# Laboratorio di Sviluppo delle Applicazioni Software

## Progettazione (Parte I)

T3 - Prof. Mirko Polato

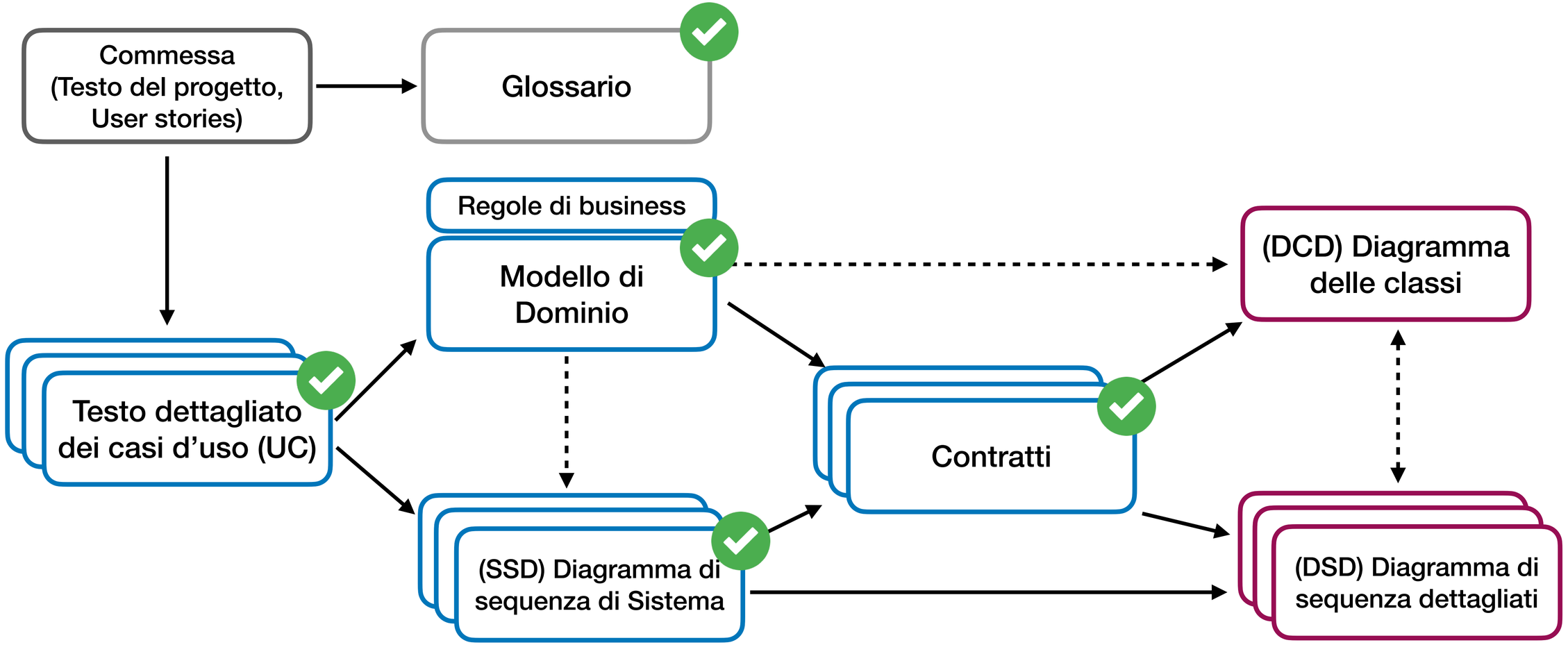
10/05/2023

# Punto della situazione

