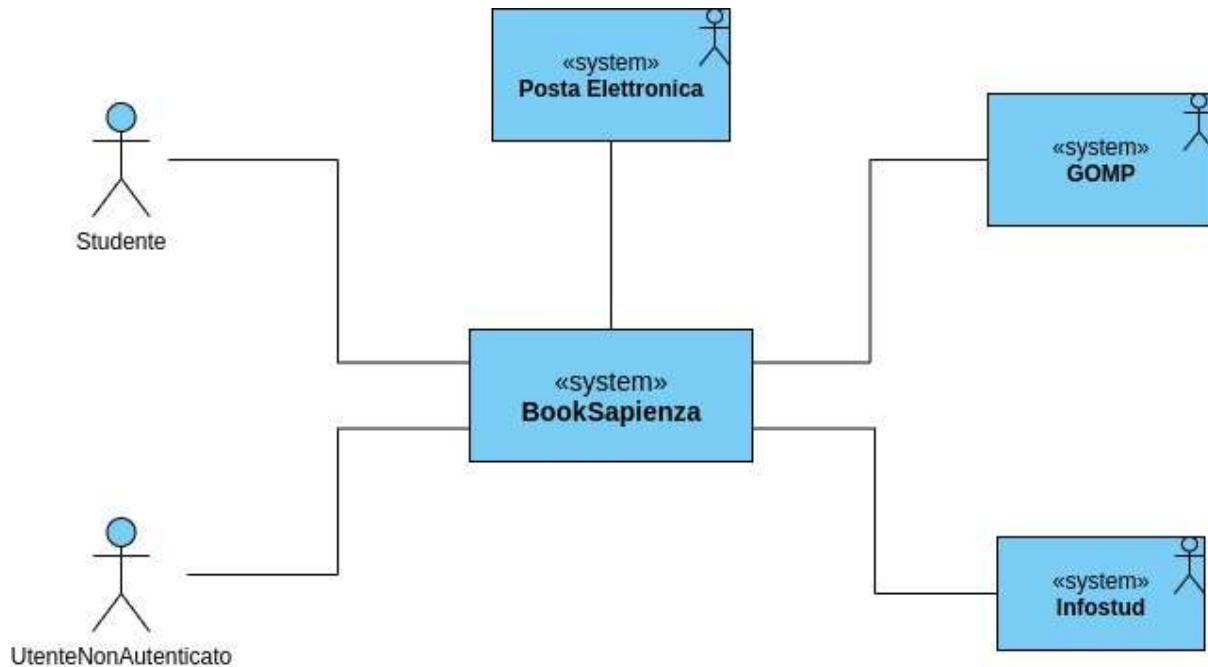


Analisi di contesto



BookSapienza è un sistema di prenotazione per le lezioni in presenza dell'università La Sapienza di Roma. Esso interagisce con altri sistemi quali:

- **Infostud**, dal quale le informazioni personali dello studente (nome, cognome, matricola, password, email), e il suo percorso formativo (una serie di corsi);
- Il **GOMP** dal quale riceve le informazioni relative alle aule (il nome, la capienza, l'indirizzo dell'edificio, il codice dell'edificio) e alle lezioni (il corso, l'aula dove si svolge, il professore che la tiene, l'orario e i giorni della settimana) che lo studente avrà nella settimana successiva. Per reperire tali informazioni il GOMP sfrutta il percorso formativo fornito da Infostud;
- Il **server di posta elettronica**, che viene usato per comunicare l'avvenuta prenotazione ad una lezione tramite la ricevuta via mail.

Inoltre, l'utente che interagisce con il sistema può essere:

- Un **utente non autenticato**, cioè un utente che non ha effettuato l'accesso al sistema, e quindi può solo fare il login;
- Uno **studente**, cioè un utente che ha correttamente effettuato l'accesso al sistema mediante le sue credenziali di Infostud, e quindi può usufruire delle funzionalità del sistema. Gli studenti dell'università “La Sapienza” di Roma sono gli unici autorizzati ad effettuare l'accesso in quanto sono i soli che necessitano di prenotare un posto in aula per le lezioni.

Specifica dei requisiti

Ogni tabella che dettaglia un requisito è composta da:

- **ID:** l'identificativo univoco del requisito;
- **Priorità:** indica il grado di importanza del requisito. Abbiamo scelto di adottare il metodo MoSCoW, che divide in 4 categorie l'interesse degli stakeholders verso un determinato requisito. Tali categorie sono:
 - Must: il requisito è essenziale affinché il sistema sia un successo. Se anche un solo requisito Must have non è incluso, il progetto dovrebbe essere considerato non ultimato;
 - Should: il requisito è importante ma non critico come un requisito Must, senza un requisito Should il sistema ne risente a livello funzionale o non-funzionale;
 - Could: il requisito è apprezzabile, senza tale requisito il sistema ne risente a livello qualitativo e di esperienza dell'utente;
 - Won't: il requisito è stato considerato non appropriato in questo momento dagli stakeholders, ma potrà essere considerato in aggiornamenti futuri.
- **Descrizione:** descrizione del requisito in linguaggio naturale, più o meno approfondita a seconda che si tratti di System o User requirement.

User requirements

Requisiti funzionali:

ID:	F1
Priorità:	Must
Descrizione:	Il sistema deve permettere ad un utente non autenticato di poter accedere al sistema BookSapienza usando le credenziali Infostud.

ID:	F2
Priorità:	Must
Descrizione:	Il sistema deve permettere ad uno studente di poter visualizzare tutte le lezioni prenotabili in base al suo percorso formativo, se la sua matricola rientra nella turnazione prevista.

ID:	F3
Priorità:	Must
Descrizione:	Il sistema deve permettere ad uno studente di poter prenotare un posto in aula per una delle sue lezioni tenendo conto della turnazione settimanale, basata sulle ultime due cifre del numero di matricola (da 00 a 49 / da 50 a 99). La prenotazione per ciascun turno può essere effettuata solo dal lunedì al venerdì della settimana precedente. A prenotazione effettuata, lo studente verrà notificato tramite una mail.

ID:	F4
Priorità:	Should
Descrizione:	Il sistema deve permettere ad uno studente di poter cancellare la prenotazione ad una lezione. La cancellazione per ciascun turno può essere effettuata solo dal lunedì al venerdì della settimana precedente.

ID:	F5
Priorità:	Should
Descrizione:	Il sistema deve permettere ad uno studente di poter aggiungersi/rimuoversi da una coda di attesa, se i posti ad una lezione sono tutti occupati.

ID:	F6
Priorità:	Must
Descrizione:	Il sistema deve permettere a tutti gli studenti, anche coloro che non rientrano nella turnazione, di poter visualizzare le prenotazioni effettuate o il proprio posto nella coda d'attesa.

ID:	F7
Priorità:	Should
Descrizione:	Il sistema deve permettere a tutti gli studenti di poter visualizzare e scaricare la ricevuta di una prenotazione.

ID:	F8
Priorità:	Must
Descrizione:	Il sistema deve permettere ad uno studente, dopo aver terminato l'uso del sistema, di poter effettuare il logout.

Requisiti non funzionali:

ID:	NF1
Priorità:	Must
Descrizione:	La comunicazione in rete con il sistema deve essere sicura;

ID:	NF2
Priorità:	Should
Descrizione:	Il sistema deve essere intuitivo e facile da usare per gli utenti;

ID:	NF3
Priorità:	Must
Descrizione:	Il sistema deve garantire la protezione dei dati degli utenti e delle loro prenotazioni;

ID:	NF4
Priorità:	Should
Descrizione:	Il sistema deve essere sempre online per il maggior tempo possibile;

ID:	NF5
Priorità:	Should
Descrizione:	Il tempo di risposta alle azioni dell'utente deve essere breve;

ID:	NF6
Priorità:	Must
Descrizione:	Il sistema deve essere accessibile dai principali dispositivi e browser presenti sul mercato;

ID:	NF7
Priorità:	Must
Descrizione:	Il sistema deve resistere ad eventuali perdite di dati o fallimenti.

System requirements

Requisiti funzionali:

ID:	F1
Priorità:	Must
Descrizione:	Il sistema deve permettere ad un utente non autenticato di effettuare l'accesso al sistema di prenotazione BookSapienza usando la sua matricola e la sua password di Infostud, quest'ultimo si occuperà di verificare la validità dei dati inseriti e consentirà o meno l'accesso.

ID:	F2
Priorità:	Must
Descrizione:	Il sistema presenta allo studente, se la sua matricola rientra nella turnazione prevista, tutte e sole le lezioni prenotabili in base al suo percorso formativo, che reperisce da Infostud. Le lezioni contengono il nome del corso, la data, l'orario, il nome del docente e l'aula.

ID:	F3
Priorità:	Must
Descrizione:	Il sistema deve permettere ad uno studente di prenotare un posto per una lezione, la quale si tiene in una specifica aula che ha una determinata capienza. Se l'aula non è piena la prenotazione deve sempre essere accettata. La prenotazione avviene seguendo una turnazione settimanale, basata sulle ultime due cifre del numero di matricola (da 00 a 49 / da 50 a 99). La prenotazione per ciascun turno può essere effettuata solo dalle 00.00 del lunedì alle 23:59

	del venerdì della settimana precedente le lezioni dedicate a quel turno. Verrà inviata una mail allo studente come conferma della prenotazione, contenente la ricevuta.
--	---

ID:	F4
Priorità:	Should
Descrizione:	Il sistema deve permettere ad uno studente che rientra nella turnazione di cancellare la prenotazione ad una lezione. La cancellazione per ciascun turno può essere effettuata solo dalle 00.00 del lunedì alle 23:59 del venerdì della settimana precedente.

ID:	F5
Priorità:	Should
Descrizione:	Se una lezione ha tutti i posti occupati il sistema deve fare in modo che lo studente che rientra nella turnazione non si possa prenotare, ma solo inserire in una coda d'attesa. Quando uno studente che era già prenotato a quella lezione si cancella, libera un posto in aula e fa prenotare automaticamente la prima persona in coda d'attesa per quella lezione. Uno studente è sempre libero di cancellarsi da una coda d'attesa in cui si è inserito, facendo così avanzare di una posizione tutti gli utenti in coda dopo di lui. Le code d'attesa sono valide dalle 00.00 del lunedì alle 23:59 del venerdì della settimana precedente le lezioni. Gli utenti rimasti in coda fino a tale data non saranno prenotati.

ID:	F6
Priorità:	Must
Descrizione:	Il sistema deve mostrare l'elenco di tutte le prenotazioni effettuate dallo studente, filtrabili per settimana. Ogni prenotazione contiene la data, l'orario, il nome del corso e l'edificio in cui si tiene la lezione. Se lo studente per una lezione non si è prenotato ma è in coda d'attesa allora visualizzerà le stesse informazioni di una prenotazione più un'informazione aggiuntiva che specifica che si trova in una coda d'attesa con la relativa posizione. Anche gli studenti che non rientrano nella turnazione settimanale possono vedere lo storico delle loro prenotazioni.

ID:	F7
Priorità:	Should
Descrizione:	Il sistema deve permettere a tutti gli studenti di visualizzare e scaricare la ricevuta di una prenotazione effettuata, nella quale sono contenute le informazioni circa la data, l'orario, l'indirizzo, il nome dell'edificio in cui si svolge la lezione, il docente che la tiene e le informazioni personali dello studente quali nome, cognome e matricola. La ricevuta contiene inoltre un QR Code univoco che permette l'accesso all'aula.

ID:	F8
Priorità:	Must
Descrizione:	Il sistema deve permettere allo studente di effettuare il logout per terminare la sessione ed uscire dal sistema BookSapienza. In seguito al logout, lo studente viene rimandato alla pagina di login.

Requisiti non funzionali:

ID:	NF1
Priorità:	Must
Descrizione:	Tutte le comunicazioni in rete del sistema con gli altri attori devono garantire autenticità e confidenzialità, ad esempio attraverso l'utilizzo del protocollo HTTPS.

ID:	NF2
Priorità:	Should
Descrizione:	Gli utenti devono essere in grado di apprendere le funzionalità del sistema dopo 30 minuti di utilizzo. In seguito alla fase di apprendimento, il numero medio di errori commessi in un'ora da un utente deve essere minore o uguale ad 1.

ID:	NF3
Priorità:	Must
Descrizione:	Il sistema deve gestire in modo sicuro i dati sensibili degli utenti e delle prenotazioni che vengono mantenuti in memoria, in modo da evitare accessi non permessi. In particolare, i dati a dover essere protetti sono quelli presi da Infostud (il nome, il cognome, la

	matricola dello studente, e la password con cui accede al sistema) e le informazioni sulle prenotazioni effettuate.
--	---

ID:	NF4
Priorità:	Should
Descrizione:	Il sistema BookSapienza non deve essere messo completamente offline da eventuali manutenzioni. Per tutto il resto del tempo BookSapienza dovrebbe essere raggiungibile in condizioni normali.

ID:	NF5
Priorità:	Should
Descrizione:	Il tempo impiegato da un utente per: <ul style="list-style-type: none"> - effettuare una prenotazione (o aggiungersi ad una coda) non deve essere superiore ad un minuto; - visualizzare una prenotazione (o la propria posizione in una coda) non deve essere superiore a 30 secondi; - cancellare una prenotazione (o rimuoversi da una coda) non deve essere superiore ad un minuto; - visualizzare e scaricare una prenotazione non deve essere superiore a 30 secondi.

ID:	NF6
Priorità:	Must
Descrizione:	Il sistema deve supportare l'accesso dai principali dispositivi (Android, iOS, Linux, Windows, MacOS) e browser presenti sul mercato (Chrome, Firefox, Edge, Safari).

ID:	NF7
Priorità:	Must
Descrizione:	Il sistema deve poter ripartire in seguito ad eventuali perdite o manomissioni di dati o funzionalità del sistema, implementando ridondanza dei dati (copie dei dati e del sistema).

Attori

ID:	A1
Nome:	UtenteNonAutenticato
Tipo:	Primario
Stereotipo:	Umano
Descrizione:	Utente che non ha effettuato l'accesso a BookSapienza, e quindi può solo fare il login.

