***UNIBite – Requisiti***

Università degli studi di Bergamo  
Ingegneria Informatica

Beccarelli Raissa Matr. 1086785  
Locatelli Giacomo Matr. 1086262  
Valceschini Marco Matr. 1086356

***Immagine che contiene testo, Carattere, Elementi grafici, simbolo

Descrizione generata automaticamente***

**1 Introduzione**

Il seguente documento ha lo scopo di definire quali sono i requisiti del progetto UNIBite. All’interno di questo saranno definiti i modi in cui i requisiti sono elicitati, definendo quali siano strettamente necessari da implementare in base alla loro priorità, grazie all’utilizzo della tecnica del MoSCoW. Il documento permetterà di avere una visione più chiara del progetto che si vuole sviluppare e delle sue funzionalità.

**2 Software Quality**

Per quanto riguarda i fattori di qualità del sistema si è deciso di utilizzare, come anche descritto nel Project Plan, alcuni fattori e criteri di qualità stabiliti nel modello di McCall e dall’ISO 9126. Per quanto riguarda il modello di McCall verranno utilizzati solamente alcuni fattori di qualità stabiliti nei settori di funzionamento del prodotto e revisione dello stesso. I fattori che sono stati scelti sono quindi:

* Correttezza: si vuole che il sistema soddisfi i requisiti descritti e permetta di raggiungere gli obiettivi preposti;
* Affidabilità: si vuole che il sistema sia in grado di eseguire la propria funzione in modo accurato e senza intoppi, garantendo che esso sia in grado di tollerare i guasti;
* Integrità: misura per cui il sistema deve mantenere i dati personali protetti, senza permettere che persone non autorizzate possano accedervi e modificarne il contenuto;
* Usabilità: si necessita che il sistema sia di facile comprensione per tutti gli utenti e che sia quindi facilmente e intuitivamente utilizzabile;
* Manutenibilità: lo sforzo per cui si garantisce che il programma possa essere modificato dopo la consegna dello stesso, nel caso in cui siano rivelati problemi o bug. Inoltre, si vuole che il programma sia facilmente leggibile dai programmatori, per rendere più facile la futura manutenibilità del sistema.

**3 Requirement engineering**

**3.1 Elicitazione dei requisiti**

In questa sezione ci occuperemo di definire i modi in cui saranno estratti i requisiti relativi al problema che il progetto si occupa di risolvere.  
I metodi con cui i requisiti sono elicitati sono:

* Analisi dello scenario

Si immagina una situazione tipo:

* Lato Utente 🡪 si suppone che un utente voglia prenotare un pasto all’interno di una mensa universitaria. Se l’utente fa parte dell’ambiente universitario (studente o docente) può fare l’accesso con le sue credenziali, altrimenti se non facente parte accede al sistema senza il bisogno di inserire dati. Una volta entrato nel sistema, l’utente può iniziare la prenotazione. Essa consiste nello scegliere tra primi, secondi, contorni, dolce e frutta. Una volta che l’utente ha inserito nel suo carrello virtuale i piatti selezionati, può passare al pagamento: decidere se pagare alla cassa (scelta obbligatoria per gli utenti esterni) oppure usufruire del credito virtuale universitario associato al suo conto.
* Lato Personale 🡪 il personale può accedere ad un’area riservata tramite credenziali prestabilite. Da questa potrà gestire il servizio inserendo i piatti del giorno e avendo la possibilità di aggiungerne, aggiornando il numero di porzioni disponibili.
* Derivazione da un sistema esistente: si prende spunto dall’idea dei totem all’interno dei fast food, utili per velocizzare il processo di ordinazione, eliminando lunghe code di attesa alle casse. Allo stesso modo prendiamo spunto anche per il metodo di pagamento, in quanto l’utente potrà decidere se pagare alla cassa oppure svolgere l’operazione di pagamento direttamente al totem.
* Prototipazione: per lo sviluppo del software si utilizzerà la prototipazione, che ci permetterà di avere man mano una versione sempre più aggiornata e con più funzionalità. Tramite questo inoltre avremo in qualsiasi fase dello sviluppo un prodotto funzionante, al quale potremo poi aggiungere migliorie.