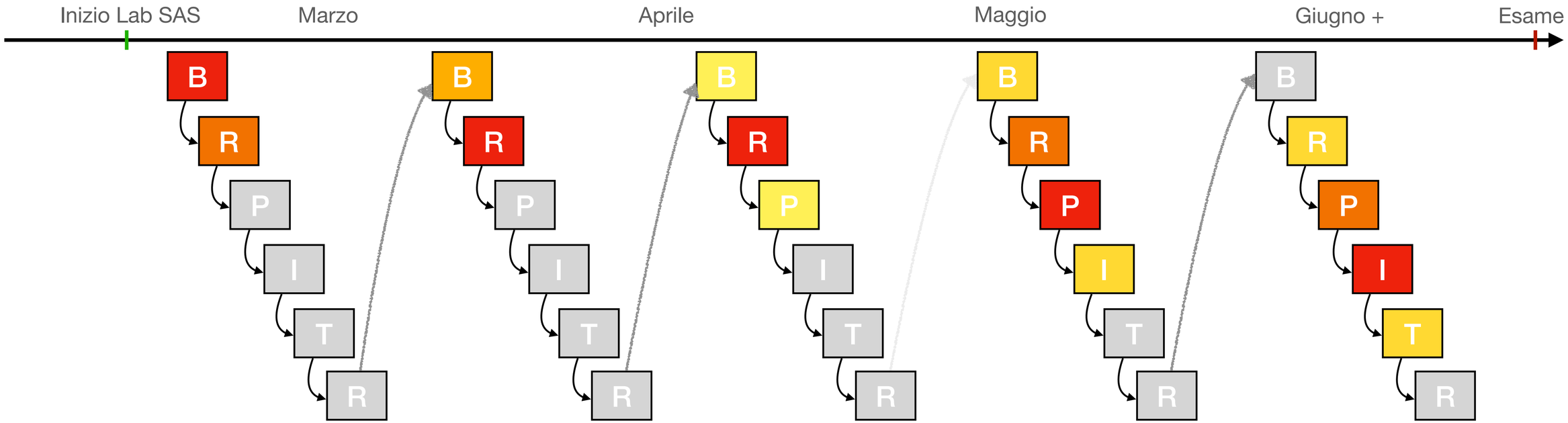
## Da dove partiamo

# Documentazione (codice escluso)

## Artefatti della metodologia UP



# Evoluzione degli artefatti di UP




✅ Glossario

✅ UC dett.

