

Esercizi in UML

Esercizio 1

Si realizzi il **sequence diagram** del seguente scenario: l'utente desidera registrarsi al sistema bibliotecario per poter accedere ai servizi di prestito e consultazione.

L'interazione inizia con la comunicazione al bibliotecario della volontà di iscriversi alla biblioteca, il bibliotecario opera come intermediario, inserendo la richiesta all'interno di un'interfaccia (un software installato su un PC) per avviare la procedura di registrazione vera e propria.

Non appena l'interfaccia riceve la richiesta di registrazione, la inoltra al sistema centrale che coordina tutte le operazioni di prestito, restituzione, ricerca e registrazione degli utenti. A questo punto, il sistema esegue un controllo interno consultando il Registro Utenti, un database in cui sono archiviati i dati dei lettori già registrati.

Se il Registro Utenti segnala che la persona non è ancora presente, il sistema di gestione procede a effettuare materialmente la registrazione, associando un identificativo univoco (IDUtente) al nuovo iscritto, altrimenti comunica di aver trovato già un account per quell'utente. Una volta terminata questa verifica e, se necessario, la creazione di un nuovo record, il sistema di gestione della biblioteca restituisce l'IDUtente all'interfaccia bibliotecaria.

In conclusione, l'interfaccia notifica al bibliotecario che la procedura di registrazione è andata a buon fine, mostrando l'IDUtente appena generato. Infine, il bibliotecario comunica all'utente l'esito della registrazione, fornendo le credenziali o il codice identificativo che permetteranno di usufruire del servizio bibliotecario.

Esercizio 2

Si realizzi lo **use-case diagram** di un sistema Museo. I visitatori possono accedervi comprando un biglietto venduto da un addetto alla biglietteria o usando biglietti acquistati online. I giovani di età compresa tra i 18 e i 30 anni che acquistino il biglietto in presenza hanno diritto a uno sconto se la verifica di

applicabilità della promozione ha esito positivo (mostrano il documento all'addetto cassa). La visite all'interno del museo avvengono in autonomia oppure con una guida.

Esercizio 3

Si realizzi lo **use-case diagram** di un totem automatico per servizi della Pubblica Amministrazione.

I cittadini hanno la possibilità di richiedere un certificato di residenza, di pagare una multa (con contanti oppure con carta di debito) e di prenotare un appuntamento per il rinnovo del passaporto.

I servizi sono resi accessibili solo previa verifica dell'identità del cittadino, che viene appunto richiesta quando si seleziona uno dei servizi sopra elencati. Questa verifica avviene tramite SPID o tramite CIE, nel primo caso comprovata dal provider di SPID e nel secondo caso comprovata dal server di autenticazione del Ministero dell'Interno.

Esercizio 4

Si realizzi lo **use-case diagram** di un Content Management System, come Wordpress, utilizzato dall'amministratore di un sito web.

In primis, si deve immaginare il sito come un quotidiano online dove gli autori pubblicano i propri articoli, ma tutto questo esula dal dominio del sistema che noi vogliamo modellare. È importante capire che gli autori possono pubblicare solo se, appunto, l'admin mediante il CMS crea per loro: una nuova categoria di articoli (es. la redazione ingaggia per la prima volta un giornalista esperto di tecnologia, perciò dovrà aprire la sezione omonima sul sito web) e un account per il nuovo giornalista.

In particolare, per quanto riguarda la creazione di un nuovo account, l'admin può creare profili di tipo "Giornalista" e "Moderatore" (con più privilegi).

Le operazioni che l'admin svolge mediante il CMS possono fallire e, in tal caso, viene scritto in automatico un nuovo record descrittivo del problema su un file di log.

Infine, le operazioni che l'admin svolge richiedono sempre un ulteriore fattore di autenticazione (es. la conferma di creazione di un nuovo account richiede l'inserimento di un codice OTP che arriva sullo smartphone dell'admin).

Esercizio 5

Si realizzi il **sequence diagram** del sistema codificato come segue, relativamente alla procedura di prelievo da sportello:

```
import java.util.Scanner;

// Questa classe rappresenta l'interfaccia con cui l'utente interagisce
class InterfacciaUtenteBancomat {

    void prelevaContanti() {
        Scanner scanner = new Scanner(System.in);

        System.out.println("Benvenuto al Bancomat");

        System.out.print("Inserisci il PIN: ");
        int pin = scanner.nextInt();

        Bancomat bancomat = new Bancomat();

        boolean pinCorretto = bancomat.verificaPIN(pin);

        if (pinCorretto) {
            System.out.print("Inserisci l'importo da prelevare: ");
            int importo = scanner.nextInt();
            bancomat.preleva(importo);
        } else {
            System.out.println("PIN errato. Operazione annullata.");
        }

        scanner.close();
    }
}

// Questa classe rappresenta il Bancomat vero e proprio
class Bancomat {

    // Metodo per verificare il PIN
    boolean verificaPIN(int pin) {
        return pin == 1234; // PIN corretto fisso per semplicità
    }

    // Metodo per eseguire il prelievo
    void preleva(int importo) {
        System.out.println("Prelievo di " + importo + " euro effettuato.");
    }
}
```

```
}

// Classe principale per eseguire il programma
public class Main {
    public static void main(String[] args) {
        InterfacciaUtenteBancomat interfaccia = new InterfacciaUtenteBancomat();
        interfaccia.prelevaContanti();
    }
}
```