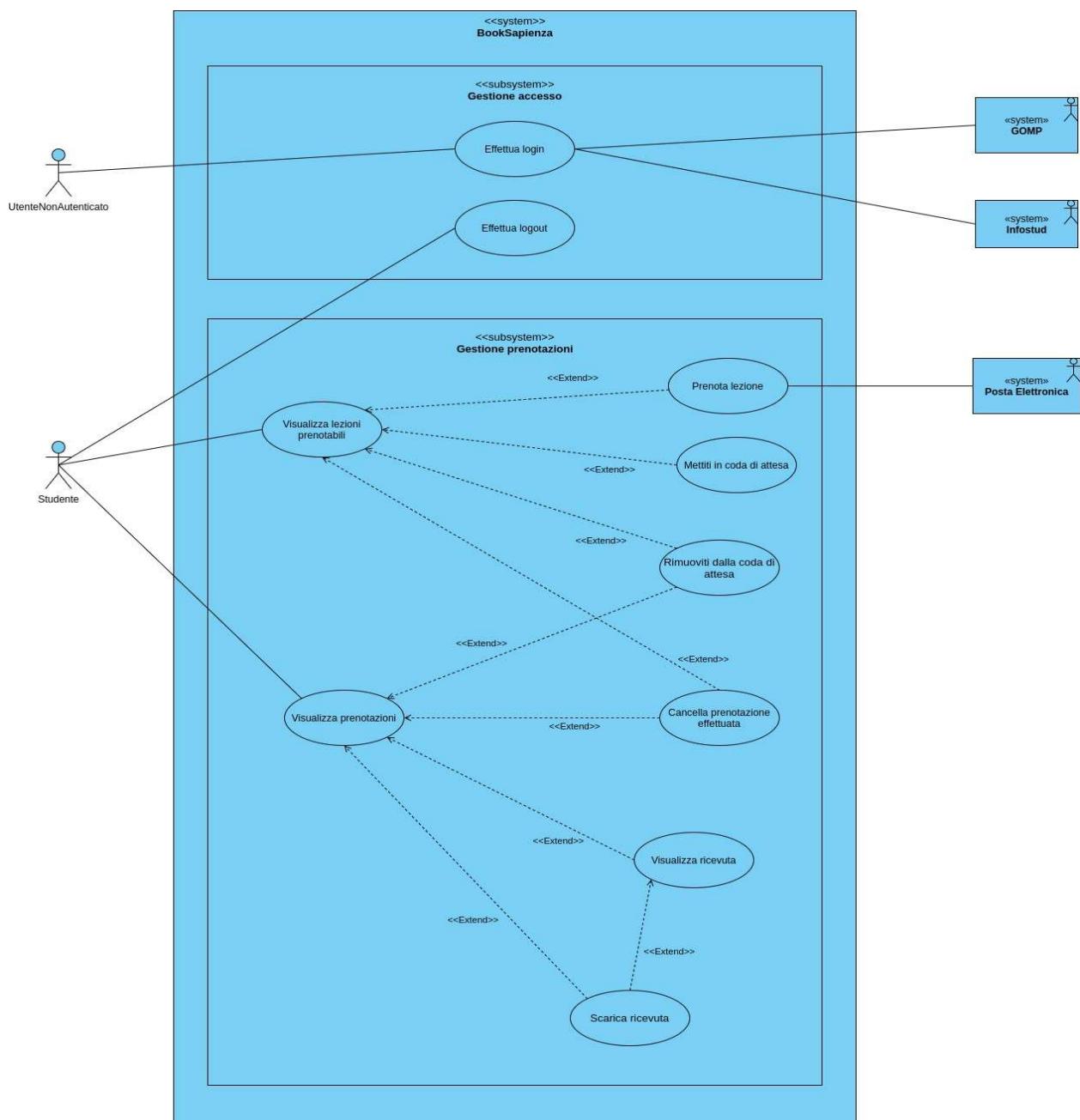
ID:	A2
Nome:	Studente
Tipo:	Primario
Stereotipo:	Umano
Descrizione:	Utente che ha correttamente effettuato l'accesso a BookSapienza mediante le sue credenziali di Infostud, e quindi può usufruire delle funzionalità del sistema.

ID:	A3
Nome:	Infostud
Tipo:	Di supporto
Stereotipo:	Sistema
Descrizione:	Rappresenta l'applicativo dal quale vengono prese le informazioni personali dello studente (nome, cognome, matricola, password, email), e il suo percorso formativo (una serie di corsi).

ID:	A4
Nome:	GOMP
Tipo:	Di supporto
Stereotipo:	Sistema
Descrizione:	Rappresenta l'applicativo che comunicherà al sistema tutte le informazioni necessarie sulle lezioni dei corsi universitari e sulle aule in cui essi si tengono, compresa la loro capienza. È sempre aggiornato circa eventuali cambi di aule, orari, professori, giorni di lezione.

ID:	A5
Nome:	Posta elettronica
Tipo:	Di supporto
Stereotipo:	Sistema
Descrizione:	Rappresenta l'applicativo di posta elettronica che viene usato per comunicare l'avvenuta prenotazione ad una lezione tramite la ricevuta via mail.

Use cases



Di seguito elenchiamo le descrizioni degli use cases del sottosistema **Gestione accesso**. Con il termine *sistema* si intende sempre BookSapienza.

ID Use Case	UC1
Nome	Effettua login
Priorità	1
Attori	UtenteNonAutenticato (rinominato utente in questo caso d'uso), Infostud, GOMP
Descrizione	Login di un utente
Pre-condizioni	L'utente non risulta loggato nel sistema
Flusso principale	<ol style="list-style-type: none"> 1. L'utente accede ad una qualsiasi pagina del sito web; 2. Il sistema chiede all'utente di inserire la matricola e la password per il login; 3. L'utente inserisce e conferma i dati; 4. Il sistema chiede ad Infostud di trovare l'utente che ha associati quei dati; 5. Infostud trova l'utente, i suoi dati e il suo percorso formativo, e approva l'accesso; 6. Il sistema verifica che l'utente rientri nella turnazione settimanale; 7. Il sistema chiede al GOMP di prelevare le informazioni sulle lezioni (dei corsi nel percorso formativo) che lo studente avrà nella settimana successiva, che contengono anche le informazioni sulle aule in cui si terranno tali lezioni; 8. GOMP trova le informazioni sulle lezioni richieste e le relative aule e le fornisce al sistema; 9. Il sistema reindirizza l'utente alla homepage.
Post-condizioni	L'utente risulta loggato nel sistema
Flusso alternativo #1	<ol style="list-style-type: none"> 5. Infostud rileva che non esiste un utente con quei dati associati 6. Il sistema segnala l'errore all'utente e chiede di inserire nuovamente i dati; 7. Il flusso riprende dal punto 3 del flusso principale.
Flusso alternativo #2	<ol style="list-style-type: none"> 4. Il sistema rileva che non tutti i campi (matricola, password) sono stati compilati; 5. Il sistema segnala l'errore all'utente e chiede di inserire nuovamente i dati; 6. Il flusso riprende dal punto 3 del flusso principale.
Flusso alternativo #3	<ol style="list-style-type: none"> 5. Il sistema rileva un fallimento nella comunicazione con l'applicativo Infostud; 6. Il sistema segnala l'errore all'utente.
Flusso alternativo #4	<ol style="list-style-type: none"> 8. Il sistema rileva un fallimento nella comunicazione con l'applicativo GOMP; 9. Il sistema segnala l'errore all'utente.

Use Case	UC2
Nome	Effettua logout
Priorità	1
Attori	Studente (rinominato utente in questo caso d'uso)
Descrizione	Logout di un utente
Pre-condizioni	L'utente risulta loggato nel sistema
Flusso principale	<ol style="list-style-type: none"> 1. L'utente chiede al sistema di effettuare il logout 2. Il sistema chiede all'utente la conferma di voler effettuare il logout; 3. L'utente dà la conferma; 4. Il sistema disconnette l'utente e gli comunica che l'operazione è avvenuta con successo.
Post-condizioni	L'utente non risulta loggato nel sistema
Flusso alternativo #1	<ol style="list-style-type: none"> 3. L'utente dà conferma negativa 4. Il sistema mantiene attiva la sessione dell'utente.

Di seguito elenchiamo le descrizioni degli use cases del sottosistema **Gestione prenotazioni**. Con il termine *sistema* si intende sempre BookSapienza.

ID Use Case	UC3
Nome	Visualizza lezioni prenotabili
Priorità	1
Attori	Studente (rinominato utente in questo caso d'uso)
Descrizione	L'utente visualizza le lezioni della settimana successiva che può prenotare
Pre-condizioni	L'utente deve essere loggato nel sistema
Flusso principale	<ol style="list-style-type: none"> 1. La matricola dell'utente rientra nella turnazione settimanale; 2. L'utente visualizza nella homepage la sua prossima settimana di lezioni.
Post-condizioni	Il sistema sta mostrando le lezioni della prossima settimana.
Flusso alternativo #1	<ol style="list-style-type: none"> 1. La matricola dell'utente non rientra nella turnazione settimanale; 2. L'utente visualizza nella homepage un avviso che lo informa che non è il suo turno.

ID Use Case	UC4
Nome	Prenota lezione
Priorità	1
Attori	Studente (rinominato utente in questo caso d'uso), Posta elettronica
Descrizione	Prenotazione di una lezione
Pre-condizioni	L'utente risulta loggato nel sistema. Il sistema sta mostrando le lezioni della prossima settimana. C'è almeno un posto libero nella lezione che l'utente vuole prenotare. L'utente non è già prenotato per quella lezione.
Flusso principale	<ol style="list-style-type: none"> 1. L'utente chiede al sistema di visualizzare maggiori informazioni sulla lezione che vuole prenotare; 2. L'utente chiede al sistema di effettuare la prenotazione per quella lezione; 3. Il sistema dà un riscontro che la prenotazione è avvenuta con successo; 4. Il sistema crea la ricevuta di prenotazione della lezione in formato PDF; 5. Il sistema invia all'indirizzo di posta elettronica dell'utente una mail contenente la ricevuta di prenotazione.
Post-condizioni	Il sistema contiene una nuova prenotazione. La lezione prenotata contiene un posto libero in meno.
Flusso alternativo #1	<ol style="list-style-type: none"> 5. Il sistema rileva un fallimento nella comunicazione con l'interfaccia di posta elettronica; 6. Il sistema segnala l'errore all'utente e ritenterà l'invio della mail successivamente.
Flusso alternativo #2	<ol style="list-style-type: none"> 3. Il sistema rileva che i posti nel frattempo sono esauriti; 4. Il sistema segnala l'errore all'utente e ricarica la pagina;

ID Use Case	UC5
Nome	Mettiti in coda d'attesa
Priorità	1
Attori	Studente (rinominato utente in questo caso d'uso)
Descrizione	Inserimento in coda d'attesa per una lezione

Pre-condizioni	L'utente risulta loggato nel sistema. Il sistema sta mostrando le lezioni della prossima settimana. Non c'è nessun posto libero nella lezione che l'utente vuole prenotare. L'utente non è né prenotato né in coda d'attesa per quella lezione.
Flusso principale	<ol style="list-style-type: none"> 1. L'utente chiede al sistema di visualizzare maggiori informazioni sulla lezione che vorrebbe prenotare ma in cui non ci sono posti disponibili; 2. L'utente chiede al sistema di mettersi in coda per quella lezione; 3. Il sistema dà un riscontro che l'inserimento in coda è avvenuto con successo;
Post-condizioni	Il sistema contiene una nuova prenotazione in coda.
Flusso alternativo #1	<ol style="list-style-type: none"> 3. Il sistema rileva che si sono liberati posti a lezione nel frattempo; 4. Il sistema segnala all'utente che si sono liberati dei posti e lo prenota a lezione;

ID Use Case	UC6
Nome	Visualizza prenotazioni
Priorità	1
Attori	Studente (rinominato utente in questo caso d'uso)
Descrizione	L'utente visualizza l'elenco delle prenotazioni effettuate e quelle per cui si è messo in coda
Pre-condizioni	L'utente deve essere loggato nel sistema
Flusso principale	<ol style="list-style-type: none"> 1. L'utente chiede al sistema di visualizzare le prenotazioni effettuate per una determinata settimana; 2. Il sistema mostra l'elenco delle prenotazioni per la settimana selezionata, facendo distinzione tra prenotazioni confermate e prenotazioni in coda d'attesa;
Post-condizioni	Il sistema sta mostrando l'elenco delle prenotazioni.

ID Use Case	UC7
Nome	Rimuoviti da una coda d'attesa
Priorità	2
Attori	Studente (rinominato utente in questo caso d'uso)
Descrizione	Rimozione da una coda d'attesa per una lezione

Pre-condizioni	L'utente risulta loggato nel sistema. Il sistema sta mostrando le lezioni della prossima settimana oppure l'elenco delle prenotazioni effettuate dall'utente. L'utente è in coda d'attesa per quella lezione.
Flusso principale	1. L'utente chiede al sistema di cancellare la prenotazione in coda per una lezione; 2. Il sistema dà un riscontro che la cancellazione è avvenuta con successo.
Post-condizioni	Il sistema contiene una prenotazione in coda in meno.
Flusso alternativo #1	2. Il sistema rileva che i posti nel frattempo sono scalati e quindi l'utente è stato prenotato automaticamente per la lezione; 3. Il sistema lo segnala all'utente e ricarica la pagina;

ID Use Case	UC8
Nome	Cancella prenotazione effettuata
Priorità	2
Attori	Studente (rinominato utente in questo caso d'uso)
Descrizione	Cancellazione di una prenotazione ad una lezione
Pre-condizioni	L'utente risulta loggato nel sistema. Il sistema sta mostrando le lezioni della prossima settimana oppure l'elenco delle prenotazioni effettuate dall'utente. L'utente è prenotato per quella lezione.
Flusso principale	1. L'utente chiede al sistema di cancellare la prenotazione per una lezione; 2. Il sistema dà un riscontro che la cancellazione è avvenuta con successo; 3. Il sistema verifica che ci sono studenti in coda d'attesa per quella lezione, e prenota automaticamente il primo;
Post-condizioni	Il sistema contiene una prenotazione in meno.
Flusso alternativo #1	3. Il sistema rileva che non ci sono studenti in coda di attesa per quella lezione; 4. Il sistema aumenta di uno i posti disponibili per quella lezione.

ID Use Case	UC9
Nome	Visualizza ricevuta
Priorità	3
Attori	Studente (rinominato utente in questo caso d'uso)
Descrizione	L'utente visualizza la ricevuta di una determinata prenotazione

Pre-condizioni	L'utente deve essere loggato nel sistema. L'utente deve aver effettuato con successo almeno una prenotazione.
Flusso principale	<ol style="list-style-type: none"> 1. L'utente chiede al sistema di visualizzare la ricevuta di una prenotazione correttamente effettuata; 2. Il sistema mostra un documento in formato PDF contenente la ricevuta di prenotazione.
Post-condizioni	Il sistema sta mostrando una ricevuta di prenotazione.

ID Use Case	UC10
Nome	Scarica ricevuta
Priorità	3
Attori	Studente (rinominato utente in questo caso d'uso)
Descrizione	L'utente scarica sul suo dispositivo la ricevuta di una determinata prenotazione
Pre-condizioni	L'utente deve essere loggato nel sistema. L'utente deve aver effettuato con successo almeno una prenotazione.
Flusso principale	<ol style="list-style-type: none"> 1. L'utente chiede al sistema di scaricare la ricevuta di una prenotazione correttamente effettuata; 2. Il sistema avvia il download del documento in formato PDF contenente la ricevuta di prenotazione; 3. L'utente salva la ricevuta sul suo dispositivo.
Post-condizioni	Nessuna
Flusso alternativo #1	2. Il download fallisce.

Matrice di tracciabilità requisiti-casi d'uso

La matrice di tracciabilità tra requisiti e casi d'uso aiuta a capire se tutti i requisiti funzionali richiesti dal sistema vengano soddisfatti dai casi d'uso.

I significati dei simboli nelle celle sono i seguenti:

- • : soddisfa, il caso d'uso nella riga soddisfa il requisito nella colonna.

(riga soddisfa colonna)	ID requisiti (colonne)	F1	F2	F3	F4	F5	F6	F7	F8
ID use cases (righe)									
UC1	•								
UC2									•
UC3		•							
UC4			•						
UC5					•				
UC6						•			
UC7						•			
UC8					•				
UC9								•	
UC10								•	

Analisi

Analisi nomi/verbi:

Progettare un sistema di prenotazione di posto a **lezione** (caratterizzata da un **id univoco**, il **nome del corso**, la **data**, un **orario**, un **docente** e un'**aula** in cui si svolge) nel rispetto della (nuova) **capienza delle aule** (caratterizzate da un **codice**, un **nome**, l'**indirizzo** e l'**edificio**). Tali informazioni sono prese dal **GOMP**. La presenza degli **studenti** è organizzata in turni prenotabili in base alle ultime due cifre del **numero di matricola** (da 00 a 49 e da 50 a 99) a settimane alterne. Il

sistema di prenotazione **acquisisce** automaticamente da **Infostud** le **informazioni personali** e il **percorso formativo** dello **studente**, in modo tale che possa **prenotare** selezionando solo la **lezione** che vuole seguire. La **prenotazione** per ciascun turno può **essere fatta** (e **cancellata**, per permettere a **studenti** in coda d'attesa di frequentare) solo dal lunedì a venerdì della settimana precedente. Nel momento in cui uno **studente** si vuole **prenotare** ad una **lezione** per cui non ci sono più **posti** disponibili **viene aggiunto** ad una **coda** di attesa. Se uno **studente** **esce** da una **coda** per una **lezione**, allora la posizione degli **studenti** in **coda** dopo di lui scalerà di un posto. Quando si libera un posto a **lezione**, il primo **studente** in lista di attesa **viene prenotato** automaticamente.

