



# UNIVERSITÀ DEGLI STUDI DI SALERNO

Ingegneria del Software

## OBJECT DESIGN DOCUMENT



ANNO ACCADEMICO 2015/2016

*Versione 1.7*

## Top Manager:

Nome
Prof. De Lucia Andrea

## Project Manager:

Nome	Matricola
De Chiara Davide	0222500088
Longo Alessandro	0222500121

## Partecipanti:

Nome	Matricola
Castellano Dario	0512102552
D'Avanzo Antonio Luca	0512102502
De Blasio Christian	0512102268
De Rosa Federico	0512101904
Di Domenico Carlo	0512102316
Esposito Fabio	0512102426
Korniychuk Alina	0512102256
Martiniello Pasquale	0512102616
Pecorelli Fabiano	0512102390

Shevchenko Sergey	0512102368
Tufano Giuseppina	0512102588
Zanin Elvira	0512102396

## Revision History:

Data	Versione	Descrizione	Autore
04/12/2015	1.0	Stesura del paragrafo 1	Korniychuk Alina Tufano Giuseppina
15/12/2015	1.1	Stesura del paragrafo 2	Korniychuk Alina Tufano Giuseppina Zanin Elvira
16/12/2015	1.2	Stesura del paragrafo 3	D'Avanzo Antonio Luca De Blasio Christian De Rosa Federico Di Domenico Carlo Esposito Fabio Martiniello Pasquale Pecorelli Fabiano Shevchenko Sergey
18/12/2015	1.3	Stesura del Glossario	Korniychuk Alina Tufano Giuseppina
30/12/2015	1.4	Aggiunta Package	Korniychuk Alina
02/02/2015	1.5	Aggiunta delle tabelle riferite ai package con relativi commenti	Tufano Giuseppina
02/01/2016	1.6	Modifica del paragrafo 3 con inserimento	Korniychuk Alina

		della documentazione "phpDoc"	
3/01/2016	1.7	Modifica dei class diagram del paragrafo 3.	D'Avanzo Antonio Luca De Blasio Christian De Rosa Federico Di Domenico Carlo Esposito Fabio Martiniello Pasquale Pecorelli Fabiano Shevchenko Sergey

# Indice

1. Introduzione .....	6
1.1 Object Design Trade-offs .....	6
1.2 Linee Guida per la Documentazione delle Interfacce .....	7
1.3 Definizioni, acronimi e abbreviazioni .....	9
1.4 Riferimenti.....	9
2. Packages .....	10
2.1 Package core.....	11
2.1.1 Package bean.....	12
2.1.2 Package control .....	13
2.1.3 Package exception .....	25
2.1.4 Package model .....	26
2.1.5 Package tests .....	28
2.1.6 Package utils .....	30
2.1.7 Package view.....	31
3. Class interfaces .....	39
3.1 Gestione Utente .....	39
3.2 Gestione Corsi di Laurea .....	40
3.3 Gestione Argomenti .....	41
3.4 Gestione Test .....	42
3.5 Gestione Statistiche .....	43
4. Glossario.....	44

# 1. Introduzione

## 1.1 Object Design Trade-offs

Dopo la realizzazione dei documenti RAD e SDD abbiamo descritto in linea di massima quello che sarà il nostro sistema e quindi i nostri obiettivi, tralasciando gli aspetti implementativi. Il seguente documento ha lo scopo di produrre un modello capace di integrare in modo coerente e preciso tutte le funzionalità individuate nelle fasi precedenti. In particolare definisce le interfacce delle classi, le operazioni, i tipi, gli argomenti e la signature dei sottosistemi definiti nel System Design. Inoltre sono specificati i trade-off e le linee guida.

### **Comprensibilità vs Tempo:**

Il codice deve essere quanto più comprensibile possibile per facilitare la fase di testing ed eventuali future modifiche. Il codice sarà quindi accompagnato da commenti che ne semplifichino la comprensione. Ovviamente questa caratteristica aggiungerà un incremento di tempo allo sviluppo del nostro progetto.

### **Prestazioni vs Costi:**

Essendo il progetto sprovvisto di budget, al fine di mantenere prestazioni elevate, per alcune funzionalità verranno utilizzati dei template open source esterni in particolare Bootstrap.

### **Interfaccia vs Usabilità:**

L'interfaccia grafica è stata realizzata in modo da essere molto semplice, chiara e concisa, fa uso di form e pulsanti disposti in maniera da rendere semplice l'utilizzo del sistema da parte dell'utente finale.

### **Sicurezza vs Efficienza:**

La sicurezza, come descritto nei requisiti non funzionali del RAD, rappresenta uno degli aspetti importanti del sistema. Tuttavia, dati i tempi di sviluppo molto limitati, ci limiteremo ad implementare sistemi di sicurezza basati su username e password degli utenti.

## 1.2 Linee Guida per la Documentazione delle Interfacce

Gli sviluppatori dovranno seguire alcune linee guida per la scrittura del codice:

### Naming Convention

- E' buona norma utilizzare nomi:
  1. Descrittivi
  2. Pronunciabili
  3. Di uso comune
  4. Lunghezza medio-corta
  5. Non abbreviati
  6. Evitando la notazione ungherese
  7. Utilizzando solo caratteri consentiti (a-z, A-Z, 0-9)

### Variabili:

- I nomi delle variabili devono cominciare con una lettera minuscola, e le parole seguenti con la lettera maiuscola. Quest'ultime devono essere dichiarate ad inizio blocco, solamente una per riga e devono essere tutte allineate per facilitare la leggibilità.

Esempio: `elaboratoSessionId`

- E' inoltre possibile, in alcuni casi, utilizzare il carattere **underscore** “\_”, ad esempio quando utilizziamo delle variabili costanti oppure quando vengono utilizzate delle proprietà statiche.

Esempio: `CREATE_ARGOMENTO`

### Metodi:

- I nomi dei metodi devono cominciare con una lettera minuscola, e le parole seguenti con la lettera maiuscola. Il nome del metodo tipicamente consiste di un verbo che identifica una azione, seguito dal nome di un oggetto.

I nomi dei metodi per l'accesso e la modifica delle variabili dovranno essere del tipo `getNomeVariabile()` e `setNomeVariabile()`. Le variabili dei metodi devono essere dichiarate appena prima del loro utilizzo e devono servire per un solo scopo, per facilitare la leggibilità. Esistono però casi particolari come ad esempio nell'implementazione dei model, dove viene utilizzata l'interfaccia CRUD.

Esempio: `getId()`, `setId()`

- I commenti dei metodi devono essere raggruppati in base alla loro funzionalità, la descrizione dei metodi deve apparire prima di ogni dichiarazione di metodo, e deve descriverne lo scopo. Deve includere anche informazioni sugli argomenti, sul valore di ritorno, e se applicabile, sulle eccezioni.

### Classi e pagine :

- I nomi delle classi e delle pagine devono cominciare con una lettera maiuscola, e anche le parole seguenti all'interno del nome devono cominciare con una lettera maiuscola. I nomi di quest'ultime devono fornire informazioni sul loro scopo.  
Esempio: RispostaMultipla.php

Ogni file sorgente php contiene una singola classe e dev'essere strutturato in un determinato modo:

- Una breve introduzione alla classe

L'introduzione indica: l'autore, la versione e la data.

```
/**
 * sommario dello scopo della classe.
 *
 * @author nome dell'autore
 * @version numero di versione della classe
 * @since data d'implementazione
 */
```

- L'istruzione include che permette di importare all'interno della classe gli altri oggetti che la classe utilizza.
- La dichiarazione di classe caratterizzata da:
  1. Dichiarazione della classe pubblica
  2. Dichiarazioni di costanti
  3. Dichiarazioni di variabili di classe
  4. Dichiarazioni di variabili d'istanza
  5. Costruttore
  6. Commento e dichiarazione metodi.

ESEMPIO:

```
class CdL {
    private $matricola;
    private $nome;

    /**
     * Costruttore di CdL
     * @param string $matricola La matricola del CdL
     * @param string $nome Il nome del CdL
     */
    public function __construct($matricola, $nome) {
        $this->matricola = $matricola;
        $this->nome = $nome; }

    /**
     * @return string La matricola del CdL
```



```

*/
public function getMatricola() {
    return $this->matricola; }

/**
 * @return string Il nome del CdL
 */
public function getNome() {
    return $this->nome; }

/**
 * Setta la matricola del CdL
 * @param string $matricola La matricola del CdL
 */
public function setMatricola($matricola) {
    $this->matricola = $matricola; }

/**
 * Setta il nome del CdL
 * @param string $nome Il nome del CdL
 */
public function setNome($nome) {
    $this->nome = $nome; }
}

```

## 1.3 Definizioni, acronimi e abbreviazioni

### Acronimi:

- RAD: Requirements Analysis Document
- SDD: System Design Document
- ODD: Object Design Document
- CRUD: Create Read Update Delete

### Abbreviazioni:

- DB: DataBase

## 1.4 Riferimenti

- B. Bruegge, A. H. Dutoit, Object Oriented Software Engineering - Using UML, Pattern and Java, Prentice Hall, 3rd edition, 2009
- Documento SDD del progetto Simplex
- Documento RAD del progetto Simplex

## 2. Packages

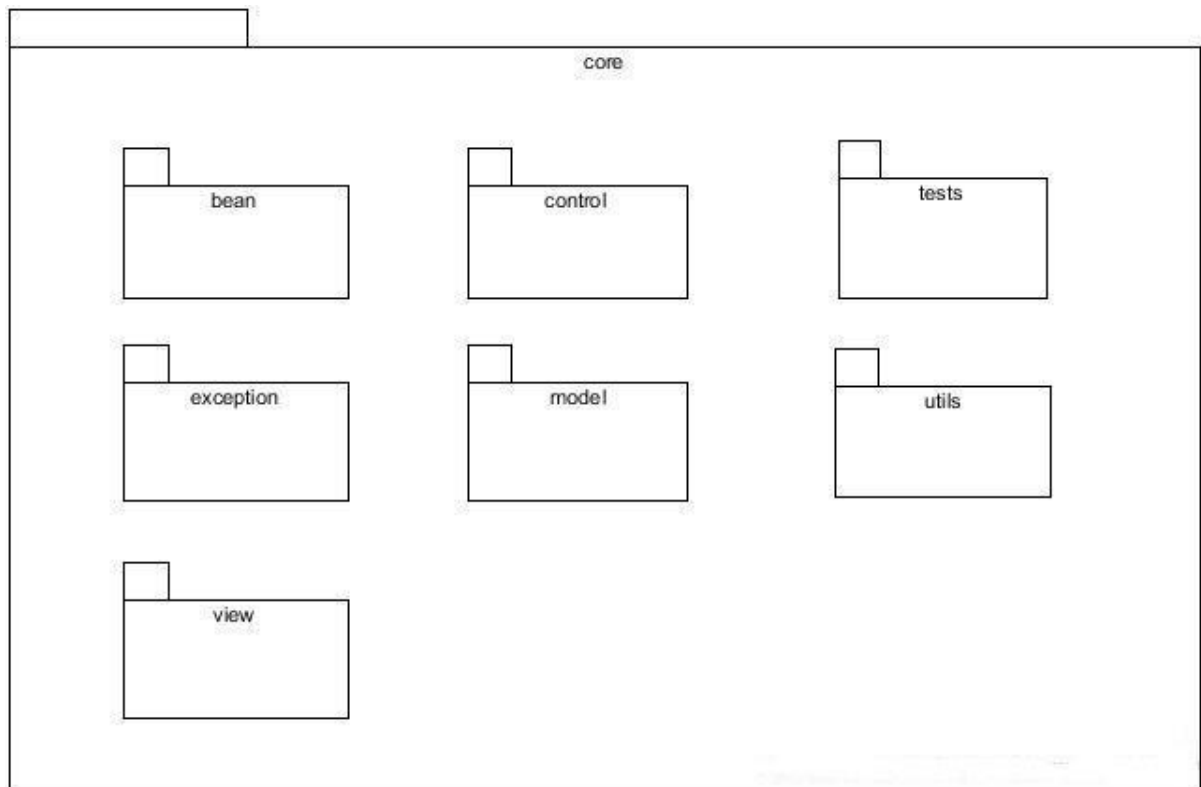
La gestione del nostro sistema è suddivisa in tre livelli (three-tier):

