

Logopedia System

Ingegneria del Software a.a. 2021-2022
Corso di Laurea in Informatica

Realizzato da

Ciano Nicola 727951 ITPS n.ciano@studenti.uniba.it
Daniele Alessandro 708814 ITPS a.daniele12@studenti.uniba.it
Lin Antonio 716778 ITPS a.lin@studenti.uniba.it

Indice

1.	SPRINT REPORT	3
2.1	SPRINT BACKLOG	4
2.2	PRODUCT REQUIREMENT SPECIFICATION	6
2.2.1	<i>Diagramma dei Casi d'uso</i>	6
2.2.2	<i>Scenari</i>	7
2.2.3	<i>Altro</i>	29
2.3	SYSTEM ARCHITECTURE	30
2.3.1	<i>Diagramma delle Componenti</i>	30
2.3.2	<i>Specifica delle componenti</i>	30
2.3.3	<i>Specifica delle interfacce</i>	31
2.4	DETAILED PRODUCT DESIGN	40
2.4.1	<i>Diagramma delle Classi</i>	40
2.4.2	<i>Specifiche delle Classi</i>	41
2.4.3	<i>Diagrammi di Sequenza</i>	45
2.5	DATA MODELING AND DESIGN	62
2.5.1	<i>Modello logico del Database</i>	63
2.5.2	<i>Struttura fisica del Database</i>	63
3.	GLOSSARIO	65
3.1	ACRONIMI	65
3.2	DEFINIZIONI	65



SPRINT REPORT N. 5

LOGOPEDIA SYSTEM



SPRINT REPORT

1.1 Sprint Backlog

Tabella di riepilogo che indica, per ognuno degli Sprint successivi allo Sprint n.0, la lista degli item del Product Backlog, evidenziando quelli che verranno implementati nell'ambito dello sprint corrente unitamente ad una descrizione esplicativa.

Per semplificare l'esposizione e salvaguardare la tracciabilità tra semilavorati si è proceduto alle seguenti assunzioni:

- All'interno di uno Sprint sono implementati un sottoinsieme di item tra quelli specificati nel Product Backlog
- Lo Sprint Backlog relativo allo sprint corrente contiene pertanto l'insieme degli item del Product Backlog in corso di implementazione
- Gli Item funzionali, ovvero le User Stories dovranno essere tracciabili uno a uno, auspicabilmente seppur non necessariamente, con i casi d'uso
- Ad ogni caso d'uso dovrà essere associato uno scenario di base più gli eventuali scenari alternativi. Lo scenario in prima istanza viene redatto a partire dalla specifica della User Story riportata nel Product Backlog
- Ad ogni caso d'uso dovrà essere associato un diagramma di sequenza.

Ogni sprint deve necessariamente produrre in output del codice funzionante. L'unica eccezione è rappresentata dallo Sprint n°0 che deve essere utilizzato per disegnare la macro architettura del sistema con le sue componenti e le sue interfacce, e che sarà utilizzata come roadmap per gli sprint successivi andando a chiarire dove si colloca quanto realizzato in ciascuno di essi.

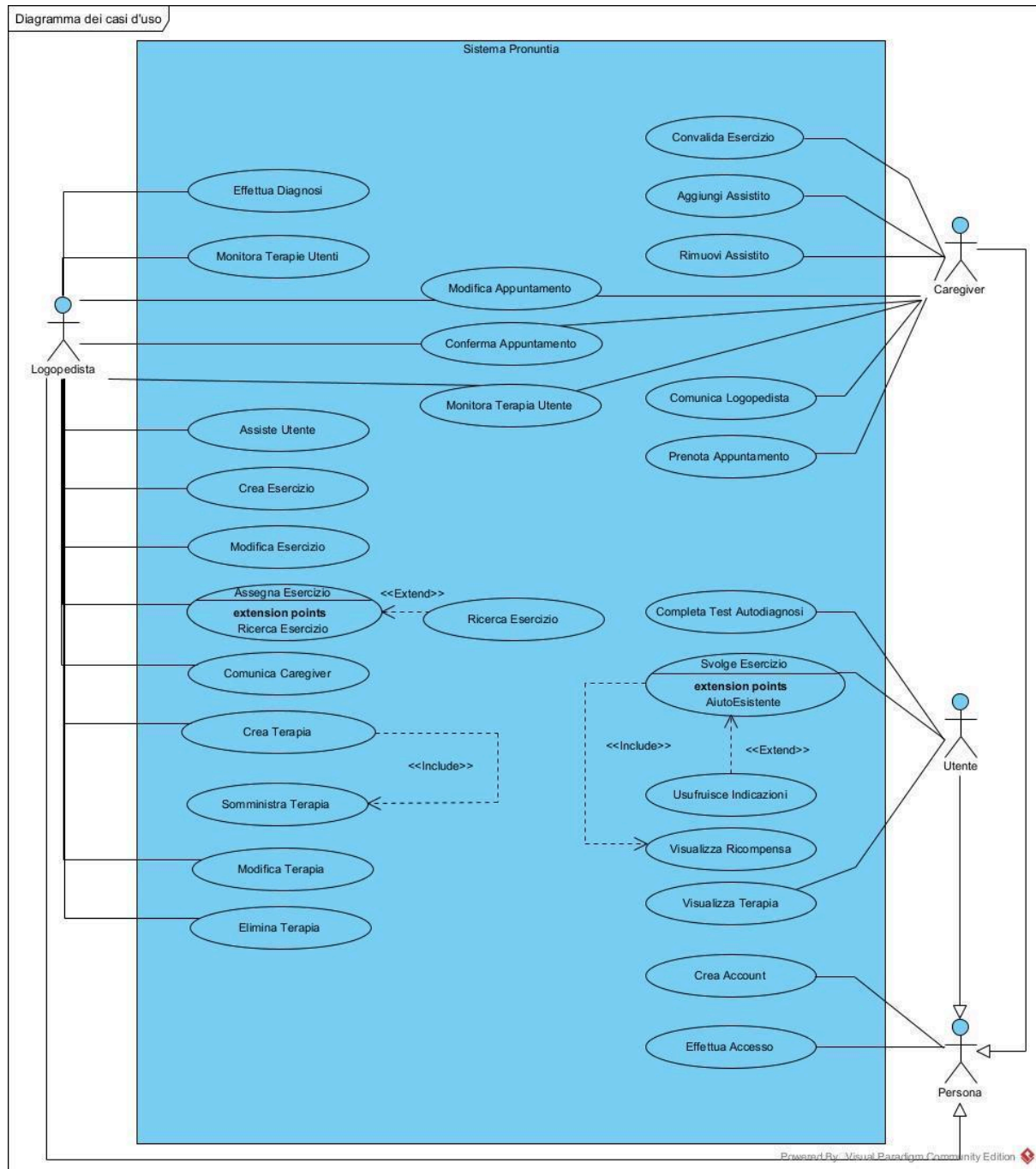
ID Caso d'uso	Codice Item	Numero Sprint	Note
PE.01	IF-23	Sprint 1	Il caso d'uso Crea Account è stato realizzato mediante il metodo CreaUtente
PE.01	IF-24	Sprint 1	Il caso d'uso Crea Account è stato realizzato mediante il metodo CreaCaregiver
PE.01	IF-25	Sprint 1	Il caso d'uso Crea Account è stato realizzato mediante il metodo CreaLogopedista
PE.01.1	IF-23	Sprint 1	
PE.01.1	IF-24	Sprint 1	



PE.01.1	IF-25	Sprint 1	
PE.02	IF-26	Sprint 1	
PE.02.1	IF-26	Sprint 1	
PE.02.2	IF-26	Sprint 1	
LO.05	IF-6	Sprint 2	Abbiamo realizzato la creazione di tutte e tre le tipologie di esercizio
LO.12	IF-7	Sprint 2	Abbiamo realizzato la modifica di tutte e tre le tipologie di esercizio
LO.01	IF-1	Sprint 3	
LO.02, LO.09	IF-2	Sprint 3	
LO.06	IF-5	Sprint 3	
LO.10, LO.11	IF-11	Sprint 3	
UT.05	IF-21	Sprint 3	
CA.02	IF-16	Sprint 4	
CA.03	IF-17	Sprint 4	
LO.04	IF-4	Sprint 4	
LOCA.01	IF-3, IF-12	Sprint 4	
CA.01	IF-14	Sprint 4	
UT.01	IF-18	Sprint 4	
CA.04	IF-13	Sprint 5	
LOCA.02	IF-27	Sprint 5	
LOCA.03	IF-28	Sprint 5	
LO.08	IF-10	Sprint 5	
CA.05	IF-15	Sprint 5	
LO.07	IF-8	Sprint 5	
UT.02	IF-19	Sprint 5	
UT.04	IF-22	Sprint 5	

1.2 Product Requirement Specification

1.2.1 Diagramma dei Casi d'uso





1.2.2 Scenari

Caso d'uso	Effettua Diagnosi
ID	LO.01
Breve descrizione	Il Logopedista effettua una diagnosi adatta per l'utente.
Attori primari	Logopedista
Attori secondari	Nessuno
Precondizioni	Il Logopedista è stato autenticato dal sistema L'utente non deve aver già ricevuto una diagnosi
Sequenza principale degli eventi	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando il Logopedista seleziona la voce per effettuare la diagnosi 2. Il sistema mostra eventualmente il disturbo riscontrato dall'esito del test autodiagnostico per avere un'anteprima del possibile problema 3. Il Logopedista, in base alla propria valutazione, formula un'opportuna diagnosi 4. Il sistema notifica al Logopedista l'esito dell'operazione
Post-condizioni	Il sistema aggiunge quella determinata diagnosi all'utente
Sequenza alternativa degli eventi	

Caso d'uso	Somministra Terapia
ID	LO.02
Breve descrizione	Il Logopedista somministra la terapia per l'utente.
Attori primari	Logopedista

Attori secondari	Nessuno
Precondizioni	<p>Il Logopedista è stato autenticato dal sistema</p> <p>L'utente deve avere ricevuto una diagnosi</p> <p>L'utente non deve avere altre terapie in corso</p>
Sequenza principale degli eventi	<ol style="list-style-type: none"> 1. Il logopedista inserisce l'utente a cui somministrare la terapia 2. Il sistema somministra la terapia all'utente
Post-condizioni	La terapia è stata somministrata all'utente
Sequenza alternativa degli eventi	

Caso d'uso	Monitora Terapia Utente
ID	LOCA.01
Breve descrizione	Il sistema deve permettere al Logopedista e al Caregiver di conoscere lo stato di avanzamento della terapia dell'utente
Attori primari	Logopedista, Caregiver
Attori secondari	Nessuno
Precondizioni	<p>Lo stakeholder (Logopedista o Caregiver) è stata autenticato dal sistema</p> <p>L'utente deve aver una terapia somministrata dal logopedista</p>

Sequenza principale degli eventi	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando il Logopedista o il Caregiver seleziona "Monitora Terapie" con la lista delle terapie 2. Selezionano la terapia da monitorare 3. Il sistema mostra le informazioni relative allo stato di avanzamento della terapia
Post-condizioni	Nessuna
Sequenza alternativa degli eventi	

Caso d'uso	Monitora Terapie Utenti
ID	LO.04
Breve descrizione	Il sistema deve permettere al Logopedista di avere un'anteprima generale delle terapie.
Attori primari	Logopedista
Attori secondari	Nessuno
Precondizioni	Il Logopedista è stato autenticato dal sistema
Sequenza principale degli eventi	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando il Logopedista seleziona "Monitora Terapie" 2. Il sistema mostra un'anteprima generale delle terapie mediante una lista
Post-condizioni	Nessuna
Sequenza alternativa degli eventi	

Caso d'uso	Crea Esercizio
ID	LO.05
Breve descrizione	Il Logopedista crea un nuovo esercizio per le terapie.
Attori primari	Logopedista
Attori secondari	Nessuno
Precondizioni	Il Logopedista è stato autenticato dal sistema
Sequenza principale degli eventi	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando il Logopedista seleziona "Crea Esercizio" 2. Se il Logopedista seleziona "Denominazione Immagini" <ol style="list-style-type: none"> 2.1. Il logopedista inserisce i dati relativi al nuovo esercizio 2.2. Il sistema crea l'esercizio con denominazione Immagini 3. altrimenti se il Logopedista seleziona "Coppie Minime" <ol style="list-style-type: none"> 3.1. Il logopedista inserisce i dati relativi al nuovo esercizio 3.2. Il sistema crea l'esercizio con Coppie Minime 4. altrimenti se il Logopedista seleziona "Ripetizione Parola" <ol style="list-style-type: none"> 4.1. Il logopedista inserisce i dati relativi al nuovo esercizio 4.2. Il sistema crea l'esercizio con Ripetizione Parola
Post-condizioni	L'esercizio viene aggiunto alla lista degli esercizi assegnabili
Sequenza alternativa degli eventi	

Caso d'uso	Assegna Esercizio
ID	LO.06
Breve descrizione	Il Logopedista assegna un esercizio per una determinata terapia di un utente.
Attori primari	Logopedista
Attori secondari	Nessuno
Precondizioni	<p>Il Logopedista è stato autenticato dal sistema</p> <p>L'esercizio deve essere presente nel sistema</p> <p>La terapia deve essere stata creata e somministrata all'utente</p>
Sequenza principale degli eventi	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando il Logopedista seleziona "Assegna Esercizi" 2. Il sistema mostra la lista degli esercizi assegnabili alla terapia 3. <i>punto di estensione: Filtra esercizi</i> 4. Il logopedista seleziona uno degli esercizi della lista 5. Il logopedista inserisce il giorno della settimana in cui l'utente dovrà svolgere quel determinato esercizio 6. Il logopedista seleziona "Assegna" 7. Il sistema comunica al Logopedista l'esito dell'assegnazione dell'esercizio
Post-condizioni	L'esercizio viene assegnato alla terapia
Sequenza alternativa degli eventi	



Caso d'uso	Assiste Utente
ID	L0.07
Breve descrizione	Il sistema deve permettere al Logopedista di assistere l'utente durante l'esecuzione dell'esercizio.
Attori primari	Logopedista
Attori secondari	Nessuno
Precondizioni	Il Logopedista è stato autenticato dal sistema L'esercizio deve essere presente nel sistema
Sequenza principale degli eventi	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando il Logopedista seleziona "Aggiungi Aiuto" 2. Il sistema mostra l'esercizio che l'utente deve svolgere 3. Il logopedista aggiunge un suggerimento per il corretto svolgimento dell'esercizio
Post-condizioni	Il sistema durante lo svolgimento dell'esercizio da parte dell'utente mostrerà il suggerimento del logopedista
Sequenza alternativa degli eventi	

Caso d'uso	Comunica Caregiver
ID	L0.08
Breve descrizione	Il sistema deve consentire al Logopedista di comunicare con il Caregiver per supportarlo con la terapia
Attori primari	Logopedista
Attori secondari	Nessuno
Precondizioni	Il Logopedista è stato autenticato dal sistema

	Il caregiver deve essere presente nel sistema
Sequenza principale degli eventi	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando il Logopedista clicca su "Comunica con un tuo Caregiver" 2. Il Logopedista inserisce i dati necessari per la comunicazione 3. Il Logopedista clicca su "Invia" 4. Il sistema mostra un messaggio di invio email eseguito con successo
Post-condizioni	Il sistema inoltra la e-mail al Caregiver
Sequenza alternativa degli eventi	

Caso d'uso	Crea Terapia
ID	LO.09
Breve descrizione	Il logopedista crea una terapia specifica per un determinato utente
Attori primari	Logopedista
Attori secondari	Nessuno
Precondizioni	Il Logopedista è stato autenticato dal sistema
Sequenza principale degli eventi	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando il Logopedista seleziona "Crea Terapia" 2. Il logopedista inserisce i dati richiesti per la creazione della terapia 3. <i>include(Somministra Terapia)</i> 4. Il sistema mostra la terapia creata.
Post-condizioni	Il sistema crea la terapia somministrata a quel determinato utente
Sequenza alternativa degli eventi	

Caso d'uso	Modifica Terapia
ID	LO.10
Breve descrizione	Il logopedista modifica l'insieme del materiale multimediale di una terapia
Attori primari	Logopedista
Attori secondari	Nessuno
Precondizioni	Il Logopedista è stato autenticato dal sistema La terapia deve essere presente nel sistema
Sequenza principale degli eventi	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando il Logopedista clicca su "Terapie" 2. Il sistema mostra l'elenco delle terapie somministrate 3. Il logopedista seleziona la terapia da modificare 4. Il logopedista apporta delle modifiche alla terapia aggiungendo, rimuovendo le attività da completare
Post-condizioni	Il sistema salva le modifiche ed aggiorna la terapia dell'utente
Sequenza alternativa degli eventi	

Caso d'uso	Elimina Terapia
ID	LO.11
Breve descrizione	Il logopedista elimina la terapia assegnata a un utente
Attori primari	Logopedista
Attori secondari	Nessuno
Precondizioni	Il Logopedista è stato autenticato dal sistema La terapia deve essere somministrata all'utente

Sequenza principale degli eventi	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando il Logopedista clicca su "Terapie" 2. Il sistema mostra l'elenco delle terapie somministrate 3. Il logopedista seleziona la terapia da eliminare 4. Il logopedista elimina la terapia
Post-condizioni	Il sistema elimina la terapia dall'elenco delle terapie somministrate.
Sequenza alternativa degli eventi	

