

A detailed, high-angle view of a futuristic electronic circuit board. The board is dark, with intricate, glowing blue and orange traces and components. A large, rectangular display on the right side shows a grid of glowing blue binary code (0s and 1s). The overall aesthetic is high-tech and digital.

UML og UP



## Hvad er UML?

UML står for '**Unified Modeling Language**' og bruges til at beskrive, designe og dokumentere komplekse softwaresystemer. Den blev udviklet i starten af 1990'erne som et fælles sprog for softwareudvikling. UML blev udviklet for at gøre samarbejde lettere og skabe en standard, der kunne bruges af teams på tværs af forskellige virksomheder og projekter.

## UML-diagrammer-teknikker

Der findes forskellige teknikker og diagrammer i UML, men de vigtigste er Use Case-diagrammer, Klassediagrammer og Sekvensdiagrammer.“

**Use Case Diagram:** Beskriver systemets funktionelle krav og viser interaktioner mellem brugere og systemet.

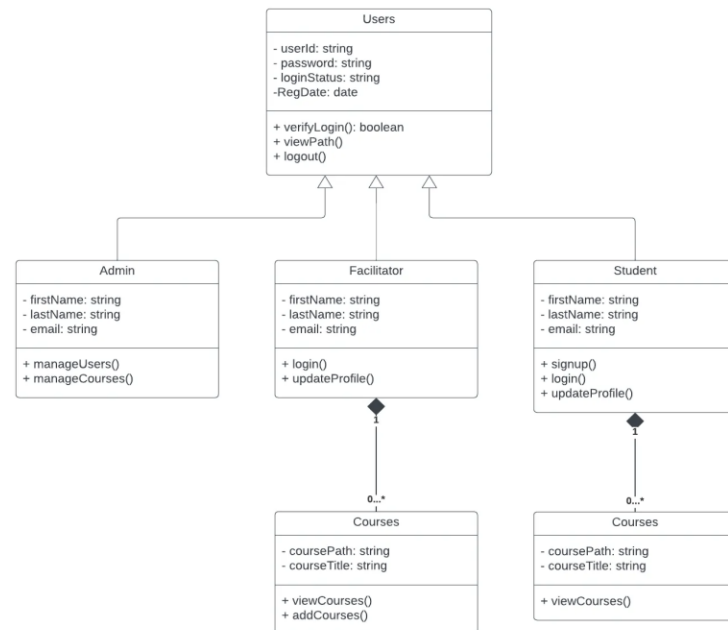
**Klassediagram:** Repræsenterer systemets statiske struktur ved at definere klasser, attributter, metoder og relationer.

**Sekvensdiagram:** Modellerer rækkefølgen af beskeder mellem objekter i et system.

# Ulemper ved UML

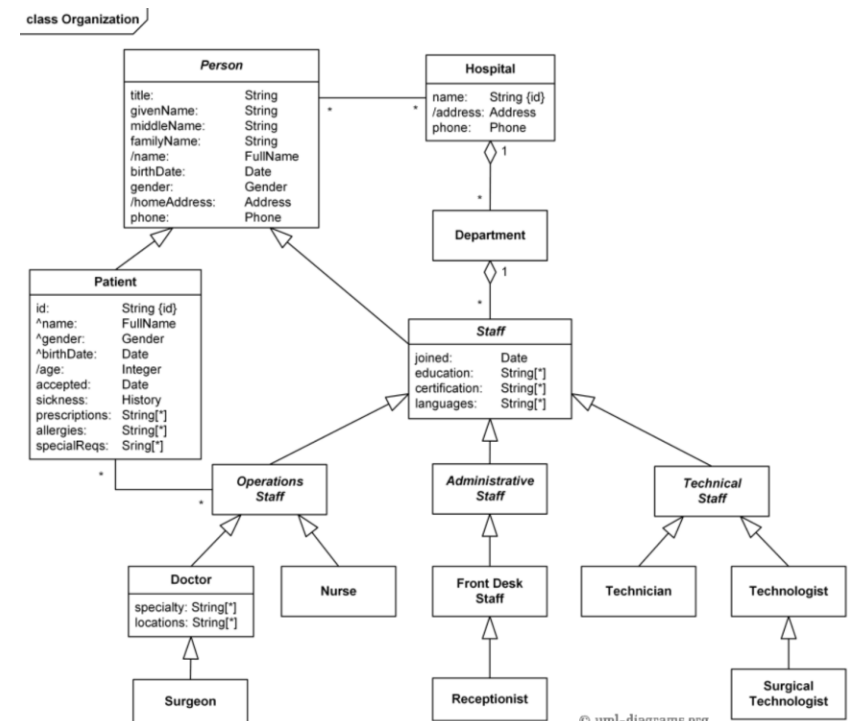
- Når der kommer nye versioner, som UML 2.x, kan der være ændringer i symboler og notationer, hvilket kan være forvirrende
- Desuden kan UML-modeller nogle gange være for abstrakte og ikke altid passe godt til den konkrete kode, man arbejder med.

