



Laurea Triennale in Informatica - Università di Salerno  
Corso di Ingegneria del Software - Prof.ssa F. Ferrucci



Easy Plan

# System Design Document

## EasyPlan

Riferimento	
Versione	2.4
Data	14/01/2019
Destinatario	Prof.ssa F. Ferrucci
Presentato da	Francesca Cerruto
Approvato da	



## Sommario

Revision History.....	3
Responsabilità.....	4
System Design Document del Progetto EasyPlan .....	5
1. Introduzione.....	5
1.1. Scopo del sistema.....	5
1.2. Design goal.....	5
1.3. Definizioni, acronimi ed abbreviazioni .....	9
1.4. Riferimenti.....	9
1.5. Panoramica.....	10
2. Architettura del sistema corrente.....	11
3. Architettura del sistema proposto .....	12
3.1. Panoramica.....	12
3.2. Decomposizione in sottosistemi .....	12
3.3. Diagramma di deployment .....	16
3.4. Mapping hardware/software.....	17
3.5. Gestione dei dati persistenti .....	17
3.6. Controllo degli accessi e sicurezza .....	20
3.7. Controllo globale del software.....	20
3.8. Condizioni limite.....	20
4. Servizi dei sottosistemi .....	21



## Revision History

Data	Versione	Descrizione	Autori
03/12/2018	0.1	Prima stesura	Francesco Aurilio Francesca Cerruto Carmine D'Angelo Andrea Junquera Pier Giuseppe Rotondo Francesco Tranzillo Emanuele Vitale
04/12/2018	1.0	Revisione prima stesura	Giuseppe Abagnale Roberto Gagliardi
05/12/2018	2.0	Modifica al design goal dell'adattabilità	Giuseppe Abagnale Roberto Gagliardi
08/12/2018	2.1	Modifica al database	Carmine D'Angelo, Francesco Tranzillo
19/12/2018	2.2	Aggiunta delle entità docente e insegnamento al database	Francesco Tranzillo
11/01/2019	2.3	Revisione Servizi dei Sottosistemi	Francesca Cerruto, Pier Giuseppe Rotondo
14/01/19	2.4	Revisione finale SDD	Francesca Cerruto



## Responsabilità

Artefatto	Autori
Introduzione, Design goal e Sistema corrente	Francesco Aurilio, Andrea Junquera, Emanuele Vitale
Sottosistemi e Mapping hardware/software	Francesca Cerruto, Pier Giuseppe Rotondo
Gestione dei dati persistenti	Carmine D'Angelo, Francesco Tranzillo
Accessi, Controllo e Condizioni limite	Francesco Aurilio, Andrea Junquera, Emanuele Vitale
Servizi dei sottosistemi	Francesca Cerruto, Pier Giuseppe Rotondo

# System Design Document del Progetto EasyPlan

## 1. Introduzione

### 1.1. Scopo del sistema

Il sistema che si intende realizzare ha l'obiettivo di offrire agli studenti dei Corsi di Laurea in Informatica dell'Università degli Studi di Salerno la possibilità di formulare un piano di studi personalizzato in base all'offerta formativa dell'anno accademico desiderato. Tutte le informazioni necessarie alla compilazione del piano potranno essere inserite nel sistema tramite un apposito lato amministrativo, che permette di accedere a diverse funzionalità per poter definire i piani di studi disponibili. In questo modo, lo studente sarà in grado di selezionare tutti gli esami desiderati in base alle proprie necessità e ai propri interessi, e avrà la possibilità di scaricare un documento in pdf che riporterà l'intero piano di studi da lui formulato, in modo che questo possa agevolarlo nel momento in cui andrà a compilare il proprio piano carriera sulla piattaforma Esse3. Il sistema proposto, infatti, non intende sostituirsi ai sistemi e ai documenti ufficiali forniti dall'ateneo e dal Dipartimento di Informatica, bensì vuole rappresentare un ulteriore supporto agli studenti.

### 1.2. Design goal

Di seguito, vengono descritti i design goal che devono essere rispettati nella realizzazione del sistema.

#### **Criteri di performance**

- Tempo di risposta:
  - o Il tempo necessario per la visualizzazione della lista degli esami prevista dal piano di studi selezionato non deve superare i 10 secondi.
- Memoria:
  - o La quantità di memoria occupata dal sistema dipende da quella necessaria al mantenimento del database.
  - o Per assicurare la consistenza dei dati, il sistema deve creare un nuovo database per ogni nuova offerta formativa.
  - o Nell'ambito della formulazione di un piano di studi, il sistema deve salvare le informazioni immesse dall'utente soltanto finché non viene chiusa la sessione corrente.



- Altro:
  - Il sistema deve essere responsive, supportando la visualizzazione su dispositivi aventi le seguenti risoluzioni: 900x1440 (laptop), 768x1024 (iPad), 375x812 (iPhone X), 360x740 (Samsung Galaxy S9/S9+).