Caso d'uso	Modifica Esercizio
ID	L0.12
Breve descrizione	Il logopedista modifica l'esercizio per adattarlo meglio alle esigenze e ai bisogni degli utenti.
Attori primari	Logopedista
Attori secondari	Nessuno
Precondizioni	<p>Il Logopedista è stato autenticato dal sistema</p> <p>L'esercizio deve essere presente nel sistema</p>
Sequenza principale degli eventi	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando il Logopedista seleziona "Modifica Esercizio" 2. Il sistema mostra la lista degli esercizi presenti nel sistema 3. Il logopedista seleziona uno degli esercizi mostrati nella lista 4. Il logopedista modifica i dati dell'esercizio 5. Il logopedista seleziona "Modifica"



	6. Il sistema mostra un messaggio di modifica avvenuta con successo
Post-condizioni	Il sistema salva le modifiche apportate sull'esercizio

Caso d'uso	Ricerca Esercizio
ID	L0.13
Breve descrizione	Il logopedista ricerca l'esercizio da aggiungere a una terapia assegnata di un utente
Attori primari	Logopedista
Attori secondari	Nessuno
Precondizioni	Il Logopedista è stato autenticato dal sistema Il logopedista deve aver somministrato una terapia
Sequenza principale degli eventi	<ol style="list-style-type: none"> 1. <i>punto di estensione: Filtra esercizi</i> 2. Il logopedista filtra gli esercizi mediante la ricerca di un determinato attributo dell'esercizio. 3. Il logopedista conferma la ricerca
Post-condizioni	Il sistema mostra la lista degli esercizi filtrata

Caso d'uso	Crea Account
ID	PE.01
Breve descrizione	Una persona in qualità di fruitore del sistema effettua la registrazione alla piattaforma
Attori primari	Persona



Attori secondari	Nessuno
Precondizioni	Nessuna
Sequenza principale degli eventi	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando il fruitore del sistema seleziona la voce per registrarsi 2. Il sistema richiede la selezione del ruolo assolto prima della creazione del profilo <ol style="list-style-type: none"> 2.1. L'utente seleziona la voce per registrarsi come logopedista <ol style="list-style-type: none"> 2.1.1. Il sistema mostra il form per la registrazione con il campo per l'inserimento le credenziali richieste 2.2. L'utente seleziona la voce per registrarsi come l'utente <ol style="list-style-type: none"> 2.2.1. La persona in qualità di utente inserisce le credenziali iniziali. 2.3. La persona in qualità di caregiver inserisce i propri dati per monitorare il proprio assistito. 3. Il sistema valida i dati e inoltra la richiesta di Registrazione : <ol style="list-style-type: none"> 3.1. Se l'esito della registrazione è positivo <ol style="list-style-type: none"> 3.1.1. Il sistema aggiorna la sessione e crea l'account e reindirizza l'utente nella schermata di login 3.2. Se l'esito della registrazione è negativo <ol style="list-style-type: none"> 3.2.1. Il sistema fornisce un messaggio di errore
Post-condizioni per il successo	L'utente si è registrato alla piattaforma
Post-condizioni per il fallimento	Il nome utente inserito durante la registrazione è già presente nel database e viene stampato un messaggio d'errore
Sequenza alternativa degli eventi	AccountEsistente

Caso d'uso	Crea Account:AccountEsistente
ID	PE.01.1
Breve descrizione	Il sistema informa che i dati inseriti durante la registrazione sono già esistenti
Attori primari	Persona
Attori secondari	Nessuno
Precondizioni	La persona ha inserito uno username già presente nel sistema
Sequenza principale degli eventi	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando la persona inserisce un username già presente nel sistema 2. Il sistema presenta un messaggio di errore in cui indica che l'username è già presente nel sistema. Consiglia alla persona di registrarsi con un altro username.
Post-condizioni	Nessuna
Sequenza alternativa degli eventi	Nessuna

Caso d'uso	Effettua Accesso
ID	PE.02
Breve descrizione	La persona in qualità di caregiver o utente si autentica con le proprie credenziali
Attori primari	Persona
Attori secondari	Nessuno
Precondizioni	La persona deve essere presente nel sistema

Sequenza principale degli eventi	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando la persona clicca su "Accedi" 2. La persona inserisce le credenziali di accesso. 3. La persona clicca su login. 4. Il sistema valida i dati e inoltra la richiesta di accesso <ol style="list-style-type: none"> 4.1. Se la richiesta da esito positivo verrà reindirizzato alla pagina iniziale del proprio profilo 4.2. Se la richiesta da esito negativo il sistema fornisce un messaggio di errore
Post-condizioni	Il sistema autentica l'utente.
Sequenza alternativa degli eventi	UsernameNonValido PasswordNonValida

Caso d'uso	Effettua Accesso:UsernameNonValido
ID	PE.02.1
Breve descrizione	Il sistema informa l'utente che l'email o il nome utente inserito non è valido
Attori primari	Persona
Attori secondari	Nessuno
Precondizioni	La persona ha inserito un username non valido
Sequenza principale degli eventi	<ol style="list-style-type: none"> 1. Il sistema presenta un messaggio di errore in cui indica che l'username non è valido o che tale dato non è presente nel sistema degli utenti, caregiver o logopedisti registrati.
Post-condizioni	Nessuna
Sequenza alternativa degli eventi	La sequenza alternativa degli eventi inizia dopo il passo 1 della sequenza principale degli eventi del caso d'uso principale "Effettua Accesso"

Caso d'uso	Effettua Accesso:PasswordNonValida
ID	PE.02.2
Breve descrizione	Il sistema informa l'utente che la password non è valida
Attori primari	Persona
Attori secondari	Nessuno
Precondizioni	La persona ha inserito una password non corretta
Sequenza principale degli eventi	1. Il sistema presenta un messaggio di errore in cui indica che la password non è valida.
Post-condizioni	Nessuna
Sequenza alternativa degli eventi	La sequenza alternativa degli eventi inizia dopo il passo 1 della sequenza principale degli eventi del caso d'uso principale "Effettua Accesso"

Caso d'uso	Svolgere esercizio
ID	UT.01
Breve descrizione	Il sistema consente all'utente di svolgere l'esercizio
Attori primari	Utente
Attori secondari	Nessuno
Precondizioni	L'utente è stato autenticato dal sistema Il logopedista ha somministrato degli esercizi da svolgere all'utente
Sequenza principale degli eventi	1. Il caso d'uso inizia quando l'utente clicca su "Svolgi Esercizi" 2. Il sistema mostra la lista degli esercizi giornalieri da svolgere 3. L'utente seleziona l'esercizio da svolgere <i>punto di estensione : AiutoEsistente</i> 4. L'utente svolge l'esercizio 5. <i>include(Visualizza Ricompensa)</i>

Post-condizioni	L'esercizio è stato contrassegnato come svolto
Sequenza alternativa degli eventi	Nessuna

Caso d'uso	Usufruisce indicazioni
ID	UT.02
Breve descrizione	Il sistema deve dare indicazioni all'utente per la corretta esecuzione dell'esercizio e rilevazione di errori
Attori primari	Utente
Attori secondari	Nessuno
Precondizioni	L'utente è stato autenticato dal sistema L'utente deve aver avviato l'esercizio L'aiuto del logopedista deve essere impostato all'interno dell'esercizio
Sequenza principale degli eventi	<i>punto di estensione: AiutoEsistente</i> 1. Il sistema durante lo svolgimento dell'esercizio mostrerà l'aiuto del logopedista nella parte iniziale dell'esercizio da svolgere.
Post-condizioni	Nessuna
Sequenza alternativa degli eventi	Nessuna

Caso d'uso	Visualizza Ricompensa
ID	UT.03
Breve descrizione	Il sistema deve permettere all'utente di visualizzare la ricompensa ottenuta dopo il completamento dell'esercizio
Attori primari	Utente
Attori secondari	
Precondizioni	L'utente è stato autenticato dal sistema L'esercizio dell'utente è stato svolto
Sequenza principale degli eventi	1. Il sistema visualizza una ricompensa all'utente
Post-condizioni	Nessuna
Sequenza alternativa degli eventi	Nessuna

Caso d'uso	Visualizza Terapia
ID	UT.04
Breve descrizione	Il sistema visualizza lo stato di avanzamento della terapia all'utente
Attori primari	Utente
Attori secondari	Nessuno
Precondizioni	L'utente è stato autenticato dal sistema E' stata assegnata la terapia all'utente

Sequenza principale degli eventi	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando l'Utente clicca su "Terapia" 2. Il sistema mostra all'utente la terapia, con i corrispettivi dati
Post-condizioni	Nessuna
Sequenza alternativa degli eventi	Nessuna

Caso d'uso	Completa Test Autodiagnostico
ID	UT.05
Breve descrizione	L'utente completa il test autodiagnostico per determinare se necessita di una terapia logopedica.
Attori primari	Utente
Attori secondari	Nessuno
Precondizioni	Nessuna
Sequenza principale degli eventi	<ol style="list-style-type: none"> 1. Al momento dell'avvio del sistema oltre alle informazioni per l'accesso viene presentato un link per effettuare un test autodiagnostico 2. L'utente risponde e completa il test autodiagnosi 3. Il sistema mostra i pulsanti per salvare l'esito del test o richiedere assistenza per risolvere il proprio problema fonetico.
Post-condizioni	Il sistema fornisce una prima verifica delle risposte del test di autodiagnosi e indica almeno l'oggetto del problema e fornisce il test al logopedista, se l'utente lo richiede, per formulare la diagnosi dettagliata

Sequenza alternativa degli eventi	
--	--

Caso d'uso	Convalida esercizio
ID	CA.01
Breve descrizione	Il caregiver deve convalidare l'esercizio svolto dall'utente per l'esito dell'esercizio
Attori primari	Caregiver
Attori secondari	Nessuno
Precondizioni	Il caregiver è stato autenticato dal sistema L'esercizio è stato svolto dall'utente
Sequenza principale degli eventi	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando il Caregiver clicca su "Convalida Esercizi" 2. Il sistema mostra la lista degli esercizi da convalidare 3. Il caregiver seleziona l'esercizio da convalidare 4. Il caregiver convalida l'esercizio
Post-condizioni	Il sistema memorizza l'esito di quel determinato esercizio Il sistema aggiorna lo stato della terapia dell'utente
Sequenza alternativa degli eventi	Nessuna

Caso d'uso	Aggiungi assistito
ID	CA.02
Breve descrizione	Il caregiver aggiunge un utente alla lista dei suoi assistiti



Attori primari	Caregiver
Attori secondari	Nessuno
Precondizioni	Il caregiver è stato autenticato dal sistema L'utente deve essere presente nel sistema
Sequenza principale degli eventi	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando il Caregiver seleziona "Aggiungi Assistito" 2. Il sistema mostra la lista degli utenti non aventi ancora un caregiver 3. Il caregiver seleziona uno degli utenti della lista 4. Il caregiver clicca su "Aggiungi" 5. Il sistema comunica al Caregiver l'esito dell'aggiunta dell'utente
Post-condizioni	Il sistema aggiunge l'utente alla lista degli assistiti del Caregiver
Sequenza alternativa degli eventi	Nessuna

Caso d'uso	Rimuovi assistito
ID	CA.03
Breve descrizione	Il caregiver rimuove un utente dalla lista dei suoi assistiti
Attori primari	Caregiver
Attori secondari	Nessuno
Precondizioni	Il caregiver è stato autenticato dal sistema L'utente è un assistito del caregiver

Sequenza principale degli eventi	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando il caregiver clicca su "Assistiti" 2. Il sistema mostra la lista degli assistiti del rispettivo Caregiver 3. Il caregiver clicca su "Rimuovi Assistito". 4. Il caregiver conferma l'eliminazione dell'utente.
Post-condizioni	L'utente è stato rimosso dalla lista degli assistiti del caregiver
Sequenza alternativa degli eventi	Nessuna

Caso d'uso	Prenota Appuntamento
ID	CA.04
Breve descrizione	Il sistema deve consentire al caregiver di prenotare un appuntamento dalla logopedista
Attori primari	Caregiver
Attori secondari	
Precondizioni	<p>Il caregiver è stato autenticato dal sistema</p> <p>Il caregiver deve avere almeno un assistito</p>
Sequenza principale degli eventi	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando il Caregiver clicca su "Appuntamenti" 2. Il Caregiver clicca su "Prenota Appuntamento" Il caregiver seleziona i dati e la data per la prenotazione dell'appuntamento 3. Il caregiver clicca su "Prenota Appuntamento" 4. Il sistema mostra un messaggio di prenotazione prenotata con successo

Post-condizioni	Il sistema inserisce l'appuntamento tra quelli in attesa di conferma da parte del Logopedista
Sequenza alternativa degli eventi	Nessuna

Caso d'uso	Comunica Logopedista
ID	CA.05
Breve descrizione	Il sistema deve consentire al caregiver di comunicare con il Logopedista per ricevere un supporto riguardante la terapia
Attori primari	Caregiver
Attori secondari	Nessuno
Precondizioni	Il caregiver è stato autenticato dal sistema Il logopedista deve essere presente nel sistema
Sequenza principale degli eventi	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando il Caregiver clicca su "Comunica con il tuo Logopedista" 2. Il caregiver inserisce i dati necessari per la comunicazione 3. Il caregiver clicca su "Invia" 4. Il sistema mostra un messaggio di invio email eseguito con successo
Post-condizioni	Il sistema inoltra la e-mail al Logopedista
Sequenza alternativa degli eventi	Nessuna

Caso d'uso	Conferma Appuntamento
ID	LOCA.02
Breve descrizione	Il sistema deve consentire al Logopedista ed al Caregiver di

	confermare l'appuntamento in quella determinata data ed orario
Attori primari	Logopedista, Caregiver
Attori secondari	Nessuno
Precondizioni	<p>La persona (Logopedista o Caregiver) è stata autenticata dal sistema</p> <p>L'appuntamento deve essere stato creato</p> <p>L'appuntamento è in attesa di conferma esclusivamente del corrispettivo stakeholder</p>
Sequenza principale degli eventi	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando lo stakeholder clicca su "Appuntamenti in attesa" 2. Il sistema mostra la lista degli appuntamenti che sono in attesa di essere confermati 3. Lo stakeholder clicca su "Conferma" in uno degli appuntamenti della lista 4. Il sistema visualizza i dati dell'appuntamento appena confermato
Post-condizioni	Il sistema inserisce l'appuntamento tra quelli confermati
Sequenza alternativa degli eventi	Nessuna

Caso d'uso	Modifica Appuntamento
ID	LOCA.03
Breve descrizione	Il sistema deve consentire al Logopedista ed al Caregiver di spostare l'appuntamento non ancora confermato, in un'altra data ed orario
Attori primari	Logopedista, Caregiver
Attori secondari	Nessuno
Precondizioni	La persona (Logopedista o Caregiver) è stata autenticata dal sistema

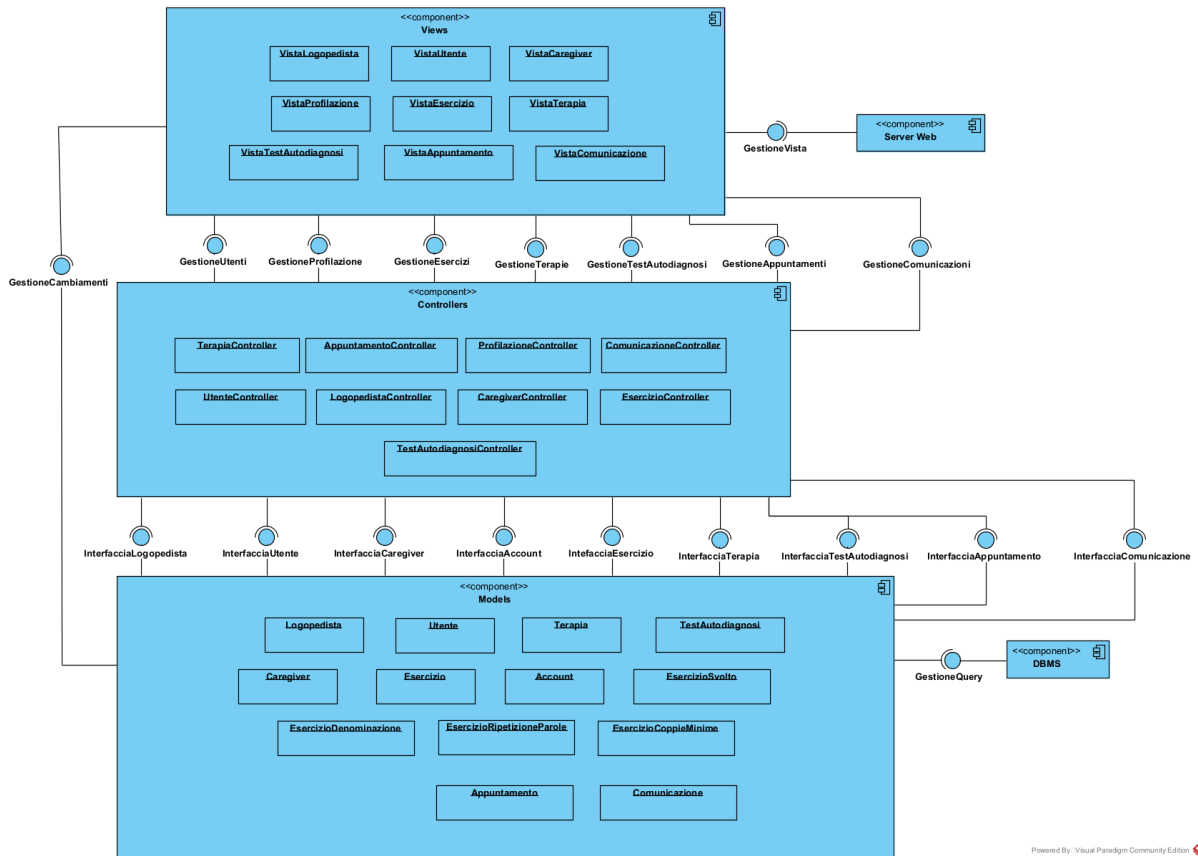


	<p>L'appuntamento deve essere stato creato</p> <p>L'appuntamento non è ancora stato confermato</p> <p>L'appuntamento è in attesa di conferma esclusivamente del corrispettivo stakeholder</p>
Sequenza principale degli eventi	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando lo stakeholder clicca su "Appuntamenti in attesa" 2. Il sistema mostra la lista degli appuntamenti che sono in attesa di essere conferma 3. Lo stakeholder clicca su "Modifica data appuntamento" 4. Lo stakeholder inserisce la nuova data per l'appuntamento 5. Il sistema mostra un messaggio di modifica avvenuta con successo
Post-condizioni	<p>Il sistema modifica la data dell'appuntamento ed inserisce l'appuntamento tra quelli in attesa del corrispettivo stakeholder</p>

1.2.3 Altro

1.3 System Architecture

1.3.1 Diagramma delle Componenti



1.3.2 Specifica delle componenti

- Views

Componente che si occupa della visualizzazione dei dati e gestisce l'interazione tra i vari utenti del sistema ed l'infrastruttura del sistema.

