



Laurea Magistrale in informatica-Università di Salerno
Corso di Gestione dei Progetti Software- Prof.ssa F.Ferrucci



System Design Document

Easy Traineeship

Riferimento	
Versione	1.0
Data	11/01/2020
Destinatario	Top Management
Presentato da	Alessandro Serritiello, Elio Testa, Felice Montella, Giuseppe Barisano, Luigi Barbato, Pellegrino Aurilia, Salvatore Amendola, Simona Grilletto
Approvato da	Vincenzo Belmonte, Vincenzo De Rosa



Revision History

Data	Versione	Descrizione	Autori
25/11/2019	0.1	Inizio stesura SDD	Amendola Salvatore
25/11/2019	0.1	Aggiunta dei capitoli 1,2,3.2	[tutti]
27/11/2019	0.2	Aggiunta del paragrafo 3.3	Amendola Salvatore/ Montella Felice
28/11/2019	0.3	Aggiunta del paragrafo 3.4	Barisano Giuseppe/ Testa Elio
28/11/2019	0.4	Aggiunta dei paragrafi 3.5, 3.6, 3.7	Grilletto Simona/ Serritiello Alessandro
29/11/2019	0.5	Aggiunta del capitolo 4	Aurilia Pellegrino/ Barbato Luigi
30/11/2019	0.6	Fine stesura SDD e aggiunta del capitolo 5	Amendola Salvatore
30/11/2019	0.7	Revisione SDD	Testa Elio
10/01/2020	0.8	Revisione del paragrafo 1.2	Amendola Salvatore/ Serritiello Alessandro
11/01/2020	1.0	Revisione dei paragrafi: 3.2, 3.4.4, 3.4.5	Pellegrino Aurilia/ Felice Montella/ Giuseppe Barisano



Sommario

Revision History	2
1. Introduzione	5
1.1. Obiettivo del sistema	5
1.2. Design Goals	6
1.2.1. Trade-off:	7
1.3. Definizioni, acronimi e abbreviazioni	8
1.4. Riferimenti	9
1.5. Organizzazione del documento	9
2. Architettura del Sistema corrente	9
3. Architettura del Sistema proposto	10
3.1. Panoramica	10
3.2. Decomposizione in sottosistemi	10
3.3. Mapping hardware/software	13
3.4. Gestione dati persistenti	13
3.4.1. Premessa	13
3.4.2. Rappresentazione UML del DataBase proposto	14
3.4.3. Rappresentazione UML del DataBase proposto Ristrutturato	15
3.4.4. Rappresentazione UML del DataBase Integrato	16
3.4.5. Schema Logico	17
3.4.6. Tavola dei Volumi	17
3.4.7. Tavola delle Operazioni	18
3.4.8. Tavola degli accessi	18
3.5. Controllo degli accessi e sicurezza	19



3.6.	Controllo flusso globale del sistema	21
3.7.	Condizione limite	21
3.7.1.	Avvio del Sistema.....	21
3.7.2.	Spegnimento del Sistema.	23
3.7.3.	Fallimento del Sistema.....	24
4.	Servizi dei sottosistemi	25
4.1.	Gestione Account.....	25
4.2.	Gestione Richieste.....	25
4.3.	Gestione Tirocinio	26
5.	Glossario.....	27



1. Introduzione

1.1. Obiettivo del sistema

Il Sistema Easy Traineeship (ET) intende realizzare una piattaforma, la quale permetta di facilitare le pratiche burocratiche riguardanti l'iter del Tirocinio, all'interno del Dipartimento di Informatica dell'Università degli Studi di Salerno.

Questo Sistema andrà ad automatizzare, sia il processo svolto dallo Studente, il quale richiede la pratica di Tirocinio, sia il processo effettuato dagli organi Amministrativi, come Segreteria ed Admin, ma anche quello degli Enti Convenzionati all'Università.

Inoltre, permetterà la gestione del documento "*Progetto Formativo*", compilato dallo Studente telematicamente, alleggerendo il processo burocratico attualmente esistente, permetterà la visualizzazione ed il controllo facile e veloce del Tirocinio di uno Studente e la visualizzazione dei propri dati.

Per raggiungere questo obiettivo è stato individuato l'attuale processo burocratico che porta alla convalida del Tirocinio attraverso il sito dell'Università.

Il miglioramento delle procedure burocratiche porterebbe eventuali benefici, anche nel caso in cui si voglia estendere il Sistema ad altri servizi simili a quello proposto.

Il Sistema progettato è una vera e propria web app a cui avranno accesso l'Admin, la Segreteria, l'Ente gli Studenti, facilitando il processo burocratico in modo da concludere la procedura nel minor tempo possibile.



1.2. Design Goals

ID	Categoria	Nome	Descrizione	Priorità
DG_1	Criteri di Utente finale:	Usabilità	Il Sistema sarà più facile ed intuitivo.	Media
DG_2	Criteri di Affidabilità:	Attendibilità	Il Sistema garantirà l'attendibilità, intesa come corretta gestione delle funzionalità.	Alta
DG_3	Criteri di Affidabilità:	Disponibilità	Il Sistema dovrà essere permanentemente fruibile dagli utenti.	Alta
DG_4	Criteri di Affidabilità:	Sicurezza	Ogni Utente è dotato di e-mail e password con cui poter effettuare l'accesso al Sistema e utilizzare solo le funzionalità associate alla propria categoria.	Bassa
DG_5	Criteri di Affidabilità:	Tolleranza ai guasti	Si eviterà la perdita dei dati e il fallimento di operazioni.	Bassa
DG_6	Criteri di Performance:	Tempi di risposta	Il prodotto Software deve consentire una navigazione rapida ai vari Utenti, quindi, tempi di risposta minimi nello svolgimento delle funzionalità da esso offerte.	Bassa
DG_7	Criteri di Performance:	Memoria	La memoria necessaria al funzionamento del sistema dipende dalla memoria	Bassa



			utilizzata per il mantenimento del Database.	
DG_8	Criteri di Costo:	Costo di sviluppo	È stimato un costo complessivo di 400 ore per la progettazione e lo sviluppo del sistema.	Alto
DG_9	Criteri di Manutenibilità:	Adattabilità	Il Sistema funziona solo per il dipartimento di informatica e per l'Università degli studi di Salerno, ma è adattabile a più dipartimenti modificando i relativi controlli.	Alta
DG_10	Criteri di Manutenibilità:	Modificabilità	Il Sistema dovrà essere facilmente modificabile.	Alta
DG_11	Criteri di Manutenibilità:	Estendibilità	Il Sistema potrà essere esteso con altre funzionalità.	Alta
DG_12	Criteri di Manutenibilità:	Tracciabilità	I requisiti del Sistema sono facilmente tracciabili, grazie all'utilizzo di una matrice di tracciabilità.	Alta

1.2.1. *Trade-off:*

- ***Attendibilità vs Tempi di risposta***

Il Sistema sarà implementato in modo tale da preferire l'attendibilità al tempo di risposta, in modo tale da garantire un controllo più accurato dei dati in input a discapito del tempo di risposta del sistema.