- Interface layer
- Application Logic layer
- Storage layer

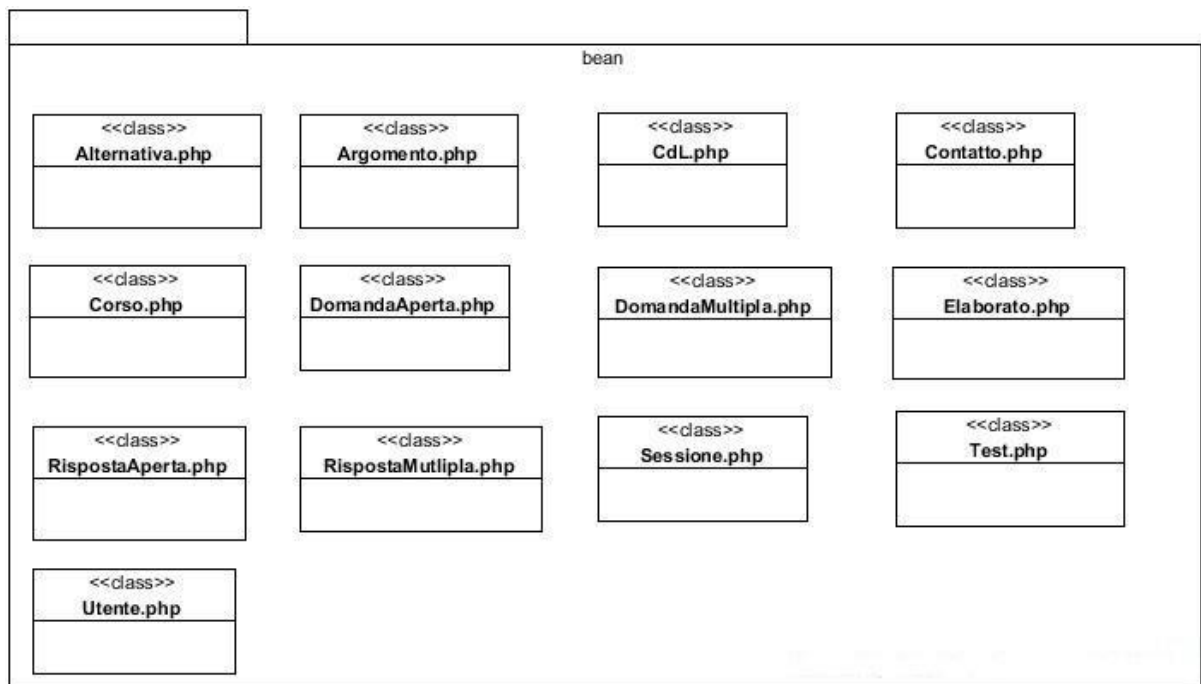
Il package Simplex contiene sottopackage che a loro volta inglobano classi atte alla gestione delle richieste utente. Le classi contenute nel package svolgono il ruolo di gestore logico del sistema.

Interface layer	Rappresenta l'interfaccia del sistema, ed offre la possibilità all'utente di interagire con quest'ultimo, offrendo sia la possibilità di inviare, in input, che di visualizzare, in output, dati.
Application Logic layer	<p>Ha il compito di elaborare i dati da inviare al client, e spesso grazie a delle richieste fatte al database, tramite lo Storage Layer, accede ai dati persistenti.</p> <p>Si occupa di varie gestioni quali:</p> <ol style="list-style-type: none"><li>1. <b>Gestione Utente</b></li><li>2. <b>Gestione Corsi di Laurea</b></li><li>3. <b>Gestione Argomenti</b></li><li>4. <b>Gestione Test</b></li><li>5. <b>Gestione Statistiche</b></li></ol>
Storage layer	Ha il compito di memorizzare i dati sensibili del sistema, utilizzando un DBMS, inoltre riceve le varie richieste dall' Application Logic layer inoltrandole al DBMS e restituendo i dati richiesti.

## 2.1 Package core

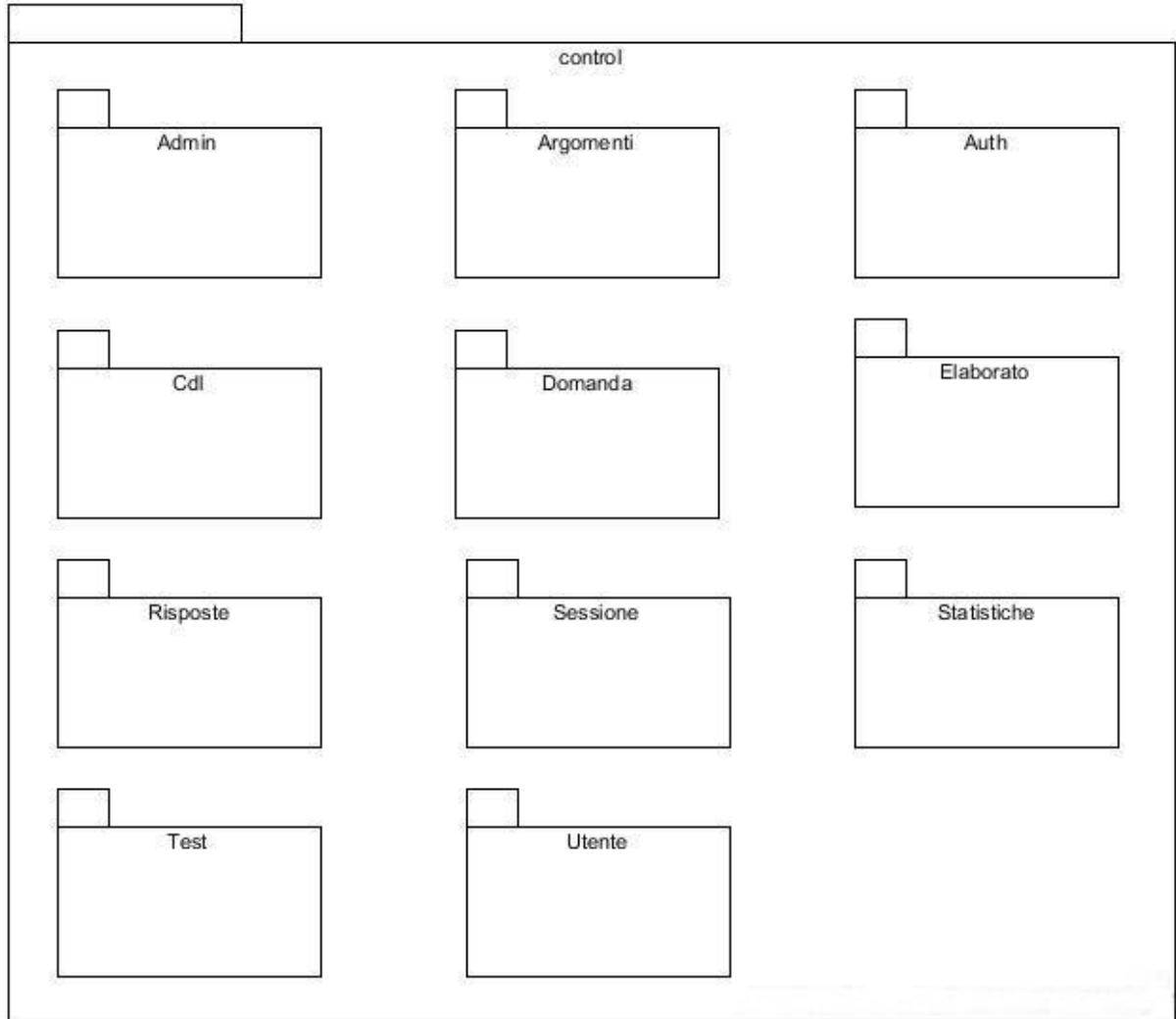


## 2.1.1 Package bean

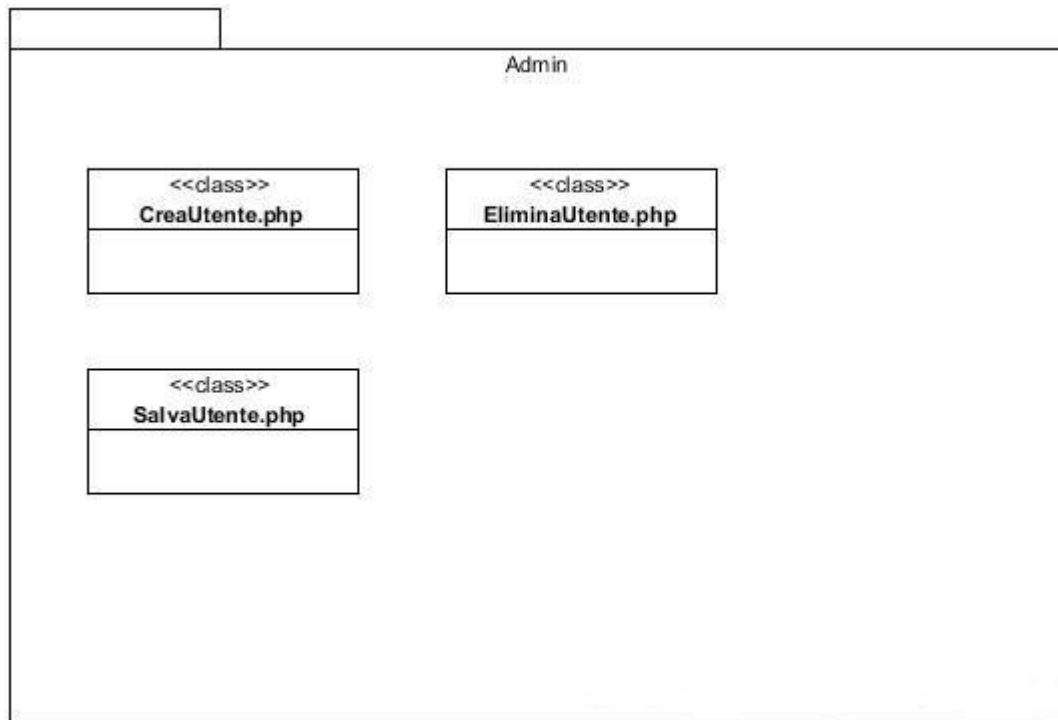


Classe:	Descrizione:
Alternativa.php	Descrive l'alternativa di una domanda multipla
Argomento.php	Descrive l'argomento di un corso
Cdl.php	Descrive un corso di laurea.
Contatto.php	Descrive il contatto di un utente.
Corso.php	Descrive un corso di un corso di laurea
DomandaAperta.php	Descrive una domanda aperta di un argomento
DomandaMultipla.php	Descrive una domanda multipla di un argomento.
Elaborato.php	Descrive l'elaborato di uno studente
RispostaAperta.php	Descrive la risposta ad una domanda aperta di un elaborato
RispostaMultipla.php	Descrive la risposta ad una domanda multipla di un elaborato
Sessione.php	Descrive la sessione di un corso
Test.php	Descrive un test di un corso che è possibile associare alle sessioni dello stesso

## 2.1.2 Package control

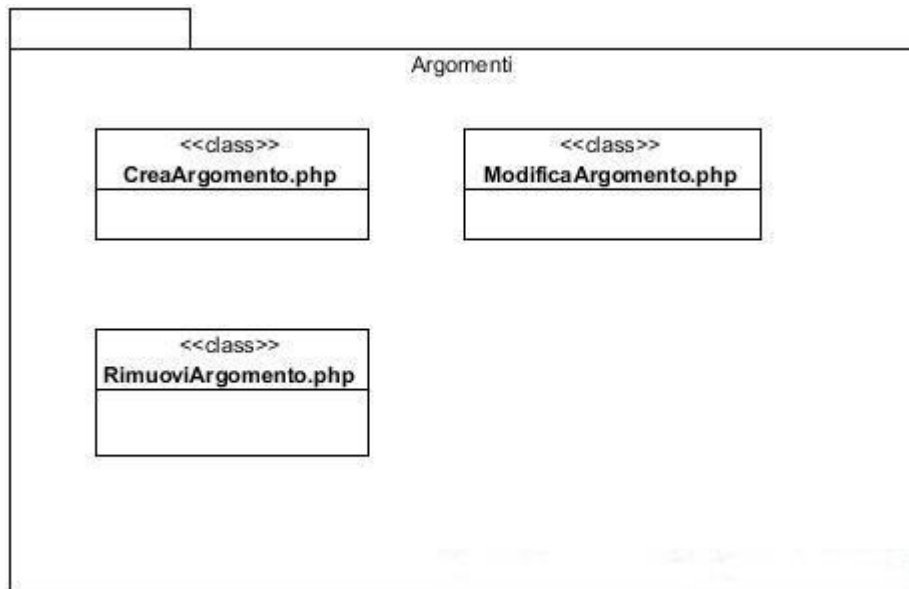


### 2.1.2.1 Package Admin



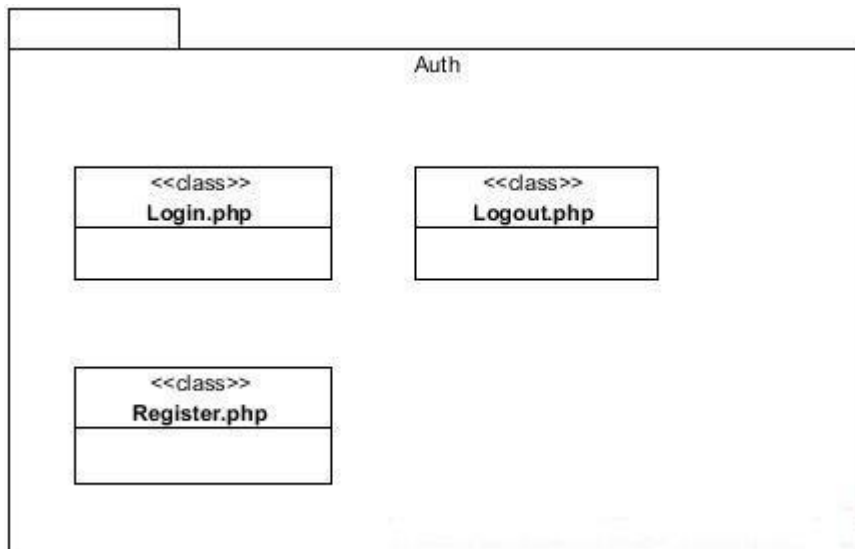
Crea Utente.php	View che riguarda la creazione dell'utente
EliminaUtente.php	View che riguarda l'eliminazione dell'utente
SalvaUtente.php	View che riguarda il salvataggio dell'utente