**3.2 Requisiti**

In questa sezione ci occuperemo di definire i requisiti relativi al problema che il progetto si occupa di risolvere.I requisiti necessari per l’implementazione del progetto sono:

1. Implementazione a scelta dell’accesso da parte di studente/docente, da parte di esterni o da parte del personale.
2. Fase di login
3. Per studenti/docenti tramite nome utente e password scelti durante la fase di registrazione;
4. Per il personale tramite nome utente e password preassegnati in fase di configurazione del sistema, che porterà all’accesso nell’area riservata;
5. Per gli esterni non viene effettuato il login, ma si passa subito alla fase di ordinazione pasto.
6. Fase di registrazione (solo per studenti/docenti non ancora registrati):
7. Tramite inserimento di nome, cognome, matricola, password, e-mail universitaria e fascia Isee (a scelta multipla tra A, B, C, D, E, F, G);
8. Studenti/docenti hanno la possibilità di indicare se sono in possesso di borsa di studio;
9. Il sistema deve anche controllare i dati inseriti dagli utenti, eliminando gli account non corretti;
10. I dati forniti sono inseriti in un embedded database;
11. Il database deve tenere i dati integri e consistenti;
12. Studenti/docenti possono modificare le loro credenziali.
13. Fase di ordinazione pasto:
14. L’utente (studente/docente ed esterno) ha la possibilità di selezionare tra tutte le pietanze disponibili a schermo;
15. Le pietanze sono divise in: primi, secondi, contorni, dolci e frutta e bibite extra;
16. Per ogni pietanza è disponibile la lista di ingredienti e allergeni;
17. Per ogni primo è possibile l’aggiunta di formaggio.
18. Fase di controllo ordine
19. Una volta selezionate le pietanze l’utente può controllare i piatti selezionati all’interno del carrello virtuale;
20. Tramite il carrello può decidere di eliminare, modificare le quantità o andare al pagamento.
21. Fase pagamento
22. Studenti/docenti ed esterni potranno scegliere il pagamento in cassa;
23. Studenti/docenti potranno scegliere il pagamento tramite un credito virtuale;
24. Il credito virtuale è associato al proprio account e viene incrementato ad ogni ordine effettuato;
25. Il credito virtuale potrà essere saldato tramite pagamento in contanti o tramite bonifico;
26. Studenti/docenti hanno diritto allo sconto base rispetto all’esterno che paga il prezzo completo;
27. Studenti/docenti hanno diritto allo sconto anche in base alla fascia Isee indicata in fase di registrazione: 80% fascia A, 60% fascia B e 30% fascia C;
28. Una volta scelto il pagamento all’utente è assegnato uno scontrino con il resoconto dell’ordine.
29. Accesso del personale all’area riservata
30. Il personale deve poter modificare il menù visibile nella schermata utente, aggiungendo ed eliminando le pietanze, modificandone inoltre le quantità;
31. Il personale deve avere la possibilità di aggiungere nuovi piatti al database, tramite nome del piatto, ingredienti, allergeni e una foto esemplificativa;
32. Il personale può avere una visione generale delle quantità di ingredienti disponibili in magazzino e le può aggiornare.