A **prenotazione** effettuata, il sistema **produce** una **ricevuta** (con **aula**, **nome studente**, **cognome**, **matricola**, **orario**, **data**, **indirizzo**, **posto in aula**, **corso**, **nome dell'edificio**, **QR Code** univoco) da **scaricare/salvare** che viene anche **inviata** per **email**. Per **accedere** al sistema servono le **credenziali** di Sapienza (**matricola** e **password** di **Infostud**).

Legenda:

- = Nomi
- = Verbi
- = Attributi

Schede CRC

Aula	
Super Classes:	
Sub Classes:	
Description: Aula dell'università La Sapienza	
Attributes:	
Name	Description
codice	Codice identificativo dell'aula
nome	Nome dell'aula
indirizzo	Indirizzo dell'aula
edificio	Nome dell'edificio in cui si trova l'aula
capienza	Capienza dell'aula
Responsibilities:	
Name	Collaborator

Powered By  Visual Paradigm Community Edition

Corso	
Super Classes:	
Sub Classes: Lezione	
Description: Classe che rappresenta un generico corso universitario.	
Attributes:	
Name	Description
codice	Codice identificativo del corso
nome	Nome del corso
docente	Nome del docente che tiene il corso
CFU	Numero di crediti dell'insegnamento
Responsibilities:	
Name	Collaborator

Powered By  Visual Paradigm Community Edition

Lezione	
Super Classes: Corso	
Sub Classes:	
Description: Classe che rappresenta una lezione di un corso universitario.	
Attributes:	
Name	Description
IDLezione	Identificativo della lezione
orario	Orario in cui si tiene la lezione
aula	Aula in cui si svolge la lezione
data	Data in cui si svolge la lezione
listaPrenotazioniEffettuate	Lista delle prenotazioni effettuate per la lezione
codaAttesa	Lista delle prenotazioni in coda per la lezione
postiLiberi	Numero di posti disponibili per la prenotazione
Responsibilities:	
Name	Collaborator

Powered By  Visual Paradigm Community Edition

Studente	
Super Classes:	
Sub Classes:	
Description: Utente autenticato nel sistema BookSapienza.	
Attributes:	
Name	Description
nome	Nome dello studente
cognome	Cognome dello studente
matricola	Matricola dello studente
email	Email dello studente
percorso formativo	Corsi dei quali lo studente deve sostenere gli esami
listaPrenotazioni	Lista di tutte le prenotazioni effettuate dall'utente
programmazioneSettimanale	Lista di tutte le lezioni che lo studente ha nella settimana successiva
puoPrenotare	Informazione che indica se lo studente si può prenotare alla successiva settimana di lezioni
Responsibilities:	
Name	Collaborator

Powered By  Visual Paradigm Community Edition

Prenotazione						
Super Classes:						
Sub Classes: PrenotazioneEffettuata, PrenotazioneInCoda						
Description: Classe astratta che rappresenta una generica prenotazione.						
Attributes:						
<table border="1"> <thead> <tr> <th>Name</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>ID studente</td> <td>identificativo della prenotazione studente che ha effettuato la prenotazione</td> </tr> <tr> <td>lezione</td> <td>lezione alla quale lo studente si vuole prenotare</td> </tr> </tbody> </table>	Name	Description	ID studente	identificativo della prenotazione studente che ha effettuato la prenotazione	lezione	lezione alla quale lo studente si vuole prenotare
Name	Description					
ID studente	identificativo della prenotazione studente che ha effettuato la prenotazione					
lezione	lezione alla quale lo studente si vuole prenotare					
Responsibilities:						
<table border="1"> <thead> <tr> <th>Name</th> <th>Collaborator</th> </tr> </thead> </table>	Name	Collaborator				
Name	Collaborator					
Powered By Visual Paradigm Community Edition						

PrenotazioneInCoda		
Super Classes: Prenotazione		
Sub Classes:		
Description: Classe che rappresenta l'utente in coda di attesa per una lezione piena.		
Attributes:		
<table border="1"> <thead> <tr> <th>Name</th> <th>Description</th> </tr> </thead> </table>	Name	Description
Name	Description	
Responsibilities:		
<table border="1"> <thead> <tr> <th>Name</th> <th>Collaborator</th> </tr> </thead> </table>	Name	Collaborator
Name	Collaborator	
Powered By Visual Paradigm Community Edition		

PrenotazioneEffettuata						
Super Classes: Prenotazione						
Sub Classes:						
Description: Prenotazione effettuata con successo da uno studente per una lezione.						
Attributes:						
<table border="1"> <thead> <tr> <th>Name</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>ricevuta</td> <td>documento in formato PDF che attesta la prenotazione effettuata</td> </tr> <tr> <td>posto</td> <td>posto dell'aula assegnato allo studente</td> </tr> </tbody> </table>	Name	Description	ricevuta	documento in formato PDF che attesta la prenotazione effettuata	posto	posto dell'aula assegnato allo studente
Name	Description					
ricevuta	documento in formato PDF che attesta la prenotazione effettuata					
posto	posto dell'aula assegnato allo studente					
Responsibilities:						
<table border="1"> <thead> <tr> <th>Name</th> <th>Collaborator</th> </tr> </thead> </table>	Name	Collaborator				
Name	Collaborator					
Powered By Visual Paradigm Community Edition						

HandlerCancellaDaCoda						
Super Classes:						
Sub Classes:						
Description: Classe che gestisce la rimozione di uno studente da una coda di attesa per una lezione.						
Attributes:						
<table border="1"> <thead> <tr> <th>Name</th> <th>Description</th> </tr> </thead> </table>	Name	Description				
Name	Description					
Responsibilities:						
<table border="1"> <thead> <tr> <th>Name</th> <th>Collaborator</th> </tr> </thead> <tbody> <tr> <td>Cancella una prenotazione dalla coda di attesa per una lezione</td> <td>Lezione , PrenotazioneInCoda</td> </tr> <tr> <td>Rimuove dalla lista di prenotazioni dello studente una prenotazione in coda</td> <td>Studente, PrenotazioneInCoda</td> </tr> </tbody> </table>	Name	Collaborator	Cancella una prenotazione dalla coda di attesa per una lezione	Lezione , PrenotazioneInCoda	Rimuove dalla lista di prenotazioni dello studente una prenotazione in coda	Studente, PrenotazioneInCoda
Name	Collaborator					
Cancella una prenotazione dalla coda di attesa per una lezione	Lezione , PrenotazioneInCoda					
Rimuove dalla lista di prenotazioni dello studente una prenotazione in coda	Studente, PrenotazioneInCoda					
Powered By Visual Paradigm Community Edition						

HandlerAggiungiACoda	
Super Classes:	
Sub Classes:	
Description: Classe che gestisce l'aggiunta di uno studente ad una coda di attesa per una lezione.	
Attributes:	
Name	Description
Responsibilities:	
Name	Collaborator
aggiunge alla coda di attesa della lezione che si vuole seguire una prenotazione in coda	Lezione , PrenotazioneInCoda
aggiunge alla lista di prenotazioni dello studente una prenotazione in coda	Studente, PrenotazioneInCoda

Powered By Visual Paradigm Community Edition

HandlerCancellaPrenotazioneEffettuata	
Super Classes:	
Sub Classes:	
Description: Classe che gestisce la cancellazione di una prenotazione effettuata.	
Attributes:	
Name	Description
Responsibilities:	
Name	Collaborator
Cancella una prenotazione effettuata dalla lista di prenotazioni per una lezione	Lezione , PrenotazioneEffettuata
rimuovi la prenotazione effettuata dalla lista delle prenotazioni dello studente	Studente, PrenotazioneEffettuata
controlla se c'è qualche studente in coda di attesa per questa lezione e in tal caso lo prenota	HandlerCancellaDaCoda, HandlerPrenotazioneEffettuata, Lezione
se non c'era nessuno studente in coda d'attesa aumenta di 1 il numero di posti disponibili per quella lezione	Lezione

Powered By Visual Paradigm Community Edition

HandlerLogin	
Super Classes:	
Sub Classes:	
Description: Classe che gestisce l'operazione di login.	
Attributes:	
Name	Description
Responsibilities:	
Name	Collaborator
effettua il login se le credenziali sono corrette	InterfacciaInfoStud
se il login è andato a buon fine e lo studente rientra nella turnazione prevista, richiede le informazioni sulle lezioni settimanali dei corsi presenti nel suo percorso formativo	InterfacciaGOMP , Studente , Utility

Powered By Visual Paradigm Community Edition

HandlerLogout	
Super Classes:	
Sub Classes:	
Description: Classe che gestisce l'operazione di logout.	
Attributes:	
Name	Description
Responsibilities:	
Name	Collaborator
effettua il logout e reindirizza alla schermata di login	Studente

Powered By Visual Paradigm Community Edition

HandlerPrenotazioneE ffectuata	
Super Classes:	
Sub Classes:	
Description: Classe che gestisce la prenotazione andata a buon fine di un posto in aula.	
Attributes:	
Name	Description
Responsibilities:	
Name	Collaborator
aggiunge la prenotazione alla lista di prenotazioni per quella lezione, assegna il posto allo studente e diminuisce di 1 i posti disponibili	Lezione, PrenotazioneE ffectuata
aggiunge la prenotazione alla lista delle prenotazioni dello studente	Studente, PrenotazioneEffettuata
crea e invia la ricevuta di prenotazione via mail allo studente	InterfacciaPosta

HandlerRicevutaPrenotazione	
Super Classes:	
Sub Classes:	
Description: Classe che gestisce le azioni eseguibili sulla ricevuta di prenotazione ad una lezione	
Attributes:	
Name	Description
Responsibilities:	
Name	Collaborator
permette allo studente di visualizzare la ricevuta di prenotazione per una lezione	PrenotazioneEffettuata
permette allo studente di scaricare sul proprio dispositivo la ricevuta di prenotazione per una lezione	PrenotazioneEffettuata

Powered By Visual Paradigm Community Edition

Powered By Visual Paradigm Community Edition

InterfacciaInfostud	
Super Classes:	
Sub Classes:	
Description: Classe che si interfaccia con il sistema Infostud	
Attributes:	
Name	Description
Responsibilities:	
Name	Collaborator
Controlla che le credenziali dello studente per il login esistano in Infostud e permette l'accesso se sono corrette	Studente
Preleva le informazioni personali dello studente e il suo percorso formativo	Studente, Corso

Powered By Visual Paradigm Community Edition

InterfacciaGOMP	
Super Classes:	
Sub Classes:	
Description: Classe che si interfaccia con il sistema GOMP.	
Attributes:	
Name	Description
Responsibilities:	
Name	Collaborator
reperisce le informazioni sulle lezioni dei corsi che si terranno nella settimana successiva e che si trovano nel percorso formativo dello studente	Corso, Lezione, Studente
reperisce informazioni sulle aule in cui si svolgono le lezioni	Aula, Lezione, Studente

Powered By Visual Paradigm Community Edition

InterfacciaPosta	
Super Classes:	
Sub Classes:	
Description: Interfaccia del sistema di posta elettronica.	
Attributes:	
Name	Description
Responsibilities:	
Name	Collaborator
Invia una mail con una ricevuta allo studente che si è prenotato	PrenotazioneEffettuata

Powered By Visual Paradigm Community Edition

UILezioni	
Super Classes:	
Sub Classes:	
Description: Classe che rappresenta gli elementi dell'interfaccia che permette di visualizzare la propria settimana di lezioni. Corrisponde alla homepage del sistema.	
Attributes:	
Name	Description
studente	utente autenticato nel sistema
Responsibilities:	
Name	Collaborator
se lo studente rientra nella turnazione prevista, mostra a schermo le informazioni sulle lezioni della settimana	Studente
altrimenti mostra a schermo un'informazione che indica che non rientra nella turnazione	Studente

Powered By Visual Paradigm Community Edition

UILogin	
Super Classes:	
Sub Classes:	
Description: Classe che rappresenta l'insieme degli elementi dell'interfaccia dell'accesso al sistema	
Attributes:	
Name	Description
Responsibilities:	
Name	Collaborator
visualizza i form per l'inserimento dei dati	
invia i dati alle classi di gestione	HandlerLogin
visualizza i messaggi di risposta	HandlerLogin

Powered By Visual Paradigm Community Edition

UIPrenotazione	
Super Classes:	
Sub Classes:	
Description: Classe che rappresenta l'insieme degli elementi dell'interfaccia che permette di effettuare o cancellare una prenotazione per una lezione	
Attributes:	
Name	Description
studente	utente autenticato nel sistema
lezione	lezione per la quale stiamo visualizzando maggiori informazioni
Responsibilities:	
Name	Collaborator
visualizza le informazioni dettagliate della lezione	
invia i dati alla classe di gestione della prenotazione	HandlerPrenotazioneEffettuata, HandlerCancellaPrenotazioneEffettuata, HandlerAggiungiACoda, HandlerCancellaDaCoda
visualizza i messaggi di risposta	HandlerPrenotazioneEffettuata, HandlerCancellaPrenotazioneEffettuata, HandlerAggiungiACoda, HandlerCancellaDaCoda