### 2.1.2.2 Package Argomenti



CreaArgomento.php	Controller che permette di creare un argomento
ModificaArgomento.php	Controller che permette di modificare un argomento
RimuoviArgomento.php	Controller che permette la rimozione di un argomento

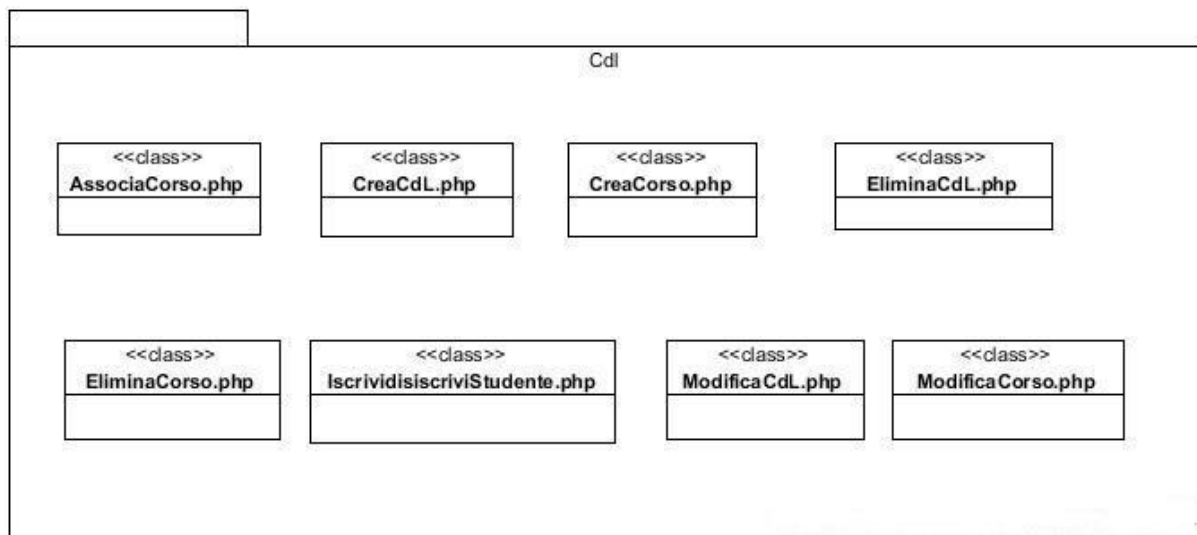
### 2.1.2.3 Package Auth



Login.php	Controller che permette la gestione del login
Logout.php	Controller che permette la gestione del logout
Register.php	Controller che permette la gestione della registrazione

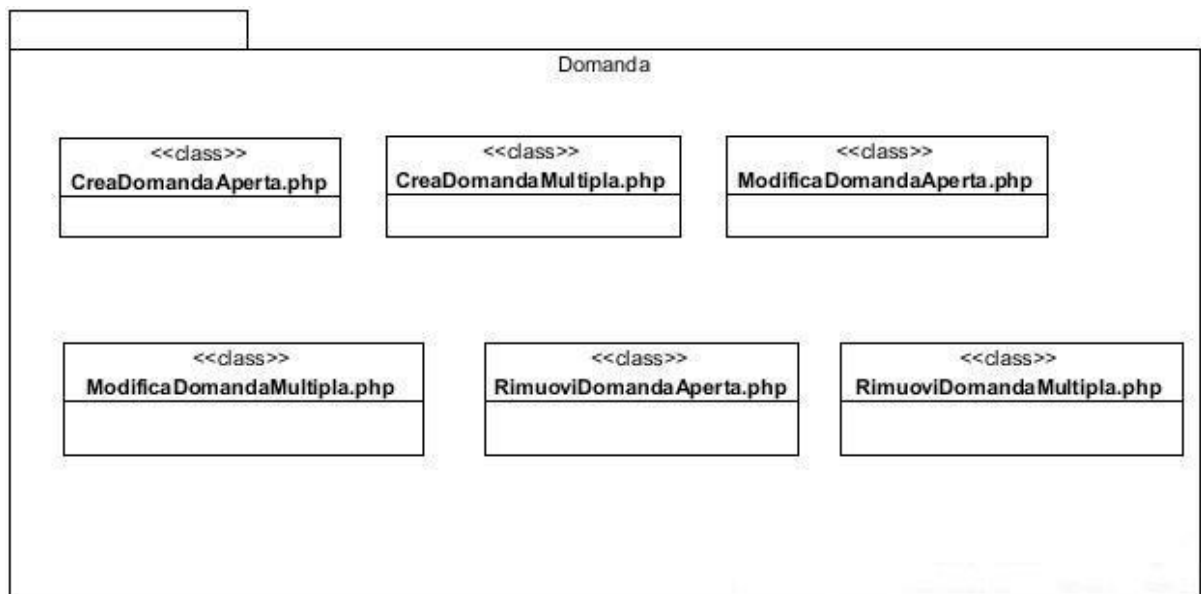


### 2.1.2.4 Package Cdl



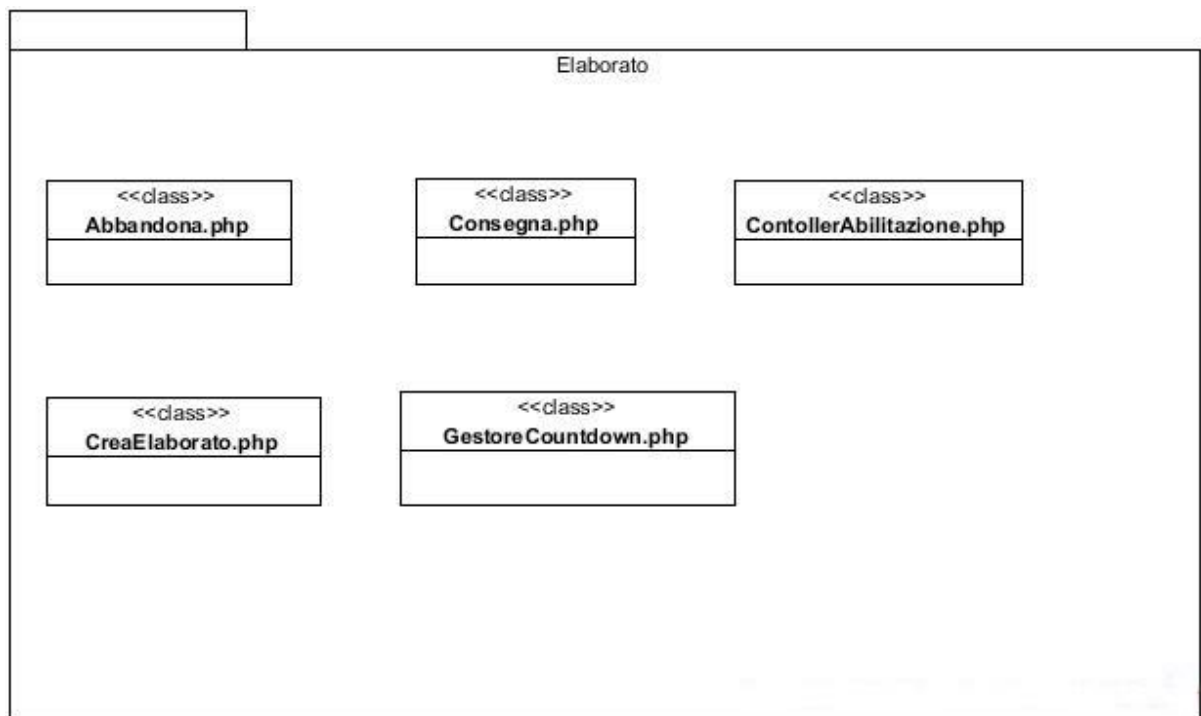
AssociaCorso.php	Controller che permette la gestione dell'associazione docente-corso
CreaCdL.php	Controller che permette la gestione della creazione di un corso di laurea
CreaCorso.php	Controller che permette la gestione della creazione di un corso
EliminaCdL.php	Controller che permette la gestione dell'eliminazione di un corso di laurea
EliminaCorso.php	Controller che permette la gestione dell'eliminazione di un corso
IscriviDisiscriviStudente.php	Controller che permette la gestione dell'iscrizione e la disiscrizione di uno studente
ModificiaCdL.php	Controller che permette la gestione della modifica di un corso di laurea
ModificaCorso.php	Controller che permette la gestione della modifica di un corso

### 2.1.2.5 Package Domanda



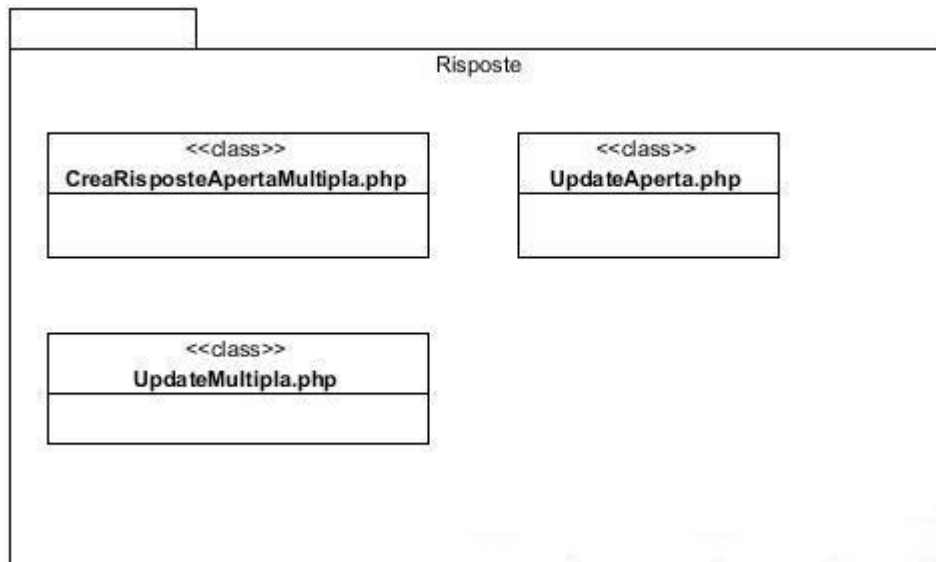
CreaDomandaAperta.php	Controller che permette di creare una domanda aperta.
CreaDomandaMultipla.php	Controller che permette di creare una domanda multipla.
ModificaDomandaAperta.php	Controller che permette di modificare una domanda aperta.
ModificaDomandaMultipla.php	Controller che permette di modificare una domanda multipla.
RimuoviDomandaAperta.php	Controller che permette di rimuovere una domanda aperta.
RimuoviDomandaMultipla.php	Controller che permette di rimuovere una domanda multipla.

