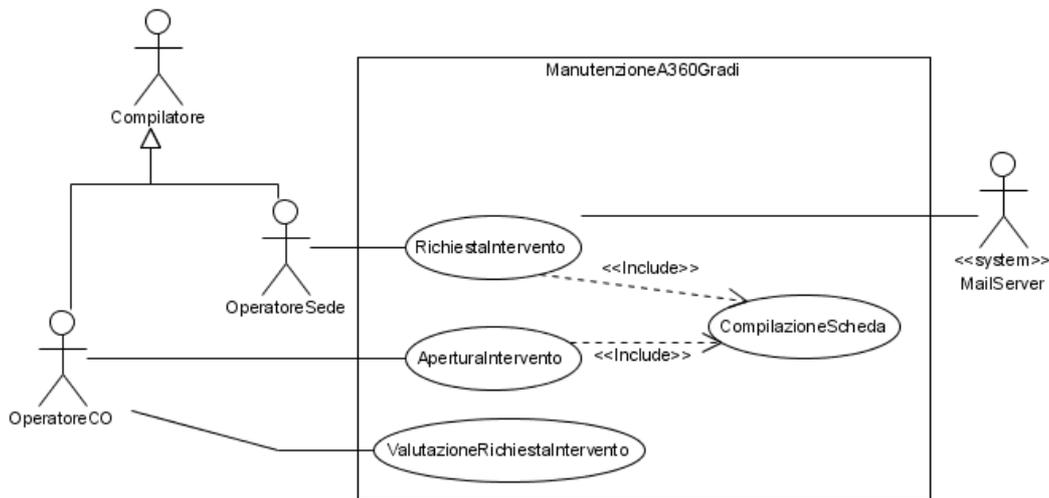


La prova si svolge a libri chiusi (non è permessa la consultazione di materiale didattico).

Traccia delle soluzioni

Si consideri il caso di studio ManutenzioneA360Gradi e in particolare si considerino i seguenti casi d'uso:



CompilazioneScheda:

Permette di compilare una SdI.

AperturaIntervento:

Permette di compilare e accettare una SdI per un guasto segnalato per telefono.

RichiestaIntervento:

Permette di compilare una SdI e inviare mail alla CO.

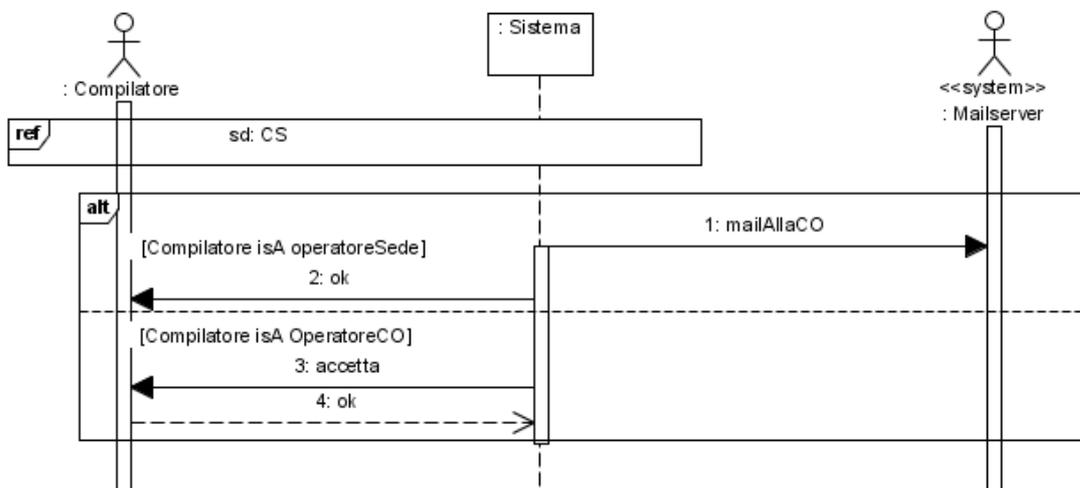
ValutazioneRichiesta:

Permette di accettare o rifiutare la SdI compilata da una sede.

