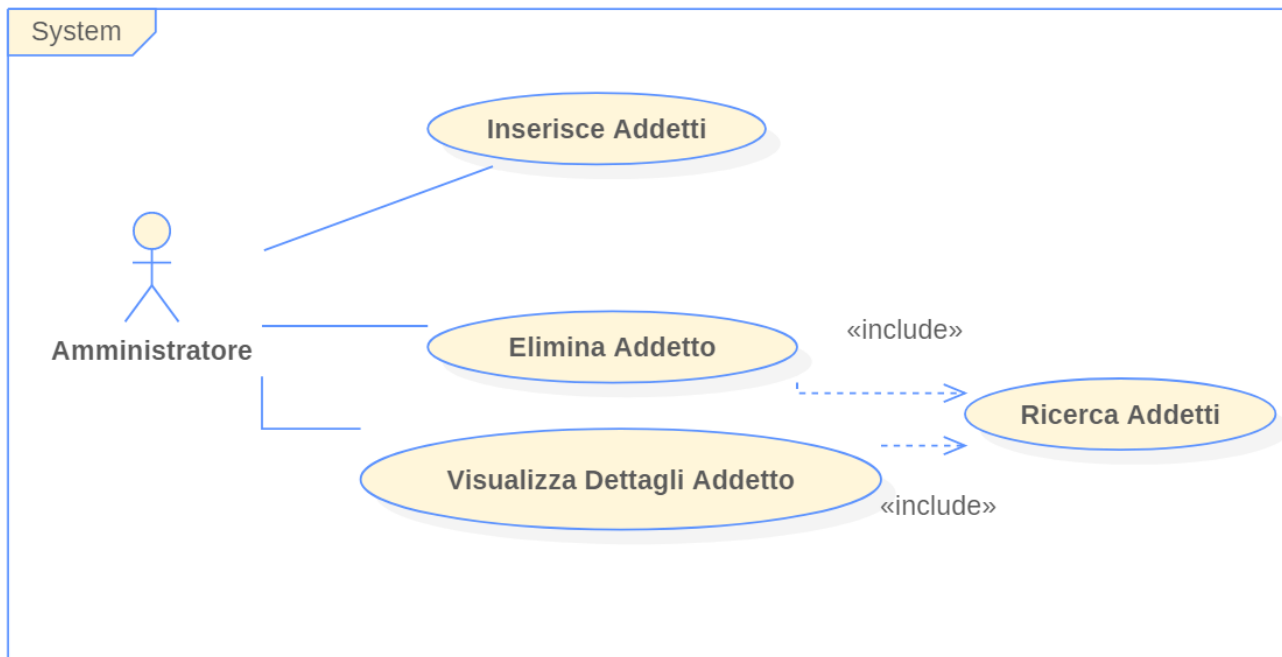


Gestione degli Addetti alla Sicurezza

Illustriamo ora la parte relativa alla gestione degli **Addetti alla Sicurezza** (da ora in poi **Addetti**), da parte di un **Amministratore** (Vedere Glossario), tramite il seguente Use Case Diagram:



Le funzionalità principali accessibili dall'amministratore sono le seguenti:

- **Inserimento** di un nuovo addetto: l'**Amministratore** può accedere a questa funzionalità cliccando il bottone **Nuovo** nella finestra di **Gestione Addetti**. L'interfaccia mostrerà un form in cui è possibile inserire i dati necessari alla memorizzazione nel Database del nuovo addetto.
- **Ricerca** degli addetti: la schermata di **Ricerca Addetti** è la prima che viene visualizzata quando viene selezionata la funzionalità di **Gestione Addetti**. La prima volta che viene avviata essa visualizza i dati principali di tutti gli addetti presenti nel Database. L'amministratore può filtrare questi dati utilizzando una qualsiasi combinazione di **Nome**, **Cognome**, o **Data di nascita** in modo da facilitare la ricerca dell'addetto desiderato.
- **Visualizzazione dettagli** di un addetto: è possibile visualizzare tutti i dettagli di un addetto selezionando l'addetto desiderato dalla schermata di ricerca, e cliccando il bottone **Visualizza Dati**. Verrà mostrata una nuova finestra con informazioni aggiuntive sull'addetto selezionato.
- **Eliminazione** di un addetto: sempre a partire dalla schermata di ricerca, si possono eliminare dal Database le informazioni di un addetto selezionandolo e cliccando su **Elimina**. N.B: in questo modo verranno eliminate dal Database solo le informazioni relative all'impiego dell'addetto; le informazioni relative alla **Persona** verranno conservate.

CockBurn:

Ora dettagliamo l'interazione tra **Amministratore** e **Sistema** tramite le seguenti tabelle di Cockburn, aiutandoci con l'utilizzo di alcuni mockup dell'interfaccia.

