

UNIVERSITÀ DEGLI STUDI DI NAPOLI “PARTHENOPE”
FACOLTÀ DI SCIENZE E TECNOLOGIE



DOCUMENTO DI ANALISI DEI REQUISITI
DOCUMENTO DI DESIGN DEL SISTEMA

Uni-Way

DOCENTE

Antonino Staiano

GRUPPO

C. Buonaugurio

C. Coppola

A. Cozzolino

G. D'Angelo

V. Zavino

Anno Accademico 2019-2020

Indice

I	Introduzione	1
1	Definizione del problema	2
1.1	Scopo del sistema	3
1.2	Ambito del sistema	3
1.3	Obiettivi e criteri di successo del progetto	4
1.4	Definizioni, acronimi e abbreviazioni	4
II	Documento di Analisi dei Requisiti	5
2	Sistema Proposto	6
2.1	Panoramica	6
2.2	Requisiti Funzionali	7
2.2.1	FR1	7
2.2.2	FR2	7
2.2.3	FR3	7
2.2.4	FR4	7
2.2.5	FR5	8
2.2.6	FR6	8
2.2.7	FR7	8
2.2.8	FR8	8
2.2.9	FR9	8
2.3	Requisiti Non Funzionali	9
2.3.1	NFR1	9
2.3.2	NFR2	9
2.3.3	NFR3	9

<i>INDICE</i>	III
---------------	-----

2.3.4	NFR4	9
2.4	Modello del Sistema	10
2.4.1	Scenari	10
2.4.2	Casi D'Uso	18
2.4.3	Modello ad Oggetti	65
2.4.4	Modello Dinamico	90
2.4.5	User Interface Mockups	98

III	Documento di Design del Sistema	99
------------	--	-----------

3	Sistema Proposto	100
3.1	Panoramica	100
3.2	Decomposizione in Sottosistemi	101
3.3	Hardware/Software Mapping	103
3.4	Gestione dei Dati Persistenti	104
3.5	Controllo Accessi e Sicurezza	105
3.6	Flusso di Controllo Globale	106
3.7	Condizioni Limite	107

IV	Glossario	108
-----------	------------------	------------

4	Divisione dei Glossari	109
4.1	Glossario del RAD	109
4.2	Glossario del SDD	110

Elenco delle tabelle

2.1	Scenario <i>cercaAula</i>	11
2.2	Scenario <i>conosciStudiante</i>	12
2.3	Scenario <i>richiestaMateriale</i>	13
2.4	Scenario <i>notificaCommentoAlTopic</i>	14
2.5	Scenario <i>visualizzaAnnuncioUni</i>	14
2.6	Scenario <i>cambiaIdeaDiPrivacy</i>	15
2.7	Scenario <i>entraInUniWay</i>	16
2.8	Scenario <i>cambioTelefono</i>	17
2.9	Caso d'uso <i>ConnessionePersa</i>	19
2.10	Caso d'uso <i>DatoMancante</i>	21
2.11	Caso d'uso <i>MultimediaMancante</i>	22
2.12	Caso d'uso <i>ContenutoNotificaMancante</i>	23
2.13	Caso d'uso <i>AmiciziaMancante</i>	24
2.14	Caso d'uso <i>CorsoMancante</i>	25
2.15	Caso d'uso <i>CommentoMancante</i>	26
2.16	Caso d'uso <i>TopicMancante</i>	27
2.17	Caso d'uso <i>StudianteMancante</i>	28
2.18	Caso d'uso <i>RegistrazioneApp</i>	30
2.19	Caso d'uso <i>Login</i>	31
2.20	Caso d'uso <i>Logout</i>	32
2.21	Caso d'uso <i>RecuperaPassword</i>	33
2.22	Caso d'uso <i>ModificaPreferenze</i>	35
2.23	Caso d'uso <i>ModificaNotifiche</i>	36
2.24	Caso d'uso <i>ModificaPrivacy</i>	37
2.25	Caso d'uso <i>ModificaDatiUtente</i>	38

2.26	Caso d'uso <i>SeguiCorso</i>	40
2.27	Caso d'uso <i>SmettiDiSeguireCorso</i>	41
2.28	Caso d'uso <i>CercaRisorse</i>	43
2.29	Caso d'uso <i>CercaCommenti</i>	44
2.30	Caso d'uso <i>VisualizzaRisorsa</i>	45
2.31	Caso d'uso <i>VisualizzaCorso</i>	45
2.32	Caso d'uso <i>VisualizzaCommento</i>	46
2.33	Caso d'uso <i>VisualizzaPost</i>	46
2.34	Caso d'uso <i>VisualizzaMultimedia</i>	47
2.35	Caso d'uso <i>VisualizzaStudiante</i>	47
2.36	Caso d'uso <i>InviaRichiestaDiAmicizia</i>	49
2.37	Caso d'uso <i>VisualizzaCorsiAmico</i>	50
2.38	Caso d'uso <i>AvviaChat</i>	51
2.39	Caso d'uso <i>RimuoviAmico</i>	52
2.40	Caso d'uso <i>MostraApprezzamento</i>	54
2.41	Caso d'uso <i>CreaTopic</i>	55
2.42	Caso d'uso <i>ScriviCommento</i>	56
2.43	Caso d'uso <i>CancellaTopic</i>	57
2.44	Caso d'uso <i>CancellaCommento</i>	58
2.45	Caso d'uso <i>RiceviNotifica</i>	60
2.46	Caso d'uso <i>RiceviNotificaRichiestaDiAmicizia</i>	61
2.47	Caso d'uso <i>RiceviNotificaAnnuncio</i>	61
2.48	Caso d'uso <i>RiceviNotificaPostAmico</i>	62
2.49	Caso d'uso <i>RiceviNotificaChat</i>	63
2.50	Caso d'uso <i>RiceviNotificaChat</i>	64
2.51	Tabella oggetti Entity - Pkg In-Out	66
2.52	Tabella oggetti Boundary - Pkg In-Out	67
2.53	Tabella oggetti Control - Pkg In-Out	69
2.54	Tabella oggetti Entity - Pkg Modifica	70
2.55	Tabella oggetti Boundary - Pkg Modifica	71
2.56	Tabella oggetti Control - Pkg Modifica	72
2.57	Tabella oggetti Entity - Pkg Corso	73
2.58	Tabella oggetti Boundary - Pkg Corso	73

2.59	Tabella oggetti Control - Pkg Corso	74
2.60	Tabella oggetti Entity - Pkg Ricerca	75
2.61	Tabella oggetti Boundary - Pkg Ricerca	76
2.62	Tabella oggetti Control - Pkg Ricerca	77
2.63	Tabella oggetti Entity - Pkg Social	78
2.64	Tabella oggetti Boundary - Pkg Social	79
2.65	Tabella oggetti Control - Pkg Social (Parte 1)	80
2.66	Tabella oggetti Control - Pkg Social (Parte 2)	81
2.67	Tabella oggetti Entity - Pkg Topic-Commenti	82
2.68	Tabella oggetti Boundary - Pkg Topic-Commenti	83
2.69	Tabella oggetti Control - Pkg Topic-Commenti	84
2.70	Tabella delle Entità	85
2.71	Tabella oggetti Boundary - Pkg Notifica	85
2.72	Tabella oggetti Control - Pkg Notifica	86
2.73	Tabella oggetti Control - Pkg Notifica	87
4.1	Glossario RAD parte I	109
4.2	Glossario RAD parte II	110

Elenco delle figure

2.1	Diagramma dei casi d'uso del package Eccezioni.	20
2.2	Diagramma dei casi d'uso del package In-Out.	29
2.3	Diagramma dei casi d'uso del package Modifica.	34
2.4	Diagramma dei casi d'uso, sezione relativa ai <i>corsi</i>	39
2.5	Diagramma dei casi d'uso relativi alle funzionalità di <i>ricerca</i>	42
2.6	Diagramma dei casi d'uso del package Social.	48
2.7	Diagramma dei casi d'uso, sezione relativa ai topic e commenti.	53
2.8	Diagramma dei casi d'uso, sezione relativa alle notifiche.	59
2.9	Diagramma delle classi.	89
2.10	Diagramma delle sequenze - Registrazione	91
2.11	Diagramma delle sequenze - Modifica Privacy	92
2.12	Diagramma delle sequenze - gestione notifica chat	93
2.13	Diagramma delle sequenze - ricerca parte 1	94
2.14	Diagramma delle sequenze - ricerca parte 2	94
2.15	Diagramma delle sequenze - gestione chat parte 1	95
2.16	Diagramma delle sequenze - gestione chat parte 2	95
2.17	Diagramma delle sequenze - Mostra Apprezzamento	96
2.18	Diagramma delle sequenze - Scrivi Commento	97
3.1	Decomposizione in sottosistemi generale (UML Component Diagram, livelli mostrati come UML package)	101
3.2	Decomposizione in sottosistemi, parte topic-commenti (UML Component Diagram)	102
3.3	Hardware/Software Mapping (UML Deployment Diagram)	103

Parte I

Introduzione

Capitolo 1

Definizione del problema

Il problema

Attualmente, non ci sono applicazioni dedicate a studenti dell'Università. Gli studenti vorrebbero frequentare i corsi universitari e vedere gli annunci relativi a tali corsi, tuttavia tali informazioni sono sparse su diversi siti web e risultano difficili da trovare. Inoltre, gli studenti vorrebbero trovare compagni di studio con gli stessi interessi e condividere le loro opinioni sui corsi e sul relativo materiale didattico, anche se purtroppo ciò ancora non è possibile. Infine, essi vorrebbero anche discutere le domande di esame e trovare il luogo giusto dove si svolgono gli esami senza ricorrere a Google.

Gli scenari

Raj, uno studente indiano, studia informatica alla University of Nowhere. Il suo piano di studi prevede Economia Aziendale come materia secondaria ed è già solito visitare i corsi nell'edificio di Informatica. I corsi di Economia Aziendale, tuttavia, si trovano in un'aula di un altro edificio nel centro di Nowhere. Raj non ha mai visitato l'altro edificio prima, quindi non sa come trovare le aule per la sua materia secondaria. Navigando attraverso i corsi nel catalogo dei corsi dell'App dell'Università è in grado di trovare il corso "Fondamenti dell'Economia Aziendale" con relativi orari e la posizione dell'aula su una mappa. Mentre frequenta il corso, entra in contatto con altri studenti ed è in grado di leggere i loro commenti. Gli piace un commento "Grandi esercizi" di Penny, che studia anche Informatica. Dalla foto di Penny, ricorda

che si sono incontrati una settimana fa alla macchina del caffè. Raj manda una richiesta di amicizia a Penny (che potrebbe aiutarlo a superare l'esame) e aggiunge un nuovo commento sulle domande d'esame delle sessioni precedenti. Durante la navigazione, Penny viene informata della richiesta di amicizia e la conferma. Raj, a sua volta, viene informato che Penny ha accettato la sua richiesta e ora passa in rassegna tutti i corsi che Penny sta visitando. Trova un altro corso interessante, "Contabilità", che vuole visitare e lo salva nella sua lista dei corsi.

1.1 Scopo del sistema

Ci sono moltissime applicazioni/servizi che permettono di poter accedere a sistemi di messaggistica con amici, visualizzazioni di informazioni relative a corsi e condivisione di materiale, ma non esiste ancora nessuna soluzione relativa all'Università di Nowhere che permetta di poter usufruire di tutte queste funzioni in pochi click con un'unica App. Uni-Way nasce quindi con lo scopo di facilitare la vita universitaria di migliaia di studenti che frequentano le aule di Nowhere.

1.2 Ambito del sistema

Uni-Way è un'App sviluppata in Java che permette di poter accedere alle funzioni soprascritte in un massimo di 3 click. Per la registrazione di uno studente l'App richiede numero di matricola e password fornite dall'Università di Nowhere, per poi relegare gli accessi successivi (a meno di un'autenticazione automatizzata) ad username e password create appositamente per il proprio account. Per poter accedere alle molteplici funzionalità sarà richiesta una connessione costante durante l'utilizzo dell'applicativo.

1.3 Obiettivi e criteri di successo del progetto

Il successo del progetto dipende dai vari obiettivi prefissati:

- Il sistema proposto deve essere ben schematizzato e semplice in modo da poter essere realizzato in modo chiaro e intuitivo in Java;
- La GUI deve risultare comprensibile e facilmente accessibile;
- L'App deve risultare performante su qualsiasi dispositivo che supporti la versione di Android/iOs minima che verrà dichiarata una volta rilasciata la build finale;
- L'App deve inoltre risultare responsive in modo tale da poter adattarsi a qualsiasi dispositivo (smartphone o tablet).

1.4 Definizioni, acronimi e abbreviazioni

App: Applicazione Uni-Way

GUI: Graphic User Interface (Interfaccia Grafica)

RAD: Requirements Analysis Document

SDD: System Design Document

CdU: Caso d'Uso

Parte II

Documento di Analisi dei Requisiti

Capitolo 2

Sistema Proposto

2.1 Panoramica

Dall'analisi del problema proposto dal cliente, è stata formulata la seguente soluzione su come strutturare il sistema finale. Esso sarà suddiviso in due *macro-aree* rispettivamente riguardanti l'aspetto didattico e l'aspetto social.

L'Aspetto didattico si occupa di raccogliere tutte le informazioni, sia attraverso il recupero di dati (quali orari, posizione dell'aule, annunci ufficiali relativi ai corsi ecc.) da fonti di terze parti riconosciute, sia attraverso la condivisione di materiale ufficiale e o prodotto dagli studenti.

L'Aspetto social consiste nella possibilità degli studenti di messaggiare privatamente e scambiarsi opinioni e materiale riguardo gli esami del piano di studi, attraverso una piattaforma di tipo *forum*.

La soluzione proposta, quindi, consiste in un'applicazione che si occupa di gestire l'acquisizione, la creazione e la persistenza dei dati di entrambe le aree, permettendo un unico punto d'accesso alle funzionalità di entrambi i tipi.

2.2 Requisiti Funzionali

2.2.1 FR1

Ricerca dei corsi disponibili: uno studente deve essere in grado di poter visualizzare tutti i corsi (del solo semestre in corso) delle sue materie e dei suoi insegnamenti a scelta. È in grado di unirsi ad un corso salvandolo nella sua lista. Può anche lasciare un corso.

2.2.2 FR2

Controllo dei dettagli di un corso: uno studente deve essere in grado di visualizzare i dettagli di un corso come gli orari del corso, la posizione dell'aula su una mappa, una lista dei partecipanti a tale corso (incluso nome e foto profilo). Nella lista del corso sono presenti i nomi di tutti i partecipanti, amici o non dello studente in questione.

2.2.3 FR3

Aggiornamento profilo: uno studente può aggiornare le sue impostazioni del profilo e la sua immagine del profilo. Può anche modificare le impostazioni di notifica e di privacy (come la possibilità di non permettere a studenti non amici di controllare la lista dei corsi seguiti).

2.2.4 FR4

Aggiunta dei commenti-topic: uno studente è in grado di creare topic ed aggiungervi commenti al di sotto di essi. Inoltre un utente sarà in grado di eliminare i propri commenti ed i propri topic (andando quindi ad eliminare tutti i relativi commenti ad esso).

2.2.5 FR5

Richiesta di amicizia: uno studente deve essere in grado di poter mandare richieste di amicizia ad altri utenti (i quali riceveranno una notifica in merito a tale richiesta). Questi ultimi saranno in grado di accettare/rifiutare. Avere un determinato studente come amico comporterà la possibilità di poter visualizzare la timeline (cronologia dei messaggi effettuati per ogni corso) di tale utente e l'utilizzo di una chat privata con tale studente.

2.2.6 FR6

Corsi degli amici: uno studente è in grado di sfogliare tutti i corsi seguiti dai suoi amici e non (a meno che non ci sia un'impostazione di privacy che non permette a studenti non amici di accedere alla lista dei corsi frequentati).

2.2.7 FR7

Visualizzazione annunci: uno studente può visualizzare gli annunci dei corsi e commentarli (anche con allegati multimediali) o apprezzarli (like). Tali annunci non possono essere creati dagli studenti (i quali possono solo creare topic nelle sezioni adatte per ogni corso).

2.2.8 FR8

Pubblicazione aggiornamenti sulla timeline: uno studente è in grado di decidere se pubblicare o meno un determinato aggiornamento sulla sua timeline. I suoi amici vengono informati di tali aggiornamenti e possono commentarli (anche con allegati multimediali) e apprezzarli (like).

