

Mercoledì

Domanda 1 Descrivere con un diagramma UML i casi d'uso del sistema. Dare la narrativa di un caso d'uso che abbia un partecipante o un invitato come attore principale.

Domanda 2 Usando un appropriato diagramma UML illustrare il processo di invito, accettazione/rifiuto ristretto agli inviti fatti da invitati a favore di amici.

Domanda 3 Sapendo che il sottosistema che gestisce gli affitti dei costumi sarà dislocato su un server distinto dal server degli organizzatori che ospita il sottosistema "inviti", definire un'architettura del sistema Mercoledì e darne la vista C&C e la vista di dislocazione, assumendo che a ogni componente corrisponda un artefatto omonimo.

Domanda 4. Si consideri il seguente codice, che definisce un proxy. Mostrare con un diagramma delle classi UML lo schema del pattern istanziato al caso considerato.

```
public class CostumeServiceProxy implements CostumeService {
    private CostumeService costumeService;

    public CostumeServiceProxy(CostumeService costumeServiceImpl) {
        costumeService = costumeServiceImpl;
    }

    public void prenotaCostume(String tipoCostume, String taglia) {
        costumeService.prenotaCostume(tipoCostume, taglia);
    }

    public void pagaCostume(String tipoCostume, float importo) {
        float costoCostume = getImporto(tipoCostume);
        if (importo >= costoCostume) {
            costumeService.pagaCostume(tipoCostume, importo);
        } else {
            System.out.println("Importo insufficiente");
        }
    }

    public float getImporto(String tipoCostume) {
        return costumeService.getImporto(tipoCostume);
    }
}
```

Domanda 5. Definire il codice delle classi che realizzano uno scaffolding per il test di unità della classe `CostumeServiceProxy`.