Domanda 1. Realizzazione dei casi d'uso.

Assumendo di aver definito il diagramma di sequenza ds:CS che descrive il caso d'uso CompilazioneScheda, dare un diagramma di sequenza che descriva sia il caso d'uso AperturaIntervento sia RichiestaIntervento. Usare i frame di interazione: ref e alt.

Risposta. Un possibile diagramma di sequenza è il seguente.



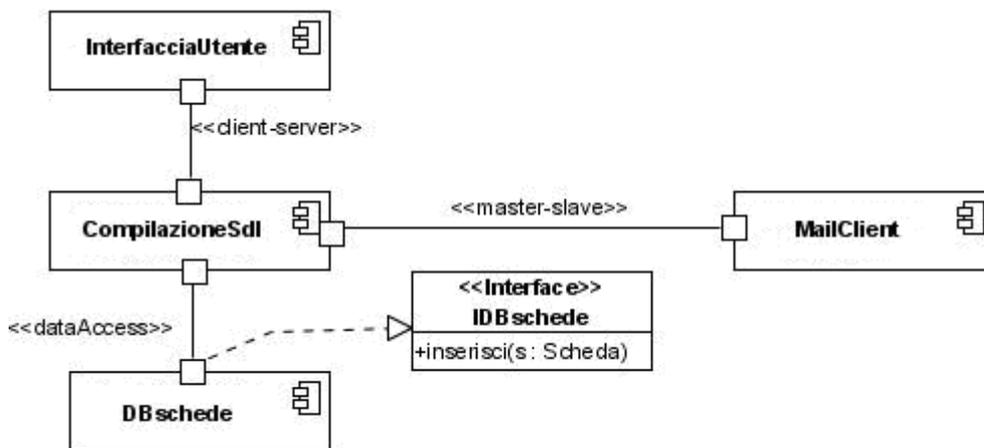
Sia **NuovoIntervento** il sottosistema che realizza i tre casi d'uso: **CompilazioneScheda**, **AperturaIntervento**, **RichiestaIntervento**. Per realizzare **NuovoIntervento** sono necessarie le seguenti componenti:

Componente	Responsabilità
InterfacciaUtente	Interfaccia che permette di inserire i dati di una SdI
CompilazioneSdI	Permette compilazione e accettazione di una SdI, con aggiornamento del DB. Manda mail alla CO.
DBschede	Mantiene le schede di intervento
MailClient	Client di posta.

Domanda 2. Architettura.

- Fornire una vista C&C dell'architettura del sotto-sistema **NuovoIntervento**.
- Mostrare l'interfaccia offerta da DBSchede, per permettere alla componente CompilazioneSdI di memorizzare le schede.

Risposta.