- ***Disponibilità vs Tolleranza ai guasti***

Il Sistema deve sempre essere disponibile all'Utente in caso di errore in una funzionalità, anche al costo di rendere non disponibile quest'ultima per un lasso di tempo.



- ***Memoria vs Estendibilità***

Il sistema deve permettere l'estendibilità a discapito della memoria utilizzata. Tale preferenza permette al cliente di richiedere agli sviluppatori nuove funzionalità, dando meno importanza alla memoria utilizzata.

- ***Criteri di Manutenibilità vs Criteri di Performance***

Il Sistema sarà implementato preferendo la manutenibilità ai tempi di risposta in modo da facilitare gli sviluppatori nel processo di aggiornamento del software a discapito delle performance del sistema.

1.3. Definizioni, acronimi e abbreviazioni

- RAD: Requirements Analysis Document.
- Tirocinio: Tirocinio curriculare esterno.
- UC: Use Case.
- RF: Requisito funzionale.
- NRF: Requisito non funzionale.
- UNISA: Università degli Studi di Salerno.
- Dipartimento: Dipartimento di informatica degli Studi di Salerno.
- EV: Sistema *"English Validation"*
- DB: DataBase;
- DBMS: DataBase Management System
- SDD: System Design Document
- MySQL: È un database Open Source basato sul linguaggio SQL, composto da un client a riga di comando e un Server.
- SQL: Structured Query Language; linguaggio standardizzato per database basati sul modello relazionale (RDBMS) progettato per: creare e modificare schemi di database.
- Utente: Un qualsiasi utilizzatore della piattaforma.
- Ente: Ente/azienda preposto al tirocinio esterno.
- Admin: Figura che rappresenta il Consiglio Didattico.
- Segreteria: Figura dell'amministrazione didattica.



1.4. Riferimenti

- Bernd Bruegge & Allen H. Dutoit, Object-Oriented Software Engineering: Using UML, Patterns and Java, (2nd edition), Prentice-Hall, 2003
- Ian Sommerville, Software Engineering, Addison Wesley
- Statement of work
- English Validation, Sistema di validazione di UNISA
- C2_RAD_Vers. 1.3

1.5. Organizzazione del documento

Introduzione: Contiene l'introduzione con l'obiettivo del sistema, i design goals e un elenco di definizioni, acronimi e abbreviazioni utili alla comprensione dell'intera documentazione.

Architettura Sistema Corrente: Descrive, nel caso esista, le funzionalità offerte dal sistema corrente.

Architettura Sistema Proposto: Viene presentata l'architettura del sistema proposto, in cui sarà gestita la decomposizione in sottosistemi, il mapping hardware/software, i dati persistenti, il controllo degli accessi e sicurezza, il controllo del flusso globale del sistema, le condizioni limite.

Servizi dei Sottosistemi: Vengono presentati i servizi dei sottosistemi.

Glossario: Raccolta di vocaboli meno comuni utilizzati nella stesura del documento.

2. Architettura del Sistema corrente

Nello stato attuale non esiste alcuna architettura software che soddisfi le problematiche del Tirocinio Esterno. Il nostro Sistema andrà ad estendere il Sistema "*English Validation*", implementando una nuova funzionalità, la quale gestisca il Tirocinio Esterno.

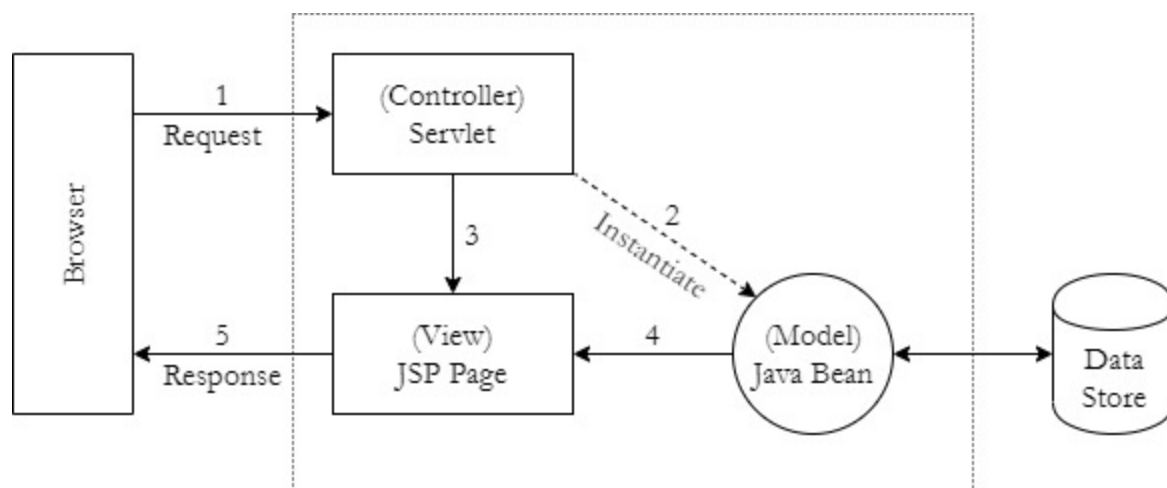
Il prodotto software corrente è finalizzato a gestire le richieste di convalida dei CFU di lingua inglese, da parte degli studenti del Dipartimento di Informatica dell'Università degli Studi di Salerno ed utilizza un'architettura sviluppata mediante il modello MVC, correlata ad un DataStore con relativo DBMS.

L'architettura utilizzata sarà riferita al prodotto software già esistente, quindi, riutilizzando il modello MVC con relativo DataStore e DBMS.

3. Architettura del Sistema proposto

3.1. Panoramica

Il pattern architetturale scelto è MVC (Model, View, Control).



Model: contiene i metodi di accesso ai dati.

View: si occupa di visualizzare i dati all'Utente e gestisce l'interazione fra quest'ultimo e l'infrastruttura sottostante.

Controller: riceve i comandi dell'Utente attraverso il View e reagisce eseguendo delle operazioni che possono interessare il Model e che portano generalmente ad un cambiamento di stato del View.

