

# PROGETTAZIONE

## AGROLUDOS

Versione 1.0

Data di rilascio: 3 aprile 2015

INGEGNERIA DEL SOFTWARE A.A. 2013-2014

**Realizzato da**

D'Alessandro Raffaele 611054 ITPS

[Dalessandroraaffaele1993@gmail.com](mailto:Dalessandroraaffaele1993@gmail.com)

Dispoto Marco 608753 I.T.P.S

[m.dispoto@studenti.uniba.it](mailto:m.dispoto@studenti.uniba.it)

---

## INDICE

---

<b>INDICE .....</b>	<b>2</b>
<b>1. ARCHITETTURA .....</b>	<b>3</b>
1.1 LIVELLI ARCHITETTURALI .....	3
1.2 DIAGRAMMA DELLE COMPONENTI .....	4
1.3 DIAGRAMMA DI CONFIGURAZIONE .....	8
<b>2. PROGETTO DI DETTAGLIO .....</b>	<b>9</b>
2.1 DIAGRAMMA DELLE CLASSI .....	9
2.2 SPECIFICHE DELLE CLASSI .....	9
2.3 DIAGRAMMI DI SEQUENZA .....	13
<b>3. PROGETTO DEI DATI .....</b>	<b>22</b>
3.1 DATABASE .....	22
3.1.1 <i>Diagramma delle Dipendenze dei Dati</i> .....	22
3.1.2 <i>Modello del Database</i> .....	23
3.1.3 <i>Dettaglio dei Dati</i> .....	23
3.2 FILE SYSTEM .....	26
3.2.1 <i>Grammatiche file XML</i> .....	26
3.2.2 <i>Altri file</i> .....	41
<b>4. APPENDICE .....</b>	<b>42</b>
4.1 PATTERNS UTILIZZATI .....	42
4.2 ALTRO .....	43

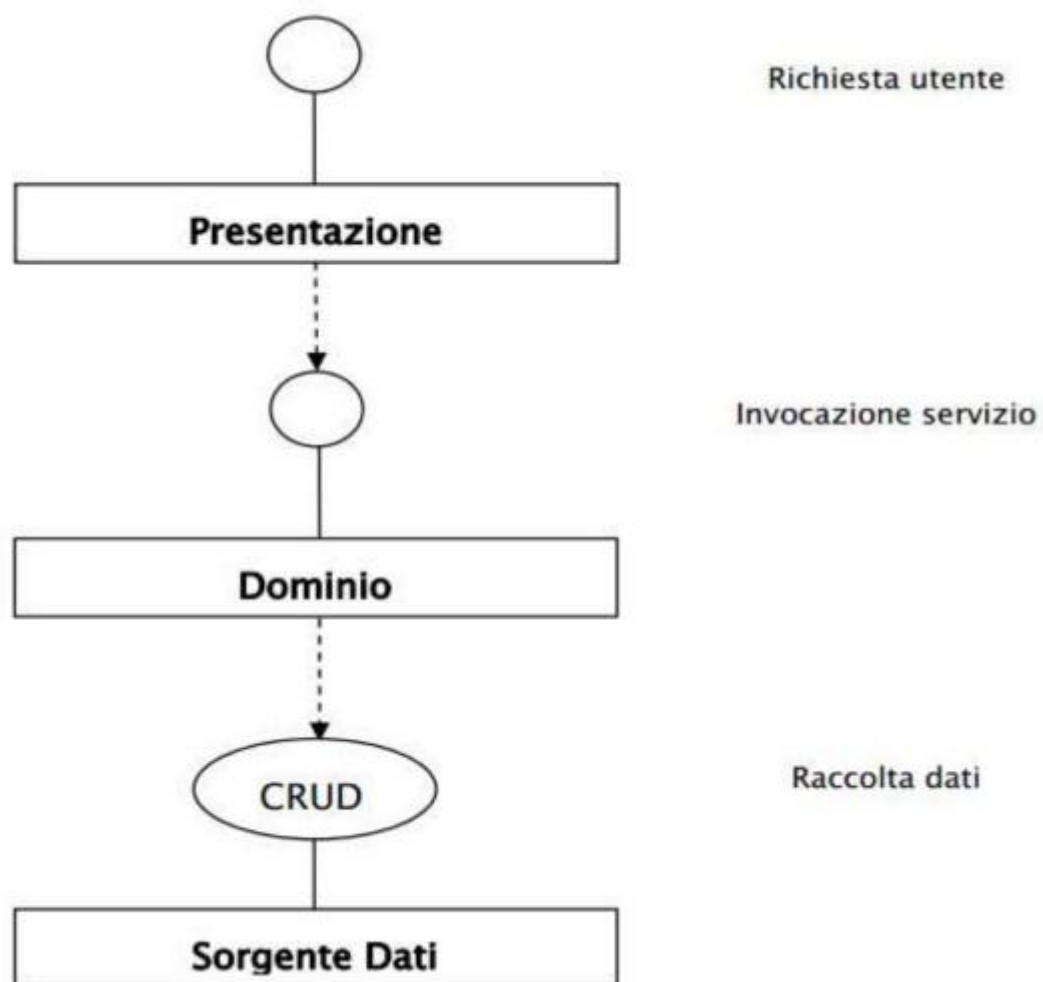


# 1. ARCHITETTURA

## 1.1 Livelli Architeturali

Come già specificato nell'analisi dei requisiti, il sistema Agroludos è caratterizzato da un'architettura Client-Server sviluppata su un sottosistema Server contenente il database, e un sottosistema Client contenente la presentazione e il dominio del sistema, quindi il programma utilizzabile dagli utenti.

Si è scelto di realizzare un'architettura a tre livelli:



### Livello di Presentazione:

Le componenti di questo livello si occupano della logica di presentazione e dell'intercettazione delle richieste da parte dei diversi client, invocando i relativi servizi di business; le componenti si occupano della gestione della navigazione tra le finestre del sistema e costruisce e restituisce le risposte al client.

### Livello di Dominio:

Le componenti di questo livello contengono la logica di business per l'esecuzione dei servizi del dominio e la gestione dell'accesso ai servizi del livello sottostante.

### Livello Sorgente dati (Server):

Le componenti di questo livello, lato Server, hanno accesso al database ed offrono le operazioni di creazione, lettura, aggiornamento e cancellazione (CRUD) del database.

## 1.2 Diagramma delle Componenti

### Componenti individuate per il livello di presentazione



- **Front controller**: gestisce le richieste provenienti dalle finestre, gestendo la navigazione fra esse e indirizzando le richieste di servizio al livello di dominio.
- **Application Controller**: riceve le richieste dal Front Controller, delega l'esecuzione del servizio al Business Delegate ed istanzia la view.

### Componenti individuate per il livello di dominio

- **Business Delegate**: riceve le richieste provenienti dall'Application Controller e le smista ed inoltra agli opportuni gestori;
- **Gestione Competizione**: esegue le operazioni di gestione delle competizioni (inserimento, modifica e annullamento competizioni;

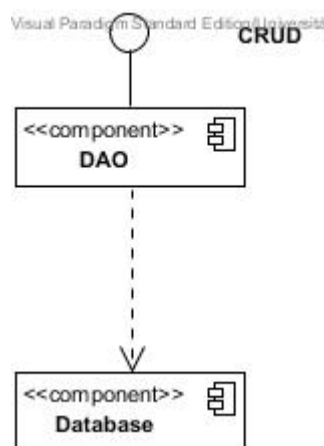


---

visualizzazione dati di competizioni; visualizzazione ed eliminazione dei partecipanti; inserimento ed annullamento tipi di competizione);

- **Gestione Partecipanti:** esegue le operazioni di gestione delle competizioni (iscrizione partecipante al sistema; gestione dell'account; iscrizione, modifica ed eliminazione di partecipanti da competizioni; visualizzazione di informazioni relative ad iscrizioni dei partecipanti alle competizioni);
- **Gestione Manager Competizione:** esegue le operazioni di gestione dei manager di competizione (iscrizione manager di competizione al sistema; gestione dell'account);
- **Gestione Optional:** esegue le operazioni di gestione degli optional (visualizzazione e modifica optional);
- **Gestione Utenti:** esegue le operazioni generali di gestione degli utenti (login al sistema; aggiornamento password e controllo di esistenza username nel sistema);

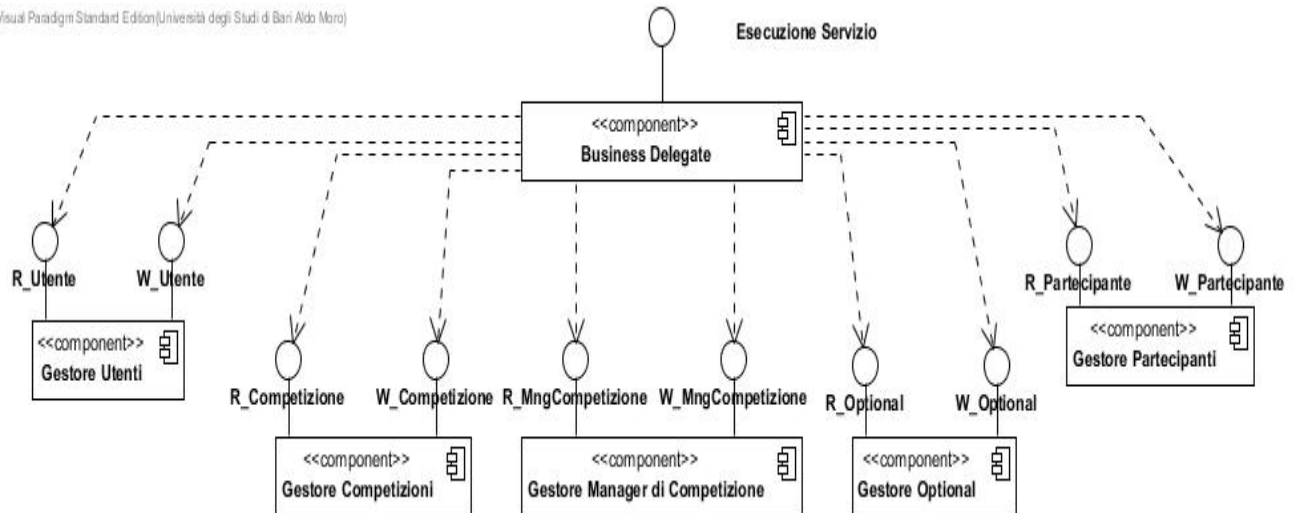
### Componenti individuate per il livello di sorgente dati



- **DAO:** consente l'isolamento tra lo strato del database e lo strato di dominio, offrendo a quest'ultimo un'unica interfaccia per eseguire operazioni di creazione, lettura, modifica ed eliminazione;
- **Database:** database del sistema Server.