2.2.9 FR9

Visualizzazione del calendario dei corsi: uno studente può vedere tutti i corsi in un calendario.

2.3 Requisiti Non Funzionali

2.3.1 NFR1

Usabilità: l'app dovrebbe essere intuitiva da usare e l'interfaccia utente dovrebbe essere semplice da capire. Tutte le interazioni devono essere completate in meno di tre click.

2.3.2 NFR2

Conformità alle linee guida: la progettazione dell'App deve essere conforme alle linee guida sull'usabilità per il sistema operativo scelto.

2.3.3 NFR3

Piattaforma di destinazione: l'App deve essere sviluppata in Java (non obbligatorio).

2.3.4 NFR4

Sistema di back-end il cliente fornisce un sistema di back-end con un paio di servizi che devono essere utilizzati nell'App.

2.4 Modello del Sistema

Si procede con la successiva fase dell'analisi dei requisiti. Vengono definiti scenari, casi d'uso, diagramma delle classi e diagrammi delle sequenze. Per ogni macro-sezione logica dell'applicativo è stato creato un relativo package, ciascuno contenente le rispettive funzioni.

2.4.1 Scenari

Qui si elencano vari scenari, un istanza del caso d'uso che descrive un insieme concreto di azioni. Gli scenari sono usati come esempi per illustrare casi comuni; il loro focus è la comprensione da parte dell'utente. I casi d'uso sono invece usati per descrivere tutti i casi possibili, il loro focus è sulla completezza. Uno scenario viene descritto usando un template con tre campi:

- Nome dello scenario (per evitare ambiguità);
- Nome degli attori partecipanti nello scenario;
- Flusso di eventi di uno scenario che descrive gli eventi passo dopo passo (negli scenari non ci sono condizioni di entrata e di uscita).

<i>Nome scenario</i>	cercaAula
<i>Attori partecipanti</i>	raj:Studente
<i>Flusso eventi</i>	<ol style="list-style-type: none">1. Raj frequenta l'Università di Nowhere.2. Raj dovrà seguire il corso di Economia durante l'anno accademico.3. Raj non ha mai visitato prima l'edificio in questione.4. Raj apre l'App, accede alla home page ed effettua la ricerca del corso interessato attraverso la barra di ricerca.5. Raj trova il corso e cliccandovi su di esso riesce ad accedere alla relativa pagina.6. Raj legge gli orari dei corsi e la posizione sulla mappa.

Tabella 2.1: Scenario *cercaAula*

<i>Nome scenario</i>	conosciStudente
<i>Attori partecipanti</i>	raj:Studente, penny:Studente
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Raj frequenta il corso di Economia Aziendale all'Università di Nowhere. 2. Raj visita la pagina del corso ed entra in contatto con altri utenti frequentanti. 3. Raj legge i loro commenti sui vari topic del corso. 4. Raj mette mi piace al commento "Grandi esercizi" di Penny, anche lei studia Informatica. 5. Raj riconosce Penny dalla suo foto profilo e le manda una richiesta di amicizia. 6. Raj aggiunge un nuovo commento sulle domande d'esame delle sessioni precedenti. 7. Penny viene informata della richiesta di amicizia e la conferma. 8. Raj, a sua volta, viene informato che Penny ha accettato la sua richiesta. 9. Raj controlla tutti i corsi che Penny sta visitando. 10. Raj trova un altro corso interessante che vuole seguire ("Contabilità") e lo salva nella sua lista dei corsi.

Tabella 2.2: Scenario *conosciStudente*

<i>Nome scenario</i>	richiestaMateriale
<i>Attori partecipanti</i>	howard, sheldon bernadette:Studente
<i>Flusso eventi</i>	<ol style="list-style-type: none"> Howard sta studiando per l'esame di Informatica. Howard si rende conto di non avere fra gli appunti la tabella delle sommatorie note. Howard allora contatta Sheldon, di cui è amico, tramite la chat dell'App per chiedergli se possiede quegli appunti. Sheldon risponde che non ha gli appunti. Howard decide di andare sulla pagina del corso. Howard cerca fra la lista del materiale pubblicato gli appunti, ma non li trova. Howard apre un topic nella pagina del corso di Informatica "Qualcuno ha appunti sulle sommatorie note?" Bernadette commenta che a breve pubblicherà gli appunti che ha chiesto. Howard trova gli appunti appena pubblicati da Bernadette e cancella il topic in cui li richiede.

Tabella 2.3: Scenario *richiestaMateriale*

<i>Nome scenario</i>	notificaCommentoAlTopic
<i>Attori partecipanti</i>	sheldon, raj:Studente
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Sheldon sta navigando sull'App Uni-Way. 2. Sheldon riceve una notifica sul cellulare dall'App. 3. Sheldon nota che la notifica è relativa ad una sua precedente domanda "Quali sono le domande più frequenti sulla segmentazione?". 4. Sheldon clicca sulla notifica e visualizza la risposta di Raj che gli suggerisce alcuni argomenti. 5. Sheldon lascia un like alla risposta e scrive un commento in cui ringrazia Raj.

Tabella 2.4: Scenario *notificaCommentoAlTopic*

<i>Nome scenario</i>	visualizzaAnnuncioUni
<i>Attori partecipanti</i>	sheldon:Studente
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Sheldon sta navigando sull'App. 2. Sheldon riceve una notifica sul cellulare dall'App. 3. Sheldon legge che la notifica è relativa al corso di Ingegneria del Software. 4. Sheldon clicca sulla notifica e viene reindirizzato ad un annuncio in cui si avvisano gli studenti che è stata posticipata di un'ora la lezione. 5. Sheldon apprezza il topic lasciando un like.

Tabella 2.5: Scenario *visualizzaAnnuncioUni*

<i>Nome scenario</i>	cambiaIdeaDiPrivacy
<i>Attori partecipanti</i>	howard:Studente
<i>Flusso eventi</i>	<ol style="list-style-type: none"> Howard sta navigando sull'App nella sezione del corso di Informatica. Howard decide di lasciare un commento su un annuncio. Howard, in seguito, ritornando sul suo profilo nota che tutta la sua cronologia di commenti nella sua timeline è visibile a qualsiasi studente. Howard decide quindi di modificare le impostazioni relative alla privacy nel suo profilo navigando nella sezione impostazioni e cambiandole in modo tale da consentire solo ai suoi amici di visualizzare la sua timeline. Howard, inoltre, decide di rendere visibile la sua lista degli amici solo ai suoi amici.

Tabella 2.6: Scenario *cambiaIdeaDiPrivacy*

<i>Nome scenario</i>	entraInUniWay
<i>Attori partecipanti</i>	penny:Studente
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Penny scarica l'App Uni-Way dallo store e aprendola visualizza la pagina di benvenuto. 2. Penny inserisce le credenziali (matricola e password dell'Università). 3. L'App mostra una nuova pagina in cui invita alla registrazione dell'account Uni-Way. 4. Penny inserisce nome, cognome, sesso, immagine, numero di telefono, e-mail e password. 5. Penny viene reindirizzata sulla sua timeline e legge il suo primo post automatico "Penny si è appena registrata".

Tabella 2.7: Scenario *entraInUniWay*

<i>Nome scenario</i>	cambioTelefono
<i>Attori partecipanti</i>	penny:Studente
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Penny cambia smartphone e scarica l'App di Uni-Way dallo store. 2. Penny avvia l'App per la prima volta sul nuovo dispositivo e si accorge di aver dimenticato le sue credenziali. 3. Penny clicca quindi sul tasto di recupero password ed inserisce la mail con cui si è registrata la prima volta. 4. Penny accede alla sua mail personale e cliccando sul link di recupero password accede ad un sito dove è in grado di inserirne una nuova. 5. Penny riaccede quindi all'App inserendo la nuova password. Effettuato l'accesso ritorna alle impostazioni e riabilita l'accesso automatico.

Tabella 2.8: Scenario *cambioTelefono*

2.4.2 Casi D'Uso

In questa sezione saranno mostrati tutti i casi d'uso rilevati durante la fase di scoperta dei requisiti. Tali CdU sono stati opportunamente divisi in package, in base alla logica delle azioni che rappresentano. I package in questione sono:

- Eccezioni
- In-Out
- Modifica
- Corso
- Ricerca
- Social
- Topic e Commenti
- Notifica

Prima di introdurre i package uno ad uno, si mostra di seguito il CdU **Connessione Persa**, il quale estende tutti gli altri. Esso, infatti, entra in gioco ogni volta che si perde la connessione, essendo l'App usufruibile solo con una connessione costante. Nelle condizioni d'entrata di ogni CdU verrà quindi omessa tale informazione.

<i>Nome caso d'uso</i>	ConnessionePersa
<i>Attori partecipanti</i>	Comunica con Studente
<i>Flusso eventi</i>	1. L'App mostra un messaggio d'errore che spiega che l'operazione che si stava eseguendo non è andata a buon fine a causa della perdita di connessione.
<i>Condizioni di entrata</i>	Questo caso d'uso estende tutti gli altri casi d'uso. Viene inizializzato dal sistema quando il dispositivo non è più connesso alla rete.
<i>Condizioni di uscita</i>	...
<i>Requisiti di qualità</i>	...

Tabella 2.9: Caso d'uso *ConnessionePersa*

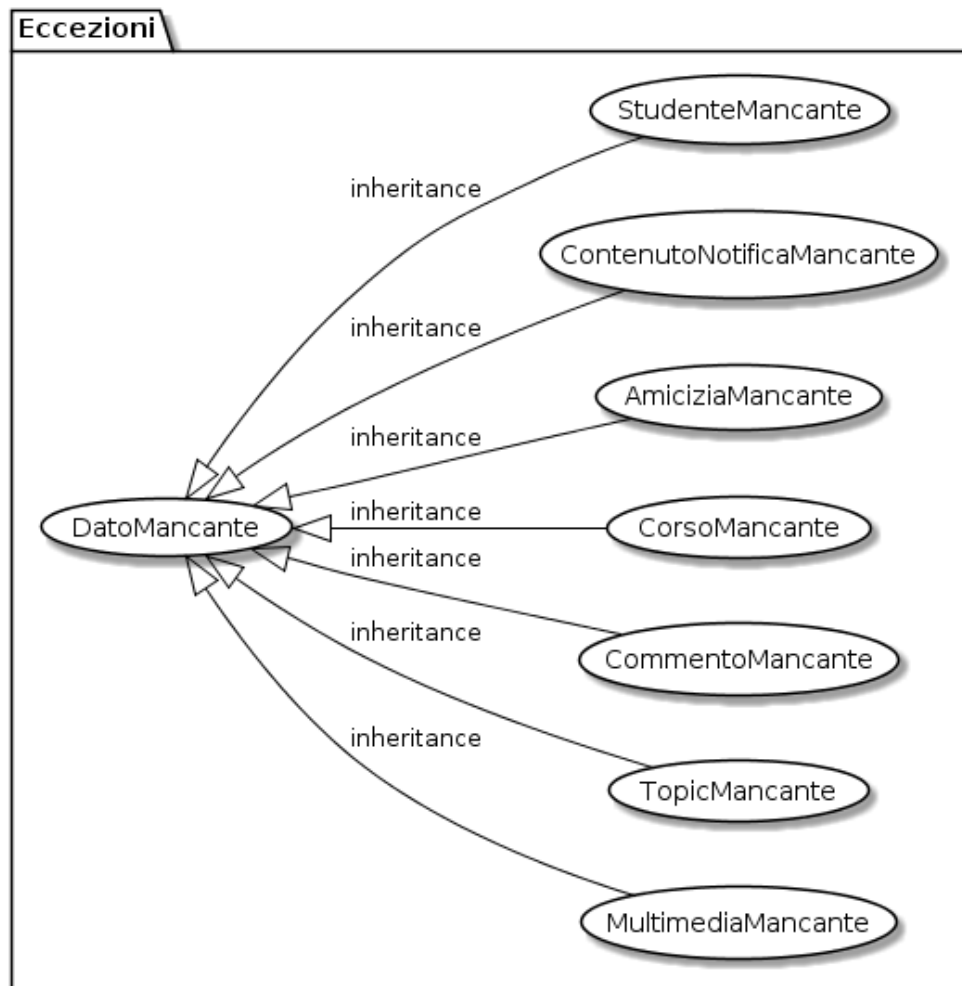
Package Eccezioni

Figura 2.1: Diagramma dei casi d'uso del package Eccezioni.

In questo package sono stati raggruppati tutti i CdU rappresentanti i casi eccezionali rilevati durante la fase di scoperta dei requisiti. In sostanza i casi d'uso in questione rappresentano delle eccezioni legate ai dati mancanti, dovuti a problemi di sincronizzazione (es: creazione di un commento in un topic che è stato appena cancellato).

<i>Nome caso d'uso</i>	DatoMancante
<i>Attori partecipanti</i>	Studente, Iniziato da Studente
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Studente richiede l'accesso ad un dato che non è più disponibile. 2. L'App mostra un apposito messaggio d'errore in base al tipo di dato mancante. 3. Studente clicca "Ok" sul messaggio di errore.
<i>Condizioni di entrata</i>	<ul style="list-style-type: none"> • Studente deve aver effettuato il login; AND • Studente deve effettuare un'operazione che porti ad un problema di sincronizzazione riguardante un dato mancante (vedi CdU).
<i>Condizioni di uscita</i>	L'utente visualizzato il messaggio d'errore torna alla schermata precedente.
<i>Requisiti di qualità</i>	L'App notifica il messaggio d'errore entro 3 secondi.

Tabella 2.10: Caso d'uso *DatoMancante*

<i>Nome caso d'uso</i>	MultimediaMancante
<i>Attori partecipanti</i>	Ereditato da DatoMancante
<i>Flusso eventi</i>	<p>1. Studente accede ad un determinato <i>corso/topic/commento</i> a cui è allegato un file multimediale di suo interesse, cliccando su quest'ultimo per visualizzarne l'anteprima.</p> <p>2. L'App non effettua l'operazione, perché nel lasso di tempo in cui Studente ha deciso di cliccare sul file, il proprietario dello stesso lo ha cancellato. Quindi l'App mostra un messaggio con il relativo errore.</p>
<i>Condizioni di entrata</i>	Ereditato da DatoMancante
<i>Condizioni di uscita</i>	Ereditato da DatoMancante
<i>Requisiti di qualità</i>	Ereditato da DatoMancante

Tabella 2.11: Caso d'uso *MultimediaMancante*

<i>Nome caso d'uso</i>	ContenutoNotificaMancante
<i>Attori partecipanti</i>	Ereditato da DatoMancante
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. L'App invia una notifica push a Studente. 2. Studente vede l'anteprima della notifica nella sezione dedicata del suo dispositivo. 3. Studente clicca sulla notifica per visualizzarne l'intero contenuto. 4. L'App non reindirizza alla pagina relativa al contenuto della notifica, perché è stato cancellato dal creatore dello stesso. L'App mostra un messaggio con il relativo errore.
<i>Condizioni di entrata</i>	Ereditato da DatoMancante
<i>Condizioni di uscita</i>	Ereditato da DatoMancante
<i>Requisiti di qualità</i>	Ereditato da DatoMancante

Tabella 2.12: Caso d'uso *ContenutoNotificaMancante*

<i>Nome caso d'uso</i>	AmiciziaMancante
<i>Attori partecipanti</i>	Ereditato da DatoMancante
<i>Flusso eventi</i>	<p>1. Studente visualizza un altro utente nella sua lista degli amici e decide di effettuare una qualsivoglia di queste 3 operazioni cliccando sull'apposito tasto:</p> <ul style="list-style-type: none"> • visualizzare i corsi a cui partecipa; • avviare una chat con lui; • rimuoverlo dalla lista degli amici. <p>2. L'App non effettua l'operazione, perché nel lasso di tempo in cui Studente ha deciso di cliccare sul tasto, l'amico lo ha rimosso dalla lista degli amici. Quindi l'App mostra un messaggio con il relativo errore.</p>
<i>Condizioni di entrata</i>	Ereditato da DatoMancante
<i>Condizioni di uscita</i>	Ereditato da DatoMancante
<i>Requisiti di qualità</i>	Ereditato da DatoMancante