Annulla

Cerca

▼ Nome Dipendente	▼ Cognome Dipendente	▼ Codice Fiscale
admin	admin	ABCDEFGHI

Visualizza

Nuovo

Elimina

Mockup_RicercaAddetti

Annulla

Inserisci

Mockup_NuovoAddetto

Nome

Telefono

Cognome

e-mail

Data di Nascita

IBAN

Codice Fiscale

Stipendio

Indietro

Mockup_DettagliAddetto



L'addetto è stato inserito con successo

OK

Mockup_AddettoInserito



L'addetto è stato eliminato con successo

OK

Mockup_AddettoEliminato



Dato non valido

OK

Mockup_AlertDatiAddetto



L'addetto è già presente nel Database

OK

Mockup_AlertAddettoEsistente



Non è stato selezionato nessun addetto

OK

Mockup_AlertSelezioneAddetto

CockBurn per lo Use Case : Ricerca Addetti

USE CASE #1	<i>Ricerca Addetti</i>		
Goal in Context	L'amministratore vuole ricercare uno o più addetti alla sicurezza presenti nel Database.		
Preconditions	L'amministratore deve essere loggato nel sistema.		
Success End Condition	Gli addetti corrispondenti vengono visualizzati in una tabella.		
Failed End Condition	I dati inseriti non corrispondono a nessun utente. La tabella viene visualizzata vuota.		
Primary Actor	Amministratore		
Trigger	Amministratore preme il bottone <i>Gestione Addetti</i> dal menu principale		
DESCRIPTION	Step	Attore 1	Sistema
	1	Preme il bottone Gestione Addetti (<i>MockUp_MenuPrincipale</i>)	
	2		Mostra schermata <i>MockUp_RicercaAddetti</i>
	3		Compila la tabella con la lista di tutti gli addetti presenti nel Database.
SUBVARIATION #1	Step	Attore 1	Sistema
	3.1	Inserisce 'Nome' nell'apposito campo (<i>opzionale</i>)	
	4.1	Inserisce 'Cognome' nell'apposito campo (<i>opzionale</i>)	
	5.1	Inserisce 'Data Nascita' nell'apposito campo (<i>opzionale</i>)	
	6.1	Preme il bottone <i>Cerca</i>	
	7.1		Mostra la tabella con gli addetti corrispondenti ai criteri di ricerca inseriti.
SUBVARIATION #2	Step	Attore 1	Sistema
	6.1.1	Preme il bottone <i>Annulla</i>	
	7.1.1		Svuota i campi <i>Nome</i> , <i>Cognome</i> , <i>Data Nascita</i>

CockBurn per lo Use Case : ' Elimina Addetto'

USE CASE #2	<i>Elimina Addetto</i>		
Goal in Context	L'amministratore vuole eliminare un addetto presente nel database.		
Scope & Level			
Preconditions	<ul style="list-style-type: none"> • L'amministratore deve essere loggato. • Deve essere stato effettuato lo Use Case <i>Ricerca Addetti</i>. 		
Success End Condition	L'addetto selezionato viene eliminato dal Database.		
Failed End Condition	L'amministratore non seleziona nessun addetto.		
Primary Actor	Amministratore		
Trigger	L'amministratore preme il bottone <i>Elimina</i> da <i>Mockup_RicercaAddetti</i>		
DESCRIPTION	Step	Attore 1	Sistema
	1	Seleziona un addetto dalla tabella (<i>Mockup_RicercaAddetti</i>)	
	2	Preme il bottone 'Elimina' (<i>Mockup_RicercaAddetti</i>)	
	3		Mostra <i>Mockup_AddettoEliminato</i>
	4	Preme il bottone 'OK' (<i>Mockup_AddettoEliminato</i>)	
	5		Mostra <i>Mockup_RicercaAddetti</i>
SUBVARIATION #1	1.1	Preme il bottone 'Elimina' senza aver selezionato un addetto (<i>Mockup_RicercaAddetti</i>)	
	2.1		Mostra <i>Mockup_AlertSelezioneAddetto</i>

CockBurn per lo Use Case: 'Visualizza Dettagli Addetto'

USE CASE #3	<i>Visualizza Dettagli Addetto</i>		
Goal in Context	L'amministratore vuole visualizzare tutti i dettagli memorizzati nel Database relativi ad un addetto.		
Scope & Level			
Preconditions	<ul style="list-style-type: none"> L'amministratore deve essere loggato. <i>Deve essere stato effettuato lo Use Case Ricerca Addetti.</i> 		
Success End Condition	Vengono visualizzati i dati relativi all'addetto selezionato.		
Failed End Condition	L'amministratore non seleziona nessun addetto.		
Primary Actor	Amministratore		
Trigger	L'amministratore preme il bottone <i>Visualizza</i> in <i>Mockup_RicercaAddetti</i>		
DESCRIPTION	Step	Attore 1	Sistema
	1	Seleziona un addetto dalla tabella <i>(Mockup_RicercaAddetti)</i>	
	2	Preme il bottone 'Visualizza' <i>(Mockup_RicercaAddetti)</i>	
	3		Mostra <i>Mockup_DettagliAddetto</i>
	4	Preme il bottone 'Indietro' <i>(Mockup_DettagliAddetto)</i>	
	5		Mostra <i>Mockup_RicercaAddetto</i>
SUBVARIATION #1	Step	Attore 1	Sistema
	1.1	Preme il bottone 'Visualizza' senza aver selezionato un addetto <i>(Mockup_RicercaAddetto)</i>	
	2.1		Mostra <i>Mockup_AlertSelezioneAddetto</i>