### **Criteri di affidabilità**

- Robustezza:
  - Eventuali input non validi immessi dall'utente devono essere segnalati tramite degli opportuni messaggi di errore.
- Affidabilità:
  - Il sistema deve garantire che, durante la formulazione di un piano di studi, un utente non può selezionare uno stesso esame più di una volta.
  - Il sistema deve garantire che ogni offerta formativa non venga influenzata da eventuali modifiche alle altre offerte formative.
  - Il sistema deve garantire che, se un esame compare in più punti diversi della stessa offerta formativa, le sue informazioni risultino sempre consistenti.
  - Il sistema deve garantire che un esame venga eliminato definitivamente soltanto quando non compare più in nessun punto dell'offerta formativa corrente.
  - Il sistema non deve garantire la veridicità ufficiale dei dati presentati, né della loro conformità a quanto presentato da Esse3; questa responsabilità è da affidare all'amministratore o a chi per lui ha inserito i dati nel sistema.
- Disponibilità:
  - Il sistema deve risultare disponibile finché l'utente non ha concluso tutte le operazioni di cui necessita.
- Tolleranza agli errori:
  - Le operazioni che richiedono l'inserimento di dati da parte dell'amministratore non devono essere completate finché non viene rispettato il formato previsto.
  - Nel caso in cui il sistema diviene inutilizzabile durante la formulazione di un piano di studi, il sistema non deve fornire alcun modo per ripristinare le scelte effettuate fino a quel momento.



- Sicurezza:
  - Le funzionalità del lato amministrativo del sistema devono essere disponibili soltanto dopo l'autenticazione dell'utente.
  - Le credenziali dell'amministratore devono essere uniche, così da ridurre la possibilità che più utenti possano accedere nello stesso momento al lato amministrativo del sistema.

### **Criteri di costo**

- Costi di sviluppo:
  - Il costo complessivo del progetto ammonta ad una stima di 324 ore di lavoro da parte dei membri del team, le quali possono estendersi fino ad un massimo di 350 ore (50 ore per ogni membro).
- Costi di manutenzione:
  - Il costo di manutenzione del sistema, nel caso in cui questa venga richiesta, ammonta a 10 ore di lavoro per ogni membro del team di sviluppo.

### **Criteri di manutenzione**

- Estendibilità:
  - Il sistema deve essere progettato in maniera tale da poter aggiungere, in futuro, dei controlli più stringenti sulle operazioni del lato amministrativo del sistema, così da garantire con la massima sicurezza il rispetto dei vincoli imposti dall'ordinamento didattico (tra cui l'assenza di esami duplicati e l'ammontare ai 60 crediti annui).
- Modificabilità:
  - Il sistema deve essere progettato in maniera tale da poter permettere la modifica o il miglioramento delle funzionalità principali del lato amministrativo (aggiunta, modifica ed eliminazione delle entità).
- Adattabilità:
  - Il sistema deve basarsi su una struttura che risulti più generica possibile, così che possa essere facilmente adattato, eventualmente, anche agli ordinamenti didattici di dipartimenti diversi da quello di Informatica.
- Portabilità:
  - Il sistema deve essere fruibile su tutti i dispositivi mobile e desktop in maniera indipendente dal sistema operativo o dall'hardware utilizzato.



- Tracciabilità dei requisiti:
  - La tracciabilità dei requisiti dev'essere garantita dall'utilizzo di una matrice di tracciabilità, che permette di ricondurre ogni artefatto del progetto al requisito associato.

### **Criteri utenti finali**

- Utilità:
  - Il sistema deve presentare una struttura e un meccanismo di utilizzo tale che possano adattarsi alla struttura dei piani di studi presentati dalle offerte formative degli ultimi cinque anni.
  - Il sistema deve presentare le informazioni fornite dalla Guida dello Studente che sono state prese in considerazione nell'ambito del progetto in maniera tale che risultino conformi a quanto descritto dalla Guida.
- Usabilità:
  - Il sistema deve riuscire a rendere la consultazione delle informazioni più semplice rispetto a come avviene per la Guida dello Studente.
  - Nell'ambito delle funzionalità del lato amministrativo, il sistema deve fornire la possibilità di precaricare dei dati già presenti per ridurre la quantità di dati da inserire.
  - Nell'ambito della formulazione di un piano di studi, il sistema non deve fornire la possibilità di ripristinare le scelte effettuate durante un precedente utilizzo.
  - Il sistema deve offrire all'utente una navigabilità tale da renderlo sempre consapevole della sezione in cui si trova e prevedere sempre la possibilità di tornare indietro a un punto precedente della navigazione.

### **Trade-off**

- Usabilità vs Memoria:
  - Poiché il sistema proposto intende fornire un supporto alternativo alla Guida dello Studente, un alto grado di usabilità risulta essenziale per il successo del progetto. Questo deve essere ottenuto anche a discapito di una dimensione del database più elevata.
- Affidabilità vs Memoria:
  - Poiché il sistema proposto deve fornire agli studenti un supporto efficace, esso deve garantire che le informazioni fornite agli utenti non risultino inconsistenti. Questo deve essere ottenuto anche a discapito di una dimensione del database più elevata.





