

UNIVERSITÀ DEGLI STUDI DI NAPOLI FEDERICO II
CORSO DI LAUREA IN INGEGNERIA INFORMATICA
CORSO DI INGEGNERIA DEL SOFTWARE
PROF. R. PIETRANTUONO- A.A. 2021 - 22

Progetto

**SISTEMA DI MANAGEMENT DOCENTI,
CORSI E VERBALI PER ESAMI**

INDICE

1. SPECIFICHE INFORMALI

2. ANALISI E SPECIFICA DEI REQUISITI

- 2.1 REVISIONE DEI REQUISITI
- 2.2 GLOSSARIO DI PROGETTO
- 2.3 SOFTWARE REQUIREMENTS SPECIFICATION

3. CASI D'USO

- 3.1 ATTORI E CASI D'USO
- 3.2 USE CASE DIAGRAM
- 3.3 SCENARI

4. DIAGRAMMI DI ANALISI

- 4.1 DIAGRAMMI DELLE CLASSI
- 4.2 DIAGRAMMI DI SEQUENZA

5. STIMA DEI COSTI

6. PROGETTAZIONE

- 6.1 DIAGRAMMA DELLE CLASSI
- 6.2 DIAGRAMMI DI SEQUENZA

7. PIANO DI TEST FUNZIONALE E TESTING

ID17: Verbalì esami digitali

Si desidera realizzare un sistema per la verbalizzazione elettronica degli esami di un corso di laurea. La segreteria studenti utilizza il sistema per aggiungere e modificare docenti e corsi, e in ogni anno accademico per associare un docente all'edizione del corso di quell'anno. Un corso ha un codice, una denominazione e un numero di crediti formativi (CFU). I docenti hanno una matricola, un cognome e un nome. (Per semplicità, non è prevista suddivisione di un corso in più cattedre.) Il sistema deve consentire ai docenti di gestire gli appelli di esame per i propri corsi (creare nuovi appelli, ed elencare gli appelli di un a.a.). Per ciascun appello il docente deve indicare la data, la sede (aula, laboratorio o altro), la (data di) scadenza delle prenotazioni, ed eventuali note (testo semplice). Non sono possibili la modifica e la cancellazione di un appello inserito. Il docente dovrà inoltre poter visualizzare l'elenco dei prenotati per un appello nell'elenco. Gli studenti utilizzano il sistema per prenotarsi per un appello. I dati degli studenti (comprese le loro credenziali di accesso, username, password e PIN) provengono dall'anagrafe di Ateneo: periodicamente, il sistema preleva i dati degli studenti dal sistema Anagrafe Studenti. In occasione di un appello, i docenti creano un verbale per verbalizzare –uno alla volta -tutti gli esami della sessione, selezionando ogni studente esaminato dall'elenco dei prenotati e inserendo il voto finale e gli argomenti trattati (testo libero). All'atto della registrazione di un esame, lo studente deve inserire il proprio codice PIN. Gli studenti assenti all'appello devono essere cancellati dal docente dalla lista dei prenotati. Se il docente desidera consentire ad uno studente non prenotato di svolgere l'esame, deve poterlo aggiungere al verbale tramite la matricola. Terminati gli esami di un appello, il docente effettua la chiusura del verbale, con apposizione della firma digitale. Il verbale firmato viene automaticamente trasmesso alla segreteria studenti. I docenti devono poter stampare un report dei verbali di un insegnamento in un dato a.a., con gli studenti esaminati e i rispettivi voti.

REVISIONE DEI REQUISITI

1. Il sistema deve verbalizzare gli esami di un corso di laurea
2. Il sistema non richiede un servizio di autenticazione, la quale sarà gestita esternamente.
3. La segreteria studenti può aggiungere e modificare i docenti.
4. I docenti sono identificati univocamente da una matricola.
5. Dei docenti si vuole memorizzare la matricola, il cognome ed il nome ed i dati di accesso (username e password).
6. La segreteria studenti può aggiungere e modificare i corsi.
7. I corsi sono identificati univocamente da un codice.
8. Dei corsi si vuole memorizzare il codice, la denominazione e i CFU(crediti formativi).
9. Il sistema deve permettere ai docenti di creare nuovi appelli.
10. Il sistema deve permettere ai docenti di elencare gli appelli di un a.a.
11. Il sistema deve memorizzare, di un appello, la data, la sede, la data di scadenza delle prenotazioni e note.
12. Il sistema non deve permettere di modificare appelli inseriti.
13. Il sistema non deve permettere la cancellazione di appelli inseriti.
14. Il sistema deve permettere ad un docente di visualizzare gli studenti prenotati all'appello.
15. Il sistema deve periodicamente prelevare i dati degli studenti dall'Anagrafe di Ateneo.
16. Degli studenti si vuole memorizzare nome, cognome, matricola, username, password, pin.
17. Il sistema deve permettere agli studenti di prenotarsi ad un appello.
18. Il sistema deve permettere ai docenti di creare verbali per gli appelli.
19. Nel verbale si vuole memorizzare voto finale, argomenti trattati e matricola dello studente.
20. Il sistema deve richiedere il Pin dello studente per salvare i dati relativi all'esame in un verbale.
21. Il sistema deve permettere ai docenti di aggiungere studenti all'elenco prenotati.
22. Il sistema deve permettere ai docenti di cancellare studenti dall'elenco prenotati.
23. Il sistema deve permettere ai docenti di chiudere il verbale, apponendo la firma digitale.
24. Il sistema deve inviare automaticamente il verbale firmato alla segreteria studenti.
25. Il sistema deve permettere ai docenti di stampare un report dei verbali riguardanti un proprio insegnamento in un dato a.a.
26. Il report deve contenere la matricola degli studenti esaminati ed i rispettivi voti.

GLOSSARIO DI PROGETTO

- **Docente:**
 - personale addetto all'istruzione all'interno dell'università.
 - Sinonimi: Professore, Insegnante
- **Corso:**
 - Serie metodica di lezioni o trattazione sistematica di una disciplina, inquadrata nell'ambito di una preparazione tecnica o di un programma di studi
 - Sinonimi: Lezione, Classe, Seminario
- **Esame:**
 - Prova, variamente costituita e condotta, mediante la quale viene accertata la preparazione conseguita dagli studenti afferenti ad un determinato corso.
 - Sinonimi: Test, Prova, Verifica
- **Anno Accademico:**
 - Periodo di tempo in cui sono svolte le attività universitarie. Per la legislazione italiana, l'anno accademico va dal 1° novembre al 31 ottobre del successivo anno solare.
 - Sinonimi: Anno Universitario
- **Sessione:**
 - Periodo di tempo, successivo all'erogazione dei corsi, nei quali è possibile per gli studenti sostenere degli esami. Una sessione è caratterizzata dal periodo dell'anno solare in cui è collocata temporalmente: estiva, invernale.
- **Appello d'esame:**
 - Date in cui viene data l'opportunità di svolgere un esame legato ad un dato corso. Essi sono legati alla specifica sessione.
 - Sinonimi: Date d'esame
- **Studente:**
 - Iscritto a un corso di studi o a una facoltà universitaria.
 - Sinonimi: Allievo
- **Verbale:**
 - Documento redatto da un docente allo scopo di attestare e ricordare, in forma sintetica ma completa ed esauriente nei dati essenziali, informazioni riguardanti l'esame sostenuto da uno o più studenti.
 - Sinonimi: Registrazione, Resoconto
- **PIN studente:**
 - Codice numerico di 5 cifre utile per il riconoscimento univoco dello studente
- **Firma digitale:**
 - metodo matematico teso a dimostrare l'autenticità di un messaggio o di un documento digitale inviato tra mittente e destinatario attraverso un canale di comunicazione non sicuro, garantendo al destinatario le proprietà di autenticazione-il mittente sia chi dice di essere-, non ripudio-il mittente non può negare di aver inviato il messaggio-, e integrità-il messaggio non sia stato alterato.
- **Report:**
 - prospetto riepilogativo di una raccolta di dati, riguardante i verbali relativi ad un determinato corso in un dato anno accademico.
 - Sinonimi: Rapporto, Relazione

Software Requirement Specification

La struttura dei Requisiti è <ID><SISTEMA><TIPO><PRIORITÀ>, ove:

- <ID> è un identificativo univoco del requisito;
- <SISTEMA> è il sistema a cui si attribuisce;
- <TIPO> indica la tipologia del requisito; esempi sono:
 - FUN: indica un requisito su una funzionalità del sistema;
 - PRE: indica un requisito prestazionale;
 - PER: indica un requisito di persistenza dei dati;
 - AFF: indica un requisito di affidabilità;
 - DIS: indica un requisito di disponibilità del sistema software;
 - SIC: indica un requisito di sicurezza.
- <PRIORITÀ> può assumere valori quali “*must*”, “*could*”, ...;

<DOC.1> <SVE> <PER> <*must*>: Il sistema deve memorizzare tutte le informazioni riguardanti i docenti.

<COR.2> <SVE> <PER> <*must*>: Il sistema deve memorizzare tutte le informazioni riguardanti i corsi.

<INS.1> <SVE> <FUN> <*must*>: Il sistema deve permettere l’inserimento di nuovi docenti.

<INS.2> <SVE> <FUN> <*must*>: Il sistema deve permettere l’inserimento di nuovi corsi.

<DOC.2> <SVE> <FUN> <*must*>: Il sistema deve permettere la modifica di docenti già inseriti.

<COR.2> <SVE> <FUN> <*must*>: Il sistema deve permettere la modifica di corsi già inseriti.

<ASS.1> <SVE> <FUN> <*must*>: Il sistema deve permettere di associare un docente all’edizione del corso legata all’anno accademico.

<APP.1> <SVE> <FUN> <*must*>: Il sistema deve permettere ai docenti di gestire gli appelli d’esame.

<APP.2> <SVE> <FUN> <*must not*>: Il sistema non deve permettere ai docenti di modificare gli appelli d’esame inseriti.

<APP.3> <SVE> <FUN> <*must not*>: Il sistema non deve permettere ai docenti di cancellare gli appelli d’esame inseriti.

<APP.4> <SVE> <FUN> <*must*>: Il sistema deve permettere ai docenti di visualizzare l’elenco dei prenotati agli appelli d’esame.

<PRE.1> <SVE> <FUN> <*must*>: Il sistema deve permettere agli studenti di prenotarsi per un appello.

<PRE.2> <SVE> <PER> <*must*>: Il sistema deve memorizzare tutte le informazioni riguardanti le prenotazioni.

<STU.1> <SVE> <PER> <*must*>: Il sistema deve memorizzare tutte le informazioni riguardanti gli studenti.

<STU.2> <SVE> <FUN> <*must*>: Il sistema deve acquisire periodicamente tutte le informazioni riguardanti gli studenti dall’Anagrafe Studenti.

<VER.1> <SVE> <FUN> <*must*>: Il sistema deve permettere ai docenti di creare un verbale.

<VER.2> <SVE> <FUN> <*must*>: Il sistema deve permettere ai docenti di verbalizzare gli esami dell’appello.

<VER.3> <SVE> <SIC> <must>: Il sistema deve richiedere l'inserimento del codice PIN dello studente per procedere alla verbalizzazione dell'esame.

<VER.4> <SVE> <FUN> <must>: Il sistema deve permettere ai docenti di cancellare uno studente da un verbale.

<VER.5> <SVE> <FUN> <must>: Il sistema deve permettere ai docenti di aggiungere uno studente ad un verbale.

<VER.6> <SVE> <PER> <must>: Il sistema deve memorizzare le informazioni riguardanti i verbali.

<VER.7> <SVE> <FUN> <must>: Il sistema deve permettere ai docenti di apporre una firma digitale sui verbali.

<VER.8> <SVE> <SIC> <must>: Il sistema deve verificare che il PIN inserito dallo studente sia corretto.

<VER.9> <SVE> <FUN> <must>: Il sistema deve inviare automaticamente un verbale alla segreteria, contestualmente alla chiusura dello stesso.

<REP.1> <SVE> <FUN> <must>: Il sistema deve permettere ai docenti di stampare un report dei verbali in un dato anno accademico.

Attori Primari:

Segreteria Studenti, Docente, Studente, Tempo

Attori Secondari:

Studente, Anagrafe Studenti

Casi d'uso:

Aggiungi Docente, Aggiungi Corso, Modifica Docente, Modifica Corso, AssociazioneDocenteCorso, CreaAppello, VisualizzaElencoPrenotati, ElencaAppelli, StampaReport, PrenotazioneAppello, PrelievoDatiStudente

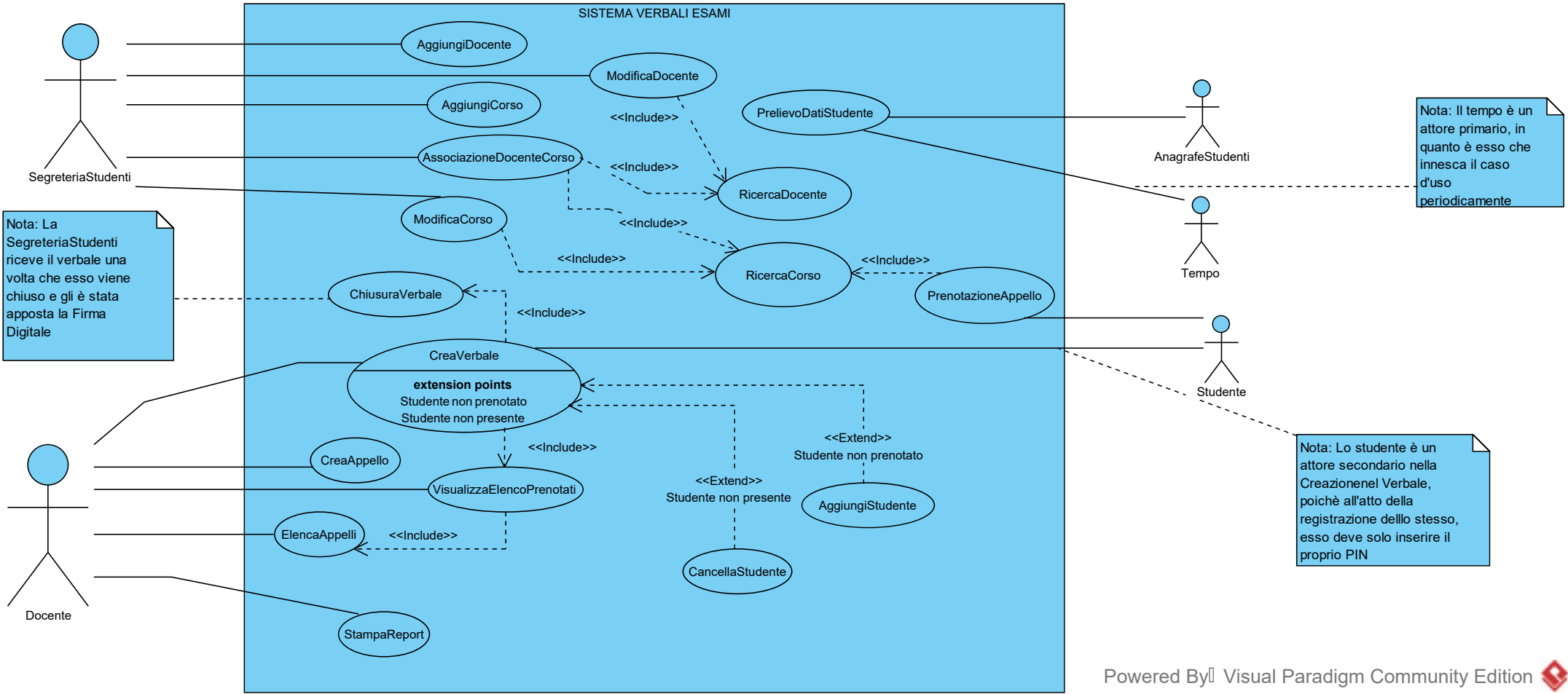
Casi d'uso di inclusione:

RicercaCorso, RicercaDocente, ChiusuraVerbale, VisualizzaElencoPrenotati, ElencaAppelli

Casi d'uso di estensione:

AggiungiStudente, CancellaStudente

Caso d'uso	Attori Primari	Attori <u>secondari</u>	Inclusione/Estensione
AggiungiDocente	SegreteriaStudenti
AggiungiCorso	SegreteriaStudenti
ModificaDocente	SegreteriaStudenti	...	I: RicercaDocente
ModificaCorso	SegreteriaStudenti	...	I: RicercaCorso
AssociazioneDocenteCorso	SegreteriaStudenti	...	I: RicercaCorso, I: RicercaDocente
Crea Verbale	Docente	Studente	I: ChiusuraVerbale I: VisualizzaElencoPrenotati E: CancellaStudente E: AggiungiStudente
CreaAppello	Docente
VisualizzaElencoPrenotati	Docente	...	I: ElencaAppelli
ElencaAppelli	Docente
StampaReport	Docente
PrelievoDatiStudente	Tempo	AnagrafeStudenti	...
PrenotazioneAppello	Studente	...	RicercaCorso



Caso d' uso

<nome>

ID	AGG_DOCENTE
BREVE DESCRIZIONE	Creazione e aggiunta di un nuovo docente
ATTORI PRIMARI	SegreteriaStudenti
ATTORI SECONDARI	Nessuno
PRECONDIZIONI	Nessuna
SEQUENZA DEGLI EVENTI PRINCIPALI	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando la SegreteriaStudenti seleziona "aggiungi docente" 2. La SegreteriaStudenti inserisce matricola, nome e cognome del docente da registrare. 3. Il sistema aggiorna il database con il nuovo docente. 4. Il sistema risponde con un messaggio di conferma del corretto inserimento.
POSTCONDIZIONI	Il database contiene il nuovo docente
SEQUENZA DEGLI EVENTI ALTERNATIVA	Al punto 2., se la matricola risulta scorretta o se non è univoca, cioè è già presente un docente con tale matricola associata, il sistema notifica un errore e richiede di modificare l'informazione scorretta.

ID	MOD_DOCENTE
BREVE DESCRIZIONE	Modifica di un docente esistente.
ATTORI PRIMARI	SegreteriaStudenti
ATTORI SECONDARI	Nessuno
PRECONDIZIONI	Nessuna
SEQUENZA DEGLI EVENTI PRINCIPALI	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando la SegreteriaStudenti seleziona "modifica docente" 2. Include(RicercaDocente). 3. La SegreteriaStudenti effettua la selezione del docente da modificare. 4. La SegreteriaStudenti modifica il campo o i campi che desidera modificare. 5. Il sistema verifica la consistenza delle informazioni ed aggiorna il database. 6. Il sistema risponde con un messaggio di conferma del corretto inserimento.
POSTCONDIZIONI	Il database contiene le informazioni aggiornate sul docente modificato.
SEQUENZA DEGLI EVENTI ALTERNATIVA	<ul style="list-style-type: none"> • Al punto 5., se la matricola risulta scorretta o se non è univoca, cioè è già presente un docente con tale matricola associata, il sistema notifica un errore e richiede di modificare l'informazione scorretta.

ID	RICERCA_DOC
BREVE DESCRIZIONE	Ricerca di un docente esistente.
ATTORI PRIMARI	SegreteriaStudenti
ATTORI SECONDARI	Nessuno
PRECONDIZIONI	Nessuna
SEQUENZA DEGLI EVENTI PRINCIPALI	<ol style="list-style-type: none"> 1. Il sistema effettua una ricerca del docente secondo la dupla [nome,cognome]. 2. Il sistema restituisce la lista dei docenti.
POSTCONDIZIONI	Viene restituita la lista dei docenti ricercati, se presenti.
SEQUENZA DEGLI EVENTI ALTERNATIVA	<ul style="list-style-type: none"> • Al punto 1. La ricerca può essere effettuata anche secondo la matricola. In questo caso il risultato è un unico docente. • Al punto 1., se non presente alcun risultato possibile per la ricerca, il sistema notifica un errore.

ID	AGG_CORSO
BREVE DESCRIZIONE	Creazione e aggiunta di un nuovo corso
ATTORI PRIMARI	SegreteriaStudenti
ATTORI SECONDARI	Nessuno
PRECONDIZIONI	Nessuna
SEQUENZA DEGLI EVENTI PRINCIPALI	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando la SegreteriaStudenti seleziona "aggiungi corso" 2. La SegreteriaStudenti inserisce codice, denominazione e CFU del corso da registrare. 3. Il sistema aggiorna il database con il nuovo corso. 4. Il sistema risponde con un messaggio di conferma del corretto inserimento.
POSTCONDIZIONI	Il database contiene il nuovo corso
SEQUENZA DEGLI EVENTI ALTERNATIVA	Al punto 2., se il codice risulta scorretto o se non è univoco, cioè è già presente un corso con tale codice associato, il sistema notifica un errore e richiede di modificare l'informazione scorretta.

ID	MOD_CORSO
BREVE DESCRIZIONE	Modifica di un corso esistente.
ATTORI PRIMARI	SegreteriaStudenti
ATTORI SECONDARI	Nessuno
PRECONDIZIONI	Nessuna.
SEQUENZA DEGLI EVENTI PRINCIPALI	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando la SegreteriaStudenti seleziona "modifica corso" 2. Include(RicercaCorso) 3. La SegreteriaStudenti effettua la selezione del corso da modificare. 4. La SegreteriaStudenti modifica il campo o i campi che desidera modificare. 5. Il sistema verifica la consistenza delle informazioni ed aggiorna il database. 6. Il sistema risponde con un messaggio di conferma del corretto inserimento.
POSTCONDIZIONI	Il database contiene le informazioni aggiornate sul corso modificato.
SEQUENZA DEGLI EVENTI ALTERNATIVA	<ul style="list-style-type: none"> • Al punto 5., se il codice risulta scorretto o se non è univoco, cioè è già presente un corso con tale codice associato, il sistema notifica un errore e richiede di modificare l'informazione scorretta.

ID	RICERCA_CORSO
BREVE DESCRIZIONE	Ricerca di un corso esistente.
ATTORI PRIMARI	SegreteriaStudenti
ATTORI SECONDARI	Nessuno
PRECONDIZIONI	Nessuna
SEQUENZA DEGLI EVENTI PRINCIPALI	<ol style="list-style-type: none"> 1. Il sistema effettua una ricerca del corso secondo la denominazione. 3. Il sistema restituisce la lista dei corsi.
POSTCONDIZIONI	Viene restituita la lista dei docenti ricercati, se presenti.
SEQUENZA DEGLI EVENTI ALTERNATIVA	<ul style="list-style-type: none"> • Al punto 1. La ricerca può essere effettuata anche secondo il codice. In questo caso il risultato è un unico corso. • Al punto 1. La ricerca può essere effettuata anche secondo la matricola del docente o per la dupla [nome,cognome] del docente. In questo caso viene restituita una lista dei corsi associati al docente. • Al punto 1., se non presente alcun risultato possibile per la ricerca, il sistema notifica un errore.

ID	ASSOC_DOC_CORSO
BREVE DESCRIZIONE	Associazione di un docente con un corso per l'anno accademico
ATTORI PRIMARI	SegreteriaStudenti
ATTORI SECONDARI	Nessuno
PRECONDIZIONI	Un corso è associabile ad un solo docente.
SEQUENZA DEGLI EVENTI PRINCIPALI	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando la SegreteriaStudenti seleziona "Associa Docente Corso" 2. Include(RicercaCorso) 3. La SegreteriaStudenti seleziona il corso che vuole associare ad un docente. 4. Il sistema verifica che il corso sia associabile ad un docente, cioè che non sia già associato ad un altro docente. 5. Include(RicercaDocente) 6. La SegreteriaStudenti seleziona il docente che vuole associare al corso. 7. Il sistema richiede una conferma dell'inserimento, con un riepilogo delle selezioni. 8. Se l'operazione è confermata: <ol style="list-style-type: none"> a. Il sistema risponde con un messaggio di conferma del corretto inserimento. 9. Altrimenti: <ol style="list-style-type: none"> a. Il sistema notifica l'annullamento e ritorna al punto 1.
POSTCONDIZIONI	Il database contiene la nuova associazione docente-corso.
SEQUENZA DEGLI EVENTI ALTERNATIVA	<ul style="list-style-type: none"> • Al punto 4., se il corso è già associato ad un docente, il sistema notifica il fatto e richiede se si vuole modificare l'associazione. Se confermato: <ul style="list-style-type: none"> ○ Continua con il punto 5. Altrimenti: <ul style="list-style-type: none"> ○ Il sistema notifica l'annullamento e ritorna al punto 1. • Al punto 7., se l'operazione è una modifica di una associazione precedentemente esistente, il sistema mostra un riepilogo dello stato precedente e delle nuove selezioni.

ID	CREA_APPELLO
BREVE DESCRIZIONE	Creazione e aggiunta di un nuovo appello
ATTORI PRIMARI	Docente
ATTORI SECONDARI	Nessuno
PRECONDIZIONI	Il docente deve essere associato al corso per cui crea l'appello.
SEQUENZA DEGLI EVENTI PRINCIPALI	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando il Docente seleziona "Crea appello". 2. Il sistema mostra i corsi a cui il docente è associato. 3. Il Docente seleziona il corso per il quale vuole creare un appello. 4. Il Docente inserisce le informazioni riguardanti l'appello: data, sede, data di scadenza prenotazioni, note. 5. Il sistema richiede una conferma dell'inserimento, fornendo un riepilogo delle informazioni inserite. 6. Se l'operazione è confermata: <ol style="list-style-type: none"> 6.1 Il sistema notifica il successo dell'operazione e registra il nuovo appello. 7. Altrimenti: <ol style="list-style-type: none"> 7.1 il sistema notifica l'annullamento ed annulla l'operazione.
POSTCONDIZIONI	Il nuovo appello è stato creato.
SEQUENZA DEGLI EVENTI ALTERNATIVA	Al punto 4., se la data di scadenza prenotazioni è successiva alla data dell'appello, il sistema notifica l'errore e imposta la data di scadenza prenotazioni al giorno precedente la data dell'appello.

ID	ELENCA_APP
BREVE DESCRIZIONE	Elenco degli appelli di un anno accademico
ATTORI PRIMARI	Docente
ATTORI SECONDARI	Nessuno
PRECONDIZIONI	Il docente deve essere associato ai corsi per cui elenca gli appelli.
SEQUENZA DEGLI EVENTI PRINCIPALI	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando il Docente seleziona "Elenca Appello" 2. Il sistema mostra a schermo tutti gli appelli associati al Docente nel dato anno accademico.
POSTCONDIZIONI	Nessuna
SEQUENZA DEGLI EVENTI ALTERNATIVA	Nessuna

ID	VIS_ELENCO_PREN
BREVE DESCRIZIONE	Visualizza l'elenco dei prenotati ad un appello
ATTORI PRIMARI	Docente
ATTORI SECONDARI	Nessuno
PRECONDIZIONI	Il docente deve essere associato ai corsi per cui visualizza i prenotati all'appello.
SEQUENZA DEGLI EVENTI PRINCIPALI	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando il Docente seleziona "Visualizza Elenco Prenotati" 2. Include(ElencaAppelli) 3. Il Docente seleziona l'appello per il quale vuole visualizzare l'elenco dei prenotati. 4. Il sistema mostra a schermo l'elenco dei prenotati al dato appello.
POSTCONDIZIONI	Nessuna
SEQUENZA DEGLI EVENTI ALTERNATIVA	Nessuna

ID	CREA_VERBALE
BREVE DESCRIZIONE	Crea il Verbale relativo ad un esame
ATTORI PRIMARI	Docente
ATTORI SECONDARI	SegreteriaStudente, Studente
PRECONDIZIONI	Il docente deve essere associato al corso per cui crea il verbale d'esame.
SEQUENZA DEGLI EVENTI PRINCIPALI	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando il Docente seleziona "Crea Verbale" 2. include(VisualizzaElencoPrenotati) 3. Se lo studente è nell'elenco dei prenotati all'appello d'esame: <ol style="list-style-type: none"> 3.1 Il Docente seleziona lo studente dall'elenco. 4. Altrimenti: <ol style="list-style-type: none"> 4.1 punto di estensione(Studente non prenotato) 4.2 Il sistema seleziona automaticamente lo studente appena aggiunto all'elenco. 5. Il sistema mostra i dati anagrafici e la matricola dello studente e richiede se il Docente desidera aprire il verbale per l'esame sostenuto dallo studente. 6. Se il Docente conferma: <ol style="list-style-type: none"> 6.1 Il Docente inserisce il voto finale e gli argomenti trattati. 7. Altrimenti: <ol style="list-style-type: none"> 7.1 Il sistema richiede se si vuole eliminare lo studente selezionato. 7.2 Se il Docente conferma: <ol style="list-style-type: none"> 7.2.1 punto di estensione(Studente non prenotato) 7.2.2 ritorna al punto 2. 7.3 Altrimenti: <ol style="list-style-type: none"> 7.3.1 ritorna al punto 2. 8. Lo Studente inserisce il proprio PIN. 9. Il sistema verifica le informazioni e richiede la conferma dell'operazione. 10. Se l'operazione è confermata:

	<p>10.1 Il verbale dell'esame viene registrato.</p> <p>11. Altrimenti:</p> <p>11.1 si ritorna al punto 8.</p> <p>12. Il sistema mostra il riepilogo del verbale con i dati anagrafici e la matricola dello Studente, il voto finale e gli argomenti trattati.</p> <p>13. Il sistema aggiorna lo studente nell'elenco dei prenotati con il voto e rendendolo non più selezionabile.</p> <p>14. Il sistema richiede al Docente se si desidera chiudere il verbale.</p> <p>15. Se il Docente conferma di voler chiudere il verbale:</p> <p>15.1 Il Docente appone la Firma Digitale.</p> <p>15.2 Il sistema richiede la conferma della chiusura del verbale.</p> <p>15.3 Se ricevuta conferma:</p> <p>15.3.1 Il sistema invia il verbale firmato alla SegreteriaStudenti.</p> <p>15.4 Altrimenti:</p> <p>15.4.1 ritorna al punto 14.</p> <p>16. Altrimenti:</p> <p>15.1 ritorna al punto 2.</p>
POSTCONDIZIONI	Il verbale firmato è stato inviato alla SegreteriaStudenti.
SEQUENZA DEGLI EVENTI ALTERNATIVA	<ul style="list-style-type: none"> Al punto 9., se il PIN dello Studente non è corretto, il sistema notifica l'errore e richiede un nuovo inserimento.

ID	AGG_STUDENTE
BREVE DESCRIZIONE	Viene aggiunto uno studente all'elenco dei prenotati
ATTORI PRIMARI	Docente
ATTORI SECONDARI	Nessuno
PRECONDIZIONI	<ol style="list-style-type: none"> Il docente deve essere associato ai corsi per cui aggiunge un prenotato all'appello. Lo Studente non deve essere prenotato all'appello (Studente non prenotato)
SEQUENZA DEGLI EVENTI PRINCIPALI	<ol style="list-style-type: none"> Il Docente inserisce i dati dello Studente. Il sistema aggiunge lo studente all'elenco dei prenotati.
POSTCONDIZIONI	Lo Studente è stato aggiunto alla lista dei prenotati.
SEQUENZA DEGLI EVENTI ALTERNATIVA	Al punto 1., se i dati dello Studente risultano già nell'elenco dei prenotati o se sono errati (non è possibile trovarli nel sistema), allora viene notificato un errore e una richiesta di reinserimento.

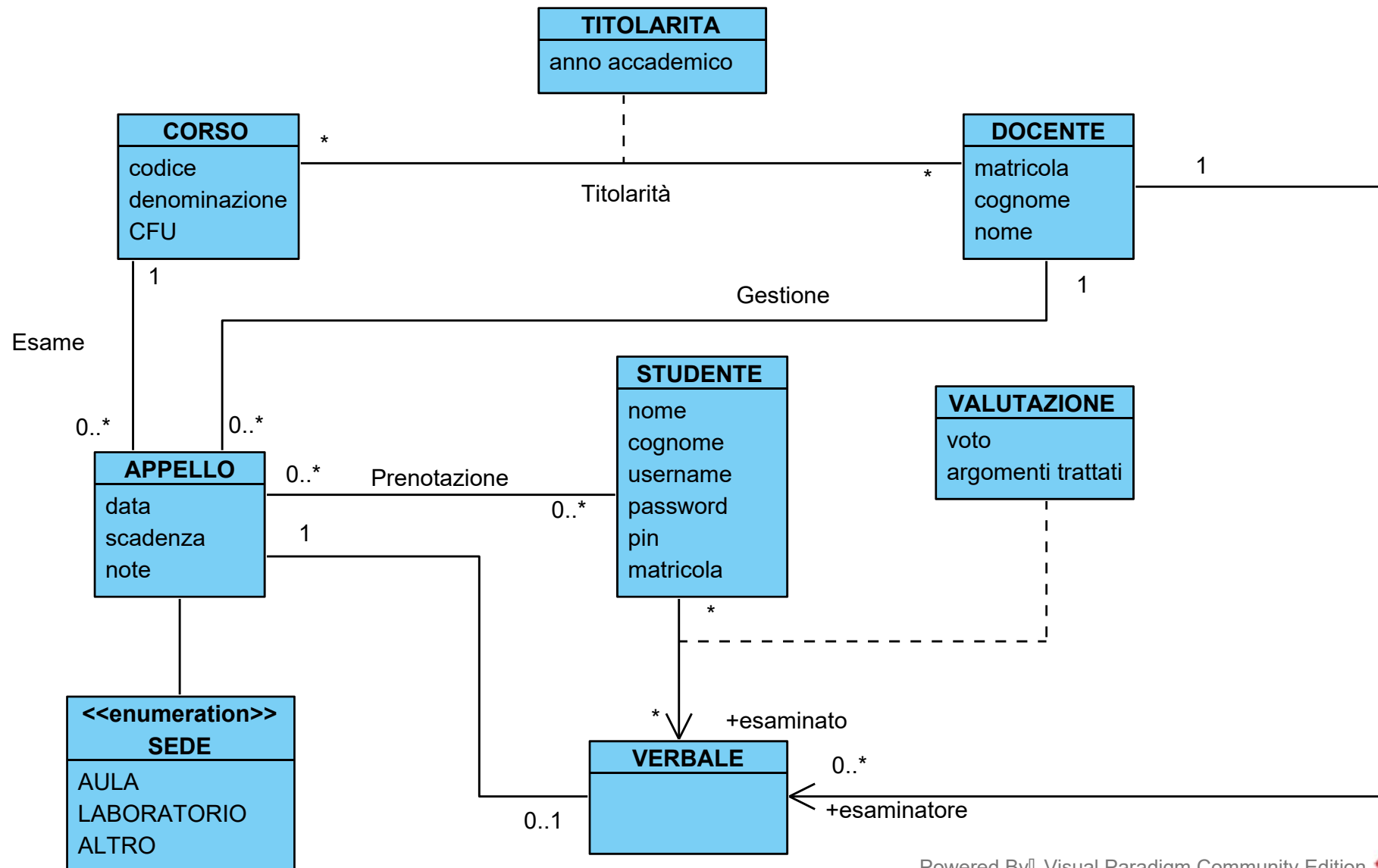
ID	STAMPA_REPORT
BREVE DESCRIZIONE	Viene stampato un report dei verbali di un insegnamento in un dato anno accademico.
ATTORI PRIMARI	Docente
ATTORI SECONDARI	Nessuno
PRECONDIZIONI	Il docente deve essere associato al corso per cui richiede la stampa del report.
SEQUENZA DEGLI EVENTI PRINCIPALI	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando il Docente seleziona "Stampa Report" 2. Il sistema mostra i corsi a cui il Docente è associato per il dato anno accademico. 3. Il Docente seleziona un corso. 4. Il sistema crea un file di report nel quale presenta tutti i verbali associati al corso, con gli studenti esaminati e i rispettivi voti.
POSTCONDIZIONI	Il report viene creato
SEQUENZA DEGLI EVENTI ALTERNATIVA	Al punto 3., se non è presente alcun verbale associato a tale corso, il sistema notifica l'errore e richiede una nuova selezione, togliendo il corso precedentemente selezionato dalla selezione. Se nessun altro corso è disponibile alla selezione, il sistema lo notifica e termina l'operazione.

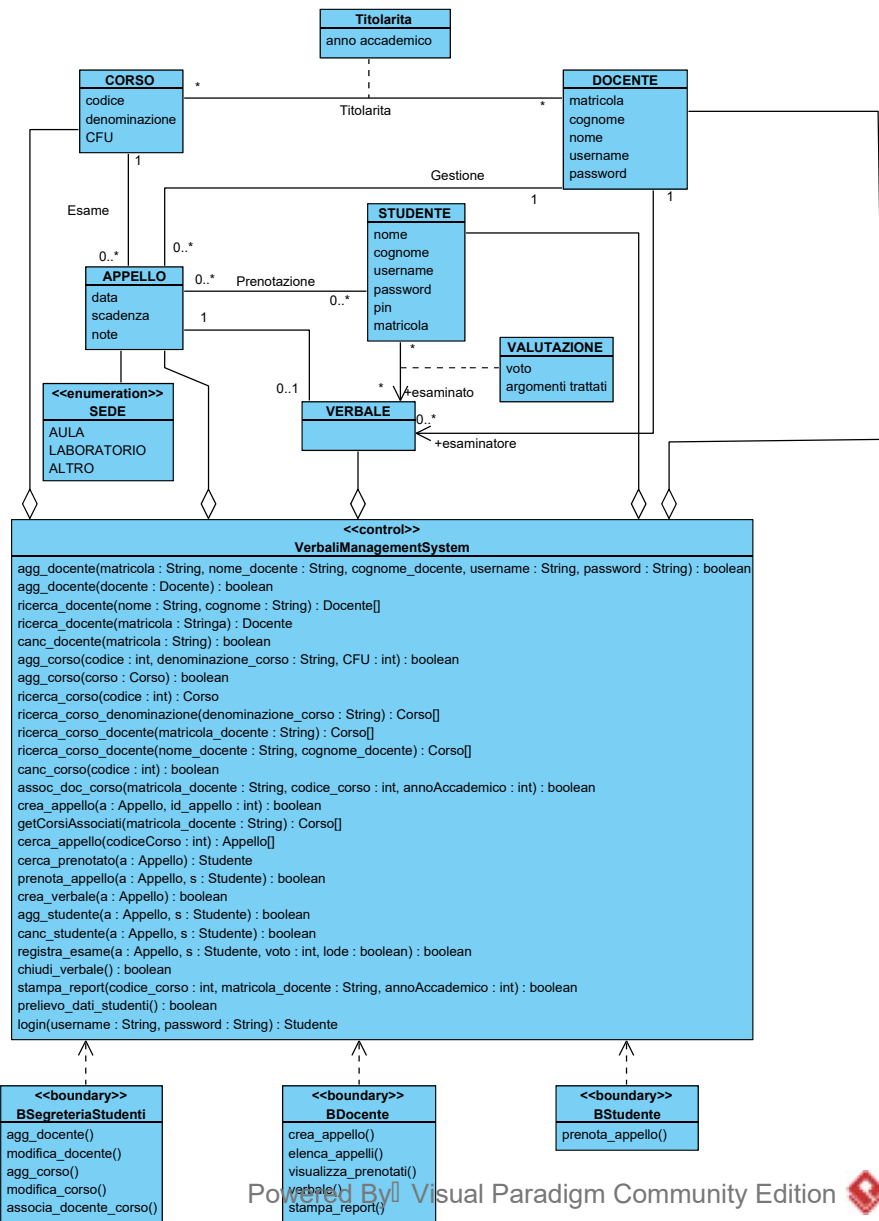
ID	PRENOTA_APP
BREVE DESCRIZIONE	Lo Studente prenota per partecipare all'appello d'esame
ATTORI PRIMARI	Studente
ATTORI SECONDARI	Nessuno
PRECONDIZIONI	Nessuno
SEQUENZA DEGLI EVENTI PRINCIPALI	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando lo Studente seleziona "Prenota Appello" 2. Include(RicercaCorso) 3. Se lo studente seleziona un corso: <ol style="list-style-type: none"> 3.1 Prosegue 4. Altrimenti: <ol style="list-style-type: none"> 4.1 Il sistema notifica l'annullamento e ritorna al punto 2. 5. Se il corso presenta degli appelli creati dal Docente: <ol style="list-style-type: none"> 5.1 Il sistema mostra gli appelli disponibili per tale corso. 6. Altrimenti: <ol style="list-style-type: none"> 6.1 Il sistema notifica l'assenza di appelli associati al corso. 6.2 Il sistema richiede se si vuole selezionare un altro corso. 6.3 Se lo Studente conferma: <ol style="list-style-type: none"> 6.3.1 Torna al punto 2 6.4 Altrimenti: <ol style="list-style-type: none"> 6.4.1 Termina operazione 7. Lo Studente seleziona un appello. 8. Il sistema mostra un riepilogo delle informazioni dell'appello e richiede la conferma della prenotazione.

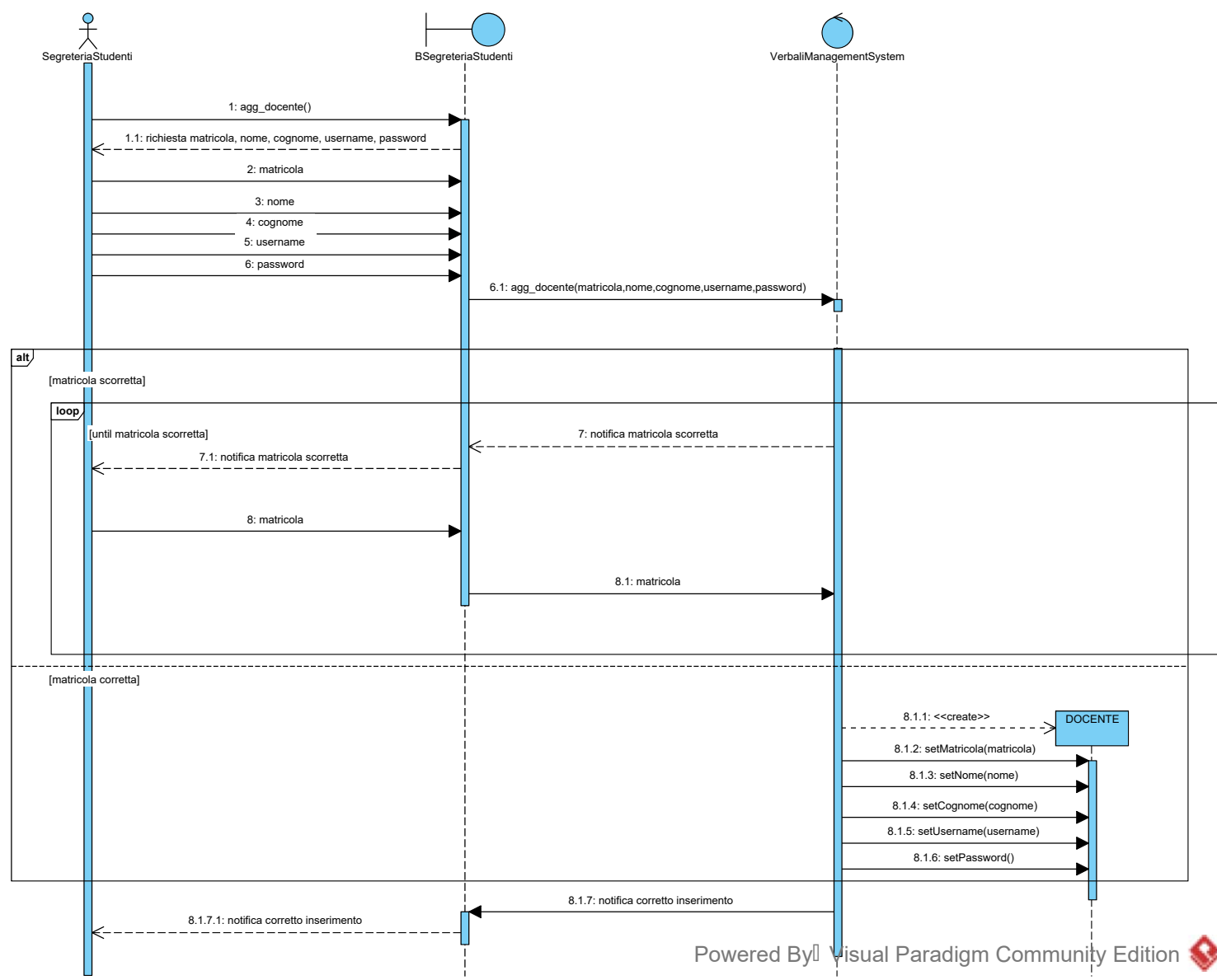
	9. Se lo Studente conferma: 9.1 Il sistema aggiunge lo studente alla lista prenotati per quell'appello. 9.2 Il sistema notifica l'avvenuta prenotazione e crea una ricevuta di prenotazione. 10. Altrimenti: 10.1 Termina Operazione
POSTCONDIZIONI	Lo studente viene aggiunto alla lista dei prenotati ad un appello.
SEQUENZA DEGLI EVENTI ALTERNATIVA	...

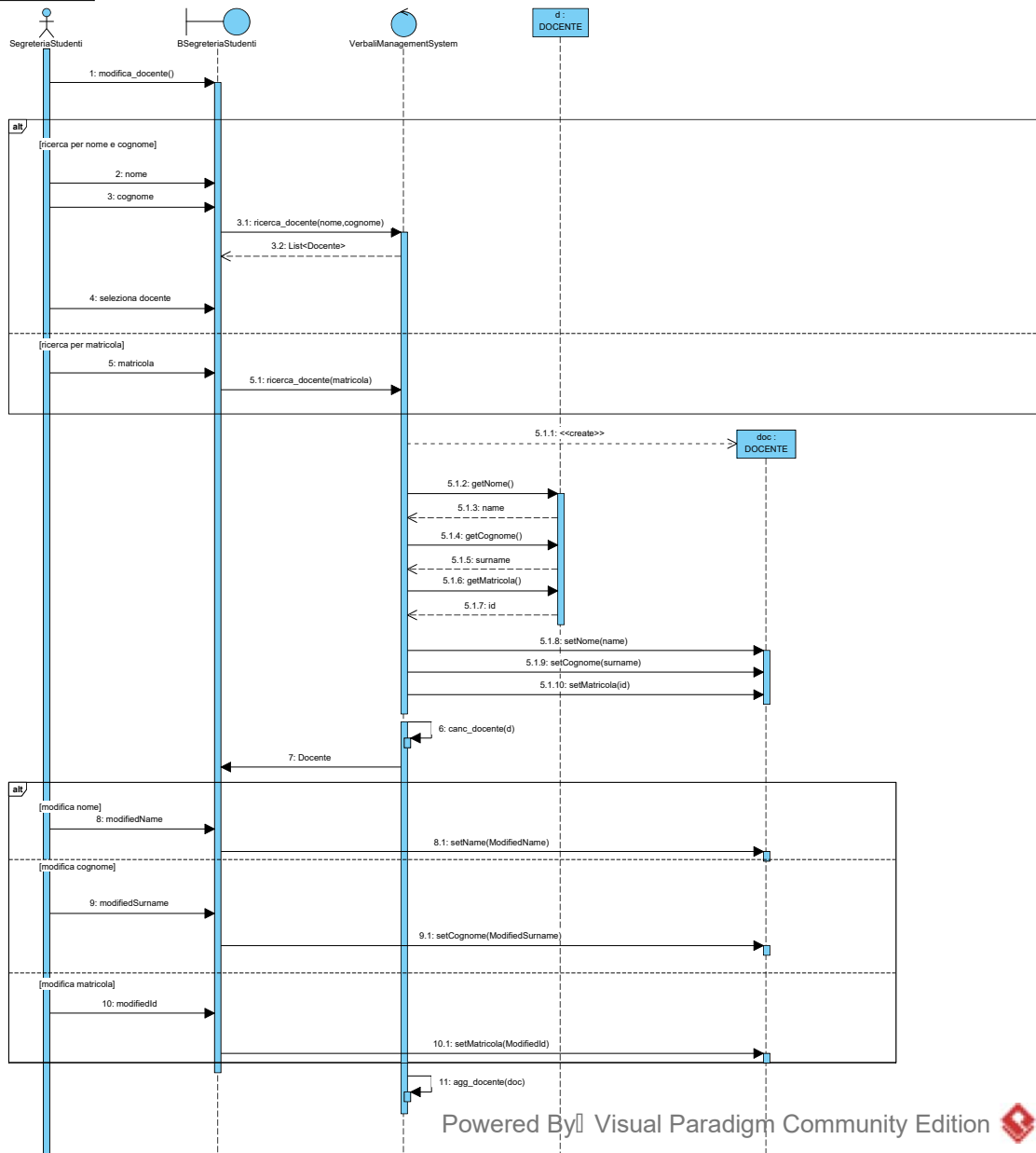
ID	PRELIEVO_DATI
BREVE DESCRIZIONE	Vengono prelevati e salvati i dati riguardanti gli studenti dall'AnagrafeStudenti
ATTORI PRIMARI	Sistema
ATTORI SECONDARI	AnagrafeStudenti, Tempo
PRECONDIZIONI	Nessuna
SEQUENZA DEGLI EVENTI PRINCIPALI	1. Il caso d'uso inizia quando Tempo seleziona "Prelievo Dati Studente" 2. Il sistema richiede i dati degli studenti dal sistema AnagrafeStudenti. 3. Il sistema riceve i dati ed aggiorna il suo Database. 4. Il sistema resetta Tempo.
POSTCONDIZIONI	Vengono aggiornati i dati riguardanti gli studenti.
SEQUENZA DEGLI EVENTI ALTERNATIVA	Al punto 3., se non è presente alcuna variazione nei dati rispetto all'operazione precedente, l'operazione termina senza modificare nulla.

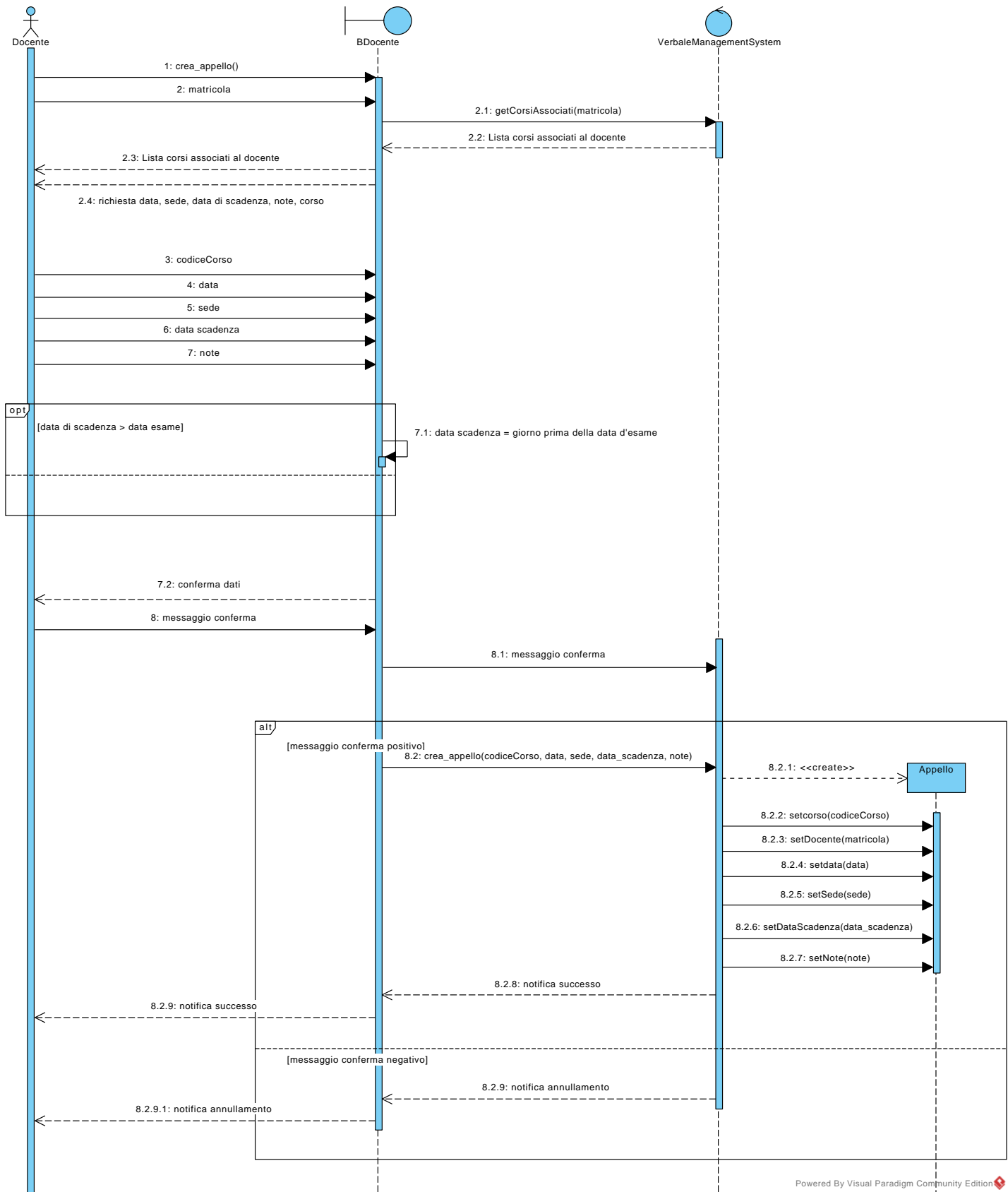
ID	CANC_STUDENTE
BREVE DESCRIZIONE	Viene eliminato uno studente dall'elenco dei prenotati
ATTORI PRIMARI	Docente
ATTORI SECONDARI	Nessuno
PRECONDIZIONI	1. Il docente deve essere associato ai corsi per cui elimina un prenotato all'appello. 2. Lo Studente non deve essere presente all'appello (Studente non presente)
SEQUENZA DEGLI EVENTI PRINCIPALI	1. punto di estensione: Studente non presente 2. Il sistema mostra il riepilogo della prenotazione dello studente e notifica il cancellamento. 3. Il sistema elimina lo studente dalla lista dei prenotati.
POSTCONDIZIONI	Lo Studente è stato eliminato dalla lista dei prenotati.
SEQUENZA DEGLI EVENTI ALTERNATIVA	Nessuna

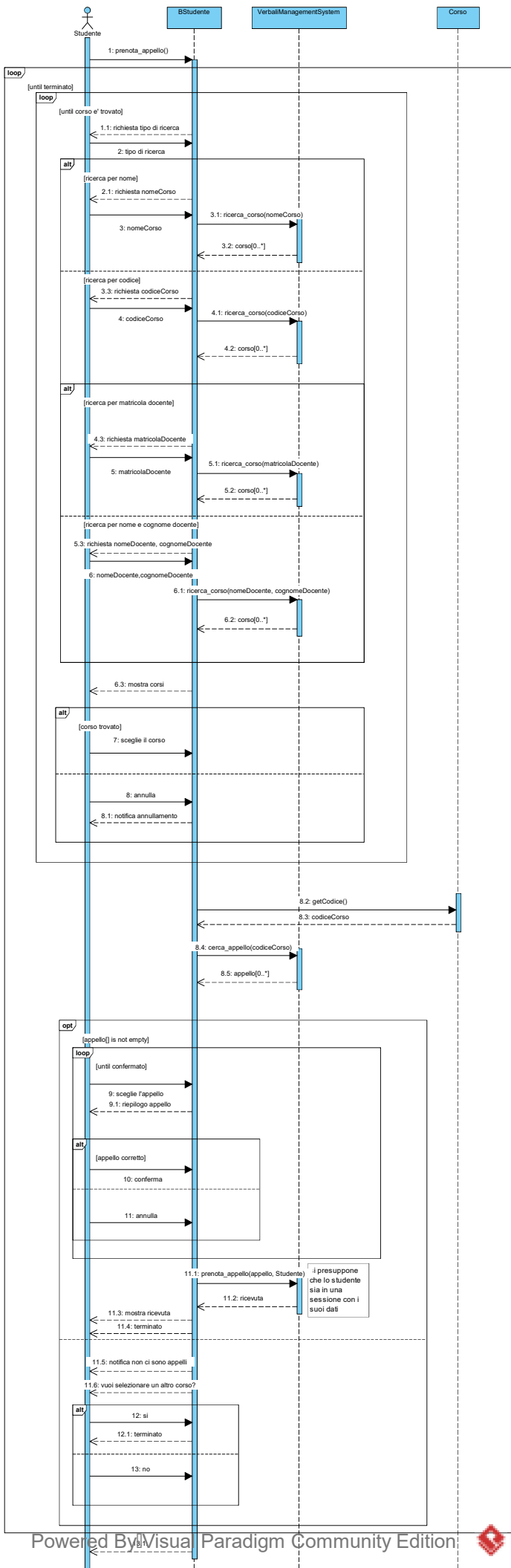












Function Point Analysis

Entità riscontrate

- Segreteria Studenti
- Docente
- Corso
- Studente
- Appello
- Verbale
- Report
- Anagrafe Studenti

Funzionalità riscontrate

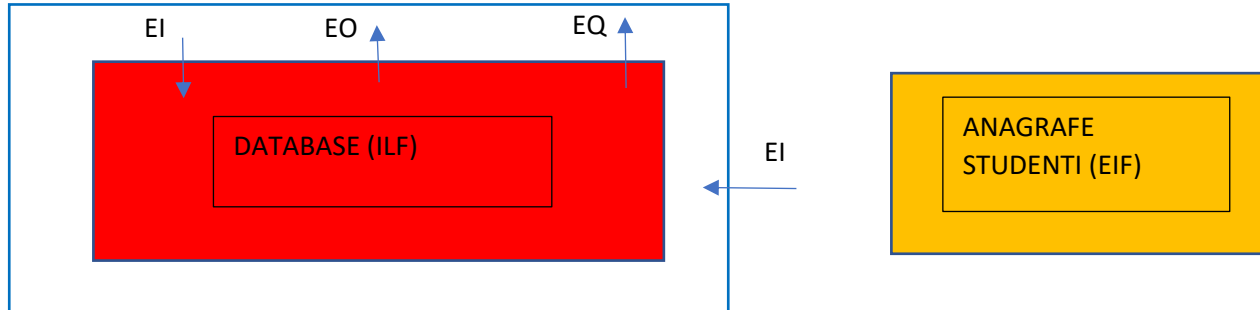
- Aggiunta Corso
- Ricerca Corso
- Modifica Corso
- Aggiunta Docente
- Ricerca Docente
- Modifica Docente
- Assegnazione Corso
- Creazione Appello
- Elenco Appelli
- Visualizzazione Elenco Prenotati Appello
- Prenotazione Appello
- Prelievo Dati Studente
- Creazione Verbale
- Aggiunta Studente al Verbale
- Cancellazione Studente dal Verbale
- Chiusura Verbale
- Stampa Report

STIMA COSTI

L'ambito di conteggio è "sviluppo progetto"

Il confine è:

EXTERNAL USER



CONTEGGIO ILF ed EIF e CALCOLO COMPLESSITÀ

È presente una base di dati per registrare **docenti**, **corsi**, **studenti**, **appelli** e **verbali**; **1 ILF**

Il sistema collabora con il sistema esterno **Anagrafe Studenti**; **1 EIF**

NILF = 1

- Tabella Docenti: 1 RET, < 5 DET;
- Tabella Corso: 1 RET, < 5 DET;
- Tabella Studente: 1 RET, < 5 DET;
- Tabella Appello: 1 RET, < 10 DET;
- Tabella Verbale: 1 RET, < 5 DET;

Totale: 5 RET, <50 DET **Rating:** AVG x10

NEIF = 1

- Anagrafe Studenti: 1 RET, < 20 DET;

Totale: 1 RET, <20 DET **Rating:** AVG x5

FUNZIONALITÀ

1. Aggiunta Corso

La funzionalità **Aggiungere Corso** modifica l'ILF, pertanto è conteggiabile come un **EI**.

Nell'aggiungere un corso, la Segreteria Studenti specifica **codice**, **denominazione** e **CFU**, cioè **3 DET**.

A questi, va aggiunto l'input per la richiesta della funzione (es. tasto) e il messaggio di risposta. (es. messaggio di errore).

FTR: 1 (Corso) **DET:** 5 **Rating:** LOW x3

2. Modifica Corso

La funzionalità **Modifica Corso** modifica l'ILF, pertanto è conteggiabile come un **EI**.

La Segreteria Studenti effettua una ricerca (già conteggiata), seleziona il corso e fa un update.

FTR: 1 (Corso) **DET:** 5 **Rating:** LOW x3

3. **Ricerca Corso**

La funzionalità **Ricerca Corso** non modifica l'ILF, pertanto è conteggiabile come un **EQ**.
Essa può essere effettuata per **codice**, **denominazione** o la coppia.

FTR: 1 (Corso) **DET:** 3 **Rating:** LOW x3

4. **Ricerca Docente**

La funzionalità **Ricerca Docente** non modifica l'ILF, pertanto è conteggiabile come un **EQ**.
Essa può essere effettuata per **matricola**(univoca) o la coppia **nome** e **cognome**.

FTR: 1 (Docente) **DET:** 2 **Rating:** LOW x3

5. **Aggiunta Docente**

La funzionalità **Aggiunta Docente** modifica l'ILF, pertanto è conteggiabile come un **EI**.
Nell'aggiungere un docente, la Segreteria Studenti specifica **matricola**, **cognome**, **nome**, **username** e **password**, cioè **5 DET**.
A questi, va aggiunto l'input per la richiesta della funzione (es. tasto) e il messaggio di risposta. (es. messaggio di errore).

FTR: 1 (Docente) **DET:** 7 **Rating:** LOW x3

6. **Modifica Docente**

La funzionalità **Modifica Docente** modifica l'ILF, pertanto è conteggiabile come un **EI**.
La Segreteria Studenti effettua una ricerca (già conteggiata), seleziona il docente e fa un update

FTR: 1 (Docente) **DET:** 7 **Rating:** LOW x3

7. **Assegnazione Corso**

La funzionalità **Assegnazione Corso** modifica l'ILF, pertanto è conteggiabile come un **EI**.
La Segreteria Studenti effettua due ricerche (già conteggiate), e associa i due risultati selezionati.

FTR: 2 (Docente, Corso) **DET:** <5 **Rating:** LOW x3

8. **Creazione Appello**

La funzionalità **Creazione Appello** modifica l'ILF, pertanto è conteggiabile come un **EI**.
Il docente inserisce le informazioni riferite all'appello, come la **data**, **data di scadenza prenotazioni**, **sede**, **note**, etc...

FTR: 1 (Appello) **DET:** <15 **Rating:** LOW x3

9. **Elenco Appelli**

La funzionalità **Elenco Appelli** non modifica l'ILF, pertanto è conteggiabile come un **EQ**.
Essa può essere effettuata per **docente**, **corso**, etc...

FTR: 1 (Appello) **DET:** <20 **Rating:** AVG x4

10. **Visualizzazione Elenco Prenotati Appello**

La funzionalità **Visualizzazione Elenco Prenotati Appello** non modifica l'ILF, pertanto è conteggiabile come un **EQ**.

Essa può essere effettuata per **docente**, **corso**, etc...

FTR: 1 (Appello) **DET:** <20 **Rating:** AVG x4

11. **Prenotazione Appello**

La funzionalità **Prenotazione Appello** modifica l'ILF, pertanto è conteggiabile come un **EI**.
Essa effettua una ricerca (già conteggiata) e registra la **matricola**, etc...

FTR: 1 (Appello) **DET:** <5 **Rating:** LOW x3

12. **Prelievo Dati Studente**

La funzionalità **Prelievo Dati Studente** modifica l'ILF, pertanto è conteggiabile come un **EI**.
Essa prende in ingresso e registra la **matricola**, il **nome**, il **cognome**, lo **username**, la **password** ed il **pin** etc...

FTR: 1 (Studente) **DET:** <15 **Rating:** LOW x3

13. Creazione Verbale

La funzionalità **Creazione Verbale** modifica l'ILF, pertanto è conteggiabile come un **EI**.
Essa prende in ingresso e registra la **matricola** dello studente, il **codice** del corso, il **voto** e gli **argomenti trattati** etc...

FTR: 1 (Verbale)

DET: <15

Rating: LOW x3

14. Aggiunta Studente al Verbale

La funzionalità **Aggiunta Studente al Verbale** modifica l'ILF, pertanto è conteggiabile come un **EI**.
Essa prende in ingresso e registra la **matricola** dello studente.

FTR: 1 (Verbale)

DET: <15

Rating: LOW x3

15. Cancellazione Studente dal Verbale

La funzionalità **Cancellazione Studente dal Verbale** modifica l'ILF, pertanto è conteggiabile come un **EI**.

Essa prende in ingresso e registra la **matricola** dello studente.

FTR: 1 (Verbale)

DET: <15

Rating: LOW x3

16. Chiusura Verbale

La funzionalità **Chiusura Verbale** modifica l'ILF, pertanto è conteggiabile come un **EI**.

FTR: 1 (Verbale)

DET: <15

Rating: LOW x3

17. Stampa Report

La funzionalità **Stampa Report** modifica l'ILF, pertanto è conteggiabile come un **EQ**.
Essa effettua una ricerca prendendo in ingresso **corso** e **anno accademico**.

FTR: 1 (Verbale)

DET: <15

Rating: LOW x3

CONTEGGIO UFP

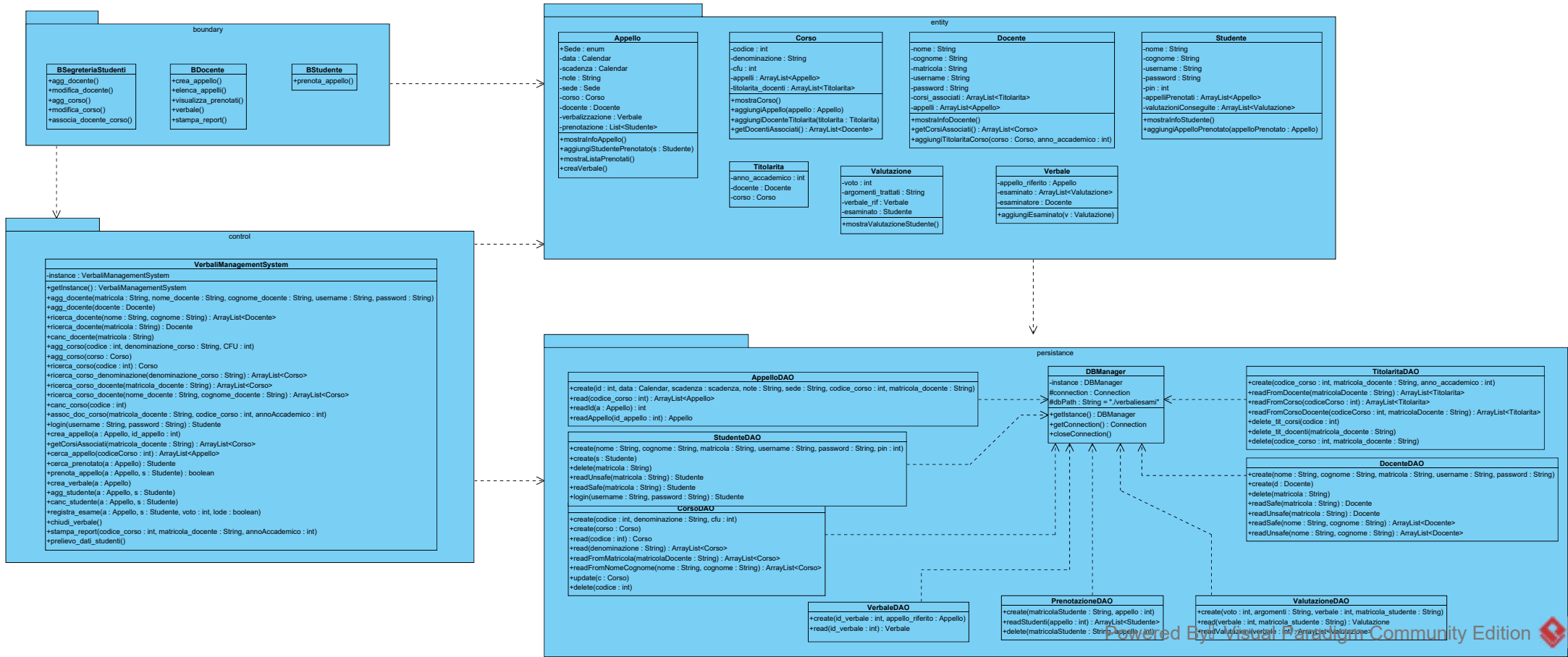
NILF = 1	10
NEIF = 1	5
NEI = 12 (tutti LOW)	36
NEO = 0	0
NEQ = 5 (3 LOW, 2 AVG)	17

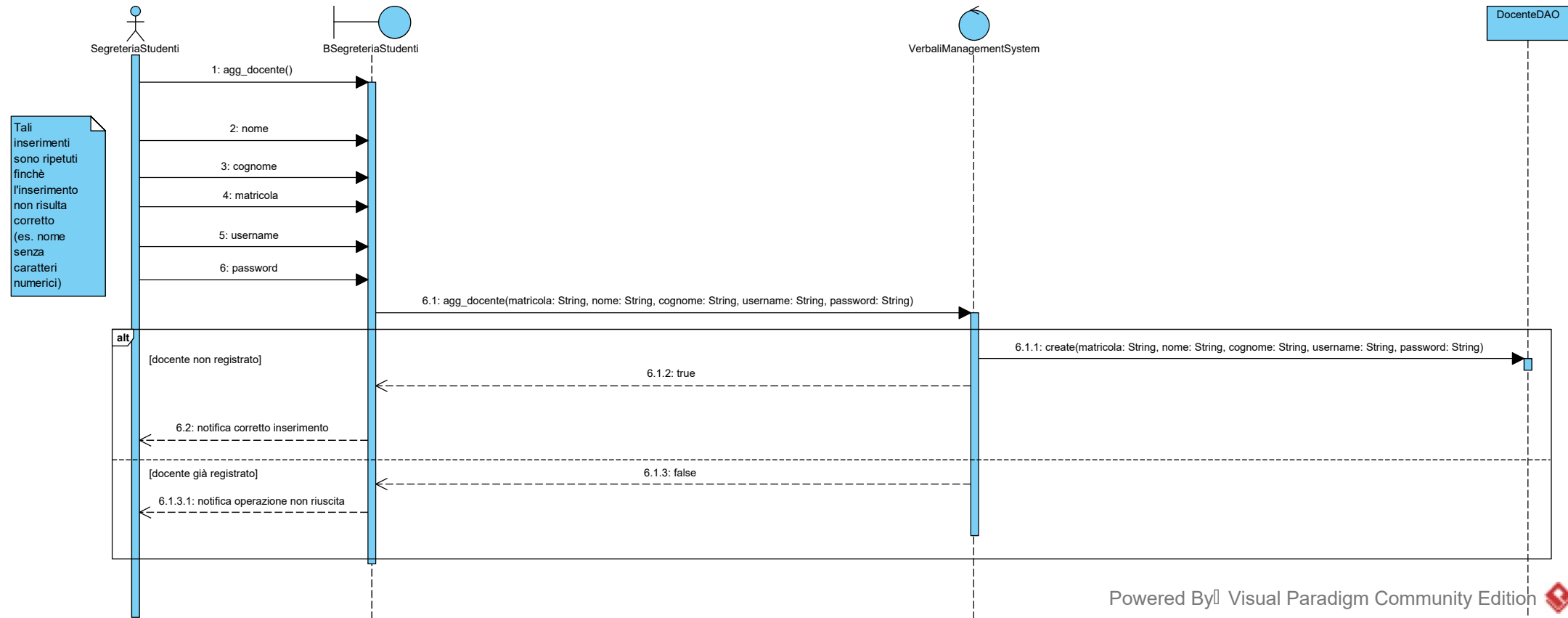
UFP = 68

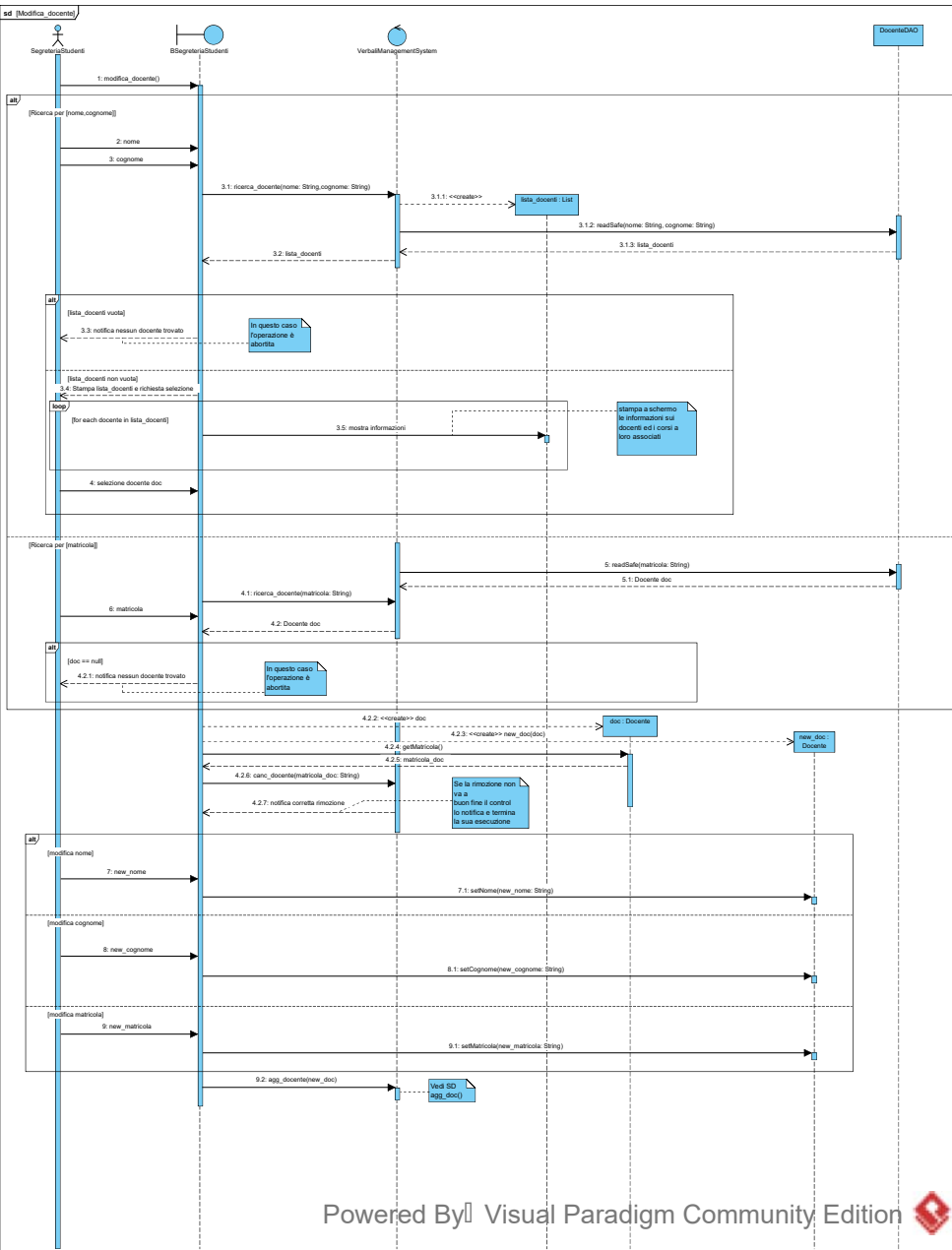
FATTORI DI AGGIUSTAMENTO

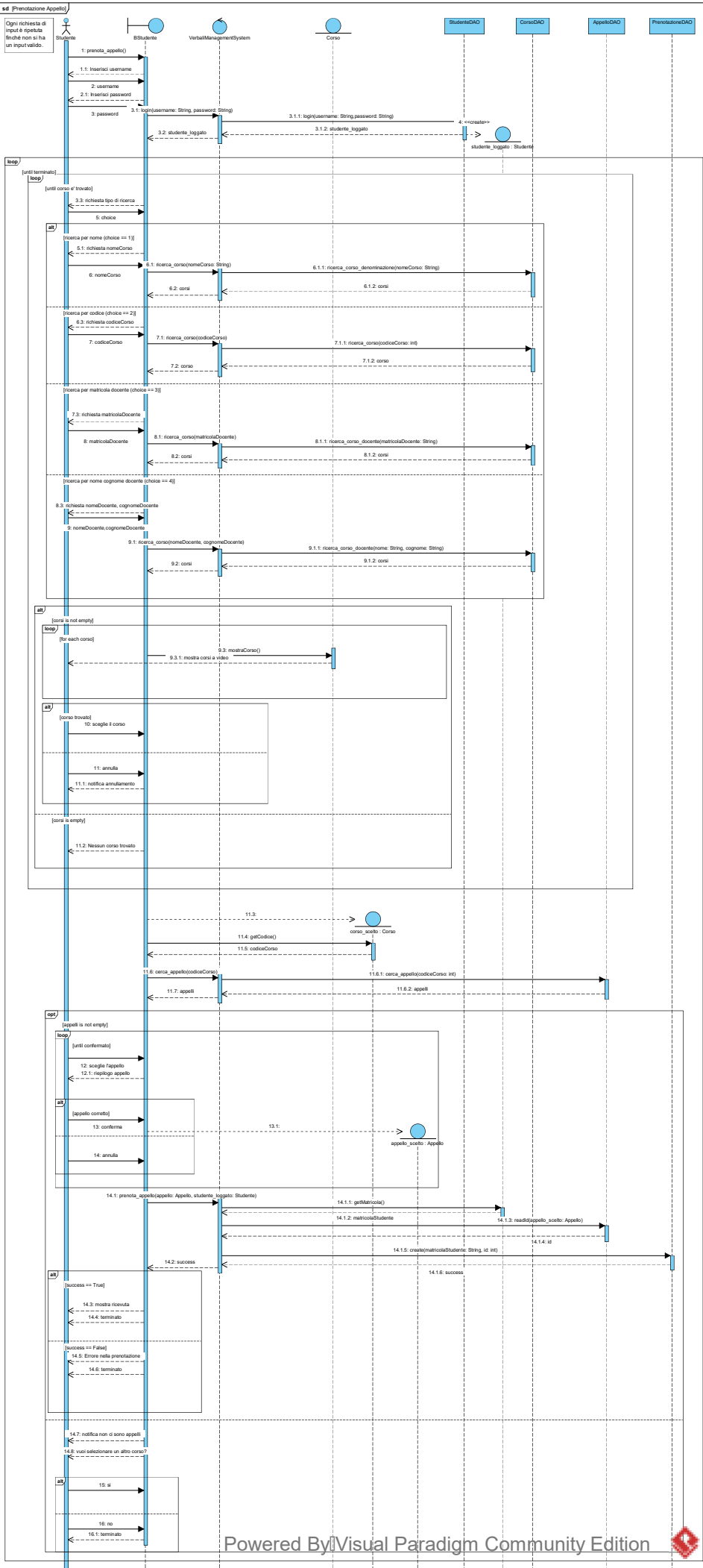
- Riusabilità 2

FP = 46









TEST FUNZIONALE

AGG_DOCENTE

CATEGORIE

- MATRICOLA identifica un docente univocamente; essa deve essere di 9 caratteri alfanumerici.
- NOME deve essere composto da sole lettere.
- COGNOME deve essere composto da sole lettere.
- USERNAME può essere qualsiasi, ma è univoco.
- PASSWORD può essere qualsiasi.

ID VALORI E VINCOLI

- MATRICOLA:
 - VALIDA (non presente nel DB).
 - INVALIDA (presente nel DB). [ERROR]
 - MALFORMATATA (!= 9 caratteri). [ERROR]
 - MALFORMATATA (caratteri non alfanumerici) [ERROR]
- NOME:
 - VALIDO.
 - MALFORMATO (caratteri non lettere). [ERROR]
 - MALFORMATO (>50 caratteri) [ERROR]
 - VUOTO. [SINGLE]
- COGNOME:
 - VALIDO.
 - MALFORMATO (caratteri non lettere). [ERROR]
 - MALFORMATO (>50 caratteri) [ERROR]
 - VUOTO. [SINGLE]
- USERNAME:
 - VALIDO (unico)
 - INVALIDO (già presente nel DB) [ERROR]
 - MALFORMATO (>50 caratteri) [ERROR]
 - VUOTO. [SINGLE]
- PASSWORD:
 - VALIDA.
 - VUOTA. [SINGLE]
 - MALFORMATATA (>50 caratteri) [ERROR]

Numero di casi di test = $(1*1*1*1*1)+3+3+3+3+2 = 15$

- testAgg_docenteDocente1(): valuta il funzionamento della funzione di aggiunta docente. Copre matricola valida, nome valido, cognome valido, username valido e password valida.
- testAgg_docenteDocente2(): valuta il comportamento della funzione aggiungi docente in condizioni di non validità. Copre matricola invalida
- testAgg_docenteDocente3(): valuta il comportamento della funzione aggiungi docente in condizioni di non validità.

- Copre matricola malformata (!= 9 caratteri)
- testAgg_docenteDocente4(): valuta il comportamento della funzione aggiungi docente in condizioni di non validità.
Copre matricola malformata caratteri non alfanumerici)
- testAgg_docenteDocente5(): valuta il comportamento della funzione aggiungi docente in condizioni di non validità.
Copre nome malformato (caratteri non lettere)
- testAgg_docenteDocente5(): valuta il comportamento della funzione aggiungi docente in condizioni di non validità.
Copre nome vuoto
- testAgg_docenteDocente7(): valuta il comportamento della funzione aggiungi docente in condizioni di non validità.
Copre cognome malformato (caratteri non lettere)
- testAgg_docenteDocente8(): valuta il comportamento della funzione aggiungi docente in condizioni di non validità.
Copre cognome vuoto
- testAgg_docenteDocente9(): il comportamento della funzione aggiungi docente in condizioni di non validità.
Copre username invalido (già presente nel DB)
- testAgg_docenteDocente10(): valuta il funzionamento della funzione aggiungi docente quando lo username risulta non compilato.
Copre username vuoto
- testAgg_docenteDocente11(): valuta il funzionamento della funzione aggiungi docente quando la password risulta non compilato.
Copre password vuota
- testAgg_docenteDocente12(): valuta il funzionamento della funzione aggiungi docente in condizioni di non validità.
Copre nome malformato (> 50 caratteri)
- testAgg_docenteDocente13(): valuta il funzionamento della funzione aggiungi docente in condizioni di non validità.
Copre cognome malformato (> 50 caratteri)
- testAgg_docenteDocente14(): valuta il funzionamento della funzione aggiungi docente in condizioni di non validità.
Copre username malformato (> 50 caratteri)
- testAgg_docenteDocente15(): valuta il funzionamento della funzione aggiungi docente in condizioni di non validità.
Copre password malformata (> 50 caratteri)

Tale testing ha portato ad evidenziare problemi nel controllo dei valori in ingresso: mancava un controllo ai valori della matricola, del nome o del cognome per oggetti creati esternamente all'inserimento da tastiera o il controllo dell'occorrenza o meno dello stesso username nella base di dati. Dopo tali FAIL, è stato necessario modificare il sistema al fine di correggere questi inconvenienti.

RICERCA_DOCENTE

La funzione di ricerca_docente è doppia: una prende in ingresso la matricola del docente e l'altra il nome e cognome.

CATEGORIE

- MATRICOLA identifica un docente univocamente; essa deve essere di 9 caratteri alfanumerici.
- NOME deve essere composto da sole lettere.
- COGNOME deve essere composto da sole lettere.

ID VALORI E VINCOLI

- MATRICOLA:
 - VALIDA (presente nel DB).
 - INVALIDA (non presente nel DB). [ERROR]
 - MALFORMATA (!= 9 caratteri). [ERROR]
 - MALFORMATA (caratteri non alfanumerici) [ERROR]

Numero di casi di test = 4

- testRicerca_docenteString1(): valuta il funzionamento della funzione di ricerca del docente. Copre matricola valida (presente nel DB).
- testRicerca_docenteString2(): valuta il comportamento della funzione di ricerca del docente in condizioni di non validità. Copre matricola invalida (non presente nel DB)
- testRicerca_docenteString2(): valuta il comportamento della funzione di ricerca del docente in condizioni di non validità. Copre matricola malformata (!= 9 caratteri)
- testRicerca_docenteString2(): valuta il comportamento della funzione di ricerca del docente in condizioni di non validità. Copre matricola malformata (caratteri non alfanumerici)

- NOME:
 - VALIDO.
 - MALFORMATO (caratteri non lettere). [ERROR]
 - MALFORMATO (<50 caratteri) [ERROR]
 - VUOTO. [SINGLE]
- COGNOME:
 - VALIDO.
 - MALFORMATO (caratteri non lettere). [ERROR]
 - MALFORMATO (<50 caratteri) [ERROR]
 - VUOTO. [SINGLE]

Numero di casi di test = $(1*1)+3+3 = 7$

- testRicerca_docenteStringString1(): valuta il funzionamento della funzione di ricerca del docente. Copre nome valido e cognome valido.
- testRicerca_docenteStringString2(): valuta il comportamento della funzione di ricerca del docente in condizioni di non validità. Copre nome vuoto
- testRicerca_docenteStringString3(): valuta il comportamento della funzione di ricerca del docente in condizioni di non validità.

- Copre nome malformato (caratteri non lettere)
- testRicerca_docenteStringString4(): valuta il comportamento della funzione di ricerca del docente in condizioni di non validità.
Copre cognome vuoto
- testRicerca_docenteStringString5(): valuta il comportamento della funzione di ricerca del docente in condizioni di non validità.
Copre cognome malformato (caratteri non lettere)
- testRicerca_docenteStringString6(): valuta il comportamento della funzione di ricerca del docente in condizioni di non validità.
Copre nome malformato (>50 caratteri)
- testRicerca_docenteStringString7(): valuta il comportamento della funzione di ricerca del docente in condizioni di non validità.
Copre cognome malformato (>50 caratteri)

Come per il testing precedente, anche in questo caso sono stati evidenziati problemi dovuti al non controllo dei valori, in questo caso necessari per la ricerca.