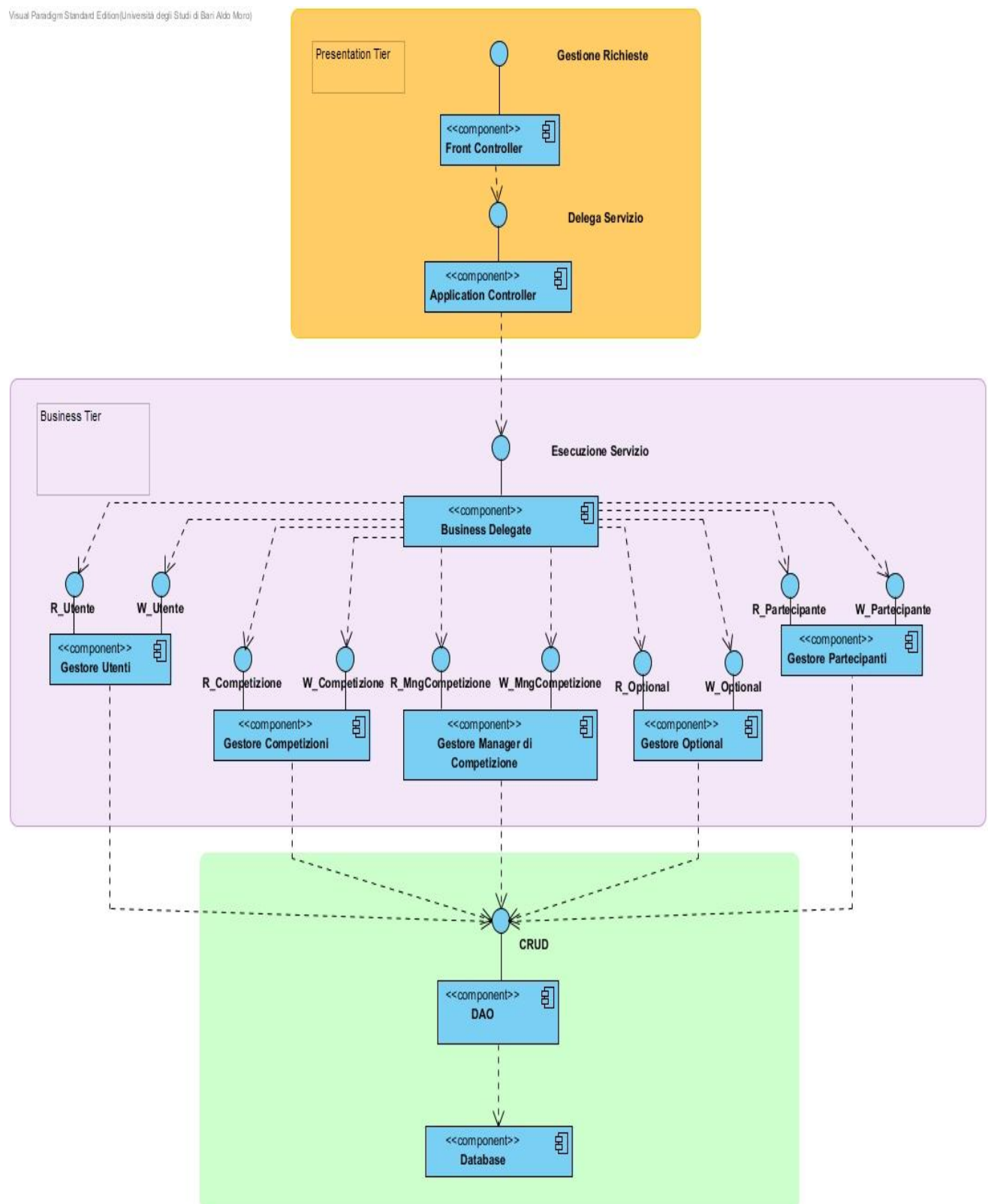
## Diagramma delle relazioni fra le componenti del livello di dominio

Visual Paradigm Standard Edition (Università degli Studi di Bari Aldo Moro)

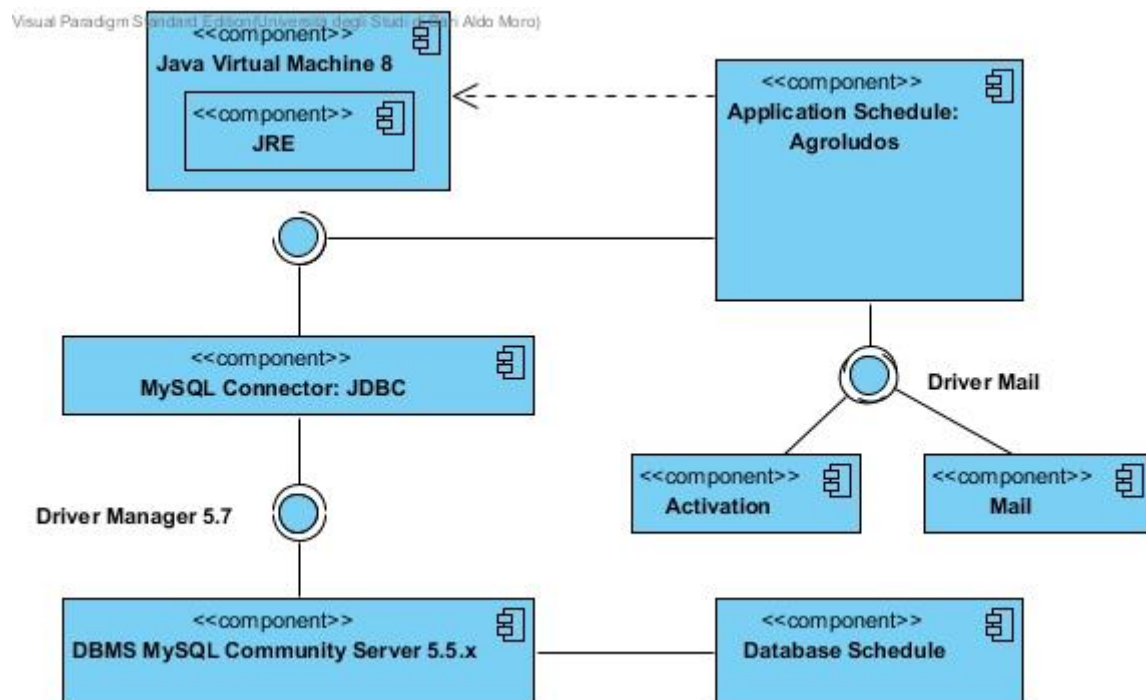


## Diagramma delle relazioni fra le componenti dei tre livelli

Visual Paradigm Standard Edition (Università degli Studi di Bari Aldo Moro)



## 1.3 Diagramma di Configurazione





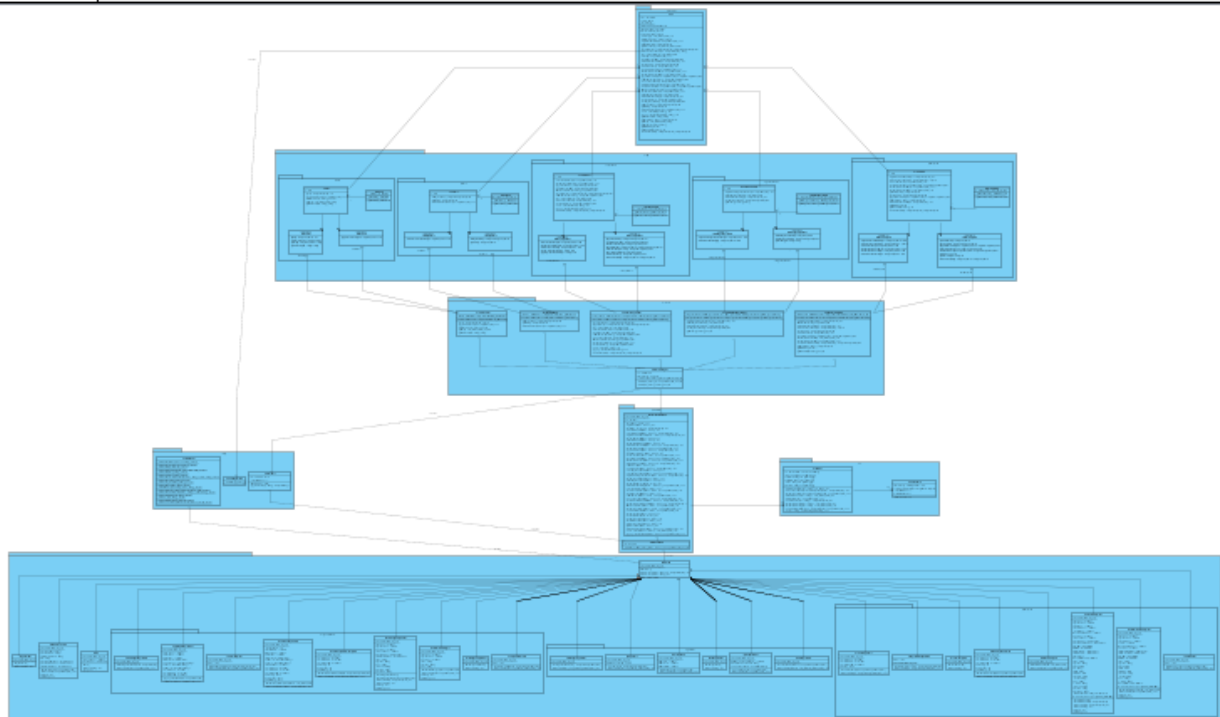
---

## 2. PROGETTO DI DETTAGLIO

---

### 2.1 Diagramma delle Classi

---



### 2.2 Specifiche delle Classi

Relazioni tra le classi a livello di presentazione:

- Tutte le classi della GUI sono in relazione “USA” con il Front Controller per effettuare una richiesta; non vi è nessun tipo di relazione diretta fra tali classi poiché le richieste dovranno essere gestite passando per il Front Controller;
- Il FrontController è in relazione “INVOCA” con l’Application Controller;
- L’ application Controller è in relazione “ISTANZIA” con tutte le Classi GUI.

Relazione tra le classi a livello di dominio

- Il Business Delegate è in relazione “INVOCA” con tutti gli Application Services che incapsulano la logica di business;



- 
- Ogni Application Service è in relazione “ISTANZIA” con la rispettiva classe di dominio.