CockBurn per lo Use Case: 'Inserisce Addetto'

USE CASE #4	<i>Inserisce Addetto</i>		
Goal in Context	L'amministratore vuole inserire un nuovo addetto nel sistema.		
Scope & Level			
Preconditions	<ul style="list-style-type: none">• L'amministratore deve essere loggato.• Il sistema deve mostrare la schermata Ricerca Addetti.		
Success End Condition	L'addetto viene correttamente inserito nel Database.		
Failed End Condition	<ul style="list-style-type: none">• L'amministratore inserisce dati errati• Un addetto con lo stesso Codice Fiscale è già presente nel Database		
Primary Actor	Amministratore		
Trigger	L'amministratore preme il bottone <i>Nuovo</i> in <i>Mockup_RicercaAddetti</i>		
DESCRIPTION	Step	Attore 1	Sistema
	1	Preme il bottone 'Nuovo' (<i>Mockup_RicercaAddetti</i>)	
	2		Mostra <i>Mockup_NuovoAddetto</i>
	3	Inserisce tutti i dati nel form	
	4	Preme il bottone 'Inserisci'	
	5		Mostra <i>Mockup_AddettoInserito</i>
SUBVARIATION #1	Step	Attore 1	Sistema
	3.1	Inserisce tutti i dati nel form, ma un campo è vuoto o errato	
	4.1	Preme il bottone 'Inserisci'	
			Mostra <i>Mockup_AlertDatiAddetto</i>
SUBVARIATION #2	Step	Attore 1	Sistema
	3.2	Inserisce correttamente i dati nel form, ma nel Database è già presente un addetto con lo stesso Codice Fiscale	
	4.2	Preme il bottone 'Inserisci'	
	5.2		Mostra <i>Mockup_AlertAddettoEsistente</i>

SUBVARIATION #3	Step	Attore 1	Sistema
	3.3	Preme il bottone 'Annulla'	
	4.3		Mostra <i>Mockup_RicercaAddetto</i>

Modelli di Dominio

Gestione degli Addetti

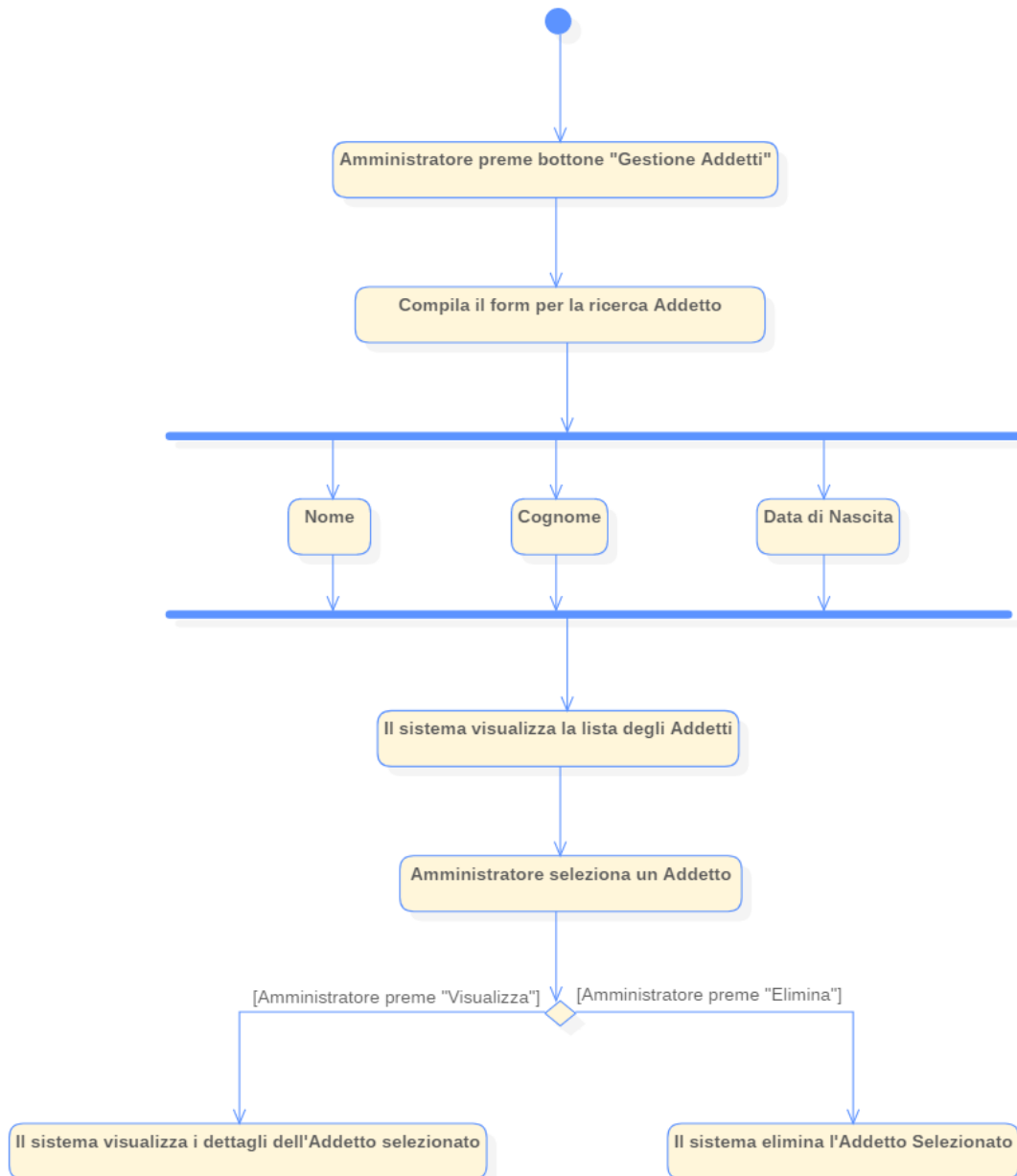
L'approccio usato per questa funzionalità è sempre quello Three-Object-Type. La principale classe **Entity** in questo caso è *Addetto*. Le classi di tipo **Boundary** e **Control** sono diverse a seconda della sottofunzionalità specifica.

