

**Corso di Ingegneria del software – Seconda verifica – 29 maggio 2014**

C, Montangero, L. Semini - Dipartimento di Informatica, Università di Pisa, a.a. 2013/14

La prova si svolge a libri chiusi (non è permessa la consultazione di materiale didattico).  
Traccia delle soluzioni

Si consideri il caso di studio *StammiBene*, e il seguente chiarimento: l'ultima frase deve intendersi come "Se non si acquista almeno un servizio per un anno, la tessera diventa *DimensionePanza*, si perdono i diritti acquisiti e il borsellino è congelato. Al primo acquisto la tessera viene riattivata, come *Sporty*."

Si consideri il seguente caso d'uso.

**Caso D'uso:** iscrizione

**Attore Principale:** Addetto

**Attori secondari:** nessuno

**Precondizione:** Cliente non iscritto a *StammiBene*

**Postcondizione:** Cliente iscritto a *StammiBene*, eventualmente con riserva

**Sequenza principale degli eventi:**

- 1 L'addetto richiede una nuova iscrizione.
- 2 Il sistema attiva lo scanner e richiede il documento d'identità.
- 3 L'addetto dispone il documento d'identità e avvia la scansione.
- 4 Il sistema genera la scheda anagrafica.
- 6 Il sistema richiede il certificato medico.
- 7 L'addetto dispone il certificato medico e avvia la scansione.
- 8 Il sistema registra i dati del cliente.
9. **SE** il nuovo cliente non ha debiti pendenti con alcun centro pre *StammiBene*
  - 9.1. Il sistema emette la tessera di riconoscimento e stampa le credenziali.

**ALTRIMENTI**

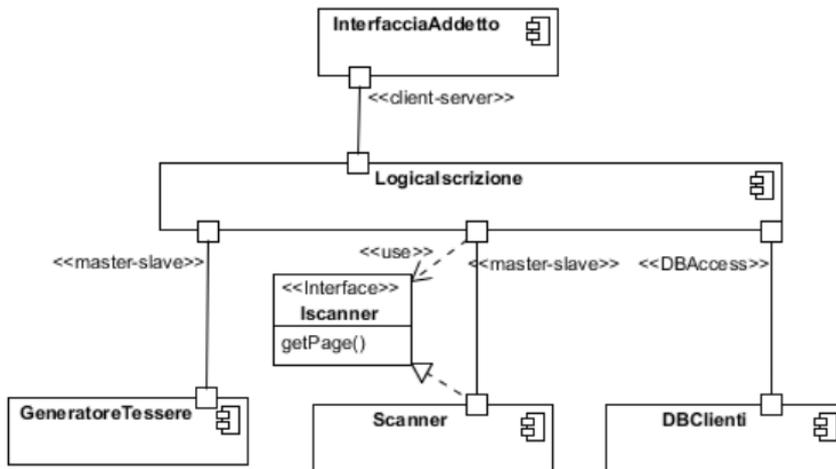
- 9.2. Il sistema dichiara il cliente in attesa di sanare la situazione.

Per realizzare il sottosistema di supporto alle iscrizioni, sono state individuate le seguenti componenti.

Componente	Responsabilità
Interfaccia Addetto	Offre un'interfaccia utente all'addetto della Reception.
LogicaIscrizione	Realizza le operazioni necessarie per l'iscrizione, interagendo con le altre componenti.
DBClienti	Mantiene le informazioni sui clienti.
Scanner	Gestisce le operazioni dello scanner.
GeneratoreTessere	Gestisce le operazioni di creazione delle tessere.

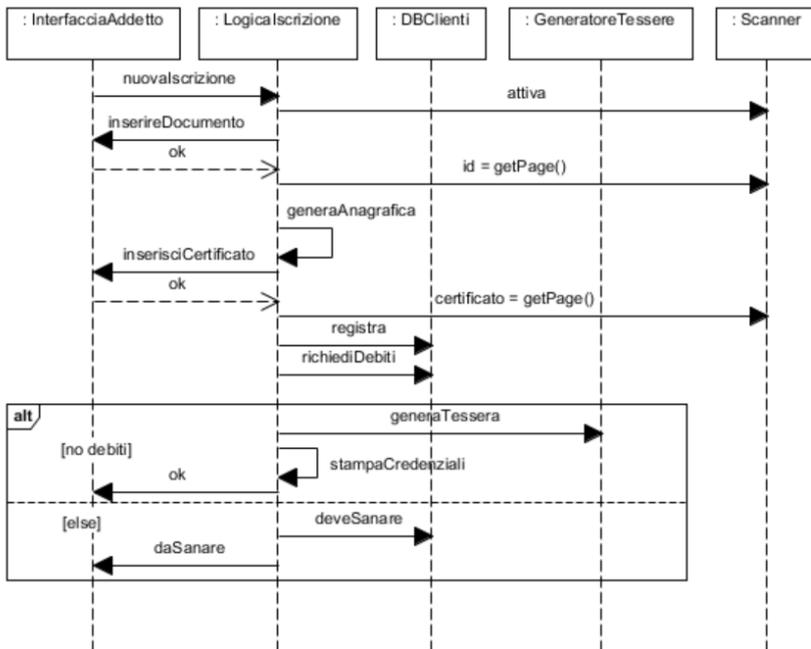
**Domanda 1.** Architettura. Dare un diagramma componenti e connettori per il sottosistema di supporto alle iscrizioni. Rappresentare anche l'interfaccia dello scanner che offre il metodo `getPage()`.