#### Relazioni tra le classi ai tre livelli

- L'Application Controller è in relazione “DELEGA” con il Business Delegate per delegare l'esecuzione di un servizio;
- Tutte le classi di dominio sono in relazione di “USA” con l'interfaccia CRUD del DAO;
- Tutte le classi Entity sono in relazione “USA” con la classe Controllo.

#### Breve descrizione delle classi:

- **MainClass:** classe delegata all'avvio del sistema.
- **MainMenu:** classe GUI astratta dalla quale tutte le altre classi GUI ereditano. Contiene, oltre ai settaggi, il metodo per la chiamata al Front Controller.
- **Menu:** classe GUI, fornisce l'interfaccia grafica del menu principale.
- **Registrazione:** classe GUI, fornisce l'interfaccia grafica della scelta del tipo di registrazione (partecipante o manager di competizione).
- **ModificaPassword:** classe GUI, fornisce l'interfaccia grafica del form di modifica password.
- **AggiungiCompetizione:** classe GUI, fornisce l'interfaccia grafica del form per la creazione di una nuova competizione da parte di un manager di competizione.
- **CompetizioniGestite:** classe GUI, fornisce l'interfaccia grafica per mostrare al manager di competizione la lista delle competizioni da lui gestite.
- **ListaPartecipanti:** classe GUI, fornisce l'interfaccia grafica per mostrare al manager di competizione la lista dei partecipanti iscritti ad una data competizione.
- **MenuMngCompetizione:** classe GUI, fornisce l'interfaccia grafica del menu principale di un manager di competizione.



- 
- **ModificaCompetizione:** classe GUI, fornisce l'interfaccia grafica del form per la modifica di una data competizione da parte di un manager di competizione.
  - **ModificaDatiManager:** classe GUI, fornisce l'interfaccia grafica per la modifica del profilo del manager di competizione.
  - **ModificaOptionalPartecipante:** classe GUI, fornisce l'interfaccia grafica per la modifica degli optional scelti da un partecipante per una data competizione, da parte del manager di competizione.
  - **NuovoMngCompetizione:** classe GUI, fornisce l'interfaccia grafica per la registrazione di un nuovo manager di competizione.
  - **ProfiloMngCompetizione:** classe GUI, fornisce l'interfaccia grafica che mostra ad un manager di competizione i dati del suo profilo.
  - **MenuMngSistema:** classe GUI, fornisce l'interfaccia grafica che mostra il menu principale del manager di sistema.
  - **PrezzoOptional:** classe GUI, fornisce l'interfaccia grafica che dà la possibilità al manager di sistema di modificare i prezzi degli optional.
  - **TipiCompetizione:** classe GUI, fornisce l'interfaccia grafica che dà la possibilità al manager di sistema di aggiungere o eliminare un tipo di competizione.
  - **TipiOptional:** classe GUI, fornisce l'interfaccia grafica che dà la possibilità al manager di sistema di leggere i dettagli degli optional.
  - **VediCompetizioni:** classe GUI, fornisce l'interfaccia grafica che mostra al manager di sistema la lista di tutte le competizioni, in atto o concluse, in sola lettura.
  - **VediMngCompetizioni:** classe GUI, fornisce l'interfaccia grafica che mostra al manager di sistema la lista dei manager di competizione registrati al sistema.
  - **CompetizioniDisponibili:** classe GUI, fornisce l'interfaccia grafica che mostra ad un partecipante la lista delle competizioni alle quali è possibile iscriversi.



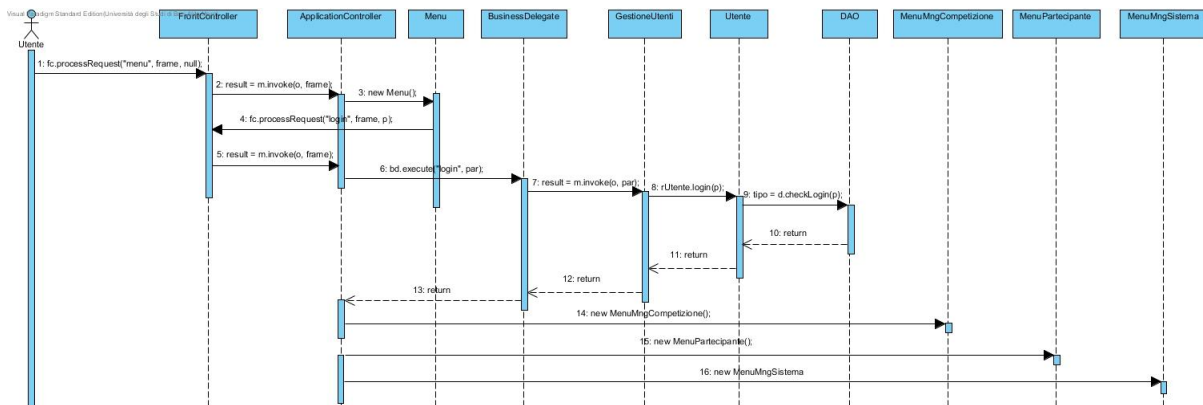
- 
- **ListaIscrizioni**: classe GUI, fornisce l'interfaccia grafica che mostra ad un partecipante la lista delle competizioni alla quale si è iscritto.
  - **MenuPartecipante**: classe GUI, fornisce l'interfaccia grafica che mostra il menu del partecipante.
  - **ModificaDatiPartecipante**: classe GUI, fornisce l'interfaccia grafica che dà la possibilità al partecipante di modificare il proprio profilo.
  - **ModificaOptionalScelti**: classe GUI, fornisce l'interfaccia grafica che dà la possibilità al partecipante di modificare gli optional selezionati per una data competizione.
  - **NuovoPartecipante**: classe GUI, fornisce l'interfaccia grafica che mostra il form di registrazione per un nuovo partecipante.
  - **ProfiloPartecipante**: classe GUI, fornisce l'interfaccia grafica che mostra ad un partecipante il suo profilo.
  - **SelezioneOptional**: classe GUI, fornisce l'interfaccia grafica che dà la possibilità ad un partecipante di selezionare gli optional per una data competizione.
  - **ControlloDati**: classe utile a realizzare controlli sui campi.
  - **DBCheck**: classe utile a realizzare controlli sul Database.
  - **DBSettingsReader**: classe utile a leggere le impostazioni del database da file di configurazione.
  - **XMLReader**: classe utile a leggere file XML.
  - **ReadConfMail**: classe utile a leggere il file di configurazione della casella di posta elettronica.
  - **SendMail**: classe che si occupa dell'invio di mail.