### 2.1.2.6 Package Elaborato



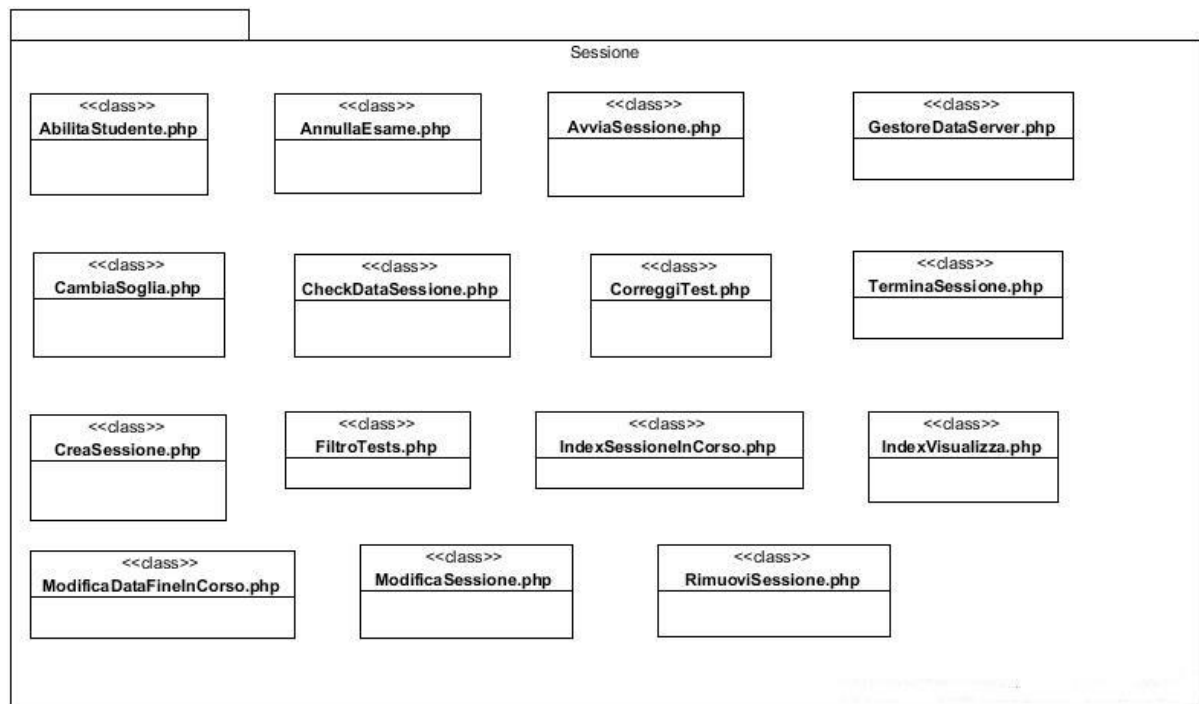
Abbandona.php	Controller per il ritiro di uno studente da una sessione
Consegna.php	Controller per il ritiro di uno studente da una sessione
ContollerAbilitazione.php	Controller che controlla che lo studente sia abilitato ad eseguire una sessione
CreaElaborato.php	Controller per la creazione di un elaborato
GestoreCountdown.php	Controller per il countdown nella pagina di esecuzione del test

### 2.1.2.7 Package Risposte



CreaRispostaApertaMultipla.php	Controller che permette la gestione della creazione di una risposta, sia aperta che multipla
UpdateAperta.php	Controller che permette la gestione della modifica di una risposta aperta
UpdateMultipla.php	Controller che permette la gestione della modifica di una risposta multipla

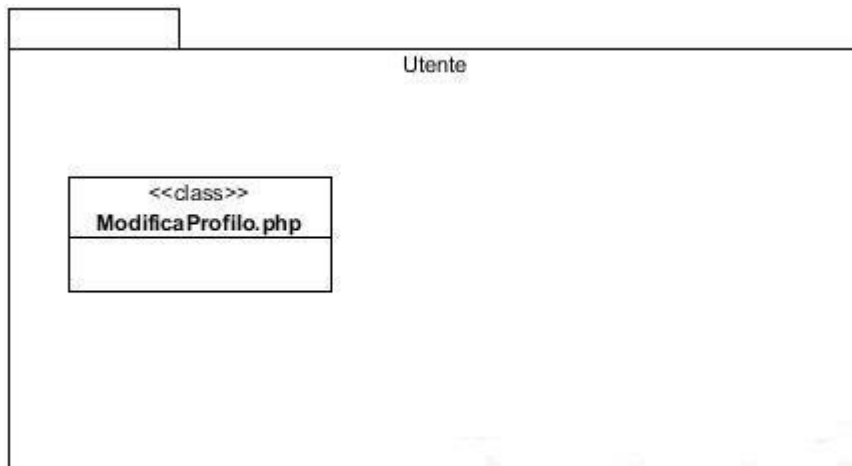
### 2.1.2.8 Package Sessione



AbilitaStuْدente.php	Controller che permette di abilitare uno studente ad una sessione.
AnnullaEsame.php	Controller che permette ad uno studente l'esame durante una sessione in corso.
AvviaSessione.php	Controller che permette di avviare una sessione manualmente.
CambiaSoglia.php	Controller che permette di modificare la soglia di ammissione di una sessione.
CheckDataSessione.php	Controller che permette di aggiornare lo stato di una sessione in relazione alla data attuale.
CorreggiTest.php	Controller che permette al docente di correggere un test.
CreaSessione.php	Controller che permette al docente di creare una sessione.
FiltroTest.php	Controller che permette di filtrare i test per punteggio massimo.
IndexSessioneInCorso.php	Da questo control possiamo raggiungere altre pagine. Viene invocato dalla view SessioneInCorso.

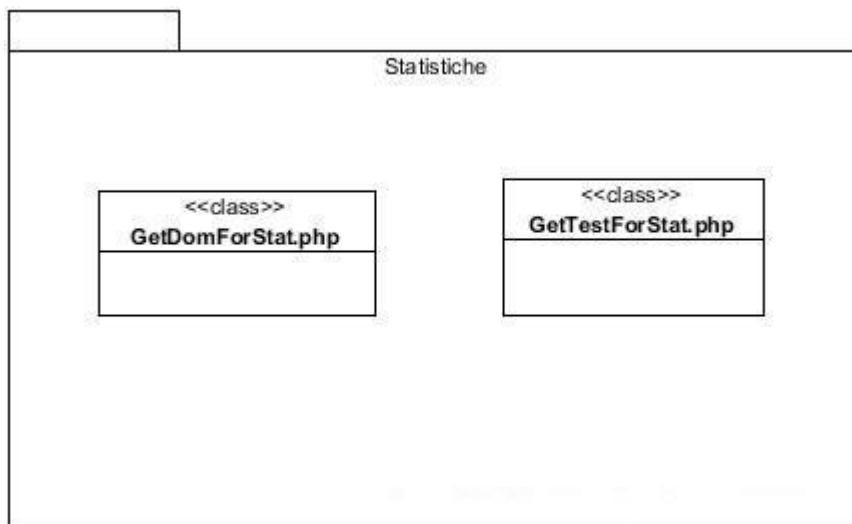
IndexVisualizza.php	Da questo control possiamo raggiungere altre pagine. Viene invocato dalla view VisualizzaSessione.
ModificaDataFineInCorso.php	Controller che permette al docente di modificare la data di termine di una sessione in corso.
ModificaSessione.php	Controller che permette al docente di modificare una sessione.
RimuoviSessione.php	Controller che permette al docente di rimuovere una sessione.
TerminaSessione.php	Controller che permette al docente di terminare una sessione manualmente.
GestoreDataServer.php	Controller ritorna alla pagina richiedente la data attuale.

### 2.1.2.9 Package Utente



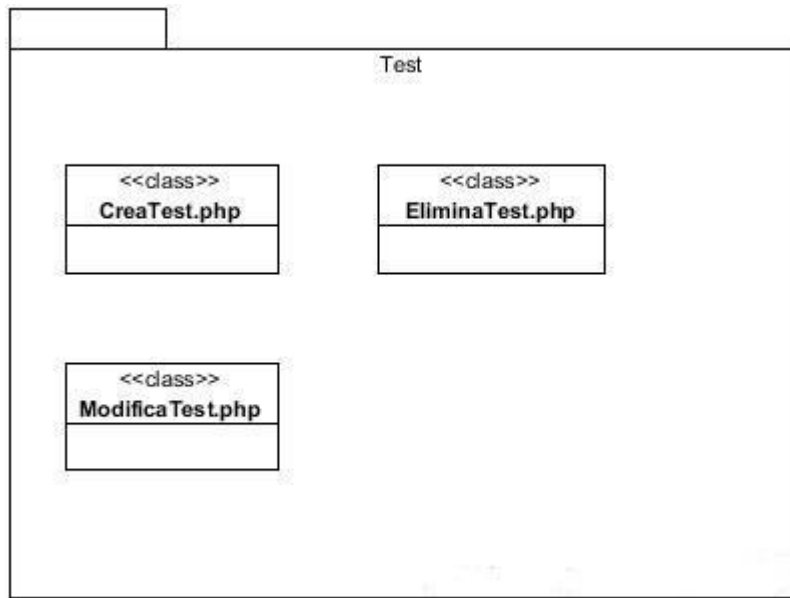
ModificaProfilo.php	Controller che permette la gestione della modifica di un profilo utente
---------------------	---

### 2.1.2.10 Package Statistiche



GetDomForStat.php	Controller per la gestione delle statistiche delle domande.
GetTestForStat.php	Controller per la gestione delle statistiche dei test.

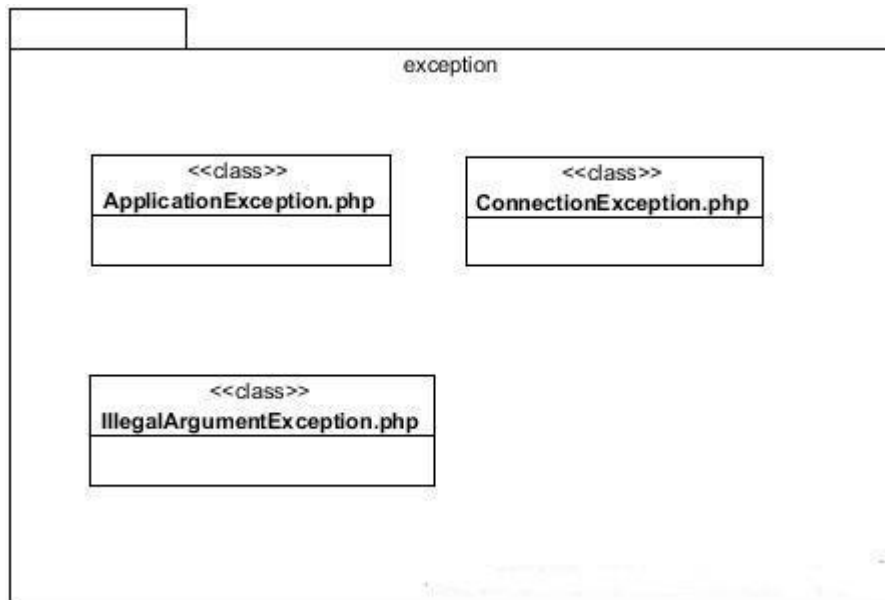
### 2.1.2.11 Package Test



CreaTest.php	Controller per la creazione del test (Manuale/Random)
EliminaTest.php	Controller per la rimozione del test.
ModificaTest.php	Controller per la modifica del test.