Powered By Visual Paradigm Community Edition

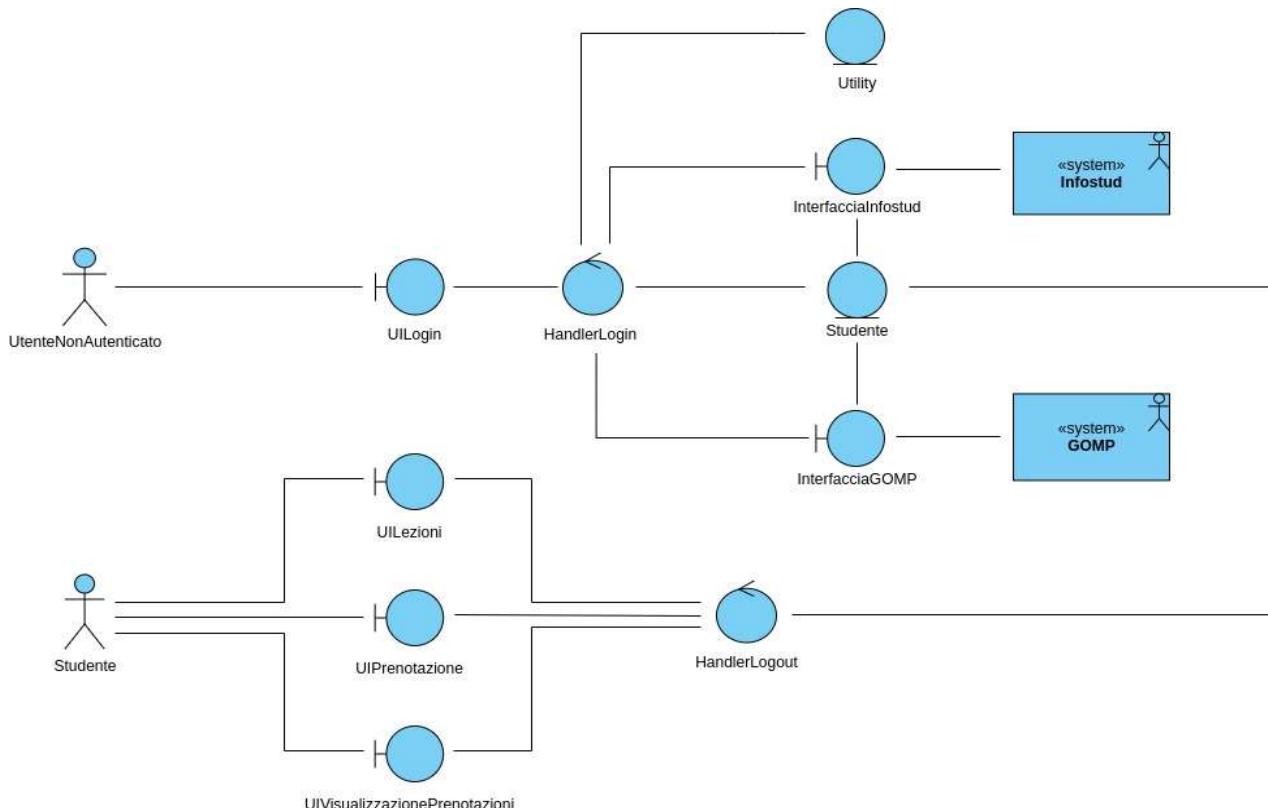
UIVisualizzaPrenotazioni	
Super Classes:	
Sub Classes:	
Description: Classe che rappresenta gli elementi dell'interfaccia che permette di visualizzare le prenotazioni delle lezioni effettuate dallo studente e le relative ricevute	
Attributes:	
Name	Description
studente	utente autenticato nel sistema
settimana	settimana per la quale sono visualizzate le prenotazioni
Responsibilities:	
Name	Collaborator
visualizza l'elenco delle prenotazioni effettuate dallo studente e le prenotazioni per cui è in coda in una data settimana	
invia i dati alla classi di cancellazione della prenotazione e di gestione della ricevuta	HandlerCancellaPrenotazioneEffettuata, HandlerCancellaDaCoda, HandlerRicevutaPrenotazione
visualizza i messaggi di risposta	HandlerCancellaPrenotazioneEffettuata, HandlerCancellaDaCoda, HandlerRicevutaPrenotazione

Utility	
Super Classes:	
Sub Classes:	
Description: Classe necessaria per raccogliere informazioni utili a tutto il sistema e promuovere la manutenibilità.	
Attributes:	
Name	Description
turnazione	indica quali matricole si possono prenotare per la prossima settimana
Responsibilities:	
Name	Collaborator
fornire informazioni utili al sistema	

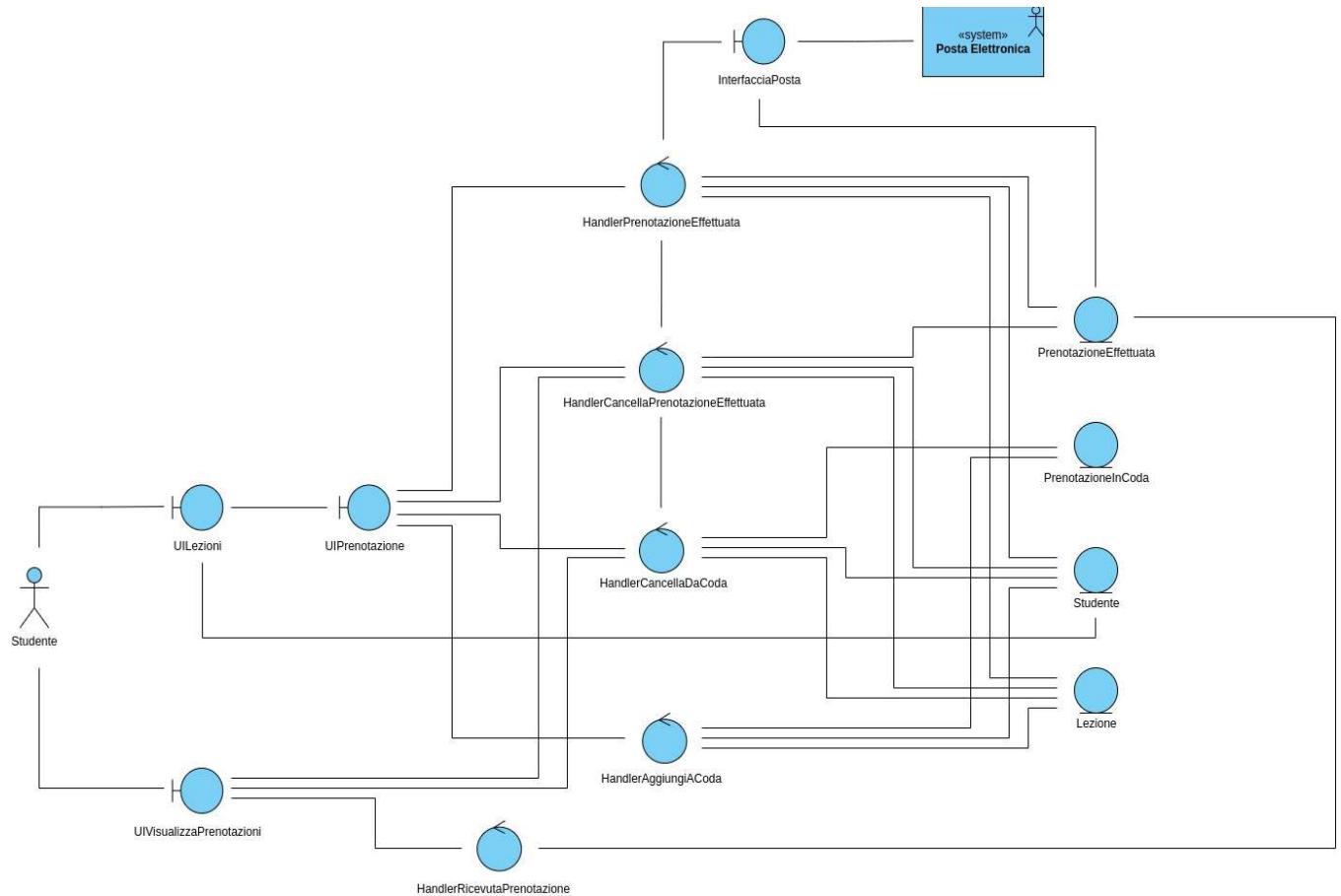
Powered By Visual Paradigm Community Edition

Classi di analisi (stereotipi entity/boundary/control)

Sottosistema gestione accesso:

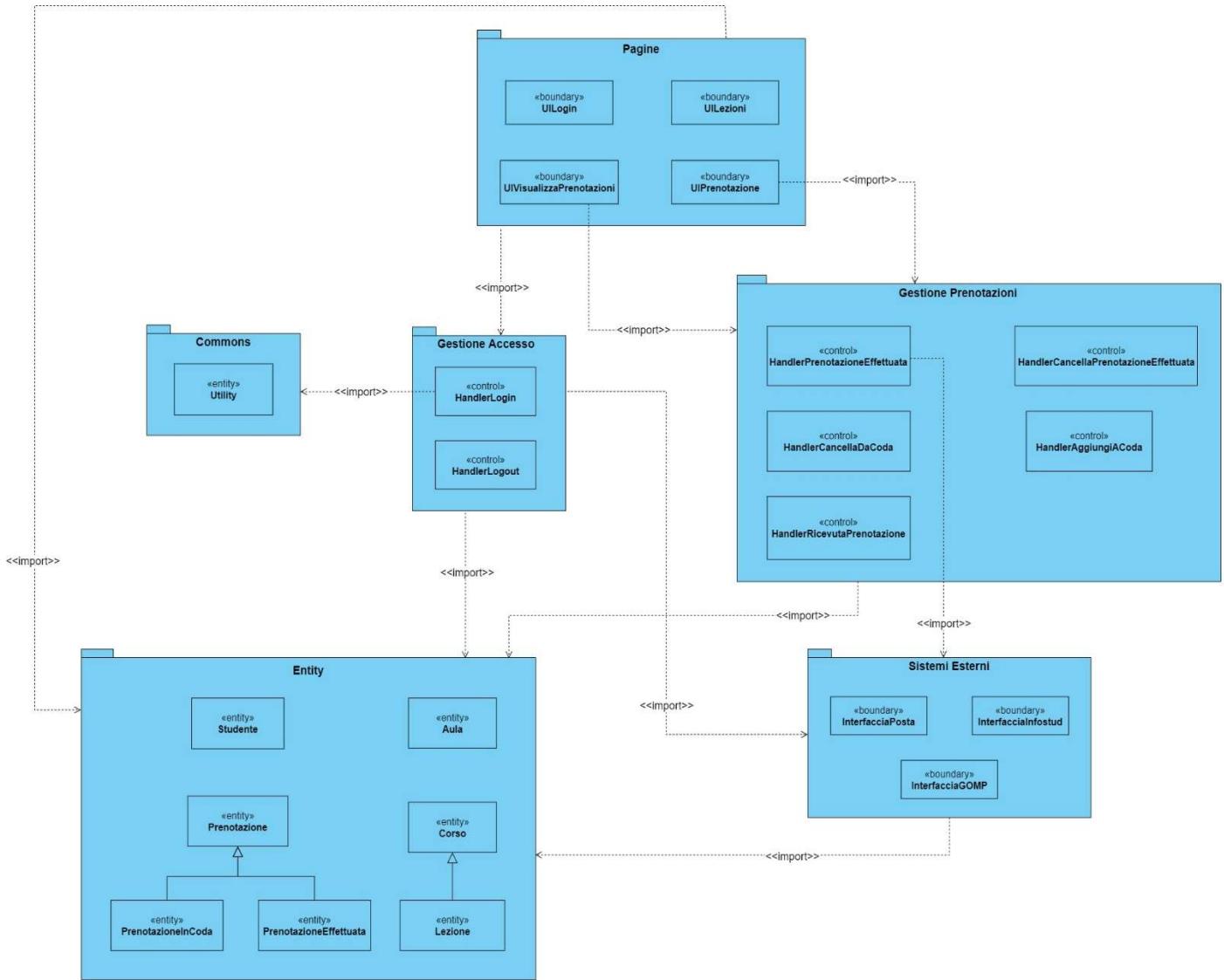


Sottosistema gestione prenotazioni:



Organizzazione in package

Descriviamo i package di analisi che compongono il sistema e le relazioni tra essi, per realizzarli abbiamo suddiviso gli elementi in relazione semantica fra loro.



Progetto

Specifiche dell'architettura

BookSapienza è una applicazione web che sfrutta un'architettura di tipo client-server.

In particolare il server è composto da:

- web server, che ascolta le richieste provenienti dalla rete;
- application server, che implementa la logica dell'applicazione;
- RDBMS (si è scelto MySQL) che gestisce l'accesso ad un database relazionale. Il database si occupa di contenere le prenotazioni effettuate dallo studente (sia quelle confermate sia quelle per cui lui è in coda d'attesa), gli studenti prenotati ad ogni lezione, le code di attesa per ogni lezione, il numero di posti disponibili ad ogni lezione. I dati nel database saranno protetti tramite crittografia.

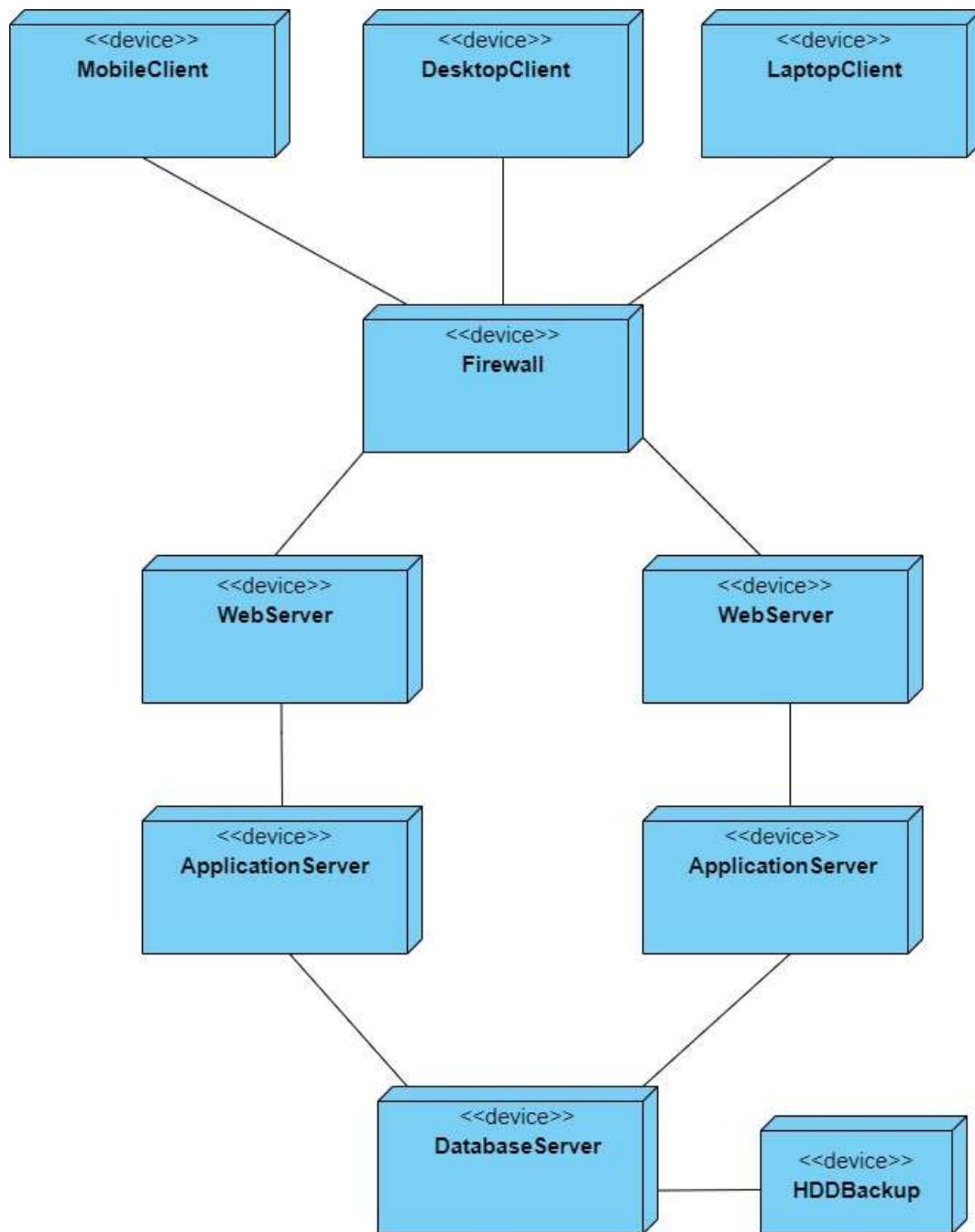
I terminali client (desktop, laptop e mobile) degli utenti comunicano con il web server tramite un comune web browser e sfruttano il protocollo HTTPS. BookSapienza si interfacerà, tramite protocolli standard di comunicazione (quali HTTPS e TCP/IP), con i sistemi esterni già esistenti, tra cui:

- Infostud, per reperire le informazioni personali dello studente e i corsi nel suo percorso formativo;
- GOMP, per ottenere le informazioni sulle lezioni nel percorso formativo dello studente che si terranno nella prossima settimana e sulle aule in cui tali lezioni si svolgono;
- il server di posta elettronica, per l'invio agli studenti delle e-mail contenenti le ricevute di prenotazioni.