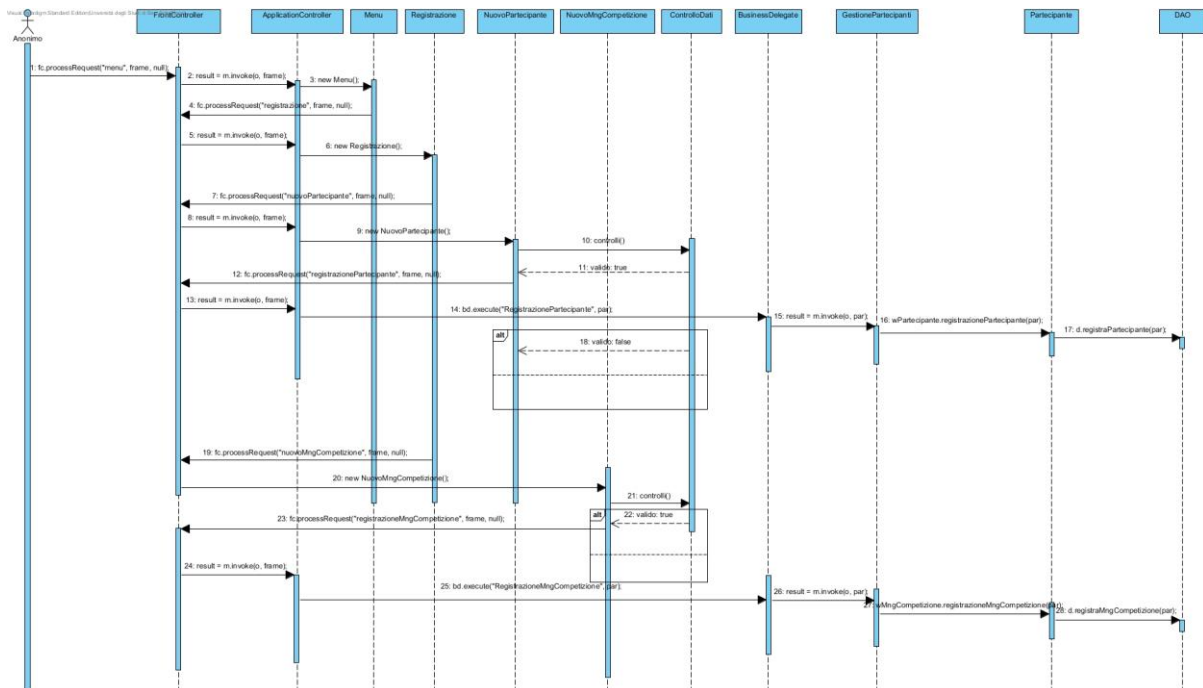
## 2.3 Diagrammi di Sequenza

### ANONIMO:

- Autenticazione

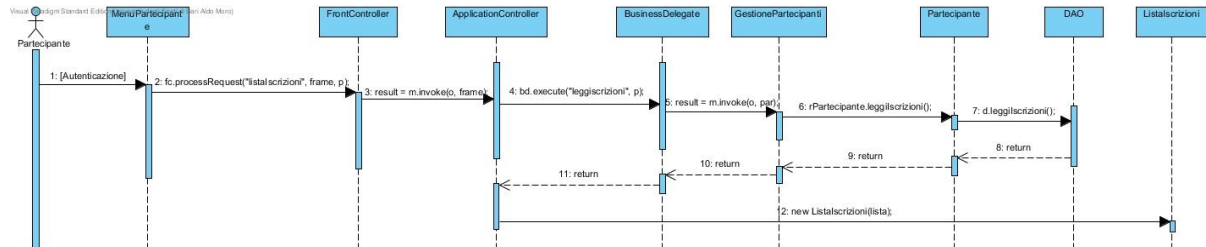


- Registrazione

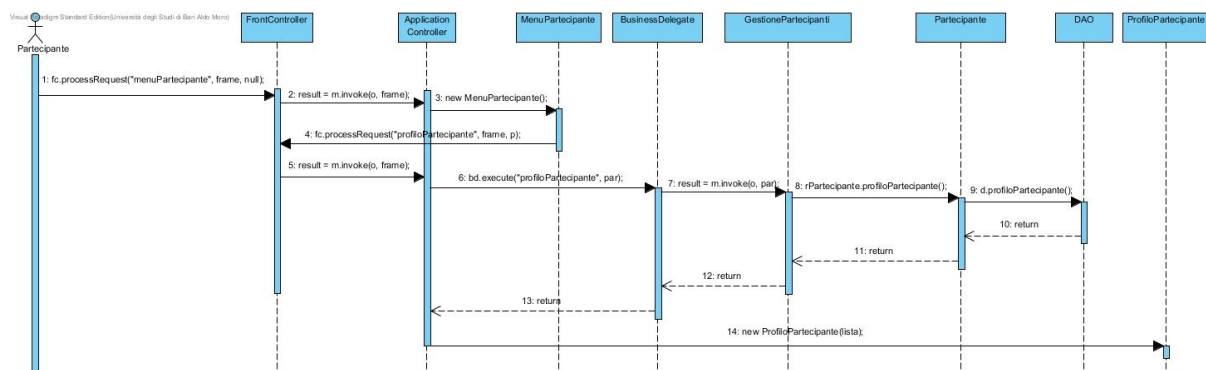


## PARTECIPANTE:

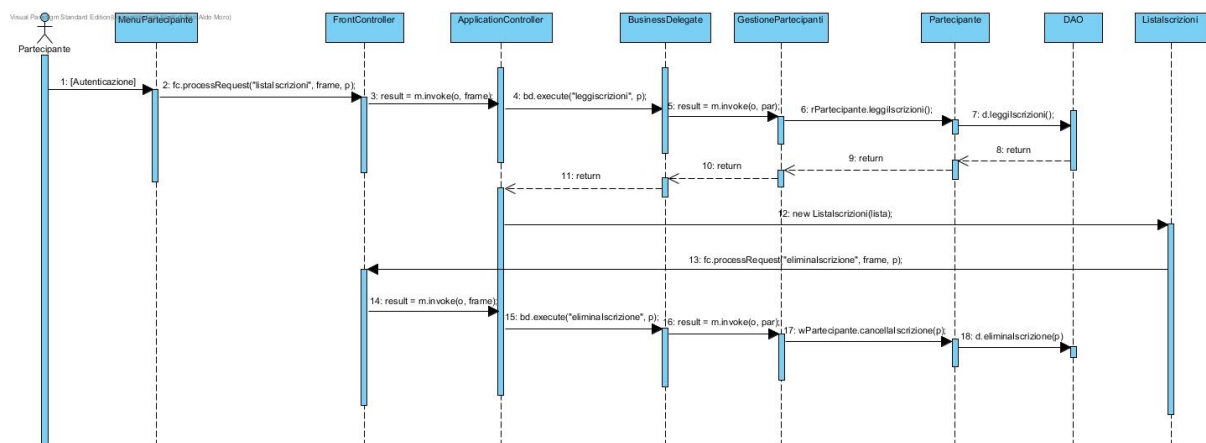
- Accesso a Iscrizioni



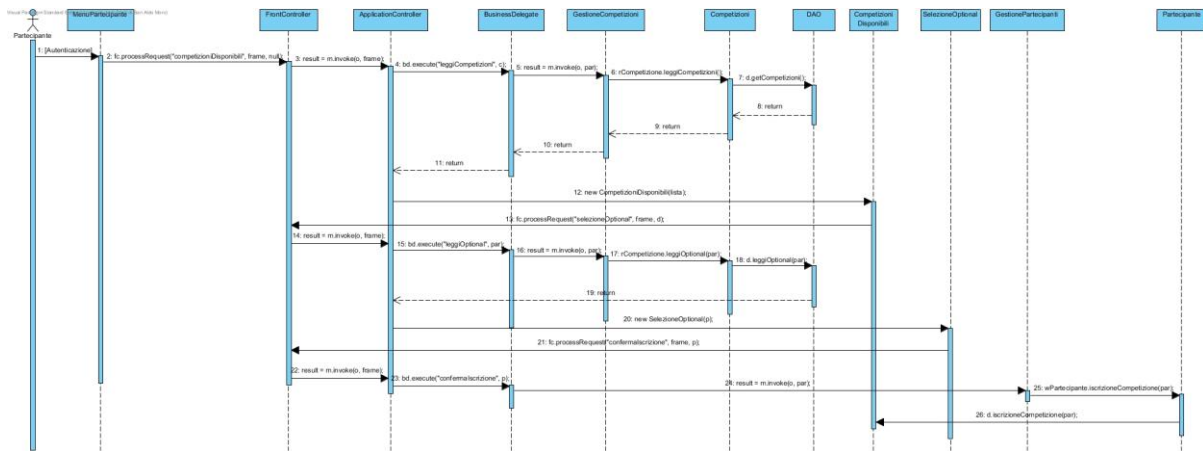
- Accesso Dati Personali



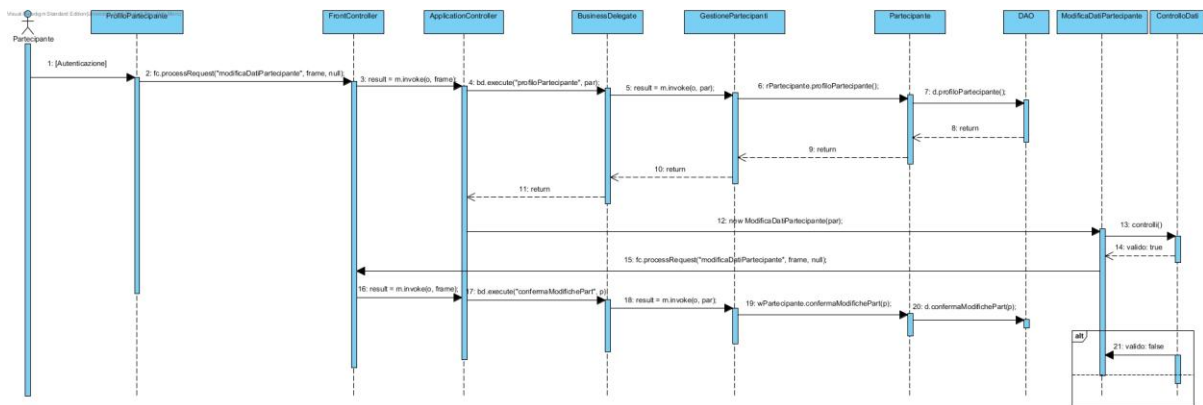
- Cancellazione da Competizione



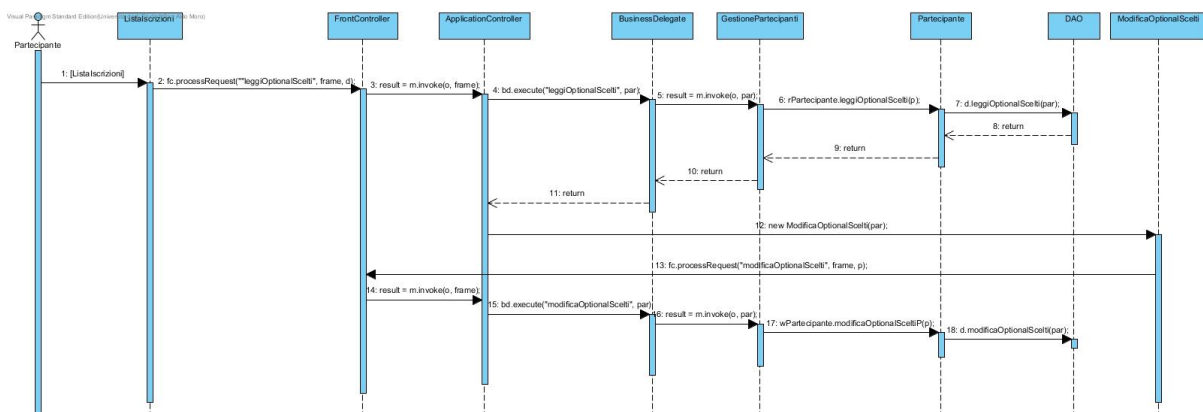
- Iscrizione a Competizione