Tabella 2.13: Caso d'uso *AmiciziaMancante*

<i>Nome caso d'uso</i>	CorsoMancante
<i>Attori partecipanti</i>	Ereditato da DatoMancante
<i>Flusso eventi</i>	<p>1. Studente accede ad un determinato corso e decide di effettuare una qualsivoglia di queste 2 operazioni cliccando sull'apposito tasto:</p> <ul style="list-style-type: none"> • seguire il corso (se non lo segue già); • visualizzare le info del corso. <p>2. L'App non effettua l'operazione, perché nel lasso di tempo in cui Studente ha deciso di cliccare sul tasto, il corso è stato eliminato dalla lista dei corsi disponibili. L'App mostra un messaggio con il relativo errore.</p>
<i>Condizioni di entrata</i>	Ereditato da DatoMancante
<i>Condizioni di uscita</i>	Ereditato da DatoMancante
<i>Requisiti di qualità</i>	Ereditato da DatoMancante

Tabella 2.14: Caso d'uso *CorsoMancante*

<i>Nome caso d'uso</i>	CommentoMancante
<i>Attori partecipanti</i>	Ereditato da DatoMancante
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Studente accede alla pagina di un determinato corso e clicca su un topic di suo interesse. 2. L'App mostra il determinato topic. 3. Studente legge i vari commenti legati al topic e notando uno in particolare decide di apprezzarlo lasciando un like. 4. L'App non registra l'operazione, in quanto, nel lasso di tempo in cui Studente stava visitando il topic, il proprietario ha cancellato il proprio commento. L'App mostra un messaggio col relativo errore.
<i>Condizioni di entrata</i>	Ereditato da DatoMancante
<i>Condizioni di uscita</i>	Ereditato da DatoMancante
<i>Requisiti di qualità</i>	Ereditato da DatoMancante

Tabella 2.15: Caso d'uso *CommentoMancante*

<i>Nome caso d'uso</i>	TopicMancante
<i>Attori partecipanti</i>	Ereditato da DatoMancante
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Studente accede alla pagina di un determinato corso e clicca su un topic di suo interesse. 2. L'App mostra il determinato topic. 3. Studente effettua qualsivoglia di queste 3 operazioni: <ul style="list-style-type: none"> • aggiunge un commento; • metti like al topic/commento; • cancella un commento. 4. L'App non registra l'operazione, in quanto, nel lasso di tempo in cui Studente stava visitando il topic, il proprietario ha cancellato il proprio topic. L'App mostra un messaggio col relativo errore.
<i>Condizioni di entrata</i>	Ereditato da DatoMancante
<i>Condizioni di uscita</i>	Ereditato da DatoMancante
<i>Requisiti di qualità</i>	Ereditato da DatoMancante

Tabella 2.16: Caso d'uso *TopicMancante*

<i>Nome caso d'uso</i>	StudenteMancante
<i>Attori partecipanti</i>	Ereditato da DatoMancante
<i>Flusso eventi</i>	<p>1. Studente visualizza il nome di un altro studente, cliccando su quest'ultimo per visualizzare le informazioni del profilo a cui ha accesso (amico oppure no).</p> <p>2. L'App non effettua l'operazione, perché nel lasso di tempo in cui Studente ha deciso di cliccare sul nome dell'altro studente, quest'ultimo è stato rimosso da uniway. Quindi l'App mostra un messaggio con il relativo errore.</p>
<i>Condizioni di entrata</i>	Ereditato da DatoMancante
<i>Condizioni di uscita</i>	Ereditato da DatoMancante
<i>Requisiti di qualità</i>	Ereditato da DatoMancante

Tabella 2.17: Caso d'uso *StudenteMancante*

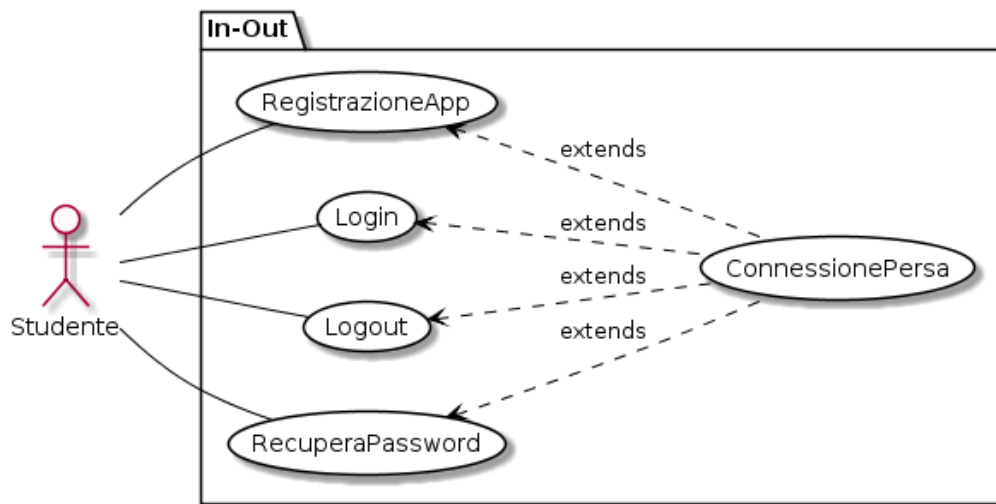
Package In-Out

Figura 2.2: Diagramma dei casi d'uso del package In-Out.

Nel package *In-Out* sono stati inseriti tutti i casi d'uso relativi alle operazioni utili per l'accesso. Troviamo **RegistrazioneApp**, **RecuperaPassword**, **Login** e **Logout**. Non sono previste eccezioni in quanto eventuali problemi, come credenziali già utilizzate, non necessitano di un CdU specifico.

<i>Nome caso d'uso</i>	RegistrazioneApp
<i>Attori partecipanti</i>	Studente, Iniziato da Studente
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Studente accede alla pagina di benvenuto. 2. L'App mostra la pagina di benvenuto, con il form per inserire le credenziali, già note, dell'Università e le credenziali per il nuovo account Uni-Way. 3. Studente inserisce le credenziali. 4. L'App controlla la correttezza di entrambe e registra lo studente.
<i>Condizioni di entrata</i>	<ul style="list-style-type: none"> • Studente deve aver avviato l'App per la prima volta; AND • Studente deve essere uno studente immatricolato all'Università di Nowhere.
<i>Condizioni di uscita</i>	Studente viene registrato correttamente e reindirizzato alla sua timeline.
<i>Requisiti di qualità</i>	L'App crea un nuovo account e reindirizza alla timeline in un tempo stimato massimo di 5 secondi (considerando una qualità ottima della connessione).

Tabella 2.18: Caso d'uso *RegistrazioneApp*

<i>Nome caso d'uso</i>	Login
<i>Attori partecipanti</i>	Studente, Iniziato da Studente
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Studente inserisce e-mail e password nella pagina per l'accesso. 2. L'App controlla e accetta le credenziali, poi reindirizza l'utente alla sua homepage.
<i>Condizioni di entrata</i>	<ul style="list-style-type: none"> • Studente deve essersi già registrato all'App in precedenza; AND • Studente non deve già aver effettuato il login.
<i>Condizioni di uscita</i>	Studente viene autenticato correttamente e reindirizzato alla sua timeline.
<i>Requisiti di qualità</i>	L'App autentica l'utente e lo reindirizza alla timeline in un tempo stimato massimo di 5 secondi (considerando una qualità ottima della connessione).

Tabella 2.19: Caso d'uso *Login*

<i>Nome caso d'uso</i>	Logout
<i>Attori partecipanti</i>	Studente, Iniziato da Studente
<i>Flusso eventi</i>	<ol style="list-style-type: none">1. Studente clicca sul pulsante di logout.2. L'App chiude tutte le operazioni in corso e chiude la sessione dell'utente.
<i>Condizioni di entrata</i>	Studente deve essere loggato.
<i>Condizioni di uscita</i>	Studente viene disconnesso correttamente e reindirizzato alla pagina di login.
<i>Requisiti di qualità</i>	L'App completa le operazioni in un tempo stimato massimo di 3 secondi (considerando una qualità ottima della connessione).

Tabella 2.20: Caso d'uso *Logout*

<i>Nome caso d'uso</i>	RecuperaPassword
<i>Attori partecipanti</i>	Studente, Iniziato da Studente
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Studente visualizza la pagina di login. 2. L'App mostra i campi per inserire email e password. 3. Studente clicca sul pulsante "Recupera password". 4. L'App mostra un campo per inserire l'email di registrazione. 5. Studente inserisce l'email con la quale si è registrato. 6. L'App invia un'email con le opzioni di ripristino password.
<i>Condizioni di entrata</i>	<ul style="list-style-type: none"> • Studente deve essere registrato all'App; AND • Studente non deve già aver effettuato il login.
<i>Condizioni di uscita</i>	Studente riceve l'email.
<i>Requisiti di qualità</i>	L'App deve inviare un'e-mail di recupero password entro 5 minuti dalla richiesta.

Tabella 2.21: Caso d'uso *RecuperaPassword*

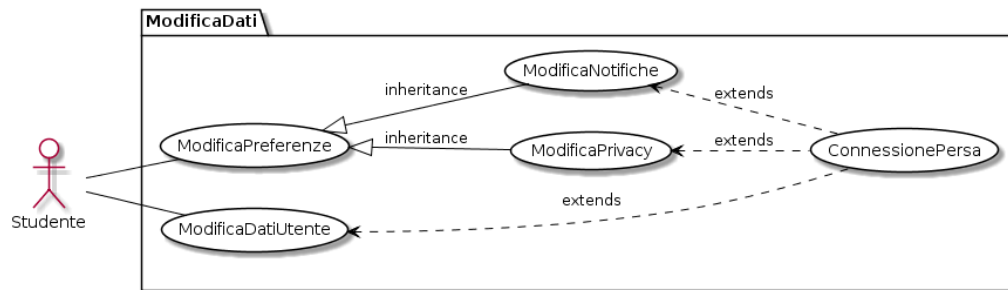
Package Modifica

Figura 2.3: Diagramma dei casi d'uso del package Modifica.

Nel package *Modifica* sono stati inseriti tutti i casi d'uso relativi alle operazioni che un utente può compiere per modificare i dati personali e le preferenze delle notifiche e della privacy.

Troviamo rispettivamente **ModificaDatiUtenti** e **ModificaPreferenze**, il cui comportamento viene differentemente definito in **ModificaNotifiche** e **ModificaPrivacy**, in base alla tipologia di preferenza in esame. Sono stati creati due casi d'uso differenti per la diversa natura delle informazioni da salvare.

<i>Nome caso d'uso</i>	ModificaPreferenze
<i>Attori partecipanti</i>	Studente, Iniziato da Studente
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Studente clicca sulla sezione delle impostazioni. 2. L'App mostra la sezione delle impostazioni. 3. Studente accede alla sottosezione "Modifica preferenze". 4. L'App mostra il menu "Modifica preferenze". 5. Studente modifica le diverse impostazioni. 6. L'App registra le modifiche.
<i>Condizioni di entrata</i>	Studente deve aver effettuato il login.
<i>Condizioni di uscita</i>	Studente legge un messaggio di corretta modifica dei dati.
<i>Requisiti di qualità</i>	L'App deve modificare le impostazioni on-the-fly.

Tabella 2.22: Caso d'uso *ModificaPreferenze*

<i>Nome caso d'uso</i>	ModificaNotifiche
<i>Attori partecipanti</i>	Ereditato da ModificaPreferenze
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Studente clicca sulla sezione delle impostazioni. 2. L'App mostra la sottosezione delle impostazioni. 3. Studente accede al menu "Modifica notifiche". 4. L'App mostra il menu "Modifica notifiche". 5. Studente seleziona che tipo di notifiche desidera ricevere. 6. L'App registra le modifiche.
<i>Condizioni di entrata</i>	Ereditato da ModificaPreferenze.
<i>Condizioni di uscita</i>	Ereditato da ModificaPreferenze.
<i>Requisiti di qualità</i>	Ereditato da ModificaPreferenze

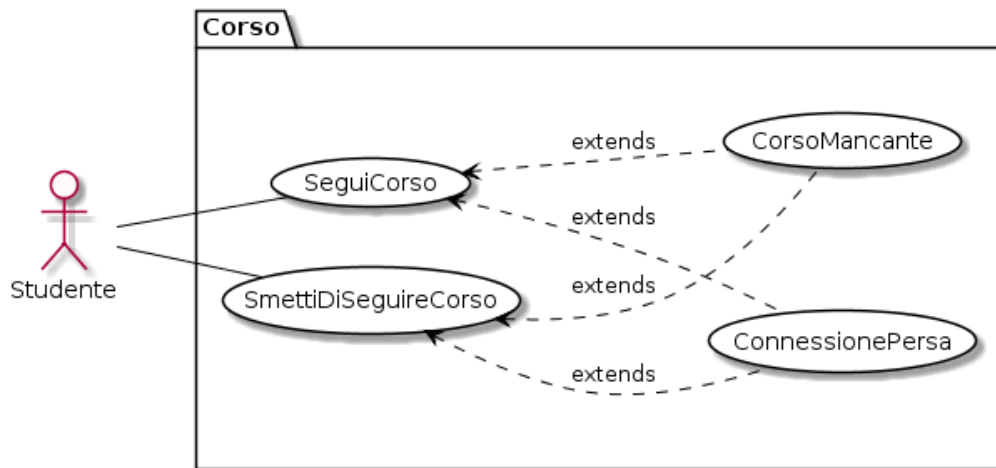
Tabella 2.23: Caso d'uso *ModificaNotifiche*

<i>Nome caso d'uso</i>	ModificaPrivacy
<i>Attori partecipanti</i>	Ereditato da ModificaPreferenze.
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Studente clicca sulla sezione delle impostazioni. 2. L'App mostra la sezione delle impostazioni. 3. Studente accede al menu "Modifica privacy". 4. L'App mostra il menu "Modifica privacy". 5. Studente modifica le diverse impostazioni di privacy in base alle proprie preferenze. 6. L'App registra le modifiche.
<i>Condizioni di entrata</i>	Ereditato da ModificaPreferenze.
<i>Condizioni di uscita</i>	Ereditato da ModificaPreferenze.
<i>Requisiti di qualità</i>	Ereditato da ModificaPreferenze.

Tabella 2.24: Caso d'uso *ModificaPrivacy*

<i>Nome caso d'uso</i>	ModificaDatiUtente
<i>Attori partecipanti</i>	Studente, Iniziato da Studente
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Studente clicca sulla sottosezione delle impostazioni. 2. L'App mostra la sezione delle impostazioni. 3. Studente accede al menu "Modifica dati personali". 4. L'App mostra il menu "Modifica dati personali". 5. Studente modifica i dati. 6. L'App registra la modifica.
<i>Condizioni di entrata</i>	Studente deve aver effettuato il login.
<i>Condizioni di uscita</i>	Studente legge un messaggio di corretta modifica dei dati.
<i>Requisiti di qualità</i>	L'App deve modificare le impostazioni on-the-fly.

Tabella 2.25: Caso d'uso *ModificaDatiUtente*

Package CorsoFigura 2.4: Diagramma dei casi d'uso, sezione relativa ai *corsi*.

Nel package *Corso* sono stati inseriti i casi d'uso che riguardano le operazioni che permettono allo studente di interagire con i corsi. I CdU in questione sono **SmettiDiSeguireCorso** e **SeguiCorso** che permettono le omonime operazioni.