- Costo di sviluppo vs Affidabilità:
  - o Poiché il sistema proposto deve fornire agli studenti un supporto efficace, esso deve garantire che il risultato delle operazioni degli utenti risulti corretto, assicurandosi che un piano di studi formulato da uno studente non contenga delle difformità strutturali. Questo sarebbe possibile soltanto imponendo una diversa serie di controlli sulle funzionalità del lato amministrativo del sistema, che impediscano di fornire agli studenti dei piani di studi non validi. Tuttavia, poiché l'implementazione di tali funzionalità risulterebbe eccessivamente dispendioso per il budget disponibile, queste non verranno implementate ma saranno tenute in considerazione per degli eventuali sviluppi futuri del sistema.
- Sicurezza vs Usabilità:
  - o Poiché il lato amministrativo del sistema potrebbe essere utilizzato contemporaneamente da più individui, con le stesse credenziali, esso avrebbe bisogno di un meccanismo di gestione della concorrenza per garantire che più operazioni eseguite nello stesso momento non provochino inconsistenze tra i dati. Tuttavia, poiché l'implementazione di tali funzionalità risulterebbe eccessivamente dispendioso per il budget disponibile, le credenziali di accesso al lato amministrativo del sistema saranno uniche, così da limitare l'accesso contemporaneo agli amministratori.

### 1.3. Definizioni, acronimi ed abbreviazioni

---

Nell'ambito del progetto, si farà spesso riferimento ad offerte formative, anni accademici, corsi di laurea e curricula. In particolare, un'offerta formativa si riferisce ad uno specifico anno accademico ed include l'intera offerta di diversi corsi di laurea. Nel caso dell'Università degli Studi di Salerno, la facoltà di Informatica prevede un Corso di Laurea Triennale (di tre anni) e uno Magistrale (di due anni). Ciascuno di questi può prevedere diversi curricula, i quali definiscono gli esami obbligatori e gli esami opzionali da sostenere per completare il relativo percorso di studi.

### 1.4. Riferimenti

---

Per stilare il presente documento, si è preso come riferimento il file *Template SDD* presente nel corso di Ingegneria del Software della piattaforma di e-learning della facoltà di Informatica. Inoltre, è stato consultato il libro di testo *Object-Oriented Software Engineering Using UML, Patterns and Java: Third Edition*, di



Bernd Bruegge ed Allen H. Dutoit. Infine, sono presenti alcuni riferimenti al documento di analisi dei requisiti *Requirement Analysis Document - EasyPlan*.

## 1.5. Panoramica

Dopo questa breve sezione di introduzione, il presente documento seguirà con la descrizione dell'architettura del sistema corrente al punto 2, mentre al punto 3 verrà illustrata quella del sistema che si intende realizzare. In particolare, questa sezione descriverà la decomposizione in sottosistemi, la corrispondenza tra hardware e software, la gestione dei dati persistenti, il controllo degli accessi, la sicurezza, il flusso di controllo e la gestione delle condizioni limite del sistema proposto. Infine, il punto 4 descriverà i servizi offerti dai sottosistemi individuati.



## 2. Architettura del sistema corrente

---

Alla situazione attuale, le funzionalità che *EasyPlan* intende fornire vengono offerte dalla Guida dello Studente e dal sito web d'ateneo. Questo tipo di sistema si basa comunemente sull'architettura MVC che include le seguenti componenti:

- Model: gli oggetti modello rappresentano le parti dell'applicazione che implementano la logica per il dominio dei dati dell'applicazione.
- View: le visualizzazioni sono i componenti che consentono di visualizzare l'interfaccia utente dell'applicazione. Questa interfaccia utente viene creata in base ai dati del modello.
- Controller: i controller sono i componenti che gestiscono l'interazione dell'utente, utilizzano il modello e selezionano infine una visualizzazione per il rendering dell'interfaccia utente.

## 3. Architettura del sistema proposto

### 3.1. Panoramica

---

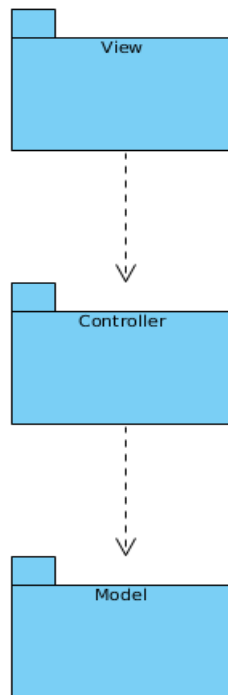
Di seguito, vengono illustrati i risultati della fase di progettazione del sistema proposto. In particolare, viene descritta l'architettura del sistema, stabilita in base alla sua struttura da sito web e alle funzionalità che deve offrire. Particolare attenzione è stata riposta nella gestione dei dati persistenti, i quali risultano fondamentali allo scopo del sistema proposto.

### 3.2. Decomposizione in sottosistemi

---

La decomposizione prevista per il sistema proposto è composta da tre layer che si occupano di gestirne i vari aspetti e funzionalità. Essa si basa sull'architettura MVC, i cui layer sono i seguenti:

1. View: mostra all'utente gli oggetti del dominio applicativo (i dati contenuti nel Model) e si occupa dell'interazione con l'utente; raccoglie e gestisce l'interfaccia grafica e gli eventi generati dall'utente.
2. Controller: responsabile della sequenza di interazioni con l'utente, riceve i comandi dell'utente attraverso il View e li esegue modificando lo stato degli altri due componenti; si occupa della gestione della logica del sistema.
3. Model: mantiene la conoscenza del dominio applicativo, fornendo le operazioni per accedere ai dati utili all'applicazione; si occupa della gestione e dello scambio dei dati tra i sottosistemi ed è in contatto diretto con il database.