Questa funzionalità, infatti, è stata suddivisa in:

- **Inserimento Addetto**
- **Ricerca Addetti**
- **Eliminazione Addetto**
- **Visualizzazione Dettagli Addetto**

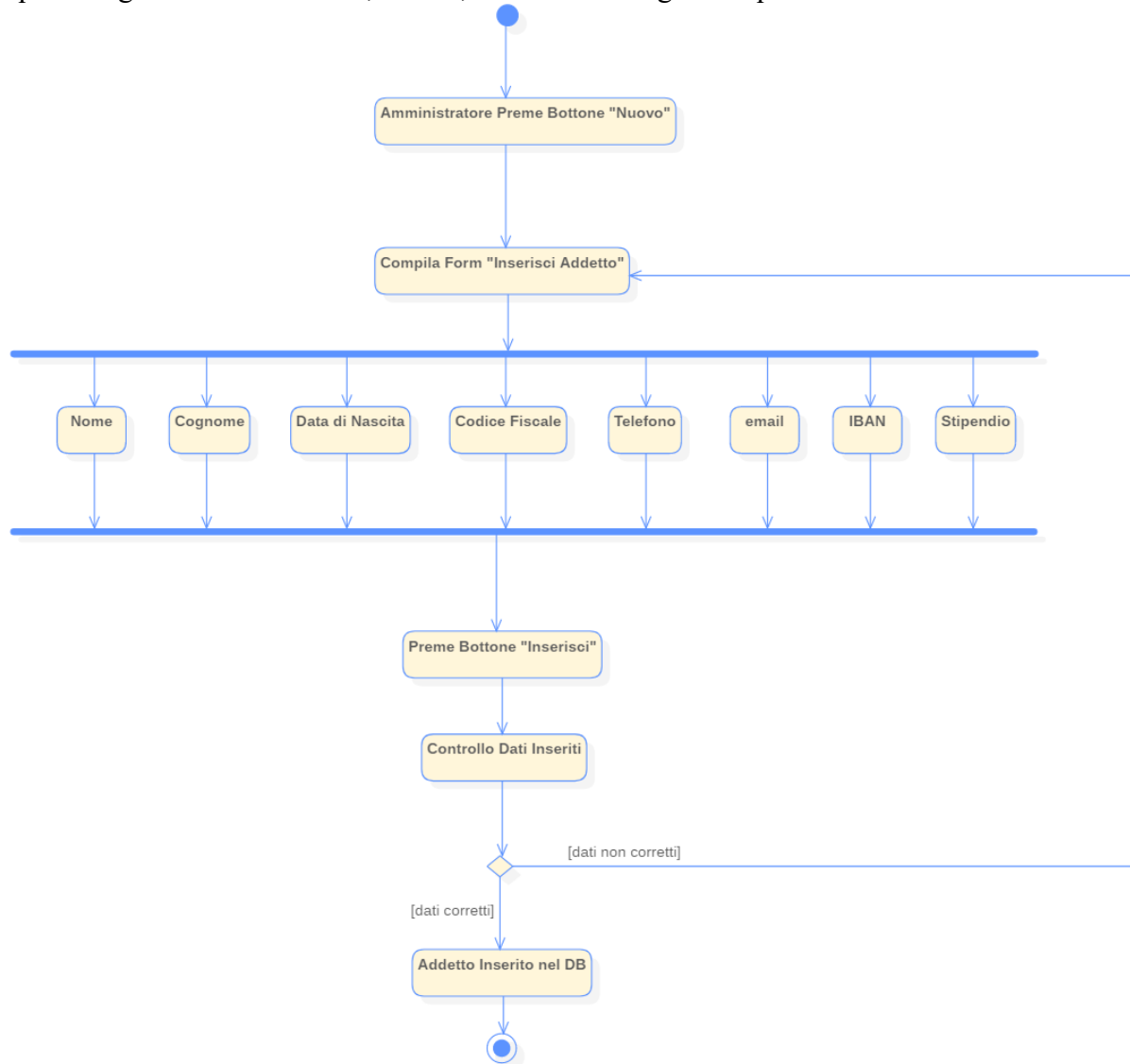
Le ultime 3 funzionalità elencate verranno gestite dalla stessa interfaccia (*Ricerca Cliente*), come evidenziato nelle tabelle di Cockburn precedenti.