Per garantire maggiore sicurezza del sistema, esso sarà protetto da un firewall che risiederà su un server apposito e si occupa di controllare e filtrare le richieste provenienti dai client.

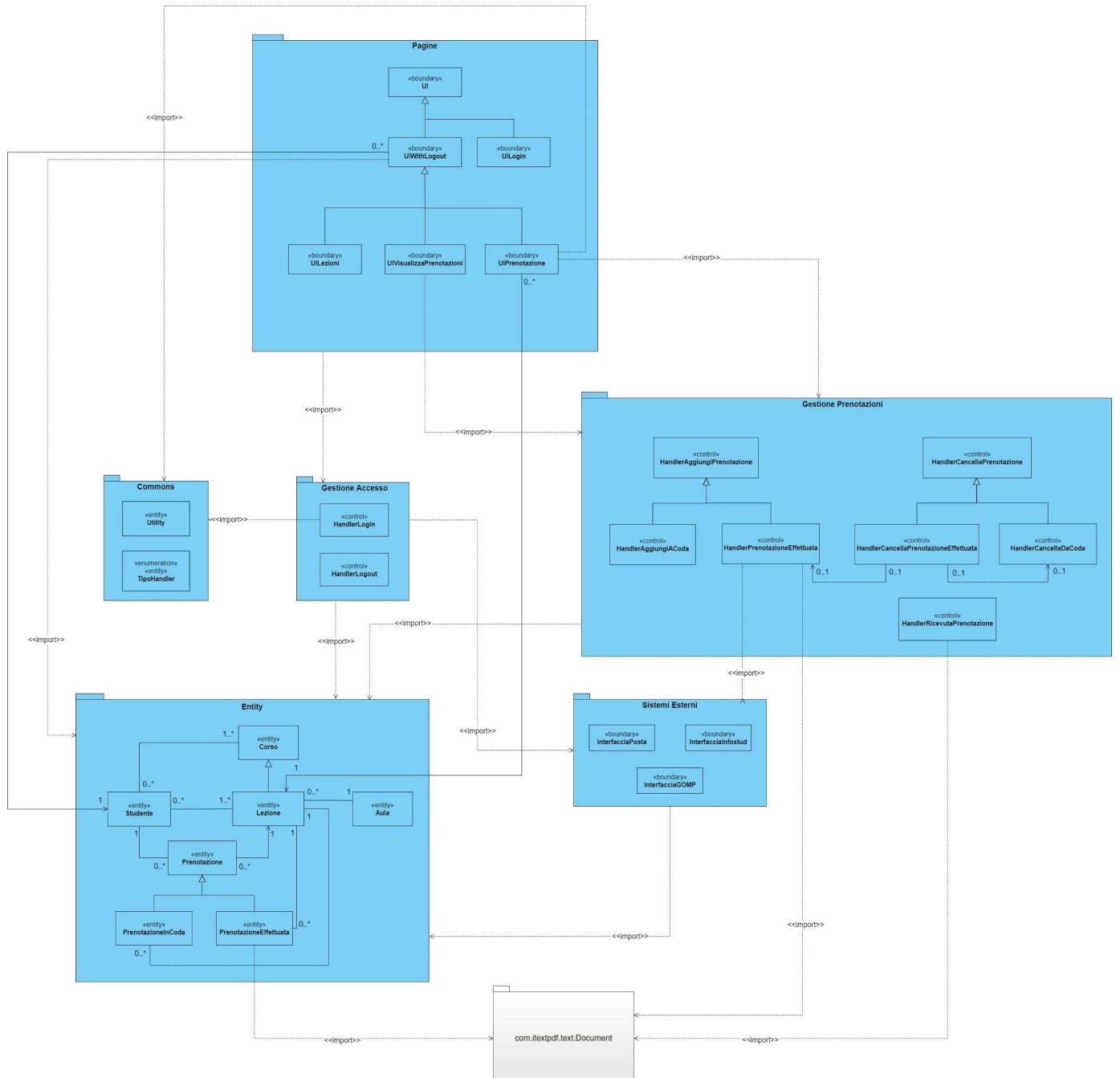
Per permettere la reperibilità del sistema, evitare singoli point of failure e resistere ad eventuali manomissioni o fallimenti, BookSapienza sarà in esecuzione su due application server in modo ridondante. Il firewall eseguirà un bilanciamento del carico delle richieste per suddividerle tra i due web server presenti, anch'essi ridondanti. In questo modo il sistema potrà garantire alta affidabilità anche in caso di fail da parte di un server. Inoltre l'aggiornamento e la manutenzione potranno essere effettuate senza dover spegnere il sistema, eseguendole prima su un server e poi sull'altro. Il database sarà unico ma si prevede inoltre un supporto di archiviazione criptato per il backup (sia dei dati del database che dell'intero sorgente del sistema), in modo da poter rimettere in piedi il tutto in caso di grossi fallimenti. Il sistema sarà implementato in Java e la comunicazione con il database avverrà tramite le API JDBC.

Il seguente diagramma mostra come le componenti interagiscono e sono distribuite in rete.

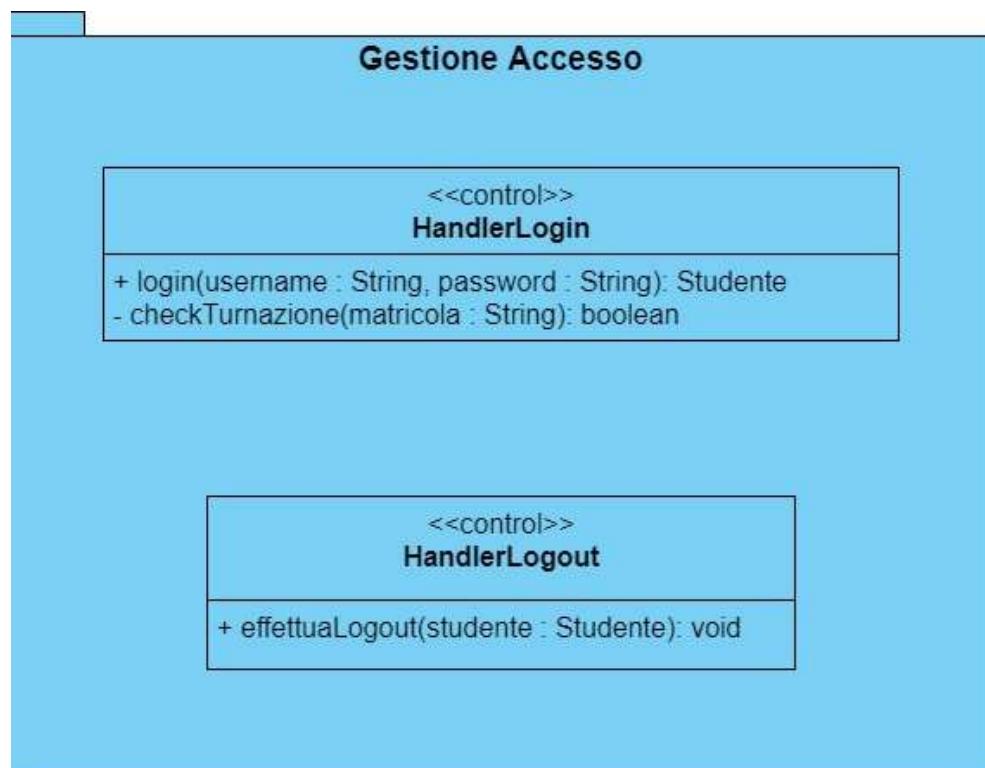
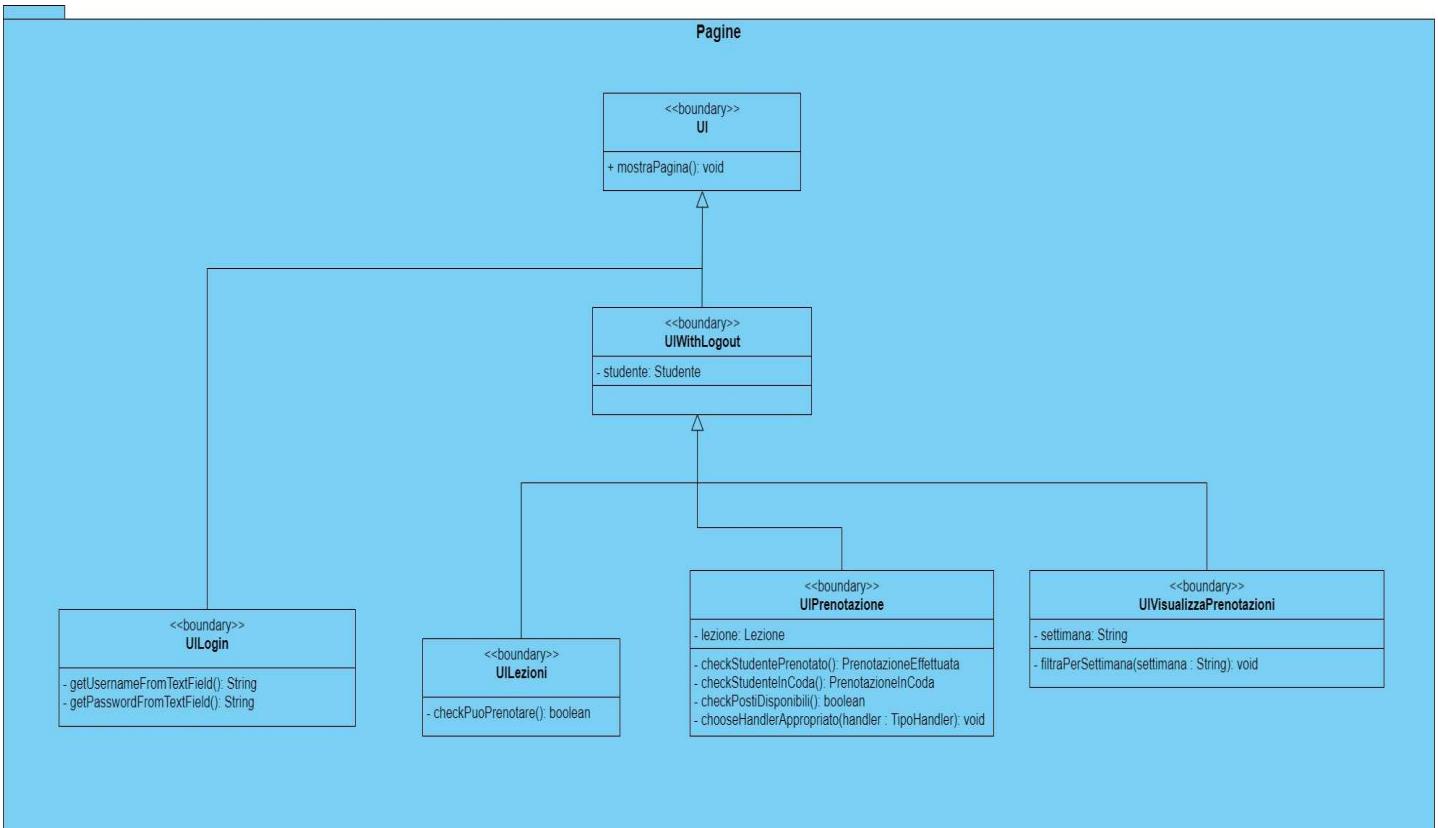


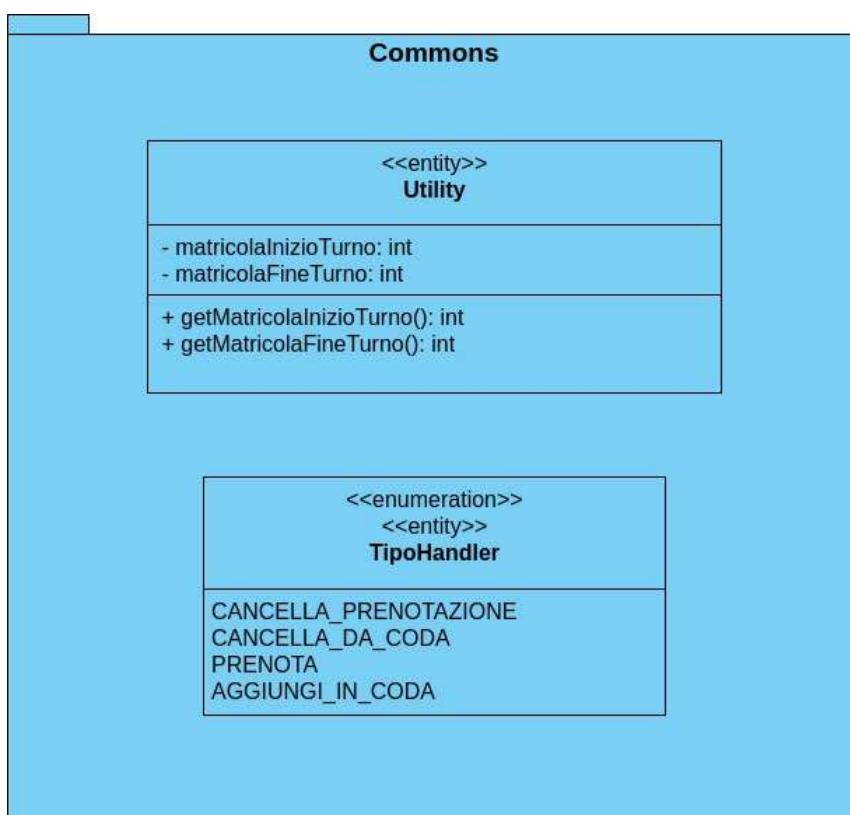
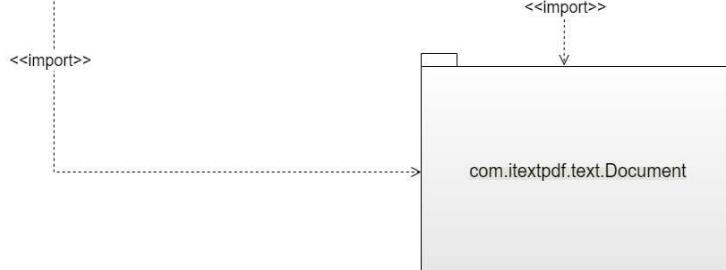
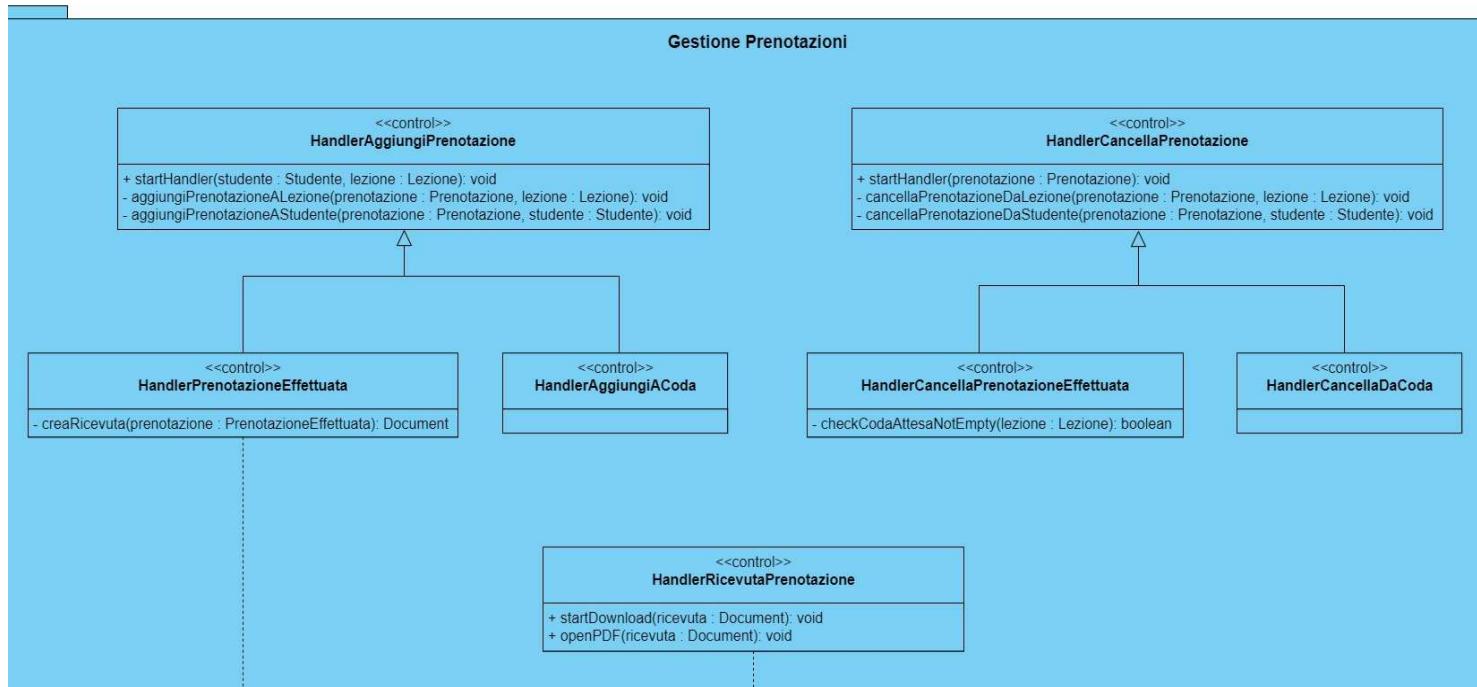
Classi di progetto, sottosistemi, componenti