- Views:: VistaLogopedista
- Views:: VistaUtente
- Views:: VistaCaregiver
- Views:: VistaEsercizio
- Views:: VistaTerapia
- Views:: VistaTestAutodiagnosi
- Views:: VistaAppuntamento
- Views:: VistaProfilazione

- **Controllers**

Componente che si occupa della logica di controllo del sistema. Dopo aver ricevuto l'azione dell'utente da parte della componente Views, esegue delle operazioni che riguardano la componente Models, portando ad un cambiamento di stato nelle viste

- Controllers:: TerapiaController
- Controllers:: ProfilazioneController
- Controllers:: UtenteController
- Controllers:: LogopedistaController
- Controllers:: CaregiverController
- Controllers:: EsercizioController
- Controllers:: TestAutodiagnosiController
- Controllers:: AppuntamentoController

- **Models**

Componente che si occupa di gestire i dati del sistema. Quindi fornisce i metodi per interagire con essi, per poi elaborarli.

- Models:: Account
- Models:: Logopedista
- Models:: Caregiver
- Models:: Utente
- Models:: Terapia
- Models:: TestAutodiagnosi
- Models:: Esercizio
- Models:: EsercizioSvolto
- Models:: EsercizioDenominazione
- Models:: EsercizioRipetizioneParole
- Models:: EsercizioCoppieMinime
- Models:: Appuntamento

1.3.3 Specifica delle interfacce


L'interfaccia "InterfacciaLogopedista" espone le seguenti operazioni :

```
Logopedista.getListaLogopedisti()
Logopedista.getId()
Logopedista.getM_atricola()
Logopedista.getNome()
Logopedista.getCognome()
Logopedista.getEmail()
Logopedista.setId()
Logopedista.setM_atricola()
Logopedista.setNome()
Logopedista.setCognome()
Logopedista.setEmail()
Logopedista.validaM_atricola()
Logopedista.trovaM_atricola()
Logopedista.creaLogopedista()
```

InterfacciaLogopedista

L'interfaccia "InterfacciaUtente" espone le seguenti operazioni :


```
Utente.getId()
Utente.getNome()
Utente.getCognome()
Utente.getEmail()
Utente.getCodice_fiscale()
Utente.getDisturbo()
Utente.getDiagnosi()
Utente.setId()
Utente.setNome()
Utente.setCognome()
Utente.setEmail()
Utente.setCodice_fiscale()
Utente.setDisturbo()
Utente.setDiagnosi()
Utente.validaCodiceFiscale()
Utente.trovaCodiceFiscale()
Utente.trovaTerapia()
Utente.creaUtente()
Utente.trovaUtente()
Utente.modificaDiagnosi()
Utente.getListaUtentiSenzaTerapia()
Utente.trovaDisturbo()
```



InterfacciaUtente

L'interfaccia "InterfacciaCaregiver" espone le seguenti operazioni :


```
Caregiver.getId()
Caregiver.getNome()
Caregiver.getCognome()
Caregiver.getEmail()
Caregiver.getCodice_fiscale()
Caregiver.getDataAssistiti()
Caregiver.setId()
Caregiver.setNome()
Caregiver.setCognome()
Caregiver.setEmail()
Caregiver.setCodice_fiscale()
Caregiver.trovaTerapiaUtente()
Caregiver.creaCaregiver()
Caregiver.aggiungiAssistito()
Caregiver.eliminaAssistito()
Caregiver.modificaAssistito()
Caregiver.cercaAssistito()
```

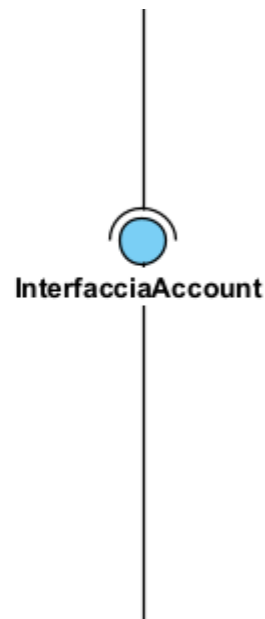


InterfacciaCaregiver

The diagram shows a vertical line with a circle at the top, representing the interface.

L'interfaccia "InterfacciaAccount" espone le seguenti operazioni :

```
Account.getId()
Account.getUsername()
Account.getPassword()
Account.getRuolo()
Account.setId()
Account.setUsername()
Account.setPassword()
Account.setRuolo()
Account.getAccount()
Account.validaAccount()
Account.validaUsername()
Account.trovaUsername()
Account.creaAccount()
Account.login()
```



InterfacciaAccount


The diagram shows a vertical line with a circle at the top, representing the interface.



L'interfaccia "InterfacciaEsercizio" espone le seguenti operazioni :

```

Esercizio.getId()
Esercizio.getNome()
Esercizio.getTipologia()
Esercizio.getDisturbo()
Esercizio.getAiuto()
Esercizio.getParola()
Esercizio.getFonema()
Esercizio.getListaEsercizi()
Esercizio.getDatiEsercizio()
Esercizio.setId()
Esercizio.setNome()
Esercizio.setTipologia()
Esercizio.setDisturbo()
Esercizio.setAiuto()
Esercizio.setParola()
Esercizio.setFonema()
Esercizio.validaNomeEsercizio()
Esercizio.trovaEserciziTerapia()
Esercizio.trovaNomeEsercizio()
Esercizio.modificaAiuto()
EsercizioCoppieMinime.getImmagine_1()
EsercizioCoppieMinime.getImmagine_2()
EsercizioCoppieMinime.setImmagine_1()
EsercizioCoppieMinime.setImmagine_2()
EsercizioCoppieMinime.getDatiEsercizio()
EsercizioCoppieMinime.creaEsercizio()
EsercizioCoppieMinime.modificaEsercizio()
EsercizioRipetizioneParole.getParola_1()
EsercizioRipetizioneParole.getParola_2()
EsercizioRipetizioneParole.getParola_3()
EsercizioRipetizioneParole.setParola_1()
EsercizioRipetizioneParole.setParola_2()
EsercizioRipetizioneParole.setParola_3()
EsercizioRipetizioneParole.getDatiEsercizio()
EsercizioRipetizioneParole.creaEsercizio()
EsercizioRipetizioneParole.modificaEsercizio()
EsercizioDenominazione.getImmagine()
EsercizioDenominazione.setImmagine()
EsercizioDenominazione.getDatiEsercizio()
EsercizioDenominazione.creaEsercizio()
EsercizioDenominazione.modificaEsercizio()
EsercizioSvolto.getId()
EsercizioSvolto.getData()
EsercizioSvolto.getRisposta()
EsercizioSvolto.getValutazione()
EsercizioSvolto.setId()
EsercizioSvolto.setData()
EsercizioSvolto.setRisposta()
EsercizioSvolto.setValutazione()
EsercizioSvolto.mostraEserciziSvolti()
EsercizioSvolto.mostraEserciziNonSvolti()
EsercizioSvolto.svolgiEsercizio()
EsercizioSvolto.convalidaEsercizio()
  
```



InterfacciaEsercizio

L'interfaccia "InterfacciaTerapia" espone le seguenti operazioni :

```

Terapia.getId()
Terapia.getNome()
Terapia.getDisturbo()
Terapia.getStato()
Terapia.setId()
Terapia.setNome()
Terapia.setDisturbo()
Terapia.setStato()
Terapia.getDatiTerapia()
Terapia.getDatiTerapie()
Terapia.getDatiTerapiaByIDUtente()
Terapia.getNomeGiornoEsercizio()
Terapia.creaTerapia()
Terapia.modificaTerapia()
Terapia.eliminaTerapia()
Terapia.assegnaEsercizio()
Terapia.rimuoviEsercizio()

```



InterfacciaTerapia

L'interfaccia "InterfacciaTestAutodiagnosi" espone le seguenti operazioni :

```

TestAutodiagnosi.getId()
TestAutodiagnosi.completaTestAutodiagnosi()
TestAutodiagnosi.getRisposta_1()
TestAutodiagnosi.getRisposta_2()
TestAutodiagnosi.getRisposta_3()
TestAutodiagnosi.getRisposta_4()
TestAutodiagnosi.setRisposta_1()
TestAutodiagnosi.setRisposta_2()
TestAutodiagnosi.setRisposta_3()
TestAutodiagnosi.setRisposta_4()

```


InterfacciaTestAutodiagnosi

L'interfaccia "InterfacciaAppuntamento" espone le seguenti operazioni :

```
Appuntamento.getId()
Appuntamento.getData()
Appuntamento.getStato()
Appuntamento.validaData()
Appuntamento.prenotaAppuntamento()
Appuntamento.confermaAppuntamento()
Appuntamento.modificaDataAppuntamento()
```


InterfacciaAppuntamento

L'interfaccia "InterfacciaComunicazione" espone le seguenti operazioni :

```
Comunicazione.getDestinatario()
Comunicazione.getOggetto()
Comunicazione.getTesto()
Comunicazione.setDestinatario()
Comunicazione.setOggetto()
Comunicazione.setTesto()
Comunicazione.comunica()
```


InterfacciaComunicazione

L'interfaccia "GestioneProfilazione" espone le seguenti operazioni:

```
ProfilazioneController.actionRegistrazione_utente()
ProfilazioneController.actionRegistrazione_logopedista()
ProfilazioneController.actionRegistrazione_caregiver()
ProfilazioneController.actionEffettua_accesso()
ProfilazioneController.actionLogout()
```


GestioneProfilazione

L'interfaccia "GestioneEsercizi" espone le seguenti operazioni:

```
EsercizioController.actionVisualizza_esercizio()  
EsercizioController.actionModifica_esercizio()  
EsercizioController.actionCrea_es_denominazione_immagini()  
EsercizioController.actionCrea_es_ripetizione_parole()  
EsercizioController.actionCrea_es_coppie_minime()  
EsercizioController.actionSvolgi_esercizio()  
EsercizioController.actionConvalida_esercizio()  
EsercizioController.actionLista_esercizi()  
EsercizioController.actionLista_esercizi_giornalieri()  
EsercizioController.actionLista_esercizi_svolti()  
EsercizioController.actionAggiungi_aiuto()  
EsercizioController.actionVisualizzaRicompenza()
```

GestioneEsercizi

L'interfaccia "GestioneTerapie" espone le seguenti operazioni:

```
TerapiaController.actionVisualizza_terapia()  
TerapiaController.actionCrea_terapia()  
TerapiaController.actionLista_terapie()  
TerapiaController.actionModifica_terapia()  
TerapiaController.actionElimina_terapia()  
TerapiaController.actionAssegna_esercizio()  
TerapiaController.actionRimuovi_esercizio()  
TerapiaController.actionMonitora_terapia_utente()  
TerapiaController.actionMonitora_terapia_utenti()
```

GestioneTerapie

L'interfaccia "GestioneTestAutodiagnosi" espone le seguenti operazioni:

```
TestAutodiagnosiController.actionCrea_test_autodiagnosi()  
TestAutodiagnosiController.actionVisualizza_test()  
TestAutodiagnosiController.actionVisualizza_pdf()
```

GestioneTestAutodiagnosi

L'interfaccia "GestioneAppuntamenti" espone le seguenti operazioni:

```
AppuntamentoController.actionPrenota_appuntamento()  
AppuntamentoController.actionModifica_data_appuntamento()  
AppuntamentoController.actionVisualizza_appuntamento()  
AppuntamentoController.actionLista_appuntamenti()  
AppuntamentoController.actionConferma_appuntamento()
```

GestioneAppuntamenti

L'interfaccia "GestioneComunicazioni" espone le seguenti operazioni:

```
ComunicazioneController.actionComunica()
```



L'interfaccia "GestioneUtenti" espone le seguenti operazioni:

```
Utente.actionIndex()  
Utente.actionVisualizza_utente()  
Utente.actionVisualizza_terapia()  
Logopedista.actionEffettua_diagnosi()  
Caregiver.actionAggiungi_assistito()  
Caregiver.actionElimina_assistito()  
Caregiver.actionModifica_assistito()  
Caregiver.actionLista_assistiti()
```

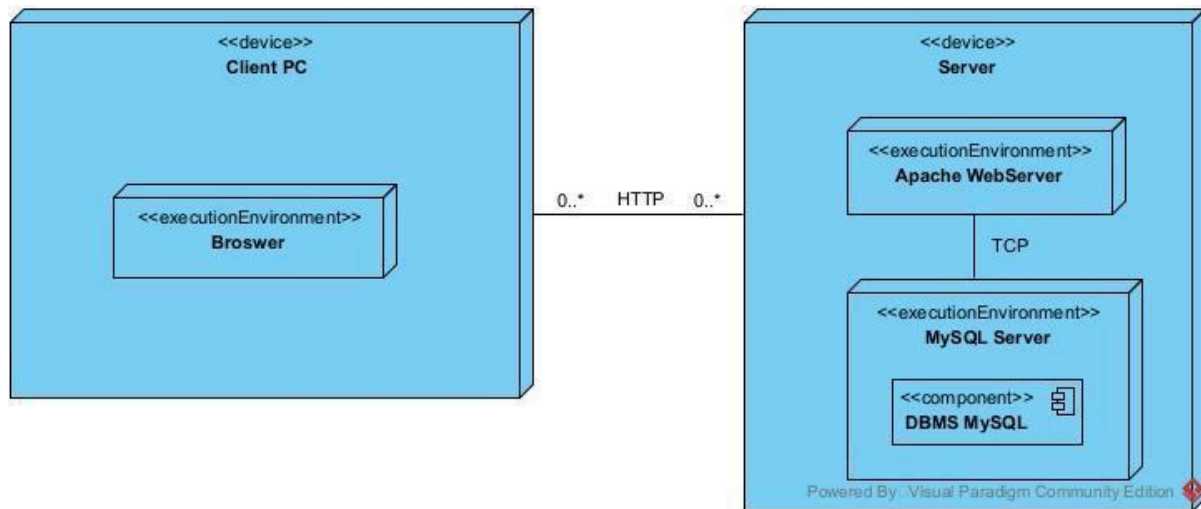


L'interfaccia "GestioneCambiamenti" espone le seguenti operazioni:

```
VistaLogopedista.notificaCambiamento()  
VistaEsercizio.notificaCambiamento()  
VistaTerapia.notificaCambiamento()  
VistaTestAutodiagnosi.notificaCambiamento()  
VistaProfilazione.notificaCambiamento()  
VistaUtente.notificaCambiamento()  
VistaCaregiver.notificaCambiamento()  
VistaAppuntamento.notificaCambiamento()  
VistaComunicazione.notificaCambiamento()  
Utente.getDatUtente()  
Caregiver.getDatAssistito()  
Esercizio.getDatEsercizio()  
EsercizioCoppieMinime.getDatEsercizio()  
EsercizioRipetizioneParole.getDatEsercizio()  
EsercizioDenominazione.getDatEsercizio()  
EsercizioSvolto.getRicompensa()  
TestAutodiagnosi.getRisposta_1()  
TestAutodiagnosi.getRisposta_2()  
TestAutodiagnosi.getRisposta_3()  
TestAutodiagnosi.getRisposta_4()  
Terapia.getDatTerapia()  
Appuntamento.getData()  
Appuntamento.getStato()
```

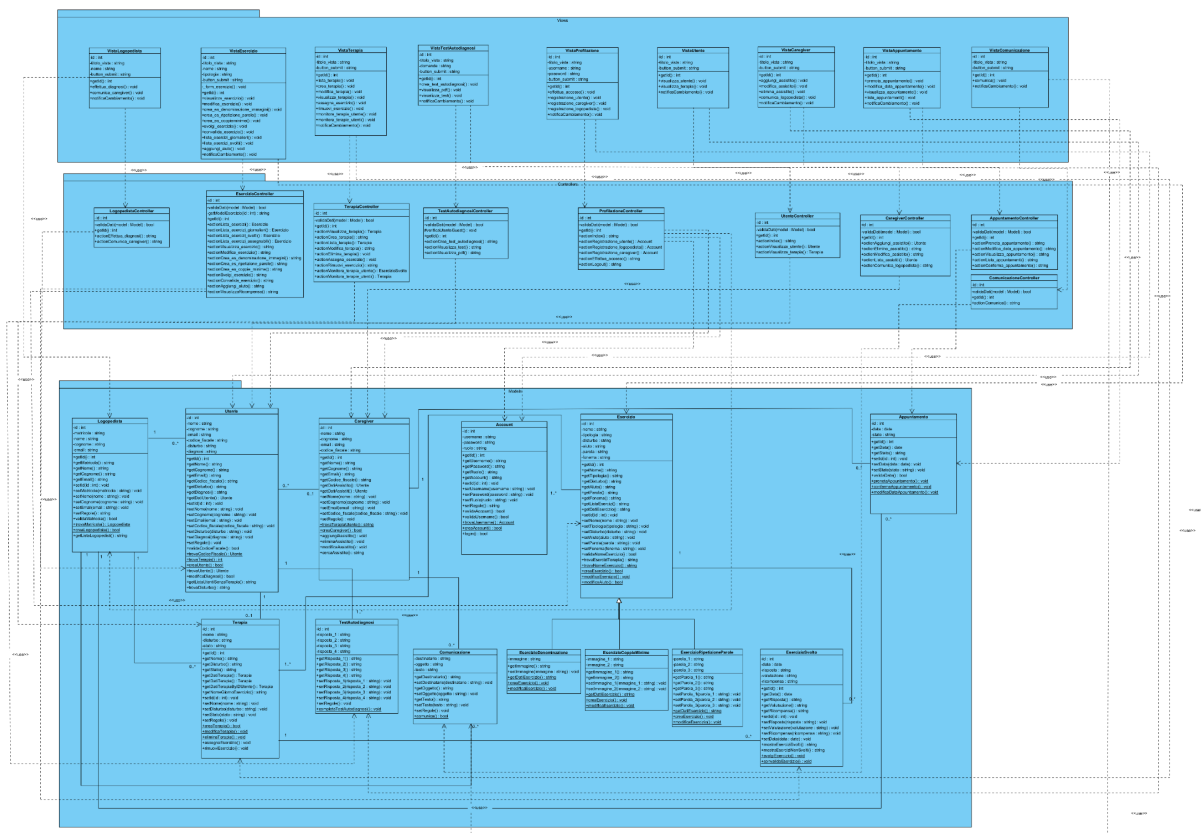
**GestioneCambiamenti**

1.3.4 Diagramma di deployment



1.4 Detailed Product Design

1.4.1 Diagramma delle Classi



1.4.2 Specifiche delle Classi

La classe `views::vistaTerapia` rappresenta l'interfaccia offerta alla logopedista per operare sulla gestione delle terapie. Le funzionalità offerte sono descritte dai metodi della classe.