✅ SSD

✅ Mod dom

✅ Contratti

⏳ DSD

⏳ DCD

Codice

Documenti ▲

Use case ▼

UC esempio

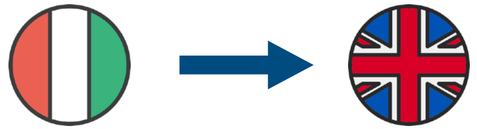
UC esercitazione

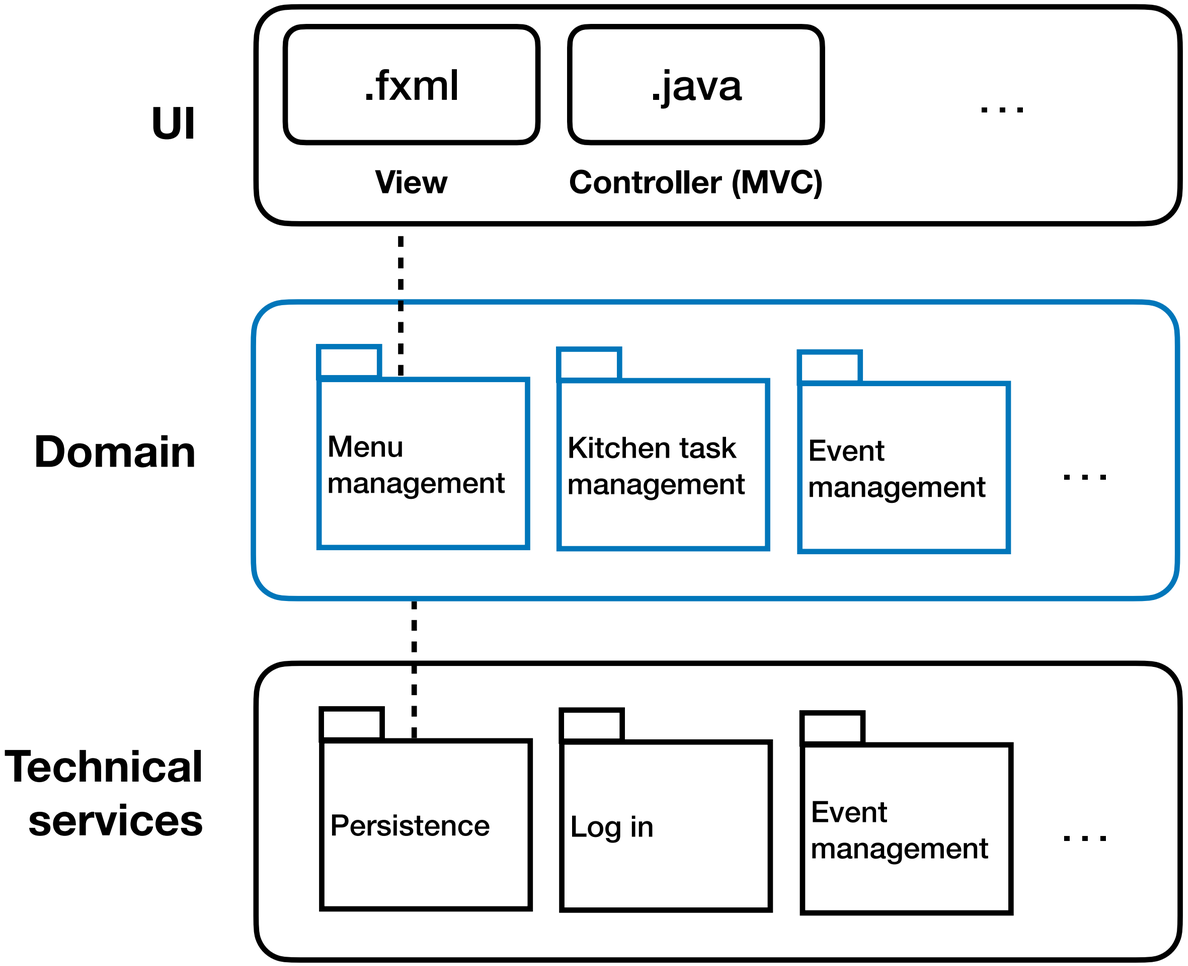
UC d'esame

# Progettazione

## Parte I

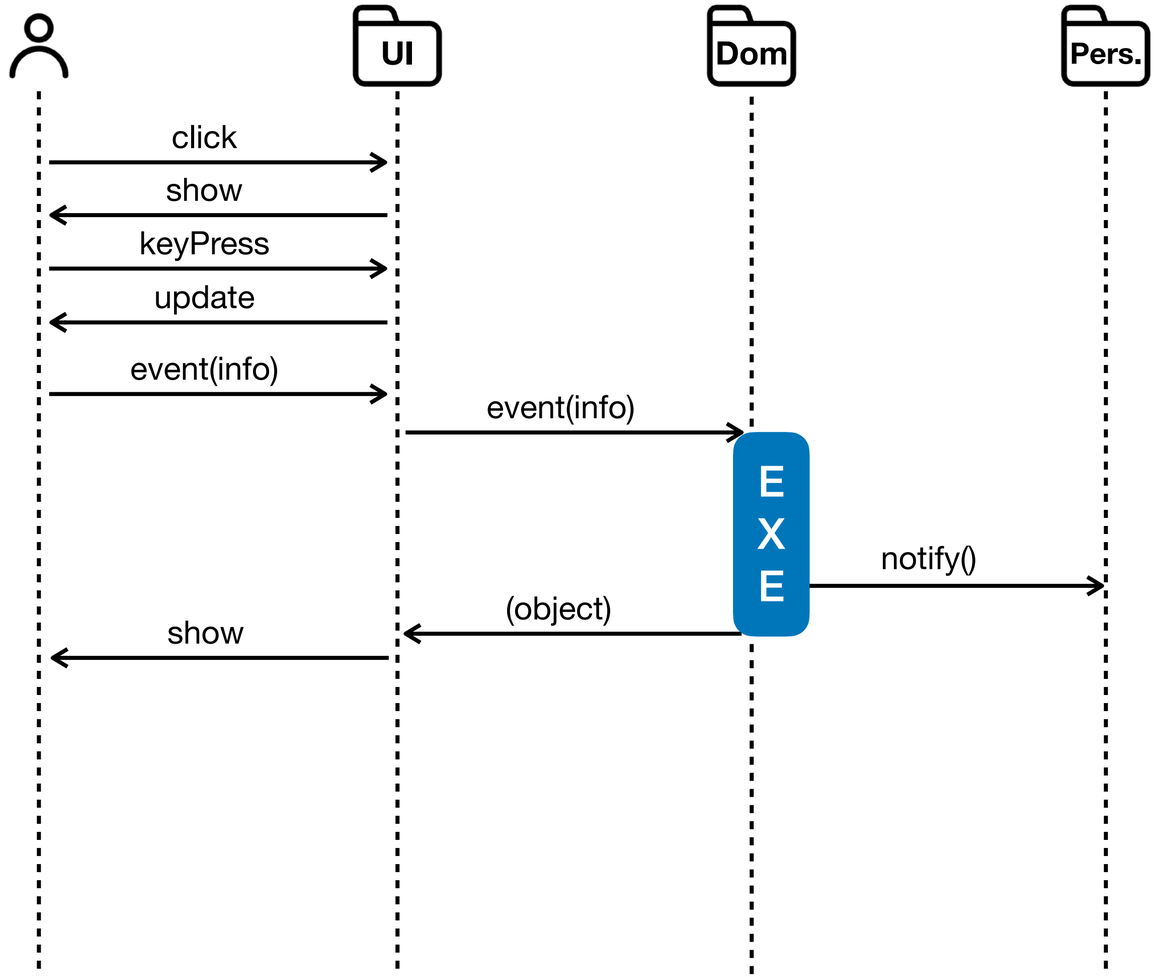
# Cat&Ring - Architettura statica





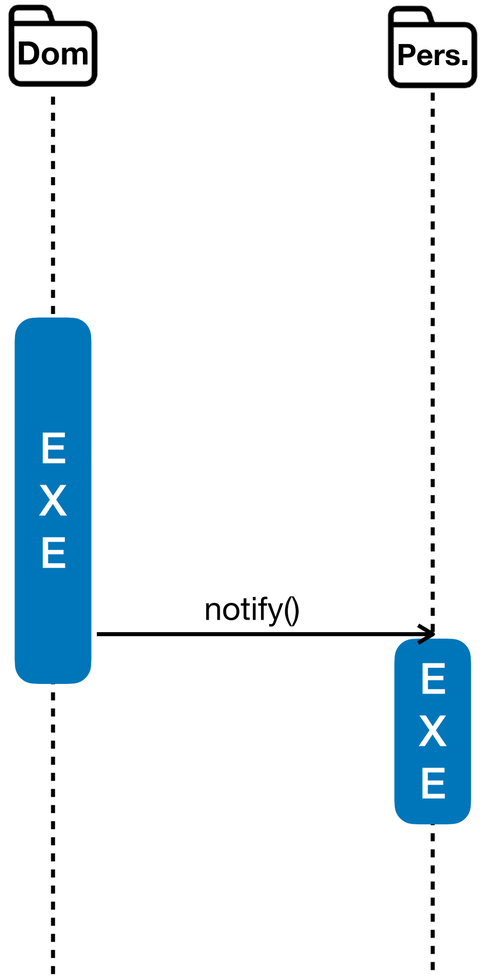
# Cat&Ring - Architettura dinamica

- La **UI non è responsabile della logica delle operazioni**, ma ha “conoscenza” del dominio
- Lo strato di dominio è responsabile della logica
- La **UI conosce il GRASP controller** che funge da “interfaccia” tra UI e Dom
- Esiste una dipendenza forte (inevitabile) tra UI e dominio
- Mentre c’è una dipendenza debole (o inesistente) tra dominio e strato tecnologico