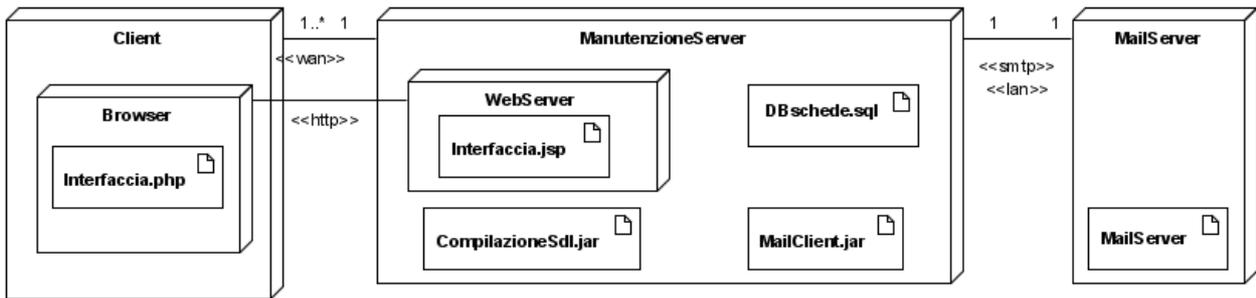


Analisi dei requisiti e progettazione hanno portato alle scelte seguenti. L'operatore di sede inserisce le SdI usando un'interfaccia web. Gli artefatti che manifestano le componenti citate sopra sono: *Interfaccia.php*, visualizzato da un browser di una macchina client; *Interfaccia.jsp* (dislocata su un web server), *CompilazioneSdI.jar*, *DBschede.sql*, *MailClient.jar*, dislocate su una macchina *ManutenzioneServer*.

Si assuma inoltre che il *MailServer* sia su una macchina dedicata, *ServerMail*, collegata in rete locale con *ManutenzioneServer*.

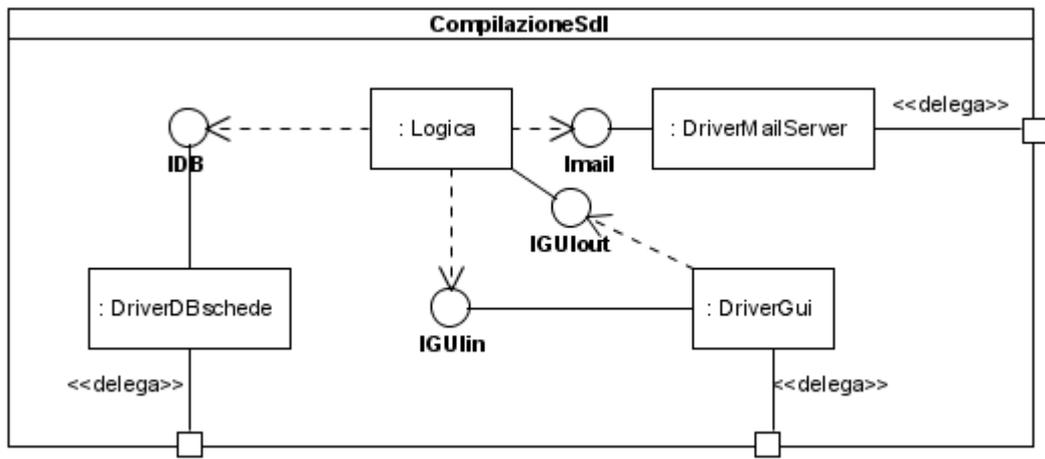
Domanda 3. Architettura: Fornire una vista di dislocazione dell'architettura del sotto-sistema **NuovoIntervento**, più il server di posta.

Risposta



Domanda 4.Progettazione di dettaglio. Dare un diagramma di struttura composta per la componente CompilazioneSdI. Di che tipo sono i connettori che collegano un driver ad un porto? Indicarlo nel diagramma usando uno stereotipo.

Risposta.



Il metodo, da usare a fine mese, per archiviare gli interventi chiusi differenzia le informazioni salvate a seconda se si tratta di

- intervento aCanone e di poca importanza (piccoloIntervento, pI),
- altri interventi aCanone, e interventi extraCanoneSottoSoglia (interventoNormale, iN),
- interventi extraCanoneOltreSoglia (eCOS).

Le informazioni vengono salvate rispettivamente in formaRidotta (fR), formaNormale (fN), e formaEstesa (fE).

Il seguente metodo determina la forma in cui devono essere salvate le informazioni:

```
public String[]
    formaArchiviazione (Intervento[] listaInterventi);
```

Senza entrare nel dettaglio concreto delle SdI, indicheremo con interX gli interventi, con la convenzione che X sia un intero, e che valori di X nella prima decade (tra 0 e 9) indichino interventi di tipo pI, quelli nella seconda decade interventi di tipo iN, e quelli nella terza di tipo eCOS.