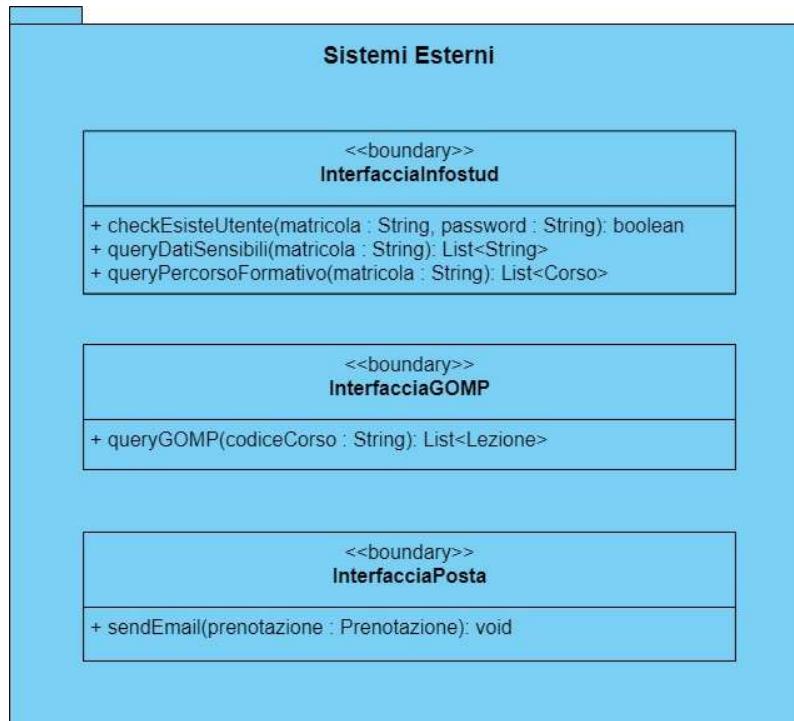
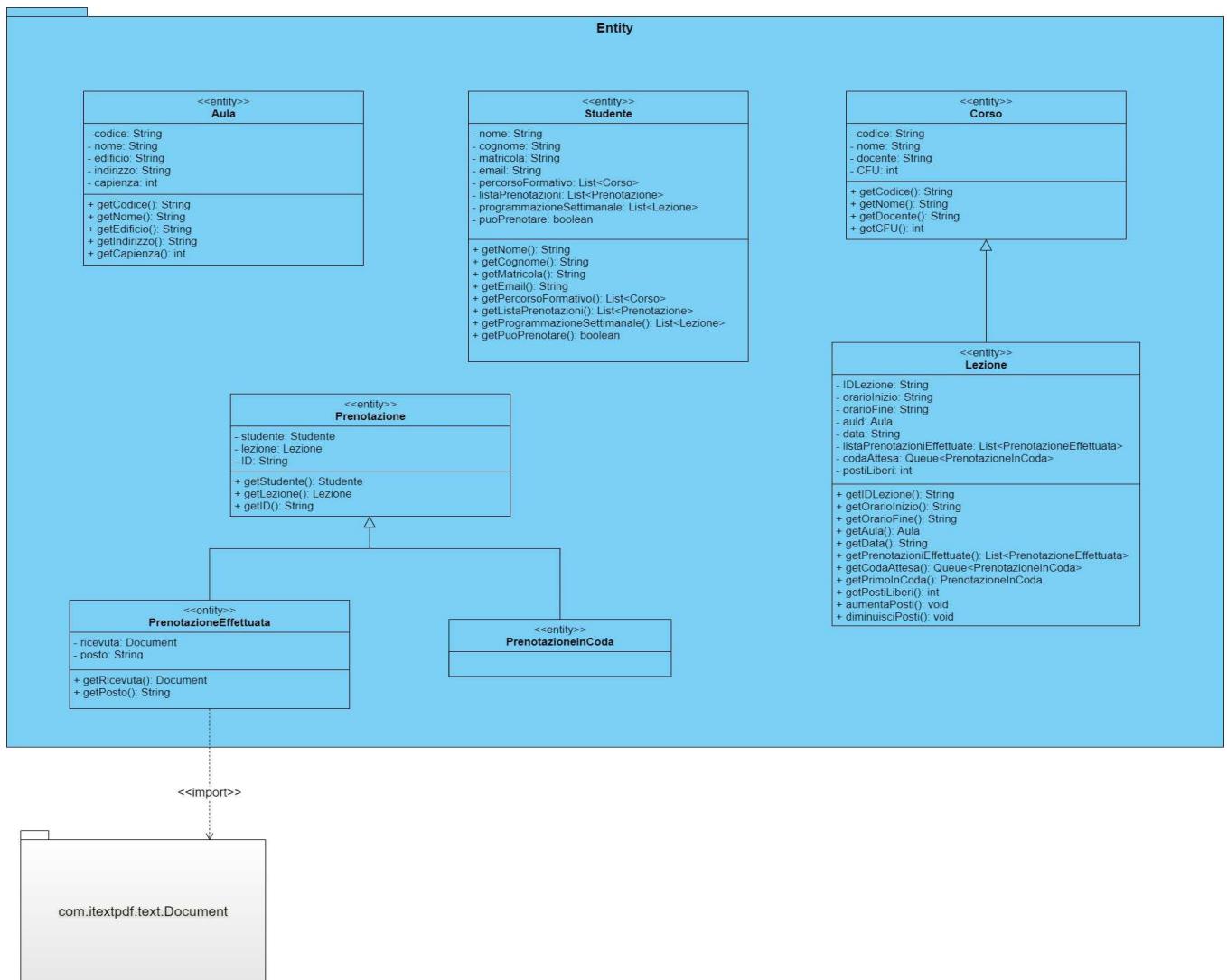
Riportiamo innanzitutto la suddivisione in package di design, che contengono le classi del sistema e le relazioni che intercorrono tra di esse. Le classi sono quelle di analisi più ulteriori classi nate in fase di design che ci sembravano appropriate per una buona implementazione in un linguaggio orientato agli oggetti come Java.



Per ogni package tutte le classi sono state dettagliate con attributi e metodi che possiedono, li riportiamo uno alla volta per maggiore leggibilità, tralasciando le relazioni che sono già visibili nello schema soprastante.



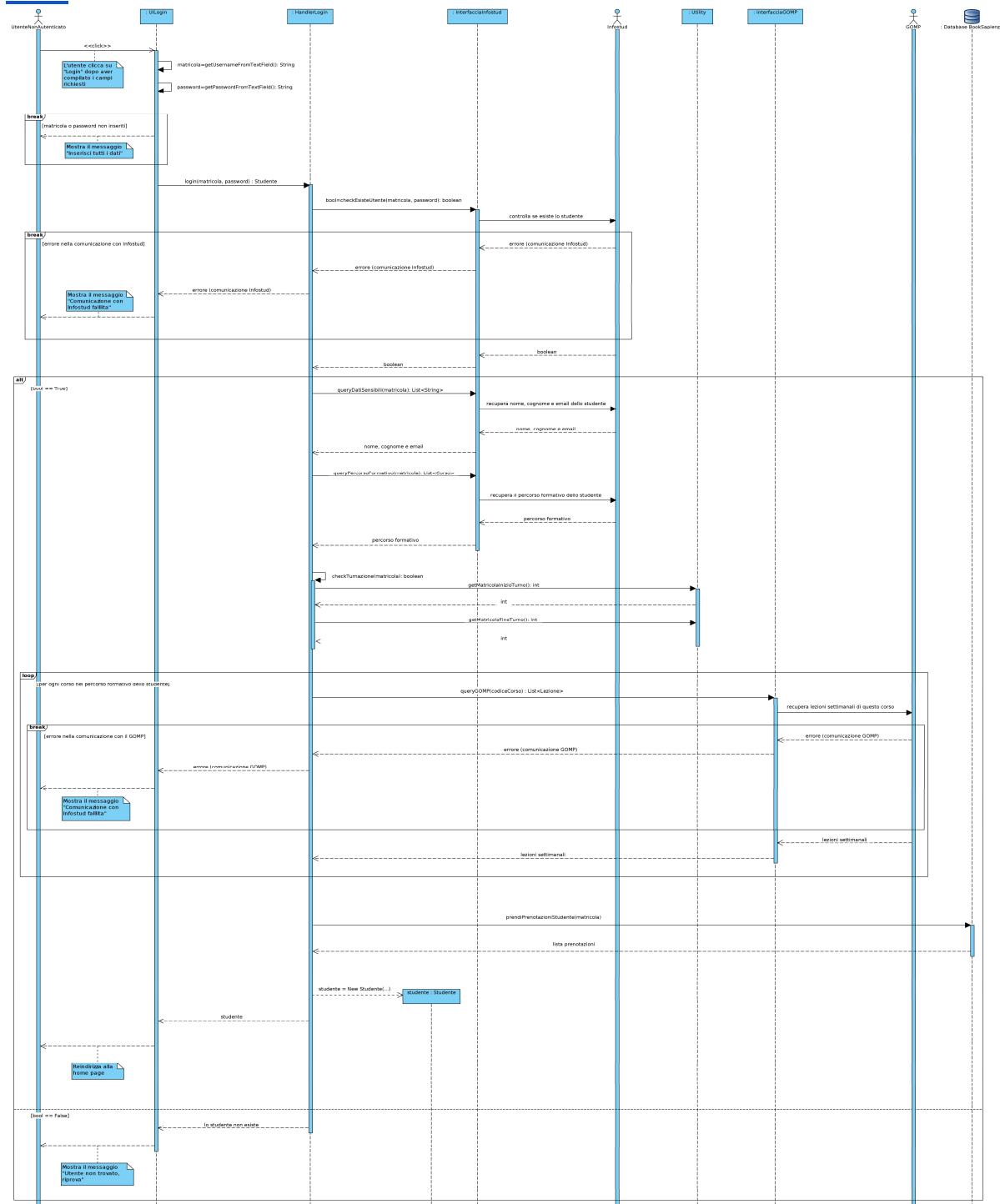




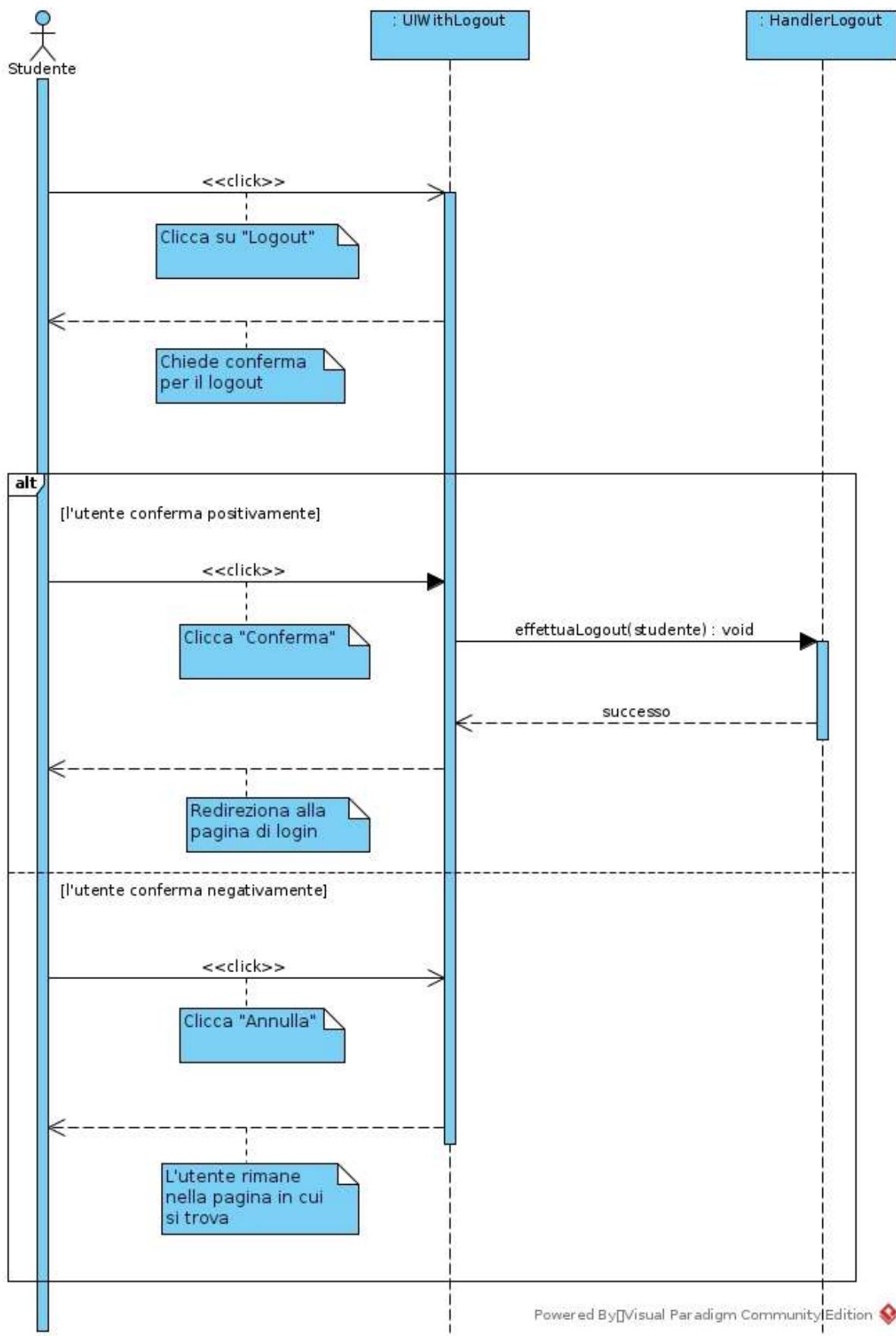
Realizzazione dei casi d'uso (sequence diagrams)

Di seguito vengono illustrati i sequence diagrams, che sono le rappresentazioni esplicite dei casi d'uso. Nei seguenti diagrammi per comunicare con il database abbiamo assunto di avere dei metodi (con nomi autoesPLICATIVI) che rappresentano le operazioni di SELECT, INSERT e di UPDATE. È stato inoltre usato nei diagrammi lo stereotipo <>click>> (che rappresenta il click dell'utente su un componente dell'interfaccia) per facilitare la comprensione della comunicazione con l'interfaccia utente.