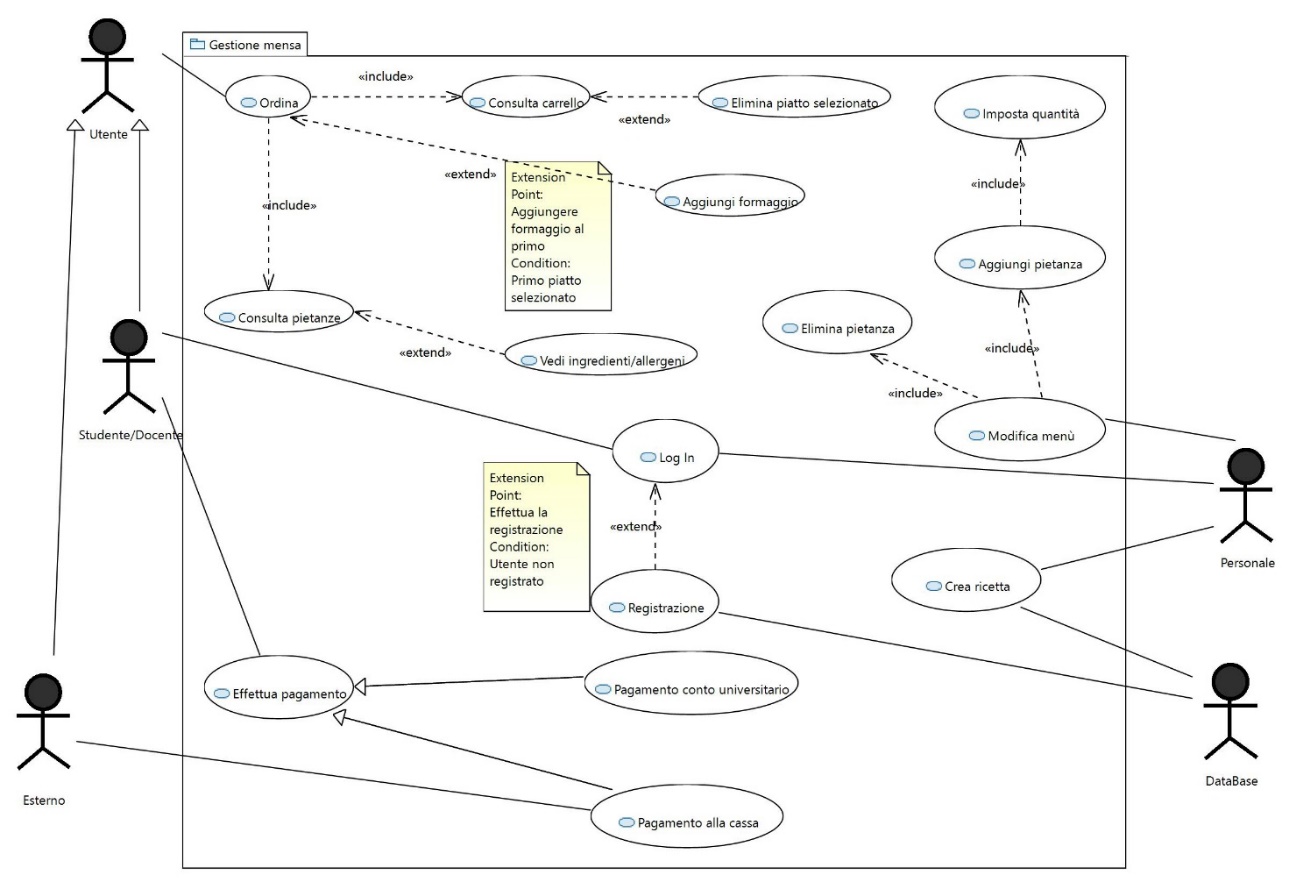
**3.3 MoSCoW**

I requisiti elencati nel punto precedente sono suddivisi in base alle loro priorità di implementazione così come segue:

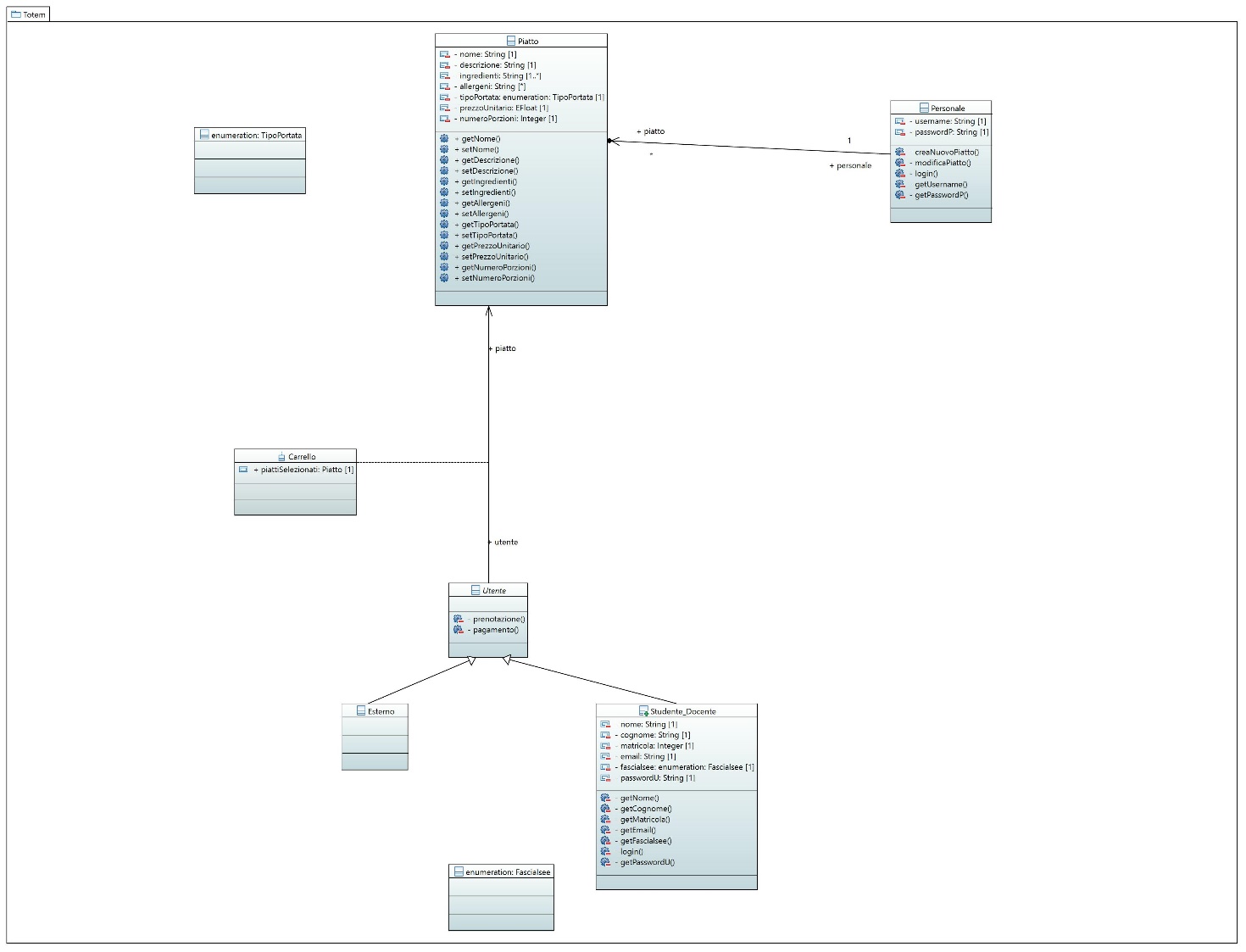
|  |  |  |  |
| --- | --- | --- | --- |
| **MUST HAVE** | **SHOULD HAVE** | **COULD HAVE** | **WON’T HAVE** |
| 1 |  |  |  |
| 2.a, 2.b, 2.c |  |  |  |
| 3.a, 3.d | 3.b, 3.e | 3.c | 3.f |
| 4.a, 4.b | 4.c | 4.d |  |
|  | 5.a, 5.b |  |  |
| 6.a | 6.b, 6.c, 6.e, 6.g | 6.f | 6.d |
| 7.a | 7.b |  | 7.c |