Domanda 5. (Verifica)

Si completi con i valori attesi il seguente insieme di casi di test, in cui utilizziamo la notazione Java per l'inizializzazione degli array, per indicare un valore di tipo array:

a)

input	output
{}	
{inter9, inter10, inter20}	

b) Si consideri la seguente realizzazione di formaArchiviazione:

```
public String [] formaArchiviazione(Intervento[] listaInterventi) {
    String [] lia = new String[listaInterventi.length]; //1
    for(int i=0; i < listaInterventi.length; i++) {
        Intervento intervento = listaInterventi[i]; //2
        if (intervento.tipo()== "aCanone" && intervento.piccolo()) { //3
            lia[i] = "fR";
        } else if (intervento.tipo()== "extraCanoneOltreSoglia") { //4
            lia[i] = "fN";
        } else { //5
            lia[i] = "fE";
        }
    }
    return lia; //6
}
```

Dare un diagramma di attività per il grafo di flusso del metodo, utilizzando i numeri a sinistra come nomi dei nodi.

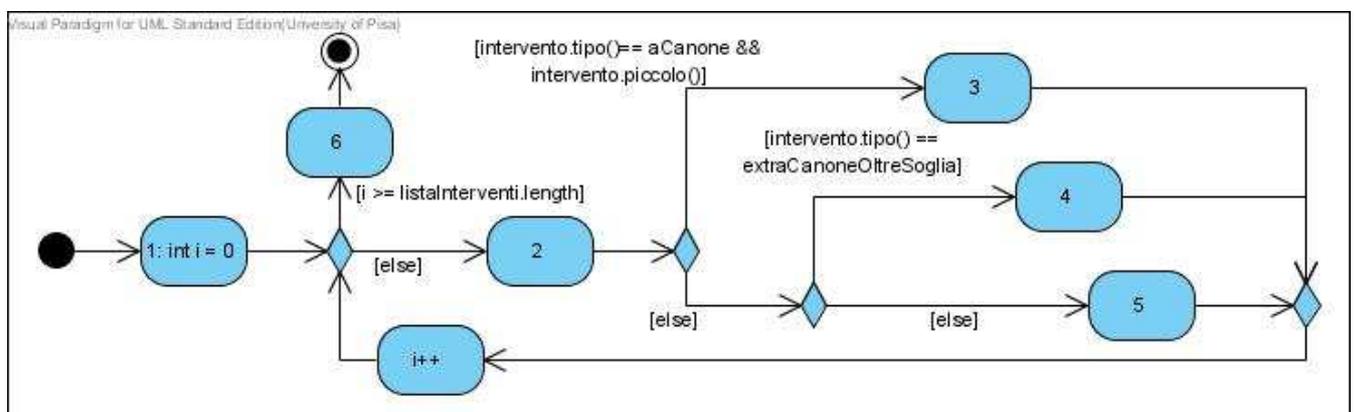
c) Cosa si può dire sulla copertura del codice ottenuta eseguendo i casi di test dati in a), rispetto al criterio delle condizioni? Giustificare la risposta.

Risposta:

a)

Input	output
{}	{}
{inter9, inter10, inter20}	{"fR", "fN", "fE"}

b)



- c) Indicando con
 A la condizione $i < \text{listaInterventi.length}$,
 B la condizione $\text{intervento.tipo() == aCanone}$
 C la condizione $\text{intervento.piccolo()}$
 D la condizione $\text{intervento.tipo == extraCanoneOltreSoglia}$

Le due possibilità per A sono verificate dal secondo caso di input. La seguente tabella indica quali altre condizioni sono valutate per i vari input:

	B	!B	C	!C	D	!D
Inter9	x		x			
Inter10		?		?		x
Inter20		x			x	

Inter10 soddisfa $B \ \& \ !C$, oppure $!B$, per cui una delle condizioni non viene controllata. Abbiamo quindi una copertura di almeno $i \ 7/8 = 87\%$ circa.