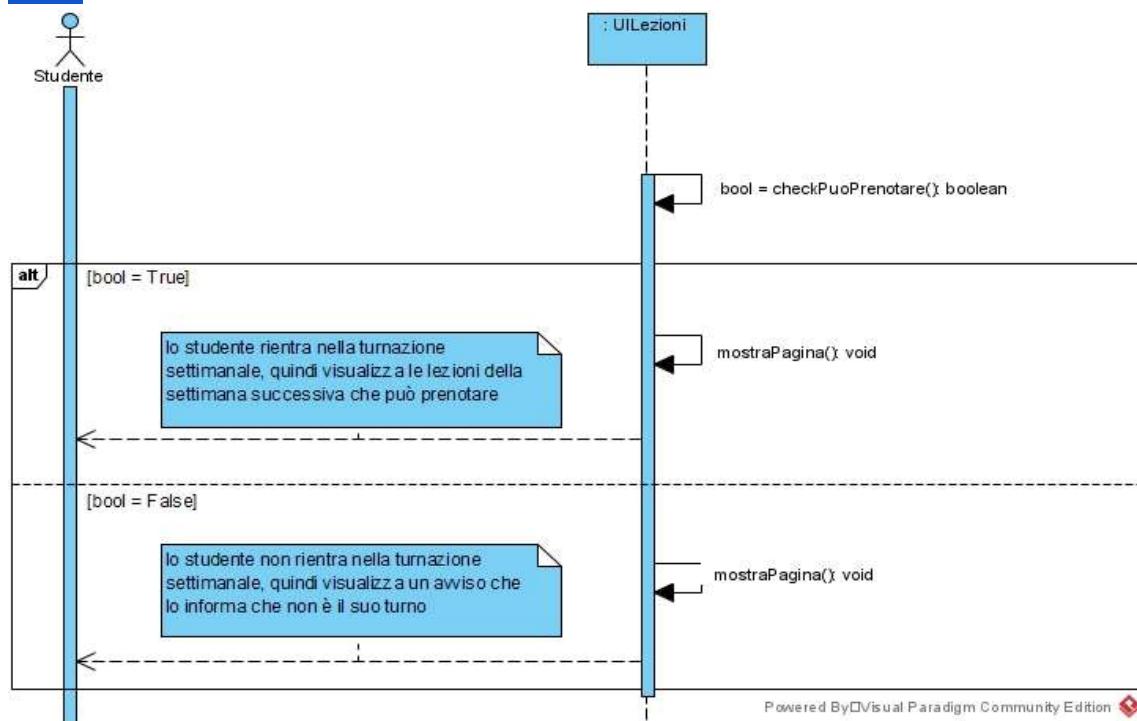
UC1



UC2

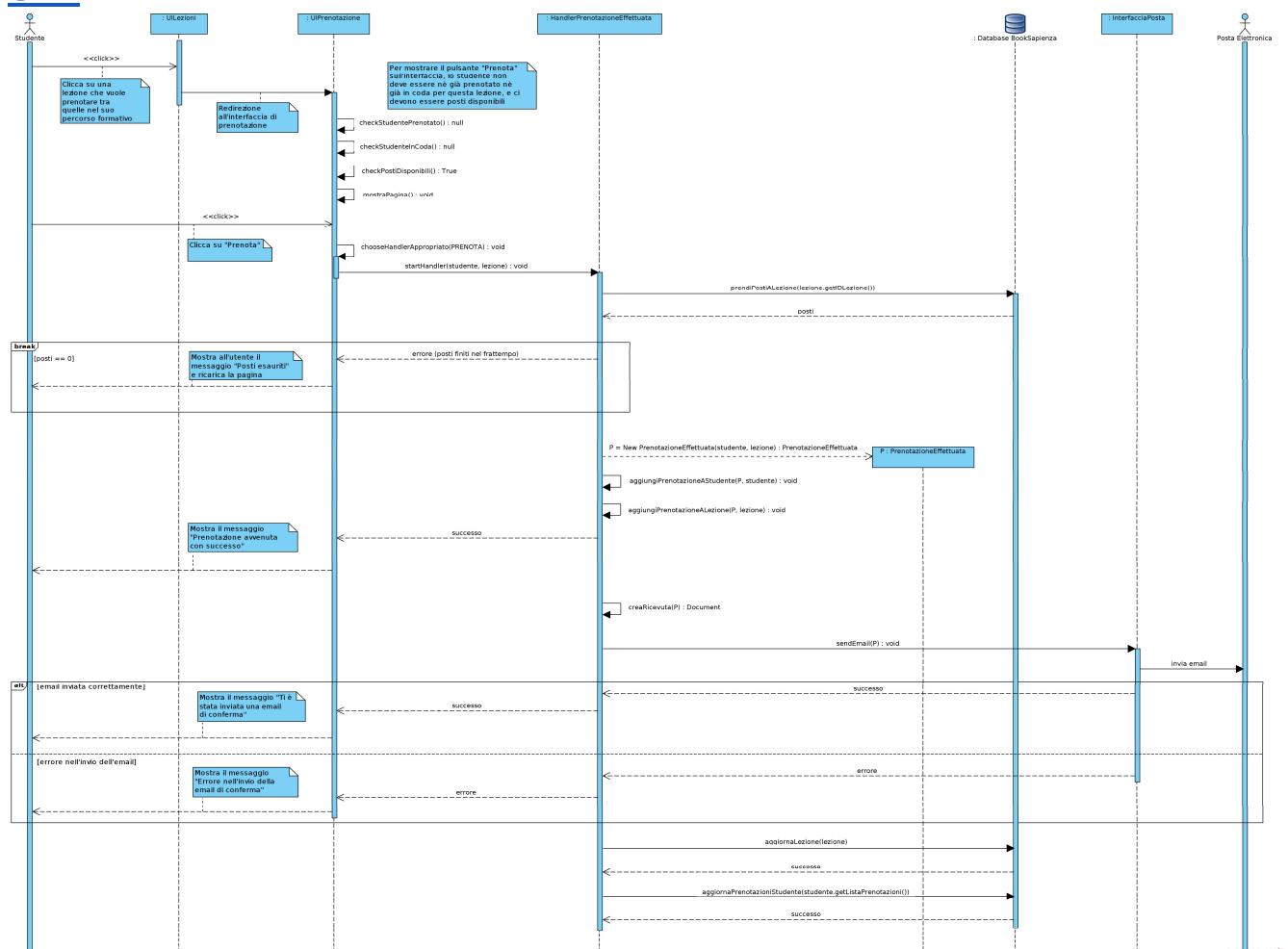


UC3

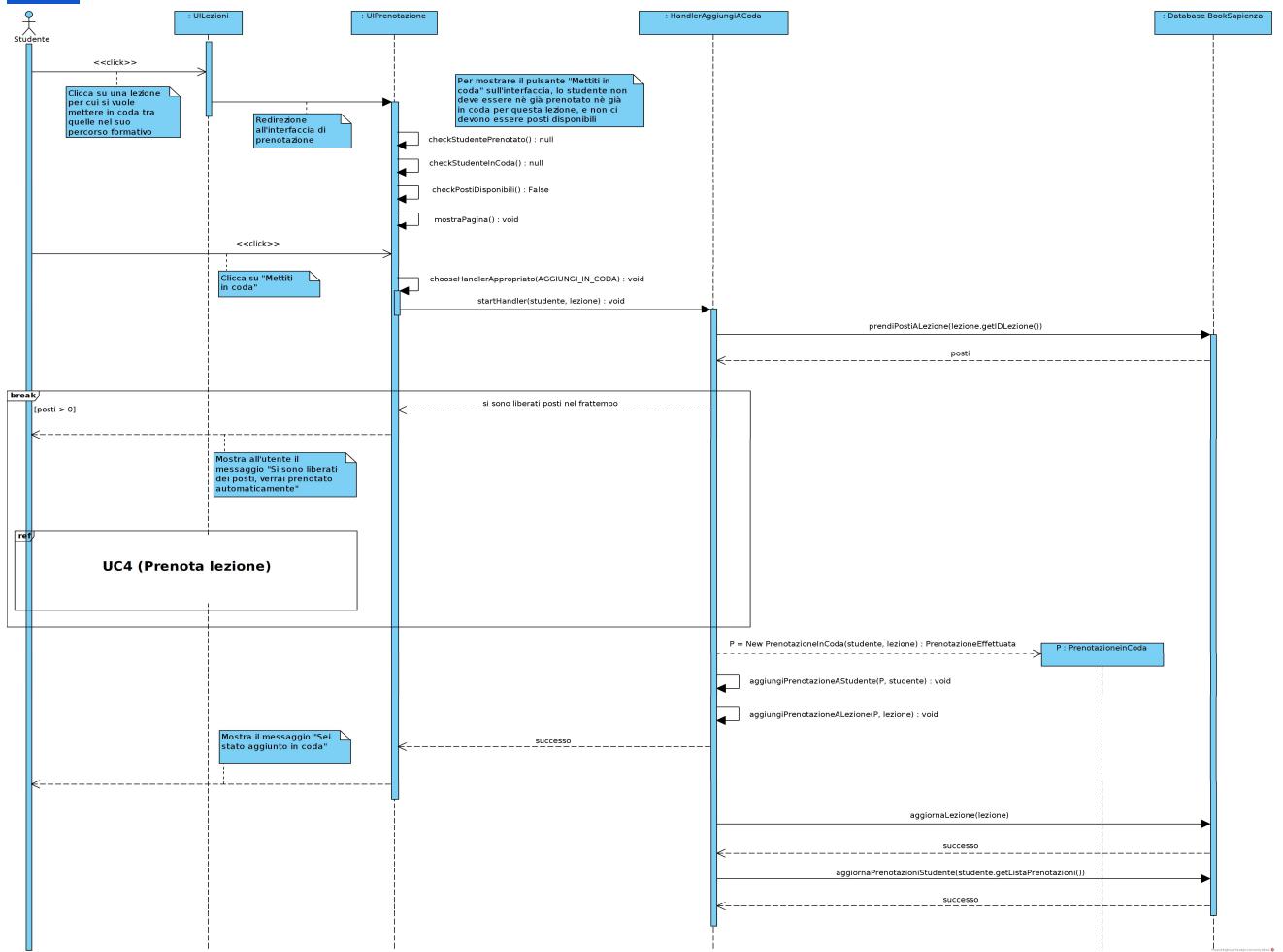


Powered By Visual Paradigm Community Edition

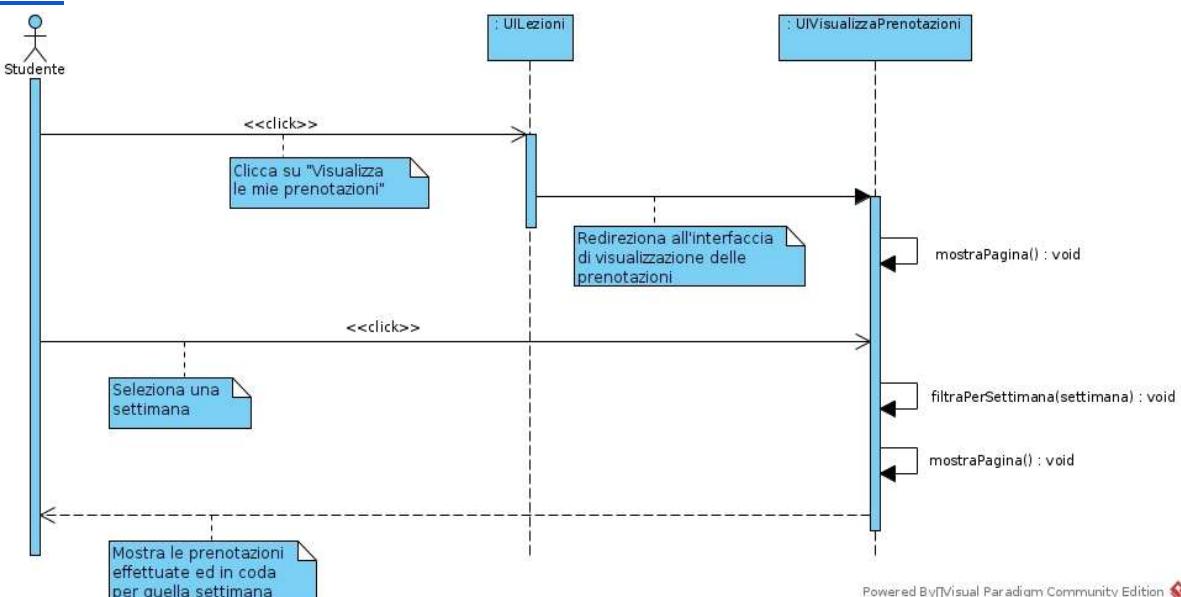
UC4



UC5

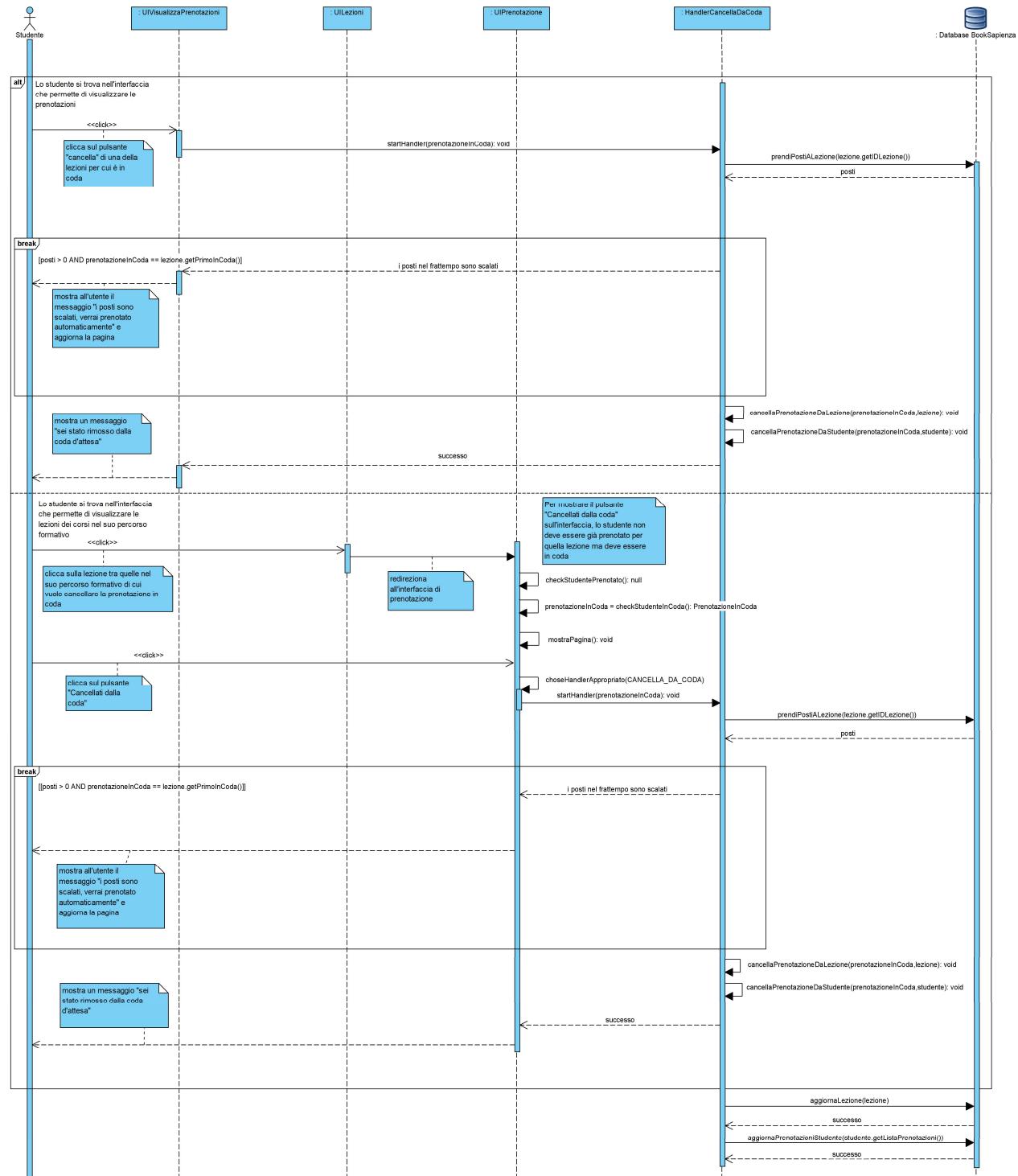


UC6

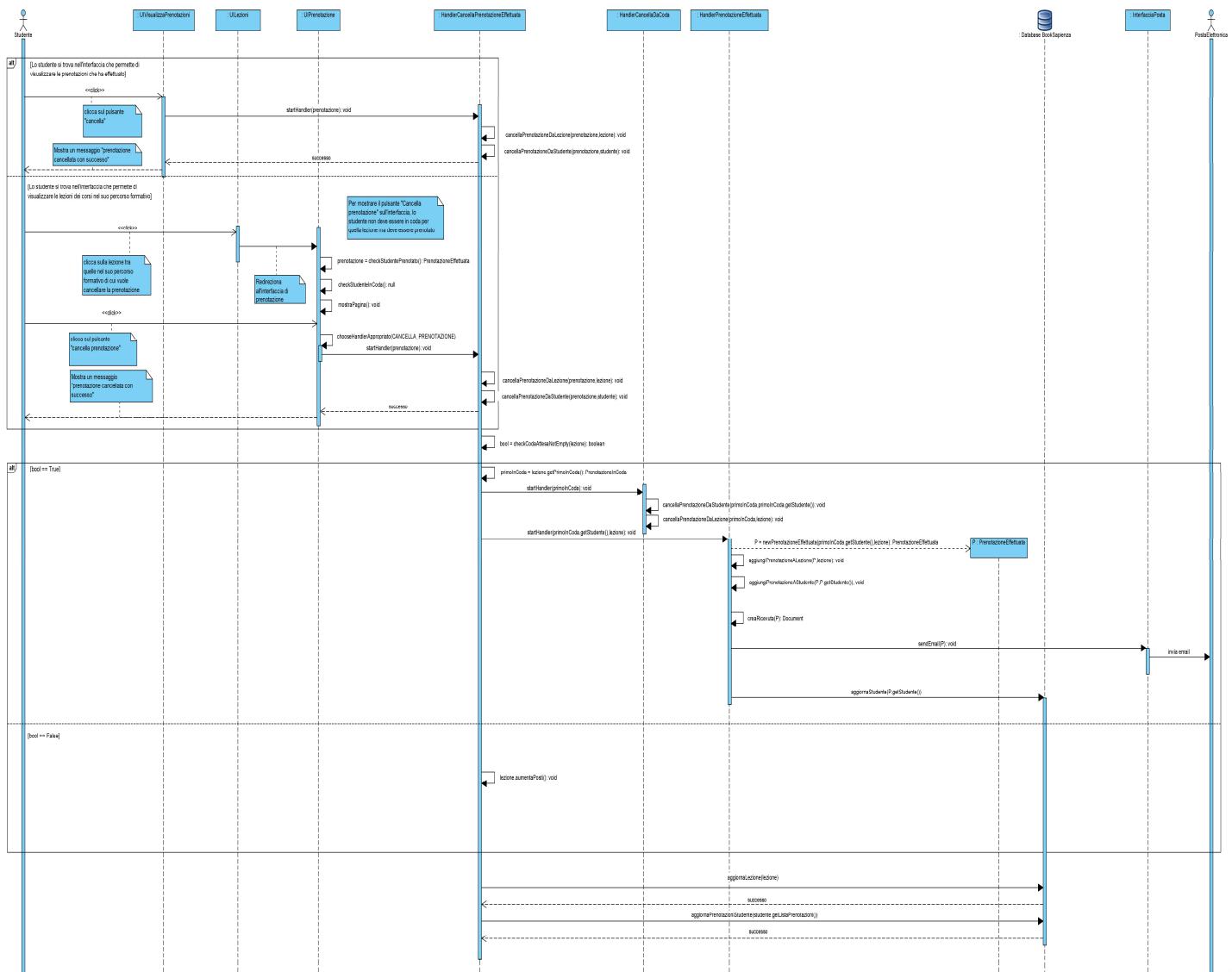


Powered By Visual Paradigm Community Edition

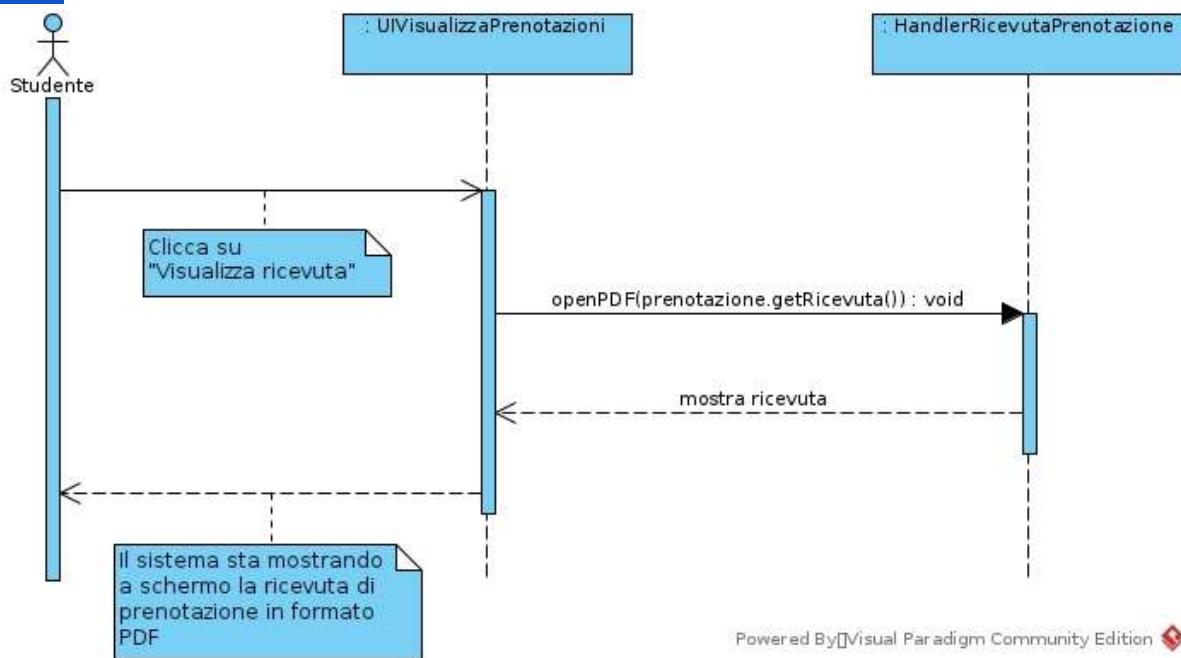
UC7



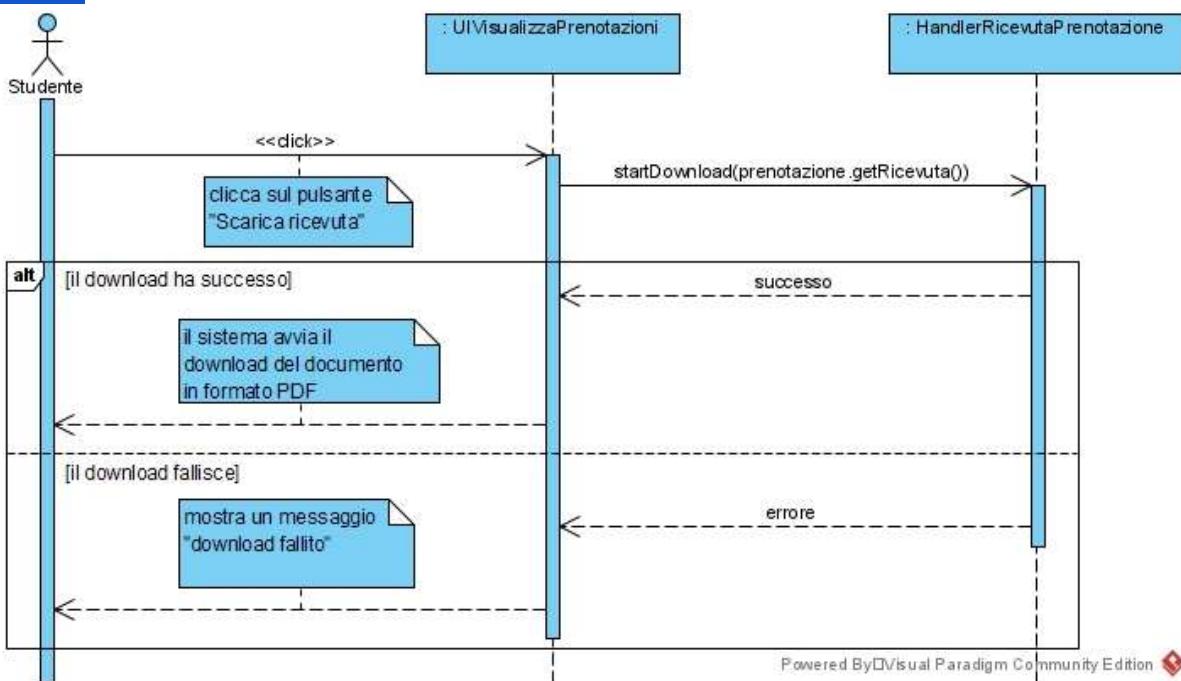
UC8



UC9



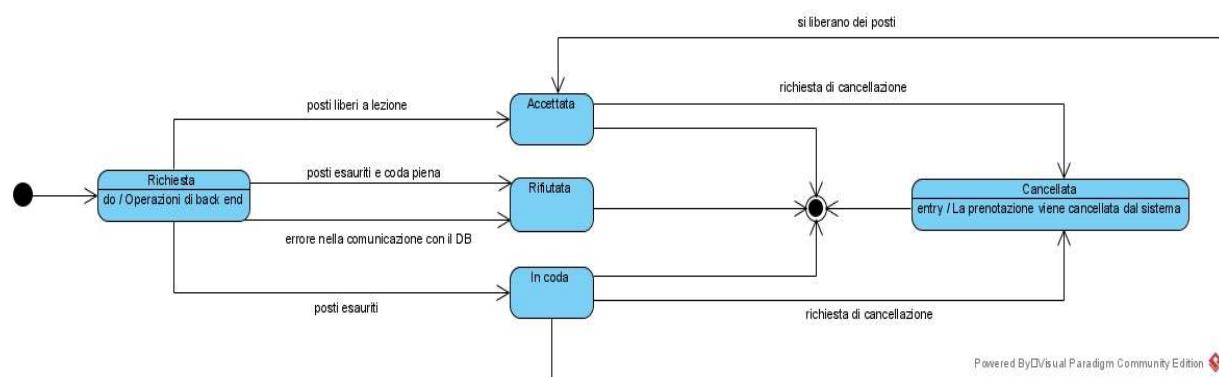
UC10



Cicli di vita di classi significative

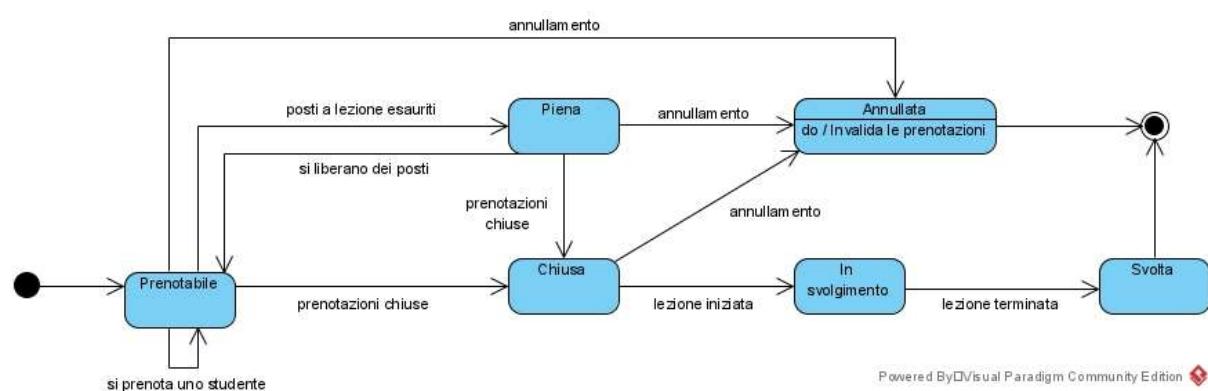
Prenotazione

L'istanza di Prenotazione inizia la sua vita nello stato *Richiesta* (l'utente vuole prenotarsi ad una lezione), e può transitare nei tre differenti stati *Accettata*, *Rifiutata* e *In coda*. Questi stati possono essere finali, a meno che non arrivi una richiesta di cancellazione della prenotazione da parte dell'utente. In tal caso la prenotazione viene eliminata dal sistema e transita nello stato *Cancellata*. Una prenotazione nello stato *In coda* può transitare nello stato *Accettata* nel caso in cui si liberi un posto a lezione.



Lezione

