

# Conclusioni del corso



*Prof. Paolo Ciancarini*  
*Corso di Ingegneria del Software*  
*CdL Informatica Università di Bologna*

People who love sausages and software  
should never watch either being made

David Lee Todd, Product Manager

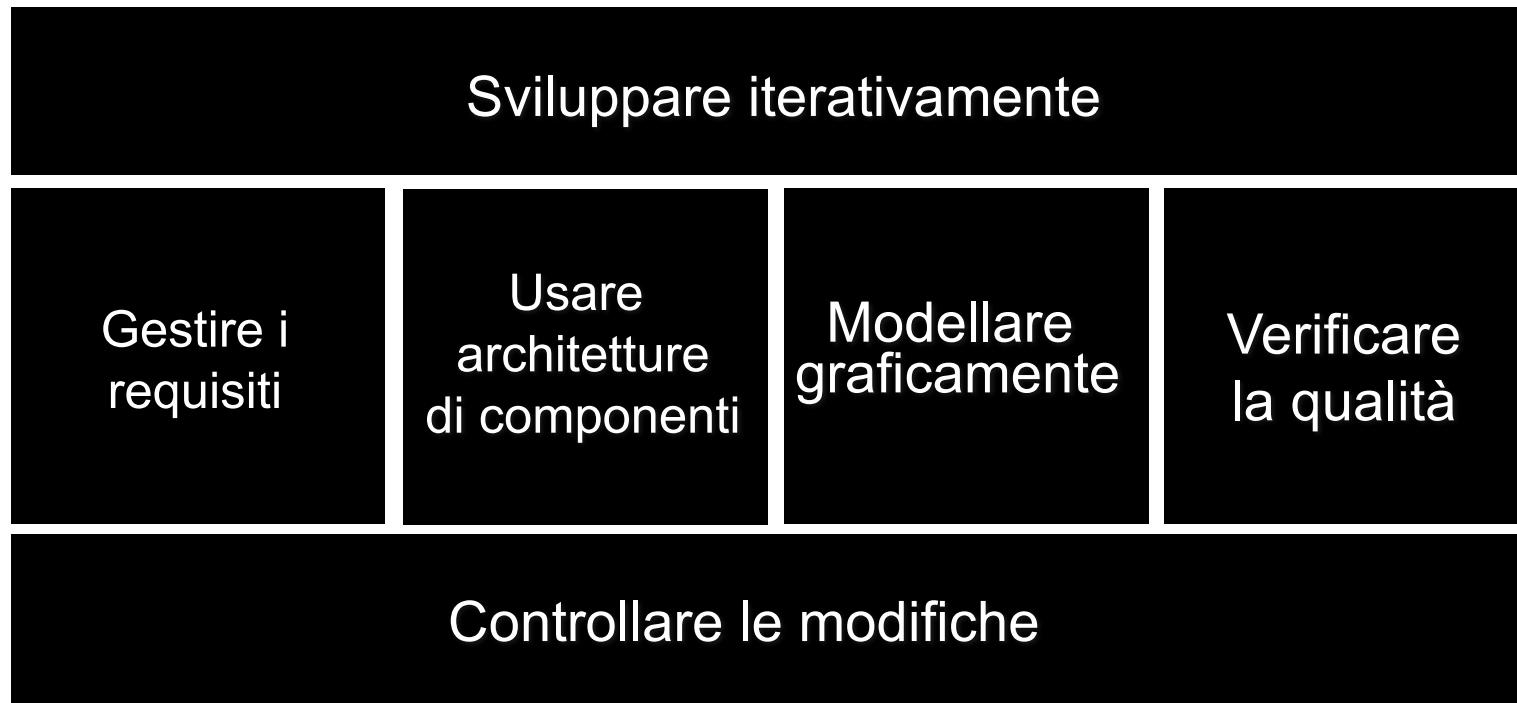
# Scopo del corso

Presentare e sperimentare *metodi e strumenti* di

- analisi,
- modellazione,
- progettazione, e
- misura

di sistemi e applicazioni software

# Principi guida dello sviluppo software



# Legge di Conway (rifrasata da me)

Il software è  
il prodotto di un processo sociale,  
e ne incorpora la struttura

# Lo sviluppo del software

1. Il principio di incertezza (legge di Ziv): lo sviluppo di un software è imprevedibile
2. Legge di Humphrey: i requisiti di un nuovo sistema non saranno ben chiari finché gli utenti non iniziano a usarlo
3. Lemma di Wegner: non è possibile specificare o testare completamente un sistema interattivo
4. Lemma di Langdon: il software evolve più rapidamente quando si avvicina a regioni caotiche
5. Le leggi di Lehman sull'evoluzione del software: i programmi grandi diventano parte del mondo reale, lo cambiano ed evolvono insieme ad esso
6. Un problema non è ben compreso finché non si sviluppa una soluzione
7. Il cono dell'incertezza: lo sforzo necessario allo sviluppo può risultare alla fine da 4 volte a  $\frac{1}{4}$  della stima iniziale

# Le sfide

- Scrivere software è difficile, i metodi e gli strumenti dell'ingegneria del software aiutano
- Anticipare i trend futuri: partecipare alla ricerca e incorporare i risultati nel corso
- Separare i principi eterni dalle pratiche superate
- Aggiornare continuamente il materiale del corso
- Conoscere le pratiche industriali e applicarle in progetti didattici anche piccoli
- Aiutare i neofiti a imparare a imparare
- Offrire aggiornamento ai professionisti

# Cosa abbiamo visto

- Gli standard di produzione del software
- Il ciclo di vita dei prodotti software
- La modellazione del software con UML
- L'analisi e la specifica dei requisiti
- La progettazione del software
- Project Management per sistemi software
- Controllare e misurare la qualità del software
- La manutenzione dei sistemi software
- Strumenti di gestione delle configurazioni



# Adesso sapete rispondere?

- Come si *modella* il software?
- Come si *progetta* il software?
- Come si *riusa* il software?
- Quali *strumenti* sono disponibili per chi costruisce software?
- Quanto *costa* costruire il software?
- Quanto *tempo* ci vuole?
- Come si valuta la *qualità* del software?

Requisiti  
del software

Design  
del software

Costruzione  
del software

Processo  
di sviluppo

Testing  
del software

Qualità  
del software

Evoluzione  
del software

Strumenti  
e metodi

Gestione  
del progetto

Gestione delle  
configurazioni

Ingegneria  
del software

```
graph TD; A((Ingegneria del software)) --- B[Requisiti del software]; A --- C[Design del software]; A --- D[Costruzione del software]; A --- E[Processo di sviluppo]; A --- F[Testing del software]; A --- G[Qualità del software]; A --- H[Evoluzione del software]; A --- I[Strumenti e metodi]; A --- J[Gestione del progetto]; A --- K[Gestione delle configurazioni];
```

# Domande?