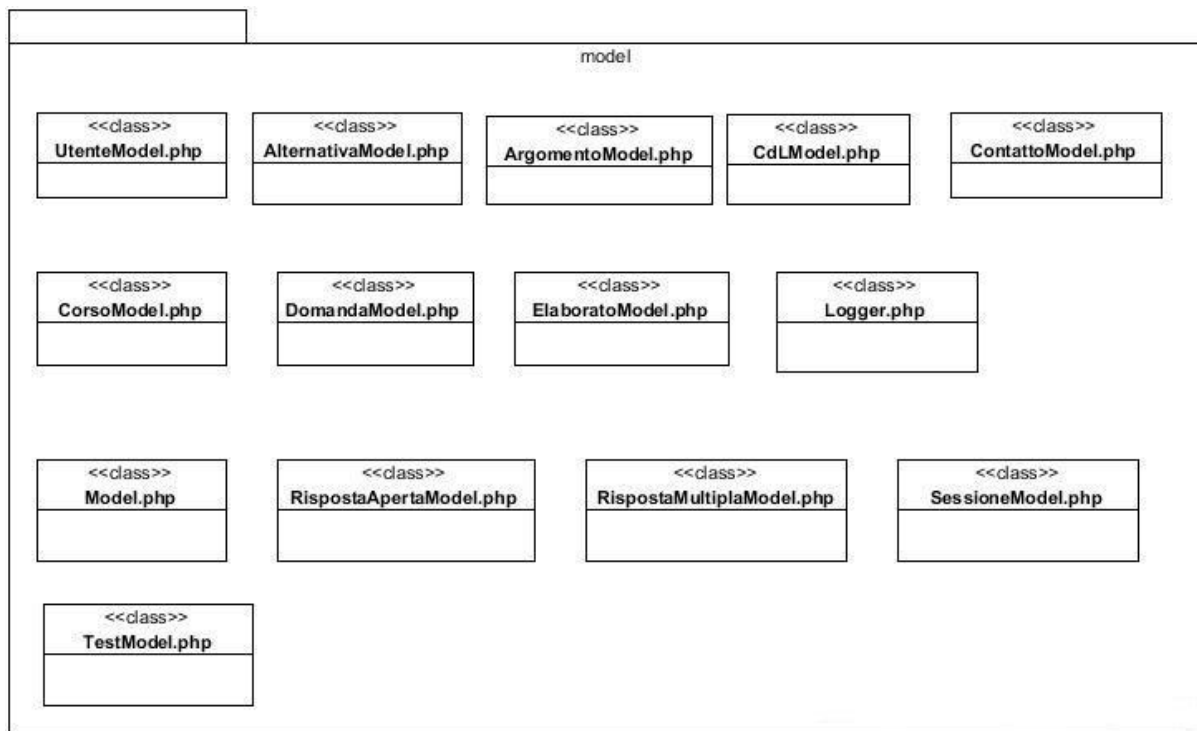


### 2.1.3 Package exception



ApplicationException.php	L'eccezione che viene lanciata quando non viene eseguita una query.
ConnectionException.php	L'eccezione che viene lanciata quando non c'è connessione al database.
IllegalArgumentException.php	L'eccezione che viene lanciata quando i parametri passati ad un metodo non sono corretti.

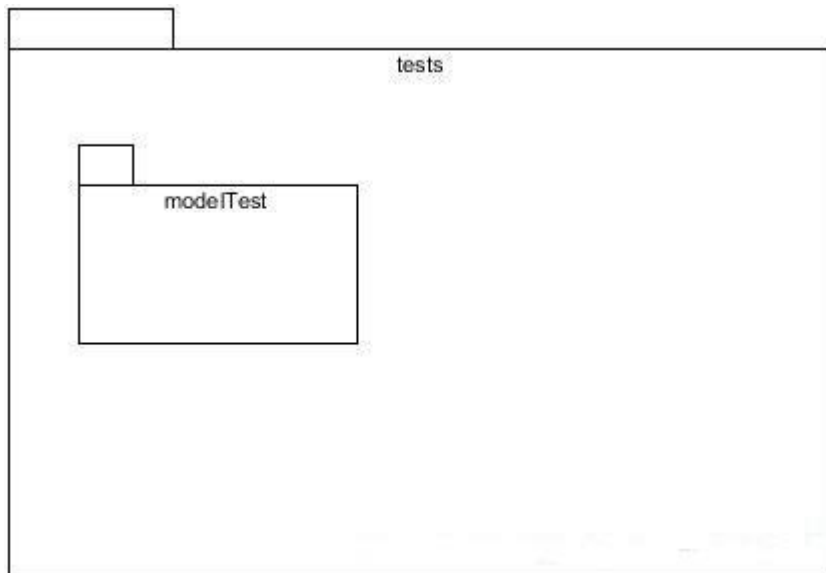
## 2.1.4 Package model



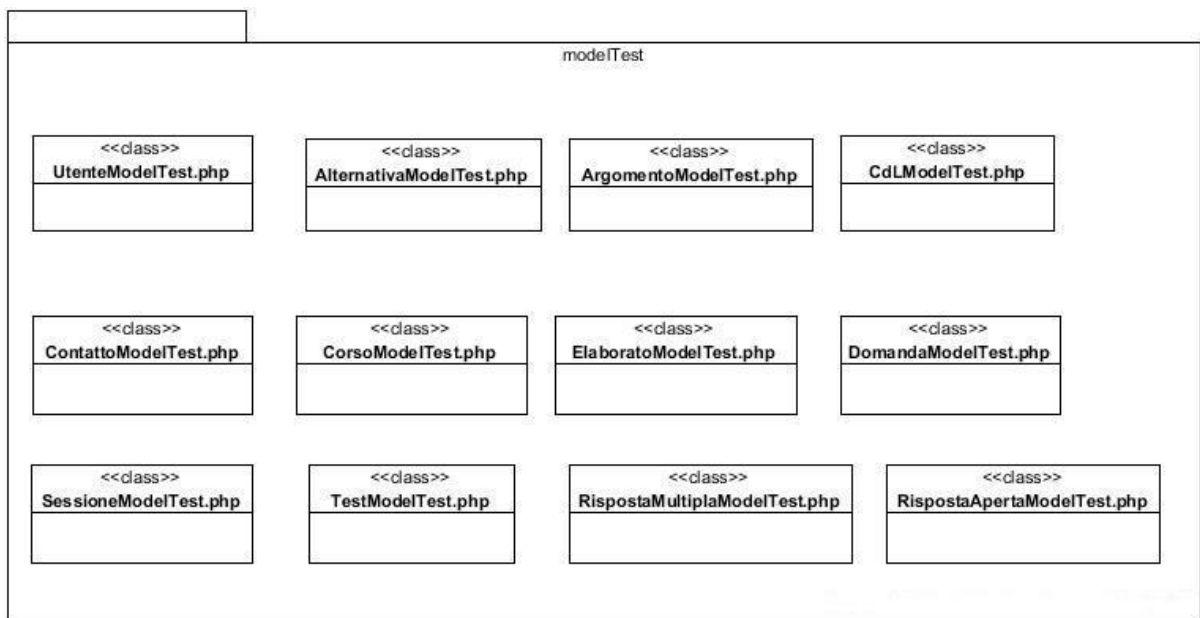
UtenteModel.php	Il model che effettua tutte le query riguardanti le funzionalità degli account, interfacciandosi al db al quale è connesso
AlternativaModel.php	Il model che effettua tutte le query riguardanti le funzionalità delle alternative di una domanda multipla, interfacciandosi al db al quale è connesso
ArgomentoModel.php	Il model che effettua tutte le query riguardanti le funzionalità degli argomenti di un corso, interfacciandosi al db al quale è connesso
CdLModel.php	Il model che effettua tutte le query riguardanti le funzionalità dei corsi di laurea, interfacciandosi al db al quale è connesso
ContattoModel.php	Il model che effettua tutte le query riguardanti le funzionalità legate ai contatti di un utente, interfacciandosi al db al quale è connesso
CorsoModel.php	Il model che effettua tutte le query riguardanti le funzionalità legate ai corsi di un CdL, interfacciandosi al db al quale è connesso
DomandaModel.php	Il model che effettua tutte le query riguardanti le funzionalità legate alle domande di un

	argomento, interfacciandosi al db al quale è connesso
ElaboratoModel.php	Il model che effettua tutte le query riguardanti le funzionalità degli elaborati, interfacciandosi al db al quale è connesso
Logger.php	Il model che effettua tutte le query riguardanti le funzionalità del logger
Model.php	Il model che effettua la connessione al db
RispostaApertaModel.php	Il model che effettua tutte le query riguardanti le funzionalità legate alle risposte aperte di un elaborato, interfacciandosi al db al quale è connesso
RispostaMultiplaModel.php	Il model che effettua tutte le query riguardanti le funzionalità legate alle risposte multiple di un elaborato, interfacciandosi al db al quale è connesso
SessioneModel.php	Il model che effettua tutte le query riguardanti le funzionalità legate alle sessioni di un corso, interfacciandosi al db al quale è connesso
TestModel.php	Il model che effettua tutte le query riguardanti le funzionalità legate ai test di un corso, interfacciandosi al db al quale è connesso

## 2.1.5 Package tests



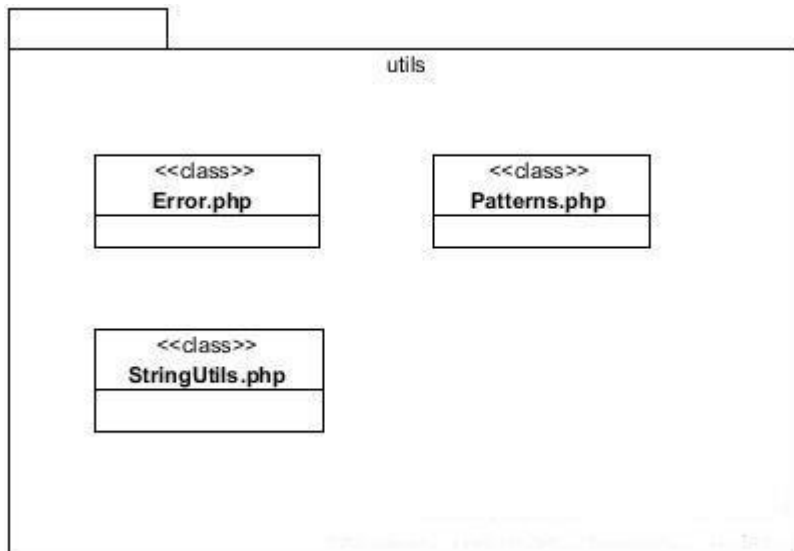
### 2.1.5.1 Package modelTest



UtenteModelTest.php	La classe che consente di testare i metodi del model Utente.
AlternativaModelTest.php	La classe che consente di testare i metodi del model Alternativa.
ArgomentoModelTest.php	La classe che consente di testare i metodi del model Argomento.

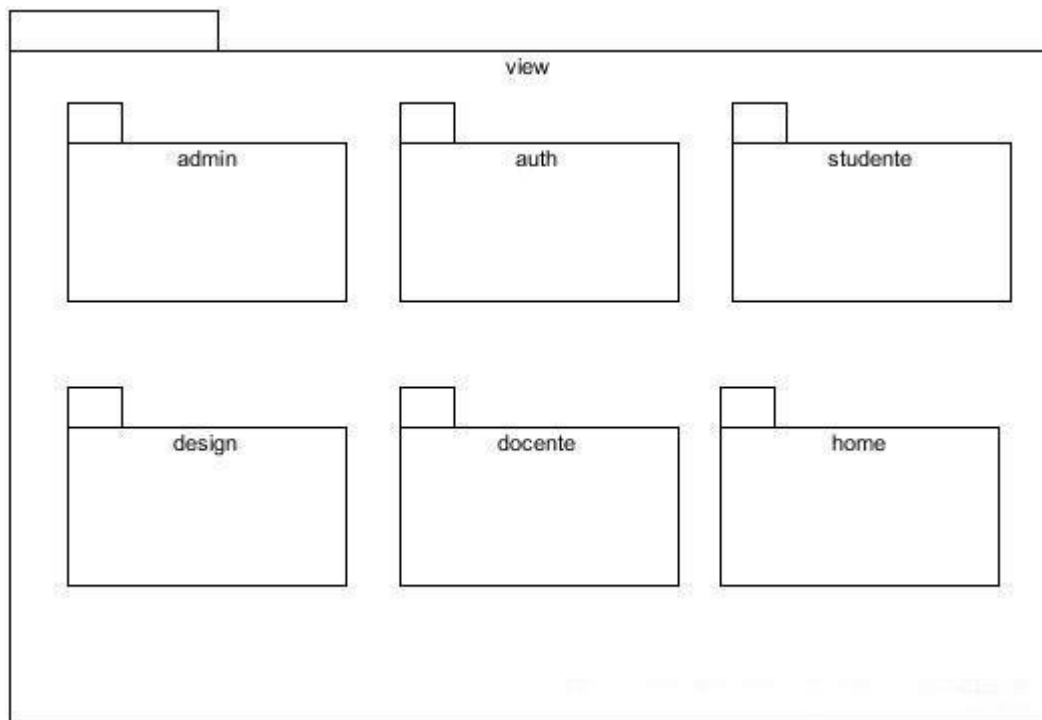
CdLModelTest.php	La classe che consente di testare i metodi del model CdL.
ContattoModelTest.php	La classe che consente di testare i metodi del model Contatto.
CorsoModelTest.php	La classe che consente di testare i metodi del model Corso.
ElaboratoModelTest.php	La classe che consente di testare i metodi del model Elaborato.
DomandaModelTest.php	La classe che consente di testare i metodi del model Domanda.
SessioneModelTest.php	La classe che consente di testare i metodi del model Sessione.
TestModelTest.php	La classe che consente di testare i metodi del model Test.
RispostaMultiplaModelTest.php	La classe che consente di testare i metodi del model RispostaMultipla.
RispostaApertaModelTest.php	La classe che consente di testare i metodi del model RispostaAperta.

