dei servizi che il sistema offre. Un activity diagram descrive

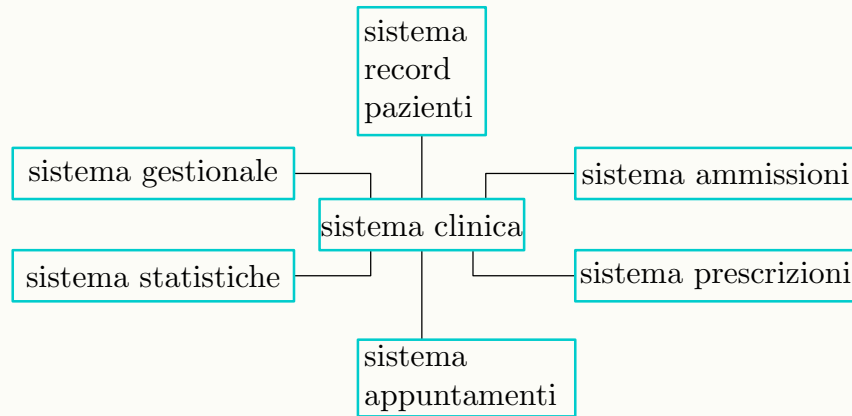


Figura 1.1: context diagram

l'evoluzione di una certa attività/task che il sistema deve poter implementare, si considere l'esempio in figura 1.2 riguardante l'inserimento di un paziente nella clinica.

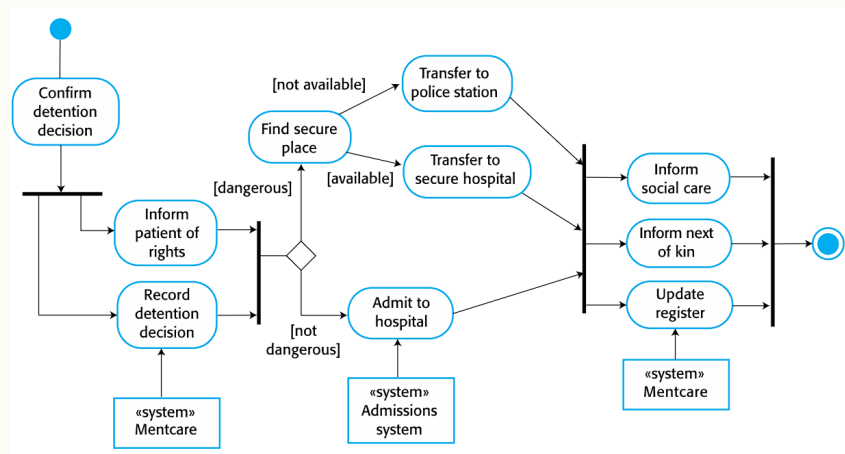


Figura 1.2: activity diagram