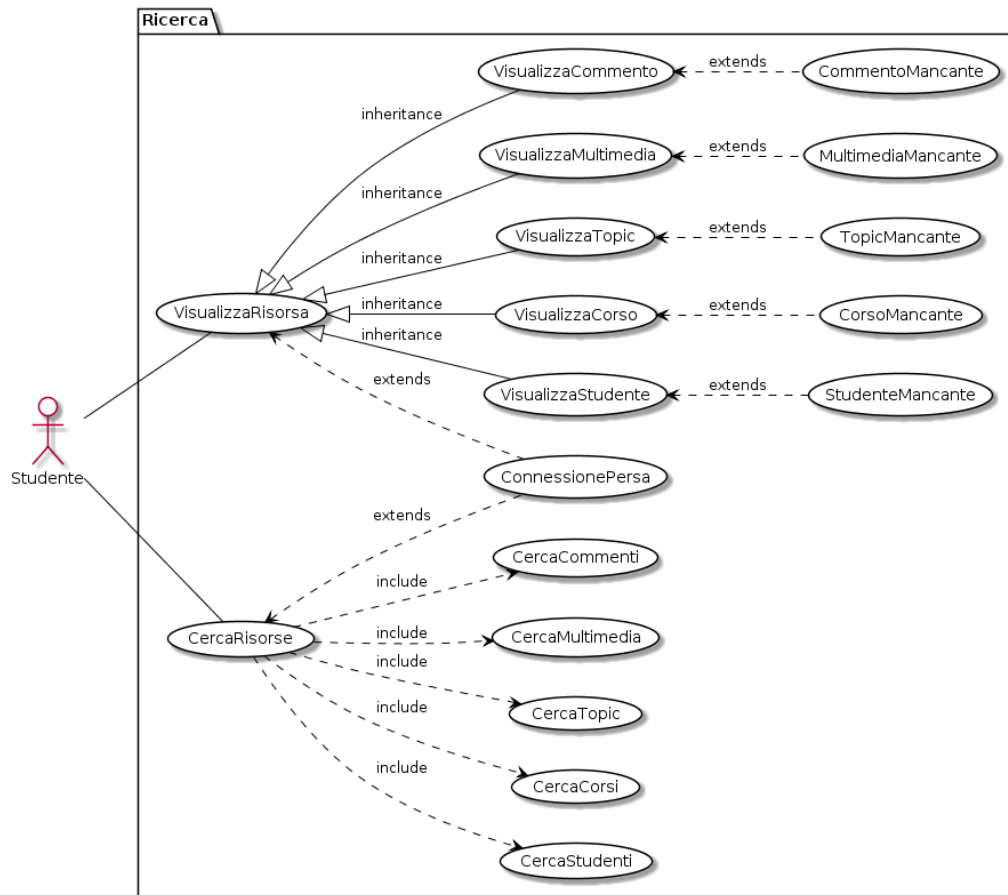
<i>Nome caso d'uso</i>	SeguiCorso
<i>Attori partecipanti</i>	Studente, Iniziato da Studente
<i>Flusso eventi</i>	<p>1. Studente visualizza il corso a cui vuole iscriversi con l'apposito pulsante "Segui Corso" di fianco che clicca.</p> <p>2. L'App mostra un messaggio di avvenuta registrazione al corso.</p>
<i>Condizioni di entrata</i>	<ul style="list-style-type: none"> • Studente non deve essere già iscritto al corso; <p>AND</p> <ul style="list-style-type: none"> • Studente deve aver effettuato il login.
<i>Condizioni di uscita</i>	L'App registra l'avvenuta iscrizione dello studente al corso.
<i>Requisiti di qualità</i>	L'App registra l'avvenuta iscrizione in 3 secondo.

Tabella 2.26: Caso d'uso *SeguiCorso*

<i>Nome caso d'uso</i>	SmettiDiSeguireCorso
<i>Attori partecipanti</i>	Studente, Iniziato da Studente
<i>Flusso eventi</i>	<p>1. Studente visualizza sul display il corso da cui vuole disiscriversi con l'apposito pulsante "Smetti di seguire" di fianco che clicca.</p> <p>2. L'App conferma l'avvenuta rimozione.</p>
<i>Condizioni di entrata</i>	<ul style="list-style-type: none"> • Studente deve essere già iscritto al corso; AND • Studente deve aver effettuato il login.
<i>Condizioni di uscita</i>	L'App registra l'avvenuta rimozione dello studente dal corso.
<i>Requisiti di qualità</i>	L'App registra l'avvenuta iscrizione in 3 secondi.

Tabella 2.27: Caso d'uso *SmettiDiSeguireCorso*

Package Ricerca

Figura 2.5: Diagramma dei casi d'uso relativi alle funzionalità di *ricerca*.

Nel package *Ricerca* sono stati inseriti i casi d'uso che permettono allo studente di cercare e visualizzare le risorse nel sistema.

I CdU in questione sono: **CercaRisorse** che include **CercaCommenti**, **CercaMultimedia**, **CercaPost**, **CercaStudenti** e **CercaCorsi** che sono tra di loro simili ma con filtro differente, perciò verrà preso in analisi solo **CercaCommenti** come esempio.

VisualizzaRisorse che viene ereditato da **VisualizzaCommento**, **VisualizzaCorso**, **VisualizzaPost**, **VisualizzaMultimedia** e **VisualizzaStu-**

dente che indicano le operazioni da eseguire nel caso un utente clicchi su una specifica risorsa.

<i>Nome caso d'uso</i>	CercaRisorse
<i>Attori partecipanti</i>	Studente, Iniziato da Studente
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Studente inserisce il nome della risorsa da lui cercata nel campo di ricerca dell'App (che sia essa un semplice commento, un corso, un post o un file multimediale), opportunamente aiutandosi con dei filtri, i quali comportano l'inclusione del relativo caso d'uso. 2. L'App mostra il risultato della ricerca.
<i>Condizioni di entrata</i>	Studente deve aver effettuato il login.
<i>Condizioni di uscita</i>	<ul style="list-style-type: none"> • L'App mostra a schermo il risultato della ricerca effettuata dall'utente; OR • L'App mostra a schermo un messaggio che indica l'assenza di risultati nella ricerca effettuata.
<i>Requisiti di qualità</i>	L'App mostra gli elementi della ricerca entro 3 secondi.

Tabella 2.28: Caso d'uso *CercaRisorse*

<i>Nome caso d'uso</i>	CercaCommenti
<i>Attori partecipanti</i>	Studente, Iniziato da Studente
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Studente inserisce il nome della risorsa da lui cercata nel campo di ricerca dell'App, selezionando il filtro "Commenti" 2. L'App mostra il risultato della ricerca.
<i>Condizioni di entrata</i>	Studente deve aver effettuato il login.
<i>Condizioni di uscita</i>	<ul style="list-style-type: none"> • L'App mostra a schermo il risultato della ricerca effettuata dall'utente filtrando i commenti che hanno il nome ricercato dallo Studente; <p>OR</p> <ul style="list-style-type: none"> • L'App mostra a schermo un messaggio che indica l'assenza di risultati nella ricerca effettuata.
<i>Requisiti di qualità</i>	L'App mostra gli elementi della ricerca entro 3 secondi.

Tabella 2.29: Caso d'uso *CercaCommenti*

<i>Nome caso d'uso</i>	VisualizzaRisorsa
<i>Attori partecipanti</i>	Studente, Iniziato da Studente
<i>Flusso eventi</i>	<p>1. Studente clicca sulla risorsa che ha intenzione di visualizzare.</p> <p>2. L'App si occupa di caricare la risorsa e viene avviato un diverso caso d'uso in base alla risorsa cliccata.</p>
<i>Condizioni di entrata</i>	Studente deve aver effettuato il login.
<i>Condizioni di uscita</i>	Lo Studente visualizza la risorsa.
<i>Requisiti di qualità</i>	L'App carica la risorsa in 3 secondi.

Tabella 2.30: Caso d'uso *VisualizzaRisorsa*

<i>Nome caso d'uso</i>	VisualizzaCorso
<i>Attori partecipanti</i>	Ereditato da VisualizzaRisorsa
<i>Flusso eventi</i>	<p>1. Studente clicca sul nome o immagine del corso che ha intenzione di visualizzare.</p> <p>2. L'App mostra la pagina del corso che lo studente ha selezionato.</p>
<i>Condizioni di entrata</i>	Ereditato da VisualizzaRisorsa .
<i>Condizioni di uscita</i>	Ereditato da VisualizzaRisorsa .
<i>Requisiti di qualità</i>	Ereditato da VisualizzaRisorsa

Tabella 2.31: Caso d'uso *VisualizzaCorso*

<i>Nome caso d'uso</i>	VisualizzaCommento
<i>Attori partecipanti</i>	Ereditato da VisualizzaRisorsa
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Studente clicca sul commento che ha intenzione di visualizzare 2. L'App mostra il Post che contiene il commento da visualizzare, centrando la visuale sul commento in questione.
<i>Condizioni di entrata</i>	Ereditato da VisualizzaRisorsa .
<i>Condizioni di uscita</i>	Ereditato da VisualizzaRisorsa .
<i>Requisiti di qualità</i>	Ereditato da VisualizzaRisorsa .

Tabella 2.32: Caso d'uso *VisualizzaCommento*

<i>Nome caso d'uso</i>	VisualizzaPost
<i>Attori partecipanti</i>	Ereditato da VisualizzaRisorsa
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Studente clicca sul Post che ha intenzione di visualizzare 2. L'App apre il Post selezionato.
<i>Condizioni di entrata</i>	Ereditato da VisualizzaRisorsa .
<i>Condizioni di uscita</i>	Ereditato da VisualizzaRisorsa .
<i>Requisiti di qualità</i>	Ereditato da VisualizzaRisorsa .

Tabella 2.33: Caso d'uso *VisualizzaPost*

<i>Nome caso d'uso</i>	VisualizzaMultimedia
<i>Attori partecipanti</i>	Ereditato da VisualizzaRisorsa
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Studente clicca sul file multimediale che ha intenzione di visualizzare. 2. L'App mostra il Post che contiene il file multimediale da visualizzare <i>oppure</i> l'App centra la visuale sul commento che contiene in allegato il file multimediale se quest'ultimo è stato caricato in un commento.
<i>Condizioni di entrata</i>	Ereditato da VisualizzaRisorsa .
<i>Condizioni di uscita</i>	Ereditato da VisualizzaRisorsa .
<i>Requisiti di qualità</i>	Ereditato da VisualizzaRisorsa .

Tabella 2.34: Caso d'uso *VisualizzaMultimedia*

<i>Nome caso d'uso</i>	VisualizzaStudente
<i>Attori partecipanti</i>	Ereditato da VisualizzaRisorsa
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Studente clicca sull'immagine del profilo o il nome dello studente che vuole visualizzare. 2. L'App apre il profilo dello studente selezionato.
<i>Condizioni di entrata</i>	Ereditato da VisualizzaRisorsa .
<i>Condizioni di uscita</i>	Ereditato da VisualizzaRisorsa .
<i>Requisiti di qualità</i>	Ereditato da VisualizzaRisorsa .

Tabella 2.35: Caso d'uso *VisualizzaStudente*

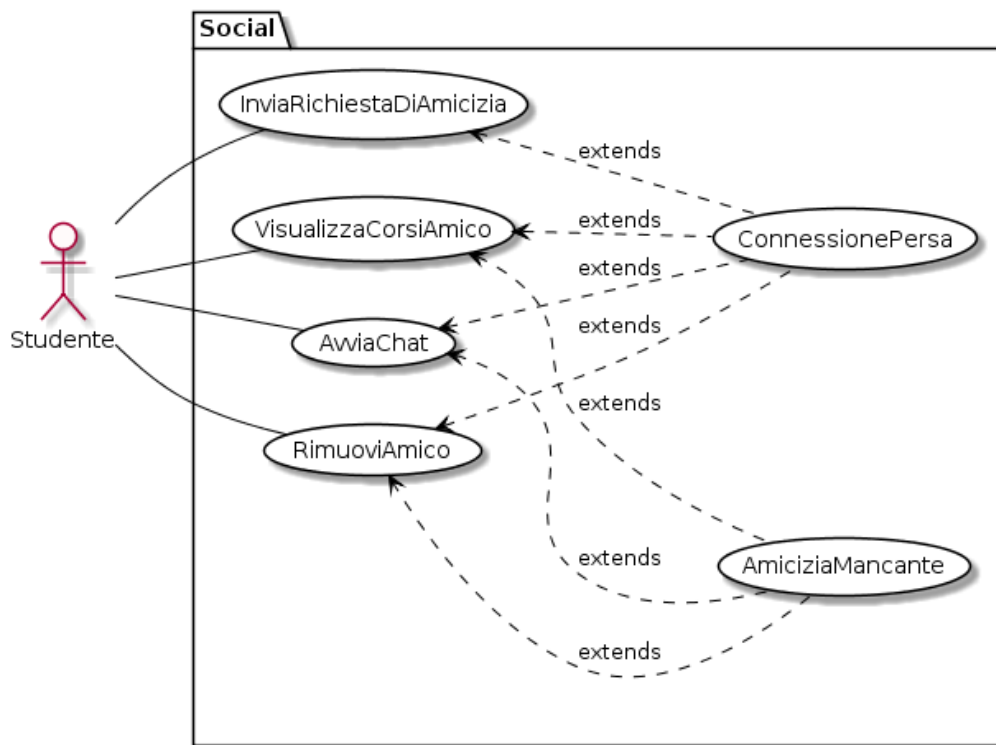
Package Social

Figura 2.6: Diagramma dei casi d'uso del package Social.

Nel package *Social* sono stati inseriti tutti i casi d'uso relativi alle funzionalità con aspetto social riscontrate durante la raccolta dei requisiti.

In questo package si trovano i casi d'uso: **InviaRichiestaDiAmicizia**, **VisualizzaCorsiAmico**, **RimuoviAmico** e **AvviaChat**.

A parte **InviaRichiestaDiAmicizia** gli altri casi d'uso vengono estesi tutti da **AmiciziaMancante**, CdU che si occupa di rappresentare il comportamento adottato dal sistema nel caso in cui una delle sopracitate operazioni, effettuabili dallo studente, venga avviata con un altro studente che non è presente nella sua lista amici.

<i>Nome caso d'uso</i>	InviaRichiestaDiAmicizia
<i>Attori partecipanti</i>	Studente1, Studente2, Iniziato da Studente1
<i>Flusso eventi</i>	<p>1. Studente1, a seguito di varie possibili operazioni come ricerca mirata per nome e lista di partecipanti ad un corso in comune, visualizza il nome di Studente2 e clicca sul pulsante "RichiediAmicizia".</p> <p>2. L'App mostra a Studente1 un messaggio che indica la buonuscita dell'operazione.</p>
<i>Condizioni di entrata</i>	<ul style="list-style-type: none"> • Studente1 deve aver effettuato il login; AND • Studente1 non deve essere già amico di Studente2.
<i>Condizioni di uscita</i>	L'App notifica Studente2 della richiesta d'amicizia di Studente1.
<i>Requisiti di qualità</i>	L'App invia la richiesta entro 5 secondi.

Tabella 2.36: Caso d'uso *InviaRichiestaDiAmicizia*

<i>Nome caso d'uso</i>	VisualizzaCorsiAmico
<i>Attori partecipanti</i>	Studente1, Studente2, Iniziato da Studente1
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Studente1 seleziona il profilo di Studente2. 2. L'App mostra il profilo di Studente2. 3. Studente1 seleziona la sezione relativa ai corsi frequentati da Studente2. 4. L'App mostra la sezione dei corsi frequentati da Studente2.
<i>Condizioni di entrata</i>	Studente1 deve aver effettuato il login.
<i>Condizioni di uscita</i>	<ul style="list-style-type: none"> • I corsi frequentati da Studente2 vengono visualizzati correttamente; OR • Viene visualizzato un messaggio che avvisa che Studente2 non è iscritto a nessun corso.
<i>Requisiti di qualità</i>	L'App mostra la lista dei corsi on-the-fly.

Tabella 2.37: Caso d'uso *VisualizzaCorsiAmico*

<i>Nome caso d'uso</i>	AvviaChat
<i>Attori partecipanti</i>	Studente, Iniziato da Studente
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Studente clicca sulla sezione relativa alle chat con gli amici. 2. L'App apre la sezione e mostra la lista degli amici. 3. Studente seleziona il nome dell'amico a cui intende scrivere. 4. L'App apre una chat con l'amico, recuperando tutti i messaggi precedenti. 5. Lo studente scrive il messaggio e clicca sull'apposito pulsante per inviare il messaggio. 6. L'App salva il messaggio e invia la relativa notifica al destinatario.
<i>Condizioni di entrata</i>	Studente1 deve aver effettuato il login.
<i>Condizioni di uscita</i>	Studente inizia una conversazione con l'amico.
<i>Requisiti di qualità</i>	Il sistema messaggistico dell'App deve garantire la consegna dei messaggi in un tempo massimo di 5 secondi.

Tabella 2.38: Caso d'uso *AvviaChat*

<i>Nome caso d'uso</i>	RimuoviAmico
<i>Attori partecipanti</i>	Studente1, Studente2, Iniziato da Studente1
<i>Flusso eventi</i>	<p>1. Studente1, a seguito di varie possibili operazioni come ricerca mirata per nome e lista di partecipanti ad un corso in comune, visualizza sul display il nome di Studente2 e clicca sul pulsante "RimuoviAmico".</p> <p>2. L'App mostra a Studente1 un messaggio che indica la buonuscita dell'operazione.</p>
<i>Condizioni di entrata</i>	<ul style="list-style-type: none"> • Studente1 deve essere amico di Studente2; <p>AND</p> <ul style="list-style-type: none"> • Studente1 deve aver effettuato il login.
<i>Condizioni di uscita</i>	<ul style="list-style-type: none"> • Studente2 viene tolto dalla lista degli amici di Studente1; OR • L'operazione non viene effettuata a causa di un errore di rete la cui descrizione viene mostrata a Studente1.
<i>Requisiti di qualità</i>	L'App effettua l'operazione entro 5 secondi.