La classe `views::vistaAppuntamento` rappresenta l'interfaccia offerta al caregiver per operare sulla gestione degli appuntamenti. Le funzionalità offerte sono descritte dai metodi della classe.

La classe `views::vistaCaregiver` rappresenta l'interfaccia offerta al caregiver per effettuare la gestione degli assistiti. Le funzionalità offerte sono descritte dai metodi della classe.

La classe `views::VistaEsercizio` si occupa di fornire l'interfaccia al logopedista per creare, modificare e visualizzare un esercizio; l'interfaccia all'utente per svolgere o visualizzare un esercizio e di visionare la lista degli esercizi giornalieri e quelli appena svolti; l'interfaccia al caregiver per convalidare gli esercizi svolti dall'assistito.

La classe `views::VistaTestAutodiagnosi` si occupa di fornire l'interfaccia per compilare il test di autodiagnosi e dell'interfaccia per visualizzare le risposte fornite come pdf e come contenuto web.

La classe `views::VistaLogopedista` contiene attributi e metodi che riguardano la visualizzazione dei dati e la visualizzazione delle varie funzionalità che l'entità Logopedista può effettuare. Permette a tale stakeholder di interagire con il sistema, mostrando le azioni che può svolgere, che sono principalmente l'effettuare una diagnosi ad un utente e comunicare con un caregiver. Le restanti funzionalità offerte sono descritte dai metodi della classe.

La classe `views::VistaUtente` contiene attributi e metodi che riguardano la visualizzazione dei dati e la visualizzazione delle varie funzionalità che l'entità Utente può effettuare. Permette a tale stakeholder di interagire con il sistema, mostrando le azioni che può svolgere, che sono principalmente la visualizzazione dei dati della terapia e la visualizzazione dei dati dell'utente stesso. Le restanti funzionalità offerte sono descritte dai metodi della classe.

La classe `views::VistaProfilazione` contiene attributi e metodi che riguardano la visualizzazione dei dati e la visualizzazione delle varie funzionalità che le entità del sistema possono effettuare per tutto ciò che riguarda la Profilazione. Permette ai

vari stakeholder di interagire con il sistema, mostrando le azioni che possono svolgere, che sono principalmente la registrazione, l'accesso ed il logout.

La classe `views::VistaComunicazione` contiene attributi e metodi che riguardano la gestione della comunicazione con il logopedista o il caregiver mediante i dati forniti dall'entità `Comunicazione`.

La classe `controllers::CaregiverController` gestisce le richieste per l'aggiunta e rimozione degli assistiti del caregiver nel database.

La classe `controllers::TerapiaController` gestisce le richieste per la ricerca della lista di terapie, visualizzazione terapia, monitoraggio terapie utenti e la creazione, modificazione, eliminazione della terapia nel database, inoltre essa ha anche la responsabilità di aggiungere e rimuovere esercizi dalla terapia.

La classe `controllers::AppuntamentoController` gestisce le richieste per la prenotazione dell'appuntamento, modifica dell'appuntamento e rimozione dell'appuntamento nel database.

La classe `controllers::EsercizioController` gestisce le richieste pervenute dai form delle view per la creazione, la modifica, la convalida e lo svolgimento degli esercizi. Mediante le regole definite nei vari models effettua dei controlli di validità prima dell'inoltro completo delle richieste al database.

La classe `controllers::TestAutodiagnosiController` gestisce le richieste pervenute dai form delle view per la creazione del test di autodiagnosi. Verifica se l'utente che sta completando il test è un utente loggato o un utente guest. Mediante le regole definite nella classe `TestAutodiagnosi` effettua dei controlli di validità prima dell'inoltro completo delle richieste al database.

La classe `controllers::LogopedistaController` contiene attributi e metodi che riguardano la gestione delle richieste di input da parte del Logopedista. Permette al sistema di elaborare le azioni in input per tale entità, che sono principalmente l'effettuare una diagnosi e comunicare con un caregiver. Le restanti funzionalità offerte sono descritte dai metodi della classe.

La classe `controllers::UtenteController` contiene attributi e metodi che riguardano la gestione delle richieste di input da parte dell'Utente. Permette al sistema di elaborare le azioni in input per tale entità, che sono principalmente la

visualizzazione dei dati della terapia e la visualizzazione dei dati dell'utente stesso. Le restanti funzionalità offerte sono descritte dai metodi della classe.

La classe `controllers::ProfilazioneController` contiene attributi e metodi che riguardano la gestione delle richieste di input per tutto ciò che riguarda la Profilazione. Permette al sistema di elaborare le azioni in input delle varie entità del sistema, che possono principalmente effettuare la registrazione, l'accesso ed il logout. Le restanti funzionalità offerte sono descritte dai metodi della classe.

La classe `controllers::ComunicazioneController` contiene attributi e metodi per la gestione delle richieste di input fornite dagli stakeholder(nel seguente caso riguarda solamente caregiver e logopedista) per tutto ciò che riguarda l'entità Comunicazione. Permette al sistema di elaborare le richieste di contatto da inoltrare per email al relativo stakeholder. Le restanti funzionalità offerte sono descritte dai metodi della classe.

La classe `models::Terapia` rappresenta l'insieme delle operazioni che permettono di creare, modificare, eliminare , ottenere i dati delle terapie dal database di tutti gli utenti. Le funzionalità offerte sono descritte dai metodi della classe.

La classe `models::Caregiver` rappresenta l'insieme delle operazioni che permettono di creare l'account caregiver , ottenere la lista assistito e trovare le terapie degli utenti dal database. Le funzionalità offerte sono descritte dai metodi della classe.

La classe `models::Appuntamento` rappresenta l'insieme delle operazioni che permettono di fissare un appuntamento al caregiver dalla logopedista, modificare eventuali dati dell'appuntamento e annullare l'appuntamento. Le funzionalità offerte sono descritte dai metodi della classe.

La classe `models::Esercizio` è una superclasse che fornisce attributi e operazioni per gestire le operazioni relative ai dati principali e comuni di ogni esercizio. Viene utilizzata dal logopedista per creare e modificare gli esercizi. Uno o più esercizi possono essere associati alle diverse terapie somministrate agli utenti. Questa classe viene specializzata dalle sottoclassi per gestire i dati per le varie tipologie di esercizio (`EsercizioDenominazione`, `EsercizioCoppieMinime`, `EsercizioRipetizioneParole`).

La classe `models::EsercizioDenominazione` è una sottoclasse che gestisce i metodi della classe base e i metodi getter e setter per l'immagine dell'esercizio.

La classe `models::EsercizioCoppieMinime` è una sottoclasse che gestisce i metodi della classe base e i metodi getter e setter per le due immagini associate. Nelle future implementazioni verrà aggiunto un file audio da poter riprodurre.

La classe `models::EsercizioRipetizioneParole` è una sottoclasse che gestisce i metodi della classe base e i metodi getter e setter delle tre parole associate.

La classe `models::EsercizioSvolto` gestisce gli esercizi relativi alla singola terapia somministrata e viene creata un'istanza ogni volta che un utente svolge un esercizio memorizzando la data di completamento, la risposta formulata e la valutazione conseguita (per comprendere se l'esercizio è stato svolto correttamente o meno). Dopo lo svolgimento dell'esercizio, i dati relativi a tale esercizio potranno essere convalidati dal caregiver assegnandogli una valutazione. Nelle future implementazioni questa funzionalità verrà effettuata automaticamente dal sistema.

La classe `models::TestAutodiagnosi` gestisce i metodi per la visualizzazione e la memorizzazione delle risposte attribuite al test di autovalutazione e autodiagnosi. Questo test può essere svolto esclusivamente dall'utente indipendentemente se esso ha effettuato l'accesso o meno. Nelle future implementazioni il sistema formulerà autonomamente mediante le risposte il disturbo dell'utente per semplificare l'operazione di formulazione della diagnosi del logopedista.

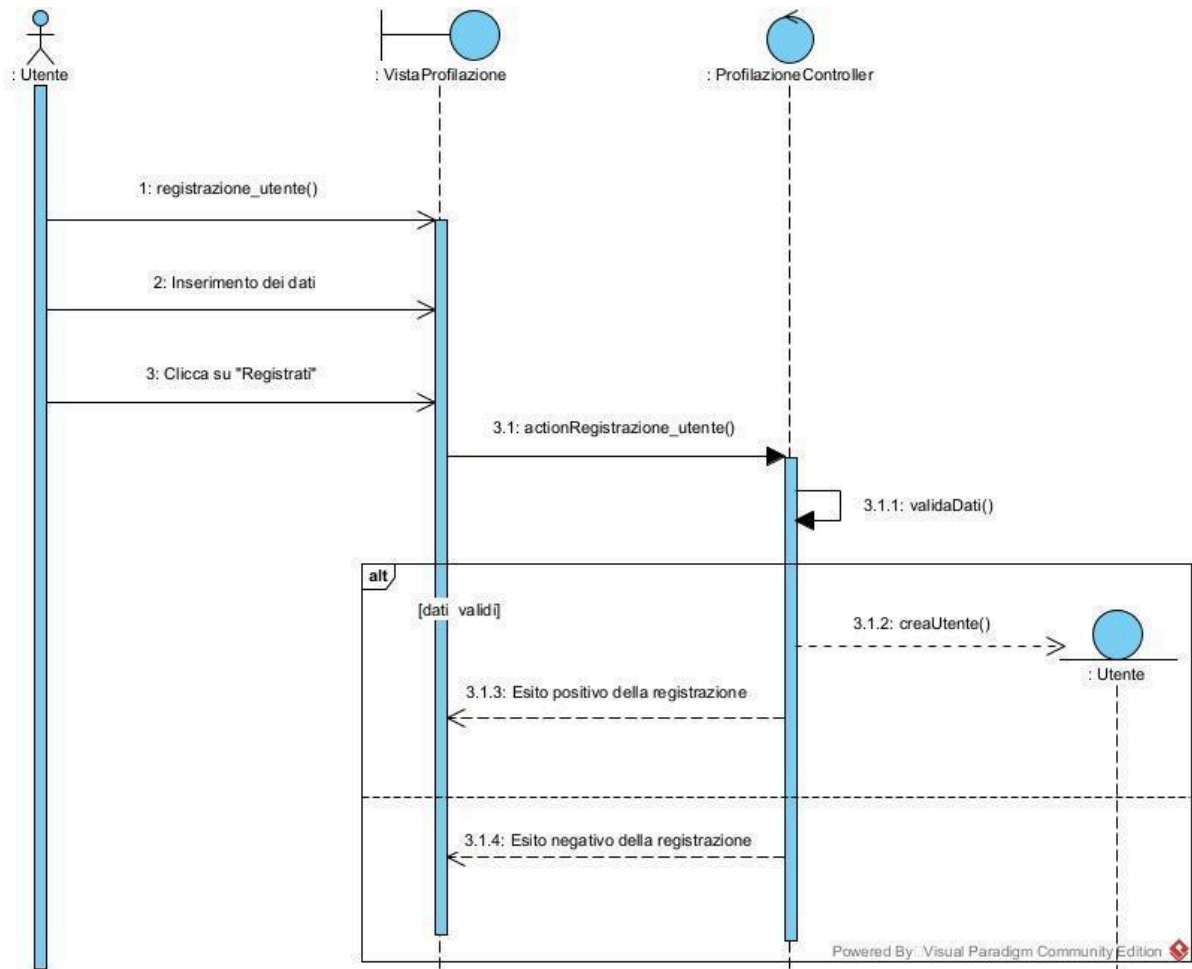
La classe `models::Logopedista` contiene attributi e metodi che riguardano l'entità Logopedista. Permette al sistema di memorizzare ed elaborare i dati di tale entità. Le funzionalità offerte sono descritte dai metodi della classe.

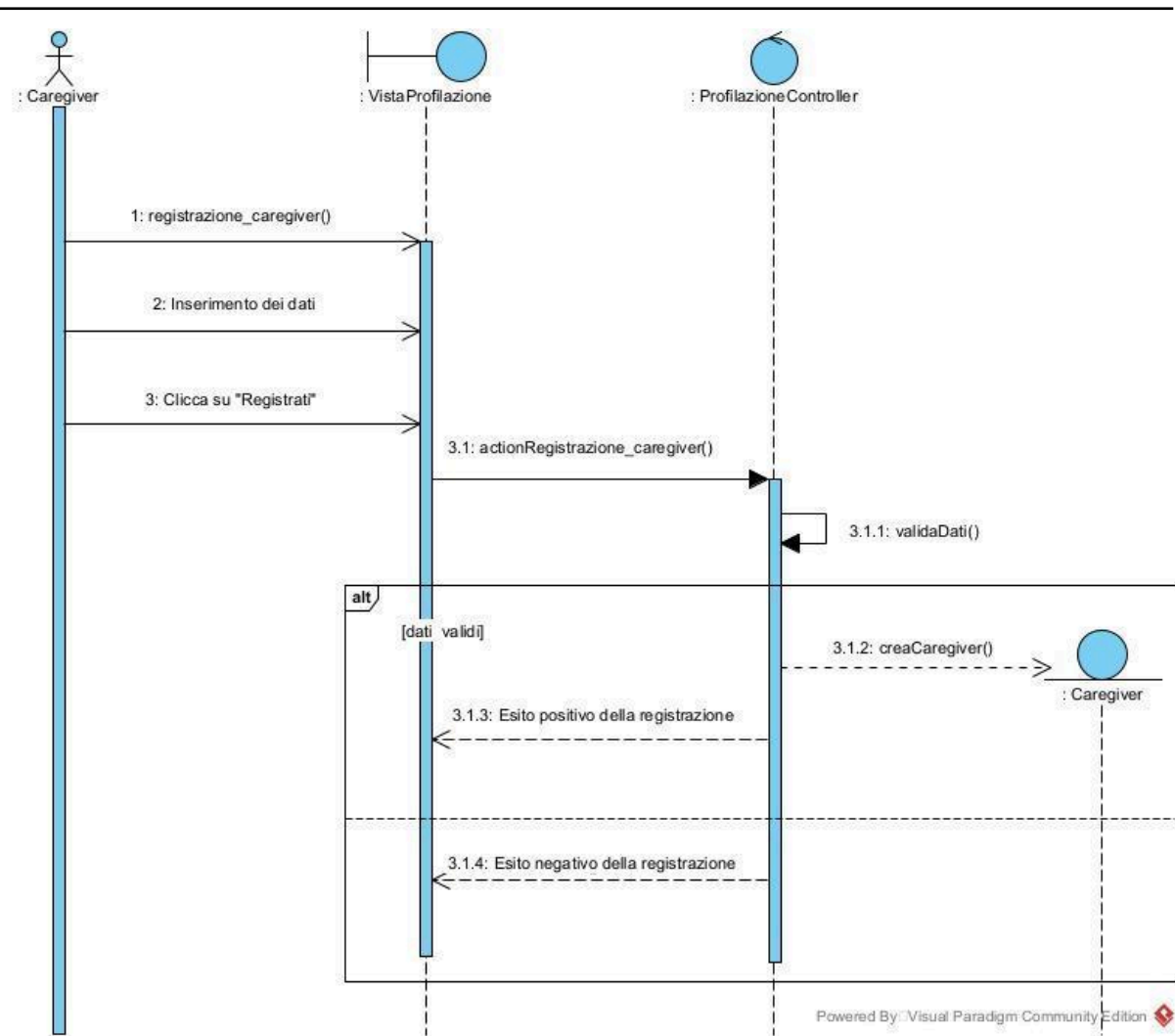
La classe `models::Utente` contiene attributi e metodi che riguardano l'entità Utente. Permette al sistema di memorizzare ed elaborare i dati di tale entità. Le funzionalità offerte sono descritte dai metodi della classe.