L'istanza di Lezione inizia la sua vita nello stato *Prenotabile* (la lezione è disponibile per la prenotazione e tutti i posti sono vuoti), e ci rimane fino a quando non si esauriscono i posti disponibili, transitando nello stato *Piena*. Se si liberano abbastanza posti ritornerà *Prenotabile*. Terminata la finestra di prenotazione la lezione passa nello stato *Chiusa* (non ci si può più prenotare). Una lezione può essere *Annnullata* prima di iniziare per vari motivi logistici, e in tal caso si invalidano tutte le prenotazioni per quella lezione. Quando la lezione termina si transita nello stato *Svolta*.



Aula

L'istanza di Aula inizia la sua vita nello stato *Agibile* (è un'aula libera all'interno del sistema GOMP, assegnabile ad una lezione), e può transitare in *Assegnata* nel caso in cui una determinata Lezione verrà svolta in quell'aula. Dagli stati *Agibile* e *Assegnata* l'aula può transitare in *Inagibile* a causa di guasti alla strumentazione necessaria per la lezione, oppure per problemi legati alla sicurezza infrastrutturale. Nello stato *Inagibile* la capienza viene simbolicamente settata a 0. Quando inizia la lezione assegnata a quell'aula essa diventa *Occupata*, e quando finisce ritornerà *Agibile*, in attesa di essere assegnata ad un'altra lezione.