## 2.1.6 Package utils

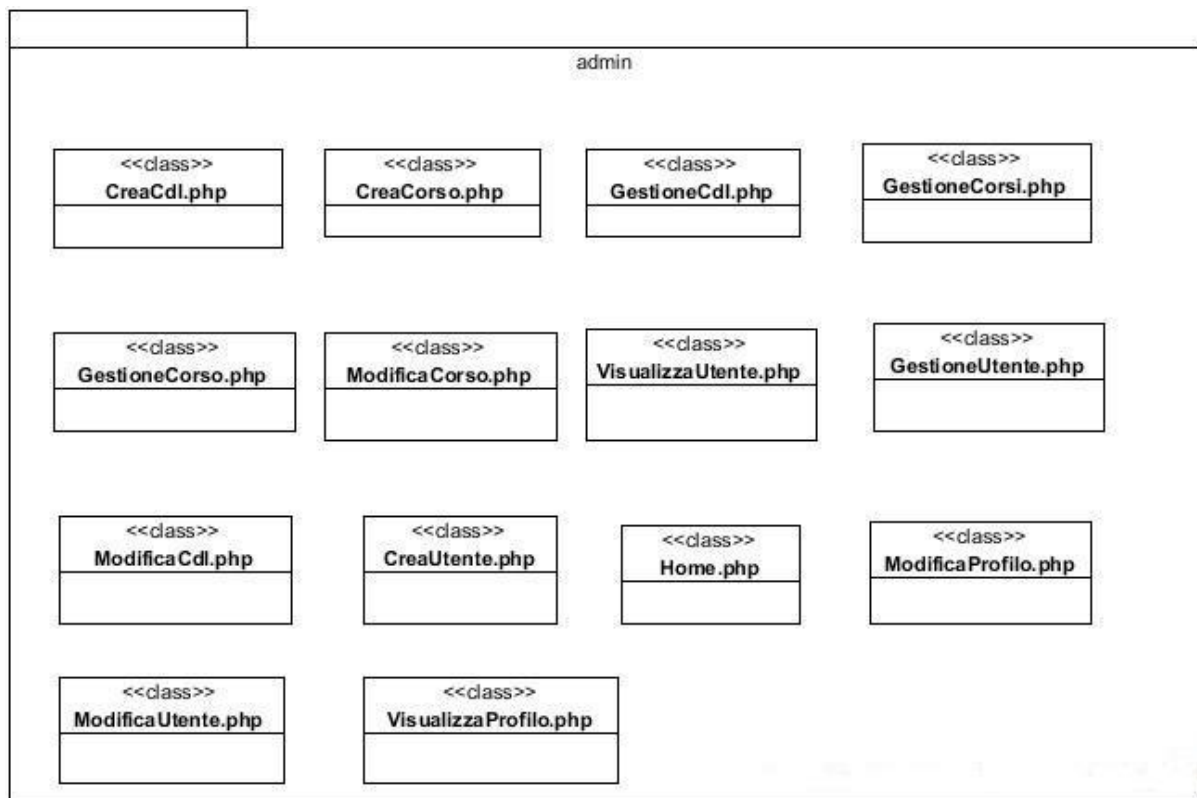


Error.php	La classe che contiene tutte le costanti stringhe utilizzate nei model per quanto riguarda le eccezioni.
Patterns.php	La classe che contiene tutte le espressioni regolari sottoforma di stringhe costanti.
StringUtils.php	La classe contiene dei metodi statici necessari per elaborare le stringhe

## 2.1.7 Package view



### 2.1.7.1 Package admin

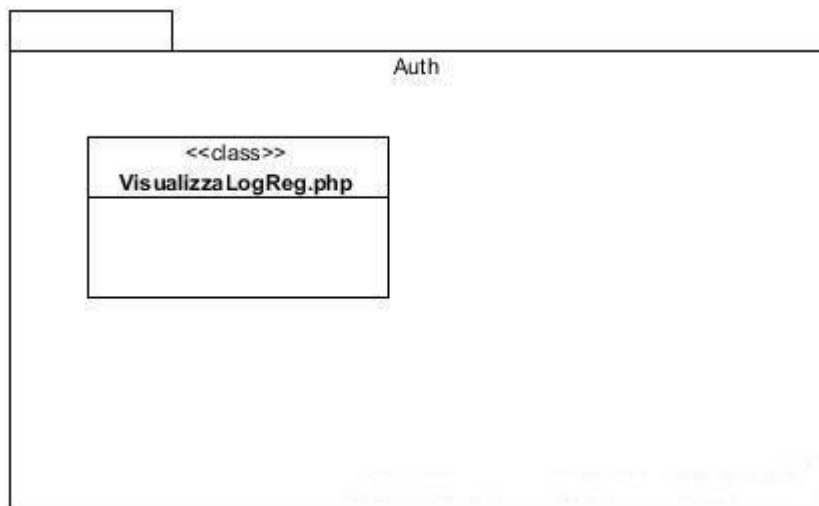


CreaCdl.php	La view che consente all'admin di creare un cdl
CreaCorso.php	La view che consente all'admin di creare un corso
CreaUtente.php	La view che consente all'admin di creare un utente
GestioneCdl.php	La view che consente all'admin di gestire un cdl
GestioneCorsi.php	La view che consente all'admin di gestire i corsi
GestioneCorso.php	La view che consente all'admin di gestire un corso di un cdl
ModificaCdl.php	La view che consente all'admin di modificare un Cdl
ModificaCorso.php	La view che consente all'admin di modificare un corso
ModificaProfilo.php	La view che consente all'admin di modificare il suo profilo



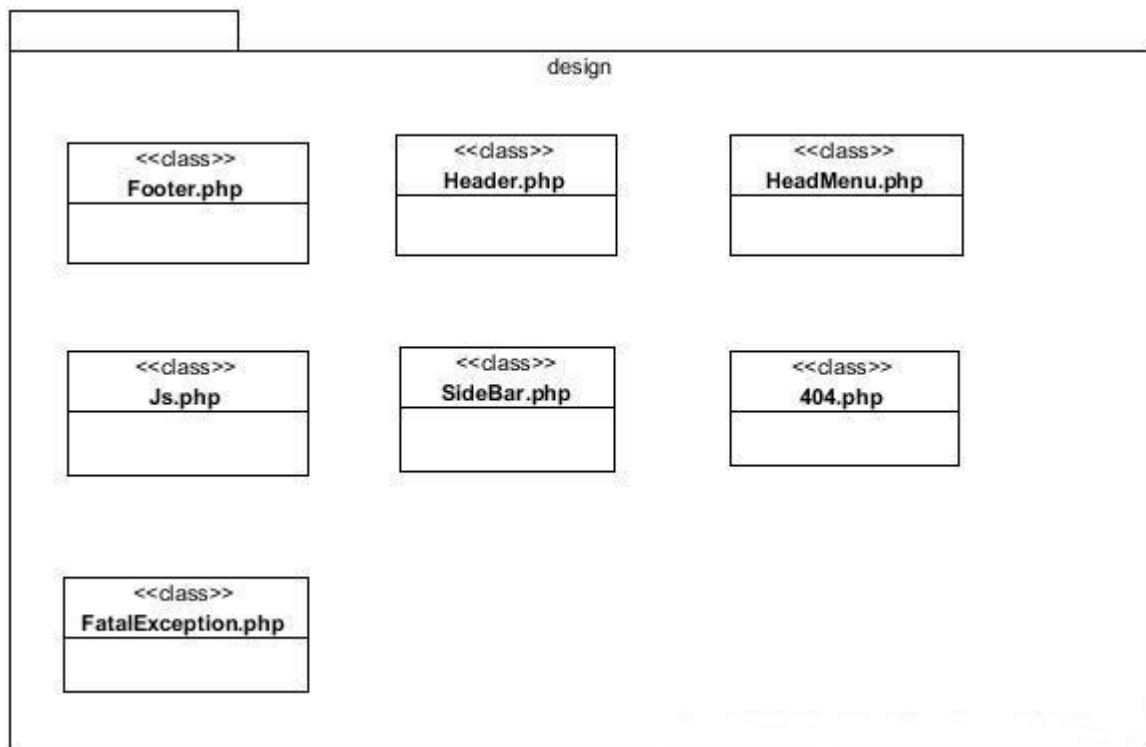
ModificaUtente.php	La view che consente all'admin di modificare un utente
VisualizzaUtente.php	La view che consente all'admin di visualizzare il profilo di un utente
GestioneUtente.php	La view che consente all'admin di gestire un utente
VisualizzaProfilo.php	La view che consente all'admin di visualizzare il profilo

### 2.1.7.2 Package auth



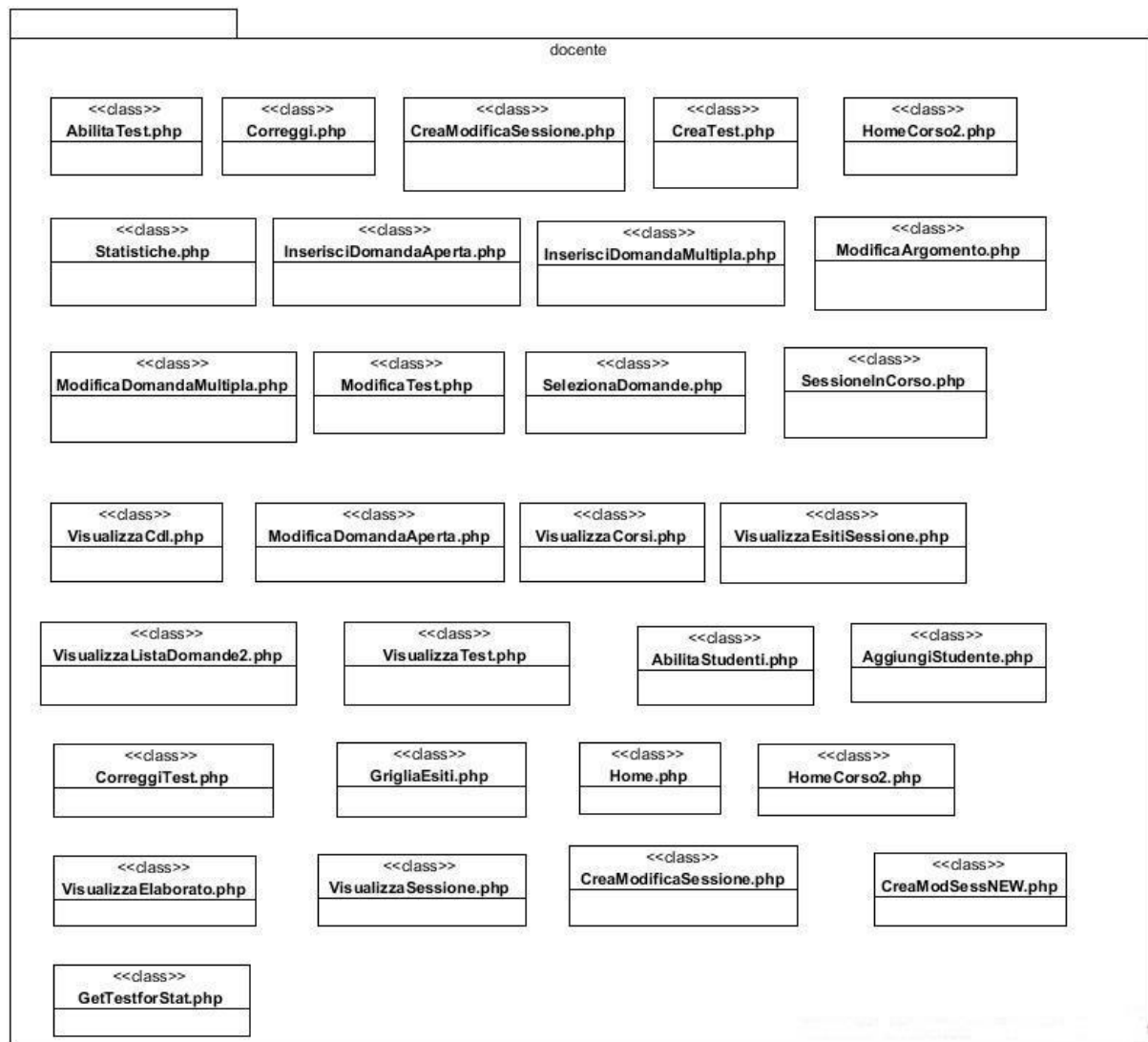
VisualizzaLogReg.php	Gestisce tutti i log della registrazione
----------------------	--

### 2.1.7.3 Package design



Footer.php	Contiene il footer comune a tutte le pagine
Header.php	Contiene l'header comune a tutte le pagine
HeadMenu.php	Contiene il menu superiore comune a tutte le pagine
Js.php	Contiene gli include di JS comune a tutte le pagine
SideBar.php	Contiene menù a sinistra comune a tutte le pagine
404.php	View che visualizza l'errore 404 pagina non trovata
FatalException.php	View che visualizza il messaggio di errore interno del sistema.