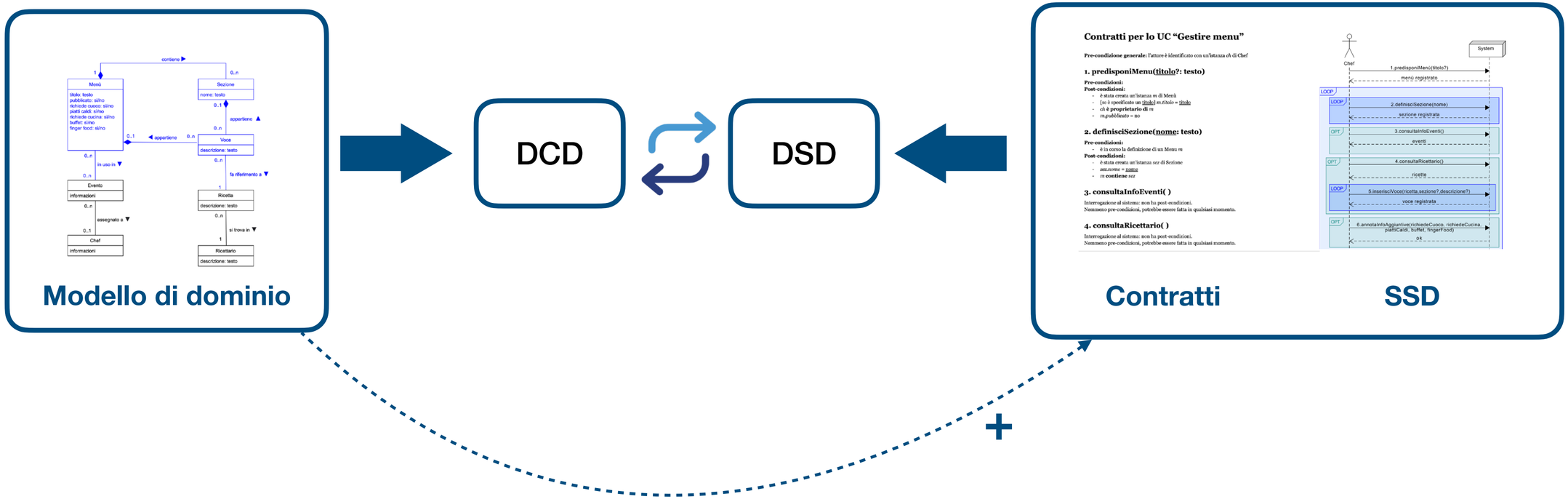
# Pattern Event Receiver (Observer)

- La persistenza sarà gestita utilizzando il pattern Event Receiver (**Observer**)
- Lo strato di persistenza (una sua classe, *observer*) viene **notificato** dallo strato di dominio (*subject*) che il suo stato è cambiato
- Lo strato di dominio deve solamente conoscere chi sono gli osservatori e non la loro logica interna
- Lo strato di dominio e di technical service sono debolmente accoppiati



# DCD e DSD

## Come procedere…



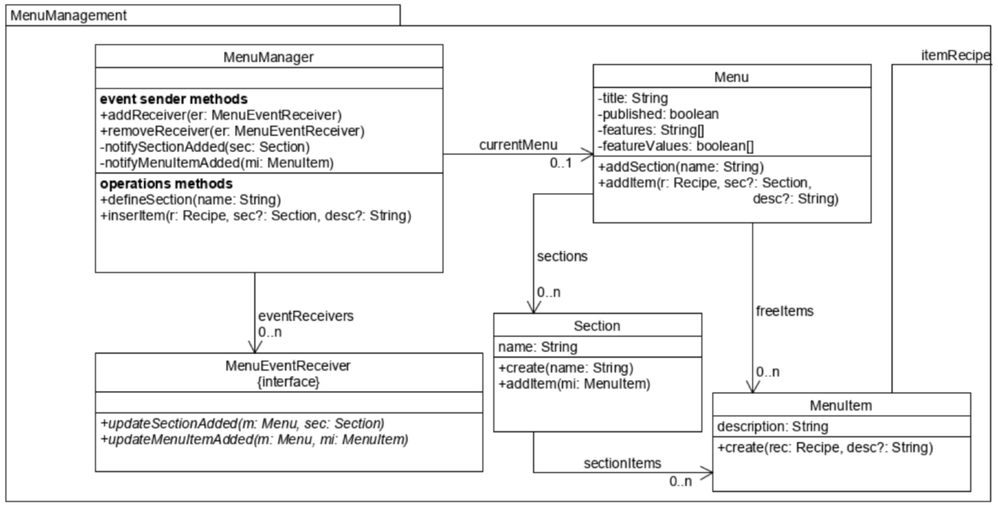
# Design Class Diagram (DCD)