**4 UML**

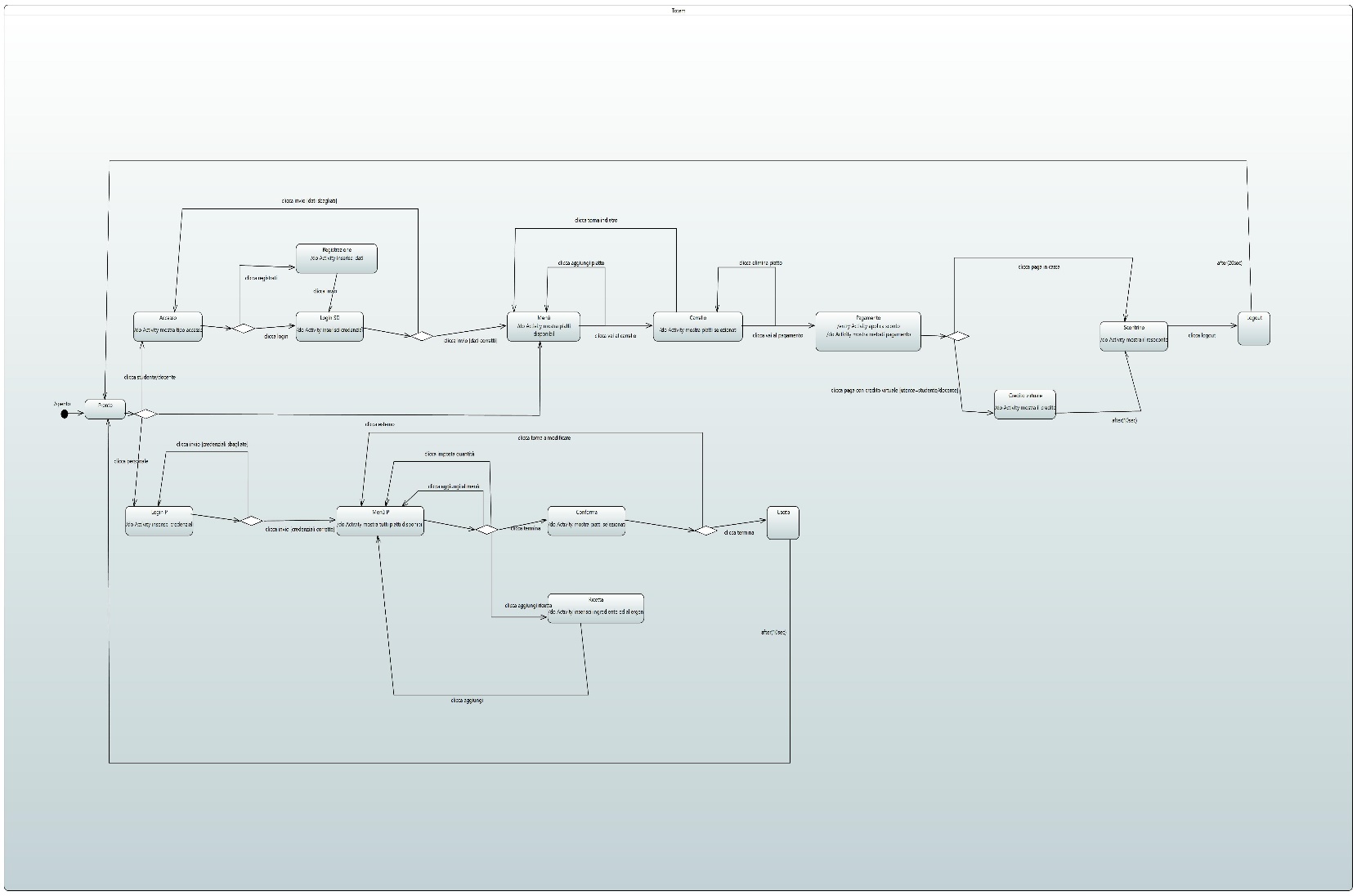
**4.1 Use Case Diagram**

****

**4.2 Class Diagram**

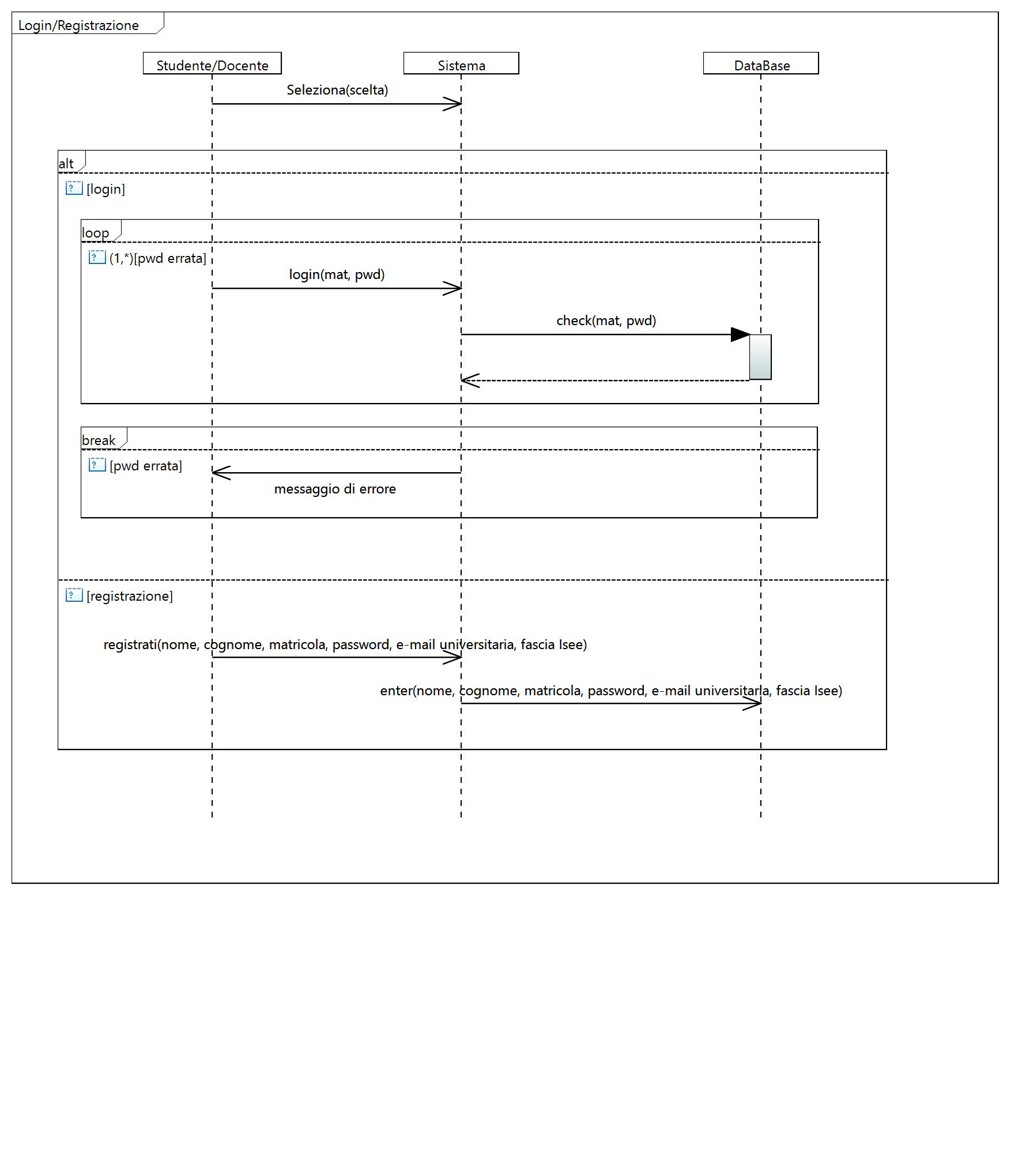