### 2.1.7.4 Package docente

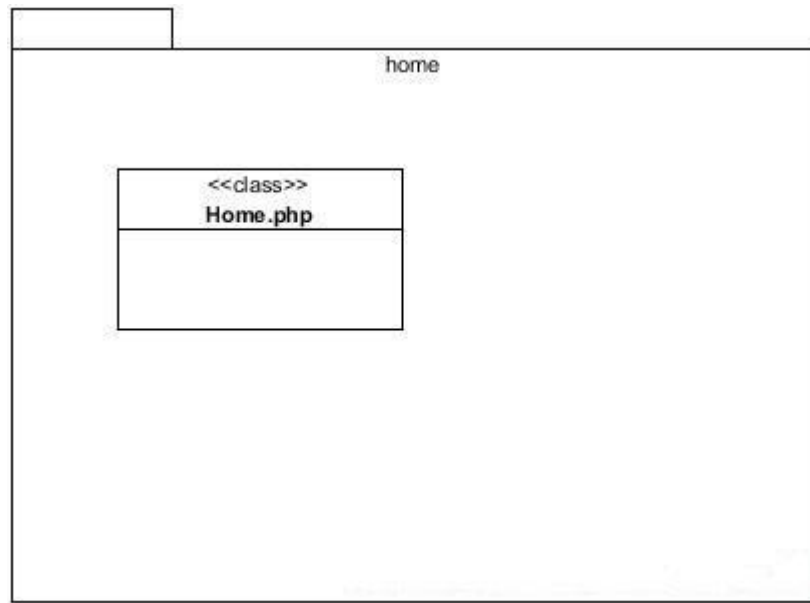


AbilitaStudenti.php	La view che consente al docente di abilitare gli studenti alla sessione
AggiungiStudiante.php	La view che consente al docente di aggiungere uno studente alla sessione
AbilitaTest.php	La view che consente al docente di associare i test ad una sessione
CorreggiTest.php	La view che consente al docente di correggere l'elaborato di uno studente
GrigliaEsiti.php	La view che consente al docente di visualizzare gli esiti di un test
VisualizzaElaborato.php	La view che consente al docente di visualizzare l'elaborato di un test di uno studente

VisualizzaSessione.php	La view che consente al docente di visualizzare una sessione
CreaModificaSessione.php	La view che consente al docente di creare e modificare una sessione
CreaTest.php	La view che consente al docente di creare un test
HomeCorso2.php	La view che consente al docente di visualizzare l'home del corso
InserisciArgomento.php	La view che consente al docente di creare un argomento
InserisciDomandaAperta.php	La view che consente al docente di creare una domanda aperta
InserisciDomandaMultipla.php	La view che consente al docente di creare una domanda multipla
ModificaArgomento.php	La view che consente al docente di modificare un argomento
ModificaDomandaMultipla.php	La view che consente al docente di modificare una domanda multipla
ModificaTest.php	La view che consente al docente di modificare un test
SelezioneDomande.php	La view che consente al docente di selezionare una domanda per un test
SessioneinCorso.php	La view che consente al docente di visualizzare i dettagli di una sessione in corso
VisualizzaCdl.php	La view che consente al docente di visualizzare i dettagli di un cdl
ModificaDomandaAperta.php	La view che consente al docente di modificare una domanda aperta
VisualizzaCorsi.php	La view che consente al docente di visualizzare i corsi di un cdl
VisualizzaEsitiSessione.php	La view che consente al docente di visualizzare gli esiti di una sessione
VisualizzaListaDomande2.php	La view che consente al docente di visualizzare le domande di un corso
VisualizzaTest.php	La view che consente al docente di visualizzare i test di un corso
Home.php	La view che consente al docente di visualizzare

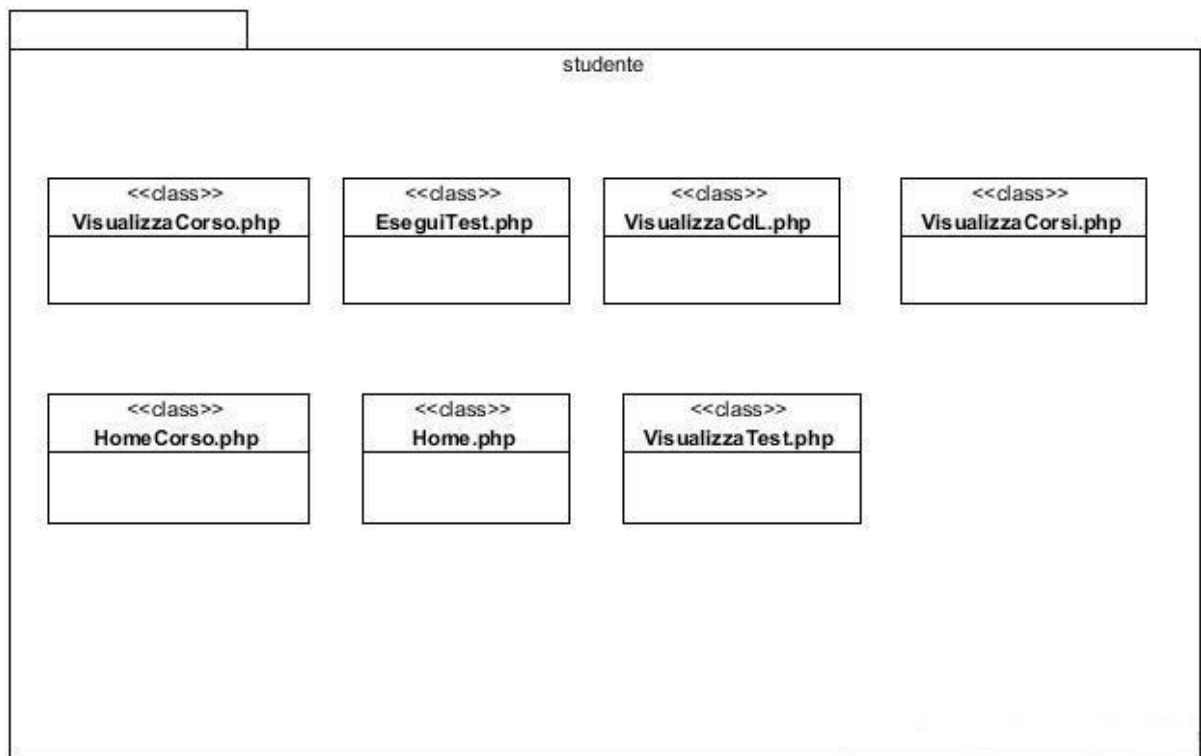
	la sua home
--	-------------

#### 2.1.7.5 Package home



Home.php	La homePage di SimplEx
----------	------------------------

### 2.1.7.6 Package studente

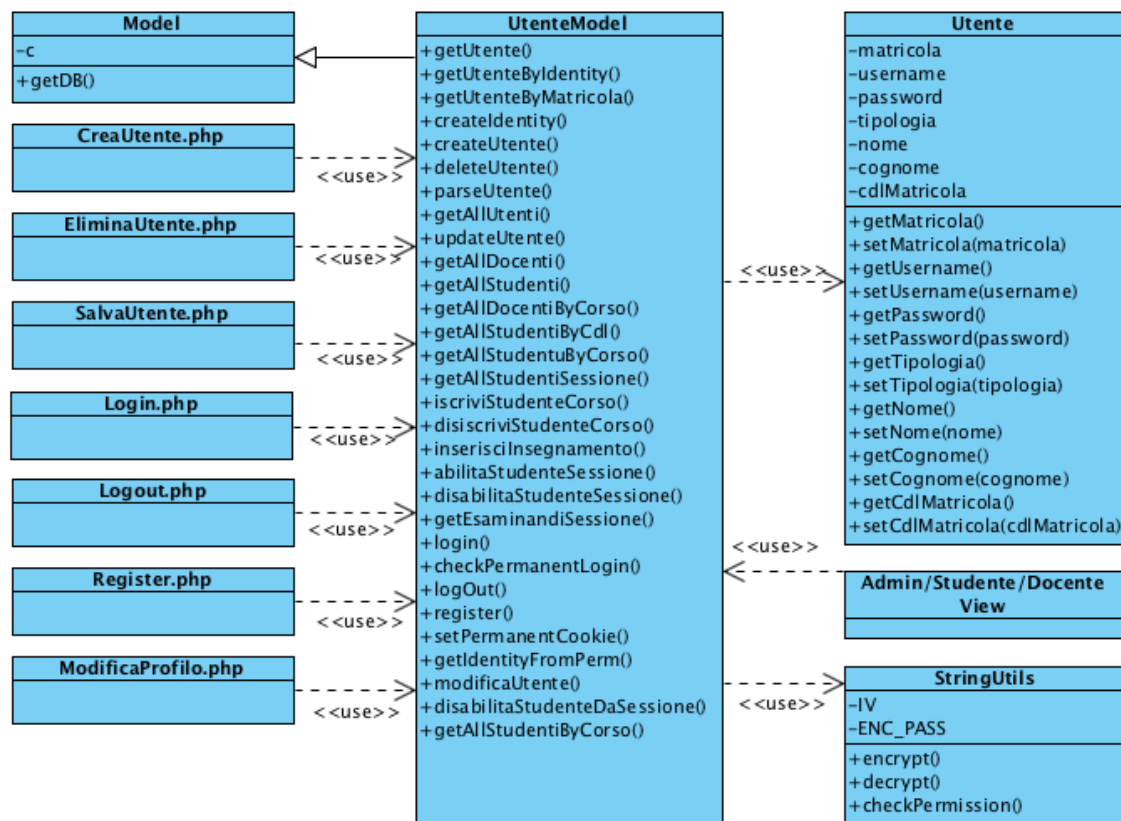


EseguiTest.php	View che consente allo studente di eseguire un test
Home.php	Home dello studente
HomeCorso.php	View che visualizza i dettagli del corso, quali le sessioni per i quali si è abilitati, test eseguiti e relativi esiti dello studente
VisualizzaCdL.php	View che consente allo studente di visualizzare tutti i cdL
VisualizzaCorsi.php	View che consente allo studente di visualizzare i corsi del cdL al quale è iscritto
VisualizzaCorso.php	View che consente allo studente di visualizzare i dettagli di un corso al quale è iscritto
VisualizzaTest.php	View che consente allo studente di visualizzare i dettagli di un test