*Figura 1 Architettura Model-View-Controller*

Dopo un'attenta analisi funzionale, si è stabilito di suddividere il sistema come illustrato di seguito. Questa decisione è dettata dal fatto che, per la realizzazione di un sito web multiplatforma, la suddivisione in sottosistemi prevista da questa architettura corrisponde in maniera ideale al partizionamento logico pensato per il sistema.

Le funzionalità sono suddivise in base all'area di gestione e prevedono un'interfaccia intermedia tra i sistemi della logica di business (Controller) ed il database (a cui ci si interfaccia grazie al layer del Model), in modo da non rendere necessari cambiamenti significativi nel caso in cui ci sia la necessità di modificare la base di dati.

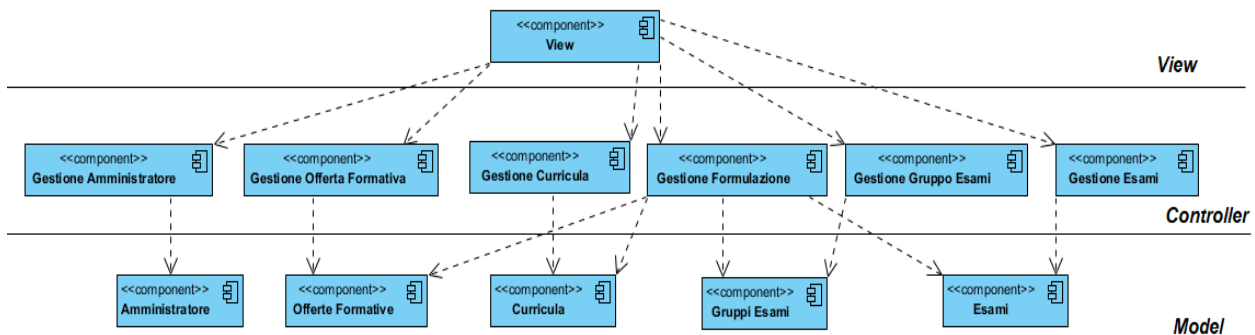


Figura 2 Diagramma della suddivisione in sottosistemi

L'intero sistema si compone dei seguenti dodici componenti.

Di seguito, viene descritta nel dettaglio ciascuna componente, specificandone la sua descrizione, gli oggetti del sistema che la compongono e i requisiti non funzionali e/o gli use case a cui fanno riferimento.

Nome componente	Descrizione componente	Oggetti correlati	Use Case e/o requisiti non funzionali correlati
<b>View</b>	Contiene le classi che si occupano di presentare graficamente i dati all'utente.	Tutti gli oggetti Boundary, Compila il tuo piano su Esse3, Scarica Piano	UC_EP_FPS UC_EP_SPS UC_EP_CPS NFR[1] NFR[5] NFR[14]
<b>Gestione Amministratore</b>	Permette all'amministratore di effettuare l'accesso alla sua sezione dedicata.	Login control	UC_EP_LGA NFR[6]
<b>Gestione offerta formativa</b>	Raggruppa le classi che si occupano della gestione dell'offerta formativa: aggiunta, rimozione e visibilità (visibile o non visibile dall'utente).	Aggiunta offerta formativa control, Eliminazione offerta formativa control	UC_EP_GOF_ADD UC_EP_GOF_REM NFR[2]
<b>Gestione curricula</b>	Raggruppa le classi che si occupano della gestione dei curricula: aggiunta, rimozione.	Aggiunta curriculum control, Eliminazione curriculum control	UC_EP_GCU_ADD UC_EP_GCU_REM NFR[4]
<b>Gestione gruppo di esami</b>	Raggruppa le classi che si occupano della gestione dei gruppi di esami: aggiunta di	Modifica gruppo di esami control,	UC_EP_GGE_ADD UC_EP_GGE_REM UC_EP_GGE_UPD



Laurea Triennale in Informatica - Università di Salerno  
Corso di *Ingegneria del Software* - Prof.ssa F. Ferrucci

	gruppi di esami obbligatori o opzionali; modifica dei dati dei gruppi opzionali, rimozione dei gruppi.	Eliminazione gruppo di esami control	NFR[9] NFR[12] NFR[13]
<b>Gestione esame</b>	Raggruppa le classi che si occupano della gestione dei dati degli esami: aggiunta; rimozione; modifica dei dati.	Aggiunta esame control, Modifica esame control, Eliminazione esame control	UC_EP_GES_ADD UC_EP_GES_ADDE UC_EP_GES_REM UC_EP_GES_UPD NFR[9] NFR[11] NFR[12] NFR[13] NFR[19]
<b>Gestione formulazione</b>	Si occupa della gestione della formulazione del piano di studi, permettendo di selezionare o deselezionare esami.	Formula piano di studi, Offerta formativa, Scelta corso di laurea, Scelta curriculum, Compila piano	UC_EP_FPS NFR[7] NFR[8] NFR[15] NFR[18]
<b>Offerte Formative</b>	Si interfaccia con il database, garantendo le operazioni di scrittura e lettura relative ad un'offerta formativa	Entity Offerta Formativa, Laurea.	UC_EP_GOF_ADD UC_EP_GOF_REM UC_EP_GEP NFR[2]
<b>Curricula</b>	Si interfaccia con il database, garantendo le operazioni di scrittura e lettura relative ad un curriculum	Entity Curriculum, Anno	UC_EP_GCU_ADD UC_EP_GCU_REM UC_EP_GEP NFR[4]
<b>Gruppi esami</b>	Si interfaccia con il database, garantendo le operazioni di scrittura e lettura relative ad un gruppo d'esami	Entity Gruppo di esami	UC_EP_GGE_ADD UC_EP_GGE_REM UC_EP_GGE_UPD UC_EP_GEP NFR[9] NFR[12] NFR[13]
<b>Esami</b>	Si interfaccia con il database, garantendo le operazioni di scrittura e lettura relative ad esame	Entity Esame	UC_EP_GES_ADD UC_EP_GES_ADDE UC_EP_GES_REM UC_EP_GES_UPD UC_EP_GEP NFR[9] NFR[11] NFR[12] NFR[13] NFR[19]
<b>Amministratore</b>	Si interfaccia con il database, garantendo le operazioni di	Entity Amministratore	UC_EP_LGA UC_EP_GEP NFR[6]



