

Esercizio 2

Si vuole progettare uno Use Case Editor (chiamato **UCed**) in grado di aiutare un'analista software a scrivere use case testuali in linguaggio naturale che seguono un certo formato. **UCed** sarà dotato di funzionalità CRUD (Create, Read, Update, Delete) e permetterà di effettuare l'interazione utente tramite un GUI. Oltre alle funzionalità CRUD, **UCed** dovrà anche fornire un modulo per la verifica degli use case (sostanzialmente uno smell detector), uno per la simulazione (che elencherà tutti gli scenari possibili), uno per la generazione di activity diagram corrispondenti allo use case selezionato ed uno di generazione automatica di casi di test a partire dallo use case scelto. Gli use case (e in generale tutti i file prodotti da **UCed**) una volta inseriti saranno salvati in modo persistente in un DBMS. Un esempio di use case gestito da **UCed** è mostrato sotto a destra.

Title: User login

Scope: Patient Monitoring System

Level: Sub function

Primary Actor: User (e.g. Doctor, Nurse)

Participants: User

Goal: A User wants to use a PM system to perform an activity such as admitting a patient or changing system parameters.

Precondition: System is ON AND NO user is logged in AND NO card is inserted

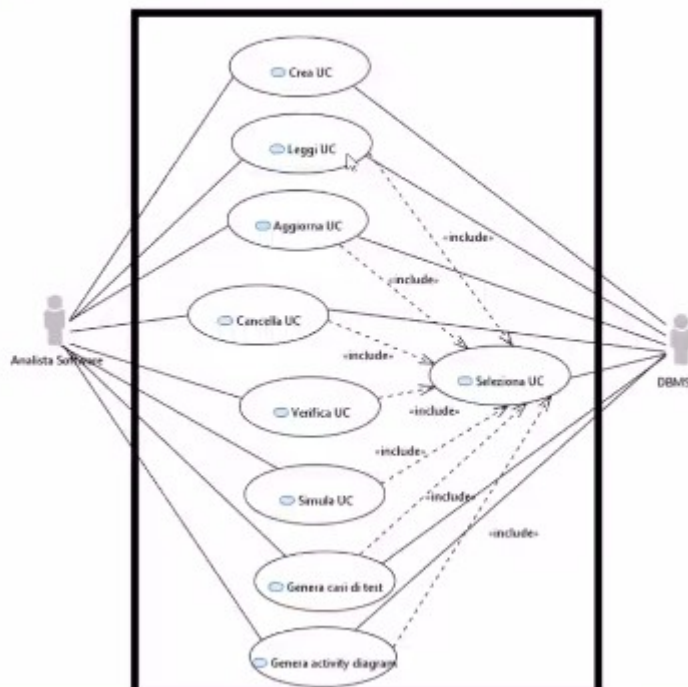
Steps:

- 1: User inserts a Card
- 2: System asks for Personal Identification Number (PIN)
- 3: User types PIN
- 4: System validates USER identification
- 5: System displays a welcome message to USER
- 6: System ejects Card

Exceptions:

- 1a: Card is not regular
 - 1a1: System emits alarm
 - 1a2: System ejects Card
- 4a: User identification is invalid
 - 4a1: IF number of attempts < 4
 - 4a11 Go back to Step 2
 - 4a2: IF number of attempts ≥ 4
 - 4a21: System emits alarm

a) Disegnare il diagramma dei casi d'uso per il sistema UCed. (Punti 4)

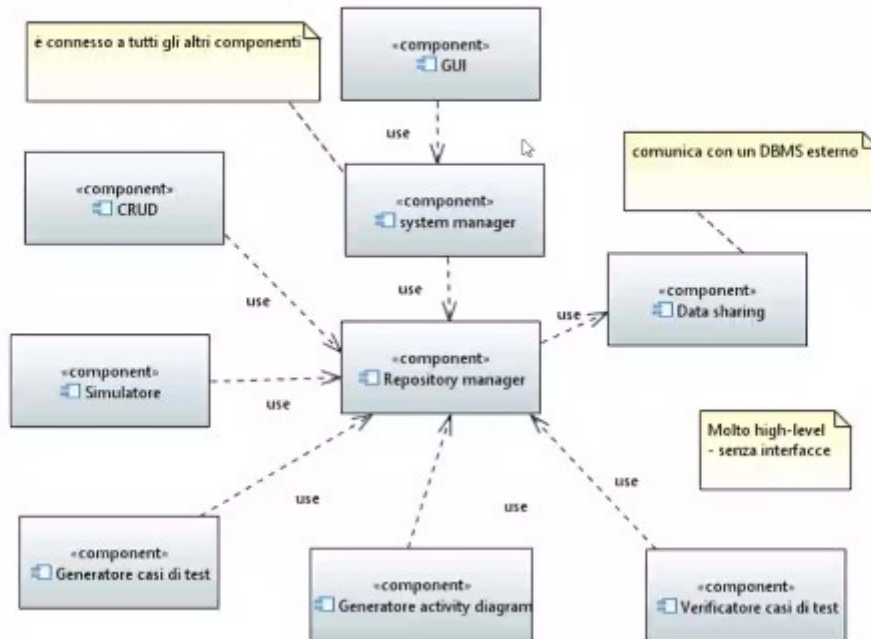


b) Specificare il nome dello stile architetturale che ritenete più appropriato per il sistema software sopra descritto. Commentare la scelta fatta. (Punti 2)

Repository model (simile all'esempio visto in classe su CASE tool)

- c) Disegnare un diagramma che descrive l'architettura del sistema in accordo a tale stile. Commentare brevemente a parole il funzionamento e marcando (quando utile) input e output dei vari componenti. (Punti 4)

Ho usato un component diagram UML ma ad un livello alto di astrazione



- d) Disegnare un activity diagram "equivalente" allo use case *User Login* mostrato a destra. (Punti 3)