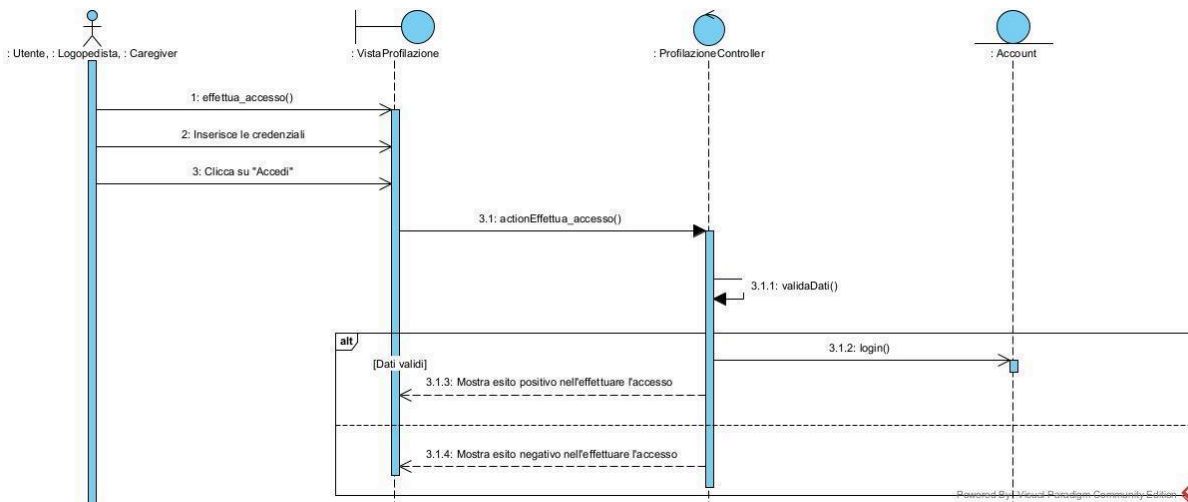
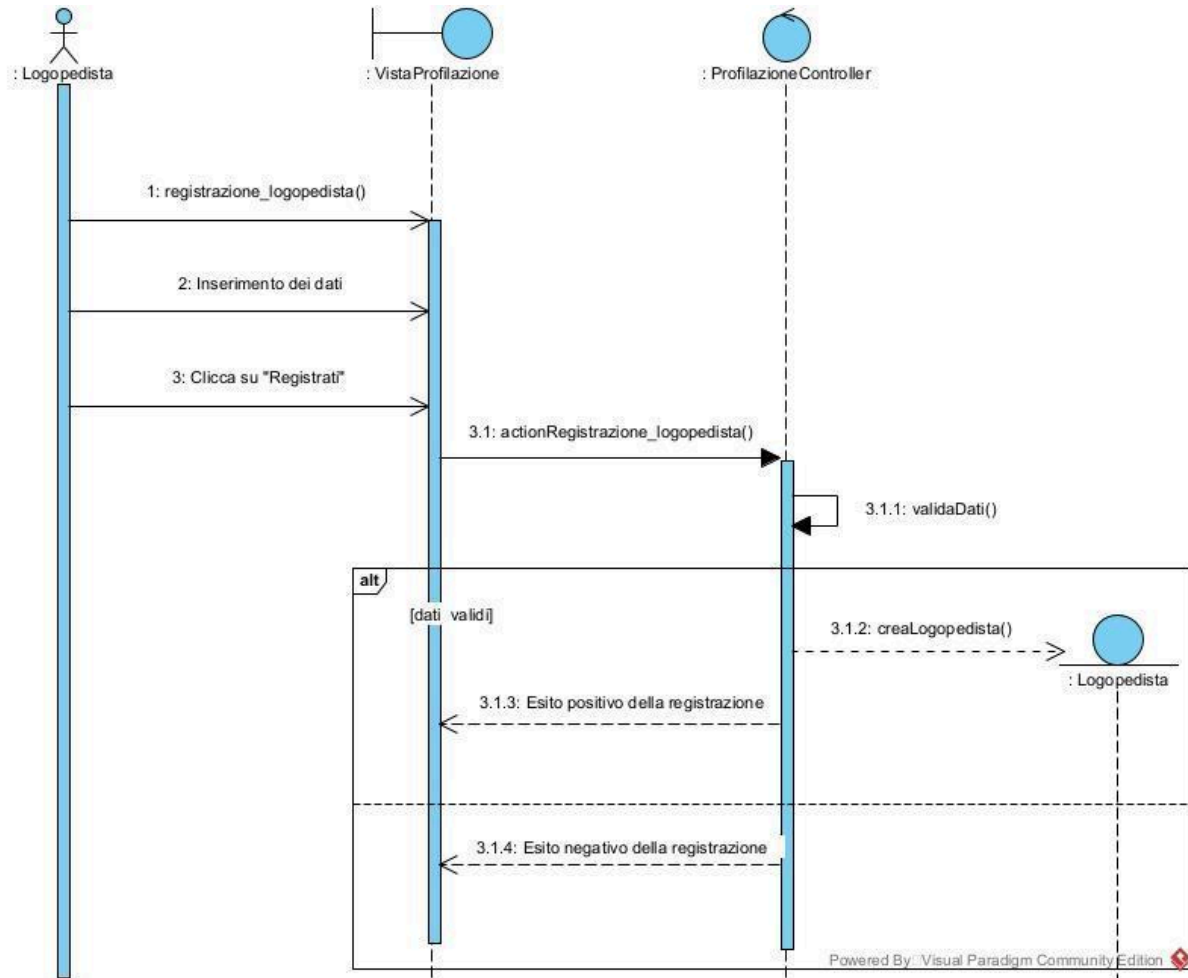
La classe `models::Account` contiene attributi e metodi che riguardano l'entità Account. Permette al sistema di memorizzare ed elaborare i dati di tale entità. Le funzionalità offerte sono descritte dai metodi della classe.

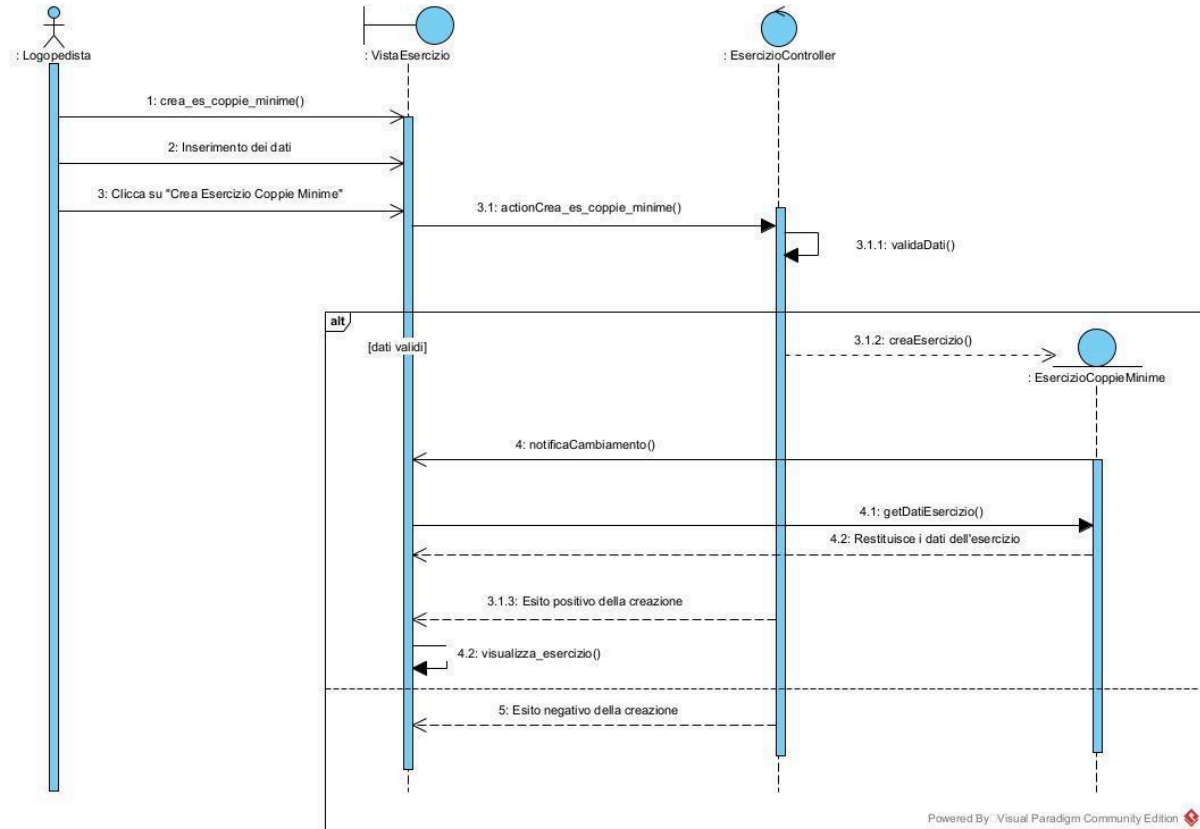
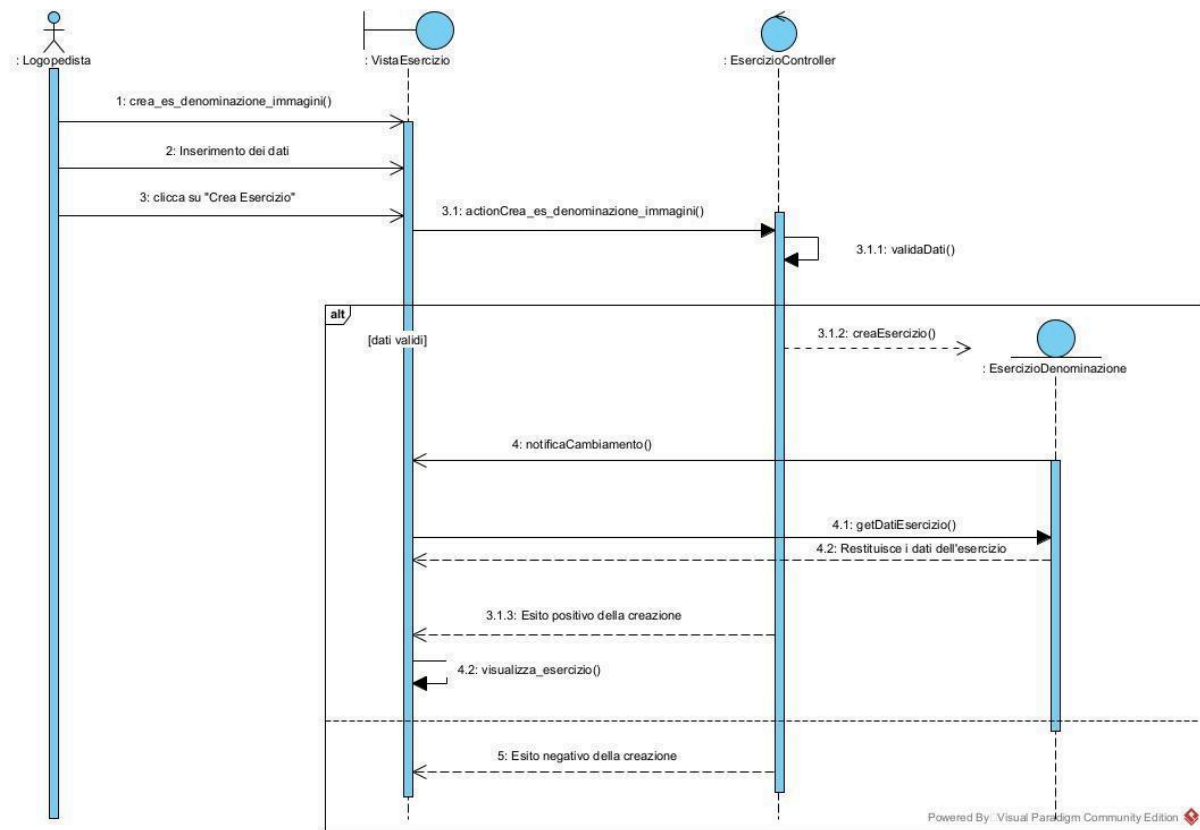
La classe `models::Comunicazione` contiene attributi e metodi che riguardano l'entità Comunicazione. Permette al sistema elaborare i dati di tale entità. Le funzionalità offerte sono descritte dai metodi della classe.