## Diagramma delle classi

# Diagramma delle classi (DCD)

## Visione statica della relazione tra le classi

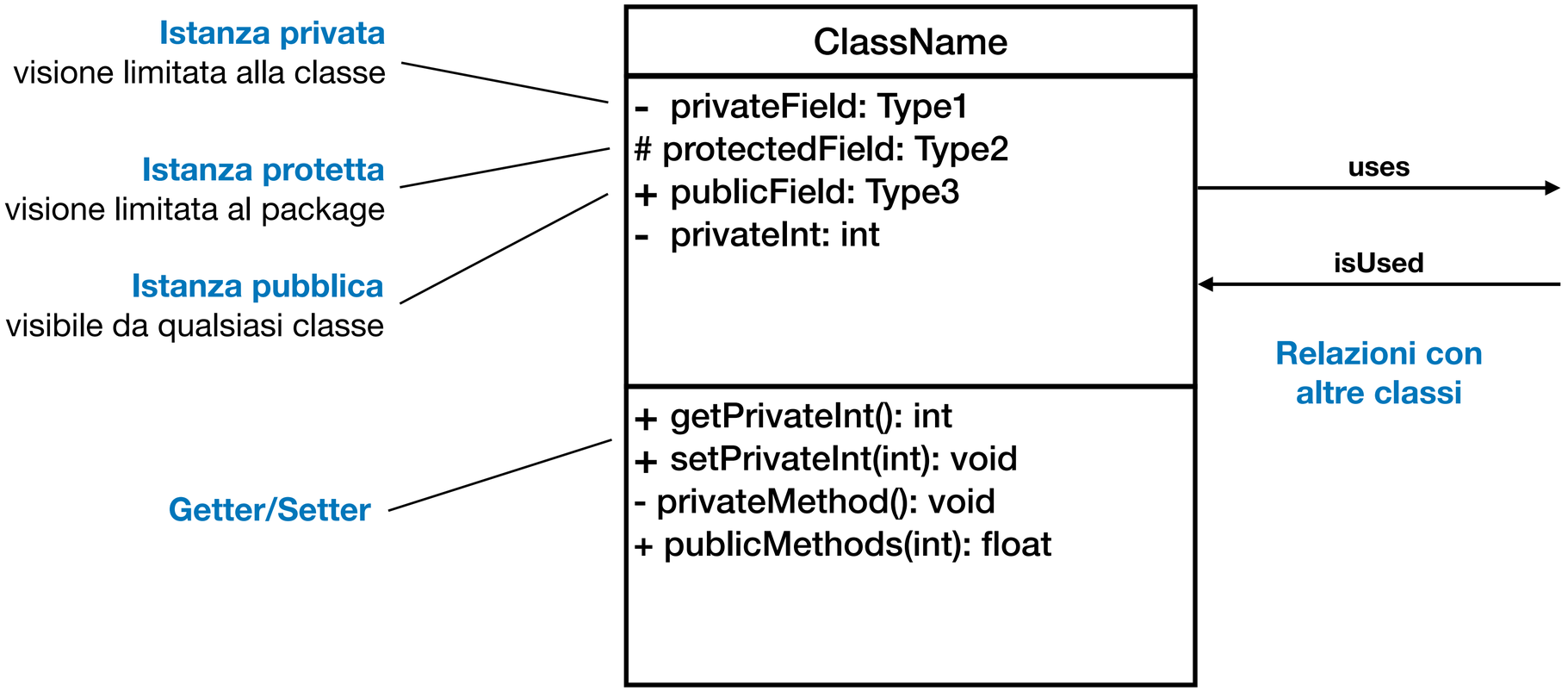
- La creazione dei DCD è parallela a quella dei DSD
- Il DCD **non** è una copia del modello di dominio
- Il **modello di dominio** funge da **ispirazione** per il DCD
- Utilizzare moduli/package per “isolare” i diversi UC
- “Sfruttare” i pattern GoF, quando possibile