3.2. Decomposizione in sottosistemi

Il Sistema è stato diviso in quattro sottosistemi principali:

Il Sottosistema **Gestione Account** gestisce l'autenticazione dell'Utente, in particolare le funzioni di Login, Logout, Recupero Password e Registrazione. Inoltre, permette agli utenti la visualizzazione della Lista Enti, per la consultazione dei dettagli dell'Ente prescelto; in particolare: nome, eventuali sedi, e-mail (affinché l'Utente possa richiedere maggiori informazioni), descrizione per poter reperire maggiori informazioni riguardanti il Progetto Formativo offerto dall'Ente preso in considerazione.

Questo Sottosistema, inoltre, permette alla Segreteria di gestire la Registrazione, la Modifica e la Rimozione degli Enti Convenzionati (o non) all'Università.

Il Sottosistema **Gestione Richieste** gestisce l'inoltro delle Richieste di Tirocinio da parte dello Studente, ai vari utenti coinvolti, con l'eventuale Annullamento.



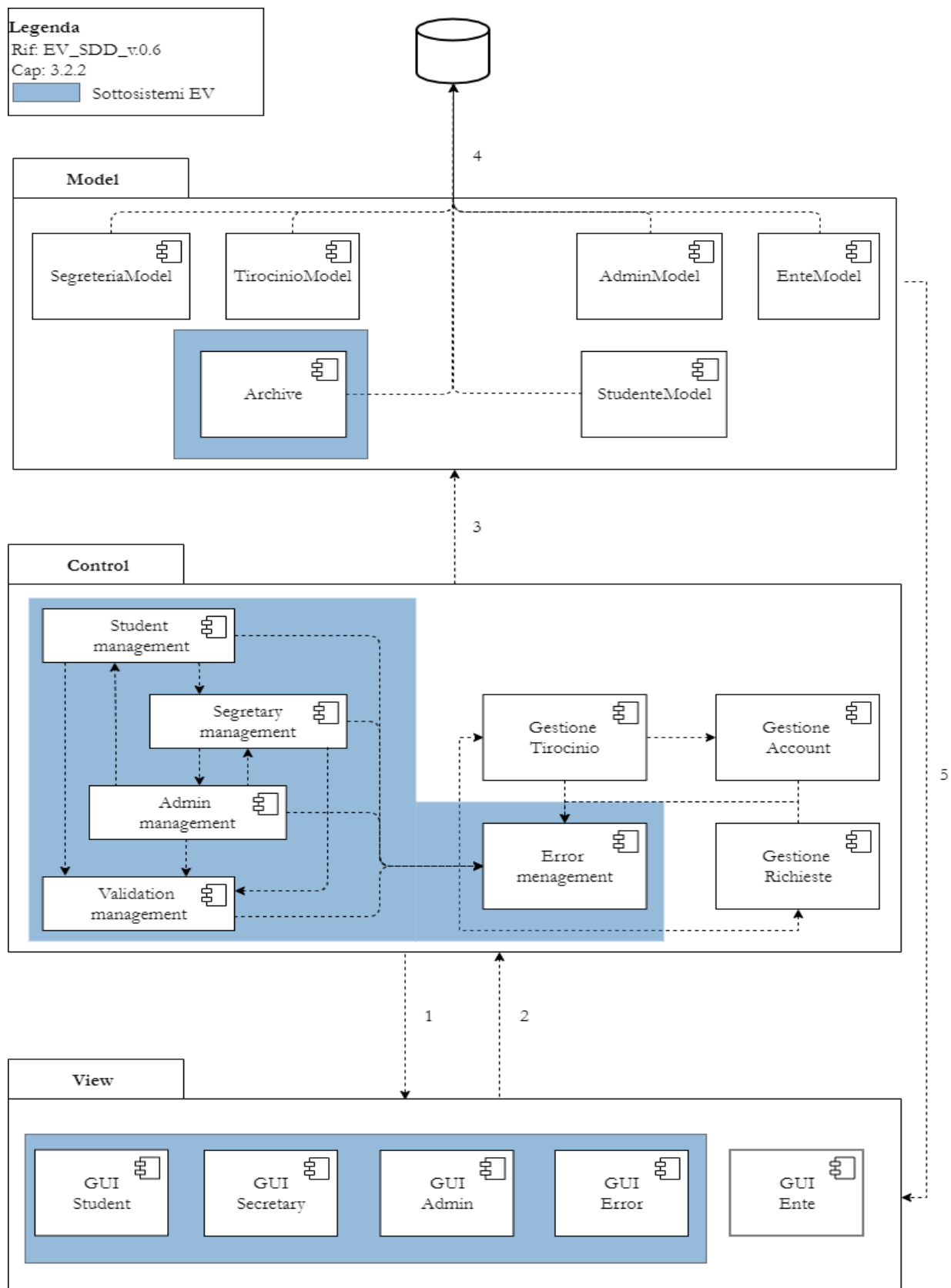
Inoltre, consente alla Segreteria e all'Ente il Rifiuto o l'Accettazione delle Richieste inviate dallo Studente, mediante la visualizzazione dell'elenco di tutte le Richieste di Tirocinio.

Il Sottosistema ***Gestione Tirocinio*** gestisce la stesura del “*Progetto Formativo*”, generato contemporaneamente alla composizione delle Richieste di Tirocinio. Gestisce anche la Visione, l'Upload ed il Download del documento “*Progetto Formativo*” in modo tale da permettere agli utenti presi in considerazione di poterlo visionare e firmare.

Lo Studente, all'interno di quest'ultimo Sottosistema può controllare lo Stato del proprio Tirocinio e lo storico di tutte le sue richieste.

Siccome ogni Sottosistema è sottoposto a possibilità di errore, è stato introdotto il Sottosistema ***Error management***, il quale permette di risolverli oppure di indirizzarli verso una “*GUI Errore*”, in modo da rendere l'Utente informato di tale errore.

I sottosistemi, da noi individuati, saranno estesi a quelli di EV presenti nel documento: EV_SDD_0.6
Cap: 3.2.2.





3.3. Mapping hardware/software

Poiché il Sistema proposto sarà sviluppato come estensione di “English Validation” e poiché si è deciso di non effettuare un re-engineering di tale sistema per non eccedere sia in costi che effort di tutto il team, al fine di preservare le funzionalità già esistenti, si è preferito conservare il mapping hardware/software del suddetto Sistema (documento: “EV_SDD”, paragrafi 3.3.2 e 3.3.3).