CANC_DOCENTE

CATEGORIE

- MATRICOLA identifica un docente univocamente; essa deve essere di 9 caratteri alfanumerici.

ID VALORI E VINCOLI

- MATRICOLA:
 - VALIDA (presente nel DB).
 - INVALIDA (non presente nel DB). [ERROR]
 - MALFORMATATA (!= 9 caratteri). [ERROR]
 - MALFORMATATA (caratteri non alfanumerici) [ERROR]

Numero di casi di test = 4

- testCanc_docente1(): valuta il funzionamento della funzione di cancellazione del docente.
Copre matricola valida (presente nel DB).
- testCanc_docente2(): valuta il comportamento della funzione di cancellazione del docente in condizioni di non validità.
Copre matricola invalida (non presente nel DB)
- testCanc_docente3(): valuta il comportamento della funzione di cancellazione del docente in condizioni di non validità.
Copre matricola malformata (!= 9 caratteri)
- testCanc_docente4(): valuta il comportamento della funzione di cancellazione del docente in condizioni di non validità.
Copre matricola malformata (caratteri non alfanumerici).

Anche in questo caso sono stati evidenziati problemi dovuti al non controllo dei valori, in questo caso necessari per la ricerca del docente da eliminare.

TEST FUNZIONALE

CREA_APPELLO

CATEGORIE:

- DATA ESAME : deve essere posteriore al 01/01/1225 e l'anno deve essere di 4 cifre
- DATA SCADENZA : deve essere posteriore al 01/01/1225 e precedente alla data d'esame e l'anno deve essere di 4 cifre
- NOTE : possono essere qualsiasi
- SEDE : può essere qualsiasi
- CODICE CORSO : deve identificare un corso esistente
- MATRICOLA DOCENTE : deve essere di 9 caratteri alfanumerici
- ID APPELLO : deve essere maggiore di zero

ID VALORI E VINCOLI:

- DATA ESAME:
 - VALIDA
 - NON VALIDA (precedente al 01/01/1225) [ERROR]
 - NON VALIDA (anno di più di 4 cifre) [ERROR]
- DATA SCADENZA
 - VALIDA
 - VALIDA MA DATA SCADENZA PRENOTAZIONE > DATA ESAME
 - NON VALIDA (precedente al 01/01/1225) [ERROR]
 - NON VALIDA (anno di più di 4 cifre) [ERROR]
- NOTE
 - VALIDO
- SEDE
 - VALIDA
- CODICE CORSO
 - VALIDO
 - NON VALIDO (non coincide con nessun corso presente nel DB) [ERROR]
- MATRICOLA DOCENTE
 - VALIDA
 - NON VALIDA (!= 9 caratteri) [ERROR]
 - NON VALIDA (non coincidente con nessun docente nel DB) [ERROR]
 - NON VALIDA (contenente caratteri speciali, ad es: ! @ ? * ecc) [ERROR]
- ID APPELLO
 - VALIDO
 - NON VALIDO (minore o uguale a zero) [ERROR]

Numero di casi di test da fare:

$$(1 \times 2 \times 1 \times 1 \times 1 \times 1) + (2 + 2 + 1 + 3 + 1) = 2 + 9 = 11$$

Crea_appello_1() : Valuta il comportamento del sistema quando viene chiamata la funzione crea_appello(Appello a, int id_appello) con tutti valori d'ingresso validi

Crea_appello_2() : Valuta il comportamento della funzione crea_appello((Appello a, int id_appello) quando in ingresso è passato un appello con data di esame precedente al 01/01/1225.
Copre DATA ESAME NON VALIDA

Crea_appello_3() : Valuta il comportamento della funzione crea_appello((Appello a, int id_appello) quando in ingresso è passata un appello con data di scadenza prenotazione precedente al 01/01/1225.
Copre DATA SCADENZA NON VALIDA

Crea_appello_4() : Valuta il comportamento della funzione crea_appello((Appello a, int id_appello) quando in ingresso è passato un appello con data di scadenza prenotazione successiva alla data di esame.
Copre DATA SCADENZA VALIDA MA > DATA ESAME

Crea_appello_5() : Valuta il comportamento della funzione crea_appello((Appello a, int id_appello) quando in ingresso è passato un appello con un codice corso invalido, cioè non coincidente con nessun corso che ha quel codice nel DB.
Copre CODICE CORSO NON VALIDO

Crea_appello_6() : Valuta il comportamento della funzione crea_appello((Appello a, int id_appello) quando in ingresso è passato un appello con una matricola docente invalida, cioè con un numero di caratteri diverso da 9.
Copre MATRICOLA NON VALIDA

Crea_appello_7() : Valuta il comportamento della funzione crea_appello((Appello a, int id_appello) quando in ingresso è passato un appello con una matricola docente invalida, cioè non coincidente con nessuna matricola di docente già presente nel DB.
Copre MATRICOLA NON VALIDA

Crea_appello_8() : Valuta il comportamento della funzione crea_appello((Appello a, int id_appello) quando in ingresso è passato un appello con una matricola docente invalida, cioè con caratteri speciali (@ ! ...)
Copre MATRICOLA NON VALIDA

Crea_appello_9() : Valuta il comportamento della funzione crea_appello((Appello a, int id_appello) quando in ingresso è passato un id_appello con valore minore o uguale a zero
Copre ID APPELLO NON VALIDO

Crea_appello_10() : Valuta il comportamento della funzione crea_appello((Appello a, int id_appello) quando in ingresso è passata un appello con data che presenta un anno con numero di cifre maggiore di 4 (maggiore al 31/12/9999).
Copre DATA ESAME NON VALIDA

Crea_appello_11() : Valuta il comportamento della funzione crea_appello((Appello a, int id_appello) quando in ingresso è passata un appello con data di scadenza che presenta un numero di cifre maggiore di 4 (maggiore al 31/12/9999).

Copre DATA ESAME NON VALIDA

Questo testing è servito ad evidenziare gli errori commessi nel non controllare valori di ingresso: erano stati dimenticati controlli su data e data di scadenza, controlli sulla matricola docente, id del corso non coincidente con nessun corso nel DB e id dell'appello. Tali controlli venivano effettuati solamente quando venivano richiesti tramite inserimento da tastiera.

Dopo aver notato tali errori, sono state apportate modifiche al sistema per correggerli.

NOTA: avendo utilizzato la classe Calendar per la gestione delle date, quando si prova a creare una data, viene automaticamente gestito il fatto che alcuni mesi hanno meno di 31 giorni (ad esempio se si prova ad inserire [anno/mese/giorno] 2022/02/29, la data settata sarà il 2022/03/01), cioè sia i giorni sia i mesi vengono automaticamente shiftati in avanti o all'indietro quando non sono validi. (Allo stesso modo il mese -1 viene visto come l'ultimo mese dell'anno precedente, oppure il giorno 32 è visto come il giorno 1 del mese successivo e così via).

In questo modo, qualsiasi data in ingresso risulta sempre valida, quindi non è stato fatto testing su mese e giorno nelle date.

(Per Calendar i mesi vanno da 0 a 11).

TEST FUNZIONALE

PRENOTA_APPELLO

CATEGORIE

- Appello identifica un oggetto appello, esso è costituito da una data, una scadenza, note, una sede, un codice corso, una matricola docente.
- Studente identifica un oggetto studente, esso è costituito da un nome, cognome, matricola, username, password, pin.

ID VALORI E VINCOLI

- APPELLO:
 - VALIDO (nel DB).
 - VALIDO (solo data, codice corso e matricola docente coincidono). [SINGLE]
 - INVALIDO (non nel DB). [ERROR]
 - INVALIDO (già prenotato) [if STUDENTE]
- STUDENTE:
 - VALIDO (nel DB) [property STUDENTE]
 - VALIDO (solo matricola coincide). [SINGLE]
 - INVALIDO (non nel DB). [ERROR]

Numero di casi di test = $(1*1) + 4 + 1 = 6$

- test1(): valuta il funzionamento della funzione prenota_appello.
Copre appello valido, studente valido, prenotazione non effettuata.
- test2(): valuta il comportamento della funzione prenota_appello.
Copre appello valido solo per data, codiceCorso, matricolaDocente.
- test3(): valuta il comportamento della funzione prenota_appello.
Copre studente valido solo per matricola.
- test4(): valuta il comportamento della funzione prenota_appello in condizioni di non validità.
Copre appello invalido.
- test5(): valuta il comportamento della funzione prenota_appello in condizioni di non validità.
Copre studente invalido.
- test6(): valuta il comportamento della funzione prenota_appello in condizioni di non validità.
Copre prenotazione già effettuata.

Il test4 ha evidenziato un problema nella gestione di null pointers fornendo un appello invalido. È stato possibile risolvere gestendo l'eccezione all'interno di prenota_appello.

RICERCA_CORSO

La funzione di ricerca_corso è quadrupla:

- ricerca per codice corso
- ricerca per matricola docente
- ricerca per nome corso
- ricerca per nome e cognome docente.

CATEGORIE

- MATRICOLA identifica un docente univocamente; essa deve essere di 9 caratteri alfanumerici.
- NOME CORSO identifica il nome del corso, caratteri alfabetici e spazi
- NOME DOCENTE identifica il nome del docente, caratteri alfabetici e spazi
- COGNOME DOCENTE identifica il cognome del docente, caratteri alfabetici e spazi
- CODICE CORSO numero che identifica univocamente un corso

ID VALORI E VINCOLI

- MATRICOLA:
 - VALIDA (presente nel DB).
 - VALIDA (non ci sono corsi associati). [SINGLE]
 - MALFORMATA [ERROR]
 - NON PRESENTE (non presente nel DB) [SINGLE]

Numero di casi di test = 4

- test_matricolaDocente1(): valuta il funzionamento della funzione di ricerca con matricola docente. Copre matricola valida (presente nel DB).
- test_matricolaDocente2(): valuta il funzionamento della funzione di ricerca con matricola docente. Copre matricola valida (non ci sono corsi associati)
- test_matricolaDocente3(): valuta il comportamento della funzione di ricerca con matricola docente malformata.
- Copre matricola malformata
- test_matricolaDocente4(): valuta il funzionamento della funzione di ricerca con matricola docente.
- Copre matricola non presente nel DB.

Tramite questo test è stato rilevato un problema nel check del pattern della matricola.

- NOME DOCENTE:
 - VALIDO. [property NOME]
 - MALFORMATO [ERROR]
 - NON PRESENTE [SINGLE]
- COGNOME DOCENTE:
 - VALIDO.
 - MALFORMATO [ERROR]
 - NON PRESENTE [SINGLE]
 - PRESENTE MA NON VALIDO CON NOME [if NOME]

Numero di casi di test = $(1 \cdot 1) + 4 + 1 = 6$

- test_nomeCognomeDocente1(): valuta il funzionamento della funzione di ricerca. Copre nome valido e cognome valido.
- test_nomeCognomeDocente2(): valuta il comportamento della funzione di ricerca in condizioni di non validità. Copre cognome non presente

- test_nomeCognomeDocente3(): valuta il comportamento della funzione di ricerca in condizioni di non validità.
Copre nome non presente
- test_nomeCognomeDocente4(): valuta il comportamento della funzione di ricerca in condizioni di non validità.
Copre cognome presente ma non valido con nome
- test_nomeCognomeDocente5(): valuta il comportamento della funzione di ricerca in condizioni di non validità.
Copre nome malformato
- test_nomeCognomeDocente6(): valuta il comportamento della funzione di ricerca in condizioni di non validità.
Copre cognome malformato

Non sono stati rilevati problemi con questo test.

- CODICECORSO:
 - VALIDO.
 - NON PRESENTE [SINGLE]

Numero casi di test: 2

- test_codiceCorso1 (): valuta il funzionamento della funzione di ricerca.
Copre codice valido
- test_codiceCorso2 (): valuta il funzionamento della funzione di ricerca.
Copre non presente

Non sono stati rilevati problemi con questo test.

- NOME CORSO:
 - VALIDO.
 - MALFORMATO [ERROR]
 - NON PRESENTE [SINGLE]

Numero casi di test: 3

- test_nomeCorso1 (): valuta il funzionamento della funzione di ricerca.
Copre nome valido
- test_codiceCorso2 (): valuta il funzionamento della funzione di ricerca.
Copre non presente
- test_codiceCorso3 (): valuta il funzionamento della funzione di ricerca.
Copre nome malformato

Non sono stati rilevati problemi con questo test.