



UNIVERSITY
OF TRENTO

Dipartimento di Ingegneria e Scienza dell'Informazione

CORSO DI INGEGNERIA DEL SOFTWARE



Documento di progetto: Specifica dei requisiti

Gruppo: T33

Indice

1. Scopo del documento.....	3
2. Requisiti Funzionali.....	3
2.1 Utente anonimo.....	3
2.1.1 UML generale utente anonimo.....	6
2.2 Utente studente.....	7
2.2.1 UML generale utente studente.....	18
2.3 Utente offerente	19
2.3.1 UML generale utente offerente	30
3. Requisiti Non Funzionali.....	31
4. Analisi del Contesto.....	36
4.1 Utenti e sistemi esterni.....	36
4.2 Diagramma di contesto.....	37
5. Analisi dei Componenti.....	39
5.1 Definizione dei componenti	39
5.2 Diagramma dei componenti	43
6. Note ed eventuali	51

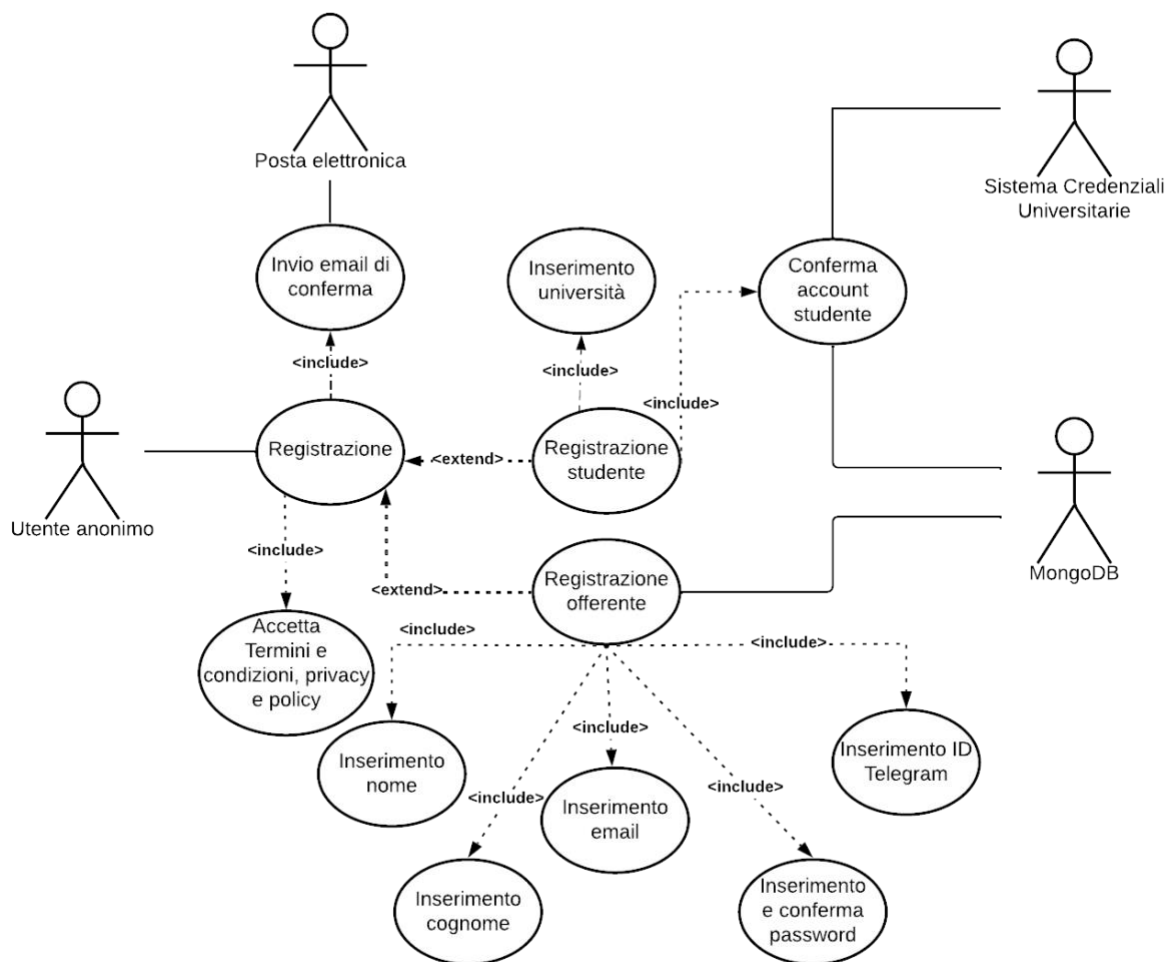
1. Scopo del documento

Il documento propone la specifica dei requisiti funzionali e non funzionali del progetto *StayBusy* attraverso l'utilizzo di diagrammi Unified Modelling Language (UML) e tabelle strutturate, accompagnati da una descrizione degli stessi in linguaggio naturale. Nello specifico, i primi sono stati utilizzati per la descrizione dei requisiti funzionali mentre le tabelle strutturate per la rappresentazione dei requisiti non funzionali. Inoltre, tenendo conto di tali requisiti, viene presentato il design del sistema e dei componenti tramite l'utilizzo di diagrammi di contesto.

2. Requisiti Funzionali

2.1 Utente anonimo

RF1 Registrazione, RF3 Invio e-mail di conferma



Titolo: Registrazione utente