Il Sistema, che si desidera sviluppare, utilizza un’architettura Client/Server, in cui un Server fornisce servizi a più client. Su una macchina client è eseguito un web browser che consente all’Utente di interagire con il livello View (lato Server) per inoltrare richieste e visualizzare le risposte ricevute. Il System gestisce la logica applicativa, mentre la macchina "DataBase Server" gestisce i dati persistenti. La comunicazione tra Client e Server avviene tramite protocollo HTTP.

HTTP è un protocollo di trasferimento di ipertesti che consente a due macchine, client e Server, di interagire attraverso un meccanismo di richiesta/risposta. Il Client inoltra una richiesta al Server, che verrà soddisfatta con la risposta di quest’ultimo.

Le specifiche hardware e software necessarie per il Client sono rispettivamente una macchina dotata di connessione a Internet e un sistema operativo con un web browser installato.

Le specifiche hardware necessarie per il Server consistono in una macchina connessa a Internet, la quale sia capace di immagazzinare grandi quantità di dati.

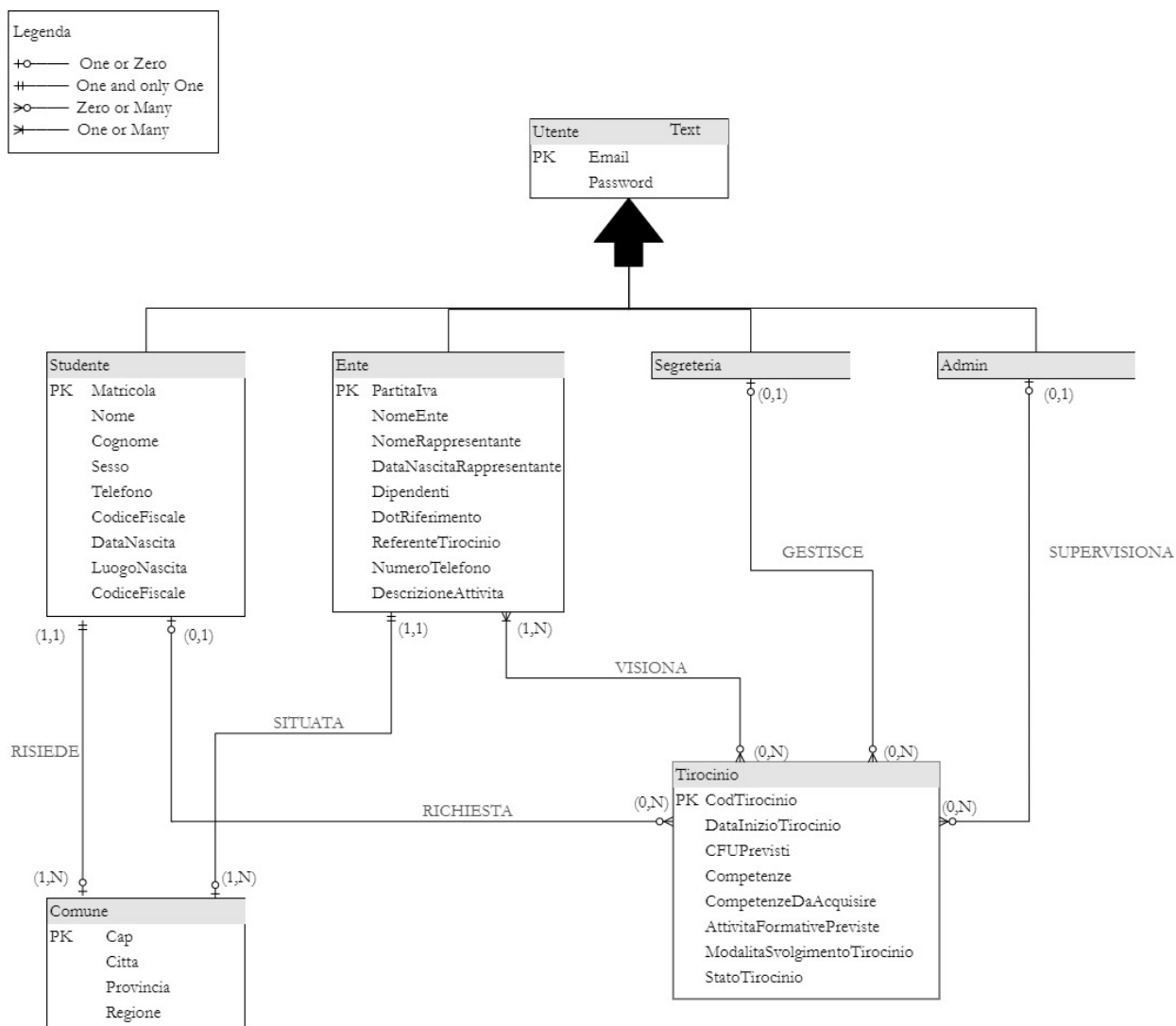
Le specifiche software necessarie comprendono un Database Management System (*MySQL*) per la gestione dei dati persistenti, un System per la gestione della logica applicativa e della comunicazione con più client.