Il flusso delle attività per queste 3 funzioni, verrà gestito come mostrato nel seguente Activity Diagram:



Nel quale la ricerca viene completata nel passo *“Il sistema visualizza la lista degli Addetti”*, mentre gli ultimi 2 passi corrispondono alle funzionalità di Visualizzazione e Eliminazione. L'inserimento dei dati *Nome*, *Cognome*, e *Data di Nascita* è opzionale. Nel caso in cui non vengano inseriti, verranno visualizzati tutti gli addetti presenti nel sistema.

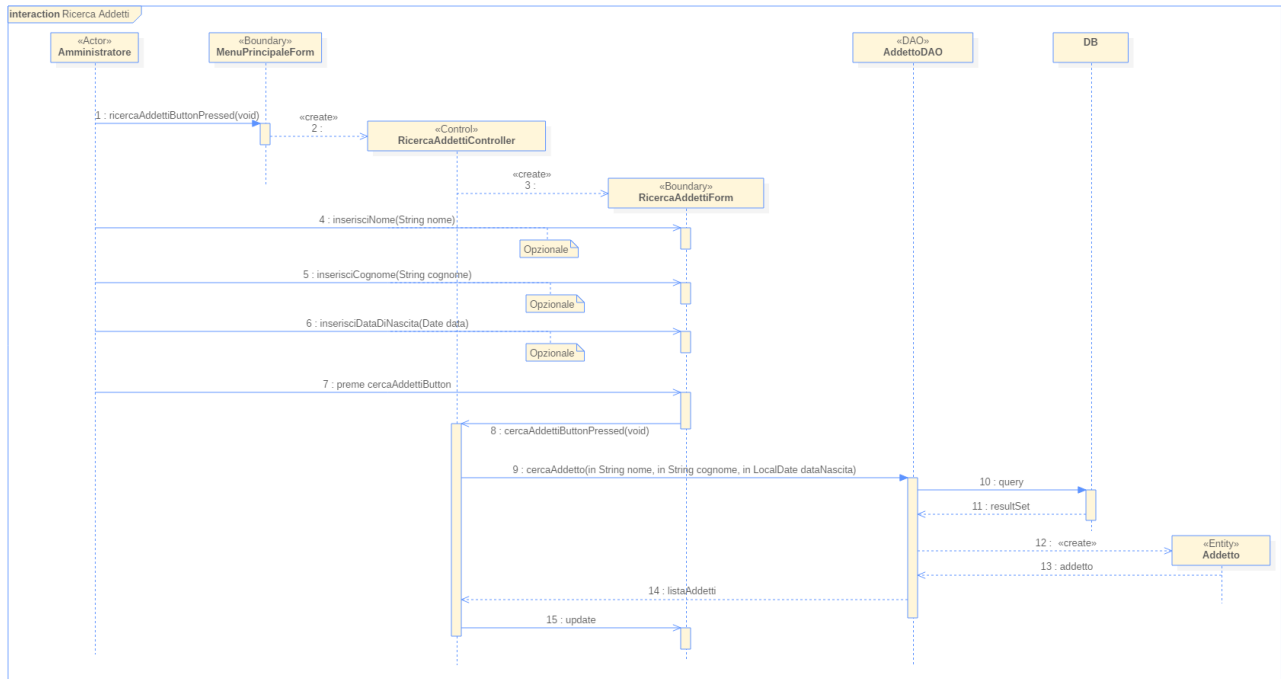
Per quanto riguarda l'Inserimento, invece, il flusso si svolgerà in questo modo:



Passiamo all'analisi delle interazioni per ogni sottofunzionalità.

Ricerca Addetto

Presentiamo per primo il Sequence Diagram derivato dal flusso mostrato in precedenza della funzione di Ricerca Addetto, in quanto come detto in precedenza, da questa interfaccia è poi possibile accedere alle altre funzionalità.