	lettura relative alle credenziali d'accesso per l'amministratore		
--	---	--	--

Gli oggetti correlati sono quelli specificati nella tabella *Oggetti entity, boundary e control* del capitolo 3.4.3 del documento *Requirement Analysis Document - EasyPlan*, dove viene specificato il loro tipo e la loro definizione.

Si noti come tutte le componenti sono state divise secondo la divisione in layer sopra descritta.

Le componenti che si trovano nel layer Model sono quelle a diretto contatto con il database che si occuperanno di effettuare operazioni di lettura e scrittura su di esso. Si compongono di tutti gli oggetti persistenti del sistema ovvero gli oggetti entity.

Le componenti che si trovano nel layer Controller si occupano della logica del sistema e si compongono di tutti gli oggetti di tipo Control.

La componente del layer View si occupa della gestione grafica del sistema pertanto contiene tutti gli oggetti Boundary.

### 3.3. Diagramma di deployment

L'utente richiede le funzionalità al sistema tramite il browser. Il client ospita anche la parte server poiché nei requisiti non è presente la richiesta di suddivisione in nodi client-server separati. Tuttavia, qualora in futuro si desiderasse separare client e server su nodi differenti, le modifiche da effettuare sarebbero minime.

La componente server, implementata tramite Tomcat, avrà il compito di ricevere le richieste dal browser, di elaborarle e, se necessario, di interfacciarsi con il database per effettuare delle interrogazioni. Una volta completate le operazioni, esso invierà una risposta al browser, che si occuperà di mostrarla all'utente. Poiché il sistema è ospitato da una sola macchina, il sito web sarà ospitato ed eseguito sulla porta 8080 del localhost.



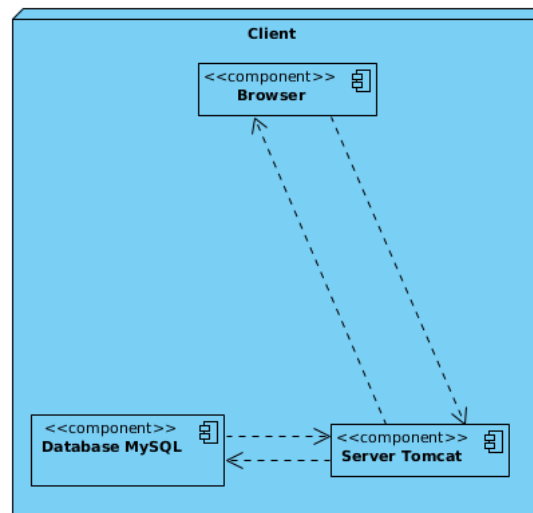


Figura 3 Diagramma di deployment

### 3.4. Mapping hardware/software

Il sistema proposto consiste di un sito web multiplatforma inizialmente installato ed eseguito su un solo computer (localhost), utilizzando un DBMS MySQL installato sullo stesso ed un server Tomcat. Client e server saranno inizialmente lo stesso computer, ma in caso di futura necessità potranno essere facilmente separati, grazie al fatto che i servizi saranno progettati separatamente. Il DBMS sarà in relazione con il database per la gestione dei dati persistenti.

Il sistema supporterà differenti tipi di risoluzioni come specificato nel RNF[14] del RAD, risultando così adattabile a qualunque tipo di dispositivo.

### 3.5. Gestione dei dati persistenti

Dato che il sistema ha bisogno di effettuare query complesse sui dati, c'è la necessità di gestire attentamente la memorizzazione dei dati persistenti e fornire meccanismi di backup; per questo, è stato deciso di utilizzare le procedure MySQL per la manipolazione del database. Grazie al supporto di Visual Paradigm, utilizzato come tool di supporto nello sviluppo dello schema E-R, il mapping logico è stato realizzato automaticamente.