3.4. Gestione dati persistenti

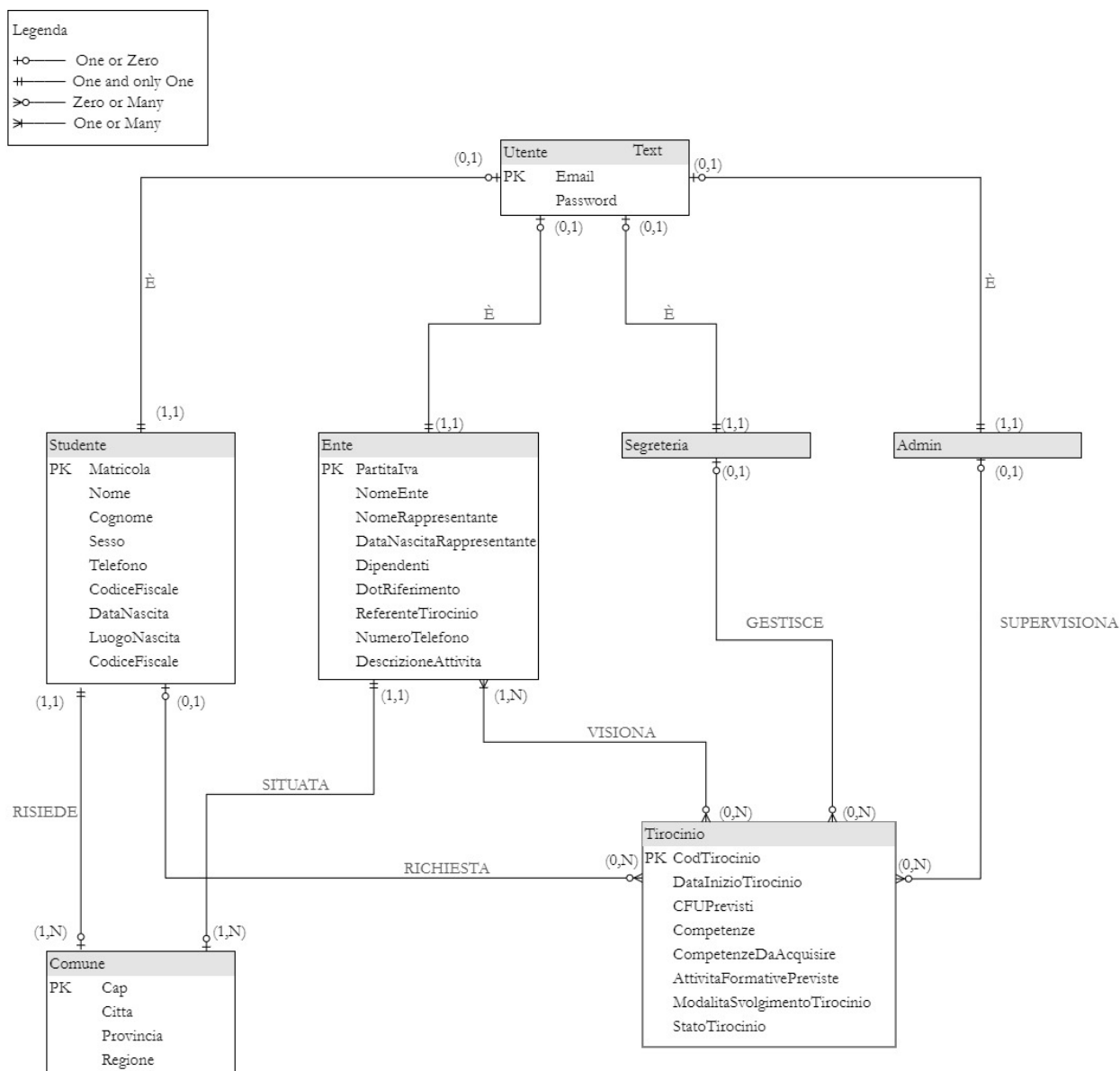
3.4.1. Premessa

Poiché il Sistema proposto sarà sviluppato come estensione di “English Validation”, per non eccedere sia in costi che effort di tutto il team e mantenere i criteri di Manutenibilità e Performance, si è preferito non re-ingegnerizzare il modello ER di “English Validation” (documento: “EV_SDD”, paragrafo 3.4) ma, piuttosto, di integrare le nuove entità con quelle esistenti.