Sommario delle classe usate:

- *MenuPrincipaleForm*: Già presentato in precedenza.
- *RicercaAddettiForm*: Classe di tipo **Boundary**, che rappresenta l'interfaccia con cui interagirà l'**Attore**. In questa interfaccia sarà possibile inserire i dati (nome, cognome, data di nascita) per eseguire la ricerca degli addetti corrispondenti nel Database. I risultati della ricerca saranno mostrati in una tabella. Da questa interfaccia, inoltre, sarà possibile accedere alle funzionalità di inserimento, visualizzazione e eliminazione tramite appositi bottoni. Questa classe comunica al **Controller** quali azioni vengono eseguite dall'utente.
- *RicercaAddettiController*: Classe **Control**, si occupa di chiamare le funzioni richieste dall'utente alle classi corrispondenti. Nel diagramma mostrato, comunica alla classe **DAO** i dati inseriti per la ricerca, ma nei diagrammi che seguiranno vedremo che la classe si occuperà anche di creare le interfacce per le altre funzionalità presenti, oltre a comunicare sempre con la classe **DAO** per inserimento ed eliminazione.
- *AddettoDAO*: Classe **DAO**, comunica solo con il **Control** e l'implementazione della basi di dati scelta.
- *Addetto*: Classe **Entity**, modella l'addetto.

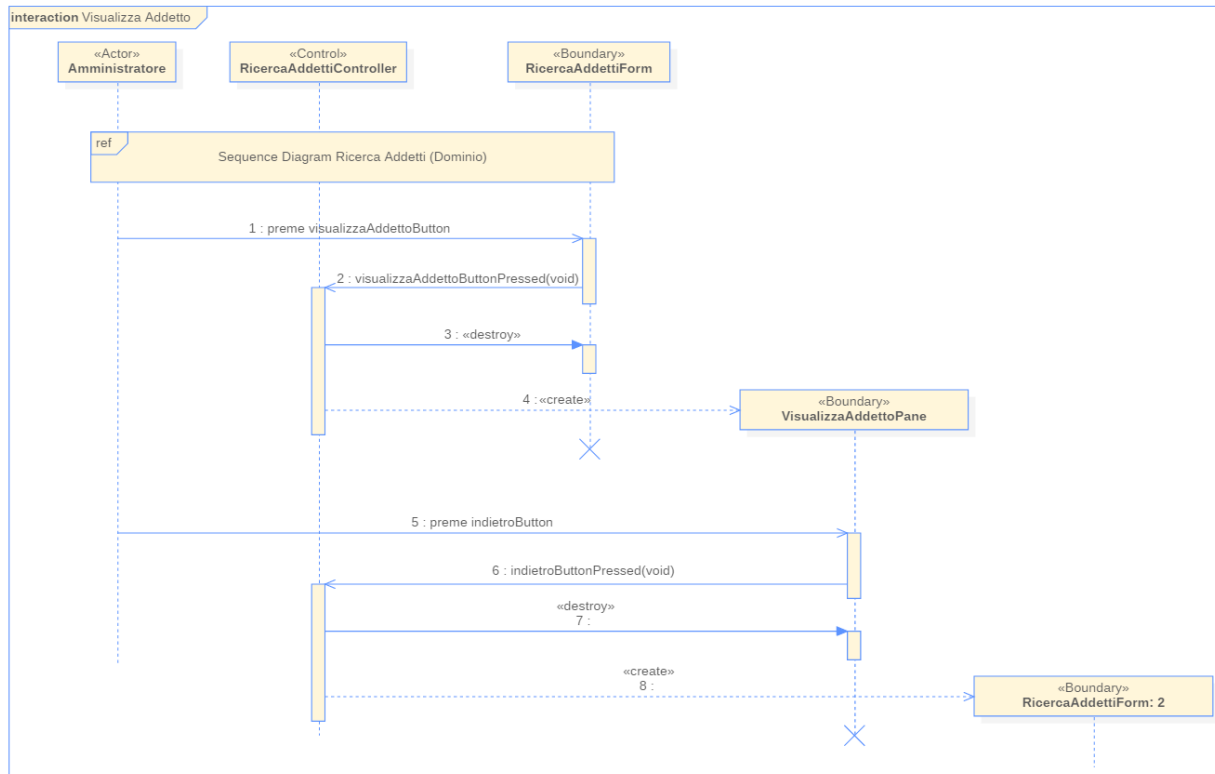
Vincoli:

- Prima di poter modificare, eliminare o visualizzare un addetto, l'Attore deve averlo selezionato dall'apposita lista di addetti.

Mostreremo il Class Diagram subito dopo aver presentato la funzionalità di **Visualizzazione Dettagli**, in quanto le classi per entrambi i casi d'uso possono essere rappresentate in uno stesso diagramma.