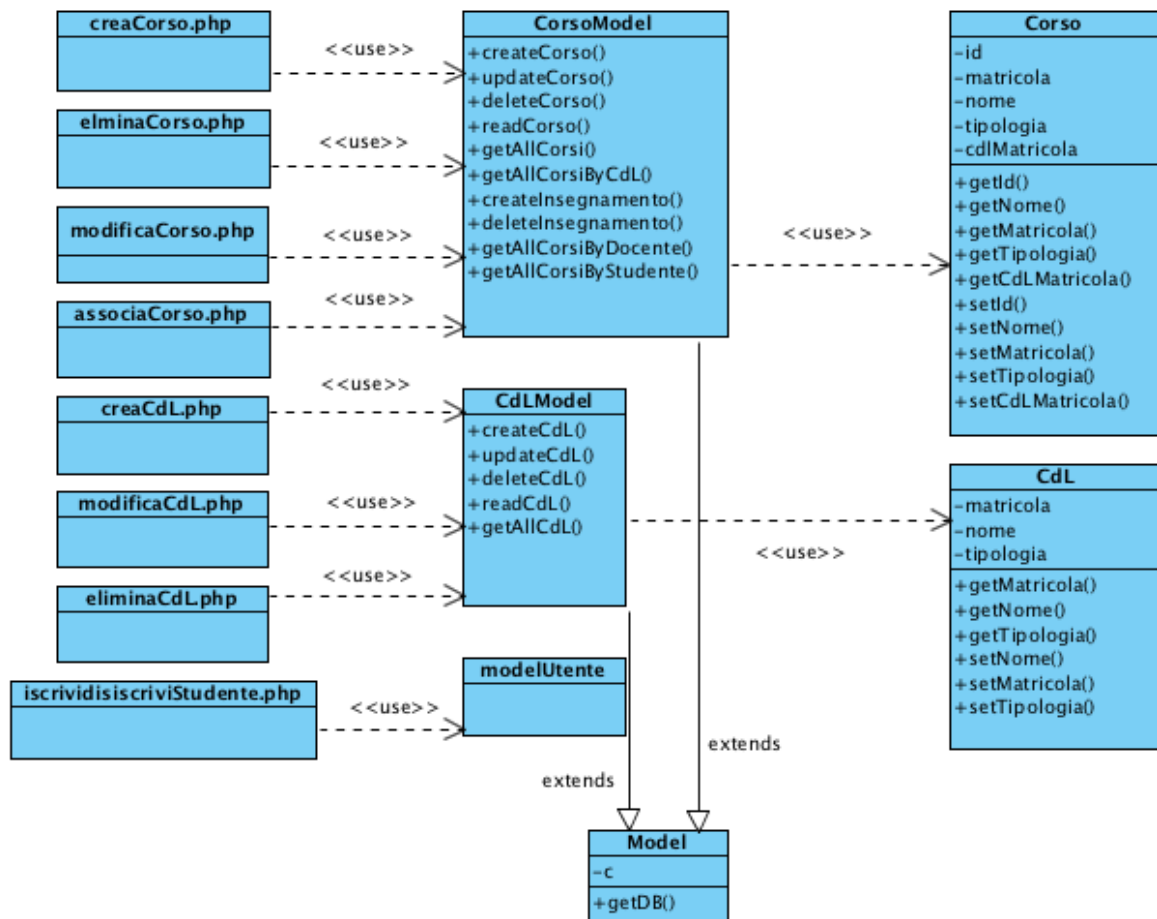
### 3. Class interfaces

In allegato all'ODD c'è la cartella "Documentazione" contenente la documentazione phpDoc.

#### 3.1 Gestione Utente

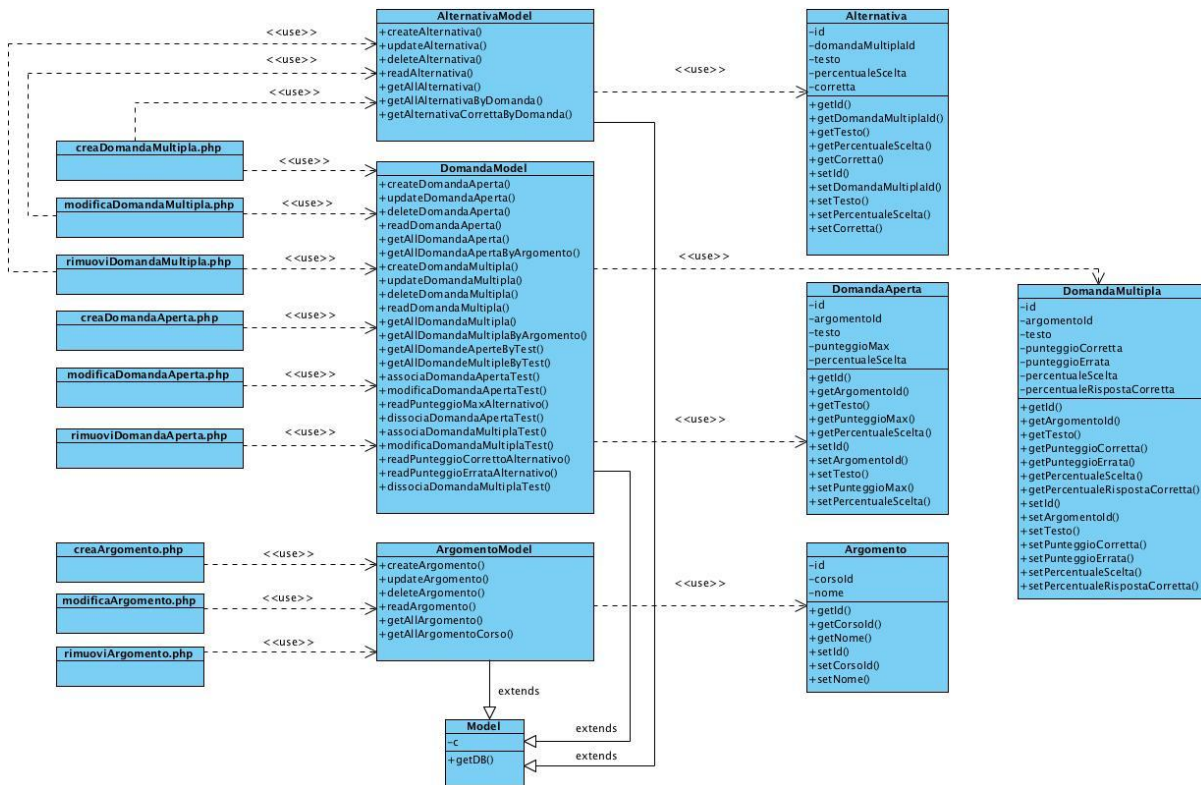


## 3.2 Gestione Corsi di Laurea

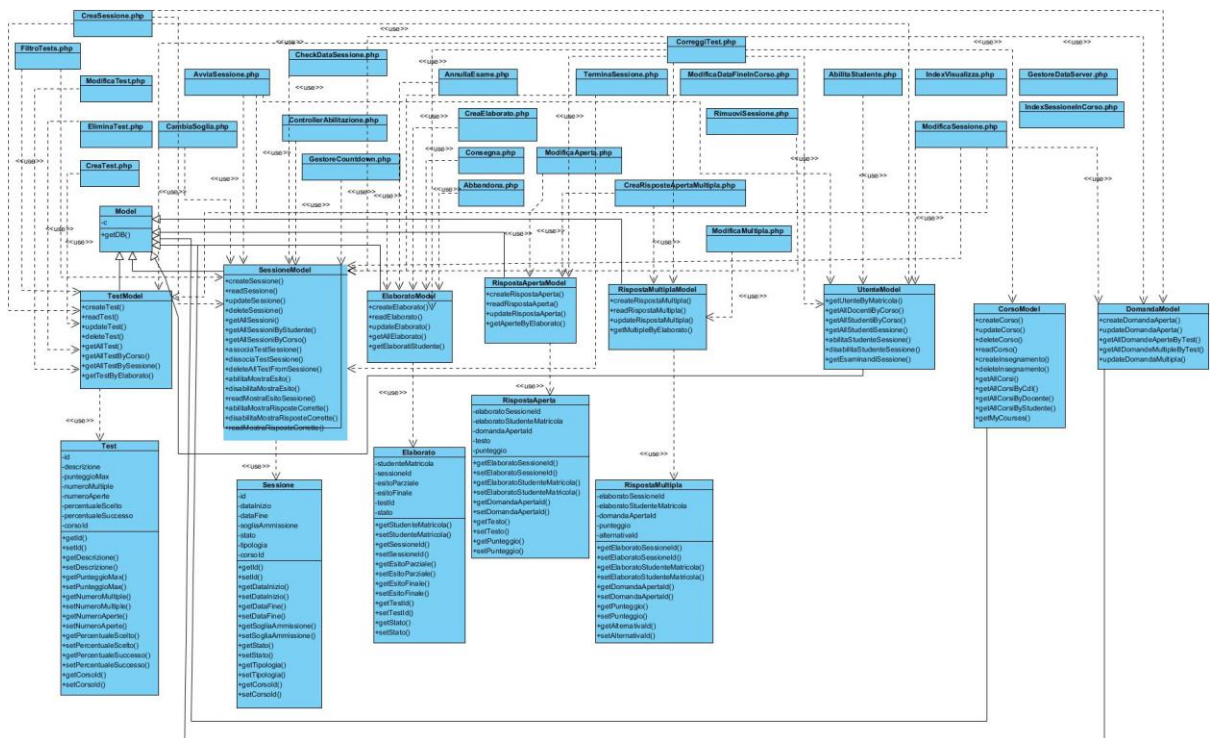




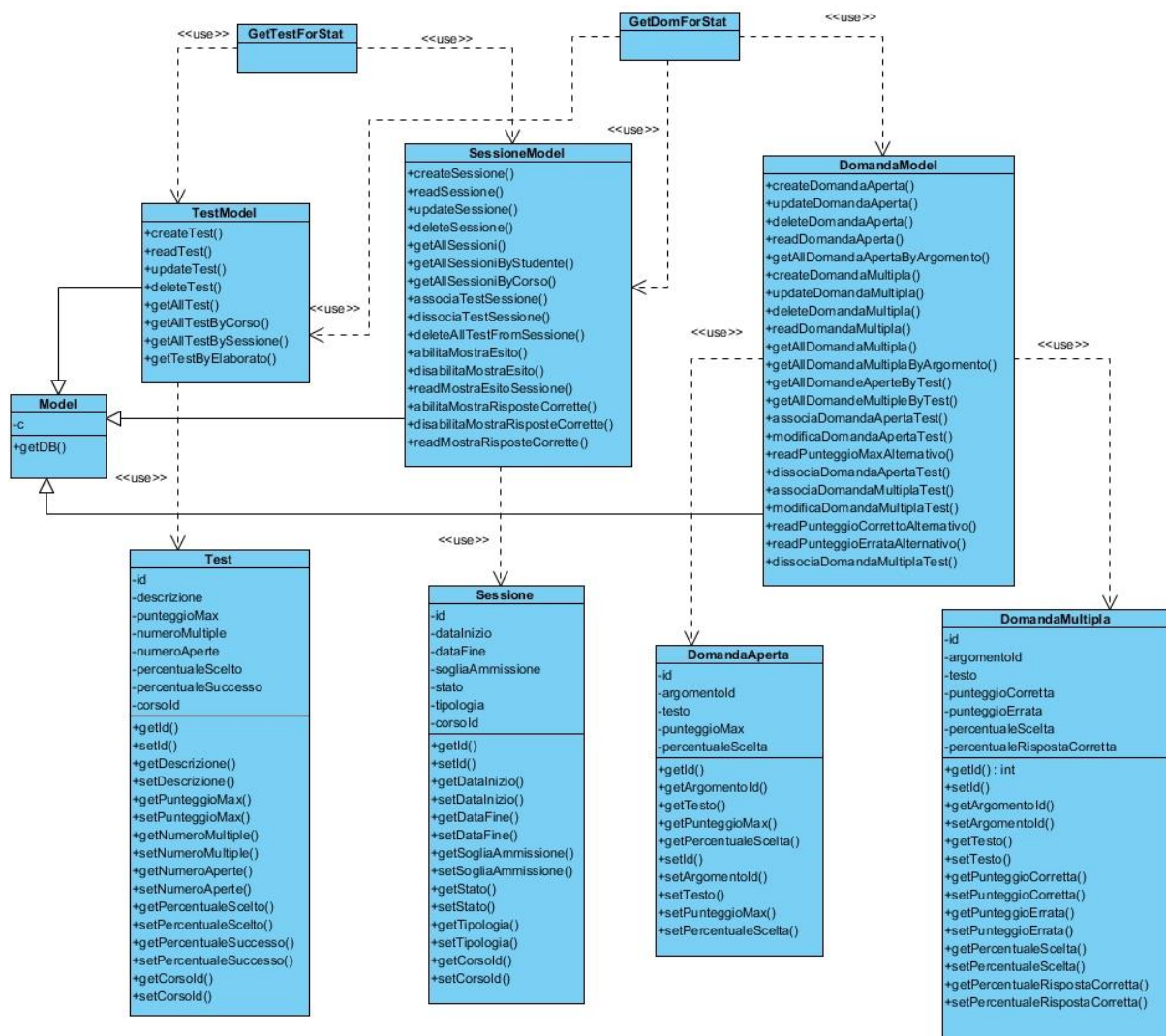
### 3.3 Gestione Argomenti



### 3.4 Gestione Test



### 3.5 Gestione Statistiche



## 4. Glossario

Identità utente:	L'identità utente è un hash, prodotto dalla funzione md5, che prende in input email, password e SALT
SimplEx:	Nome del sistema
Query:	Termine utilizzato per indicare l'interrogazione da parte di un utente di un database.
Cookie:	File di informazioni che i siti web memorizzano sul computer dell'utente di Internet durante la navigazione.