## Ny version



UML class Diagram of E-Learning Management platform

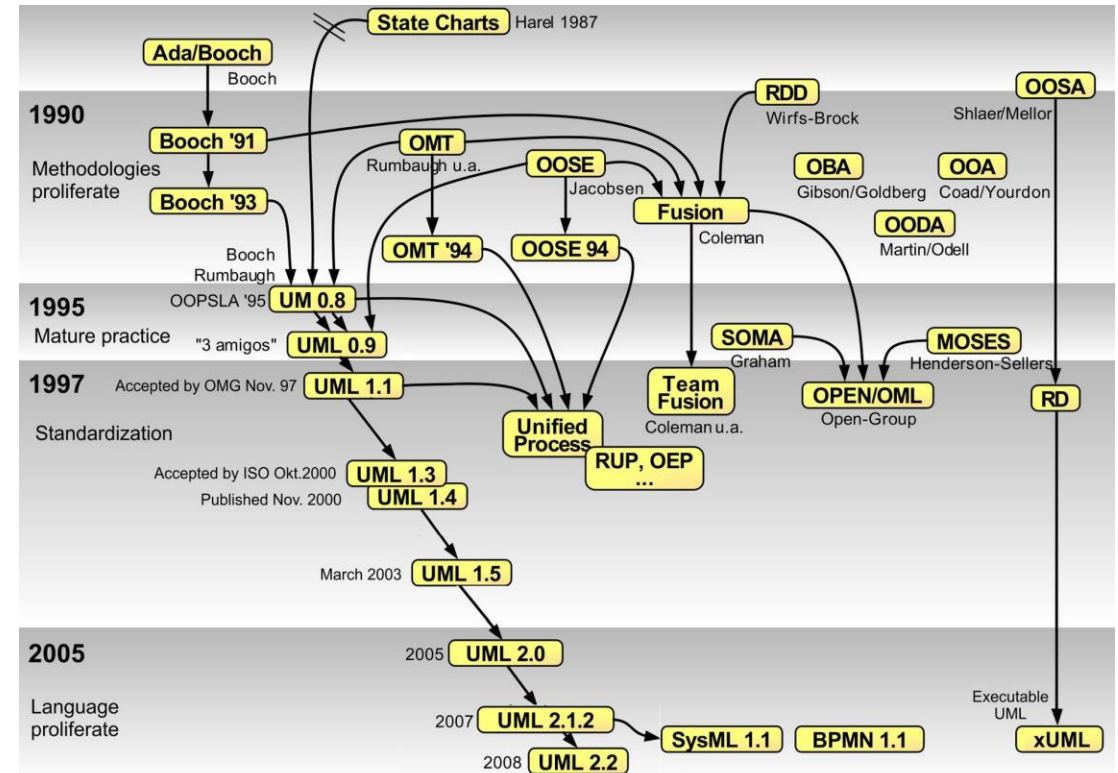
## Gammel version



© uml-diagrams.org

# Fremtiden for UML

- UML blev oprindeligt udviklet for at tilpasse sig til behovene for modellering af softwareapplikationer og systemer.
- Med den fleksibilitet UML har, er det sandsynligt at UML fortsat vil udvikle sig og tilpasse sig nye systemer og tilgange i fremtiden



# Iterative

I den iterative tilgang gentager man en proces flere gange for at forbedre resultatet.

Hver gentagelse, eller iteration, bygger videre på det tidligere arbejde og justeres baseret på feedback og erfaringer.