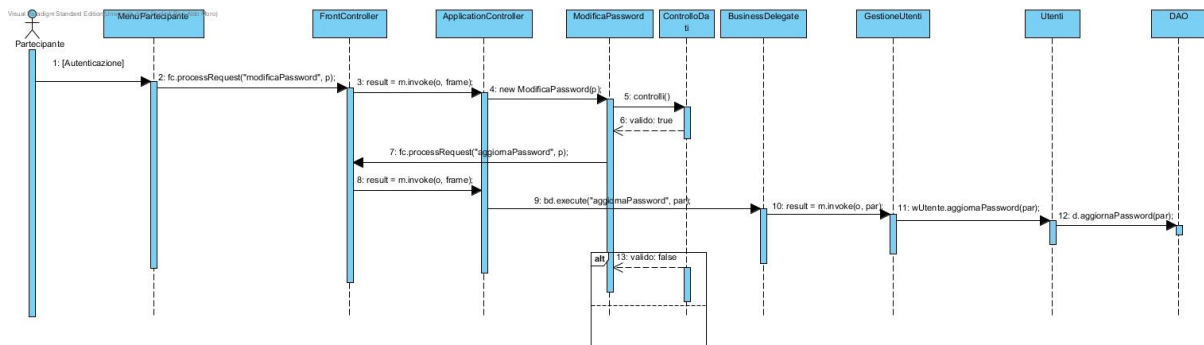
- Modifica Dati Personali



- Modifica Optional Scelti

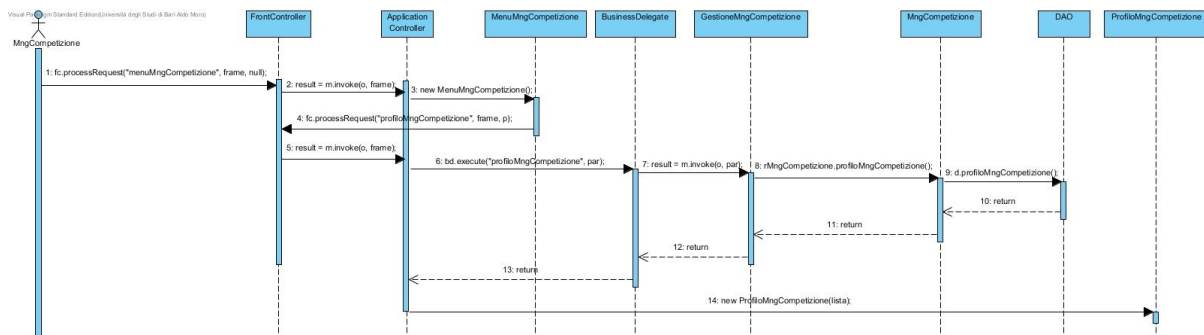


- Modifica Password

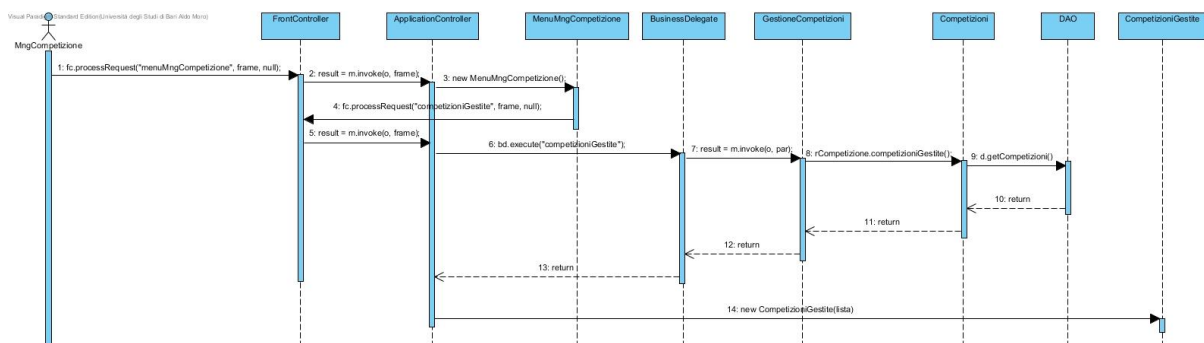


## MANAGER DI COMPETIZIONE

- Accesso Dati Personali

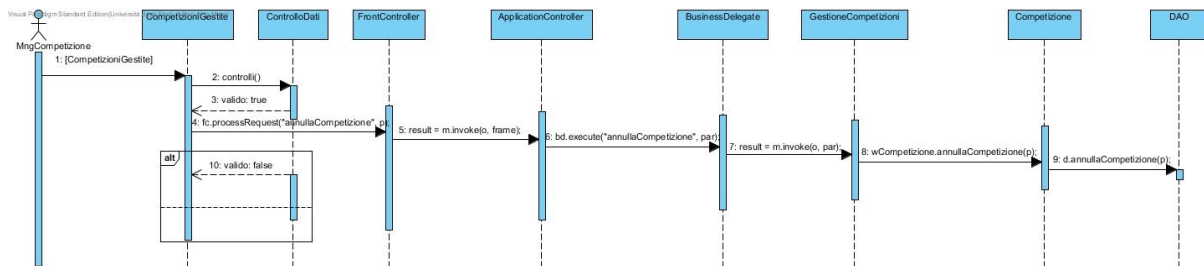


- Accesso Informazioni Competizioni

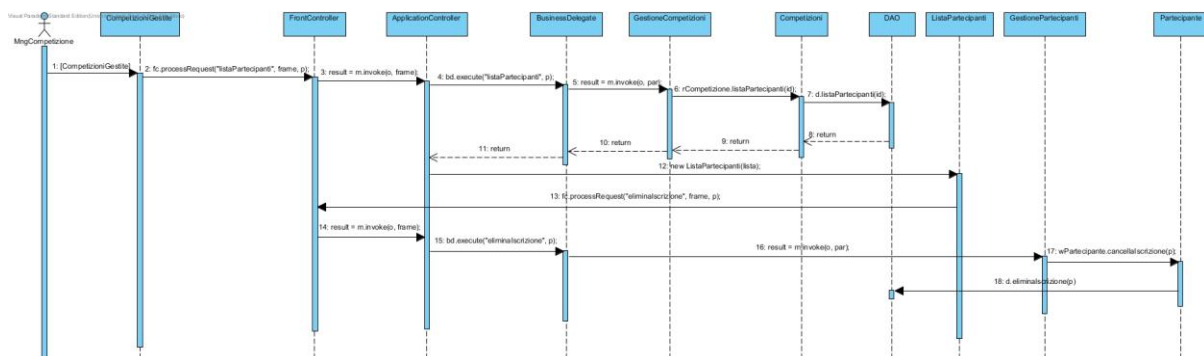




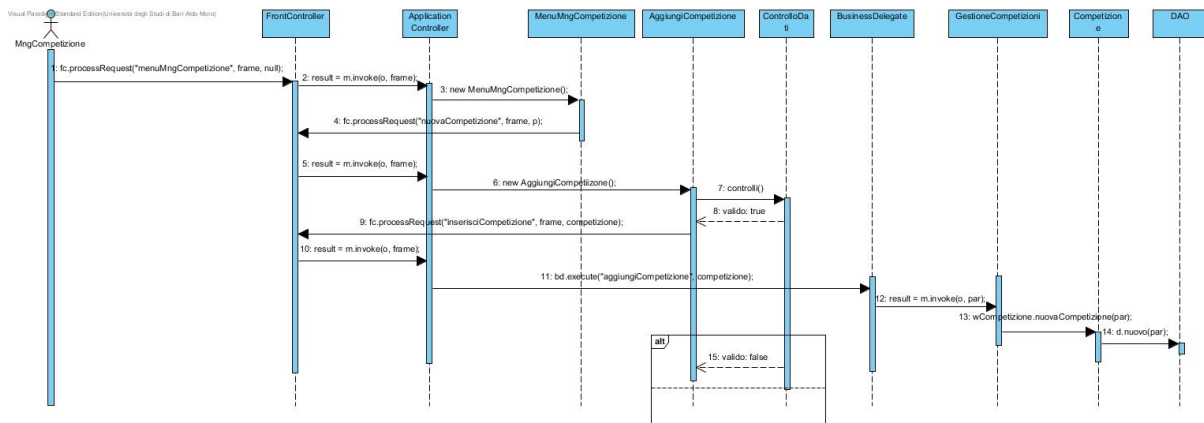
- Annullamento Competizione



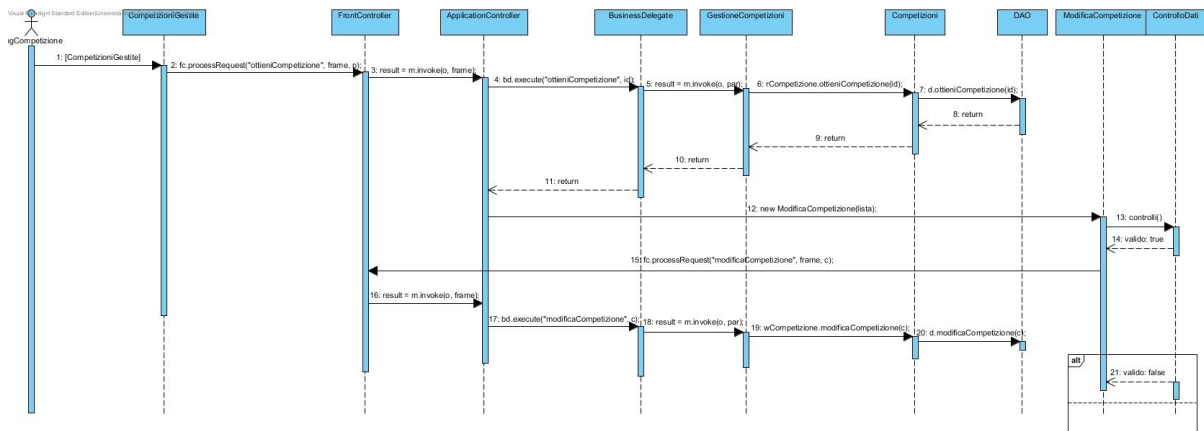
- Cancellazione Partecipante