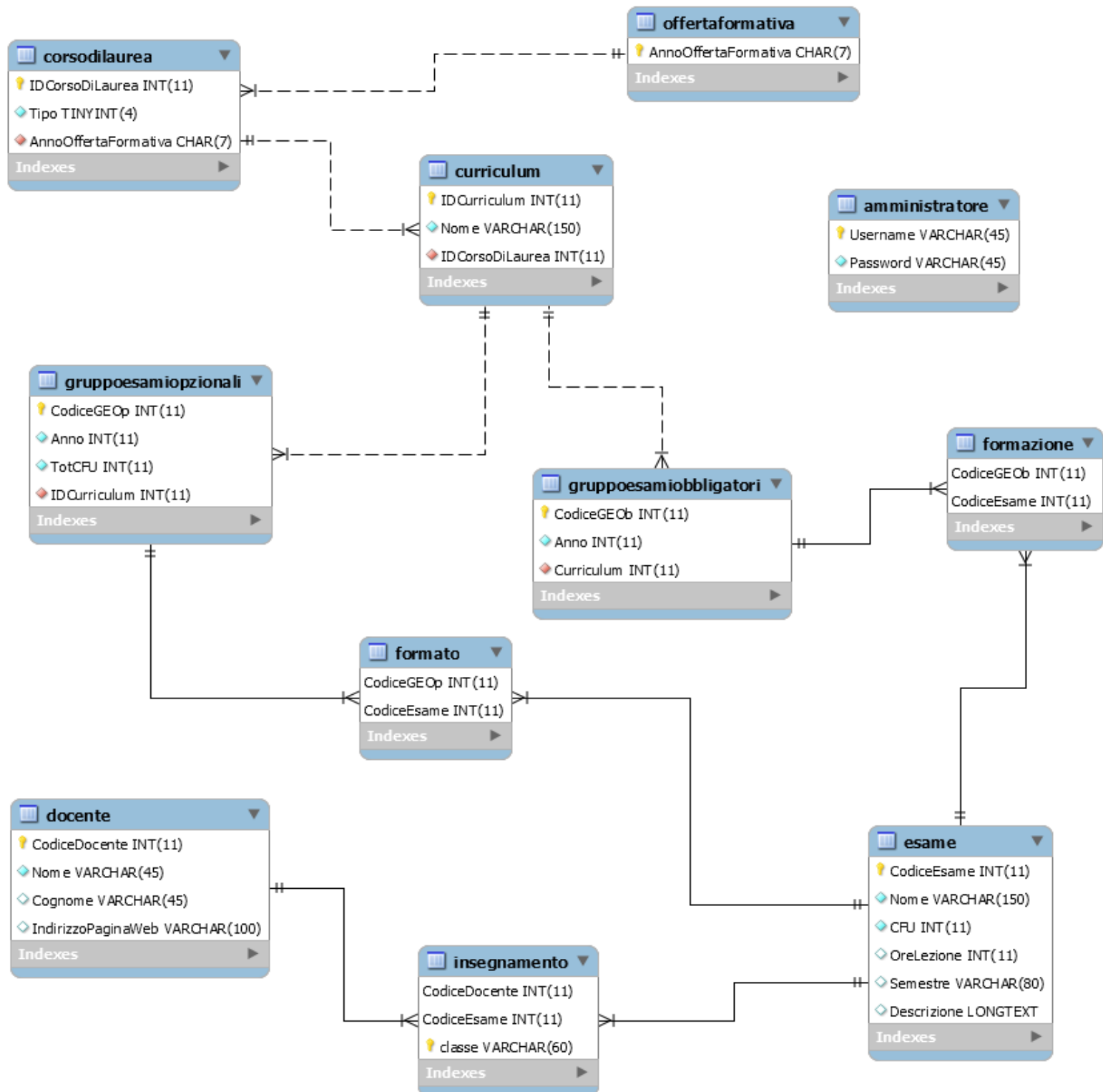


Figura 4 Schema E-R del database



Regole di vincolo
(RV1) Un'offerta formativa DEVE avere due Corsi di laurea.
(RV2) L'attributo tipo di Corsi di Laurea DEVE assumere solo due valori (Laurea triennale, Laurea Magistrale).
(RV3) Un esame presente in vari gruppi di esami della <u>stessa</u> offerta formativa DEVE essere lo stesso in ogni gruppo.
(RV4) Un esame presente in vari gruppi di esami di offerte formative <u>diverse</u> NON DEVE essere lo stesso in ogni gruppo.

Vincoli di integrità referenziale
(VIF1) La chiave esterna "Offerta formativa anno" della relazione Corso di Laurea ha un vincolo di integrità referenziale con la chiave primaria "anno" della relazione Offerta Formativa.
(VIF2) La chiave esterna "Corso di Laurea id" della relazione Curriculum ha un vincolo di integrità referenziale con la chiave primaria "id" della relazione Corso di Laurea.
(VIF3) La chiave esterna "Curriculum id" della relazione Gruppo di esami obbligatori ha un vincolo di integrità referenziale con la chiave primaria "id" della relazione Curriculum.
(VIF4) La chiave esterna "Curriculum id" della relazione Gruppo di esami opzionali ha un vincolo di integrità referenziale con la chiave primaria "id" della relazione Curriculum.
(VIF5) "Codice esame", che è parte della chiave primaria della relazione "Formazione" ha un vincolo di integrità referenziale con la chiave "Codice" della relazione esame.
(VIF6) "Gruppo Esami obbligatorio codice", che è parte della chiave primaria della relazione "Formazione" ha un vincolo di integrità referenziale con la chiave "Codice" della relazione Gruppo di esami obbligatori.
(VIF7) "Codice", che è parte della chiave primaria della relazione "Formato" ha un vincolo di integrità referenziale con la chiave "Codice" della relazione esame.
(VIF8) "Gruppo Esami Opzionale codice", che è parte della chiave primaria della relazione "Formato" ha un vincolo di integrità referenziale con la chiave "Codice" della relazione Gruppo di esami obbligatori.
(VIF9) "Esame Codice", che è parte della chiave primaria della relazione "insegnamento" ha un vincolo di integrità referenziale con la chiave primaria "Codice" della relazione "Esame".