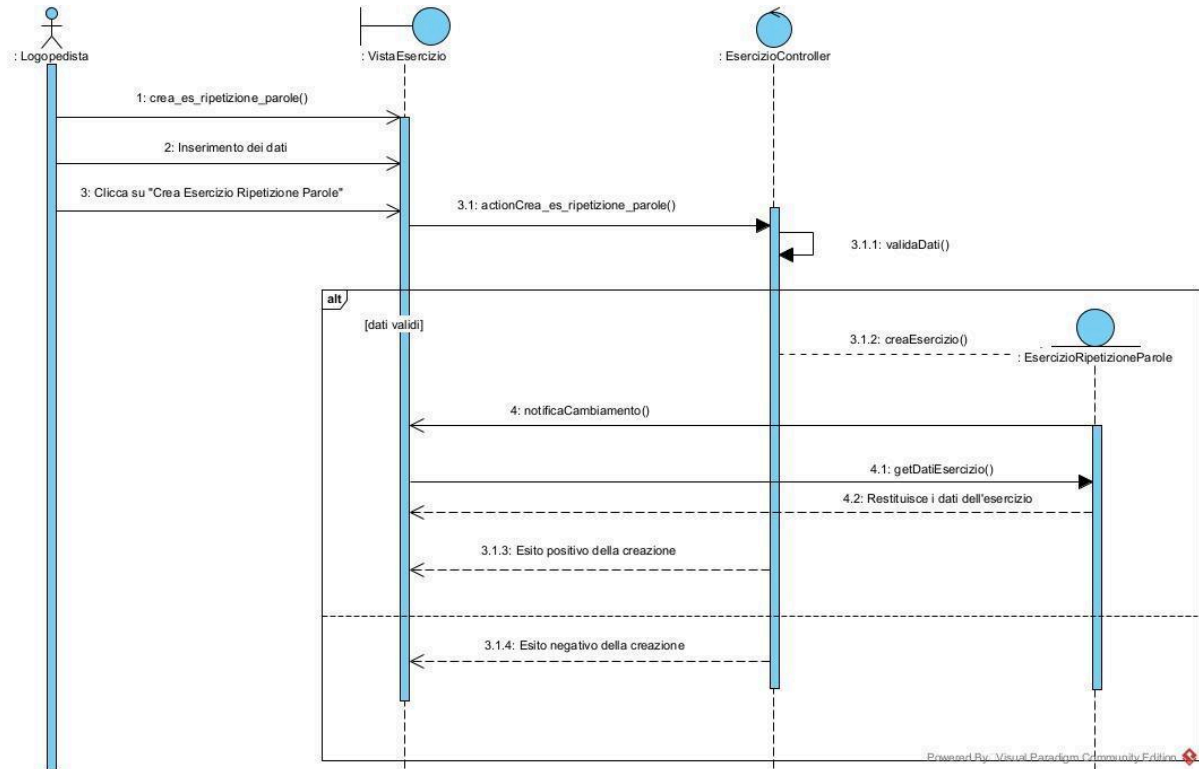
1.4.3 Diagrammi di Sequenza

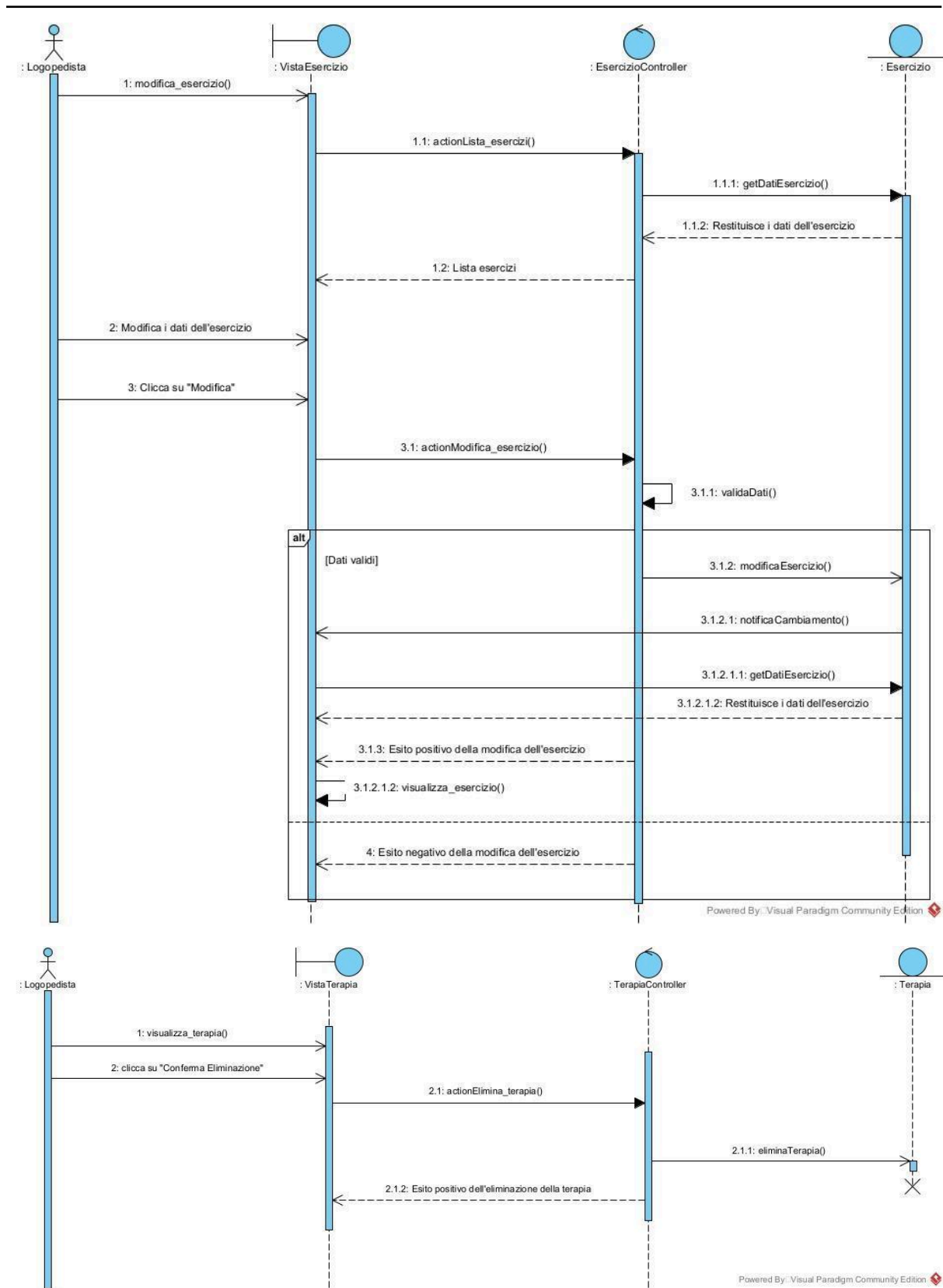


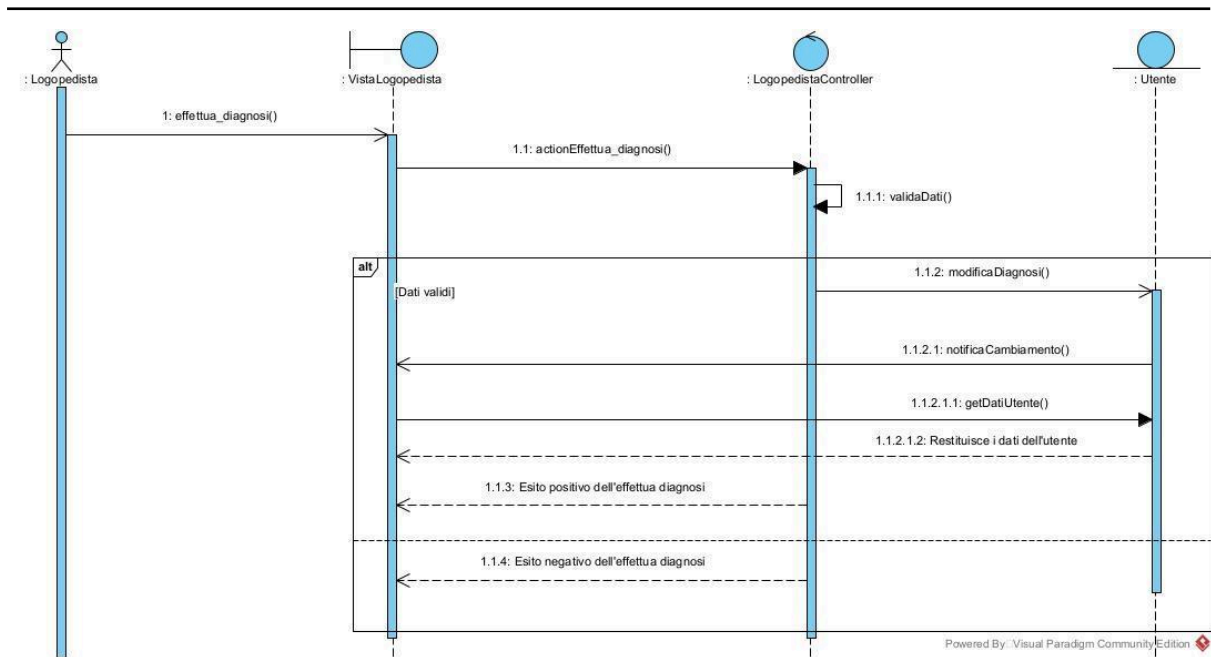


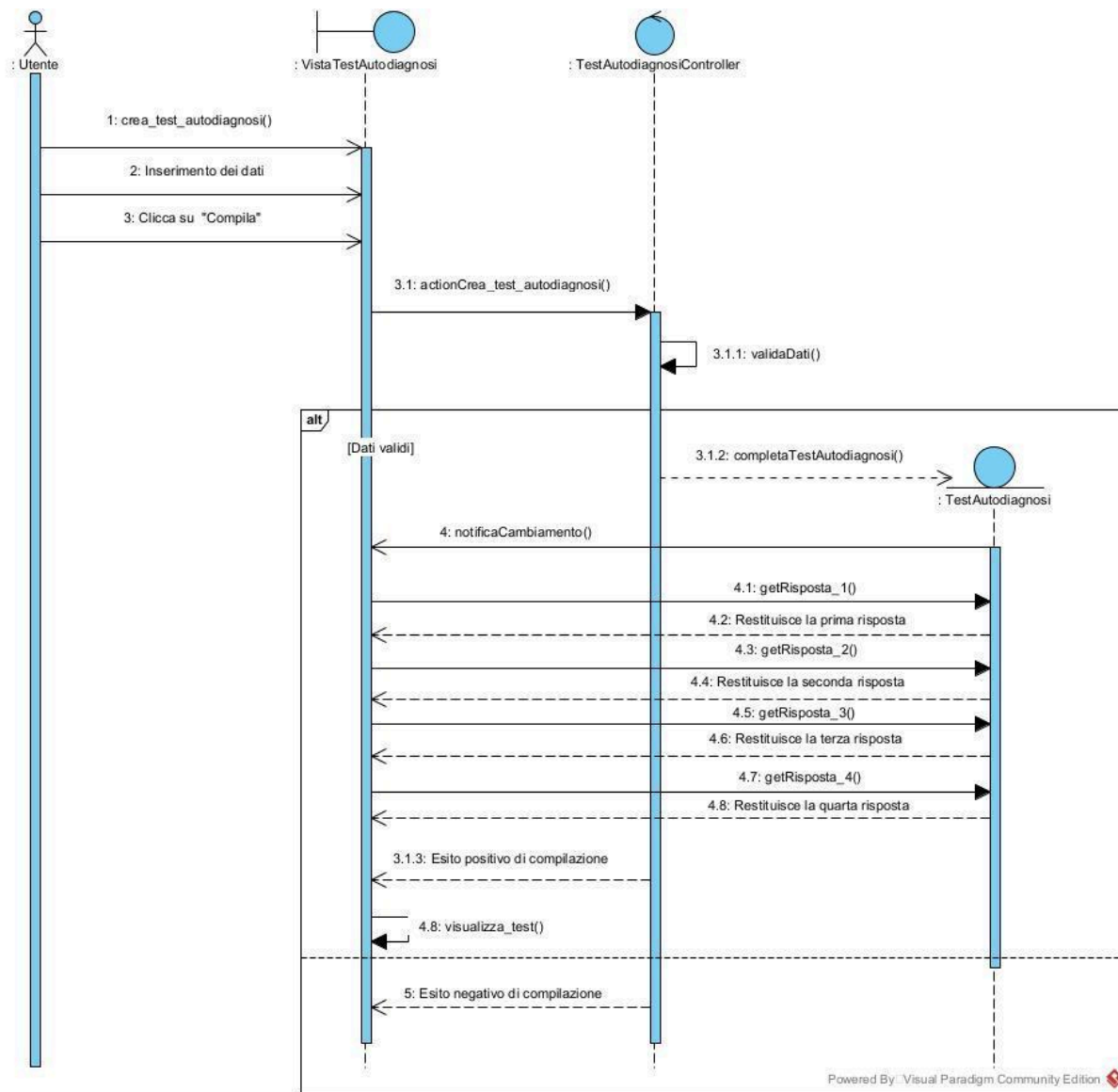


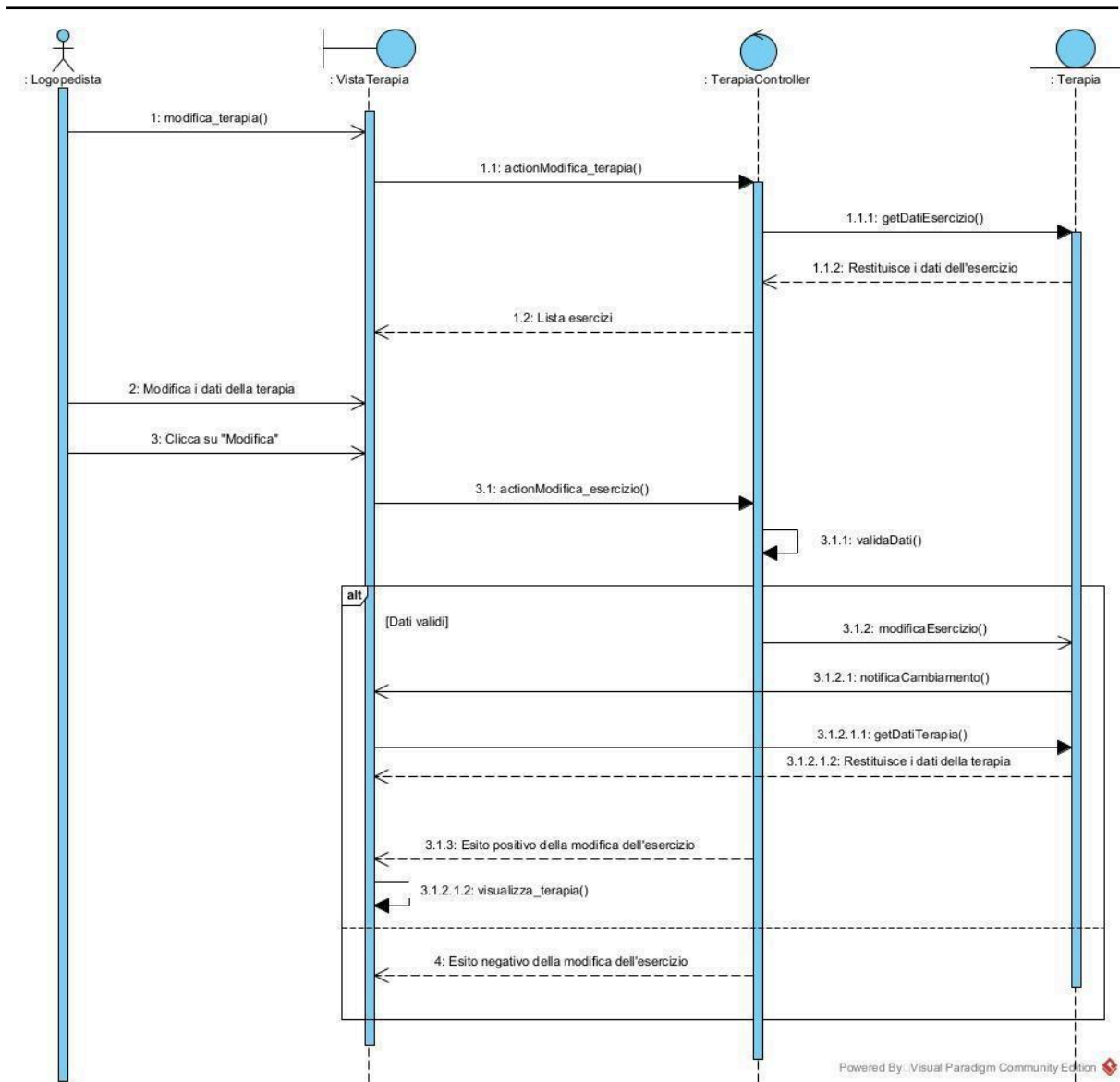


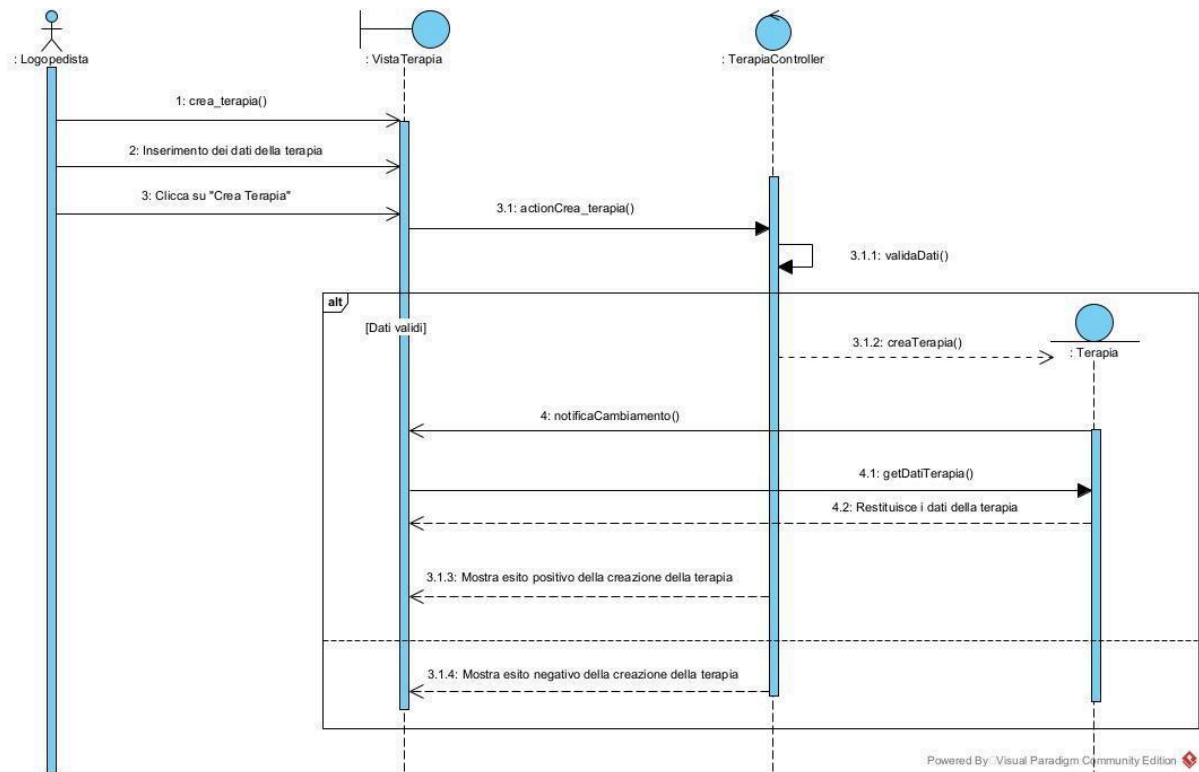
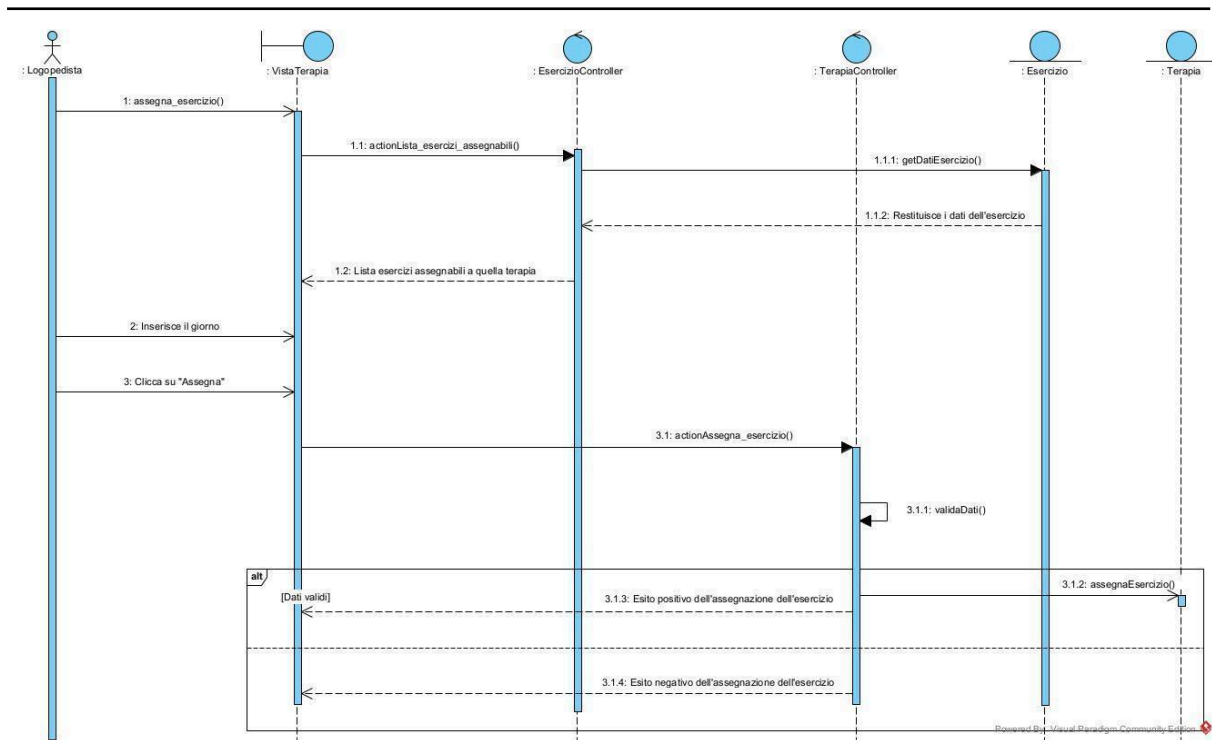


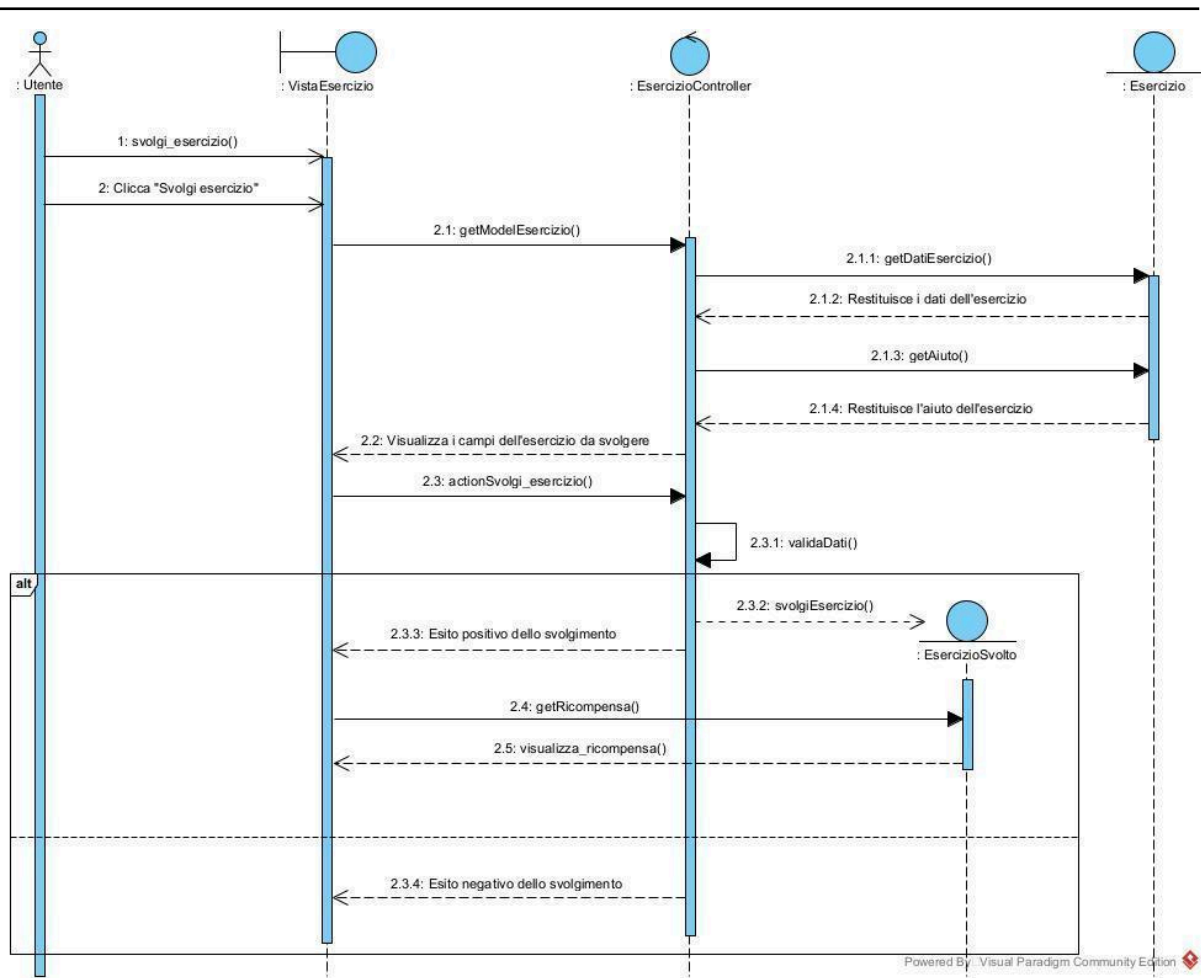


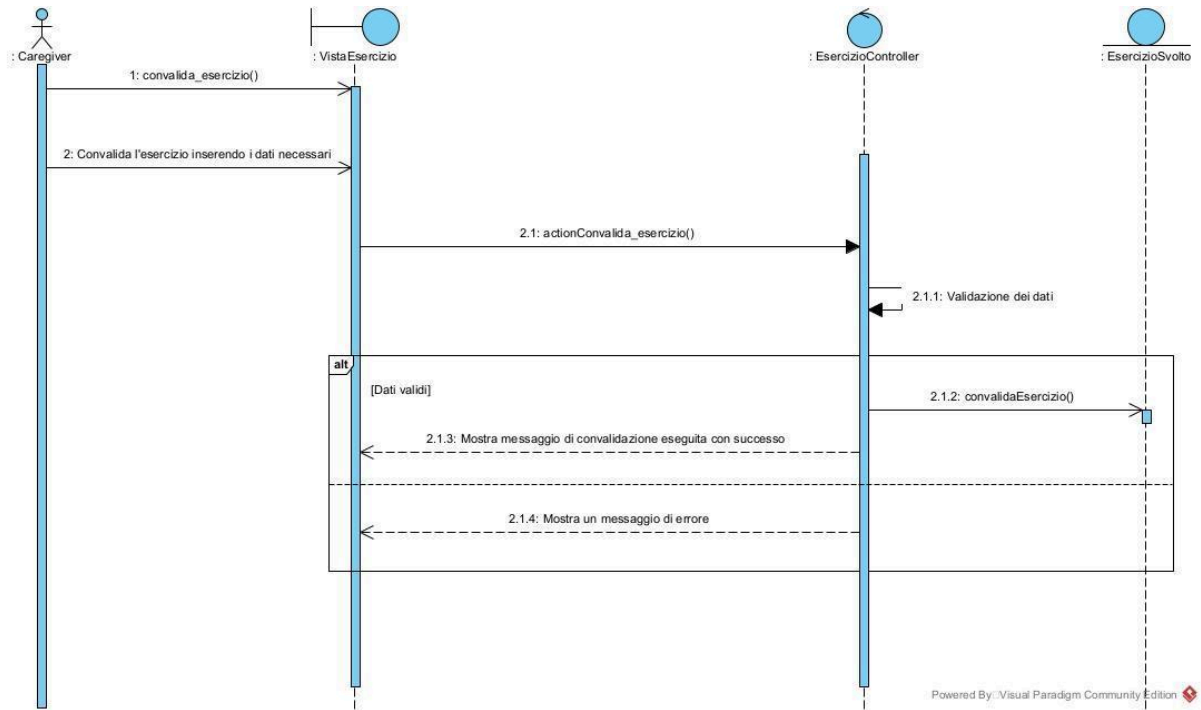


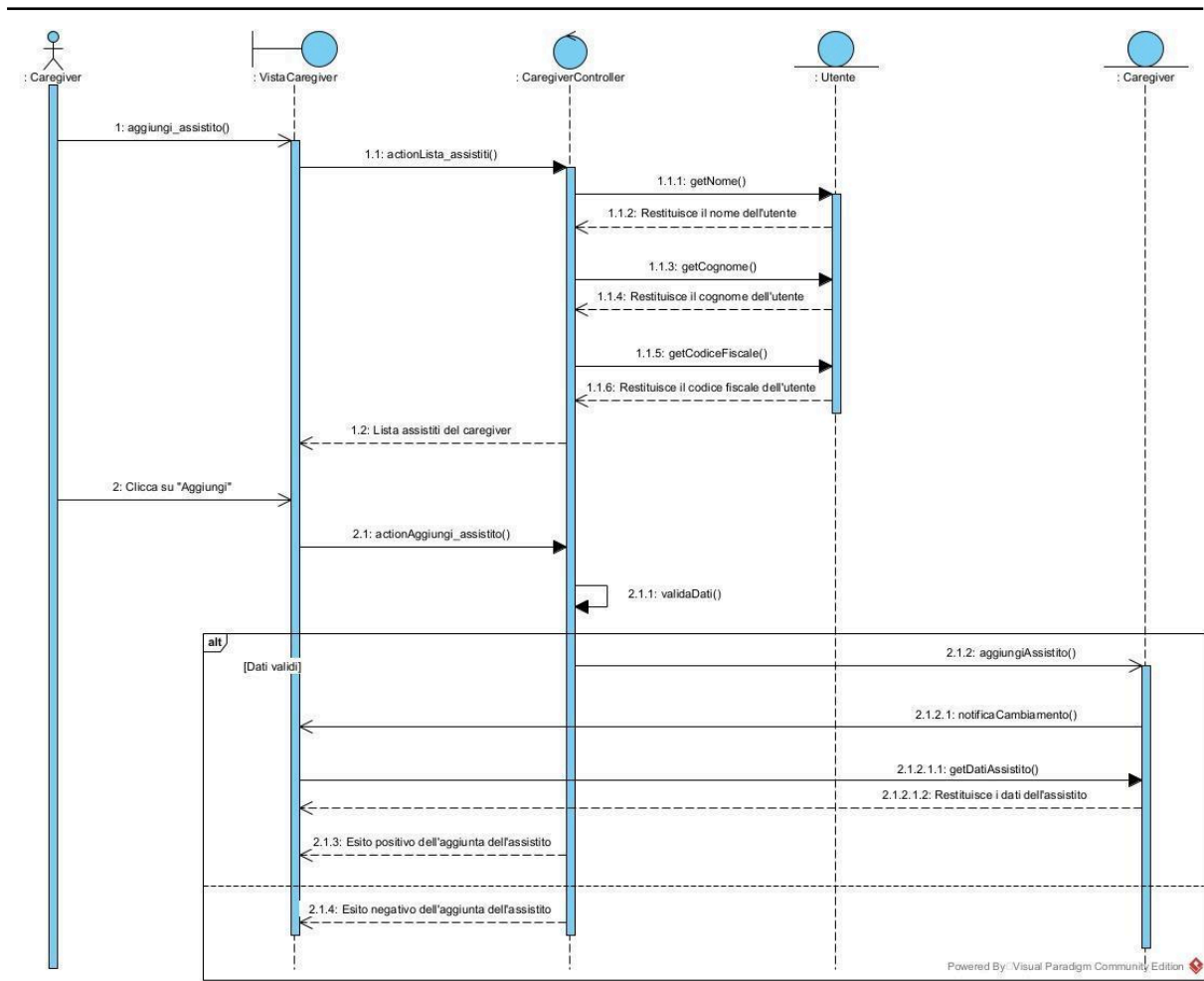


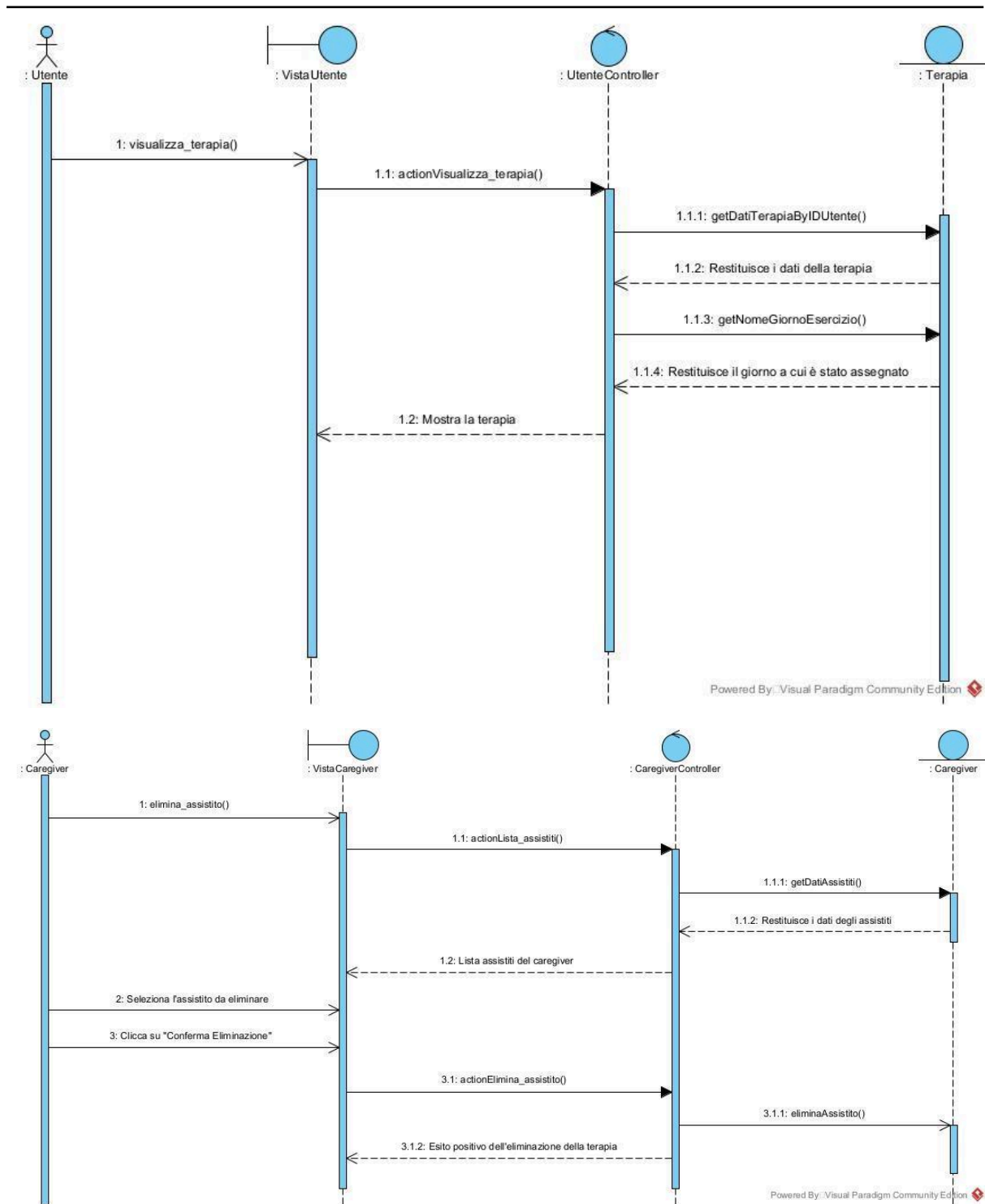


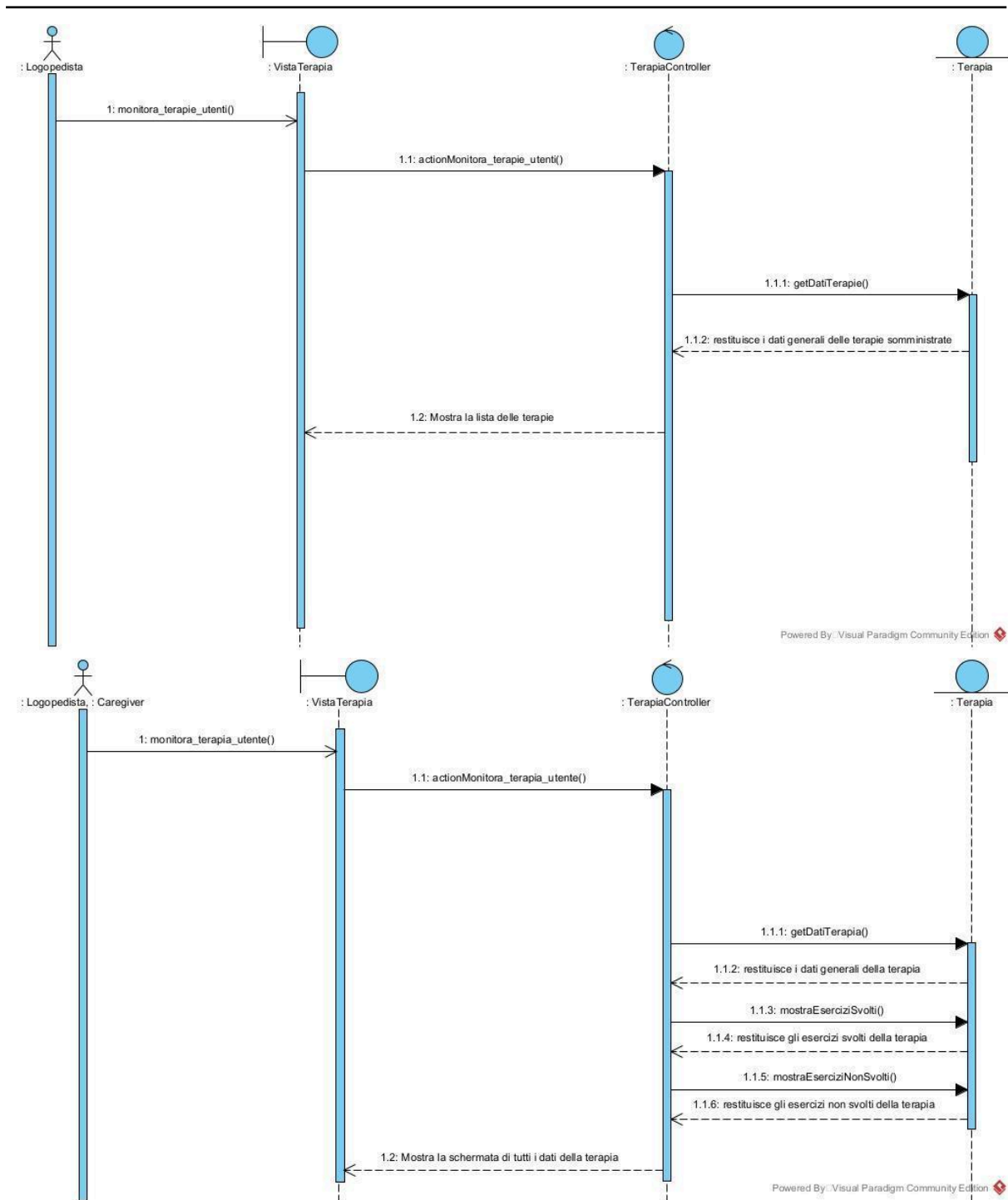


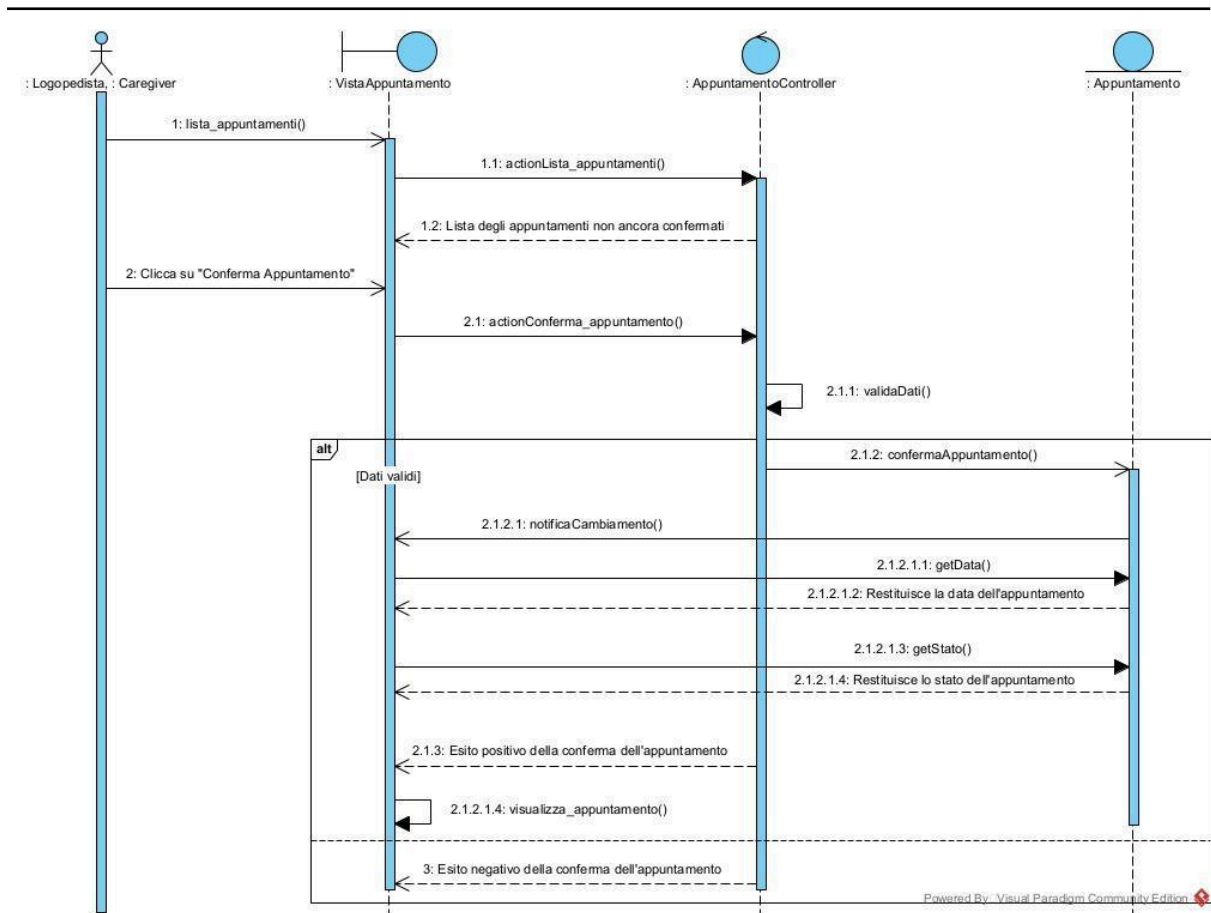


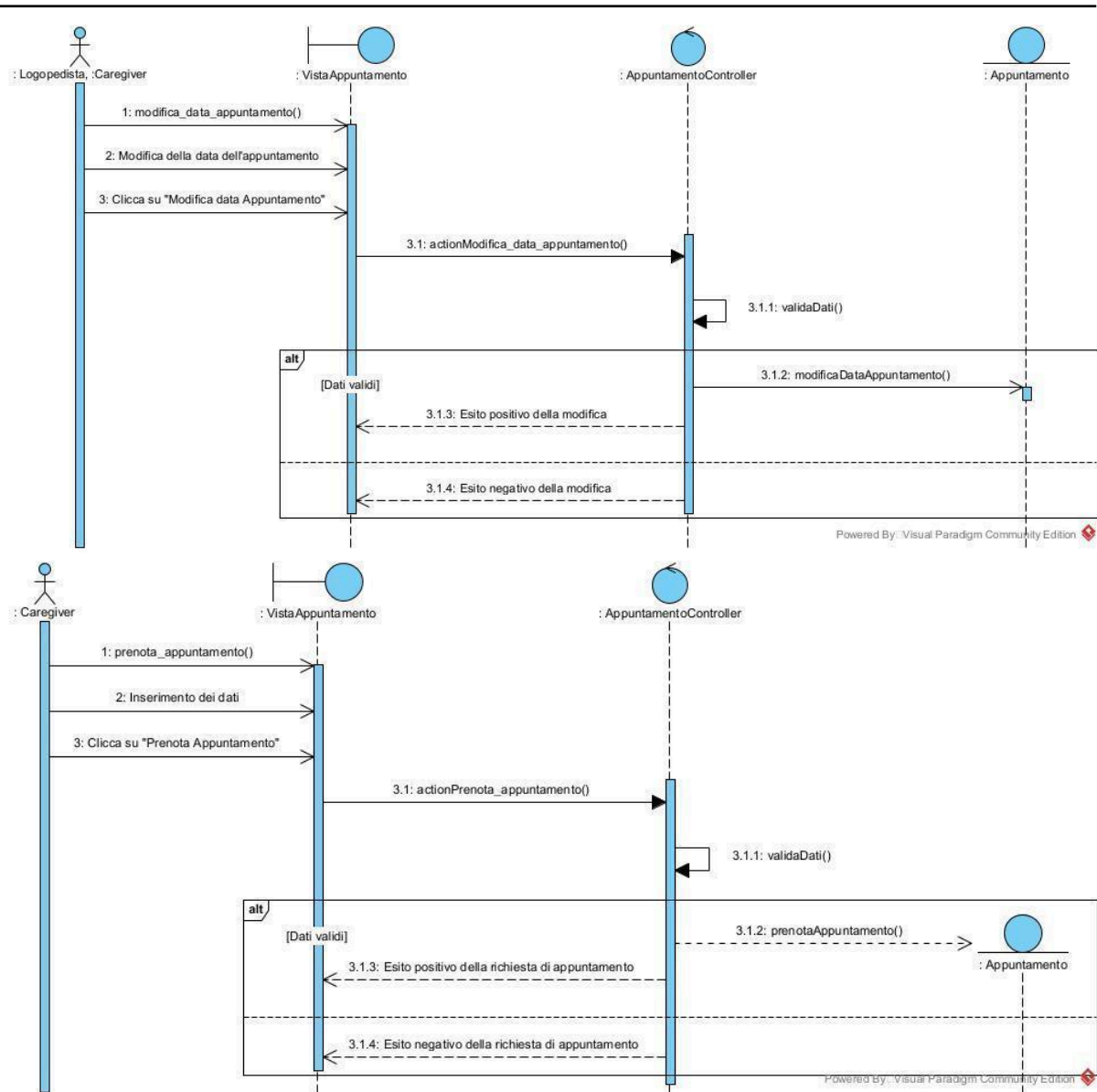


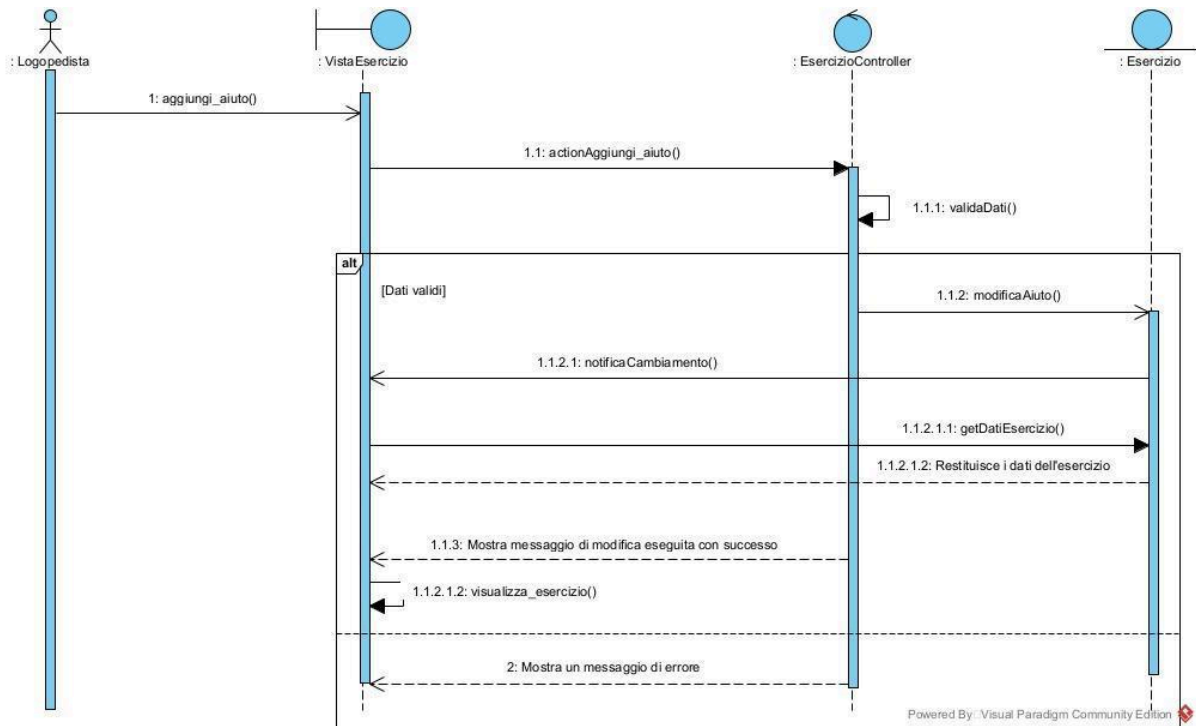
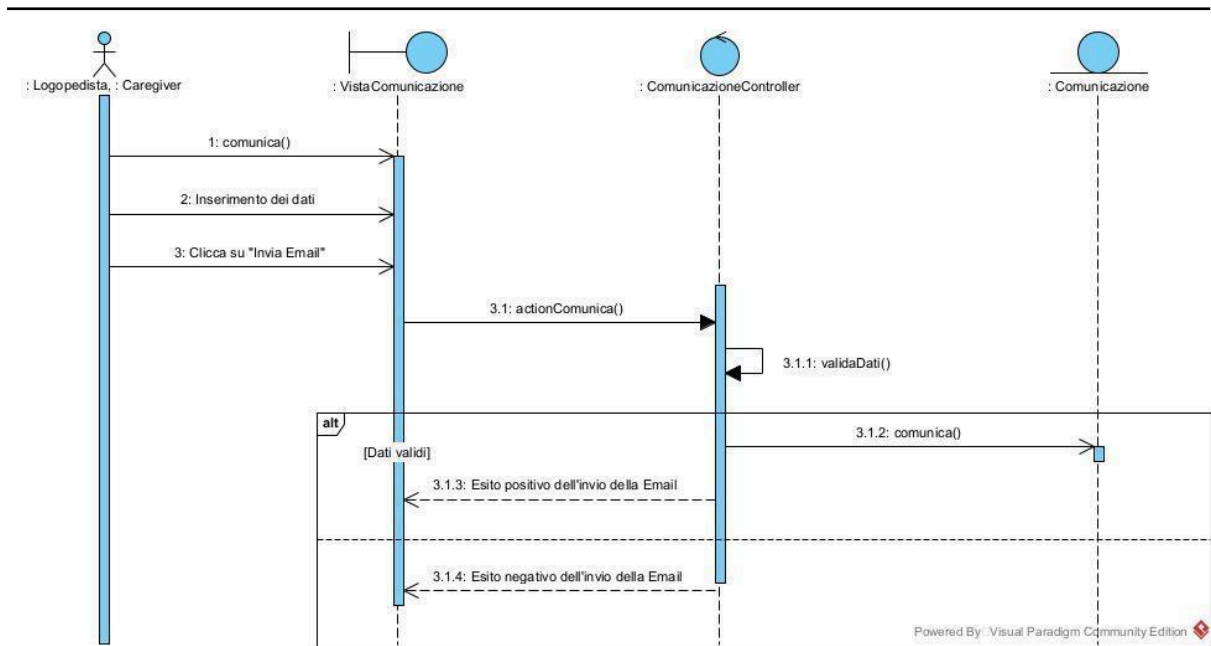








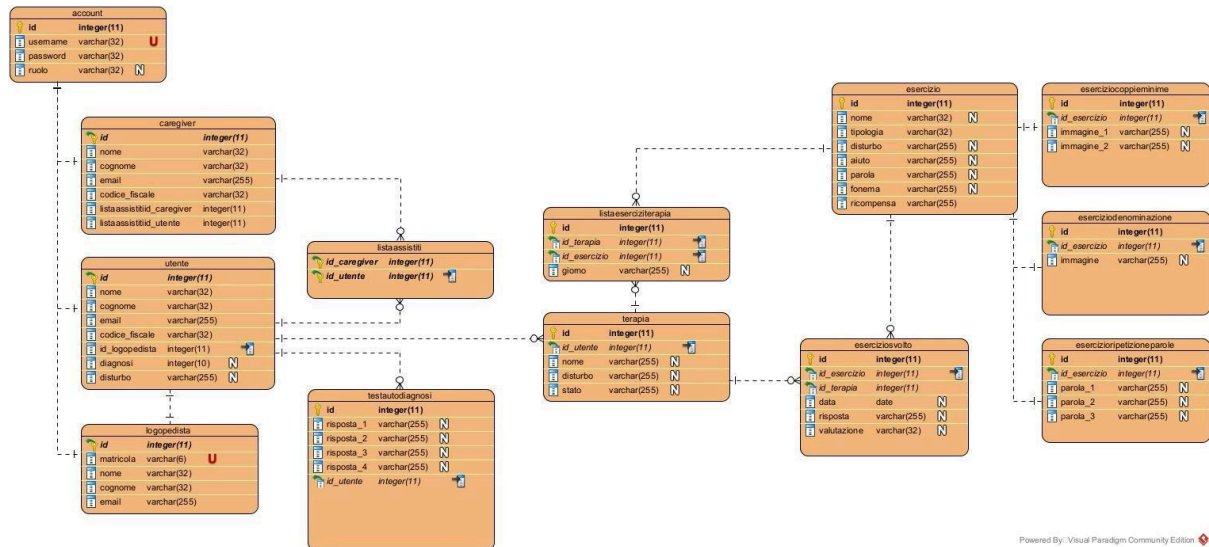




1.5 Data modeling and design

Qui va fornita la specifica di tutti i dati e le informazioni scambiate dal sistema in corso di realizzazione con l'utenza di riferimento e/o gli eventuali altri sistemi con cui esso comunica. Deve essere descritto il modello logico della base di dati e la sua struttura fisica.

1.5.1 Modello logico del Database



1.5.2 Struttura fisica del Database

1.6 Implementazioni Sprint Futuri

- **Comunica Caregiver - Comunica Logopedista**

La comunicazione prevederà un sistema di chat remota istantanea tra i due stakeholder.

- **Visualizza Ricompensa**

La ricompensa sarà un sistema di ricompensa autonomo basata sulle preferenze dell'utente. L'utente potrà scegliere la propria ricompensa scegliendo un link (video) oppure un avatar da personalizzare.

- **Completa Test Autodiagnosi**