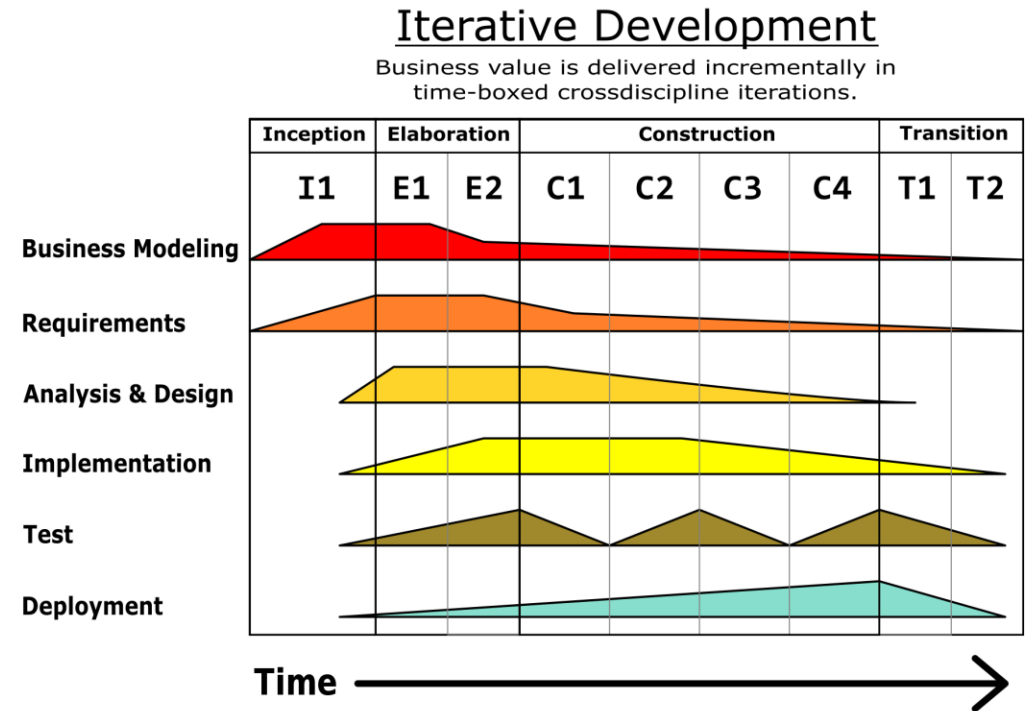
Det betyder, at man hele tiden justerer og optimerer, indtil man opnår det ønskede resultat.

Unified Process er en softwareudviklingsmetode, der arbejder i gentagelser og trinvis forbedring.

Det er en ramme, der hjælper med at organisere og styre softwareprojekter, fra idéfasen til den endelige levering.

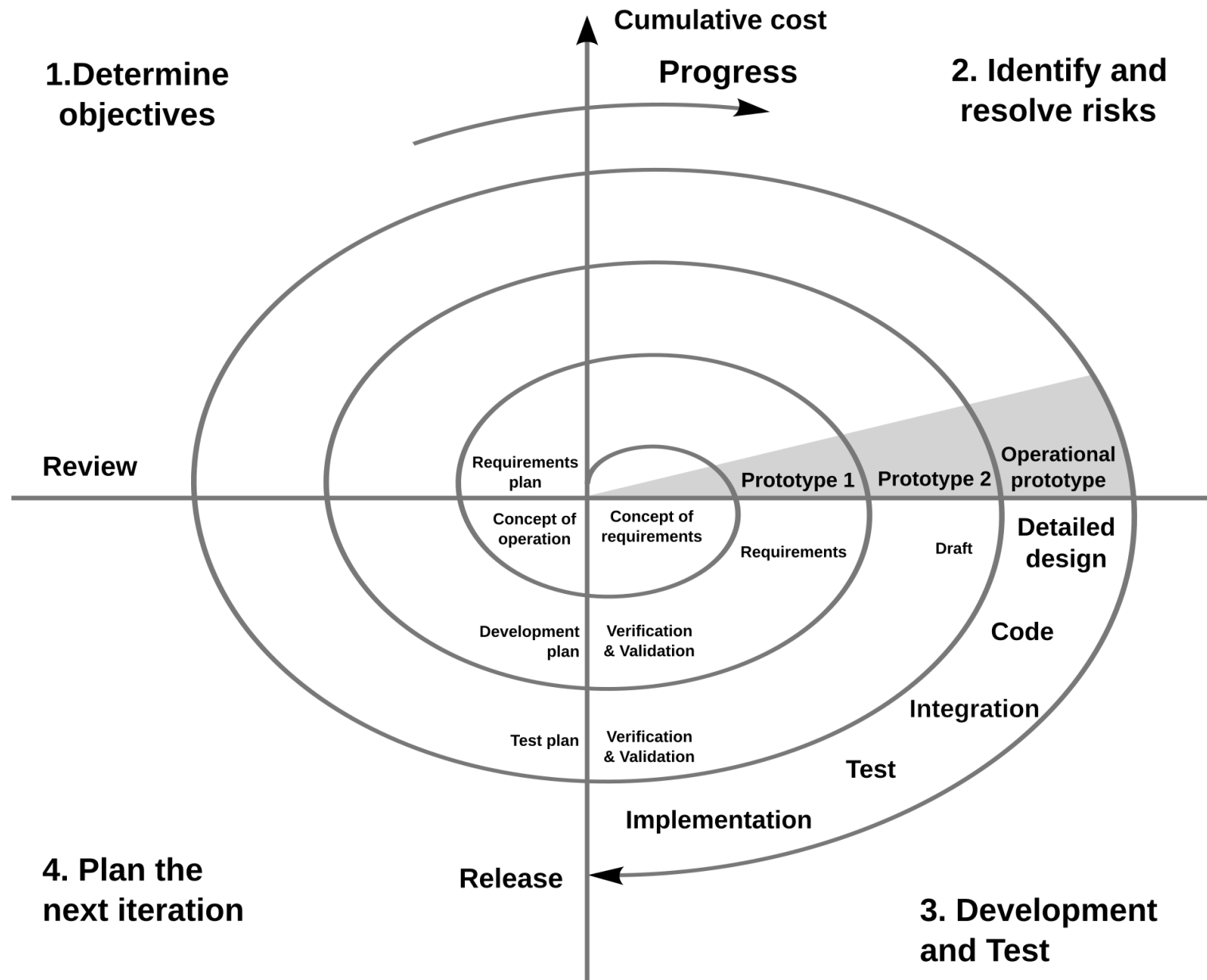
### UP er struktureret i faser, der typisk inkluderer:

- Inception (Opstartsfasen):  
Hvor man fastlægger projektets omfang og mål.
- Elaboration (Uddybning):  
Hvor man udfører en mere detaljeret analyse og design.
- Construction (Udvikling):  
Hvor den faktiske kode skrives, og produktet bygges.
- Transition (Overgang):  
Hvor systemet implementeres og overleveres til brugerne.



# spiral model

- Den Spiral Model er en softwareudviklingsmetode, der kombinerer den iterative natur af prototyper med de kontrollerede og systematiske aspekter af Vandfaldsmodellen. Den er særligt velegnet til store, komplekse og højriskoprojekter og tilbyder en fleksibel tilgang, der muliggør inkrementel fremgang, samtidig med at risici håndteres effektivt. Her er, hvordan Spiral Model kan anvendes i konteksten af Unified Process (UP), en ramme for objektorienteret softwareudvikling:





# Fordele og Ulemper ved UP

- Bryder udviklingsprocessen ned i mindre og håndterbare dele
- Fokus på tidlig risikoidentifikation og -håndtering.
- Kan tilpasses forskellige typer projekter og organisationer
- Brugere og interessenter kan inddrages kontinuerligt gennem udviklingen, hvilket sikrer, at det endelige produkt opfylder deres behov.
- Kan være komplekst at implementere.
- Kræver ofte mange ressourcer i form af tid, penge og menneskelige ressourcer.
- Dokumentationskravene er høje og kræver meget arbejde.
- For mindre og simple projekter kan UP være for tung og omfattende, hvilket kan gøre det ineffektivt sammenlignet med lettere metoder som Scrum eller Kanban.

# Kilder på Unified Process

- Her er nogle pålidelige kilder, hvor du kan finde yderligere information om Unified Process (UP):
- **Bøger:**
  - "The Unified Software Development Process" af Ivar Jacobson, Grady Booch, og James Rumbaugh. Denne bog er skrevet af de oprindelige skabere af UP og giver en dybdegående forståelse af metodens principper og praksis.
  - "Rational Unified Process: An Introduction" af Philippe Kruchten. Dette er en autoritativ guide til RUP, en af de mest kendte implementeringer af UP.
- **Akademiske artikler:**
  - Kruchten, P. (2000). "The Rational Unified Process - An Introduction." En introduktion til RUP, hvor de vigtigste aspekter af UP og dets anvendelse diskuteres.
  - Jacobson, I., Booch, G., & Rumbaugh, J. (1999). "The Unified Software Development Process." En vigtig artikel, der præsenterer de grundlæggende ideer bag UP.
- **Websites:**
  - IBM Knowledge Center - IBM's officielle dokumentation og ressourcer om Rational Unified Process, der giver en omfattende oversigt over metodikken.
  - Rational Unified Process - An Overview - En PDF, der giver en kortfattet introduktion til RUP, en specifik implementering af UP.
- **WikiPedia:**
  - [https://da.wikipedia.org/wiki/Unified\\_Process%E2%80%8B](https://da.wikipedia.org/wiki/Unified_Process%E2%80%8B)

# Kilder på UML

- **Software Engineering Textbooks:** Books like "Applying UML and Patterns" by Craig Larman or "The Unified Modeling Language User Guide" by Grady Booch, James Rumbaugh, and Ivar Jacobson.
- Stack Overflow or Reddit
- [https://en.wikipedia.org/wiki/Unified\\_Modeling\\_Language](https://en.wikipedia.org/wiki/Unified_Modeling_Language)