Tabella 2.39: Caso d'uso *RimuoviAmico*

Package Topic-Commenti

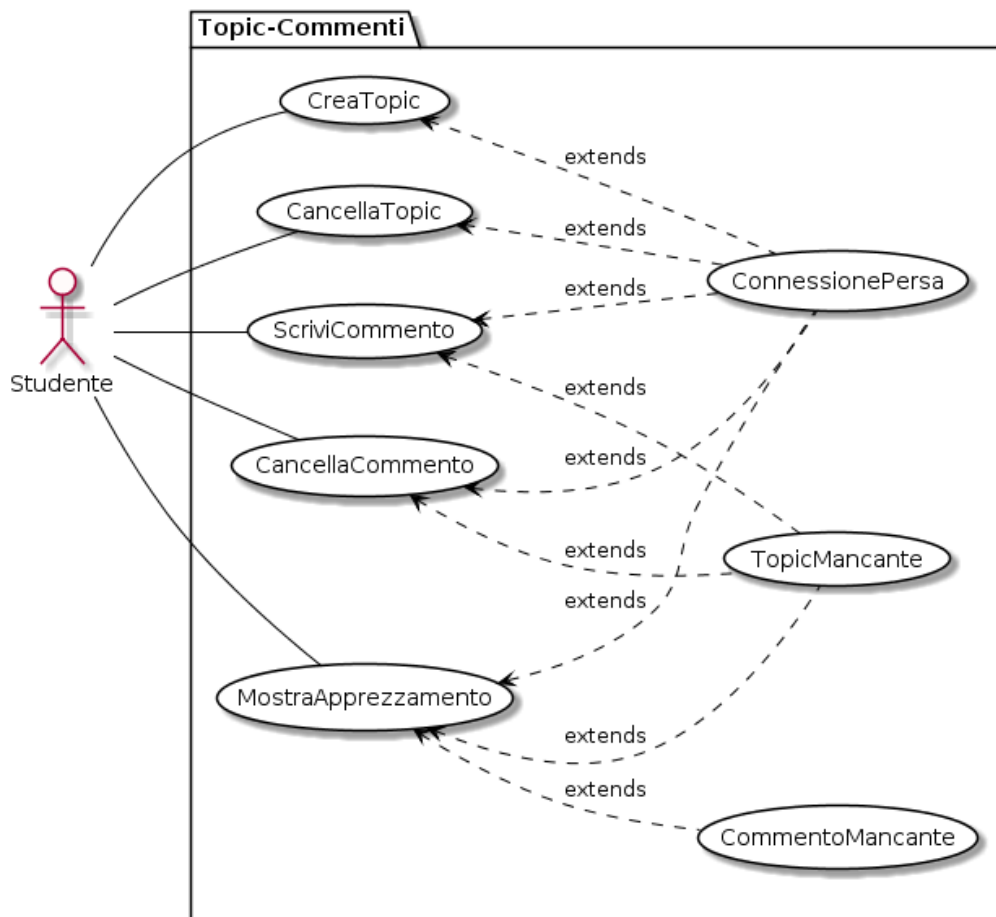


Figura 2.7: Diagramma dei casi d'uso, sezione relativa ai topic e commenti.

Il package *Topic-Commenti* prevede tutte le funzionalità e le eccezioni legate all'interazione di *Studente* con i topic, commenti e risorse legate ad un particolare corso. In questo package si trovano i casi d'uso: **CreaTopic**, **CancellaTopic**, **ScriviCommento**, **CancellaCommento** e **MostraApprezzamento**.

Il package **Eccezioni** si occuperà di *TopicMancante* e *CommentoMancante*. La buonuscita di ogni operazione dipenderà dalla connessione del dispositivo che dovrà essere costante durante l'utilizzo dell'applicativo. Le operazioni

per la cancellazione, creazione e apprezzamento di un topic o di un commento saranno legati invece alla persistenza di tali dati, e quindi a questioni di sincronizzazione.

<i>Nome caso d'uso</i>	MostraApprezzamento
<i>Attori partecipanti</i>	Studente, Iniziato da Studente
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Studente accede ad un topic dalla pagina di un corso o tramite ricerca/timeline. 2. L'App mostra il topic. 3. Studente clicca sull'apposito tasto per mostrare l'apprezzamento del topic o di un particolare commento (aggiunta del like). 4. L'App aggiunge il like.
<i>Condizioni di entrata</i>	Studente deve aver effettuato il login.
<i>Condizioni di uscita</i>	L'App registra l'operazione e aggiorna il contatore dei like.
<i>Requisiti di qualità</i>	L'App aggiorna il contatore dei like entro 3 secondi.

Tabella 2.40: Caso d'uso *MostraApprezzamento*

<i>Nome caso d'uso</i>	CreaTopic
<i>Attori partecipanti</i>	Studente, Iniziato da Studente
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Studente accede al corso interessato. 2. L'App mostra la pagina del corso. 3. Studente crea un topic nell'apposita sezione del corso tramite il tasto "Crea topic", inserendo la domanda o il materiale da condividere che aveva intenzione di pubblicare. 4. L'App crea infine il topic.
<i>Condizioni di entrata</i>	Studente deve aver effettuato il login.
<i>Condizioni di uscita</i>	Studente visualizza il topic correttamente creato dall'App.
<i>Requisiti di qualità</i>	L'App pubblica il topic entro 5 secondi.

Tabella 2.41: Caso d'uso *CreaTopic*

<i>Nome caso d'uso</i>	ScriviCommento
<i>Attori partecipanti</i>	Studente, Iniziato da Studente
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Studente accede al topic che vuole commentare. 2. L'App mostra il topic. 3. Studente posta il commento da lui scritto tramite il tasto "Inserisci commento" (che sia esso una semplice risposta o abbia come allegato un file multimediale). 4. L'App crea il commento.
<i>Condizioni di entrata</i>	Studente deve aver effettuato il login.
<i>Condizioni di uscita</i>	Studente visualizza il commento correttamente inserito dall'App.
<i>Requisiti di qualità</i>	L'App pubblica il commento entro 5 secondi.

Tabella 2.42: Caso d'uso *ScriviCommento*

<i>Nome caso d'uso</i>	CancellaTopic
<i>Attori partecipanti</i>	Studente, Iniziato da Studente
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Studente accede al topic che vuole cancellare. 2. L'App mostra il topic. 3. Studente cancella il topic tramite l'apposito tasto "Cancella topic". 4. L'App cancella il topic.
<i>Condizioni di entrata</i>	Studente deve aver effettuato il login.
<i>Condizioni di uscita</i>	Studente viene rimandato correttamente alla pagina del corso (dove era presente il topic).
<i>Requisiti di qualità</i>	L'App rende effettive le modifiche entro 5 secondi.

Tabella 2.43: Caso d'uso *CancellaTopic*

<i>Nome caso d'uso</i>	CancellaCommento
<i>Attori partecipanti</i>	Studente, Iniziato da Studente
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Studente accede al commento che vuole cancellare. 2. L'App mostra il topic contenente il commento che l'utente vuole cancellare. 3. Studente seleziona il commento da lui scritto e clicca sul tasto "Cancella commento" (che sia esso una semplice risposta o allegato ad un file multimediale). 4. L'App cancella il commento.
<i>Condizioni di entrata</i>	Studente deve aver effettuato il login.
<i>Condizioni di uscita</i>	Studente visualizza il topic senza il commento.
<i>Requisiti di qualità</i>	L'App rende effettive le modifiche entro 5 secondi.

Tabella 2.44: Caso d'uso *CancellaCommento*

Package Notifica

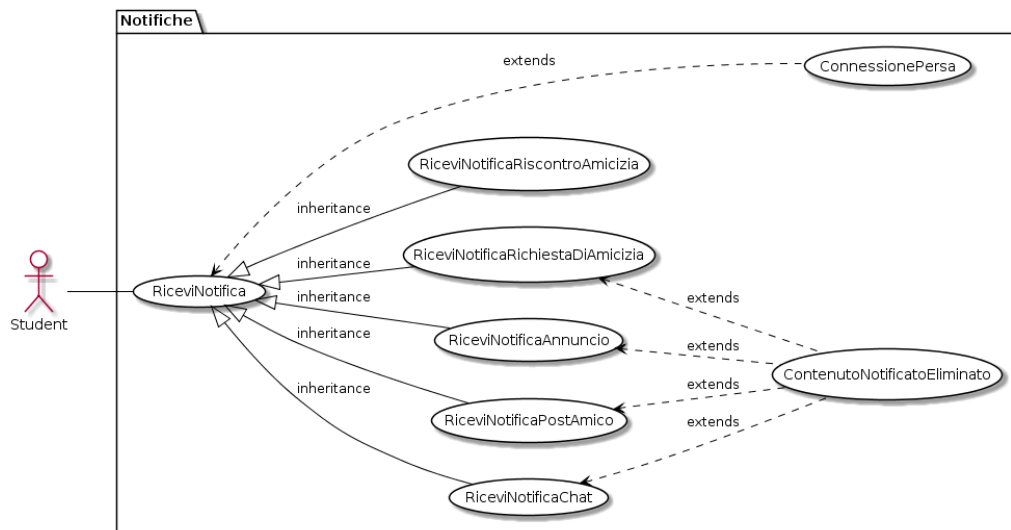


Figura 2.8: Diagramma dei casi d'uso, sezione relativa alle notifiche.

Nel package *Notifiche* sono stati inseriti tutti i casi d'uso relativi alla ricezione di notifiche. Il caso d'uso generale **RiceviNotifica** si specializza in **RiceviNotificaRichiestaDiAmicizia**, **RiceviNotificaAnnuncio**, **RiceviNotificaPostAmico**, **RiceviNotificaChat** e **RiceviNotificaRiscontroAmicizia**. Come si evince dal Diagramma dei casi d'uso, è stato individuato un CdU d'eccezione, ovvero **ContenutoNotificaMancante**, il quale si ricorda essere una specializzazione del caso d'uso **DatoMancante**, entrambi descritti nel package *Eccezioni*. Di seguito, i relativi CdU testuali.

<i>Nome caso d'uso</i>	RiceviNotifica
<i>Attori partecipanti</i>	Studente, Iniziato da Studente
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Studente riceve un aggiornamento in merito ad un determinato evento. 2. L'App mostra la notifica.
<i>Condizioni di entrata</i>	Studente deve aver effettuato il login.
<i>Condizioni di uscita</i>	<ul style="list-style-type: none"> • Studente ignora la notifica; OR • Studente clicca sulla notifica venendo reindirizzato alla pagina di competenza.
<i>Requisiti di qualità</i>	L'App manda la notifica entro un tempo limite di 5 secondi.

Tabella 2.45: Caso d'uso *RiceviNotifica*

<i>Nome caso d'uso</i>	RiceviNotificaRichiestaDiAmicizia
<i>Attori partecipanti</i>	Ereditato da RiceviNotifica.
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Studente riceve un aggiornamento dall'App relativa ad una richiesta d'amicizia. 2. L'App mostra la notifica allo studente. 3. Studente clicca sulla notifica. 4. L'App reindirizza Studente al profilo dello studente che gli ha inviato la richiesta.
<i>Condizioni di entrata</i>	Ereditato da RiceviNotifica.
<i>Condizioni di uscita</i>	Ereditato da RiceviNotifica.
<i>Requisiti di qualità</i>	Ereditato da RiceviNotifica.

Tabella 2.46: Caso d'uso *RiceviNotificaRichiestaDiAmicizia*

<i>Nome caso d'uso</i>	RiceviNotificaAnnuncio
<i>Attori partecipanti</i>	Ereditato da RiceviNotifica.
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Studente riceve un aggiornamento dall'App relativo ad un nuovo annuncio di un corso seguito. 2. L'App mostra la notifica allo studente. 3. Studente clicca sulla notifica. 4. L'App reindirizza Studente al topic dell'annuncio.
<i>Condizioni di entrata</i>	Ereditato da RiceviNotifica.
<i>Condizioni di uscita</i>	Ereditato da RiceviNotifica.
<i>Requisiti di qualità</i>	Ereditato da RiceviNotifica.

Tabella 2.47: Caso d'uso *RiceviNotificaAnnuncio*

<i>Nome caso d'uso</i>	RiceviNotificaPostAmico
<i>Attori partecipanti</i>	Ereditato da riceviNotifica
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Studente riceve un aggiornamento dall'App relativo ad un nuovo topic creato da un suo amico. 2. L'App mostra la notifica allo studente. 3. Studente clicca sulla notifica. 4. L'App reindirizza Studente al nuovo topic creato dall'amico.
<i>Condizioni di entrata</i>	Ereditato da RiceviNotifica.
<i>Condizioni di uscita</i>	Ereditato da RiceviNotifica.
<i>Requisiti di qualità</i>	Ereditato da RiceviNotifica.

Tabella 2.48: Caso d'uso *RiceviNotificaPostAmico*

<i>Nome caso d'uso</i>	RiceviNotificaChat
<i>Attori partecipanti</i>	Ereditato da riceviNotifica
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Studente riceve un aggiornamento dall'App relativo ad un nuovo messaggio su una chat avviata da un amico. 2. L'App mostra la notifica allo studente. 3. Studente clicca sulla notifica. 4. L'App reindirizza Studente alla relativa chat con l'amico.
<i>Condizioni di entrata</i>	Ereditato da RiceviNotifica.
<i>Condizioni di uscita</i>	Ereditato da RiceviNotifica.
<i>Requisiti di qualità</i>	Ereditato da RiceviNotifica.

Tabella 2.49: Caso d'uso *RiceviNotificaChat*

<i>Nome caso d'uso</i>	RiceviNotificaRiscontroAmicizia
<i>Attori partecipanti</i>	Ereditato da riceviNotifica
<i>Flusso eventi</i>	<ol style="list-style-type: none"> 1. Studente riceve un aggiornamento dall'App relativo alla risposta di una richiesta di amicizia. 2. L'App mostra la notifica allo studente. 3. Studente clicca sulla notifica. 4. L'App reindirizza Studente al profilo dell'amico.
<i>Condizioni di entrata</i>	Ereditato da RiceviNotifica.
<i>Condizioni di uscita</i>	Ereditato da RiceviNotifica.
<i>Requisiti di qualità</i>	Ereditato da RiceviNotifica.

Tabella 2.50: Caso d'uso *RiceviNotificaChat*

2.4.3 Modello ad Oggetti

Il Modello ad Oggetti è costituito da tre tipologie di oggetti.

Gli oggetti **Entity** rappresentano le informazioni persistenti tracciate dal sistema. Gli oggetti **Boundary** rappresentano le interazioni tra gli attori e il sistema. Gli oggetti **Control** sono responsabili della realizzazione dei casi d'uso.

Alcuni oggetti Entity verranno ripetuti tra i vari package. In questa sezione saranno mostrati tutti gli oggetti relativi ai diversi package precedentemente presentati anche nella sottosezione dei casi d'uso.

I package in questione sono:

- In-Out;
- Modifica;
- Corso;
- Ricerca;
- Social;
- Topic e Commenti;
- Notifica.

Gli oggetti saranno suddivisi per package e non per ogni singolo CdU.

Package In-Out

Nell'analisi di questo package non è stato rilevato nessun caso particolare.

Entity Objects	Definizione
Studente	Persona iscritta all'Università di Nowhere, registrata all'App Uni-Way.

Tabella 2.51: Tabella oggetti Entity - Pkg In-Out

Boundary Objects	Definizione
RegistrazioneButton	Bottone da premere per dare il via alla registrazione al primo utilizzo dell'app.
FormDiBenvenuto	Form usato per inserire i propri dati dell'Università di Nowhere e per registrare i dati per l'account Uni-Way la prima volta che si accede all'App.
LoginButton	Bottone da premere per dare il via al login.
FormLogin	Form usato per inserire i dati per l'accesso al proprio account Uni-Way e per recuperare la password in caso di problemi.
LogoutButton	Bottone da premere per effettuare il logout dall'applicazione.
RecuperoPasswordButton	Bottone da premere per recuperare la password in caso di problemi.