Il Test Autodiagnosi prevederà un questionario preso dal ASCB (Abilità Socio-Conversazionali del Bambino). Inoltre una volta terminato il documento, il sistema restituirà all'utente il risultato di quello che potrebbe essere suo il disturbo comunicativo e linguistico.

- **Effettua Diagnosi**

Il logopedista avrà una piena panoramica dei possibili disturbi diagnosticabili dall'utente, e potrà confermare una delle diagnosi formulate dal sistema.

- **Assistito**

La gestione dell'assistito prevederà una suddivisione di quest'ultimo.

Gli utenti che sono autosufficienti, non dovranno necessariamente avere un Caregiver che gestisca la propria Terapia (per la convalida degli esercizi ecc).

- Registrazione vocale

La gestione della registrazione vocale nel sistema prevederà la registrazione, non appena si cliccherà sull'apposita icona del microfono, dell'audio da parte dell'utente o del caregiver. Successivamente potrà essere memorizzata ed ascoltata per la convalida degli esercizi ed il monitoraggio della terapia (sia per il Caregiver che per il Logopedista).



GLOSSARIO

- **Acronimi**

- **Definizioni**

UTENTE: è colui che ha un disturbo comunicativo e/o linguistico e segue la terapia fornitogli dal Logopedista per risolverlo.

CAREGIVER: è colui che occupa un ruolo informale di cura, supporto e di vicinanza e che è partecipe dell'esperienza di malattia del malato e che si impegna nelle attività quotidiane di cura della persona