Visualizzazione Dettagli Addetto

Si potrà accedere alla funzionalità di **Visualizzazione Dettagli** dall'interfaccia di **Ricerca**, cliccando il tasto corrispondente dopo aver selezionato l'**Addetto** di cui si vogliono visualizzare i dati. Il controller mostrerà una nuova finestra in cui saranno visualizzati nel dettaglio i dati dell'addetto presenti nel Database. Il Sequence Diagram che segue, dunque, richiede che venga eseguito prima quello mostrato nel punto precedente, come evidenziato dal riferimento inserito:

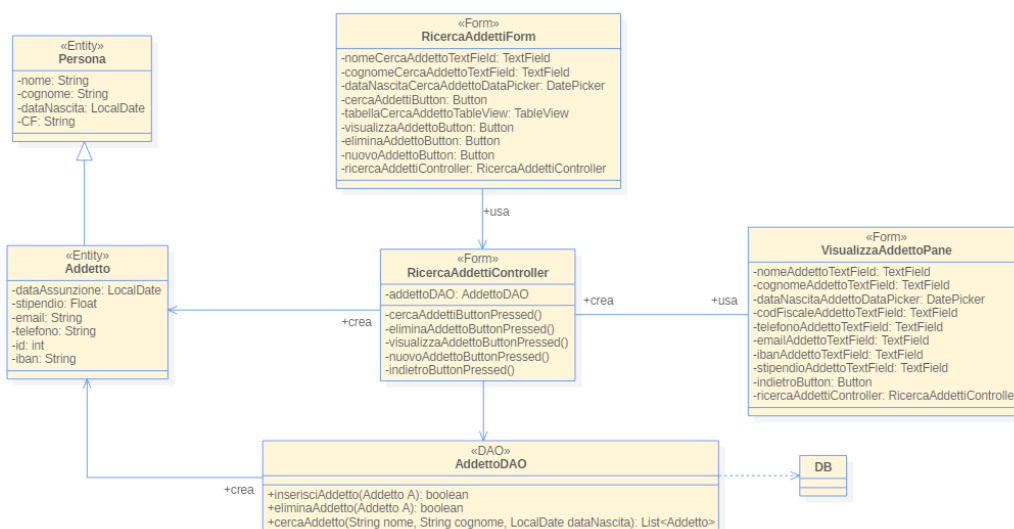


Sommario delle classe usate:

- *VisualizzaAddettoPane*: Classe di tipo **Boundary**, che mostrerà le informazioni complete dell'addetto selezionato. L'unica altra funzione presente in questa interfaccia sarà quella di poter tornare indietro al menu di Ricerca.

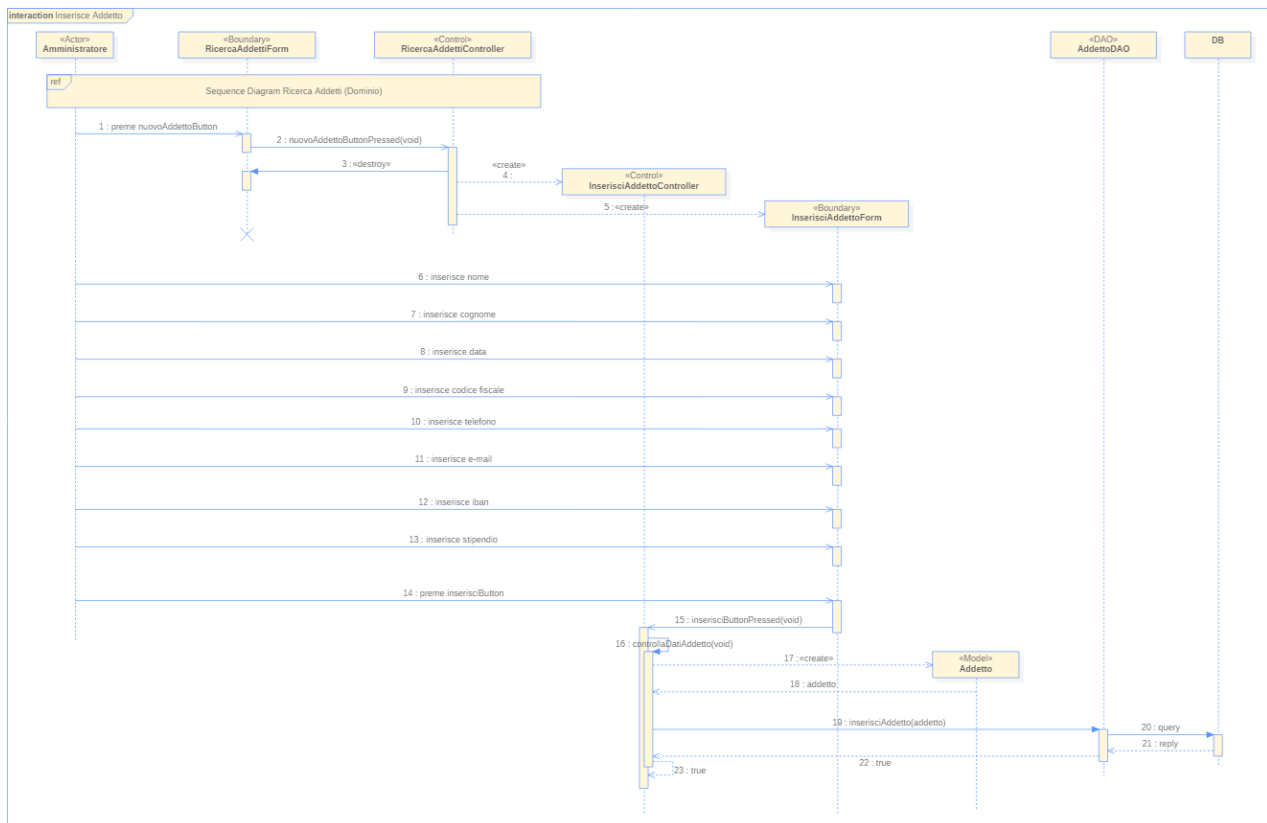
Le rimanenti classi sono già state descritte nel punto precedente.