Tabella 2.52: Tabella oggetti Boundary - Pkg In-Out

Control Objects	Definizione
RegistrazioneAppControl	Gestisce la funzione di registrazione di un utente sull'App Uni-Way. Crea un FormDiBenvenuto e lo presenta a Studente. Dopo la corretta compilazione del form, registra tali dati e reindirizza Studente sulla sua homepage.
LoginControl	Gestisce il login. Crea un FormLogin dove chiede all'utente e-mail e password per l'accesso o l'opzione di recupero password, in caso di dimenticanza. Nel primo caso controlla le credenziali: se corrette reindirizza l'utente alla sua homepage, se sbagliate lo reinvita a reinserirle. Nel secondo caso, lascia RecuperoPasswordControl.
LogoutControl	Gestisce il logout. Chiude tutte le operazioni in sospeso e reindirizza l'utente al FormLogin.

RecuperoPasswordControl	Gestisce la possibilità di recuperare la password in caso di problemi. Crea un FormRecuperaPassword, chiedendo l'e-mail con la quale l'utente si è registrato all'App, alla quale verrà inviato un messaggio con le istruzioni per il recupero della password
-------------------------	--

Tabella 2.53: Tabella oggetti Control - Pkg In-Out

Package Modifica

Data la natura di questo package, in cui troviamo operazioni di modifica di dati e preferenze dell'utente, bisogna trattare opportunamente il conseguente salvataggio delle informazioni. Per quanto riguarda i dati personali, non è necessario avere un'entity a parte, in quanto sono informazioni che definiscono la natura stessa di **Studiante**.

Entity Objects	Definizione
Studiante	Persona iscritta all'Università di Nowhere, registrata all'App Uni-Way.
Preferenze	Preferenze sulle impostazioni delle notifiche e della privacy di Studiante

Tabella 2.54: Tabella oggetti Entity - Pkg Modifica

Boundary Objects	Definizione
ModificaNotificheButton	Bottone da premere per iniziare a modificare le preferenze sulle notifiche.
FormNotifiche	Form usato per impostare le preferenze sulle notifiche.
ModificaPrivacyButton	Bottone da premere per iniziare a modificare le preferenze sulla privacy.
FormPrivacy	Form usato per impostare le le preferenze sulla privacy.
ModificaDatiButton	Bottone da premere per iniziare a modificare le informazioni personali.
FormDatiPersonali	Form usato per modificare le informazioni sui dati personali.

Tabella 2.55: Tabella oggetti Boundary - Pkg Modifica

Control Objects	Definizione
ModificaNotificheControl	Gestisce la funzione della modifica delle impostazioni di notifica di un utente. Crea un FormNotifiche e, dopo il settaggio, ne registra le modifiche.
ModificaPrivacyControl	Gestisce la funzione della modifica delle impostazioni della privacy di un utente. Crea un FormPrivacy e, dopo il settaggio, ne registra le modifiche.
ModificaDatiPersonaliControl	Gestisce la funzione della modifica delle informazioni personali di un utente. Crea un FormDatiPersonali e, dopo la corretta modifica, ne registra le informazioni.

Tabella 2.56: Tabella oggetti Control - Pkg Modifica

Package Corso

Entity Objects	Definizione
Studente	Persona iscritta all'Università di Nowhere, registrata all'App Uni-Way.
Corso	Insegnamento presente nell'Università di Nowhere.

Tabella 2.57: Tabella oggetti Entity - Pkg Corso

Boundary Objects	Definizione
FollowButton	Pulsante che avvia la procedura che consente allo studente di seguire un corso.
UnfollowButton	Pulsante che avvia la procedura che consente allo studente di smettere di seguire un corso

Tabella 2.58: Tabella oggetti Boundary - Pkg Corso

Control Objects	Definizione
FollowControl	Gestisce la funzione di iscrizione di un utente ad un corso esistente. Viene avviato in seguito alla pressione del pulsante "FollowButton", una volta elaborata la richiesta, mostra un messaggio all'utente che lo informa della corretta registrazione al Corso.
UnfollowControl	Gestisce la funzione di cancellazione di un utente ad un corso di cui segue gli aggiornamenti. Viene avviato in seguito alla pressione del pulsante "UnfollowButton", una volta elaborata la richiesta, mostra un messaggio all'utente che lo informa della corretta cancellazione dal Corso.

Tabella 2.59: Tabella oggetti Control - Pkg Corso

Package Ricerca

Entity Objects	Definizione
Studente	Persona iscritta all'Università di Nowhere, registrata all'App Uni-Way. Ogni studente è identificato univocamente da una matricola.
Corso	Corso di studi disponibile per gli studenti iscritti all'Università di Nowhere.
Commento	Risposta ad un determinato topic da parte di uno studente. Può essere una semplice risposta, un ulteriore dubbio e può essere allegato ad un file multimediale.
Post	Annuncio pubblicato da un professore o Topic creato da uno studente inerente ad un determinato corso. Può essere una semplice domanda o una richiesta di informazioni e può essere allegato con un file multimediale.
Multimedia	Un'immagine, un audio o un qualsiasi oggetto multimediale allegabile ad un topic o ad un commento..

Tabella 2.60: Tabella oggetti Entity - Pkg Ricerca

Boundary Objects	Definizione
RicercaForm	Form usata per inserire i dati per effettuare una ricerca. Contiene, oltre l'elemento da ricercare, i filtri e opzioni di ricerca.
BottoneVisualizzaStudente	Area dove vengono visualizzate le informazioni essenziali di uno Studente.
BottoneVisualizzaCorso	Area dove vengono visualizzate le informazioni essenziali di un Corso.
BottoneVisualizzaCommento	Area dove vengono visualizzate le informazioni essenziali di un Commento.
BottoneVisualizzaPost	Area dove vengono visualizzate le informazioni essenziali di Post.
BottoneVisualizzaMultimedia	Area dove vengono visualizzate le informazioni essenziali di un File Multimediale.

Tabella 2.61: Tabella oggetti Boundary - Pkg Ricerca

Control Objects	Definizione
RicercaControl	<p>Gestisce la funzione di ricerca.</p> <p>Viene attivato da un RicercaForm.</p> <p>In base ai filtri selezionati nel Form, vengono inclusi o esclusi nella ricerca varie risorse (Cor-si, Commenti, Studenti, Multimedia, Post).</p>
VisualizzaStudenteControl	<p>Attivato dall'oggetto Boundary BottoneVisualizzaStudente, Si occupa di gestire la redirectione di un utente al profilo dello Studente.</p>
VisualizzaCorsoControl	<p>Attivato dall'oggetto Boundary BottoneVisualizzaCorso, Si occupa di gestire la redirectione di un utente al Corso.</p>
VisualizzaCommentoControl	<p>Attivato dall'oggetto Boundary BottoneVisualizzaCommento, Si occupa di gestire la redirectione di un utente al Topic che contiene il Commento.</p>
VisualizzaMultimediaControl	<p>Attivato dall'oggetto Boundary BottoneVisualizzaMultimedia, Si occupa di gestire la redirectione di un utente al Post che contiene il Multimedia.</p>
VisualizzaPostControl	<p>Attivato dall'oggetto Boundary BottoneVisualizzaPost, Si occupa di gestire la redirectione di un utente al Topic o all'Annuncio.</p>

Tabella 2.62: Tabella oggetti Control - Pkg Ricerca

Package Social

Entity Objects	Definizione
Studiante	Persona iscritta all'Università di Nowhere, registrata all'App Uni-Way.
Corso	Corso di studi disponibile per gli studenti iscritti all'Università di Nowhere.
Messaggio	Oggetto che rappresenta un messaggio inviato da uno studente ad un altro suo amico.
RichiestaDiAmicizia	Oggetto che rappresenta una richiesta d'amicizia pendente/risolta inviata da uno studente all'altro.

Tabella 2.63: Tabella oggetti Entity - Pkg Social

Boundary Objects	Definizione
PulsanteRichiediAmicizia	Pulsante usato da uno studente per richiedere l'amicizia ad un'altro studente di cui non è amico.
PulsanteVisualizzaCorsiAmico	Pulsante usato da uno studente per visualizzare i corsi di un'altro studente di cui è amico.
SezioneChat	Sezione dell'App che una volta selezionata mostra la lista degli amici con cui è possibile avviare una chat.
PulsanteAvviaChat	Pulsante usato da uno studente per avviare una chat con uno degli amici che vede nella sezione chat.
BoxTestoMessaggio	InputBox usato per inserire il testo del messaggio da inviare allo studente con cui si è avviata una chat.
PulsanteInviaMessaggio	Pulsante usato da uno studente per inviare il messaggio scritto nell'apposito InputBox.
PulsanteRimuoviAmico	Pulsante usato da uno studente per rimuovere uno studente dalla sua lista degli amici.

Tabella 2.64: Tabella oggetti Boundary - Pkg Social

Control Objects	Definizione
InviaRichiestaDiAmiciziaControl	<p>Attivato dal pulsante</p> <p>RichiediAmicizia, questo oggetto Control si occupa di creare un oggetto RichiestaDiAmicizia descrivendo lo studente che invia la richiesta e quello che la riceve, recapitando a quest'ultimo l'apposita notifica.</p>
VisualizzaCorsiAmicoControl	<p>Attivato dal pulsante</p> <p>VisualizzaCorsiAmico, questo oggetto Control si occupa di mostrare tutte le informazioni relative ai corsi frequentati dall'amico a cui fa riferimento il pulsante.</p>
RimuoviAmicoControl	<p>Attivato dal pulsante</p> <p>RimuoviAmico, questo oggetto Control si occupa di rimuovere l'amico a cui fa riferimento il pulsante, dalla lista degli amici di uno studente.</p>

Tabella 2.65: Tabella oggetti Control - Pkg Social (Parte 1)

Control Objects	Definizione
ChatControl	Attivato dai pulsanti AvviaChat o InviaMessaggio, questo oggetto Control si occupa di implementare le due corrispondenti funzionalità. Se attivato dal pulsante AvviaChat, ChatControl si occupa di avviare una chat con uno degli amici presenti nell'apposita lista, recuperando tutti i messaggi che sono stati scambiati (nel caso ci siano) in precedenza. Se viene attivato invece, dal pulsante InviaMessaggio, ChatControl si occupa del salvataggio del nuovo messaggio inviato da uno studente, scritto nell'apposito Box oltre che della notifica dello stesso .

Tabella 2.66: Tabella oggetti Control - Pkg Social (Parte 2)

Package Topic-Commenti

Entity Objects	Definizione
Studente	Persona iscritta all'Università di Nowhere, registrata all'App Uni-Way. Ogni studente è identificato univocamente da una matricola.
Corso	Corso di studi disponibile per gli studenti iscritti all'Università di Nowhere.
Topic	È un particolare tipo di post inerente ad una pagina di un corso, che viene creato dagli studenti. Può essere una semplice domanda o una richiesta di informazioni, magari allegata ad un file multimediale.
Commento	Risposta ad un determinato topic da parte di uno studente. Può essere una semplice risposta, un ulteriore dubbio e può essere allegato ad un file multimediale.
Multimedia	Un'immagine, un audio o un qualsiasi oggetto multimediale allegabile ad un topic o ad un commento.
Like	Apprezzamento ad un determinato topic o commento.

Tabella 2.67: Tabella oggetti Entity - Pkg Topic-Commenti

Boundary Objects	Definizione
ButtonCreaTopic	Pulsante usato per la creazione dell'oggetto CreazioneTopicControl che presenterà all'utente FormCreaTopic.
FormCreaTopic	Form usata per inserire domande, dubbi, informazioni o materiale all'interno di un determinato corso.
ButtonCancellaTopic	Pulsante usato per cancellare un determinato topic. Tale azione eliminerà anche le eventuali risposte (commenti) legati al topic.
FormInserisciCommento	Form usata per inserire una risposta ad un determinato topic, può essere una semplice risposta o una condivisione multimediale.
ButtonRimuoviCommento	Pulsante usato rimuovere un commento sotto un determinato topic.
ButtonAggiungiLike	Pulsante usato per aggiungere like ad un determinato topic/commento.

Tabella 2.68: Tabella oggetti Boundary - Pkg Topic-Commenti

Control Objects	Definizione
CreazioneTopicControl	Gestisce la creazione di un topic. L'oggetto è creato quando Studente clicca ButtonCreaTopic e a sua volta viene creato FormCreaTopic da far riempire a Studente.
CancellazioneTopicControl	Gestisce la cancellazione di un topic dalla pagina di un corso. L'oggetto crea ButtonCancellaTopic da mostrare a Studente.
InserimentoCommentoControl	Gestisce l'inserimento di un commento sotto un determinato topic. L'oggetto crea FormInserisciCommento da far compilare a Studente.
RimozioneCommentoControl	Gestisce la rimozione di un commento inserito precedentemente dallo stesso Studente. L'oggetto crea ButtonRimozioneCommento da mostrare a Studente.
AggiuntaLikeControl	L'oggetto provvede alla gestione dei like ogni qualvolta uno studente interagisce con ButtonAggiungiLike.

Tabella 2.69: Tabella oggetti Control - Pkg Topic-Commenti

Package Notifica

Entity Objects	Definizione
Studente	Persona iscritta all'Università di Nowhere, registrata all'App Uni-Way.
Topic	È un post inerente ad una pagina di un corso, che viene creato dagli studenti. Può essere una semplice domanda o una richiesta di informazioni, magari allegata ad un file multimediale.
Annuncio	È una particolare tipologia di post, che contiene informazioni ufficiali relativi ad un corso.
Messaggio	Oggetto che rappresenta un messaggio inviato da uno studente ad un altro suo amico.

Tabella 2.70: Tabella delle Entità

Boundary Objects	Definizione
NotificaPush	Permette di visualizzare le informazioni della notifica in modo dettagliato, indirizzando lo studente alla relativa sezione interessata.

Tabella 2.71: Tabella oggetti Boundary - Pkg Notifica

Control Objects	Definizione
NotificaRichiestaDiAmiciziaControl	<p>Oggetto Control che gestisce la ricezione di una notifica riguardo una nuova richiesta di amicizia e le informazioni ad essa relative.</p> <p>Attraverso l'oggetto boundary NotificaPush, l'oggetto Control si occupa, inoltre, di far visualizzare in modo dettagliato la richiesta di amicizia.</p>
NotificaAnnuncioControl	<p>Oggetto Control che gestisce la ricezione di una notifica su un nuovo Topic di tipo Annuncio e le informazioni ad essa relative.</p> <p>Attraverso l'oggetto boundary NotificaPush, l'oggetto Control si occupa, inoltre, di far visualizzare in modo dettagliato l'Annuncio.</p>
NotificaPostAmicoControl	<p>Oggetto Control che gestisce la ricezione di una notifica su un post scritto da un Amico e le informazioni ad esso relative.</p> <p>Attraverso l'oggetto boundary NotificaPush, l'oggetto Control si occupa, inoltre, di far visualizzare in modo dettagliato il post.</p>

Tabella 2.72: Tabella oggetti Control - Pkg Notifica

Control Objects	Definizione
NotificaChatControl	Oggetto Control che gestisce la ricezione di una notifica su un messaggio di una Chat e le informazioni ad essa relative. Attraverso l'oggetto boundary NotificaPush , l'oggetto Control si occupa, inoltre, di far visualizzare in modo dettagliato il messaggio ricevuto.
NotificaRiscontroAmiciziaControl	Oggetto Control che gestisce la ricezione di una notifica sul riscontro di una richiesta di amicizia e le informazioni ad esso relative. Attraverso l'oggetto boundary NotificaPush , l'oggetto Control si occupa, inoltre, di far visualizzare in modo dettagliato il post.

Tabella 2.73: Tabella oggetti Control - Pkg Notifica

Modellazione ad oggetti. Il diagramma delle classi è usato per descrivere la struttura del sistema. Le classi sono astrazioni che specificano la struttura comune ed il comportamento di una serie di oggetti. Gli oggetti sono istanze di classi che sono create, modificate e distrutte durante l'esecuzione del sistema. Un oggetto ha uno stato che include i valori dei suoi attributi e i suoi collegamenti con altri oggetti. Il diagramma delle classi descrive il sistema in termini di oggetti, classi, attributi, operazioni e associazioni.