Riassunto: questo use case descrive come avviene la registrazione dell'utente all'interno del nostro sistema.

Descrizione:

- L'utente seleziona la tipologia di account che vuole creare. In particolare, può scegliere tra l'opzione *Studente* e quella *Offerente*.
 - Il sistema mostra all'utente un form:
 - o (scelta studente) l'utente dovrà inserire il nome dell'ateneo Università che frequenta [exception 1] e accettare obbligatoriamente i “*Termini e condizioni, privacy e policy*”. Cliccando su “*Registrati*”, l'utente verrà indirizzato sulla pagina di login della propria università, dove gli sarà richiesto di inserire le credenziali istituzionali. Alla fine di tale operazione, il sistema otterrà automaticamente tutte le informazioni sullo studente tramite l'account istituzionale.
 - o (scelta offerente) l'utente dovrà inserire le seguenti informazioni:
 - nome
 - cognome
 - e-mail [exception 2]
 - password [exception 3]
 - password di conferma [exception 4]
- Completata correttamente tale operazione, l'utente dovrà obbligatoriamente accettare i “*Termini e condizioni, privacy e policy*” e cliccare su “*Registrati*”.
- Una volta completato tutto il procedimento di registrazione, sia per l'account studente che per l'account offerente, arriverà una e-mail di conferma.
 - La gestione dei dati e la verifica della correttezza delle credenziali di registrazione sono affidate a *MongoDB*.

Exceptions:

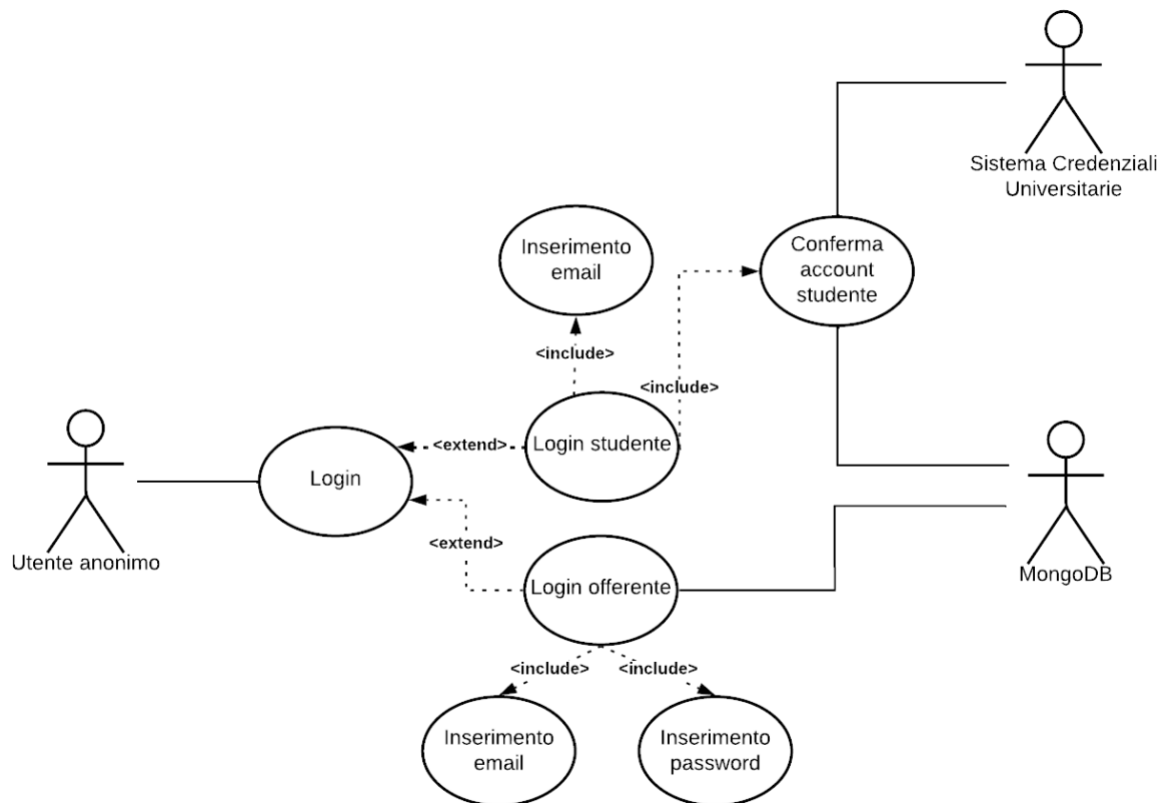
[exception 1] se l'ateneo universitario inserito non è presente nel sistema, l'utente non può registrarsi come studente.