Possiamo ora passare alla presentazione del Class Diagram relativo alle funzioni di **Ricerca** e **Visualizzazione Dettagli**:



Inserimento Addetto

Si potrà accedere alla funzionalità di **Inserimento** cliccando l'apposito tasto nell'interfaccia di **Ricerca**.



Sommario delle classe usate:

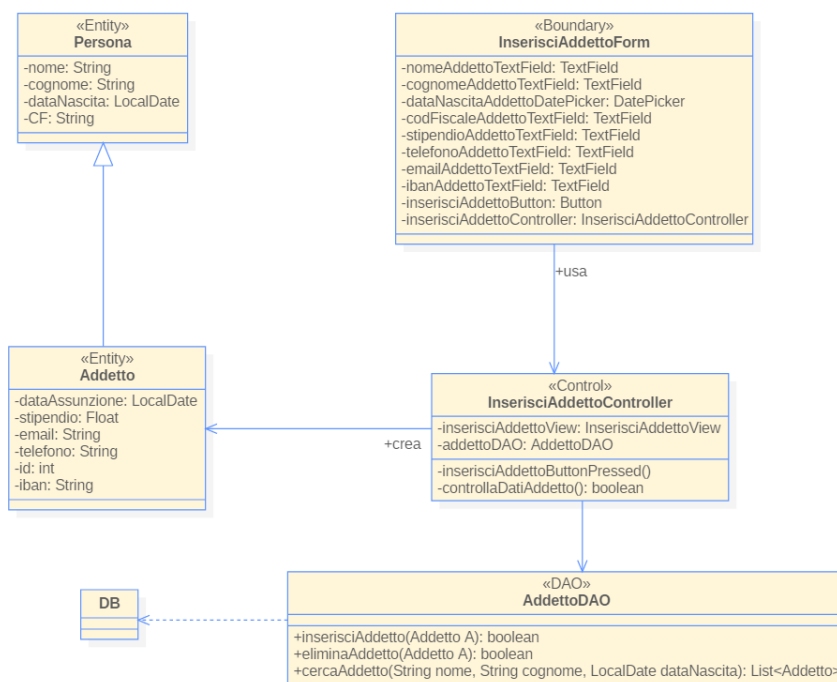
- *InserisciAddettoForm*: Classe di tipo **Boundary**, che rappresenta l'interfaccia con cui interagirà l'**Attore**. In questa interfaccia sarà possibile inserire i dati (nome, cognome, data di nascita, codice fiscale, telefono, e-mail, iban, stipendio) relativi all'addetto da inserire, e di inoltrarli al Database. L'unica altra funzione presente in questa interfaccia sarà quella di poter tornare indietro al menu di Ricerca. Questa classe comunica al **Controller** quali azioni vengono eseguite dall'utente.
- *InserisciAddettoController*: Classe **Control**, controllerà la correttezza dei dati secondo i vincoli che verranno successivamente descritti in dettaglio, e creerà l'addetto con i dati correnti, comunicando con la classe **DAO**.

Le rimanenti classi sono già state descritte nel punto precedente.

Vincoli:

- Prima di poter eliminare o visualizzare un addetto, l'Attore deve averlo selezionato dall'apposita lista di addetti.

Mostriamo le classi trovate in un Class Diagram:



Eliminazione Addetto

Per questa funzione, ci limitiamo a presentare il relativo Sequence Diagram con le interazioni, in quanto le classi utilizzate sono le stesse dei punti precedenti.

Si potrà accedere alla funzionalità di **Eliminazione** dall'interfaccia di **Ricerca**, cliccando il tasto corrispondente dopo aver selezionato l'**Addetto** da eliminare. Anche in questo caso, quindi, sarà necessario eseguire prima la funzione di ricerca.

Presentiamo il Sequence Diagram:

