

CORSO DI LAUREA IN INGEGNERIA INFORMATICA CORSO DI INGEGNERIA DEL SOFTWARE PROF. R. PIETRANTUONO- A.A. 2023 – 24

Progetto

GESTIONE LEZIONI ASSOCIAZIONE MUSICALE



GESTIONE LEZIONI ASSOCIAZIONE MUSICALE

Team

SEA²



Studenti:

Antonio Pio Montagna Salvatore Raiola Emmanuel Russo Antonio Varese

Indice

1 - Specifiche informali	4
2 - Analisi e specifica dei requisiti	5
2.1 - Analisi nomi - verbi	
2.2 – Revisione dei requisiti	
2.3 - Glossario dei termini	
2.4 – Classificazione dei requisiti	9
2.4.1 – Requisiti funzionali	
2.4.2 – Requisiti sui dati	10
2.4.3 – Vincoli / Altri requisiti	10
2.5 - Modellazione dei casi d'uso	11
2.5.1 – Attori e casi d'uso	11
2.5.2 – Diagramma dei casi d'uso	13
2.5.3 – Scenari	
2.6 - Diagramma delle classi	18
2.7 - Diagrammi di sequenza	20
3 – Stima dei costi	24
4 – Piano di test funzionale	27
5 - Progettazione	46
5.1 - Diagramma delle classi	46
5.2 - Diagrammi di sequenza	47
6 - Implementazione	51
7 - Testing	54
7.1 - Test funzionale	

1 - Specifiche informali

Si vuole realizzare una applicazione web per la gestione delle prenotazioni delle lezioni in una scuola serale di musica.

L'associazione che gestisce la scuola promuove l'apprendimento della musica a qualunque età, attraverso lezioni da parte di maestri, che offrono la propria disponibilità per due giorni a settimana per lezioni a prezzi modici.

Le lezioni si tengono nella scuola in tutti i giorni della settimana e sono individuali, tutte della durata di 1 h, tra le 17 e le 21. Le lezioni sono di due livelli – principianti e intermedio. Il costo di ogni lezione dipende dal livello (non dal maestro né dallo strumento).

La segreteria della scuola registra le informazioni relative agli insegnanti: nome, il cognome, una foto, gli strumenti (es.: chitarra classica, chitarra jazz, mandolino, pianoforte, etc.), e-mail, numero di telefono. I maestri si collegano al sistema per inserire/modificare i giorni di disponibilità, e per consultare le lezioni svolte mese per mese.

Tutti i maestri sono capaci di impartire lezioni sia a principianti, sia di livello intermedio.

Il sistema offre agli aspiranti musicisti la possibilità di cercare i maestri in base allo strumento, di visualizzare per ciascuno di essi i giorni di disponibilità, e di prenotare una o più lezioni.

Gli studenti registrati possono prenotare le lezioni online, effettuando il login tramite username e password. Per la registrazione, lo studente deve fornire alla segreteria cognome, nome, data di nascita, numero di telefono, e-mail. Quando la segreteria della scuola inserisce uno studente, il sistema genera le credenziali, che vengono inviate per e-mail allo studente.

Per effettuare una prenotazione, lo studente sceglie il maestro, la data ed un orario disponibile. Il sistema automaticamente rimuoverà la disponibilità per la fascia oraria prenotata.

La segreteria può visualizzare in qualunque momento le prenotazioni, sia quelle svolte, sia quelle dei giorni a venire.

Se necessario, un discente deve poter disdire una lezione prenotata: in caso di disdetta, il sistema invia una notifica per e-mail alla segreteria. Se la disdetta avviene con almeno 3 giorni d'anticipo, la lezione non deve essere addebitata al discente.

Gli studenti registrati possono visualizzare le date delle lezioni che hanno svolto nel mese, sì da poter effettuare a fine mese il bonifico bancario alla scuola. L'importo da bonificare viene calcolato dal sistema.

Dopo aver consultato i bonifici ricevuti, la segreteria può consultare il sistema per registrare i discenti in regola con i pagamenti. Ogni prima settimana del mese il sistema genera un elenco delle ore di lezione svolte da ogni docente nel mese precedente, in modo che la segreteria possa effettuare i bonifici ai maestri per le lezioni effettivamente svolte, dopo aver decurtato una percentuale (impostabile una tantum dalla segreteria) per la gestione della scuola.

2 - Analisi e specifica dei requisiti

2.1 - Analisi nomi - verbi

Si vuole realizzare una applicazione web per la gestione delle prenotazioni delle lezioni in una scuola serale di musica.

L'associazione che gestisce la scuola promuove l'apprendimento della musica a qualunque età, attraverso lezioni da parte di maestri, che offrono la propria disponibilità per due giorni a settimana per lezioni a prezzi modici.

Le lezioni si tengono nella scuola in tutti i giorni della settimana e sono individuali, tutte della durata di 1 h, tra le 17 e le 21.

Le lezioni sono di due livelli – principianti e intermedio.

Il costo di ogni lezione dipende dal livello (non dal maestro né dallo strumento).

La <mark>segreteria</mark> della scuola <mark>registra le informazioni relative agli insegnanti: nome, il cognome, una foto, gli strumenti (es.: chitarra classica, chitarra jazz, mandolino, pianoforte, etc.), e-mail, numero di telefono.</mark>

I <mark>maestri</mark> si collegano al sistema per <mark>inserire/modificare i giorni di disponibilità,</mark> e per consultare le lezioni svolte mese per mese.

Tutti i maestri sono capaci di impartire lezioni sia a principianti, sia di livello intermedio.

Il sistema offre agli aspiranti musicisti la possibilità di cercare i maestri in base allo strumento, di visualizzare per ciascuno di essi i giorni di disponibilità, e di prenotare una o più lezioni.

Gli studenti registrati possono prenotare le lezioni online, effettuando il login tramite username e password.

Per la registrazione, lo <mark>studente</mark> deve fornire alla segreteria <mark>cognome</mark>, <mark>nome</mark>, data di nascita, numero di telefono, e-mail.

Quando la segreteria della scuola inserisce uno studente, il sistema genera le credenziali, che vengono inviate per e-mail allo studente.

Per effettuare una prenotazione, lo studente sceglie il maestro, la data ed un orario disponibile.

Il sistema automaticamente rimuoverà la disponibilità per la fascia oraria prenotata.

La segreteria può visualizzare in qualunque momento le prenotazioni, sia quelle svolte, sia quelle dei giorni a venire.

Se necessario, <mark>un discente deve poter disdire una lezione prenotata</mark>: in caso di disdetta, <mark>il sistema invia una notifica per e-mail</mark> alla segreteria.

Se la disdetta avviene con almeno tre giorni d'anticipo, la lezione non deve essere addebitata al discente.

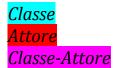
Gli studenti registrati possono <mark>visualizzare le date delle lezioni che hanno svolto nel mese</mark>, sì da poter effettuare a fine mese il bonifico bancario alla scuola.

L'importo da bonificare viene calcolato dal sistema.

Dopo aver consultato i bonifici ricevuti, <mark>la segreteria può consultare il sistema per registrare i discenti in regola con i pagamenti</mark>.

Ogni prima settimana del mese il sistema genera un elenco delle ore di lezione svolte da ogni docente nel mese precedente, in modo che la segreteria possa effettuare i bonifici ai maestri per le lezioni effettivamente svolte, dopo aver decurtato una percentuale (impostabile una tantum dalla segreteria) per la gestione della scuola.

LEGENDA:



<mark>Funzionalità</mark> Attributo

2.2 - Revisione dei requisiti

- 1. Le lezioni si tengono nella scuola in tutti i giorni della settimana.
- 2. Le lezioni sono individuali.
- 3. Le lezioni sono tutte della durata di 1h.
- 4. Le lezioni si svolgono tra le ore 17 e le ore 21.
- 5. Le lezioni sono di due livelli "principianti" e "intermedio".
- 6. Le lezioni hanno un costo.
- 7. Il costo di ogni lezione dipende dal livello, non dal maestro né dallo strumento.
- 8. Ogni maestro deve offrire la propria disponibilità per due giorni alla settimana.
- 9. Ogni maestro può impartire lezioni sia a principianti, sia di livello intermedio.
- Il sistema deve offrire alla segreteria una funzionalità per registrare le informazioni relative ad un maestro.
- 11. Di ogni maestro si vuole memorizzare nome, cognome, una foto, uno o più strumenti, e-mail e numero di telefono.
- 12. Il sistema deve offrire al maestro una funzionalità per inserire i propri giorni di disponibilità.
- 13. Il sistema deve offrire al maestro una funzionalità per modificare i propri giorni di disponibilità.
- 14. Il sistema deve offrire al maestro una funzionalità per consultare le proprie lezioni svolte mese per mese.
- 15. Il sistema deve offrire allo studente una funzionalità per cercare i maestri in base allo strumento.
- 16. Il sistema deve offrire allo studente una funzionalità per visualizzare i giorni di disponibilità di un maestro, per ciascuno di essi.
- 17. Il sistema deve offrire alla segreteria una funzionalità per registrare le informazioni relative ad uno studente.
- 18. Per gli studenti registrati si vuole memorizzare all'interno del sistema: nome, cognome, data di nascita, e-mail e numero di telefono.
- 19. Dopo una registrazione, il sistema deve generare le credenziali ed inviarle allo studente tramite e-mail.
- 20. Dopo la registrazione, uno studente registrato potrà effettuare il login nel sistema.
- 21. Uno studente registrato può effettuare tutte le operazioni di uno studente non registrato.
- 22. Il sistema deve offrire allo studente registrato, dopo il login, una funzionalità per prenotare una o più lezioni.
- 23. Per effettuare una prenotazione, lo studente deve specificare il maestro, la data ed un orario disponibile.
- 24. Il sistema deve offrire alla segreteria una funzionalità per visualizzare le prenotazioni, sia quelle svolte che per i giorni a venire.

- 25. Il sistema deve offrire allo studente registrato, dopo il login, una funzionalità per disdire una lezione prenotata.
- 26. Dopo una disdetta, il sistema deve inviare una notifica di disdetta alla segreteria tramite e-mail.
- 27. Il sistema deve offrire allo studente registrato, dopo il login, una funzionalità per visualizzare le date delle lezioni che ha svolto nel mese.
- 28. Ogni mese, il sistema deve calcolare automaticamente l'importo che lo studente deve bonificare alla scuola.
- 29. Il sistema deve offrire alla segreteria una funzionalità per registrare gli studenti in regola con i pagamenti.
- 30. Ogni prima settimana del mese, il sistema deve generare un elenco delle ore di lezioni svolte da ogni docente nel mese precedente.
- 31. Il sistema deve offrire alla segreteria una funzionalità per impostare la percentuale di decurtazione.

2.3 - Glossario dei termini

Termine	Descrizione	Sinonimi
Lezione	Incontro didattico tra un docente ed uno	
Lezione	studente.	
Studente	Studente Persona che desidera ricevere lezioni musicali.	
Studente registrato	Studente che ha effettuato la registrazione.	
Maestro	Maestro Persona responsabile dell'insegnamento.	
Segreteria	Ente responsabile dell'amministrazione della scuola.	
Prenotazione Procedura per riservare in anticipo una lezione.		

2.4 - Classificazione dei requisiti

2.4.1 - Requisiti funzionali

ID	Requisito	Origine (n. frase
		dei requisiti
		revisionati)
RF01	Il sistema deve offrire alla segreteria una funzionalità per registrare le informazioni relative ad un maestro.	10
RF02	Il sistema deve offrire al maestro una funzionalità per inserire i	12
102	propri giorni di disponibilità.	12
RF03	Il sistema deve offrire al maestro una funzionalità per modificare	13
	i propri giorni di disponibilità.	
RF04	Il sistema deve offrire al maestro una funzionalità per	14
	consultare le proprie lezioni svolte mese per mese.	
RF05	Il sistema deve offrire allo studente una funzionalità per cercare i maestri in base allo strumento.	15
RFo6	Il sistema deve offrire allo studente una funzionalità per	16
100	visualizzare i giorni di disponibilità di un maestro, per ciascuno	10
	di essi.	
RF07	Il sistema deve offrire alla segreteria una funzionalità per	17
,	registrare le informazioni relative ad uno studente.	,
RFo8	Dopo una registrazione, il sistema deve generare le credenziali ed	19
	inviarle allo studente tramite e-mail.	-
RF09	Dopo la registrazione, uno studente registrato potrà effettuare il	20
	login nel sistema.	
RF10	Uno studente registrato può effettuare tutte le operazioni di uno	21
	studente non registrato.	
RF11	Il sistema deve offrire allo studente registrato, dopo il login, una	22
	funzionalità per prenotare una o più lezioni.	
RF12	Il sistema deve offrire alla segreteria una funzionalità per	24
	visualizzare le prenotazioni, sia quelle svolte che per i giorni a	
DE	venire.	
RF13	Il sistema deve offrire allo studente registrato, dopo il login, una	25
DE.	funzionalità per disdire una lezione prenotata.	- (
RF14	Dopo una disdetta, il sistema deve inviare una notifica di disdetta alla segreteria tramite e-mail.	26
RF15	Il sistema deve offrire allo studente registrato, dopo il login, una	27
KI 15	funzionalità per visualizzare le date delle lezioni che ha svolto	27
	nel mese.	
RF16	Ogni mese, il sistema deve calcolare automaticamente l'importo	28
	che lo studente deve bonificare alla scuola.	
RF17	Il sistema deve offrire alla segreteria una funzionalità per	29
	registrare gli studenti in regola con i pagamenti.	
RF18	Ogni prima settimana del mese, il sistema deve generare un	30
	elenco delle ore di lezioni svolte da ogni docente nel mese	_
	precedente.	

RF19	Il sistema deve offrire alla segreteria una funzionalità per	31
	impostare la percentuale di decurtazione.	

2.4.2 - Requisiti sui dati

ID	Requisito	Origine frase	(n.
			dei
		requisiti	
		revisionati)	
RDo1	Le lezioni si tengono in tutti i giorni della settimana.	1	
RDo2	Le lezioni sono individuali.	2	
RDo3	Le lezioni sono tutte della durata di 1h.	3	
RD04	Le lezioni si svolgono tra le ore 17 e le ore 21.	4	
RDo ₅	Le lezioni sono di due livelli – "principianti" e "intermedio".	5	
RDo6	Le lezioni hanno un costo.	6	
RDo7	Il costo di ogni lezione dipende dal livello, non dal maestro né		
	dallo strumento.		
RDo8	Ogni maestro deve offrire la propria disponibilità per due giorni	8	
	alla settimana.		
RD09	Ogni maestro può impartire lezioni sia a principianti, sia di livello	9	
	intermedio.		
RD10	Di ogni maestro si vuole memorizzare nome, cognome, una foto,	11	
	uno o più strumenti, e-mail e numero di telefono.		
RD11	Per gli studenti registrati si vuole memorizzare all'interno del	18	
	sistema: nome, cognome, data di nascita, e-mail e numero di		
	telefono.		
RD12	Per effettuare una prenotazione, lo studente deve specificare il	23	
	maestro, la data ed un orario disponibile.		

2.4.3 - Vincoli / Altri requisiti

RNFo1 - Per l'invio delle credenziali deve essere disponibile un server di posta elettronica esterno al sistema.

RNF02 - Per l'invio della notifica di avvenuta disdetta deve essere disponibile un server di posta elettronica esterno al sistema.

2.5 - Modellazione dei casi d'uso

2.5.1 - Attori e casi d'uso

Attori Primari:

- Studente
- StudenteRegistrato
- Maestro
- Segreteria
- Tempo

Attori Secondari:

ServizioMail

Casi d'uso:

- UC1: CercaMaestro
- UC2: VisualizzaDisponibilità
- UC3: PrenotaLezione
- UC4: DisdiciLezione
- UC5: VisualizzaDateLezioniSvolte
- UC6: InserisciGiorno
- UC7: ModificaGiorno
- UC8: ConsultaLezioniSvolte
- UC9: InserisciDatiMaestro
- UC10: InserisciDatiStudente
- UC11: VisualizzaPrenotazioni
- UC12: Imposta%Decurtazione
- UC13: CalcolaImportoMaestro
- UC14: RegistraStudenteInRegola
- UC15: CalcolaImportoStudente
- UC16: GeneraElencoLezioniSvolte

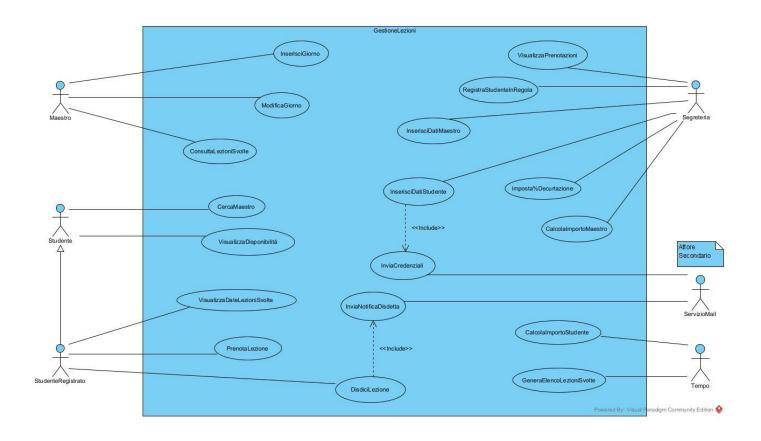
Casi d'uso d'inclusione:

- UC17: InviaCredenziali
- UC18: InviaNotificaDisdetta

Elaborato di Ingegneria del Software - T18

Caso d'uso	Attori Primari	Attori Secondari	Incl. / Ext.	Requisiti corrispondenti
UC1: CercaMaestro	Studente	-	-	RF05
UC2: VisualizzaDisponibilità	Studente	-	-	RFo6
UC3: PrenotaLezione	StudenteRegistrato	-	-	RF11
UC4: DisdiciLezione	StudenteRegistrato	-	Include InviaNotificaDisdetta	RF13
UC5: VisualizzaDateLezioniSvolte	StudenteRegistrato	-	-	RF15
UC6: InserisciGiorno	Maestro			RF02
UC7: ModificaGiorno	Maestro	-	-	RF03
UC8: ConsultaLezioniSvolte	Maestro	-	-	RFo4
UC9: InserisciDatiMaestro	Segreteria	-	-	RF01
UC10: InserisciDatiStudente	Segreteria	-	Include InviaCredenziali	RF07
UC11: VisualizzaPrenotazioni	Segreteria	-	-	RF12
UC12: Imposta%Decurtazione	Segreteria	-	-	RF19
UC13: CalcolaImportoMaestro	Segreteria	-	-	-
UC14: RegistraStudenteInRegola	Segreteria	-	-	RF17
UC15: CalcolaImportoStudente	Tempo	-	-	RF16
UC16: GeneraElencoLezioniSvolte	Tempo	-	-	RF18
UC17: InviaCredenziali	Segreteria	ServizioMail	Incluso in InserisciDatiStudente	RFo8
UC18: InviaNotificaDisdetta	StudenteRegistrato	ServizioMail	Incluso in DisdiciLezione	RF14

2.5.2 - Diagramma dei casi d'uso



2.5.3 - Scenari

- UC3: PrenotaLezione

Caso d'uso:	PrenotaLezione		
Attore primario	StudenteRegistrato		
Attore secondario	-		
Descrizione	Uno Studente Registrato, a seguito del login, prenota una o più lezioni per maestro in base allo strumento, in una data, ad un orario e ad un livello di difficoltà.		
Pre-Condizioni	Lo studente deve aver effettuato l'accesso e devono esistere una o più lezioni disponibili per la prenotazione.		
Sequenza di eventi principale	 disponibili per la prenotazione. Il caso d'uso inizia quando lo Studente Registrato richiede la prenotazione della\e lezione\i. 1.1 Lo Studente Registrato inserisce lo strumento, il maestro, la data, l'orario ed il livello. Il sistema controlla che la lezione sia disponibile per lo strumento, il maestro, la data, l'ora selezionati. 2.1 Se la lezione non è disponibile il sistema restituisce un ERRORE allo Studente Registrato. Il sistema seleziona la\e lezione\i prenotata\e e rimuoverà la\e fascia\e oraria prenotata\e come disponibili. 		
Post-Condizioni	Lo studente ha prenotato una o più lezioni.		
Casi d'uso correlati	-		
Sequenza di eventi alternativi	-		

- UC4: DisdiciLezione

Caso d'uso:	Disdici Lezione		
Attore primario	StudenteRegistrato		
Attore secondario	ServizioMail		
Descrizione	Uno studente disdice una lezione prenotata.		
	Una volta disdetta la lezione, il sistema invia una notifica via e-mail alla segreteria		
	per informarla della disdetta.		
Pre-Condizioni	Lo studente ha prenotato almeno una lezione.		
Sequenza di eventi	1. Lo studente effettua il login.		
principale	2. Lo studente visualizza la lista delle lezioni prenotate.		
	3. Lo studente individua la lezione che desidera disdire.		
	4. Lo studente seleziona l'opzione per disdire.		
	5. Lo studente conferma la disdetta.		
	6. Il sistema non addebita il costo se la disdetta è avvenuta con tre giorni di anticipo.		
	7. Il sistema registra la disdetta e rimuove la prenotazione dal sistema.		
	8. Il sistema invia una notifica via e-mail alla segreteria, informandola della		
	disdetta della lezione da parte dello studente.		
Post-Condizioni	La lezione prenotata è stata disdetta correttamente.		
Casi d'uso correlati	InviaNotificaDisdetta		
Sequenza di eventi	- Se lo studente cambia idea prima di confermare la disdetta,		
alternativi	può annullare l'operazione e tornare alla visualizzazione		
	delle prenotazioni.		

- UC6: InserisciGiorno

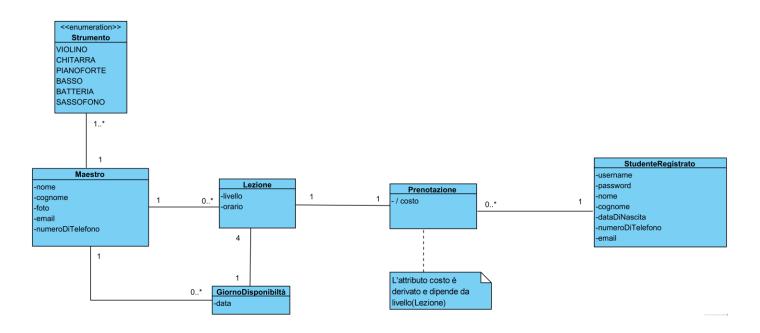
Caso d'uso:	InserisciGiorno			
Attore primario	Maestro			
Attore secondario	-			
Descrizione	In ogni momento il Maestro può, attraverso il sistema, inserire giorni di disponibilità in cui potrà tenere lezioni.			
Pre-Condizioni	La Data inserita non deve coincidere né con quella in cui viene effettuata l'operazione né con quella di un altro giorno di disponibilità già presente nel sistema.			
Sequenza di eventi principale	 Il Maestro inserisce la data del giorno di disponibilità. Il sistema verifica la validità dell'INPUT (pre-condition). Il sistema crea un nuovo Giorno di disponibilità. Il sistema chiede la conferma per l'inserimento. Il maestro CONFERMA l'operazione. Il nuovo Giorno di disponibilità viene inserito nella lista del sistema. Il sistema chiede se si vuole continuare ad inserire. 			
Post-Condizioni	Il Giorno di disponibilità indicato dal maestro verrà inserito nel sistema.			
Casi d'uso correlati				
Sequenza di eventi alternativi	 Il Maestro inserisce la data del giorno di disponibilità. Il sistema verifica la validità dell'INPUT (pre-condition). Il sistema crea un nuovo Giorno di disponibilità. Il sistema chiede la conferma per l'inserimento. Il maestro NON CONFERMA l'operazione. Il nuovo Giorno di disponibilità viene distrutto. Il sistema chiede se si vuole continuare ad inserire. 			

- UC7: ModificaGiorno

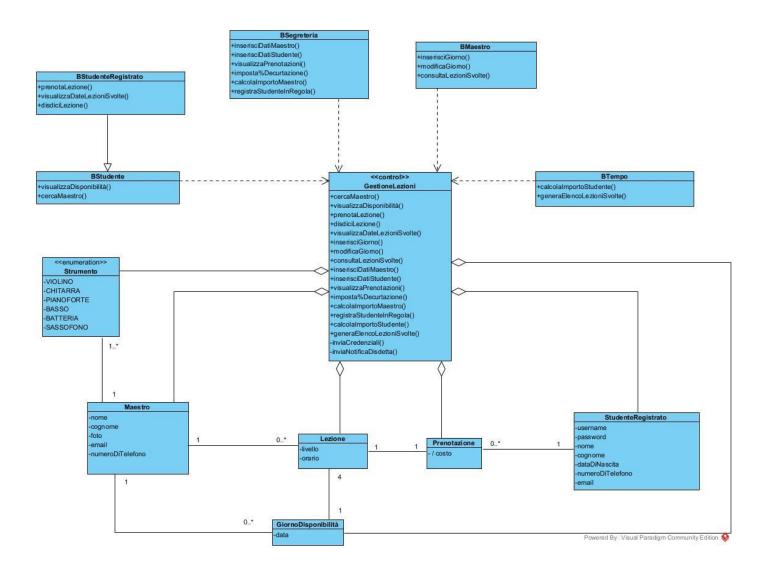
Caso d'uso:	Invio newsletter			
Attore primario	Maestro			
Attore secondario	-			
Descrizione	In ogni momento il Maestro può, attraverso il sistema, modificare una delle date in			
	cui è disponibile per le lezioni.			
Pre-Condizioni	Il Maestro ha precedentemente inserito almeno un giorno di disponibilità.			
Sequenza di eventi	1. Il caso d'uso inizia quando il Maestro richiede la modifica di un giorno di			
principale	disponibilità.			
	2. Il Maestro inserisce la data del giorno di disponibilità che desidera modificare.			
	3. Il sistema verifica la validità di questa data, ovvero se essa è presente tra i			
	giorni di disponibilità precedentemente inseriti dal Maestro. 4. Il Maestro inserisce la nuova data in cui è disponibile.			
	5. Il sistema verifica la validità di questa data, ovvero se essa non precede la			
	data di utilizzo del caso d'uso oppure se essa non coincide con la data di un			
	altro giorno di disponibilità esistente.			
	6. Il Maestro conferma di voler effettuare la modifica.			
	7. Il sistema sostituisce la data da modificare con la nuova data di			
D. (C. III.	disponibilità. La data in cui il Maestro è disponibile per le lezioni è stata modificata correttamente.			
Post-Condizioni	La data in cui il Maestro e disponibile per le lezioni e stata modificata correttamente.			
Casi d'uso correlati				
Sequenza di eventi alternativi	Se il Maestro inserisce come data da modificare una data non valida (non presente			
alternativi	nei giorni di disponibilità precedentemente inseriti dal Maestro), allora il sistema restituisce messaggio d'errore e termina il caso d'uso.			
	restraisee messaggio d'errore e termina il caso d'aso.			
	Dopo aver inserito una data da modificare valida, se il Maestro inserisce come nuova			
	data di disponibilità una data precedente alla data d'utilizzo del caso d'uso oppure			
	uguale alla data di un altro giorno di disponibilità esistente, allora il sistema			
	restituisce messaggio d'errore e termina il caso d'uso.			
	Se il Maestro non conferma la modifica oppure inserisce come nuova data di			
	disponibilità una data uguale alla data da modificare, allora il sistema non modifica			
	alcun giorno di disponibilità ed il caso d'uso si conclude.			
1				

2.6 - Diagramma delle classi

- Diagramma delle classi di prima analisi

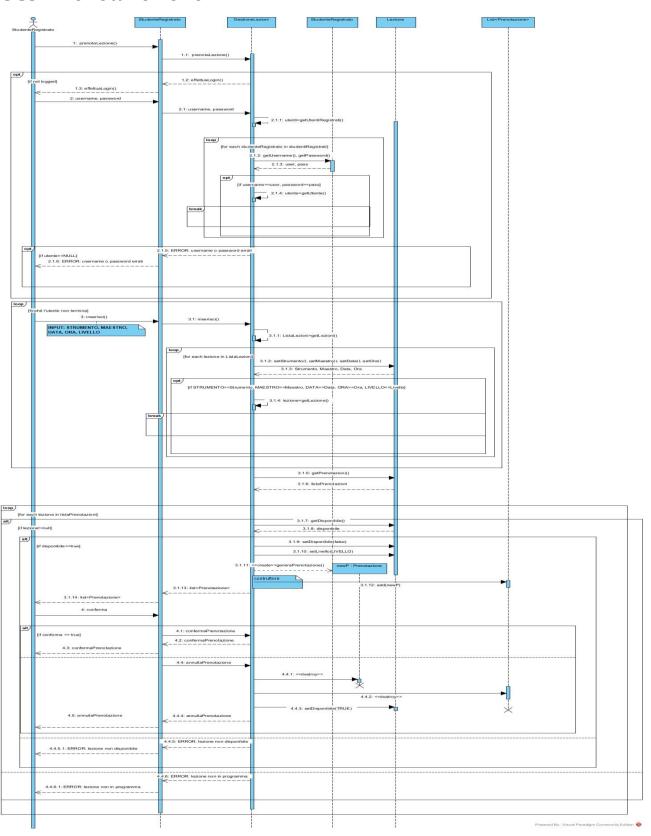


- Diagramma delle classi raffinato secondo il modello BCE

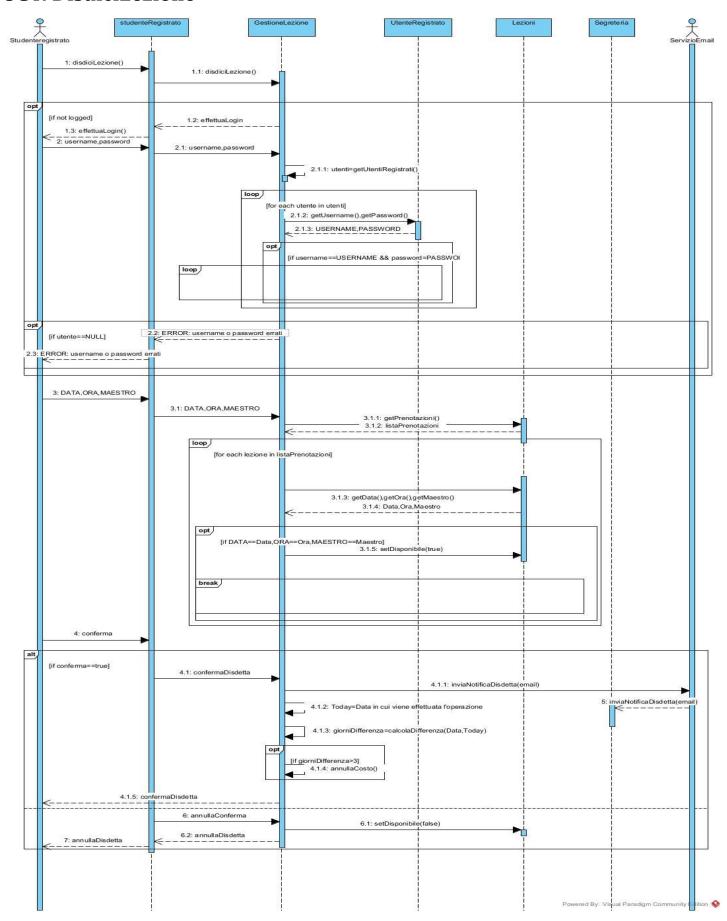


2.7 - Diagrammi di sequenza

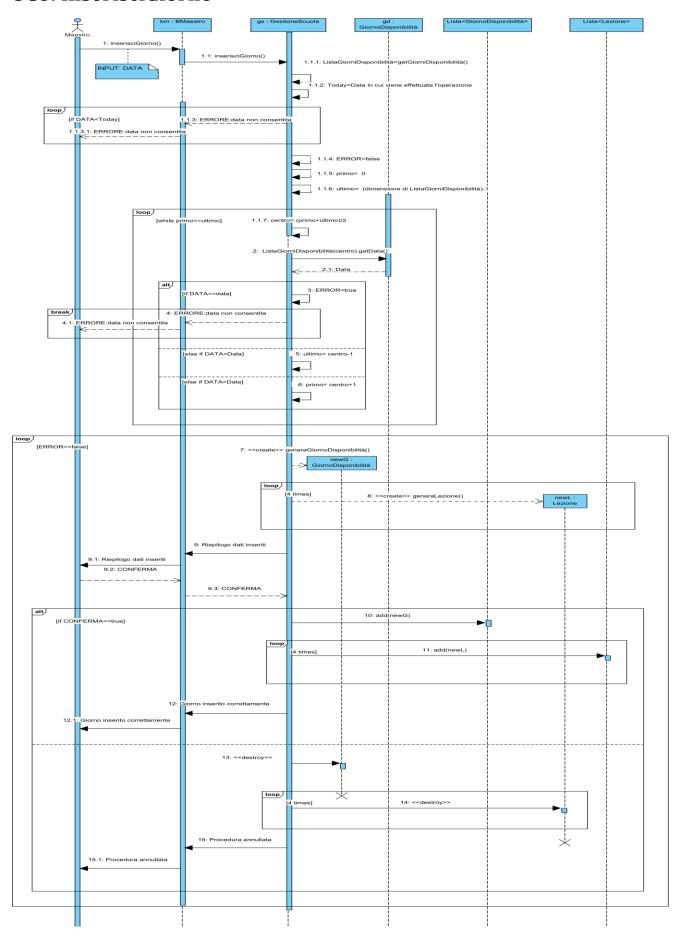
UC3: PrenotaLezione



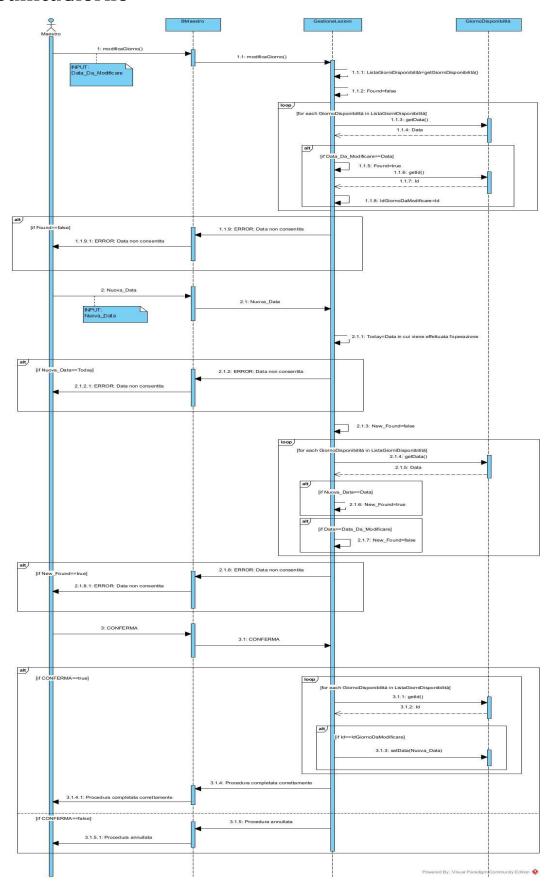
UC4: DisdiciLezione



UC6: InserisciGiorno



UC7: ModificaGiorno



3 - Stima dei costi

• Tabella di riferimento per le complessità di dati e transazioni

	SEMPLICE	MEDIO	COMPLESSO
NILF	3	4	6
NEIF	4	5	7
NEI	3	4	6
NEO	7	10	15
NEQ	5	7	10

- UC3: PrenotaLezione

	VALORE	SEMPLICE	MEDIO	COMPLESSO	ТОТ
NILF	1		4		4
NEIF	0				
NEI	5	3			15
NEO	1	7			7
NEQ	0				

UFP = 26
LLOC/FP = 1378

- UC4: DisdiciLezione

	VALORE	SEMPLICE	MEDIO	COMPLESSO	ТОТ
NILF	1		4		4
NEIF	0				
NEI	3	3			9
NEO	1	7			7
NEQ					

UFP = 20	
LLOC/FP = 1060	

- UC6: InserisciGiorno

	VALORE	SEMPLICE	MEDIO	COMPLESSO	TOT
NILF	1		4		4
NEIF	0				
NEI	1	3			3
NEO	1	7			7
NEQ	0				

UFP = 14				
LLOC/FP = 742				

FP =13.58
JAVA= 719.74

- UC7: ModificaGiorno

	VALORE	SEMPLICE	MEDIO	COMPLESSO	ТОТ
NILF	1		4		4
NEIF					
NEI	2	3			6
NEO	1	7			7
NEQ					

UFP = 17	
LLOC/FP = 901	

• Tabella elenco dei fattori correttivi (il cui valore è compreso tra 0 e 5)

FATTORI CORRETTIVI

COMUNICAZIONE DATI	2
DISTRIBUZIONE ELABORAZIONE	1
PRESTAZIONI	3
UTILIZZO INTENSIVO CONFIGURAZIONE	3
FREQUENZA DELLE TRANSAZIONI	3
INSERIMENTO DATI INTERATTIVO	2
EFFICIENZA PER L'UTENTE FINALE	4
AGGIORNAMENTO INTERATTIVO	3
COMPLESSITA' ELABORATIVA	0
RIUSABILITA'	3
FACILITA' INSTALLAZIONE	2
FACILITA' GESTIONE OPERATIVA	2
MOLTEPLICITA' DI SITI	1
FACILITA' DI MODIFICA	3

4 - Piano di test funzionale

PIANO DI TEST UTILIZZANDO IL METODO DEL CATEGORY-PARTITION TESTING PER LA FUNZIONALITÀ "Prenota Lezione"

Username	Password	Maestro	Data	Ora	Strumento	Livello	Elemento di sistema database
 Stringa di caratteri di lunghez za <= 50 Stringa di caratteri di lunghez za > 50 [ERROR] 	 Stringa di caratteri di lunghezz a >= 8 e <= 50 Stringa di caratteri di lunghezz a > 50 [ERROR] Stringa di caratteri di lunghezz a < 8 [ERROR] 	 Nome stringa caratter i <=50 e Cogno me stringa caratter i <=50 Nome stringa caratter i >50 [ERROR] Cogno me Stringa caratter i >50 [ERROR] 	 Data con form ato valid o (gg/mm/aaaa) Data con form ato non valid o [ERR OR] 	 Ora con formato valido [hh:mm], compres a tra le 17.00 e le 20.00. Ora con formato non valido [ERROR] Ora non compres a tra le 17.00 e le 20.00 [ERROR] 	Lo strument o deve essere uno tra i seguenti: violino, chitarra, pianofort e, batteria, basso, sassofon o. Altri strument i non validi [ERROR]	Livello della lezione: "principian te" o "intermedi o". Altri livelli non validi [ERROR]	 Lezione disponibile. Lezione non presente nel database [ERROR] Lezione già prenotata [ERROR] Username e/o password non presenti nel Database [ERROR]

Il numero di test da effettuarsi senza particolari vincoli è: 2*3*3*2*3*2*4 = 1728 Introduciamo i vincoli [ERROR].

Il numero di test da eseguire per testare singolarmente i vincoli è 13 (1 per Username, 2 per Password, 2 per Maestro, 1 per Data, 2 per Ora, 1 per Strumento, 1 per Livello, 3 per ElementoDiSistemaDatabase).

Il numero di test risultante è (1*1*1*1*1*1*1) + 13 = 14

TEST SUITE

Test	Descrizione	Classi di	Pre-condizioni	Input	Output	Post-
Case		equivalenza			Ottenuti	condizioni
ID		coperte				Attese
1	Tutti gli	Username	Il database è	{	Prenotazion	Lo studente
	input sono	valido,	inizializzato	Username:	e avvenuta	ottiene la sua
	validi.		correttamente.	"anto24",	con	prenotazione.
		Password valida,			successo.	
			Lo studente	Password:		
		Maestro valido,	effettua il login	"ciaoraga20",		
			correttamente.			
		Data valida,		Nome Maestro:		
			Le lezioni sono	"Andrea ",		
		Ora valida,	disponibili per			
			lo strumento	Cognome		
		Strumento	nella data e ora	Maestro:		
		valido,	selezionate.	"Pirlo",		
		Livello valido,		Data:		
				"24/06/2024",		
		Elementi di				
		sistema		Ora: "18:00",		
		database validi.				
				Strumento:		
				"Sassofono",		
				Livello:		
				"Base"		
				}		

2	Username:	Username:	Lo studente	{	Username	Il sistema
	stringa > 50	stringa > 50	vuole accedere	Username:	non valido.	chiede
	caratteri.	caratteri	per effettuare	"aaaaaaaaaaaaa		nuovamente
	our accorn	[ERROR],	la	aaaaaaaaaaaaaa		di inserire
		[=	prenotazione.	ааааааааааааааа		l'username.
			premotazionei	aaaaaaaaaaaaaa		
		Validi:		aaaaaaaaaaaaaa		
		· · · · · · · · · · · · · · · · · · ·		aaaaaaaaa",		
		Password,				
		Maestro,		Password:		
		Data,		"nocappo813",		
		Ora,				
		Strumento,		Nome Maestro:		
		Livello,		"Andrea ",		
		ElementoDiSiste		,		
		maDatabase.		Cognome		
				Maestro:		
				"Pirlo",		
				,		
				Data:		
				"24/06/2024",		
				_ ,, , , , , , , , , , , , , , , , , ,		
				Ora: "18:00",		
				Strumento:		
				"Sassofono",		
				,		
				Livello:		
				"Base"		
				}		
				,		
3	Password:	Password:	Lo studente	{	Password	Il sistema
	stringa di	stringa di	registrato	Username:	non valida.	chiede
	caratteri di	caratteri di	seleziona la	"dottoraro2",		nuovamente
	lunghezza >	lunghezza >50	funzionalità	<u> </u>		di inserire la
	50	[ERROR],	"Prenota	Password:		password
		<u>"</u>	Lezione".	"htppp111",		dell'utente.
				, , , ,		
		Validi:		Nome Maestro:		
			Il sistema	"Giancarlo",		
		Username,	richiede	,		
		Maestro,	all'utente	Cognome		
		Data,	registrato	Maestro: "Zita",		
		Ora,				
		J14,				

		Strumento, Livello, ElementiDiSiste maDatabase.	l'inserimento della password.	Data: "25/06/2024", Ora: "17:00", Strumento: "Batteria", Livello: "Base" }		
4	Password: stringa di caratteri di lunghezza <8	Password: stringa di caratteri di lunghezza <=8 [ERROR], Validi: Username, Maestro, Data, Ora, Strumento, Livello, ElementiDiSiste maDatabase.	Lo studente registrato seleziona la funzionalità prenota lezione. Il sistema richiede all'utente registrato l'inserimento della password.	{ Username: "angy7537", Password: "FNS!1926", Nome maestro: "Mauro", Cognome maestro: "Sassi", Data: "29/06/2024", Ora: "20:00", Strumento: "sassofono", Livello: "Base" }	Password non valida.	Il sistema chiede nuovamente di inserire la password dell'utente.
5	Nome Maestro: stringa > 50 caratteri.	Nome Maestro stringa > 50 caratteri [ERROR],	Lo studente vuole accedere per effettuare la prenotazione.	{ Username: "aaaaaaaaaaa", Password: "nocappo813",	Nome maestro non valido.	Il sistema chiede nuovamente di inserire il maestro.

	1	Validi:		Nome massins:		1
		validi:		Nome maestro:		
				"Andreaaaaaaaa		
		Username,		ааааааааааааа		
		Password,		ааааааааааааа		
		Data,		ааааааааааааа		
		Ora,		aaaaaaaaaaa",		
		Strumento,				
		Livello,		Cognome		
		ElementoDiSiste		Maestro:		
		maDatabase.		"Galimberti",		
				Data:		
				"24/06/2024",		
				Ora: "18:00",		
				Strumento:		
				"Sassofono",		
				3033010110 ,		
				Livello:		
				"Base"		
				}		
				,		
6	Cognome	Cognome	Lo studente	{	Cognome	Il sistema
	Maestro:	Maestro stringa	vuole accedere	Username:	maestro	chiede
	stringa > 50	> 50 caratteri	per effettuare	"aaaaaaaaaaaa",	non valido.	nuovamente
	caratteri.	[ERROR],	la	Password:		di inserire il
		[=::::],	prenotazione.	"nocappo813",		cognome
				,		maestro.
		Validi:		Nome Maestro:		
				"Andrea",		
		Username,		ĺ		
		Password,		Cognome		
		Data,		Maestro:		
		Ora,		"Galimbertiiiiiiiii		
		Strumento,				
		Livello,				
		ElementoDiSiste		111111111111111111111111111111111111111		
		maDatabase.		iiiiiiiiiiiiii",		
				ĺ		
				Data:		
				"24/06/2024",		
				21/00/2021,		
				21,00,2021,		

7	Data con formato non valido.	Data con formato non valido [ERROR], Validi: Username, Password, Maestro, Ora, Strumento, Livello, ElementoDiSiste maDatabase.	Lo studente vuole effettuare la prenotazione. Il sistema richiede all'utente l'inserimento della data.	Strumento: "Sassofono", Livello: "Base" } { Username: "oaksakoaos", Password: "nocappo813", Nome Maestro: "Mario", Cognome Maestro: "Rossi", Data: "24/06/24", Ora: "14:00",	Data inserita non esistente oppure formato errato. Riprovare	Il sistema chiede di inserire nuovamente la data.
8	Ora con formato non valido.	Ora con formato non valido [ERROR],	Lo studente vuole effettuare la	Ora: "14:00", Strumento: "Sassofono", Livello: "Base" } { Username: "oaksakoaos",	Non ci sono Lezioni per l'orario	Il sistema chiede nuovamente
		Validi: Username, Password, Maestro, Data,	Il sistema richiede all'utente l'inserimento dell'ora.	Password: "nocappo813", Nome Maestro: "Mario",	Le Lezioni iniziano solo alle 17:00, alle 18:00, alle 19:00	di inserire l'ora.

		Strumento, Livello, ElementoDiSiste maDatabase.		Cognome Maestro: "Rossi", Data: "24/06/2024", Ora: "19", Strumento: "Sassofono", Livello: "Base" }	oppure alle 20:00. Riprovare	
9	Ora non compresa tra le 17.00 e le 20.00.	Ora non compresa tra le 17.00 e le 20.00 [ERROR], Ora con formato valido, Validi: Username, Password, Maestro, Data, Strumento, Livello, ElementoDiSiste maDatabase.	Lo studente vuole effettuare la prenotazione. Il sistema richiede all'utente l'inserimento dell'ora.	{ Username: "oaksakoaos", Password: "nocappo813", Nome Maestro: "Mario", Cognome Maestro: "Rossi", Data: "24/06/2024", Ora: "14:00", Strumento: "Sassofono", Livello: "Base" }	Non ci sono Lezioni per l'orario scelto. Le Lezioni iniziano solo alle 17:00, alle 18:00, alle 19:00 oppure alle 20:00. Riprovare	Il sistema chiede nuovamente di inserire l'ora.
10	Altri strumenti non validi.	Strumento non valido [ERROR], Validi:	Lo studente vuole effettuare la prenotazione.	{ Username: "oaksakoaos", Password:	Non vengono erogate lezioni per lo	Il sistema chiede di inserire nuovamente lo strumento.

		Username, Password, Maestro, Data, Ora, Livello, ElementoDiSiste maDatabase.	Il sistema richiede all'utente l'inserimento dello strumento.	"nocappo813", Nome Maestro: "Mario", Cognome Maestro: "Rossi", Data: "24/06/2024", Ora: "18:00", Strumento: "Clarinetto", Livello: "Base" }	Strumento scelto. Riprovare scegliendo un altro strumento	
11	Altri livelli non validi.	Livello non valido [ERROR], Validi: Username, Password, Maestro, Data, Ora, Strumento, ElementiDiSiste maDatabase.	Il livello della lezione inserito non è presente nel sistema.	{ Username: "salvo10ra", Password: "noleg39day", Nome Maestro: "Ludovico", Cognome Maestro: "Lemme", Data: "14/06/2024", Ora: "18:00", Strumento: "Chitarra", Livello: "Avanzato" }	Le lezioni possono essere solo di 2 livelli: BASE (costo di 15.00 euro) oppure INTERMEDI O (costo di 30.00 euro).	Il sistema chiede nuovamente di inserire il livello della lezione.

12	Lezione non presente nel database [ERROR]	ElementoDiSiste maDatabase non valido [ERROR], Validi: Username, Password, Maestro, Data, Ora, Strumento, Livello.	La lezione non risulta nel database.	{ Username: "lory446", Password: "circuitodon", Nome Maestro: "Andrea", Cognome Maestro: "Presti", Data: "13/06/2024", Ora: "21:00", Strumento: "Batteria", Livello: "Intermedio" }	Lezione non trovata.	Il sistema non trova lezione in input e non può procedere per la prenotazione.
13	Lezione già prenotata [ERROR]	ElementoDiSiste maDatabase non valido [ERROR], Validi: Username, Password, Maestro, Data, Ora, Strumento, Livello.	La lezione risulta essere già prenotata.	{ Username: "pablo72", Password: "aitriplei22", Nome Maestro: "Stefano", Cognome Maestro: "Pioli", Data: "26/06/2024",	La lezione non è disponibile.	Il sistema non può prenotare la lezione. Il sistema richiede di inserire nuovamente i dati della lezione.

				Ora: "19:00", Strumento: "Pianoforte", Livello: "Base" }		
14	Username e/o password non presenti nel Database [ERROR]	ElementoDiSiste maDatabase non valido [ERROR], Validi: Username, Password, Maestro, Data, Ora, Strumento, Livello.	Username e/o password utente non sono presenti nel Database.	{ Username: "manu223", Password: "Maestroregistra to", Nome maestro: "Mario", Cognome maestro: "Rossi", Data: "22/07/2024", Ora: "17:00", Strumento: "Sassofono", Livello: "Intermedio" }	Studente non registrato.	Il sistema non può prenotare la lezione.

PIANO DI TEST UTILIZZANDO IL METODO DEL CATEGORY-PARTITION TESTING PER LA FUNZIONALITÀ "Disdici Lezione"

Nome Maestro	Cognome Maestro	Data	Ora	Elemento di sistema database
• Nome stringa di caratteri di lunghezza <=50	• Cognome stringa di caratteri di lunghezza <=50	Data con formato valido (gg/mm/a aaa)	Ora con formato valido (hh.mm), compresa tra le 17.00 e le 20.00.	 La lezione è presente nel database. Lezione non presente nel database [ERROR]
Nome stringa di caratteri di lunghezza >50 [ERROR]	Cognome stringa di caratteri di lunghezza >50 [ERROR]	Data con formato non valido [ERROR]	 Ora con formato non valido [ERROR] Ora non compresa tra le 17.00 e le 20.00 [ERROR] 	Lezione non prenotata [ERROR]

Il numero di test da effettuarsi senza particolari vincoli è: 2*2*2*3*3 = 72

Introduciamo i vincoli [ERROR].

Il numero di test da eseguire per testare singolarmente i vincoli è 7 (1 per Nome Maestro, 1 per Cognome Maestro, 1 per Data, 2 per Ora, 2 per Elemento di sistema database).

Il numero di test risultante è (1*1*1*1*1) + 7 = 8

TEST SUITE

Test Case ID	Descrizione	Classi di equivalenza coperte	Pre-condizioni	Input	Output Ottenuti	Post-condizioni Attese
1	Tutti gli input sono validi.	Nome Maestro valido, Cognome Maestro valido, Data valida, Ora valida, ElementoDi SistemaData base valido.	Lo studente ha effettuato il login. Il database è inizializzato correttamente. La lezione esiste per la data, ora e per il maestro selezionato.	{ Nome Maestro: "Andrea", Cognome Maestro: "Pirlo", Data: "27/06/2024", Ora: "19.00" }	DISDETTA AVVENUTA CON SUCCESSO!	La lezione è stata disdetta correttamente. Il sistema notifica via mail la disdetta alla segreteria.
2	Nome Maestro: stringa > 50 caratteri	Nome Maestro stringa > 50 caratteri [ERROR], Validi: Cognome, Data, Ora, ElementoDi SistemaData base.	Lo studente ha inserito il maestro per la disdetta della lezione.	{ Nome Maestro: "Andreaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa	Nome maestro non valido.	Il sistema chiede nuovamente di inserire il nome maestro.
3	Cognome Maestro: stringa > 50 caratteri	Cognome Maestro stringa > 50 caratteri [ERROR],	Lo studente ha inserito il maestro per la disdetta della lezione.	{ Nome Maestro: "Andrea",	Cognome maestro non valido	Il sistema chiede nuovamente di inserire il

		Validi: Nome, Data, Ora, ElementoDi SistemaData base.		Cognome Maestro: "Pirlooooooooo ooooooooooo ooooooooooo oooooo		cognome maestro.
4	Data con formato non valido.	Data con formato non valido [ERROR], Validi: Nome, Cognome, Ora, ElementoDi SistemaData base.	Lo studente vuole effettuare la disdetta.	{ Nome Maestro: "Mario", Cognome Maestro: "Rossi", Data: "24/06/24", Ora: "18:00" }	Data inserita non esistente oppure formato errato. Riprovare	Il sistema chiede nuovamente di inserire la data.
5	Ora con formato non valido.	Ora con formato non valido [ERROR], Validi: Nome, Cognome, Data, ElementoDi SistemaData base.	Lo studente vuole effettuare la disdetta.	{ Nome Maestro: "Mario", Cognome Maestro: "Rossi", Data: "24/06/2024", Ora: "18" }	Non ci sono Lezioni per l'orario scelto. Le Lezioni iniziano solo alle 17:00, alle 18:00, alle 19:00 oppure alle 20:00. Riprovare	Il sistema chiede nuovamente di inserire l'ora.

6	Ora non compresa tra le 17.00 e le 20.00.	Ora non compresa tra le 17.00 e le 20.00 [ERROR], Validi: Nome, Cognome, Data, ElementoDi SistemaData base.	Lo studente vuole effettuare la disdetta.	{ Nome Maestro: "Mario" Cognome Maestro: "Rossi", Data: "24/06/2024", Ora: "14.00" }	Non ci sono Lezioni per l'orario scelto. Le Lezioni iniziano solo alle 17:00, alle 18:00, alle 19:00 oppure alle 20:00. Riprovare	Il sistema chiede nuovamente di inserire l'ora.
7	Lezione non presente nel database [ERROR].	ElementoDi SistemaData base non valido [ERROR], Validi: Nome, Cognome, Data, Ora.	La lezione non è stata prenotata o è stata già disdetta.	{ Nome Maestro: "Stefano", Cognome Maestro: "Pioli", Data: "26/06/2024", Ora: "19:00" }	Lezione non trovata.	Il sistema non può effettuare la disdetta per la lezione richiesta. Il sistema richiede di inserire nuovamente i dati della lezione.
8	Lezione non prenotata [ERROR].	ElementoDi SistemaData base non valido [ERROR], Validi: Maestro, Ora, Data.	La lezione non è stata prenotata.	{ Nome Maestro: "Stefano", Cognome Maestro: "Pioli", Data: "26/06/2024", Ora: "19:00" }	Lezione non trovata.	Il sistema non può effettuare la disdetta per la lezione richiesta.

PIANO DI TEST UTILIZZANDO IL METODO DEL CATEGORY-PARTITION TESTING PER LA FUNZIONALITÀ "Inserisci Giorno"

Data	Elemento di sistema database			
Data con formato valido (gg/mm/aaaa).	Data non presente nel database.			
Data con formato non valido [ERROR]	Data già presente nel database [ERROR]			

Il numero di test da effettuarsi senza particolari vincoli è: 2*2 = 4

Introduciamo i vincoli [ERROR].

Il numero di test da eseguire per testare singolarmente i vincoli è 2 (1 per Data, 1 per ElementoDiSistemaDatabase).

Il numero di test risultante è (1*1) + 2 = 3

TEST SUITE

Test	Descrizione	Classi di	Pre-condizioni	Input	Output	Post-condizioni
Case		equivalenza			Attesi	Attese
ID		coperte				
1	Tutti gli	Data con	II database è	{	Un nuovo	Il giorno di
	input sono	formato valido.	inizializzato		giorno di	disponibilità è
	validi.		correttamente.	Data:	disponibilità	stato inserito
		Elementi di	Il maestro ha	"09/07/2024"	è stato	correttamente.
		sistema	effettuato	}	inserito con	
		database validi.	l'accesso.		successo.	
2	Data con	Data con	Il database è	{	Data inserita	Il sistema chiede
	formato	formato non	inizializzato		non	di inserire
	non valido.	valido [ERROR].	correttamente.	Data:	esistente	nuovamente la
			Il maestro ha	"09/07/24"	oppure	data.
		Elementi di	effettuato		formato	
		sistema	l'accesso.	}	errato.	
		database validi.				
					Riprovare	

3	Data già	Data con	Il database è	{	Giorno già	Il caso d'uso
	presente	formato valido.	inizializzato		presente.	termina
	nel		correttamente.	Data:		notificando la
	database.	Data già	II maestro ha	"09/10/2024"		presenza del
		presente nel	effettuato	}		giorno di
		database	l'accesso.			disponibilità.
		[ERROR].				

PIANO DI TEST UTILIZZANDO IL METODO DEL CATEGORY-PARTITION TESTING PER LA FUNZIONALITÀ "Modifica Giorno"

Data da modificare	Nuova data di disponibilità	Elemento di sistema database
Data da modificare con	Nuova data di disponibilità	Data da modificare presente nel
formato valido	con formato valido	database e nuova data
(gg/mm/aaaa)	(gg/mm/aaaa)	disponibile.
Data da modificare con	Nuova data di disponibilità	 Data da modificare non
formato non valido	con formato non valido	presente nel database [ERROR] Nuova data non disponibile
[ERROR]	[ERROR]	[ERROR]

Il numero di test da effettuarsi senza particolari vincoli è: 2*2*3 = 12

Introduciamo i vincoli [ERROR].

Il numero di test da eseguire per testare singolarmente i vincoli è 4 (1 per Data da modificare, 1 per Nuova data di disponibilità, 2 per Elemento di sistema database).

Team: SE(A)^2

Il numero di test risultante è (1*1*1) + 4 = 5

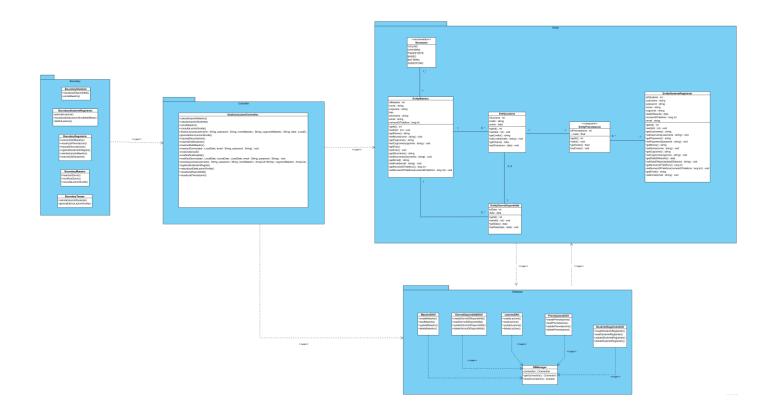
TEST SUITE

Test Case ID	Descrizione	Classi di equivalenza coperte	Pre-condizioni	Input	Output Attesi	Post- condizioni Attese
1	Tutti gli input sono validi.	Data da modificare con formato valido. Nuova data di disponibilità con formato valido. Elementi di sistema database validi.	Il database è inizializzato correttamente. Il maestro ha effettuato l'accesso.	{ Data da modificare: "22/05/2025" Nuova data: "25/05/2025" }	Giorno di disponibilità modificato con successo!	La data del giorno di disponibilità selezionato è stata modificata con successo.
2	Data da modificare con formato non valido.	Data da modificare con formato non valido [ERROR]. Nuova data di disponibilità con formato valido. Elementi di sistema database validi.	Il database è inizializzato correttamente. Il maestro ha effettuato l'accesso.	{ Data da modificare: "22-05-2025" Nuova data: "25/05/2025" }	Data inserita non esistente oppure formato errato. Riprovare	Il sistema chiede di inserire nuovamente la data da modificare.
3	Nuova data di disponibilità con formato non valido.	Data da modificare con formato valido. Nuova data di disponibilità	Il database è inizializzato correttamente. Il maestro ha effettuato l'accesso.	{ Data da modificare: "22/05/2025"	Data inserita non esistente oppure formato errato.	Il sistema chiede di reinserire la nuova data di disponibilità.

		con formato		Nuova data:		
		non valido		Nuova data: "25-05-2025"		
		[ERROR].		25 05 2025		
				}		
		Elementi di				
		sistema				
		database				
		validi.				
4	Data da	Data da	Il database è	{	Giorno	Il sistema
	modificare	modificare	inizializzato		indicato non	chiede di
	non presente	con formato	correttamente.	Data da	presente nel	inserire
	nel database.	valido.		modificare:	sistema.	nuovamente
		N data di	Il maestro ha	"24/05/2025",		la data da
		Nuova data di	effettuato	Nivers data		modificare.
		disponibilità con formato	l'accesso.	Nuova data: "25/05/2025"		
		valido.		23/03/2023		
		vanao.		}		
		Data da		,		
		modificare				
		non presente				
		nel database				
		[ERROR].				
5	Nuova data	Data da	Il database è	{	Nuovo giorno	Il sistema
	non	modificare	inizializzato		indicato già	chiede di
	disponibile.	con formato	correttamente.	Data da	presente nel	reinserire la
		valido.	11	modificare:	sistema.	nuova data di
		Nuova data di	Il maestro ha	"22/05/2025"		disponibilità.
		Nuova data di disponibilità	effettuato l'accesso.	Nuova data:		
		con formato	Taccesso.	"22/05/2025"		
		valido.		22,03,2023		
				}		
		Nuova data di		,		
		disponibilità				
		già presente				
		nel database				
		[ERROR].				

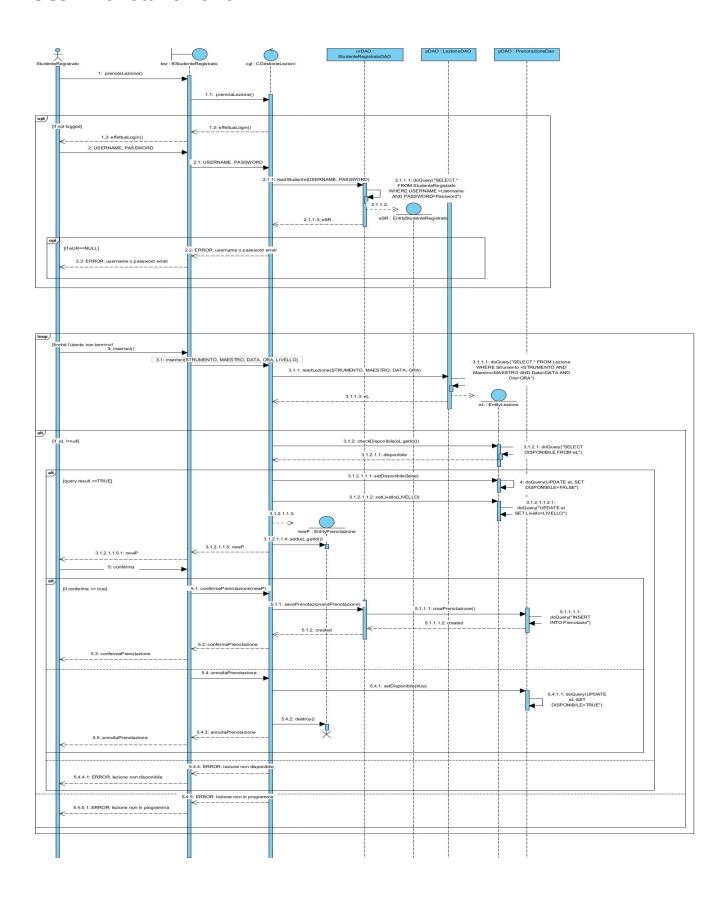
5 – Progettazione

5.1 - Diagramma delle classi

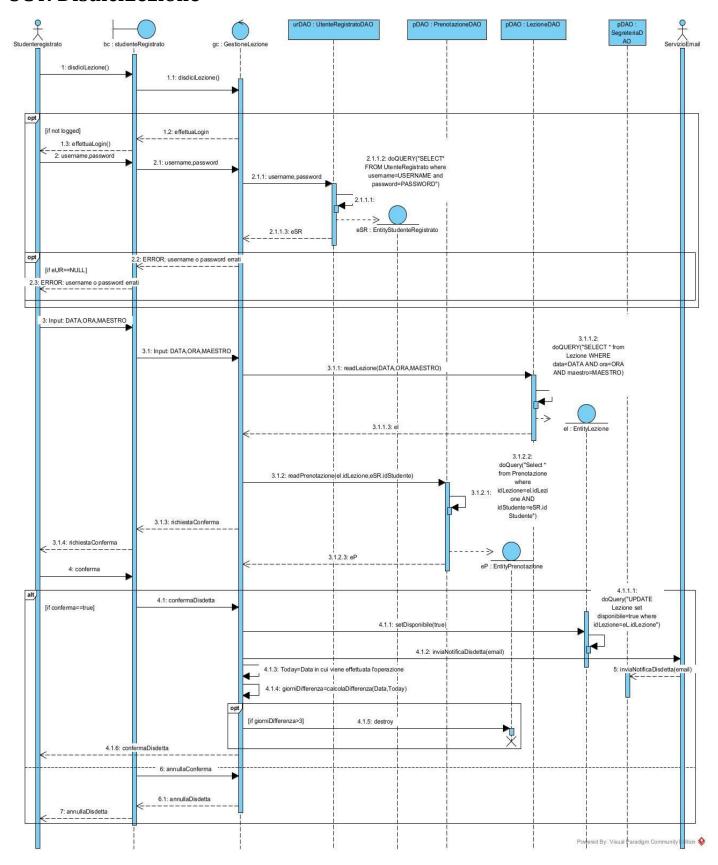


5.2 - Diagrammi di sequenza

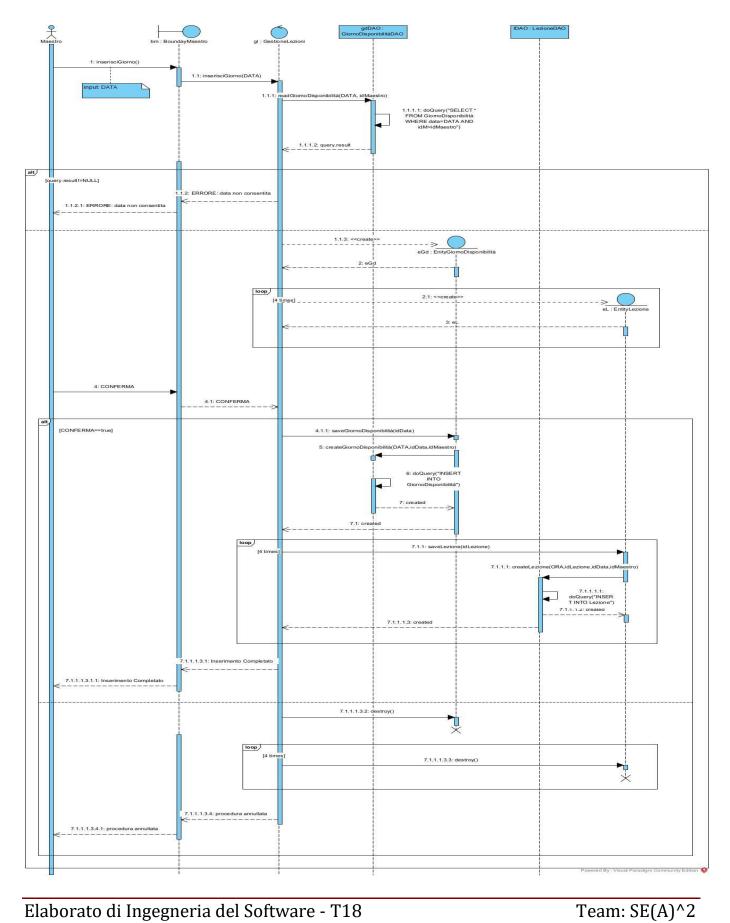
UC3: PrenotaLezione



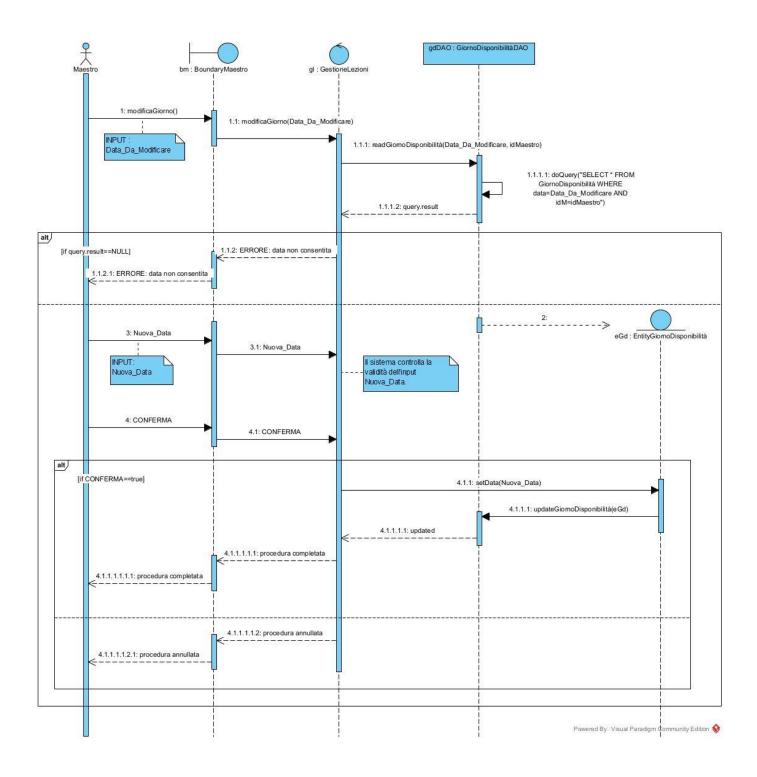
UC4: DisdiciLezione



UC6: InserisciGiorno



UC7: ModificaGiorno



6 - Implementazione

Package:

- Control
- Boundary
- Entity
- DAO
- Exception
- Test

Package Control:

Classi

GestioneLezioni

Librerie Utilizzate

- import java.sql.Time;
- import java.time.LocalDate;
- import java.time.LocalTime;
- import java.time.temporal.ChronoUnit;
- import java.util.ArrayList;

Package Boundary:

Classi

- BoundayStudenteRegistrato
- BoundayMaestro

Librerie Utilizzate

- import java.time.LocalDate;
- import java.time.LocalTime;
- import java.time.format.DateTimeFormatter;

- import java.util.ArrayList;
- import java.util.Scanner;
- import java.util.regex.Matcher;

• import java.util.regex.Pattern;

Package Entity:

Classi

- Studente (abstract)
- StudenteRegistrato
- Maestro
- Lezione
- Prenotazione
- GiornoDisponibilità

Librerie Utilizzate

- import java.time.LocalTime;
- import java.time.LocalDate;
- import java.sql.Date;

Package DAO:

Classi

- DBManager
- GiornoDisponibilitàDAO
- MaestroDAO
- StudenteRegistratoDAO
- LezioneDAO
- PrenotazioneDAO

Librerie Utilizzate

- import java.sql.Connection;
- import java.sql.Date;
- import java.sql.PreparedStatement;

- import java.sql.ResultSet;
- import java.sql.SQLException;
- import java.time.LocalDate;

- import java.time.format.DateTimeFormatter;
- import java.sql.Time;
- import java.time.LocalTime;
- import java.util.ArrayList;
- import java.sql.DriverManager;

Package Exception

Classi

- DAOException
- DBConnectionException
- OperationException

Artefatti necessari per l'installazione ed esecuzione del programma

- mysql-connector-j-8.3.0.jar
- junit-4.13.2-javadoc.jar
- junit-4.13.2.jar
- hamcrest-core-1.3-javadoc.jar
- hamcrest-core-1.3.jar
- XAMPP
- MySQL Workbench 8.0.36

Nel caso venga utilizzato Visual Studio Code si consiglia di aggiungere le estensioni:

- Database Client JDBC
- JUnit JAR Downloader
- Extension Pack for Java

7 - Testing

7.1 - Test funzionale

FUNZIONALITÀ "Prenota Lezione"

Test Case ID	Descrizione	Classi di equivalenza coperte	Pre- condizioni	Input	Output Attesi	Post- condizioni Attese	Output Ottenu ti	Post- condizi oni Ottenut e	Esito (FAIL, PASS)
1	Tutti gli input sono validi.	Username valido, Password valida, Maestro valido, Data valida, Strumento valido, Livello valido, Elementi di sistema database validi.	inizializzat o correttam ente. Lo studente effettua il login correttam ente. Le lezioni sono disponibili per lo strumento nella data e ora selezionat e.	{ Userna me: "anto24 ", Passwor d: "ciaorag a20", Nome Maestro : "Andrea ", Cognom e Maestro : "Pirlo", Data: "24/06/ 2024", Ora: "18:00", Strume	Prenot azione avvenu ta con success o.	Lo studente ottiene la sua prenotazio ne.	Prenot azione avvenu ta con success o.	Lo student e ottiene la sua prenota zione.	PASS
				nto:					