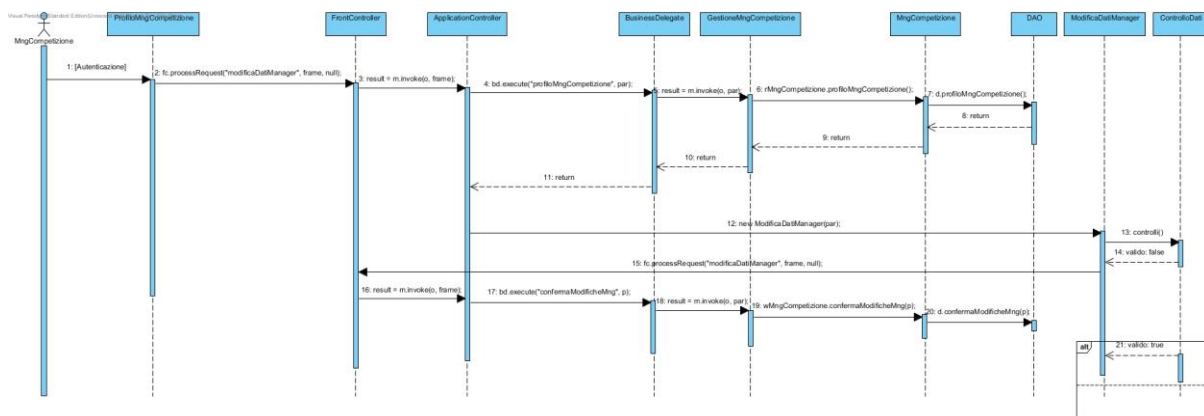
- Inserimento Competizione



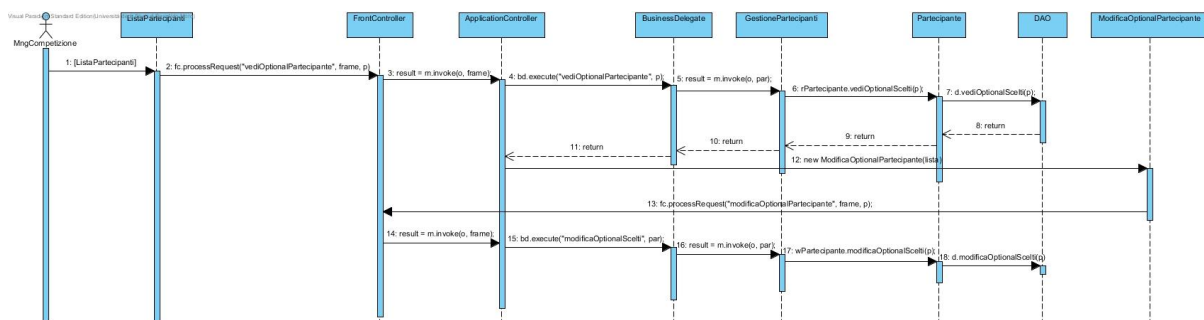
- Modifica Competizione



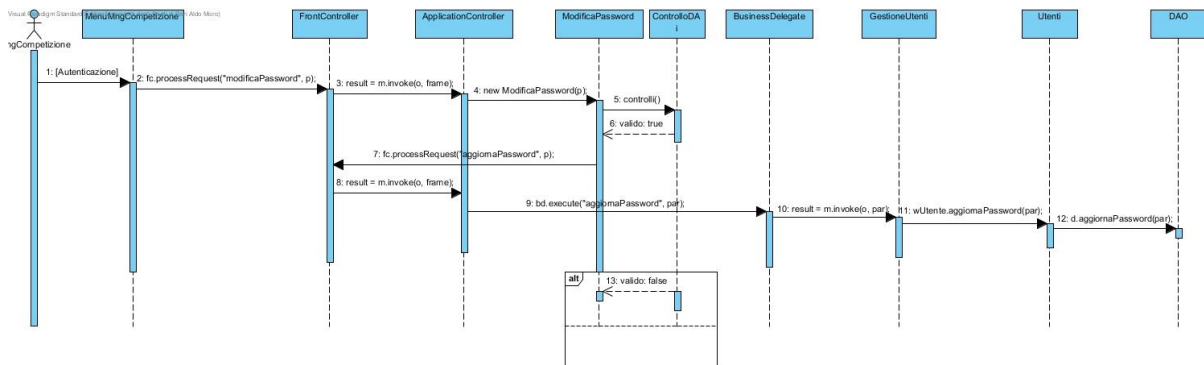
- Modifica Dati Personali



- Modifica Optional

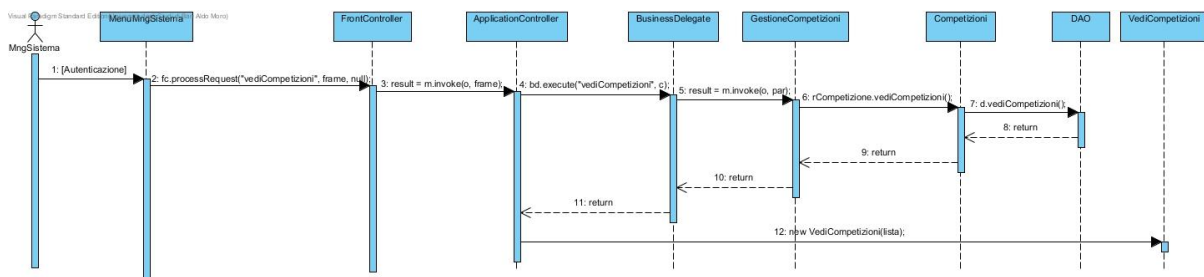


- Modifica Password

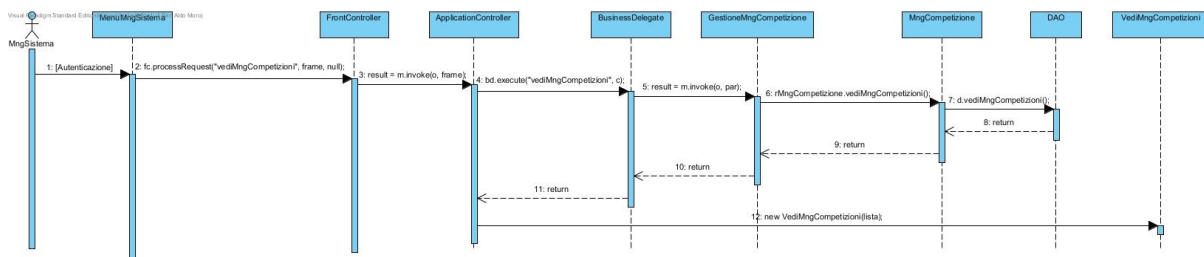


## MANAGER DI SISTEMA

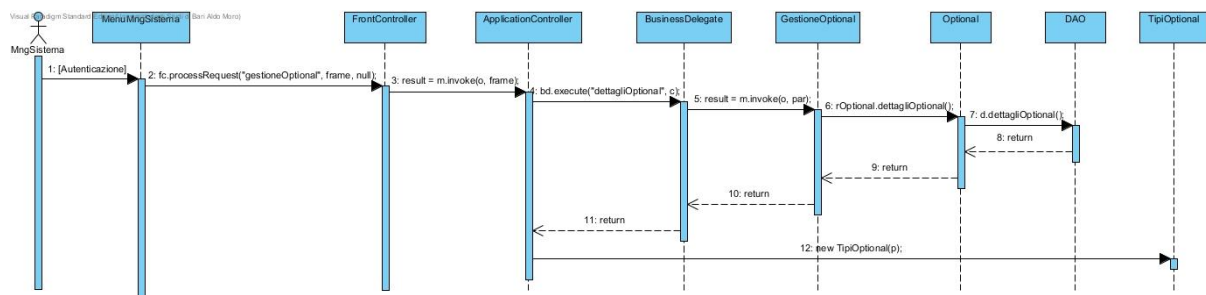
- Accesso a competizioni



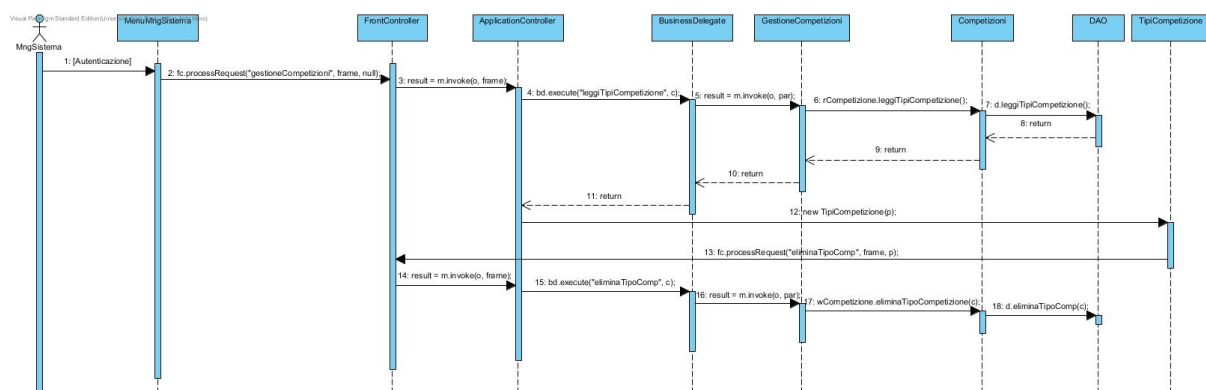
- Accesso Dati dei Manager di Competizione



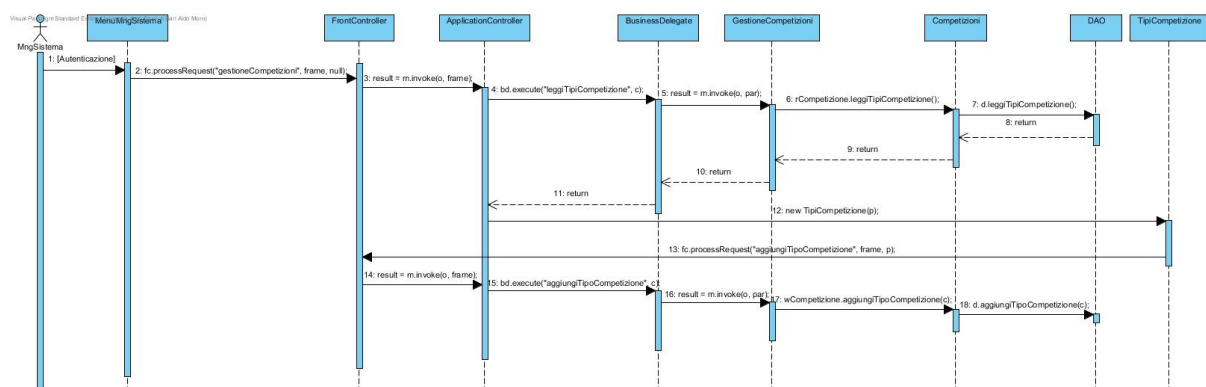
- Accesso Dati Tipi Optional



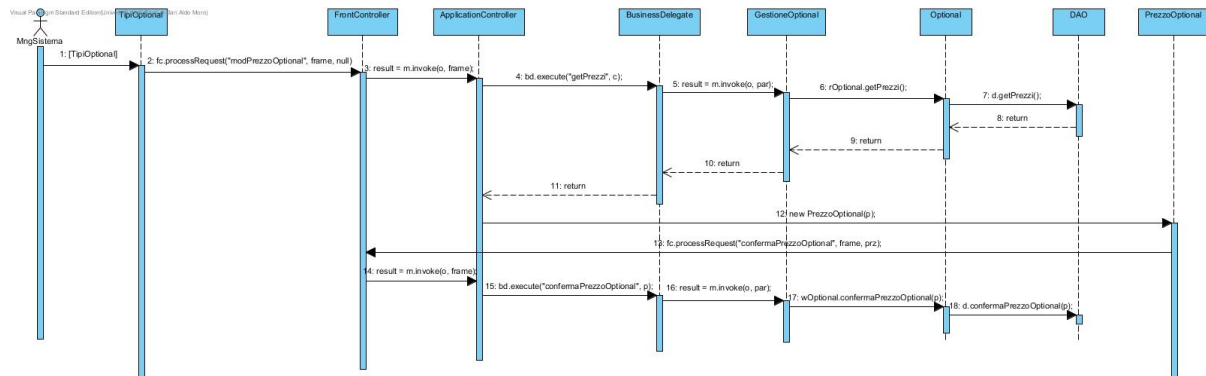
- Elimina Tipo Competizione



- Inserimento Dati Tipi di Competizione



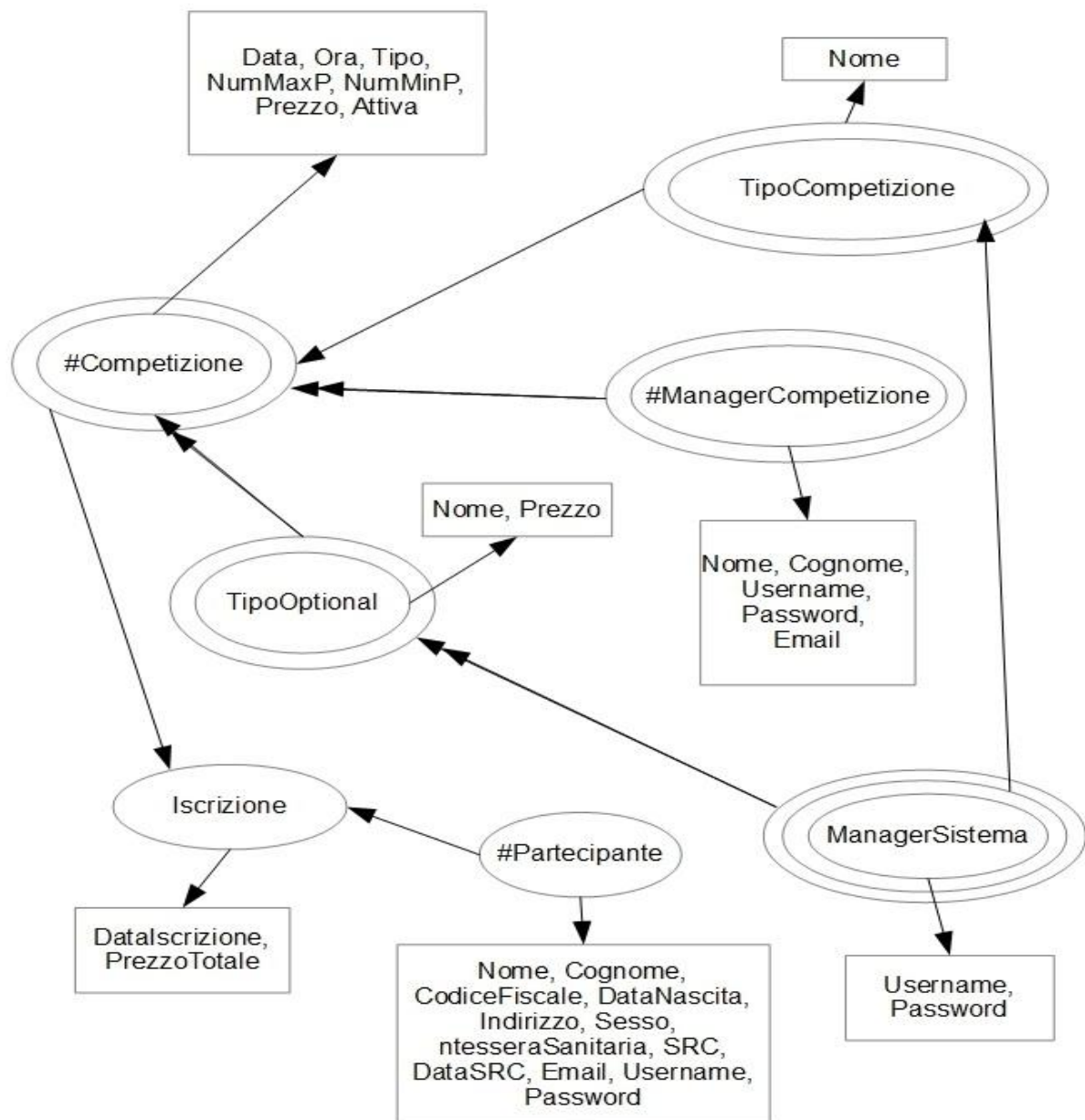
- Inserimento Dati Tipi Optional



## 3. PROGETTO DEI DATI

### 3.1 Database

#### 3.1.1 Diagramma delle Dipendenze dei Dati



### 3.1.2 Modello del Database

Si è scelto di utilizzare il modello relazionale per via della semplicità di utilizzo, diffusione ed efficienza.

### 3.1.3 Dettaglio dei Dati

PK = Primary Key

FK = Foreign Key

#### Competizioni

Campo	PK	FK	Tipo	Descrizione
idCompetizione	x		int	Identificativo della competizione
Tipo			varchar	Indica il tipo di competizione
Data			Date	Data in cui avrà luogo la competizione
Ora			Char(5)	Ora in cui inizierà la competizione
NumMaxPart			Int	Numero massimo di partecipanti che possono iscriversi alla competizione
NumMinPart			int	Numero minimo di partecipanti affinché la competizione possa avere luogo
Prezzo			double	Costo di iscrizione alla competizione
CodiceManager		x	int	Codice del manager che gestisce la competizione
Attiva			int	Settato a zero se la competizione è stata annullata
Pranzo			int	Flag di presenza dell'optional: settato a 1 se il manager ha previsto l'optional o a 0 se non è stato previsto
Merenda			int	
Pernotto			int	



### Partecipanti

Campo	PK	FK	Tipo	Descrizione
CodiceFiscale			char(16)	Codice Fiscale del partecipante
Nome			varchar	Nome del partecipante
Cognome			varchar	Cognome del partecipante
Residenza			varchar	Città di residenza
Via			varchar	Indirizzo di residenza
DataNascita			Date	Data di nascita del partecipante
Sesso			char(1)	Sesso del partecipante
NumTessSanitaria			char(20)	Numero tessera sanitaria
SRC			text	Testo del certificato di Sana e Robusta Costituzione
DataSRC			Date	Data di rilascio del certificato SRC
Email			varchar	Indirizzo di posta elettronica
Username	x		varchar	Nome utente con il quale il partecipante si è registrato al sistema
Pass			varchar	Password

### ManagerCompetizione

Campo	PK	FK	Tipo	Descrizione
idManager	x		int	Codice identificativo del manager
Nome			varchar	Nome del manager
Cognome			varchar	Cognome del manager
Email			varchar	Indirizzo di posta elettronica





Username			varchar	Nome utente con il quale il manager si è registrato al sistema
Pass			varchar	Password

### ManagerSistema

Campo	PK	FK	Tipo	Descrizione
Username	x		varchar	Nome utente del manager di sistema
Pass			varchar	Password

### Iscrizioni

Campo	PK	FK	Tipo	Descrizione
CodiceCompetizione	x	x	int	Codice identificativo della competizione
CodicePartecipante	x	x	varchar	Codice identificativo del partecipante
Pranzo			int	Flag di presenza dell'optional: settato a 1 se il partecipante ha selezionato l'optional o a 0 se non è stato selezionato
Merenda			int	
Pernotto			int	
DataIscr			Date	Data di iscrizione

### TipiOptional

Campo	PK	FK	Tipo	Descrizione
Nome	x		varchar	Nome del tipo di optional
Prezzo			double	Costo dell'optional



---

## TipiCompetizione

Campo	PK	FK	Tipo	Descrizione
Nome	x		varchar	Nome del tipo di competizione

## 3.2 File System

### 3.2.1 Grammatiche file XML

FC.xml:

```
<actions>
```

```
    <action value="menu">
```

```
        <class>business.ApplicationController</class>
```

```
        <method>menu</method>
```

```
    </action>
```

```
    <action value="registrazione">
```

```
        <class>business.ApplicationController</class>
```

```
        <method>registrazione</method>
```

```
    </action>
```

```
    <action value="login">
```

```
        <class>business.ApplicationController</class>
```

```
        <method>login</method>
```

```
    </action>
```



---

```
<action value="modificaPassword">
    <class>business.ApplicationController</class>
    <method>modificaPassword</method>
</action>
```

```
<action value="aggiornaPassword">
    <class>business.ApplicationController</class>
    <method>aggiornaPassword</method>
</action>
```

```
<!-- PARTECIPANTE -->
```

```
<action value="nuovoPartecipante">
    <class>business.ApplicationController</class>
    <method>nuovoPartecipante</method>
</action>
```

```
<action value="registrazionePartecipante">
    <class>business.ApplicationController</class>
    <method>registrazionePartecipante</method>
</action>
```

```
<action value="menuPartecipante">
    <class>business.ApplicationController</class>
    <method>menuPartecipante</method>
</action>
```

```
<action value="profiloPartecipante">
```



---

```
<class>business.ApplicationController</class>
  <method>profiloPartecipante</method>
</action>
```

```
<action value="modificaDatiPartecipante">
  <class>business.ApplicationController</class>
  <method>modificaDatiPartecipante</method>
</action>
```

```
<action value="confermaModifichePart">
  <class>business.ApplicationController</class>
  <method>confermaModifichePart</method>
</action>
```

```
<action value="competizioniDisponibili">
  <class>business.ApplicationController</class>
  <method>competizioniDisponibili</method>
</action>
```

```
<action value="leggiOptionalScelti">
  <class>business.ApplicationController</class>
  <method>leggiOptionalScelti</method>
</action>
```

```
<action value="modificaOptionalScelti">
  <class>business.ApplicationController</class>
  <method>modificaOptionalScelti</method>
```



---

```
</action>
```

```
<action value="selezioneOptional">  
    <class>business.ApplicationController</class>  
    <method>selezioneOptional</method>  
</action>
```

```
<action value="confermaIscrizione">  
    <class>business.ApplicationController</class>  
    <method>confermaIscrizione</method>  
</action>
```

```
<action value="listaIscrizioni">  
    <class>business.ApplicationController</class>  
    <method>listaIscrizioni</method>  
</action>
```

```
<action value="eliminaIscrizione">  
    <class>business.ApplicationController</class>  
    <method>eliminaIscrizione</method>  
</action>
```

```
<!-- MANAGER DI COMPETIZIONE -->
```

```
<action value="nuovoMngCompetizione">  
    <class>business.ApplicationController</class>  
    <method>nuovoMngCompetizione</method>  
</action>
```



---

```
<action value="registrazioneMngCompetizione">
    <class>business.ApplicationController</class>
    <method>registrazioneMngCompetizione</method>
</action>
```

```
<action value="menuMngCompetizione">
    <class>business.ApplicationController</class>
    <method>menuMngCompetizione</method>
</action>
```

```
<action value="nuovaCompetizione">
    <class>business.ApplicationController</class>
    <method>nuovaCompetizione</method>
</action>
```

```
<action value="inserisciCompetizione">
    <class>business.ApplicationController</class>
    <method>inserisciCompetizione</method>
</action>
```

```
<action value="profiloMngCompetizione">
    <class>business.ApplicationController</class>
    <method>profiloMngCompetizione</method>
</action>
```

```
<action value="competizioniGestite">
    <class>business.ApplicationController</class>
    <method>competizioniGestite</method>
</action>
```



---

```
<action value="modificaDatiManager">  
    <class>business.ApplicationController</class>  
    <method>modificaDatiManager</method>  
</action>
```

```
<action value="confermaModificheMng">  
    <class>business.ApplicationController</class>  
    <method>confermaModificheMng</method>  
</action>
```

```
<action value="listaPartecipanti">  
    <class>business.ApplicationController</class>  
    <method>listaPartecipanti</method>  
</action>
```

```
<action value="vediOptionalPartecipante">  
    <class>business.ApplicationController</class>  
    <method>vediOptionalPartecipante</method>  
</action>
```

```
<action value="modificaOptionalPartecipante">  
    <class>business.ApplicationController</class>  
    <method>modificaOptionalPartecipante</method>  
</action>
```

```
<action value="ottieniCompetizione">  
    <class>business.ApplicationController</class>  
    <method>ottieniCompetizione</method>
```



---

</action>

<action value="modificaCompetizione">  
    <class>business.ApplicationController</class>  
    <method>modificaCompetizione</method>  
</action>

<action value="annullaCompetizione">  
    <class>business.ApplicationController</class>  
    <method>annullaCompetizione</method>  
</action>

<action value="eliminaPartecipante">  
    <class>business.ApplicationController</class>  
    <method>eliminaPartecipante</method>  
</action>

<!-- MANAGER DI SISTEMA -->

<action value="menuMngSistema">  
    <class>business.ApplicationController</class>  
    <method>menuMngSistema</method>  
</action>

<action value="vediCompetizioni">  
    <class>business.ApplicationController</class>  
    <method>vediCompetizioni</method>  
</action>





---

```
<action value="vediMngCompetizioni">
    <class>business.ApplicationController</class>
    <method>vediMngCompetizioni</method>
</action>
```

```
<action value="gestioneCompetizioni">
    <class>business.ApplicationController</class>
    <method>gestioneCompetizioni</method>
</action>
```

```
<action value="gestioneOptional">
    <class>business.ApplicationController</class>
    <method>gestioneOptional</method>
</action>
```

```
<action value="modPrezzoOptional">
    <class>business.ApplicationController</class>
    <method>modPrezzoOptional</method>
</action>
```

```
<action value="confermaPrezzoOptional">
    <class>business.ApplicationController</class>
    <method>confermaPrezzoOptional</method>
</action>
```

```
<action value="eliminaTipoComp">
    <class>business.ApplicationController</class>
    <method>eliminaTipoComp</method>
```



---

```
</action>
```

```
<action value="aggiungiTipoCompetizione">  
    <class>business.ApplicationController</class>  
    <method>aggiungiTipoCompetizione</method>  
</action>
```

```
</actions>
```

### AC.xml:

```
<!-- GESTIONE PARTECIPANTI -->
```

```
<actions>
```

```
    <action value="login">  
        <class>business.GestioneUtenti</class>  
        <method>login</method>  
    </action>
```

```
    <action value="esisteUsername">  
        <class>business.GestioneUtenti</class>  
        <method>esisteUsername</method>  
    </action>
```

```
    <action value="getMail">  
        <class>business.GestioneUtenti</class>  
        <method>getMail</method>  
    </action>
```

```
    <action value="getDataSRC">
```



---

```
<class>business.GestionePartecipanti</class>  
<method>getDataSRC</method>  
</action>
```

```
<action value="getNumIscritti">  
    <class>business.GestionePartecipanti</class>  
    <method>getNumIscritti</method>  
</action>
```

```
<action value="registrazionePartecipante">  
    <class>business.GestionePartecipanti</class>  
    <method>registrazionePartecipante</method>  
</action>
```

```
<action value="profiloPartecipante">  
    <class>business.GestionePartecipanti</class>  
    <method>ottieniPartecipante</method>  
</action>
```

```
<action value="modificaDatiPartecipante">  
    <class>business.GestionePartecipanti</class>  
    <method>ottieniPartecipante</method>  
</action>
```

```
<action value="aggiornaPassword">  
    <class>business.GestioneUtenti</class>  
    <method>aggiornaPassword</method>  
</action>
```



---

```
<action value="confermaModifichePart">  
    <class>business.GestionePartecipanti</class>  
    <method>confermaModifichePart</method>  
</action>
```

```
<action value="leggiscrizioni">  
    <class>business.GestionePartecipanti</class>  
    <method>leggiscrizioni</method>  
</action>
```

```
<action value="eliminalscrizione">  
    <class>business.GestionePartecipanti</class>  
    <method>eliminalscrizione</method>  
</action>
```

```
<action value="leggiOptionalScelti">  
    <class>business.GestionePartecipanti</class>  
    <method>leggiOptionalScelti</method>  
</action>
```

```
<action value="modificaOptionalSceltiP">  
    <class>business.GestionePartecipanti</class>  
    <method>modificaOptionalSceltiP</method>  
</action>
```

```
<action value="modificaOptionalSceltiM">  
    <class>business.GestionePartecipanti</class>  
    <method>modificaOptionalSceltiM</method>  
</action>
```



---

```
<action value="listaPartecipanti">
    <class>business.GestioneCompetizioni</class>
    <method>listaPartecipanti</method>
</action>
```

```
<action value="vediOptionalPartecipante">
    <class>business.GestionePartecipanti</class>
    <method>vediOptionalPartecipante</method>
</action>
```

```
<action value="getPassword">
    <class>business.GestioneUtenti</class>
    <method>getPassword</method>
</action>
```

```
<action value="getMailList">
    <class>business.GestionePartecipanti</class>
    <method>getMailList</method>
</action>
```

```
<!--          GESTIONE MANAGER COMPETIZIONE          -->
```

```
<action value="registrazioneMngCompetizione">
    <class>business.GestioneMngCompetizione</class>
    <method>registrazioneMngCompetizione</method>
</action>
```

```
<action value="profiloMngCompetizione">
```



---

```
<class>business.GestioneMngCompetizione</class>
<method>profiloMngCompetizione</method>
</action>
```

```
<action value="confermaModificheMng">
  <class>business.GestioneMngCompetizione</class>
  <method>confermaModificheMng</method>
</action>
```

```
<action value="vediMngCompetizioni">
  <class>business.GestioneMngCompetizione</class>
  <method>vediMngCompetizioni</method>
</action>
```

```
<action value="getMailMng">
  <class>business.GestioneMngCompetizione</class>
  <method>getMailMng</method>
</action>
```

```
<action value="riduciPartecipanti">
  <class>business.GestioneCompetizioni</class>
  <method>riduciPartecipanti</method>
</action>
```

```
<!--          GESTIONE COMPETIZIONI          -->
```

```
<action value="leggiCompetizioni">
  <class>business.GestioneCompetizioni</class>
  <method>leggiCompetizioni</method>
```



---

</action>

<action value="leggiTipiCompetizione">  
    <class>business.GestioneCompetizioni</class>  
    <method>leggiTipiCompetizione</method>  
</action>

<action value="leggiOptional">  
    <class>business.GestioneCompetizioni</class>  
    <method>leggiOptional</method>  
</action>

<action value="iscrizioneCompetizione">  
    <class>business.GestionePartecipanti</class>  
    <method>iscrizioneCompetizione</method>  
</action>

<action value="inserisciCompetizione">  
    <class>business.GestioneCompetizioni</class>  
    <method>nuovaCompetizione</method>  
</action>

<action value="competizioniGestite">  
    <class>business.GestioneCompetizioni</class>  
    <method>competizioniGestite</method>  
</action>

<action value="annullaCompetizione">  
    <class>business.GestioneCompetizioni</class>



---

```
        <method>annullaCompetizione</method>
    </action>
```

```
<action value="ottieniCompetizione">
    <class>business.GestioneCompetizioni</class>
    <method>ottieniCompetizione</method>
</action>
```

```
<action value="modificaCompetizione">
    <class>business.GestioneCompetizioni</class>
    <method>modificaCompetizione</method>
</action>
```

```
<action value="vediCompetizioni">
    <class>business.GestioneCompetizioni</class>
    <method>vediCompetizioni</method>
</action>
```

```
<action value="dettagliOptional">
    <class>business.GestioneOptional</class>
    <method>dettagliOptional</method>
</action>
```

```
<action value="getPrezzi">
    <class>business.GestioneOptional</class>
    <method>getPrezzi</method>
</action>
```

```
<action value="confermaPrezzoOptional">
```





---

```
<class>business.GestioneOptional</class>
<method>confermaPrezzoOptional</method>
</action>

<action value="eliminaTipoComp">
    <class>business.GestioneCompetizioni</class>
    <method>eliminaTipoComp</method>
</action>

<action value="aggiungiTipoCompetizione">
    <class>business.GestioneCompetizioni</class>
    <method>aggiungiTipoCompetizione</method>
</action>

</actions>
```

### 3.2.2 Altri file

Oltre ai due file xml visti nella sezione precedente, sono stati utilizzati i seguenti file di configurazione:

- **confmail.dat**: contiene le impostazioni della casella di posta elettronica
- **database.settings**: contiene le informazioni di accesso al database.



---

## 4. APPENDICE

---

### 4.1 Patterns Utilizzati

#### Front Controller

L'uso del Front Controller permette di centralizzare la gestione delle richieste ricevute da parte delle varie interfacce.

In questo modo le finestre potranno solo richiedere servizi al Front Controller e non conterranno alcuna logica di business mentre quest'ultimo, in base alla richiesta ricevuta, potrà richiedere un servizio al livello di dominio. Il vantaggio principale legato all'utilizzo del Front Controller è quello di aumentare la manutenibilità del livello di dominio separandolo dalla logica di presentazione.

#### Application Controller

L'uso dell'Application Controller permette di centralizzare l'azione; in questo modo, infatti, è possibile riutilizzare il codice e migliorare la gestione delle richieste di estensione, come l'aggiunta di funzionalità in modo incrementale. Allo stesso tempo, inoltre, si ottiene un miglioramento della manutenibilità del sistema dovuta alla possibilità di apportare modifiche in modo più facile e di testare parti distinte del sistema in modo indipendente. Nello specifico, dunque, l'Application Controller si occuperà di ricevere dal Front Controller la richiesta in modo da delegare l'esecuzione.

#### Business Delegate

L'uso del Business Delegate migliora la separazione tra il livello di presentazione quello del dominio, raccogliendo tutti i servizi offerti dai singoli gestori a livello di dominio ed esponendone i servizi tramite un'unica interfaccia. Il Front Controller dunque invocherà un servizio del Business Delegate ignorando completamente quali siano i gestori a livello di dominio che realizzano tale servizio. L'uso del Business Delegate aumenta notevolmente la manutenibilità del livello di dominio. Ad ogni modifica, infatti, il livello di presentazione non subirà impatti rilevanti.

#### Application Service

Sarà presente una serie di classi che si occuperanno di incapsulare al loro interno la logica di uno specifico caso d'uso permettendo la centralizzazione della logica di business presente nelle componenti e nei servizi del livello.

#### DAO

Il DAO consente la separazione tra il livello di dominio e il livello di sorgente dati. Esso permette, infatti, ai gestori dei servizi a livello di dominio di accedere alla



---

sorgente dati attraverso un'unica interfaccia centralizzata e senza conoscere la reale implementazione fisica dei dati.

## **4.2 Altro**