**Il Diagramma delle classi è unico per l’intero progetto!!**

# Diagramma delle classi (DCD)

## Classe in UML



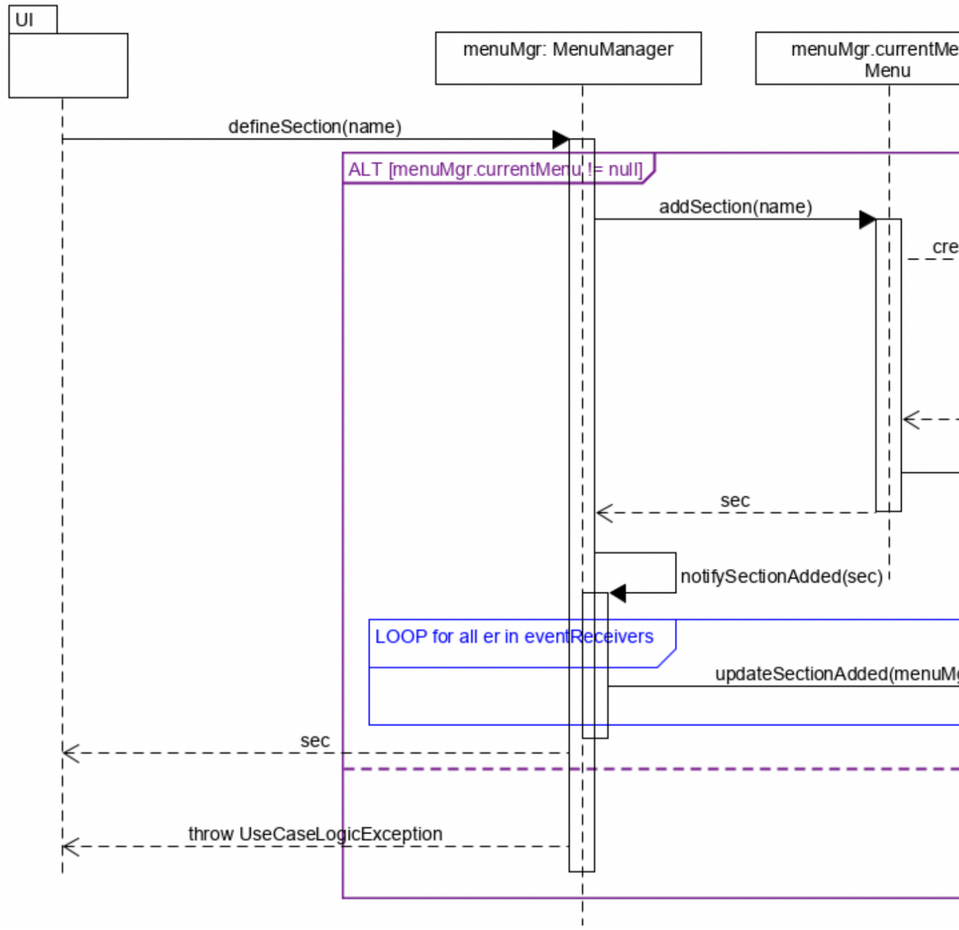
# Detailed Sequence Diagrams

## Diagrammi di sequenza dettagliati

# Diagrammi di sequenza dettagliati (DSD)

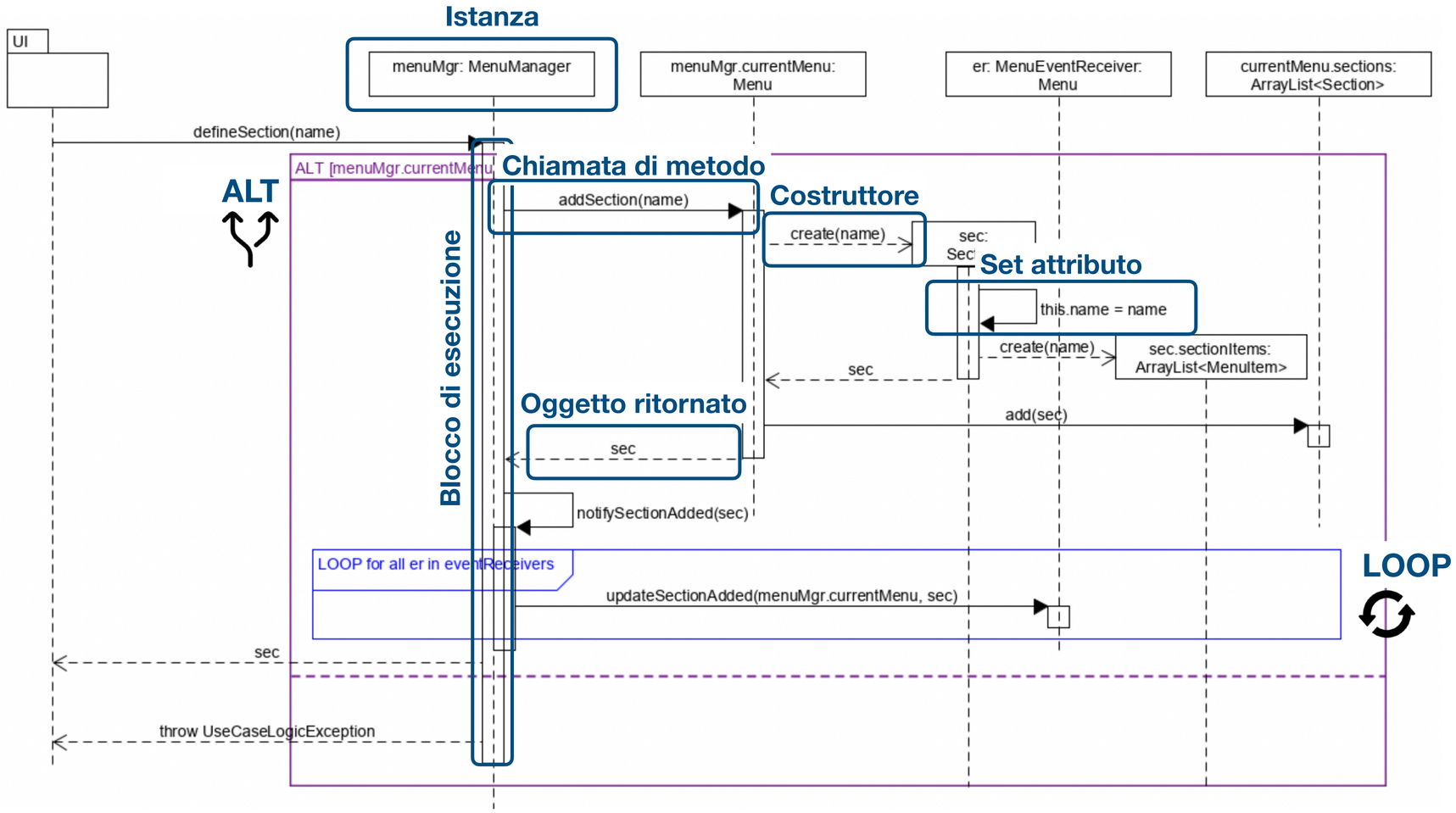
## Visione dinamica della relazione tra le classi

- Modella le interazioni tra gli oggetti in un UC
- Esiste un DSD per ogni operazione definita nei contratti
- Nella pratica, si definiscono DSD solo di operazioni critiche e importanti
- **Buona idea**: partire da DSD di operazioni “impattanti”, ovvero che vi costringono a fare delle scelte
- La prima chiamata di metodo avviene sempre tra UI e Dominio tramite il **controller GRASP**
- **Sfruttate i pattern GRASP (information expert, creator)**



# Detailed Sequence Diagram (DSD)

## DSD Cheatsheet




# Progettazione

## DCD e DSD