3.4.2. Rappresentazione UML del DataBase proposto

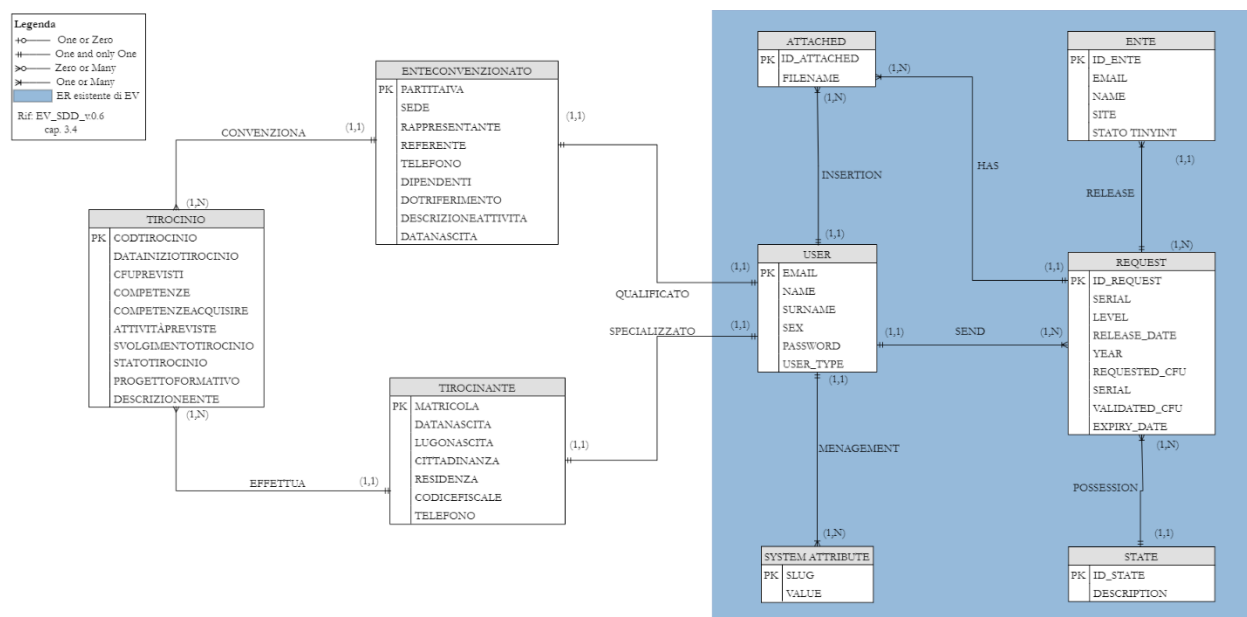


3.4.3. Rappresentazione UML del DataBase proposto Ristrutturato

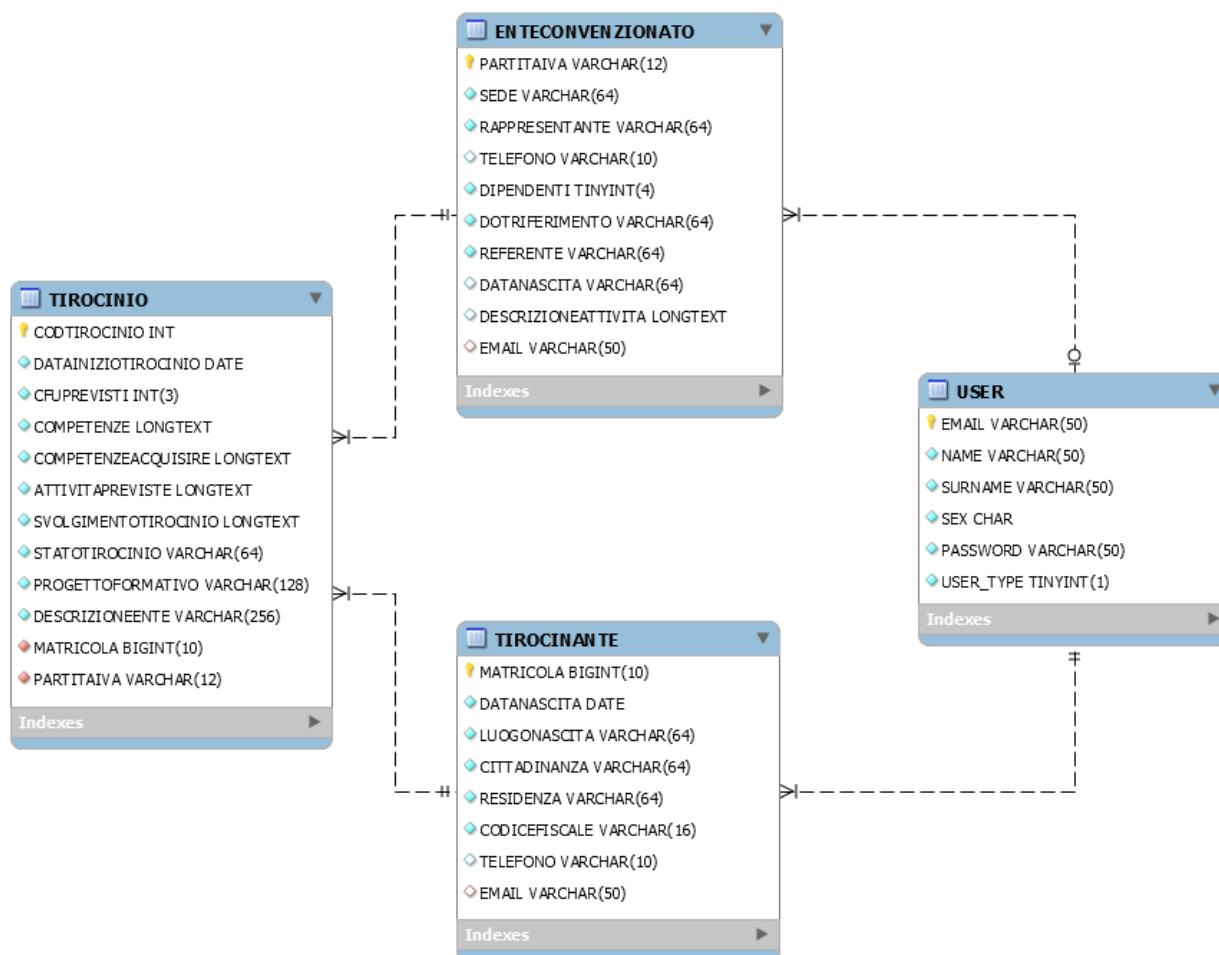


3.4.4. Rappresentazione UML del DataBase Integrato

Integrazione del modello ER di “English Validation” (documento: “EV_SDD”, paragrafo 3.4) come da premessa.



3.4.5. Schema Logico



3.4.6. Tavola dei Volumi

Concetto	Costrutto	Volume
User	Entità	1500
Tirocinante	Entità	500
EnteConvenzionato	Entità	100
Tirocinio	Entità	800
Effettua	Relazione	600
Convenzione	Relazione	400
Qualified	Relazione	100
Specialized	Relazione	500



3.4.7. *Tavola delle Operazioni*

Operazione	Tipo	Frequenza
Visualizza Lista Enti	Interattive	150 uso/mese
Visualizza Richiesta Tirocinio	Interattive	2 uso/mese
Visualizza Lista Tirocinanti	Interattive	15 uso/mese
Controlla Stato Del Tirocinio	Interattive	150 suo/mese
Controllare Stato Del Proprio Tirocinio	Interattive	20 uso/mese
Visualizzare Documenti Da Firmare	Interattive	60 uso/mese

3.4.8. *Tavola degli accessi*

Visualizza Lista Enti

Concetto	Costrutto	Accessi	Tipo
EnteConvenzionato	Entity	1	Lettura

Totale Accessi= $1*150*100=15.000$ accesso/mese.

Visualizza Richiesta Tirocinio

Concetto	Costrutto	Accessi	Tipo
Tirocinante	Entity	1	Lettura
Effettua	Relation	1	Lettura
Tirocinio	Entity	1	Lettura

Totale Accessi= $[(1*500) + (1*600) + (1*800)] *2=3.800$ accesso/mese.

Visualizza Lista Tirocinanti

Concetto	Costrutto	Accessi	Tipo
User	Entity	1	Lettura
Specialized	Relation	1	Lettura
Tirocinante	Entity	1	Lettura

Totale Accessi = $[(1*1500) + (1*500) + (1*500)] *15=37.500$ accesso/mese.



Controllare Stato Del Proprio Tirocinio

Concetto	Costrutto	Accessi	Tipo
User	Entity	1	Lettura
Specialized	Relation	1	Lettura
Tirocinante	Entity	1	Lettura
Effettua	Relation	1	Lettura
Tirocinio	Entity	1	Lettura

Totale Accessi = $[(1*1500) + (1*500) + (1*500) + (1*600) + (1*800)] * 20 = 78.000$ accesso/mese.

Visualizzare Documenti Da Firmare

Concetto	Costrutto	Accessi	Tipo
User	Entity	1	Lettura
Specialized	Relation	1	Lettura
EnteConvenzionato	Entity	1	Lettura
Convenziona	Relation	1	Lettura
Tirocinio	Entity	1	Lettura

Totale Accessi = $[(1*1500) + (1*500) + (1*100) + (1*400) + (1*800)] * 60 = 198.000$ accesso/mese.

Totale Degli Accessi = $15.000 + 3.800 + 37.500 + 120.000 + 78.000 + 198.000 = 452.300$ accesso/mese.

3.5. Controllo degli accessi e sicurezza

All'interno del Sistema, la funzione di visualizzazione lista degli Enti è accessibile a tutti gli utenti.

La funzionalità di registrazione, prevista per lo Studente, è ripresa dal Sistema corrente "English Validation".

Attraverso l'autenticazione si ha l'avvio di una sessione, la quale può essere interrotta in qualsiasi momento, mediante la funzionalità di Logout.

Nel momento in cui avviene l'autenticazione, con il corretto utilizzo di username e password, è possibile accedere alla propria area Utente, generata diversamente per ogni attore e contenente le specifiche funzioni da poter effettuare.