(VIF10) “Docente Codice”, che è parte della chiave primaria della relazione “insegnamento” ha un vincolo di integrità referenziale con la chiave primaria “Codice” della relazione “Docente”.

### 3.6. Controllo degli accessi e sicurezza

L'accesso a tutte le funzionalità del lato amministrativo del sistema viene controllato dall'utilizzo di username e password. Per le funzionalità del lato utente, invece, non è previsto alcun meccanismo di controllo, dato che le funzionalità offerte in questo caso non concedono la possibilità di modificare i dati persistenti utilizzati dal sistema.

### 3.7. Controllo globale del software

L'accesso a tutte le funzionalità del lato amministrativo del sistema viene controllato dall'utilizzo di username e password. Per le funzionalità del lato utente, invece, non è previsto alcun meccanismo di controllo, dato che le funzionalità offerte in questo caso non concedono la possibilità di modificare i dati persistenti utilizzati dal sistema.

### 3.8. Condizioni limite

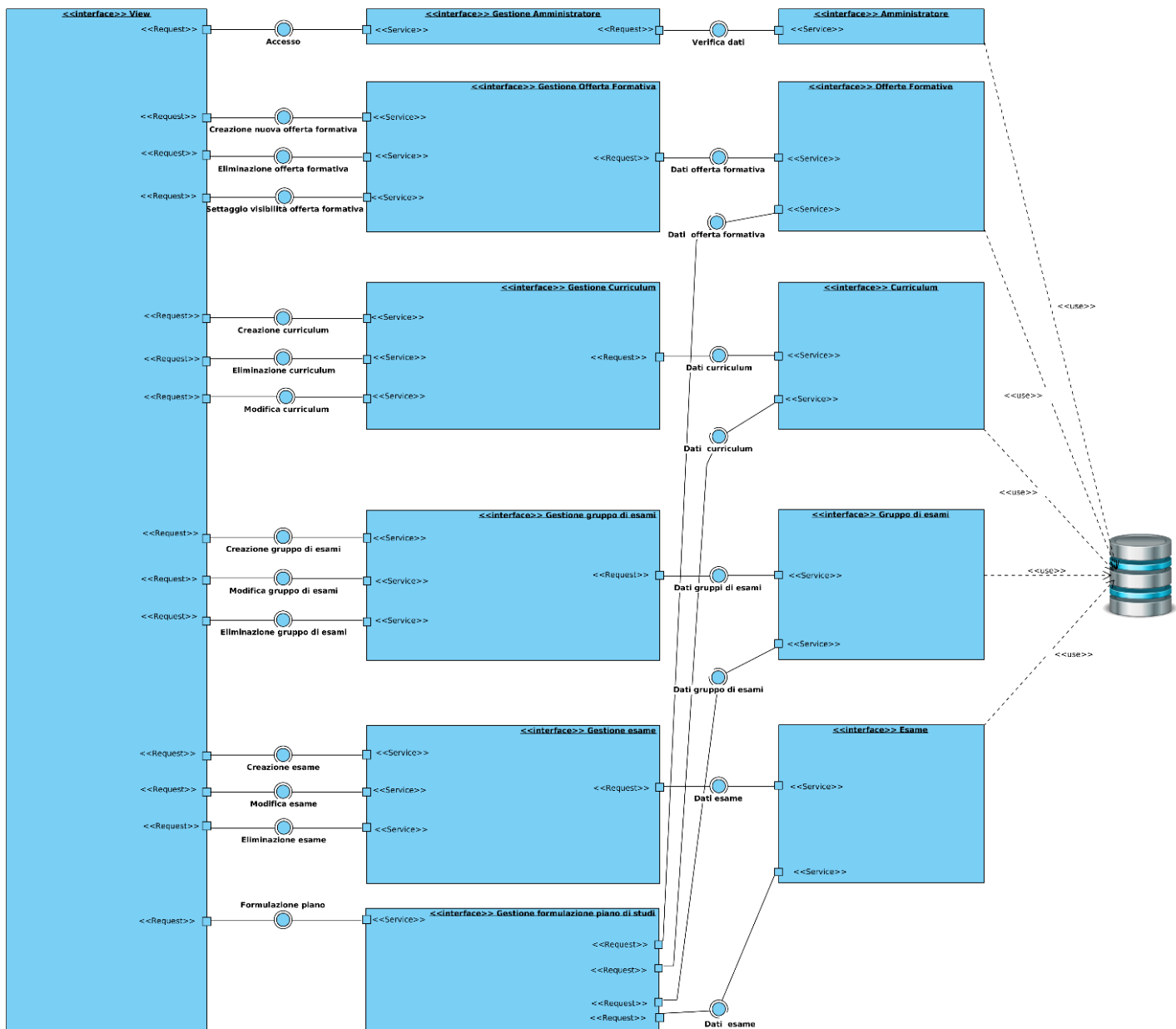
L'inizializzazione del sistema necessita dell'avvio di un web server che fornisca i servizi di un database MySQL per la gestione dei dati persistenti e l'interpretazione ed esecuzione del codice lato server. Fatto questo, l'utente potrà accedere al sito recandosi semplicemente all'apposita pagina nel browser web e avrà pieno accesso a tutte le funzionalità del sistema.