**Risposta.**



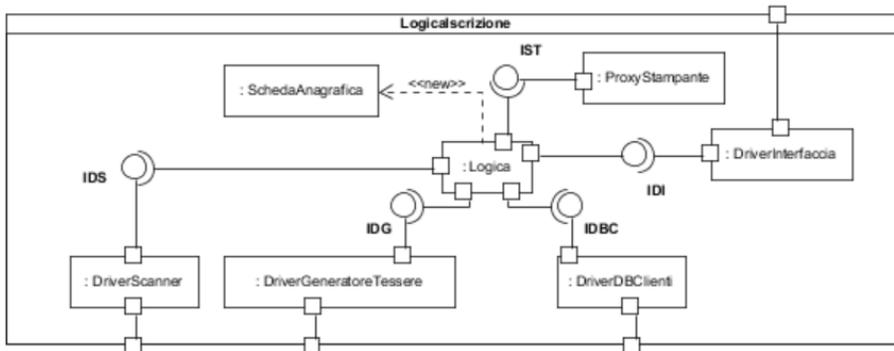
**Domanda 2.** Realizzazione dei casi d'uso. Dare un diagramma di sequenza che mostra come le componenti della vista C&C interagiscono per realizzare il caso d'uso **Iscrizione**.

**Risposta.**



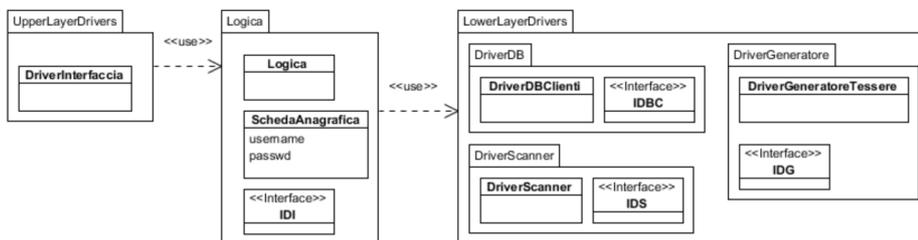
**Domanda 3.** Progettazione di dettaglio. Dare un diagramma di struttura composita per la componente Logicscrizione, raffinando il modello base per evidenziare la creazione della schedaAnagrafica.

**Risposta.**



**Domanda 4.** Architettura. Dare la vista strutturale degli usi relativo ai moduli individuati nell'esercizio 3. Per ogni parte, la classe che ne realizza il tipo e le interfacce offerte, vanno nello stesso package.

**Risposta.**



**Domanda 5.** Si consideri il metodo analizzaClienti, con la seguente specifica: dato un vettore di clienti di StammiBene, restituisce una tripla formata, rispettivamente, dal numero degli clienti che hanno una tessera:

- a) Sporty o DimensionePanza,
- b) Mundialito,
- c) Olympic

rispettivamente. Si assume che il tipo di tessera sia rappresentato da un intero da 0 a 3, con 0 che indica DimensionePanza, fino a 3 per Olympic.

```
public Tripla analizzaClienti (Vector <Cliente> clienti) {
    int sportyOdp = 0;
    int mundialito = 0;
    int olympic = 0;
```

Formattato: Rientro: Sinistro: 0 cm

```

for (Cliente c: clienti) {
    int tipoTesserata = c.getTipoTesserata();
    if (tipoTesserata == 0 | tipoTesserata == 1) sportyOdp ++;
        else if (tipoTesserata > 2) mundialito ++;
            else olympic ++;
}
return new Tripla(sportyOdp, mundialito, olympic);
}

```

**Domanda.** (Verifica)

- a) Definire un caso di test (con ambiente vuoto) con insieme di dimensione minima di valori da attribuire agli elementi del vettore `clienti` che permetta di raggiungere una copertura del 100% del codice del metodo `analizzaClienti`, secondo il criterio dei comandi.
- b) Il caso di test dato evidenzia un malfunzionamento?

**Risposta.**

- a) Una soluzione minima è la seguente: `<[c1,c2,c3], <1,1,1>, __>`

Con:

```

c1.getTipoTesserata()==0, c2.getTipoTesserata()==3
c3.getTipoTesserata()==2

```

- b) No, in quanto il risultato calcolato coincide con quello atteso, anche se il calcolo è fatto in modo scorretto (olympic conta come mundialito e viceversa).  
E' possibile con un test minimo evidenziare il malfunzionamento? Se no, dare un caso non minimo che lo faccia. -

← - - - Formattato: Allineato a sinistra, Rientro: Sinistro: 1,38 cm