Diagramma delle classi

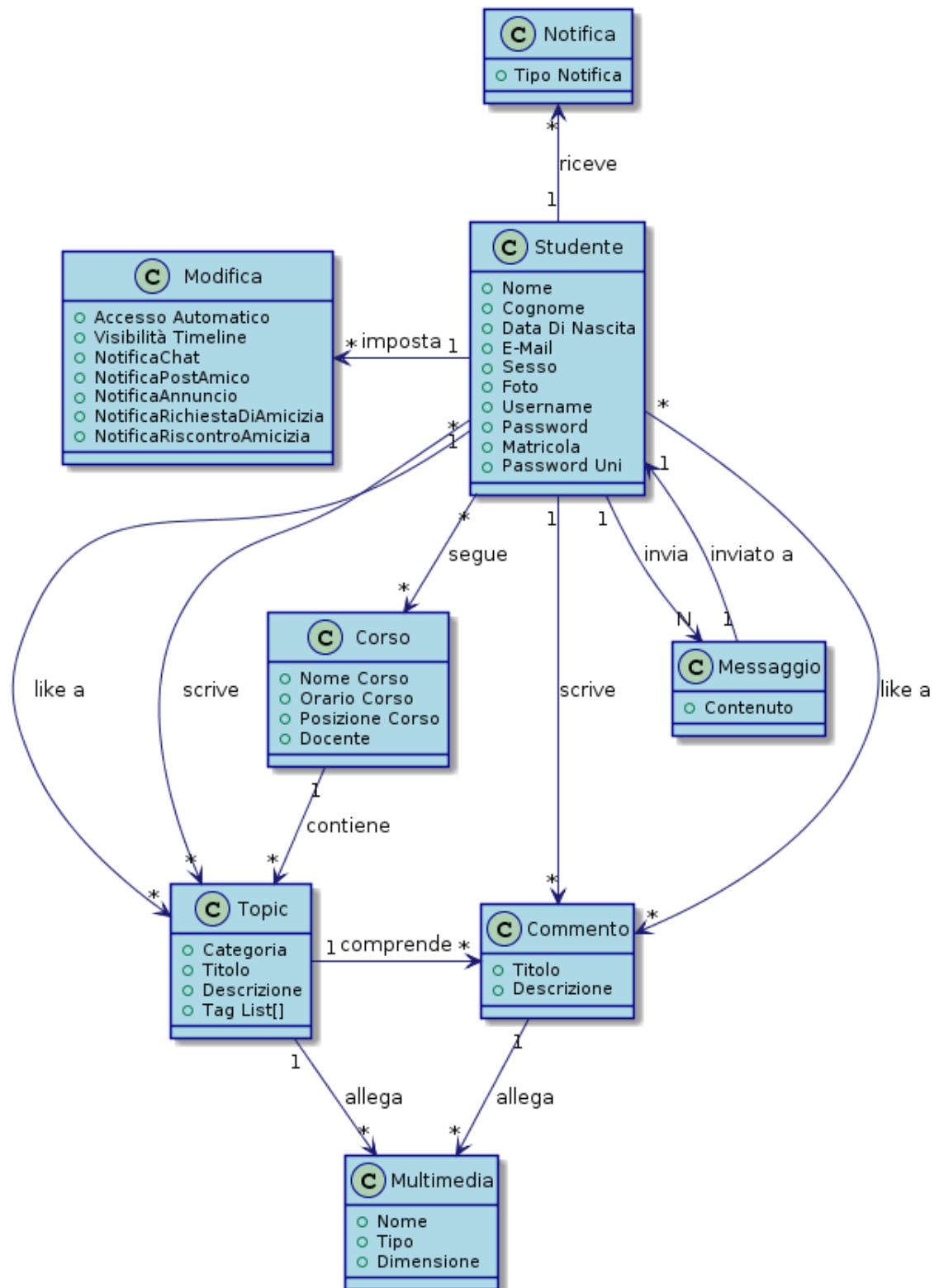


Figura 2.9: Diagramma delle classi.

2.4.4 Modello Dinamico

Sono stati identificati finora un certo numero di oggetti entity, boundary e control che partecipano ai diversi casi d'uso. Lungo la progettazione di UniWay sono stati inoltre identificati diversi attributi e associazioni. Tali oggetti verranno rappresentati in diversi diagrammi delle sequenze, raffiguranti le interazioni che si verificano durante i casi d'uso mostrati precedentemente. Nel dettaglio un diagramma delle sequenze lega i casi d'uso agli oggetti, mostrando come il comportamento di un caso d'uso (o scenario) è distribuito tra gli oggetti partecipanti.

I diagrammi di sequenza di solito non sono un buon supporto per la comunicazione con l'utente quanto i casi d'uso poiché i diagrammi delle sequenze richiedono più informazioni sulla notazione. Tuttavia per i clienti più esperti risultano più intuitivi più precisi dei casi d'uso. In ogni caso rappresentano un diverso cambio di prospettiva e consentono di trovare oggetti mancanti o aree grigie nel specifica dei requisiti.

Di seguito vengono riportati i diagrammi delle sequenze dei casi d'uso che si ritengono più complicati.

- In-Out:
 - *RegistrazioneAppControl*;
- Modifica:
 - *ModificaPrivacy*:
- Ricerca:
 - *CercaRisorse*;
- Topic e Commenti
 - *MostraApprezzamento*;
 - *CreaTopic*.

- Social:
 - *AvviaChat*:
- Notifica
 - *RiceviNotificaChat*;

Si nota che il Diagramma delle sequenze dei casi d'uso CercaRisorse e Avvia-Chat sono stati divisi in 2 parti per problemi di spazio.