Nel momento in cui viene chiusa la corrispondente pagina del browser, il sistema viene terminato. Sia che l'utente si trovasse nel lato amministrativo, sia che si trovasse nel lato utente, il sistema non si occupa di salvare eventuali dati lasciati in sospeso durante l'utilizzo. Pertanto, al riavvio del sistema, quest'ultimo non ripresenterà i dati immessi in precedenza se questi non sono stati opportunamente salvati.

Nell'ambito di utilizzo del sistema, possono verificarsi le seguenti condizioni di fallimento:

- Nel caso di fallimento dovuto ad un guasto o al sovraccarico del database, non è prevista nessuna procedura di salvataggio né di ripristino dei dati.
- Nel caso in cui si verifichi un'interruzione inaspettata dell'alimentazione, non sono previste procedure in grado di riportare il sistema nello stato in cui era prima dello spegnimento.
- Nel caso in cui si verifichi un errore nell'hardware o del software utilizzato per accedere al sistema, non è prevista alcuna procedura per correggerli o per riportare il sistema allo stato in cui si trovava.

## 4. Servizi dei sottosistemi



Di seguito, vengono illustrati i servizi offerti da ciascun sottosistema e il modo in cui questi comunicano.

**View:** Interfacce che gestiscono l'interfaccia grafica e gli eventi generati dall'interazione dell'utente con il sistema. Effettuano solamente richieste (`<<Request>>`).

**Gestione Amministratore** offre un solo servizio all'interfaccia View:

- Accesso: per permettere l'accesso all'amministratore nella propria area dedicata.

Figura 5 Diagramma dei servizi dei sottosistemi



**Gestione Offerta Formativa** offre 3 servizi all'interfaccia View:

- Elimina offerta formativa: permette all'amministratore di eliminare un'offerta formativa.
- Settaggio visibilità offerta formativa: permette all'amministratore di settare la visibilità di un'offerta formativa.
- Creazione offerta formativa: permette all'amministratore di aggiungere un'offerta formativa.

**Gestione Curriculum** offre 3 servizi all'interfaccia View:

- Creazione Curriculum: permette all'amministratore di creare un curriculum.
- Eliminazione Curriculum: permette all'amministratore di eliminare un curriculum.
- Modifica Curriculum: permette all'amministratore di modificare le informazioni relative al curriculum.

**Gestione gruppo d'esami** offre 3 servizi all'interfaccia View:

- Creazione Gruppo d'esami: permette all'amministratore di creare un gruppo d'esami.
- Eliminazione Gruppo d'esami: permette all'amministratore di eliminare un gruppo d'esami.
- Modifica Gruppo d'esami: permette all'amministratore di modificare un gruppo d'esami.

**Gestione esame** offre 3 servizi all'interfaccia View:

- Aggiungi esame: permette all'amministratore di aggiungere un esame.
- Eliminazione esame: permette all'amministratore di eliminare un esame.
- Modifica esame: permette all'amministratore di modificare un esame.

**Gestione formulazione piano di studi** offre un servizio all'interfaccia View:

- Formulazione piano: permette all'utente di formulare il piano di studi.

**Amministratore** offre un servizio a Gestione Amministratore:

- Verifica dati: verifica che le credenziali inserite dall'amministratore siano quelle corrette

**Offerte Formative** offre un servizio a Gestione Offerta Formativa

- Dati Offerta Formativa: Fornisce i dati relativi alle offerte formative

Ed un servizio a Gestione Formulazione Piano Di Studi:

- Dati Offerta Formativa: Fornisce i dati relativi alle offerte formative

**Curriculum** offre un servizio a Gestione Curriculum:

- Dati Curriculum: Fornisce i dati relativi ai curriculum

Ed un servizio a Gestione Formulazione Piano Di Studi:

- Dati Curriculum: Fornisce i dati relativi ai curriculum

**Gruppi di esame** offre un servizio a Gestione Gruppo Esami:



- Dati Gruppi esami: Fornisce i dati relativi ai gruppi d'esami

Ed un servizio a Gestione Formulazione Piano Di Studi:

- Dati Gruppi esami: Fornisce i dati relativi ai gruppi d'esami

**Esame** offre un servizio a Gestione Esame:

- Dati esame: Fornisce i dati relativi agli esami

Ed un servizio a Gestione Formulazione Piano Di Studi:

- Dati esame: Fornisce i dati relativi agli esami

**Database** offre un servizio ad **Amministratore, Offerta Formativa, Curriculum, Gruppo esami, Esame**:

- Use: fornisce tutti i dati offrendo operazioni di lettura e scrittura all'interno del database.