"Sassof ono",	
ono",	
Livello:	
"Base"	
2 Username: Username: Lo { Userna II sistema Userna II	PASS
stringa > 50 stringa > 50 studente Userna me chiede me non sistem	
caratteri. caratteri vuole me: non nuovamen valido. chiede	1
	_
per aaaaaaa inserire ente d	
effettuare aaaaaaa l'usernam inserir	
Validi: la aaaaaaa e. e. l'userr	a
prenotazi aaaaaaa me.	
Password, one. aaaaaaa	
Maestro, aaaaaaaa	
Data, aaaaaaa l	
Ora, aaaaaaaa	
Strumento, aaaaaaaa	
Livello, aaaaaaaa	
ElementoDi aa",	
SistemaData	
base. Passwor	
d:	
"nocap	
po813",	
Nome	
Maestro Maestro	
"Andrea	
Cognom	
e e e e e e e e e e e e e e e e e e e	
Maestro	
"Pirlo",	
Data:	
"24/06/	
2024",	

				Ora: "18:00", Strume nto: "Sassof ono", Livello: "Base" }					
3	Password: stringa di caratteri di lunghezza > 50	Password: stringa di caratteri di lunghezza >50 [ERROR], Validi: Username, Maestro, Data, Ora, Strumento, Livello, ElementiDiS istemaData base.	Lo studente registrato seleziona la funzionali tà "Prenota Lezione". Il sistema richiede all'utente registrato l'inserime nto della password.	{ Userna me: "dottora ro2", Passwor d: "htppp1 11", Nome Maestro : "Gianca rlo", Cognom e Maestro : "Zita", Data: "25/06/ 2024", Ora: "17:00", Strume nto:	Passwo rd non valida.	Il sistema chiede nuovamen te di inserire la password dell'utente .	Passwo rd non valida.	Il sistema chiede nuovam ente di inserire la passwor d.	PASS

4	Password: stringa di caratteri di lunghezza	Password: stringa di caratteri di lunghezza	Lo studente registrato seleziona	"Batteri a", Livello: "Base" } { Userna me: "angy75	Passwo rd non valida	Il sistema chiede nuovamen te di	Passwo rd non valida.	II sistema chiede nuovam	PASS
	<=8	>8 [ERROR], Validi: Username, Maestro, Data, Ora, Strumento, Livello, ElementiDiS istemaData base.	la funzionali tà prenota lezione. Il sistema richiede all'utente registrato l'inserime nto della password.	37", Passwor d: "FNS!19 26", Nome maestro: "Mauro" Cognom e maestro: "Sassi", Data: "29/06/ 2024", Ora: "20:00", Strume nto:		inserire la password dell'utente .		ente di inserire la passwor d dell'ute nte.	
				"sassofo no", Livello: "Base"					

				ì					
				J					
N st	Nome Maestro: tringa > 50 aratteri.	Maestro stringa > 50 caratteri [ERROR], Validi: Username, Password, Cognome Maestro, Data, Ora, Strumento, Livello, ElementoDi SistemaData base.	Lo studente vuole accedere per effettuare la prenotazi one.	{ Userna me: "aaaaaa aaaaaa" , Passwor d: "nocap po813", Nome maestro : "Andrea aaaaaaa aaaaaaa aaaaaaa aaaaaaa aaaaaa	Nome Maestr o non valido.	Il sistema chiede nuovamen te di inserire il nome del maestro.	Nome maestr o non valido.	II sistema chiede di inserire il nome del maestro .	PASS

6	Cognome	Cognome	Lo	"Sassof ono", Livello: "Base" }	Cogno	Il sistema	Cogno	II	PASS
	Maestro: stringa > 50 caratteri.	Maestro stringa > 50 caratteri [ERROR], Validi: Username, Password, Nome Maestro, Data, Ora, Strumento, Livello, ElementoDi SistemaData base.	studente vuole accedere per effettuare la prenotazi one.	Userna me: "aaaaaa aaaaaa" , Passwor d: "nocap po813", Nome Maestro : "Andrea ", Cognom e Maestro : "Galimb ertiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiii	me Maestr o non valido.	chiede nuovamen te di inserire il cognome del maestro.	me maestr o non valido.	sistema chiede di inserire nuovam ente il cognom e del maestro .	

				Ora: "18:00", Strume nto: "Sassof ono", Livello: "Base" }					
7	Data con formato non valido.	Data formato non valido [ERROR], Validi: Username, Password, Maestro, Ora, Strumento, Livello, ElementoDi SistemaData base.	Lo studente vuole effettuare la prenotazi one. Il sistema richiede all'utente l'inserime nto della data.	{ Userna me: "oaksak oaos", Passwor d: "nocap po813", Nome Maestro : "Mario" , Cognom e Maestro : "Rossi", Data: "24/06/ 24", Ora: "14:00", Strume nto:	Data inserita non esisten te oppure format o errato. Riprova re	Il sistema chiede di inserire nuovamen te la data.	Data inserita non esisten te oppure format o errato. Riprova re	Il sistema chiede di inserire nuovam ente la data	PASS

	I	I	<u> </u>	" 6 5	T .			ı	
				"Sassof					
				ono",					
				Livello:					
				"Base"					
				}					
8	Ora con	Ora con	Lo	{	Non ci	Il sistema	Non ci	II	PASS
	formato	formato non	studente	Userna	sono	chiede	sono	sistema	
	non valido.	valido	vuole	me:	Lezioni	nuovamen	Lezioni	chiede	
		[ERROR],	effettuare	"oaksak	per	te di	per	nuovam	
			la	oaos",	l'orario	inserire	l'orario	ente di	
			prenotazi		scelto.	l'ora.	scelto.	inserire	
		Validi:	one.	Passwor	Le		Le	l'ora.	
				d:	Lezioni		Lezioni		
		Username,		"nocap	inizian		iniziano		
		Password,	Il sistema	po813",	o solo		solo		
		Maestro,	richiede	, , ,	alle		alle		
		Data,	all'utente	Nome	17:00,		17:00,		
		Strumento,	l'inserime	Maestro	alle		alle		
		Livello,	nto		18:00,		18:00,		
		ElementoDi	dell'ora.	· "Mario"	alle		alle		
		SistemaData	uen ora.	IVIALIO	19:00		19:00		
				,					
		base.		Comman	oppure		oppure		
				Cognom	alle		alle		
				e	20:00.		20:00.		
				Maestro	Riprova		Riprova		
				:	re		re		
				"Rossi",					
				Data:					
				"24/06/					
				2024",					
				Ora:					
				"19",					
				Strume					
				nto:					
				"Sassof					
				ono",					
				Livello:					
				"Base"					

				}					
9	Ora non	Ora non	Lo studente	{ Userna	Non ci	Il sistema chiede	Non ci	II sistema	PASS
	compresa tra le 17.00	compresa tra le 17.00	vuole	me:	sono Lezioni	nuovamen	sono Lezioni	chiede	
	e le 21.00.	e le 21.00	effettuare	"oaksak	per	te di	per	nuovam	
		[ERROR],	la	oaos",	l'orario	inserire	l'orario	ente di	
			prenotazi		scelto.	l'ora.	scelto.	inserire	
		Ora con	one.	Passwor	Le		Le	l'ora	
		formato		d:	Lezioni		Lezioni		
		valido,		"nocap	inizian		iniziano		
			Il sistema	po813",	o solo		solo		
			richiede		alle		alle		
		Validi:	all'utente	Nome	17:00,		17:00,		
		Hearnama	l'inserime	Maestro	alle		alle		
		Username, Password,	nto dell'ora.	: "Mario"	18:00, alle		18:00, alle		
		Maestro,	ueli Uid.		19:00		19:00		
		Data,		,	oppure		oppure		
		Strumento,		Cognom	alle		alle		
		Livello,		e	20:00.		20:00.		
		ElementoDi		Maestro	Riprova		Riprova		
		SistemaData		:	re		re		
		base.		"Rossi",					
				Data:					
				"24/06/					
				2024",					
				Ora:					
				"14:00",					
				Strume					
				nto:					
				"Sassof					
				ono",					
				Livello:					
				"Base"					
				}					
10	Strumento	Strumento	Lo	{	Non	Il sistema	Non	II	PASS
	non valido	non valido	studente	Userna	vengon	chiede di	vengon	sistema	
		[ERROR],	vuole	me:	0	inserire	0	chiede	

		Validi:	effettuare la prenotazi	"oaksak oaos",	erogat e lezioni	nuovamen te lo strumento	erogate lezioni per lo	di inserire nuovam	
		Username, Password,	one.	Passwor d: "nocap	per lo Strume nto		Strume nto scelto.	ente lo strumen to	
		Maestro, Data, Ora,	Il sistema richiede all'utente	po813", Nome	scelto. Riprova re		Riprova re		
		Livello, ElementoDi SistemaData	l'inserime nto dello strumento	Maestro : "Mario"	sceglie ndo un altro		sceglie ndo un altro		
		base.		, Cognom	strume nto		strume nto		
				e Maestro					
				"Rossi", Data:					
				"24/06/ 2024",					
				Ora: "18:00",					
				Strume nto: "Clarine tto",					
				Livello: "Base" }					
11	Altri livelli.	Livello non valido [ERROR],	Il livello della lezione inserito non è presente	{ Userna me: "salvo1 Ora",	Le lezioni posson o essere solo di	Il sistema chiede nuovamen te di inserire il livello	Le lezioni posson o essere solo di	Il sistema chiede nuovam ente di inserire	PASS
		Validi:	nel sistema.	Passwor d:	2 livelli: BASE	della lezione.	2 livelli: BASE	il livello	

		Username, Password, Maestro, Data, Ora, Strumento, ElementiDiS istemaData base.		"noleg3 9day", Nome Maestro : "Ludovi co", Cognom e Maestro : "Lemm e", Data: "14/06/ 2024", Ora: "18:00", Strume nto: "Chitarr a", Livello: "Avanza to"	(costo di 15.00 euro) oppure INTER MEDIO (costo di 30.00 euro)		(costo di 15.00 euro) oppure INTER MEDIO (costo di 30.00 euro)	della lezione	
				to" }					
12	Lezione non presente nel database [ERROR]	ElementoDi SistemaData base non valido [ERROR], Validi: Username, Password,	La lezione non risulta nel database.	{ Userna me: "lory44 6", Passwor d: "circuito don",	Lezione non trovata	Il sistema non trova lezione e non può procedere per la prenotazio ne.	Lezione non trovata	Il sistema non trova la lezione non può procede re per la prenota zione	PASS

	I	Maestro,		Nome		Ι			
				Maestro					
		Data,		. Iviaesti O					
		Ora,		· "Anadros					
		Strumento,		"Andrea "					
		Livello.		,					
				Cognom					
				е					
				Maestro					
				:					
				"Presti",					
				Data:					
				"13/06/					
				2024",					
				Ora:					
				"21:00",					
				Strume					
				nto:					
				"Batteri					
				a",					
				Livello:					
				"Interm					
				edio"					
				}					
13	Lezione già	ElementoDi	La lezione	{	Lezione	Il sistema	La	П	PASS
	prenotata	SistemaData	risulta	Userna	non	non può	lezione	sistema	
	[ERROR]	base non	essere già	me:	disponi	prenotare	non è	non può	
	-	valido	prenotata.	"pablo7	bile.	la lezione.	disponi	prenota	
		[ERROR],	'	2",			bile.	re la	
]		ĺ				lezione.	
				Passwor					
		Validi:		d:					
				"aitriple					
		Username,		i22",					
		Password,		122,					
		Maestro,		Nome					
		Data,		Maestro					
		Ora,		:					
		Strumento,							

		Livello.		"Stefan o", Cognom e Maestro : "Pioli", Data: "26/06/ 2024", Ora: "19:00", Strume nto: "Pianof orte", Livello:					
14	Username e/o password non presenti nel Database [ERROR]	ElementoDi SistemaData base non valido [ERROR], Validi: Username, Password, Maestro, Data, Ora, Strumento, Livello.	Username e/o password utente non sono presenti nel Database	"Base" { Userna me: "manu2 23", Passwor d: "Maestr oregistr ato", Nome maestro : "Mario" ,	Studen te non registra to.	Il sistema non può prenotare la lezione.	Studen te non registra to.	II sistema non può prenota re la lezione.	PASS

 	-			
	Cog	nom		
	е			
	mae	stro		
	:			
	"Ros	ssi",		
	Data	a:		
	"22/			
	2024			
		. ,		
	Ora:			
	"17:			
	17.	00,		
	Stru	ma		
	nto:			
	"Sas			
	ono	<i>"</i> ,		
	Live			
	"Into			
	edio	"		
	}			

FUNZIONALITÀ "Disdici Lezione"

Test	Descrizio	Classi di	Pre-	Input	Output	Post-	Output	Post-	Esito
Case	ne	equivalenza	condizioni		Attesi	condizio	Ottenuti	condizio	(FAIL,
ID		coperte				ni Attese		ni	PASS)
								Ottenute	
1	Tutti gli	Maestro	Lo	{	DISDET	La	DISDETT	La	PASS
	input	valido,	studente	Nome	TA	lezione è	Α	lezione è	
	sono		ha	Maestro	AVVEN	stata	AVVENU	stata	
	validi.	Data valida,	effettuato	:	UTA	disdetta	TA CON	disdetta	
			il login.	"Andrea	CON	corretta	SUCCESS	corretta	
		Ora valida,		",	SUCCE	mente.	0!	mente.	
			II database		SSO!				
		ElementoDiS	è	Cognom				II	
		istemaDatab	inizializzat	е				sistema	
		ase valido.	0	Maestro				notifica	
			correttam	: "Pirlo",				via mail	
			ente.					la	
			La lezione	Data:				disdetta	
			esiste per	"27/06/				alla	
			la data,	2024",				segreteri	
			ora e per il						
			maestro	Ora:					
			selezionat	"19.00"					
			0.	}					
2	Nome	Nome	Lo	{	Nome		Nome	II	PASS
	Maestro:	Maestro	studente	Nome	Maestr	sistema	maestro	sistema	
	stringa >	stringa > 50	ha inserito	Maestro	o non	chiede	non	chiede	
	50	caratteri	il maestro	:	valido.	nuovam	valido.	nuovame	
	caratteri	[ERROR],	per la	"Andrea		ente di		nte di	
			disdetta	ааааааа		inserire		inserire il	
		Validi:	della	ааааааа		il Nome		Nome	
			lezione.	ааааааа		maestro.		maestro	
		Data,		ааааааа					
		Ora,		ааааааа					
		ElementoDiS		ааааааа					
		istemaDatab		aaaaa",					
		ase.							
				Cognom					
				е					
				Maestro					
				: "Pirlo"					

	Cognome Maestro: stringa > 50 caratteri	Cognome Maestro stringa > 50 caratteri [ERROR], Validi: Data, Ora, Nome Maestro, ElementoDiS istemaDatab ase.	Lo studente ha inserito il maestro per la disdetta della lezione.	"24/06/ 2024", Ora: "18:00" } { Nome Maestro : "Andrea ", Cognom e Maestro : "Pirlooo ooooo ooooo ooooo ooooo ooooo ooooo oooo	Cogno me maestr o non valido	Il sistema chiede di inserire nuovam ente il cognom e del maestro	Cognom e maestro non valido	Il sistema chiede di inserire nuovame nte il cognome del maestro	PASS
4 Data con Data con Lo { Data II Data II PASS formato formato non studente Nome inserita sistema sistema							Data		PASS

	non valido.	valido [ERROR], Validi: Maestro, Ora, ElementoDiS istemaDatab ase.	vuole effettuare la disdetta.	Maestro : "Mario", Cognom e Maestro : "Rossi", Data: "24/06/ 24", Ora: "18:00" }	non valida. Riprov are	chiede nuovam ente di inserire la data.	Riprovar e	chiede nuovame nte di inserire la data	
5	Ora con formato non valido.	Ora con formato non valido [ERROR], Validi: Maestro, Data, ElementoDiS istemaDatab ase.	Lo studente vuole effettuare la disdetta.	{ Nome Maestro : "Mario", Cognom e Maestro : "Rossi", Data: "24/06/ 2024", Ora: "18" }	Non ci sono Lezioni per l'orario scelto. Le Lezioni inizian o solo alle 17:00, alle 18:00, alle 19:00 oppure alle 20:00. Riprov are	Il sistema chiede nuovam ente di inserire l'ora.	Non ci sono Lezioni per l'orario scelto. Le Lezioni iniziano solo alle 17:00, alle 18:00, alle 19:00 oppure alle 20:00. Riprovar e	Il sistema chiede nuovame nte di inserire l'ora.	PASS

6	Ora non	Ora non	Lo	{	Non ci	II	Non ci	II	PASS
	compresa	compresa	studente	ા Nome	sono	'' sistema	sono	" sistema	1,733
	tra le	tra le 17.00	vuole	Maestro	Lezioni	chiede	Lezioni	chiede	
	17.00 e le	e le 21.00	effettuare	·	per	nuovam	per	nuovame	
	21.00.	[ERROR],	la disdetta.	"Mario"	l'orario	ente di	l'orario	nte di	
	21.00.	[ERROR],	la disactta.	IVIAIIO	scelto.	inserire	scelto.	inserire	
		Validi:		Cognom	Le	l'ora.	sceno.	l'ora.	
		valiui.		e	Lezioni	i ora.	Le	TOTA.	
		Maastro		_	inizian		Lezioni		
		Maestro,		Maestro					
		Data,		(Dana://	o solo		iniziano		
		ElementoDiS		"Rossi",	alle		solo alle		
		istemaDatab		Data	17:00,		17:00,		
		ase.		Data:	alle		alle		
				"24/06/	18:00,		18:00,		
				2024",	alle		alle		
					19:00		19:00		
				Ora:	oppure		oppure		
				"14.00"	alle		alle		
					20:00.		20:00.		
				}	Riprov				
					are		Riprovar		
							e		
7	Lezione	ElementoDiS	La lezione	{	Lezion	11	Lezione	II	PASS
	non	istemaDatab	non è	Nome	e non	sistema	non	sistema	
	presente	ase non	disponibile	Maestro	trovata	non può	trovata.	non può	
	nel	valido		l :		effettuar		effettuar	
	database	[ERROR],		"Stefan		e la		e la	
	[ERROR].			o",		disdetta		disdetta	
				,		per la		per la	
		Validi:		Cognom		lezione		lezione	
		valial.		e		richiesta		richiesta.	
		Maestro,		Maestro		Ticinesta		Ticinesta.	
				: "Pioli",		•			
		Data, Ora.		. FIUII ,					
		Ola.		Data					
				Data:					
				"26/06/					
				2024",					
				Ora					
	1	1	ĺ	Ora:	1				
I				((40.00"					
				"19:00"					
				"19:00" }					

8	Lezione	ElementoDiS	La lezione	{	Lezion	II	Lezione	II	PASS
	non	istemaDatab	non è	Nome	e non	sistema	non	sistema	
	prenotata	ase non	stata	Maestro	trovata	non può	trovata.	non può	
	[ERROR]	valido	prenotata.	:		effettuar		effettuar	
		[ERROR],		"Stefan		e la		e la	
				o",		disdetta		disdetta	
		Validi:				per la		per la	
				Cognom		lezione		lezione	
		Maestro,		е		richiesta		richiesta.	
		Ora,		Maestro					
		Data.		: "Pioli",					
				Data:					
				"26/06/					
				2024",					
				Ora:					
				"19:00"					
				}					

FUNZIONALITÀ "Inserisci Giorno"

Test Case ID	Descrizi one	Classi di equivalenza coperte	Pre- condizioni	Input	Output Attesi	Post- condizioni Attese	Output Ottenu ti	Post- condizi oni Ottenut e	Esito (FAIL, PASS)
1	Tutti gli input sono validi.	Data con formato valido. Elementi di sistema database validi.	Il database è inizializzat o correttame nte. Il maestro ha effettuato l'accesso.	{ Data: "09/07 /2024" }	Un nuovo giorno di disponibi lità è stato inserito con successo	Il giorno di disponibili tà è stato inserito correttam ente.	Un nuovo giorno di disponi bilità è stato inserito con success o.	Il giorno di disponi bilità è stato inserito corretta mente.	PASS
2	Data con formato non valido.	Data con formato non valido [ERROR]. Elementi di sistema database validi.	Il database è inizializzat o correttame nte. Il maestro ha effettuato l'accesso.	{ Data: "09/07 /24" }	Data inserita non esistente oppure formato errato. Riprovar e	Il sistema chiede di inserire nuovame nte la data.	Data inserita non esistent e oppure format o errato. Riprova re	Il sistema chiede di inserire nuovam ente la data.	PASS
3	Data già present e nel databas e.	Data con formato valido. Data già presente nel database [ERROR].	Il database è inizializzat o correttame nte. Il maestro ha effettuato l'accesso.	{ Data: "09/10 /2024" }	Giorno già presente	Il caso d'uso termina notificand o la presenza del giorno di disponibili tà.	Giorno già present e.	Il caso d'uso termina notifica ndo la presenz a del giorno di disponi bilità.	PASS

iborato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
borato di Ingegneria del Software - T18 Team: SE(A)^2								
	aborato di Ir	ngegneria del	l Software -	T18		Tear	n: SE(A)	^2

FUNZIONALITÀ "Modifica Giorno"

Test	Descrizi	Classi di	Pre-	Input	Output	Post-	Output	Post-	Esito
Case	one	equivalenza	condizioni	-	Attesi	condizioni	Ottenuti	condizioni	(FAIL,
ID		coperte				Attese		Ottenute	PASS)
1	Tutti gli input sono validi.	Data da modificare con formato valido. Nuova data di disponibilità con formato valido. Elementi di sistema database validi.	II database è inizializzat o correttam ente. Il maestro ha effettuato l'accesso.	{ Data da modifi care: "22/05 /2025" , Nuova data: "25/05 /2025" }	Giorno di disponi bilità modific ato con success o!	La data del giorno di disponibili tà selezionat	Giorno di disponib ilità modifica to con successo !	La data del giorno di disponibili tà selezionat o è stata modificata con successo.	PASS
2	Data da modific are con format o non valido.	Data da modificare con formato non valido [ERROR]. Nuova data di disponibilità con formato valido. Elementi di sistema database validi.	II database è inizializzat o correttam ente. II maestro ha effettuato l'accesso.	{ Data da modifi care: "22- 05- 2025", Nuova data: "25/05 /2025" }	Data inserita non esisten te oppure format o errato. Riprova re	Il sistema chiede di inserire nuovamen te la data da modificare	Data inserita non esistent e oppure formato errato. Riprovar e	Il sistema chiede di inserire nuovamen te la data da modificar e.	PASS
3	Nuova data di disponi bilità con format	Data da modificare con formato valido. Nuova data di disponibilità con	II database è inizializzat o	{ Data da modifi care:	Data inserita non esisten te oppure	Il sistema chiede di reinserire la nuova data di	Data inserita non esistent e oppure	Il sistema chiede di reinserire la nuova data di	PASS

	o non valido.	formato non valido [ERROR]. Elementi di sistema database validi.	correttam ente. Il maestro ha effettuato l'accesso.	"22/05 /2025" , Nuova data: "25- 05- 2025"	format o errato. Riprova re	disponibili tà.	formato errato. Riprovar e	disponibili tà.	
4	Data da	Data da	II	}	Giorno	Il sistema	Giorno	Il sistema	PASS
	modific are non present e nel databas e.	modificare con formato valido. Nuova data di disponibilità con formato valido. Data da modificare non presente nel database [ERROR].	database è inizializzat o correttam ente. Il maestro ha effettuato l'accesso.	Data da modifi care: "24/05 /2025" Nuova data: "25/05 /2025" }	indicat o non present e nel sistem a.	chiede di inserire nuovamen te la data da modificare	indicato non presente nel sistema.	chiede di inserire nuovamen te la data da modificar e.	
5	Nuova data di disponi bilità già present e nel databas e.	Data da modificare con formato valido. Nuova data di disponibilità con formato valido. Nuova data di disponibilità già presente nel database [ERROR].	Il database è inizializzat o correttam ente. Il maestro ha effettuato l'accesso.	{ Data da modifi care: "22/05 /2025" , Nuova data: "22/05 /2025" }	Nuovo giorno indicat o già present e nel sistem a.	Il sistema chiede di reinserire la nuova data di disponibili tà.	Nuovo giorno indicato già presente nel sistema.	Il sistema chiede di reinserire la nuova data di disponibili tà.	PASS