****

**4.3 State Machine Diagram**

****

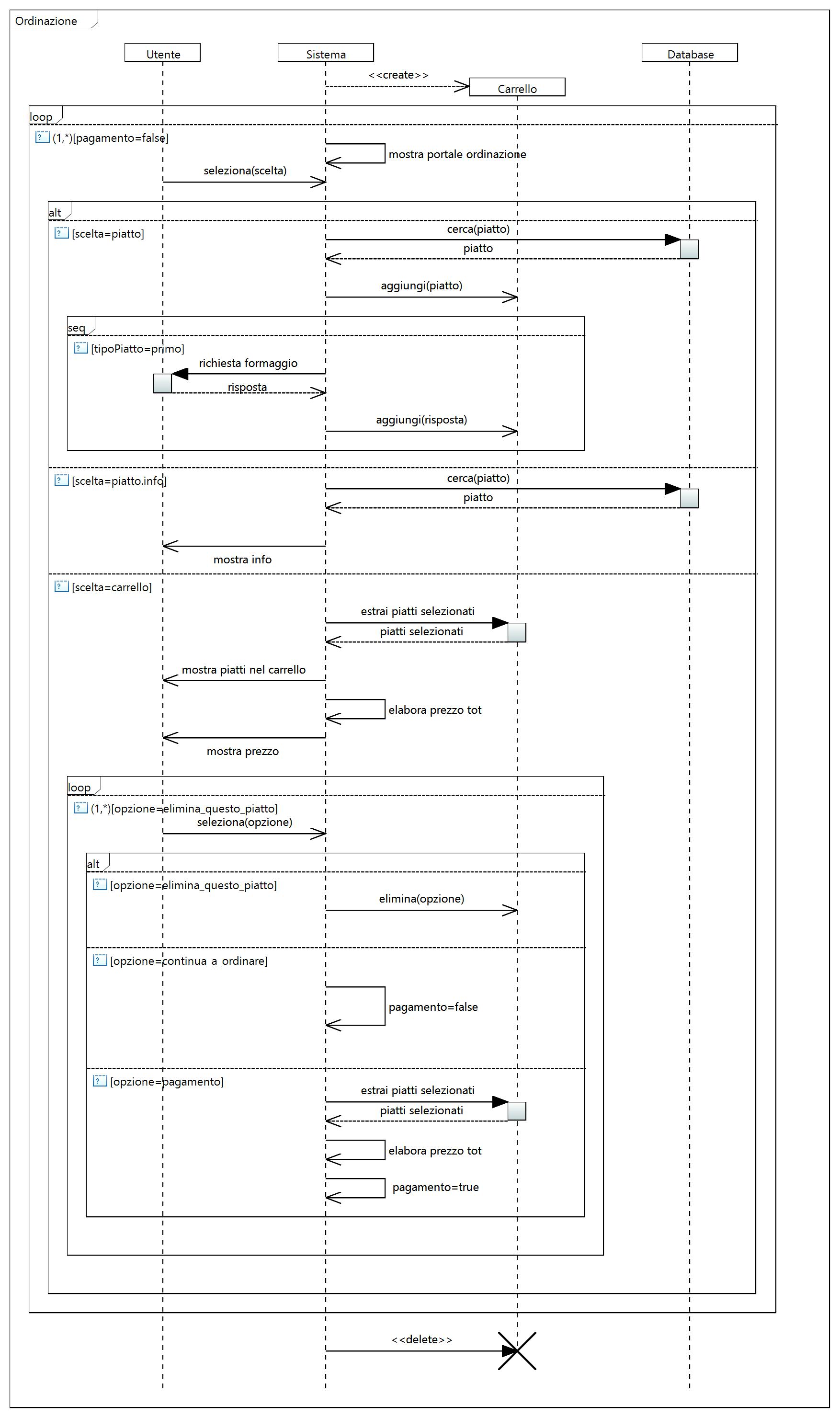
**4.4 Sequence Diagram**

**4.4.1 Login**

****

**4.4.2 Pagamento**

**4.4.3 Ordinazione**

****

**4.5 Communication Diagram**

**4.6 Activity Diagram**

**4.7 Component Diagram**