Package In-Out

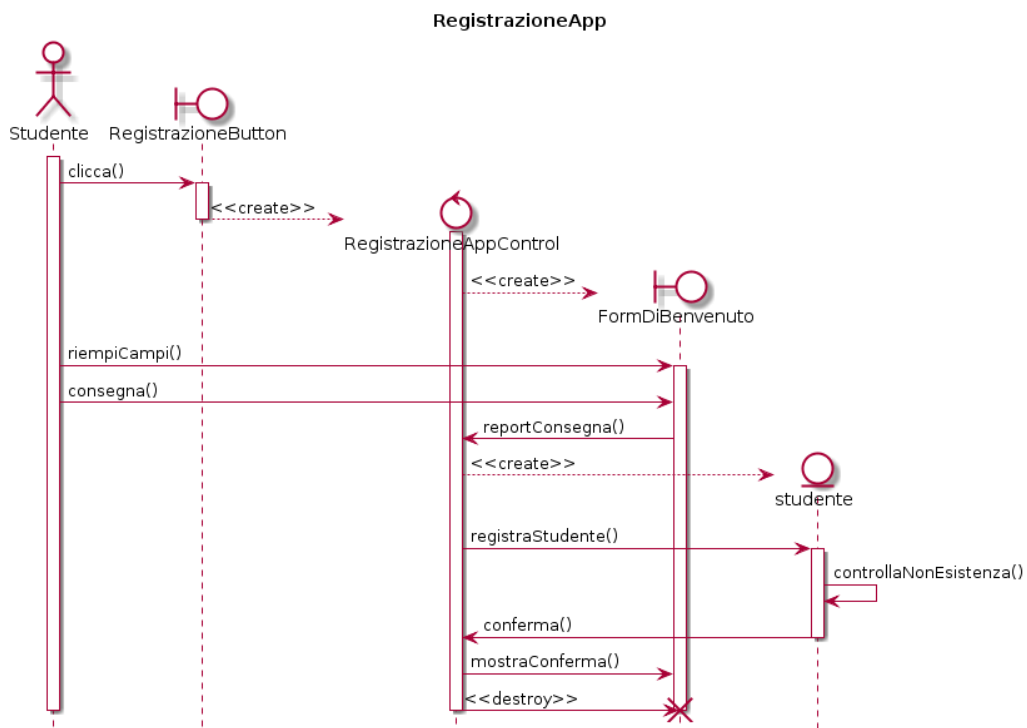


Figura 2.10: Diagramma delle sequenze - Registrazione

Package Modifica

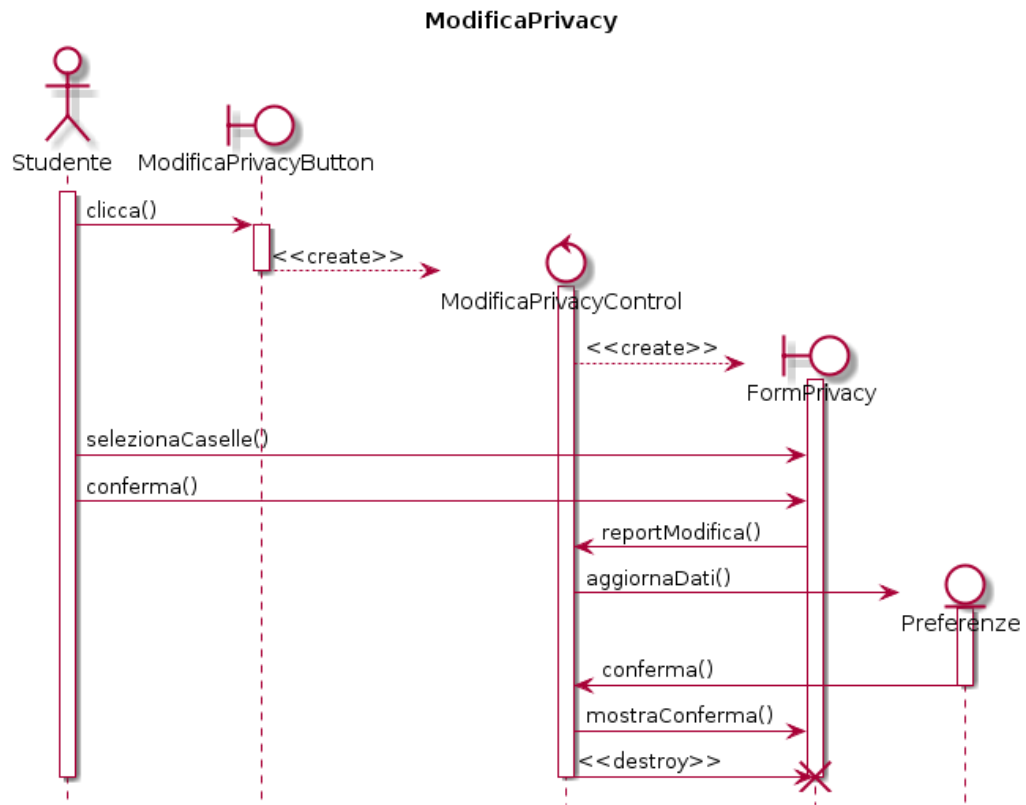


Figura 2.11: Diagramma delle sequenze - Modifica Privacy

Package Notifica

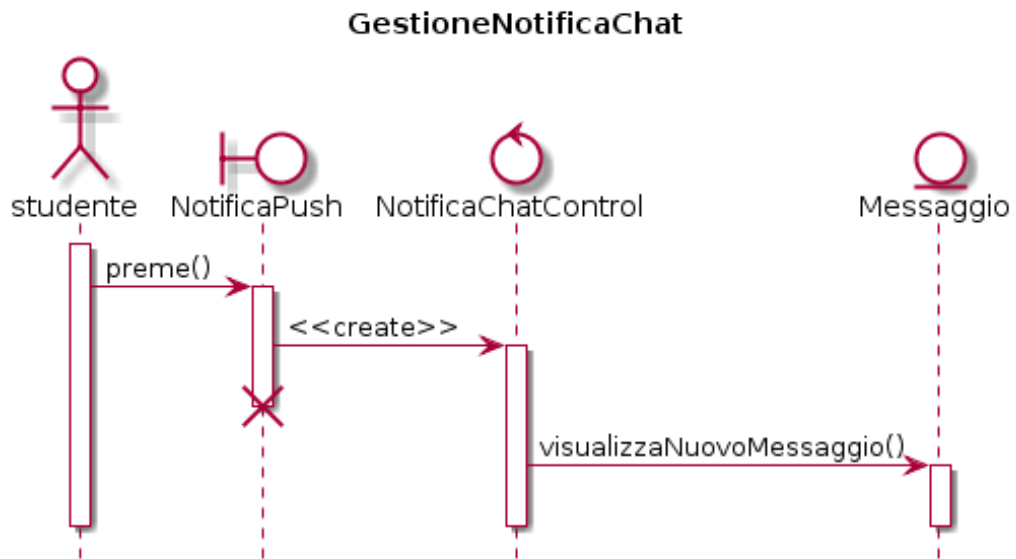


Figura 2.12: Diagramma delle sequenze - gestione notifica chat

Package Ricerca

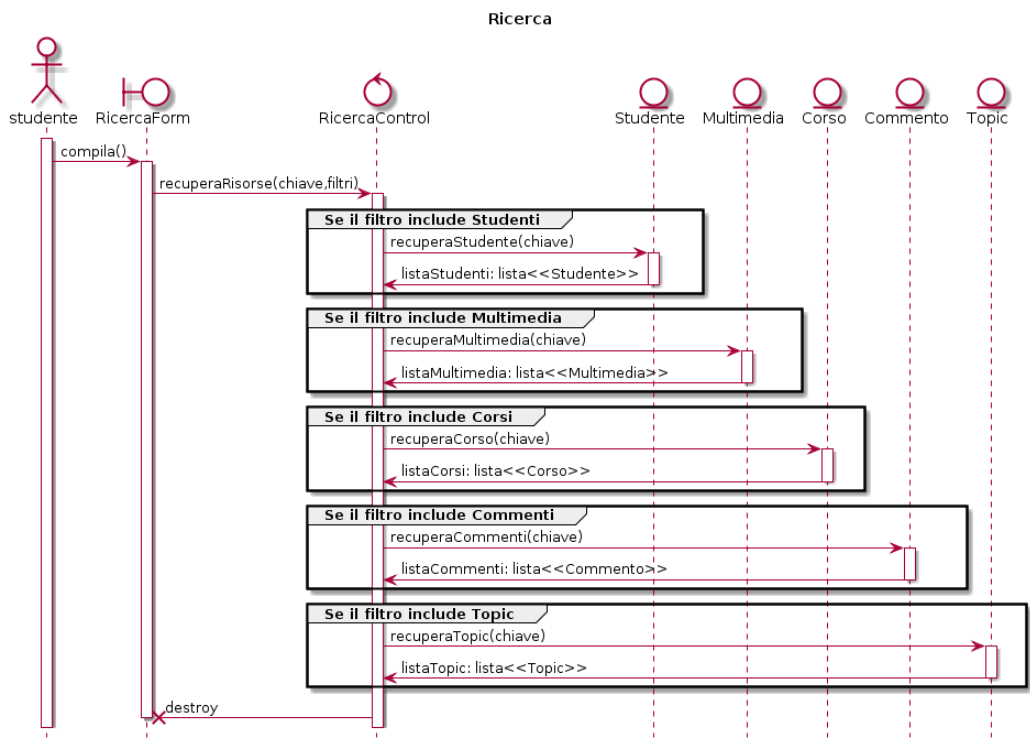


Figura 2.13: Diagramma delle sequenze - ricerca parte 1

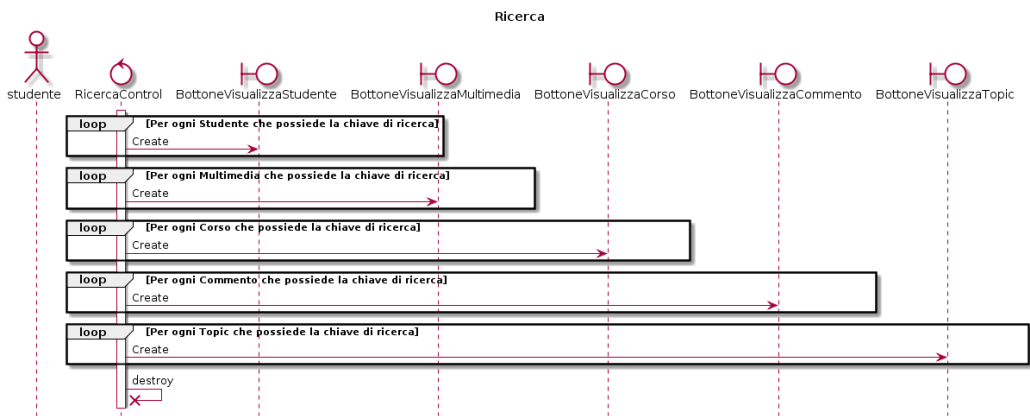


Figura 2.14: Diagramma delle sequenze - ricerca parte 2

Package Social

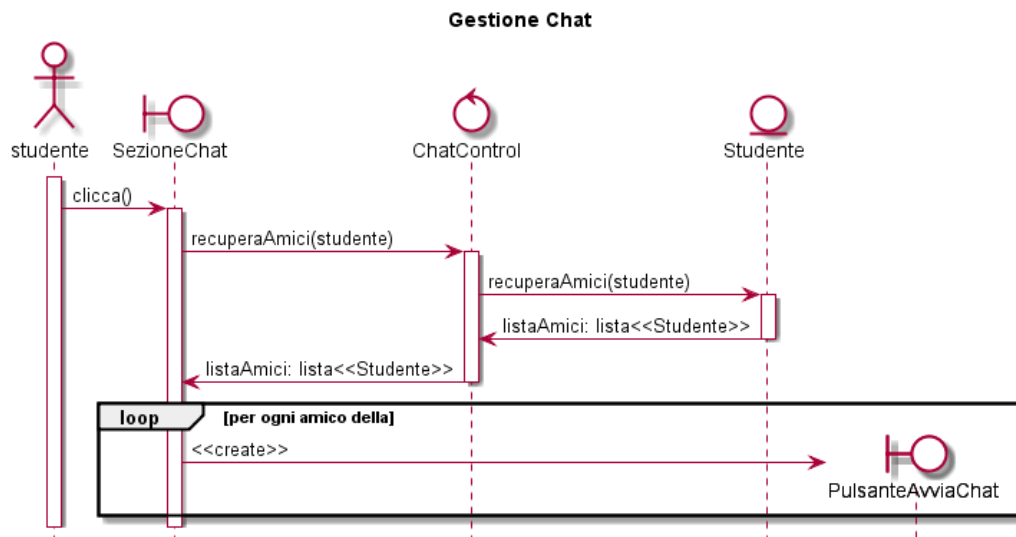


Figura 2.15: Diagramma delle sequenze - gestione chat parte 1

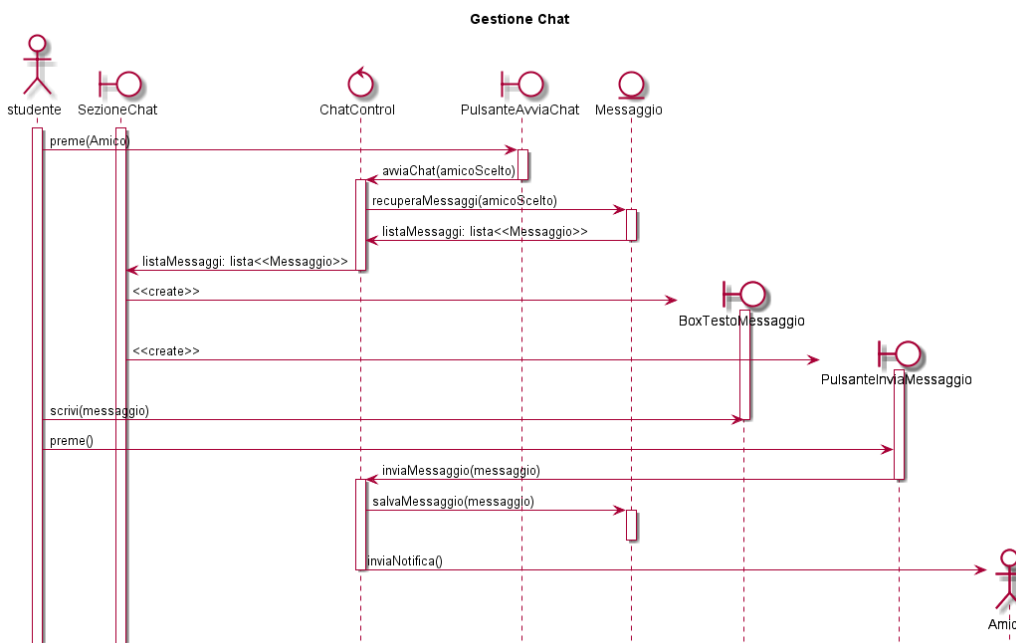


Figura 2.16: Diagramma delle sequenze - gestione chat parte 2

Package Topic-Commenti

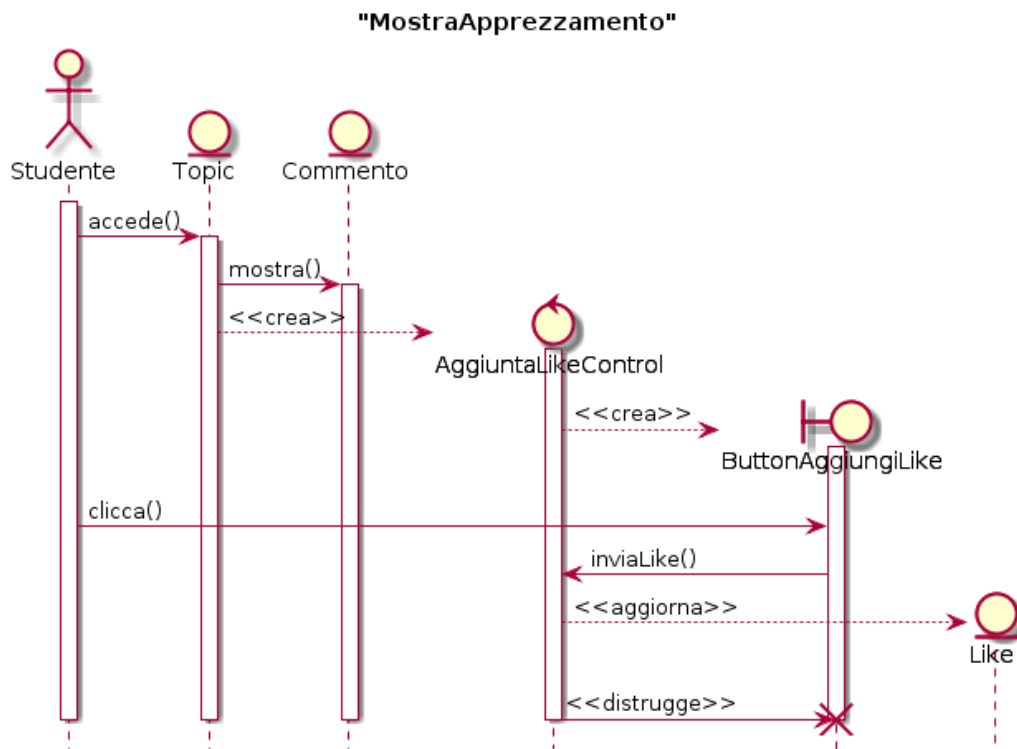


Figura 2.17: Diagramma delle sequenze - Mostra Apprezzamento

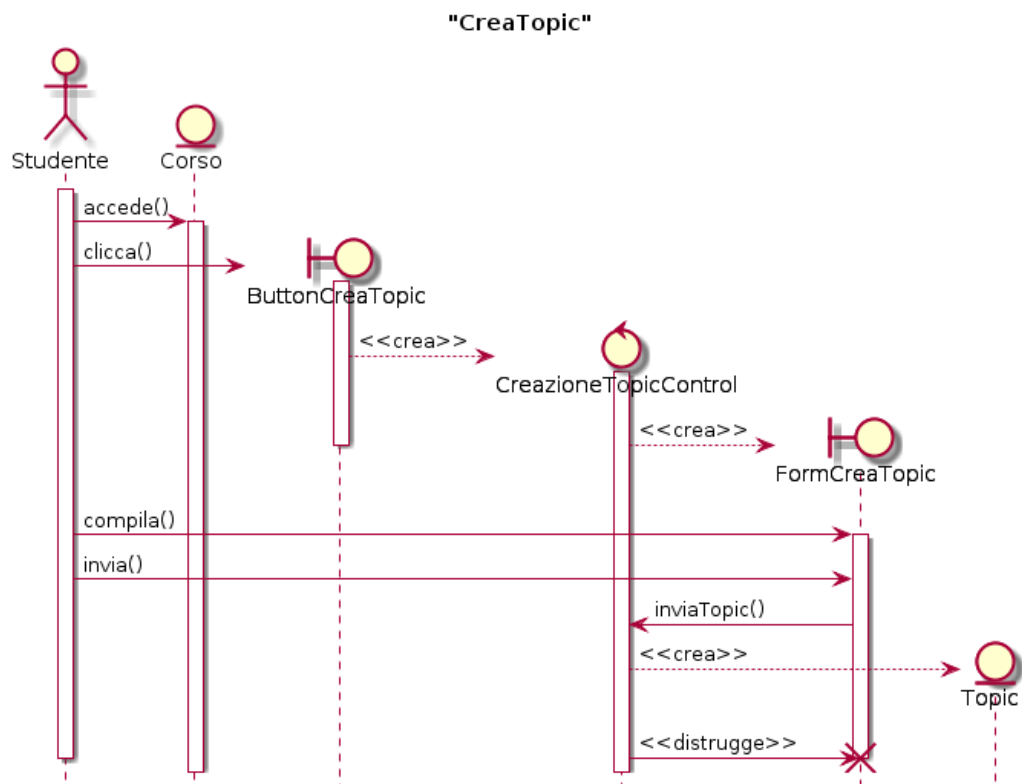


Figura 2.18: Diagramma delle sequenze - Scrivi Commento

2.4.5 User Interface Mockups

Parte III

Documento di Design del Sistema

Capitolo 3

Sistema Proposto

3.1 Panoramica

La progettazione del sistema è la trasformazione di un modello di analisi in un modello di progettazione del sistema. Durante la progettazione del sistema gli sviluppatori definiscono gli obiettivi di progettazione del progetto e scompongono il sistema in sottosistemi più piccoli. Gli sviluppatori selezionano anche strategie per costruire il sistema, come la strategia hardware/software, la gestione dei dati persistenti, il flusso di controllo globale, la politica di controllo degli accessi e la gestione delle condizioni limite. Il risultato della progettazione del sistema è un modello che include una decomposizione del sottosistema e una chiara descrizione di ciascuna di queste strategie.

3.2 Decomposizione in Sottosistemi

Per ridurre la complessità del dominio applicativo (come mostrato nel documento RAD) sono state identificate piccole parti chiamate "classi", per poi essere organizzate in package. In modo simile, per ridurre la complessità del dominio della soluzione, il sistema viene decomposto in parti più semplici, chiamate "sottosistemi", costituiti da un numero di classi di domini di soluzioni. Un sottosistema è a parte sostituibile del sistema con interfacce ben definite che incapsulano lo stato e comportamento delle sue classi contenute. Un sottosistema corrisponde in genere alla quantità di lavoro che un singolo sviluppatore o un singolo team di sviluppo possono affrontare. Decomponendo il sistema in sottosistemi relativamente indipendenti, i team simultanei possono lavorare su singoli sottosistemi con sovraccarico minimo di comunicazione.

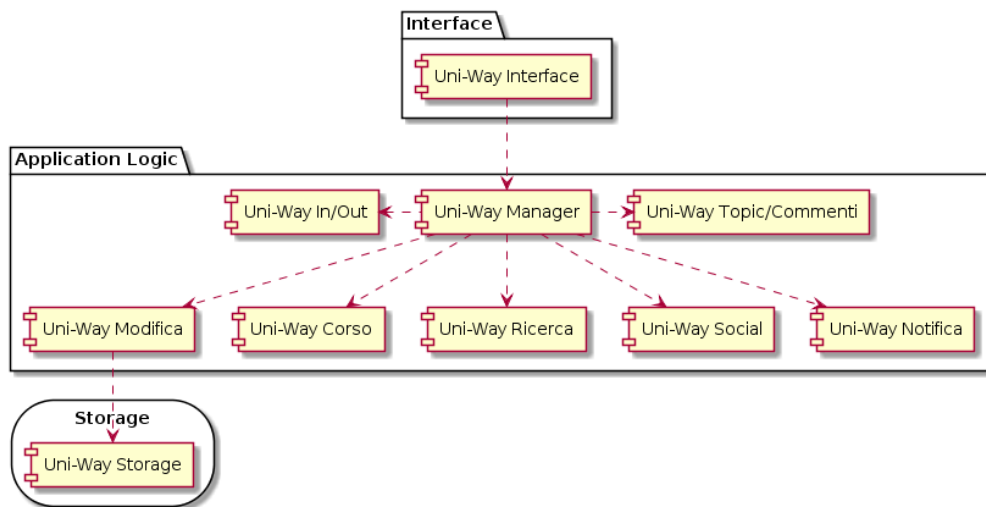


Figura 3.1: Decomposizione in sottosistemi generale (UML Component Diagram, livelli mostrati come UML package)

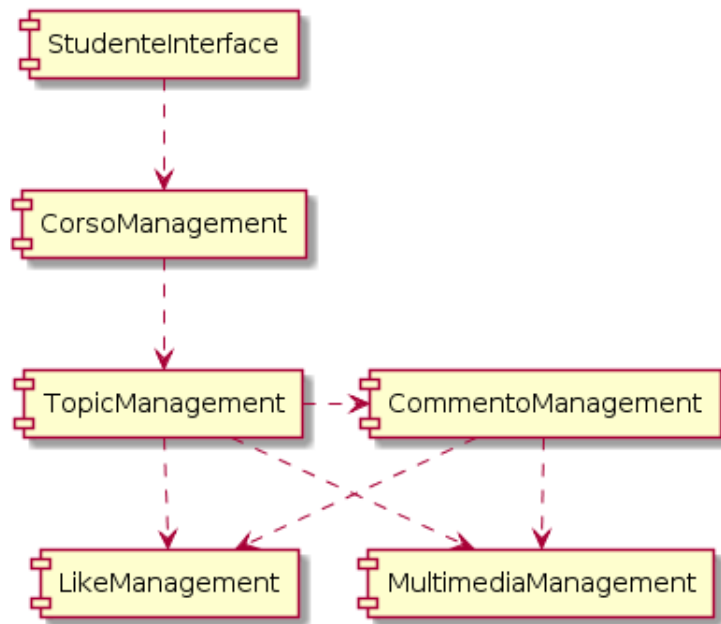


Figura 3.2: Decomposizione in sottosistemi, parte topic-commenti (UML Component Diagram)

3.4 Gestione dei Dati Persistenti

Uni-Way si occupa di poche informazioni che vengono salvate sul dispositivo dell'utente. Esse riguardano essenzialmente il package **Modifica**, in particolare vengono salvate informazioni riguardanti le notifiche, le modifiche ai dati utente (come l'accesso automatico) e la privacy. Vista l'esigua quantità e la bassa densità di informazioni viene individuato come perfetto candidato un database di tipo **flat-file** (conosciuto anche come *text database*).

3.5 Controllo Accessi e Sicurezza

Fare matrice con una colonna per l'unico attore presente (Studente) e più righe per le diverse classi a cui accedere. Es: annuncio possibile solo accedervi.

Fare anche tabella accessi globali.

3.6 Flusso di Controllo Globale

Progettazione decentralizzata (Controllo risiede in più oggetti indipendenti, dds come una scala).

3.7 Condizioni Limite

Inizializzazione: il sistema è portato da uno stato non inizializzato ad uno stato stabile;

Terminazione: le risorse sono ripulite e sono notificati altri sistemi al momento della terminazione;

Guasto: Possibili guasti: bug, errori, problemi esterni.

Parte IV

Glossario

Capitolo 4

Divisione dei Glossari

4.1 Glossario del RAD

Termine	Descrizione
Annuncio	Particolare tipo di topic inerente a eventi ufficializzati dall'Università.
Chat	Sezione in cui è possibile scambiare messaggi privatamente con un amico iscritto all'App.
Commento	Risposta di uno studente ad un topic, o ad un altro commento.
Corso	Insegnamento presente nell'Università di Nowhere che gli studenti possono frequentare.
Menu	Sezioni specifiche dell'App contenenti diverse opzioni funzionali a determinate specifiche.
Notifica	Avviso riguardante un evento specifico.

Tabella 4.1: **Glossario RAD parte I**

Termine	Descrizione
Profilo	Sezione dell'App relativa ad uno studente in cui vengono mostrate le sue informazioni personali.
Risorsa	Particolare commento o topic contenente materiale utile per un corso (che sia un link, un'informazione o un file multimediale).
Studente	Persona iscritta all'App (frequentante l'Università di Nowhere).
Timeline	Sezione in cui vengono mostrate le attività relative allo studente selezionato.
Topic	Post scritto da uno studente su una pagina di un corso.

Tabella 4.2: **Glossario RAD parte II**

4.2 Glossario del SDD

Bibliografia

- [1] B. Bruegge, A. H. Dutoit. *[Object-Oriented Software Engineering]*