Attore	Gestione Account	Gestione Richieste	Gestione Tirocinio
Utente ospite	<ul style="list-style-type: none"> Login Visualizza Enti 	NA	NA
Studente	<ul style="list-style-type: none"> Logout Recupero Password 	<ul style="list-style-type: none"> Invio richiesta inizio Tirocinio Invio richiesta Ente Annullare richiesta Ente 	<ul style="list-style-type: none"> Download Progetto Formativo Upload Progetto Formativo Controllo stato proprio Tirocinio
Segreteria	<ul style="list-style-type: none"> Logout Recupero Password Registrazione Ente Modifica Ente Eliminazione Ente 	<ul style="list-style-type: none"> Visualizzare richieste inizio Tirocinio Accettare richiesta inizio Tirocinio Annullare Tirocinio Rifiuto richiesta inizio Tirocinio 	<ul style="list-style-type: none"> Controllare stato Tirocinio Visualizzare Tirocinanti
Admin	<ul style="list-style-type: none"> Logout Recupero Password 	<ul style="list-style-type: none"> Annullare Tirocinio 	<ul style="list-style-type: none"> Download Progetto Formativo Upload Progetto Formativo Visualizzazione documenti da firmare
Ente	<ul style="list-style-type: none"> Logout Recupero Password 	<ul style="list-style-type: none"> Visualizzazione richieste Tirocinio Accettare richiesta Studente Rifiuto richiesta dall'Ente 	<ul style="list-style-type: none"> Download Progetto Formativo Upload Progetto Formativo Visualizzazione documenti da firmare



3.6. Controllo flusso globale del sistema

Il controllo del flusso viene effettuato mediante delle richieste, le quali vengono generate da un client, e dalle classi preposte a gestire quel determinato evento associato alla richiesta.

Una volta ottenuto il risultato dell'operazione, la classe gestore si preoccupa di inoltrarlo al client che aveva generato la richiesta.

Il sistema software è gestito con l'uso di Servlet e JSP. Il Server centrale attende le richieste di un client (web browser) e una volta ricevuta una richiesta, la processa e la smista alla Servlet incaricata.

3.7. Condizione limite

Il Server dovrà essere sempre attivo, permettendo l'utilizzo continuo dei servizi.

In caso di malfunzionamento, il Sistema mostrerà un avviso di manutenzione, precludendo all'Utente il normale utilizzo delle funzionalità del Sistema.

Nel caso di un crash del Sistema dovuto ad un errore, si effettuerà un ripristino al backup più recente.

3.7.1. Avvio del Sistema

Dopo l'avvio, il Sistema presenta un'interfaccia ai client, con funzionalità disponibili differenti per ogni attore, dopo aver effettuato correttamente l'autenticazione.

NOME SCENARIO	SC_3.7.1: AVVIO DEL SERVER	
ATTORI	GIACOMO: OPERATORE	
FLUSSO DEGLI EVENTI	OPERATORE	SISTEMA
	Giacomo decide di avviare il Sistema usando il comando apposito.	
		Il Sistema diventa operativo e attiva i servizi in remoto disponibili per gli utenti.

Identificativo UC_3.7.1	Avvio del server	Data	28/11/2019
		Vers.	0.00.001
		Autore	Alessandro Serritiello



Descrizione		Lo UC fornisce la funzionalità di avvio del Server.
Attore Principale		Operatore Vuole avviare il Server per rendere disponibile il Sistema.
Attori Secondari		NA
Entry Condition		Il Server è pronto all'avvio.
Exit condition On success		Il Server è avviato e il Sistema rende disponibili le funzionalità agli utenti.
Exit condition On failure		Il Server non è avviato.
Rilevanza/User Priority		Alta
Extension point		NA
Generalization of		NA
FLUSSO DI EVENTI PRINCIPALE/MAIN SCENARIO		
1	OPERATORE:	Utilizza l'apposito comando sulla macchina sul quale risiede il Server per avviarlo.
2	Sistema:	Rende disponibili le funzionalità agli utenti e notifica dell'avvio avvenuto correttamente.
I Scenario/Flusso di eventi di ERRORE: Connessione assente		
2.1	Sistema:	Mostra un messaggio di errore all'Utente. Il messaggio segnala che non è stato possibile avviare il Server per via della mancanza di connessione.
Note		
Special Requirements		
1		NA



3.7.2. *Spegnimento del Sistema.*

È possibile terminare il Sistema in modo corretto unicamente se non sono presenti connessioni ad esso.

Prima della disattivazione totale del database-Server e dell'application-Server, verranno disconnessi tutti i client connessi al Sistema; ad ogni Client arriverà una notifica per avvertirli della terminazione del Sistema.

Attraverso la funzionalità di Logout è possibile disattivare la connessione da qualsiasi postazione.

NOME SCENARIO	SC_3.72: SPEGNIMENTO DEL SERVER	
ATTORI	GIACOMO: OPERATORE	
FLUSSO DEGLI EVENTI	OPERATORE	SISTEMA
	Giacomo decide di spegnere il Sistema usando il comando apposito.	
		Il Sistema avverte gli utenti dell'avvenuto spegnimento, successivamente esegue l'arresto.

Identificativo UC_3.7.2	Spegnimento del Server	Data	28/11/2019
		Vers.	0.00.001
		Autore	Simona Grilletto
Descrizione	Lo UC fornisce la funzionalità di spegnimento del Server.		
Attore Principale	Operatore Vuole stoppare il Server.		
Attori Secondari	NA		
Entry Condition	Il Server è in attività.		
Exit condition On success	Il Server viene stoppato.		



Exit condition		Il Server non viene stoppato.
On failure		
Rilevanza/User Priority		Alta
Extension point		NA
Generalization of		NA
FLUSSO DI EVENTI PRINCIPALE/MAIN SCENARIO		
1	OPERATORE:	Utilizza l'apposito comando, all'interno della macchina, per poter stoppare il Server.
2	Sistema:	Il Sistema arresta la sua esecuzione.
I Scenario/Flusso di eventi di ERRORE: Utenti connessi		
2.1	Sistema:	Mostra un messaggio di errore alla Segreteria. Il messaggio segnala che non è stato possibile stoppare il Server a causa della mancata disconnessione di qualche utente.
Note		
Special Requirements		
1		NA

3.7.3. *Fallimento del Sistema*

Nel caso in cui il Server vada in un caso di errore, è previsto un ripristino dei dati, perché sarà effettuato un backup. Si cercherà di non perdere dati persistenti, infatti il Sistema è strutturato utilizzando un DBMS, per cui il rischio è minimo.

Per aumentare la sicurezza, saranno effettuati periodicamente, oltre ai backup, controlli hardware di manutenzione.



4. Servizi dei sottosistemi

4.1. Gestione Account

Sottosistema	Gestione Account
Descrizione	Questo sottosistema permette di effettuare tutte le operazioni relative agli account degli utenti
Servizi Offerti	
Servizio	Descrizione
Login	Questo servizio consente di accedere al Sistema
Logout	Questo servizio consente di uscire dal Sistema
recoverPassword	Questo servizio consente di recuperare la propria password per l'accesso al Sistema
newEnte	Questo servizio permette di registrare l'Ente al Sistema
modifyEnte	Questo servizio permette di effettuare modifiche all'Ente già registrato
removeEnte	Questo servizio permette di eliminare l'Ente dal Sistema
getEntiList	Questo servizio offre la visualizzazione dei diversi Enti registrati al Sistema

4.2. Gestione Richieste

Sottosistema	Gestione Richieste
Descrizione	Questo sottosistema permette di effettuare tutte le operazioni relative alla gestione delle Richieste
Servizi Offerti	
Servizio	Descrizione
askTirocinio	Questo servizio consente allo Studente di inviare la richiesta di inizio Tirocinio
sendEnteRequest	Questo servizio consente allo Studente di inviare la richiesta di Tirocinio all'Ente



cancelEnteRequest	Questo servizio consente allo Studente di annullare la richiesta all'Ente precedentemente scelto
getTirociniList	Questo servizio permette alla Segreteria di visualizzare le richieste di inizio Tirocinio
acceptTirocinio	Questo servizio permette alla Segreteria di accettare le richieste di inizio Tirocinio
cancelTirocinio	Questo servizio permette alla Segreteria ed all'Admin di annullare le richieste di Tirocinio
rejectTirocinio	Questo servizio permette alla Segreteria di rifiutare le richieste di inizio Tirocinio

4.3. Gestione Tirocinio

Sottosistema	Gestione Tirocinio
Descrizione	Questo sottosistema permette di effettuare tutte le operazioni relative alla gestione delle Richieste
Servizi Offerti	
Servizio	Descrizione
downloadPF	Questo servizio consente agli utenti di scaricare il Progetto Formativo
uploadPF	Questo servizio consente agli utenti di caricare il Progetto Formativo
checkStato	Questo servizio consente allo Studente di controllare lo stato del proprio Tirocinio
checkTirociniStates	Questo servizio permette alla Segreteria di controllare lo stato dei diversi Tirocini
getTirocinanti	Questo servizio permette alla Segreteria di visualizzare gli attuali Tirocinanti
askDocumentsToSign	Questo servizio permette all'Admin ed all'Ente di visualizzare i documenti da firmare
downloadPF	Questo servizio consente agli utenti di scaricare il Progetto Formativo



5. Glossario

A

Admin: Un tipo di utente registrato al sistema che raffigura il Consiglio Didattico del Dipartimento di Informatica degli Studi di Salerno.

Annullamento Tirocinio: nel sistema proposto è inteso come l'operazione che termina l'esperienza di Tirocinio Curriculare esterno dello studente per una sua personale scelta. Tale operazione può essere eseguita dalla Segreteria o dall'Admin.

C

Client: componente che accede a servizi e risorse di un altro componente detto Server.

D

DBMS: sistema software per creazione, manipolazione e interrogazione efficiente di database

Dipartimento: Il Dipartimento di informatica degli Studi di Salerno.

E

Ente Convenzionato: L'ente/azienda preposta al Tirocinio Esterno.

EV: Il Sistema "*English Validation*" sviluppato per la convalida dei CFU per la lingua inglese.

H

HTTP: protocollo di trasferimento di ipertesti che consente a due macchine, client e Server, di interagire attraverso un meccanismo di richiesta/risposta. Il client inoltra una richiesta al Server, che verrà soddisfatta con la risposta di quest'ultimo.

J

JSP: tecnologia di programmazione web utilizzata per fornire contenuti dinamici.

M

MVC: Architettura composta da Model, il quale fornisce i metodi per accedere ai dati utili all'applicazione, View, visualizza i dati contenuti nel model e si occupa dell'interazione con utenti e agenti, e Controller, riceve i comandi dell'utente e li attua modificando lo stato degli altri due componenti.

N

NRF: Requisito non funzionale.

P



Progetto Formativo: documento che contiene tutte le informazioni riguardanti un tirocinio.

R

RAD: Il documento dell'analisi dei requisiti è un'attività preliminare allo sviluppo di un sistema software. Lo scopo principale di tale documento è di definire le funzionalità del sistema.

RF: Un requisito funzionale è un requisito che definisce una funzione del sistema identificato durante l'analisi dei requisiti.

Rifiuto: tutte le richieste effettuabili possono essere rifiutate. Il rifiuto è solitamente una conseguenza di una mancanza di requisito oppure di dati incongruenti, in questo caso studente è tenuto a ripetere la richiesta.

Tale operazione può essere estesa a più attori: Ente, Segreteria ed Admin.

S

SDD: Il documento del System Design è un'attività preliminare allo sviluppo di un sistema software. Gli scopi del documento sono quelli di definire gli obiettivi di progettazione del sistema e di decomporre il sistema in sottosistemi più piccoli.

Segreteria: Tale figura svolge il ruolo di amministrazione didattica nell'ambito del Dipartimento d'appartenenza. Essa acquisisce una valenza significativa e nel sistema corrente ed in quello proposto.

Server: componente che gestisce traffico di informazioni e fornisce servizi e risorse attraverso la rete.

Servlet: oggetti Java all'interno del server web che permettono di creare web applications in combinazione con JSP.

Shutdown: spegnimento di un sistema

Sottosistema: un sistema subalterno e secondario rispetto a un sistema più complesso.

Startup: avvio di un sistema.

T

Tirocinio: Tirocinio Curriculare Esterno dedicato agli studenti del corso di studi del Dipartimento di Informatica.

U

UC: Uno o più Use Case vengono utilizzati durante il SDD per identificare l'interazione tra attore e sistema.

UNISA: Università degli Studi di Salerno.

Utente: Un qualsiasi utilizzatore della piattaforma.



Laurea Magistrale in informatica-Università di Salerno
Corso di *Gestione dei Progetti Software*- Prof.ssa F.Ferrucci

W

Web Browser: applicazione software installata sul client che permette di visualizzare e navigare.
le risorse del web.