[exception 2] se non rispetta lo standard delle e-mail, il sistema darà errore.

[exception 3] se non rispetta lo standard di “Strong Password”, il sistema darà errore.

[exception 4] se non coincidono Password e Conferma password, il sistema darà errore.

RF2 Login nel sistema



Titolo: Login utente

Riassunto: questo use case describe come avviene il login da parte dell'utente all'interno del nostro sistema.

Descrizione:

- È richiesto di selezionare il tipo di utenza con cui si vuole fare accesso nel sito, (Studente/Offerente).
- Il sistema mostra all'utente un form differente in base alla scelta effettuata:
 - o (opzione studente) l'utente deve inserire l'e-mail istituzionale [exception 1]. Inoltre, cliccando su "Accedi", l'utente viene indirizzato sulla pagina di login dell'università di riferimento. Completato il login, l'utente può utilizzare il sistema.
 - o (opzione offerente) l'utente deve inserire la propria e-mail personale [exception 2], fornita al momento della registrazione, e la password di riconoscimento. Cliccando su "Accedi" [exception 3] l'utente può utilizzare il sistema.
- La gestione dei dati dell'utente offerente e la verifica delle sue credenziali di accesso sono affidate a *MongoDB*.
- La gestione dei dati personali dell'utente offerente e la verifica delle sue credenziali di accesso sono affidate al *Sistema Credenziali Universitarie*.

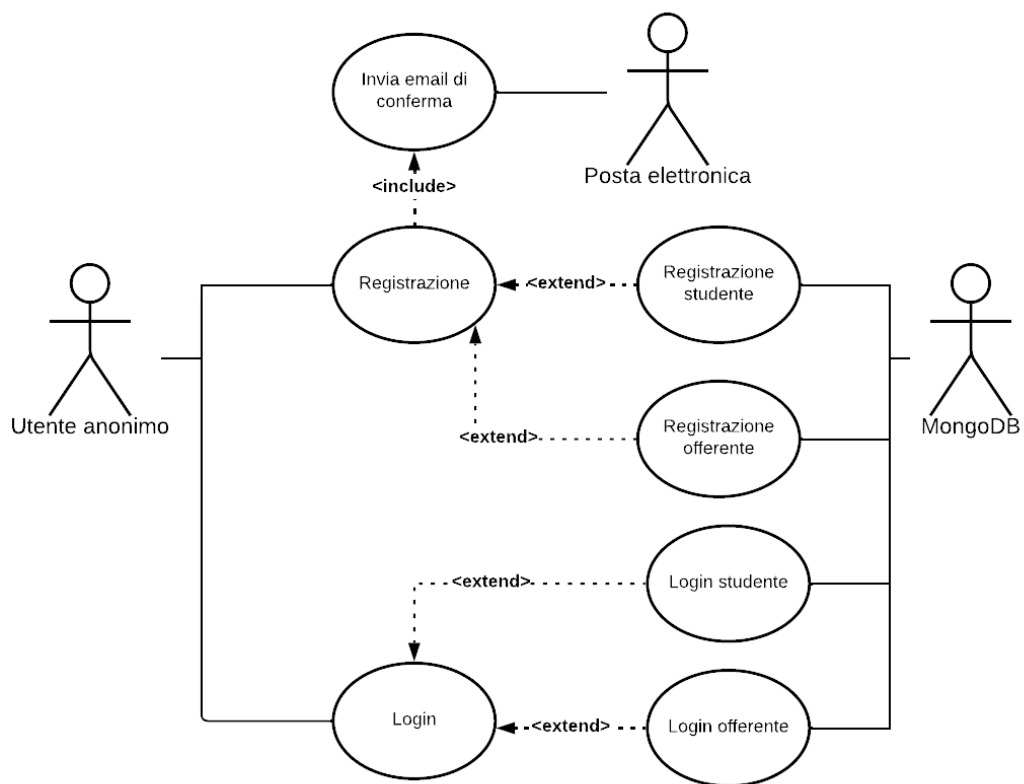
Exceptions:

[exception 1] se l'utente inserisce un'e-mail non corretta, il sistema darà errore.

[exception 2] se l'utente inserisce un'e-mail non corretta, il sistema darà errore.

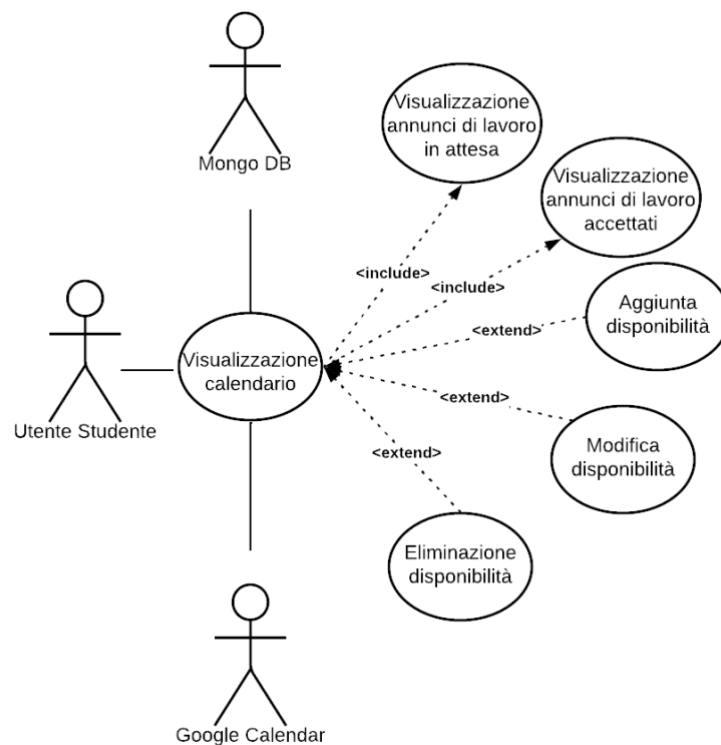
[exception 3] se le credenziali inserite non coincidono con le credenziali di registrazione, il sistema darà errore.

2.1.1 UML generale utente anonimo



2.2 Utente studente

RF4 Calendario



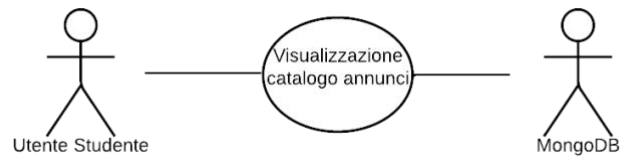
Titolo: Visualizzazione calendario

Riassunto: questo use case descrive come avviene l'interazione tra studente e calendario.

Descrizione:

- Dopo aver effettuato il primo accesso, lo studente deve inserire le proprie disponibilità temporali ovvero il/i giorno/i e l'orario di disponibilità.
- Il sistema permette allo studente di modificare le proprie disponibilità in qualsiasi momento.
- Lo studente può visualizzare un calendario che mostra:
 - gli orari di disponibilità dello studente;
 - gli orari in cui lo studente si è candidato per un annuncio;
 - gli orari in cui lo studente è occupato, ovvero in cui stato accettato per lo svolgimento di un servizio.
- Il sistema permette allo studente di cambiare le disponibilità temporali fornite, modificando giorno e/o orario.
- Il sistema permette allo studente di eliminare le disponibilità temporali selezionando l'apposita funzionalità "Elimina disponibilità".
- Le funzionalità per l'inserimento delle disponibilità dello studente sono fornite da *Google Calendar*.
- La gestione delle disponibilità temporali inserite dallo studente è affidata a *MongoDB*.

RF5 Catalogo



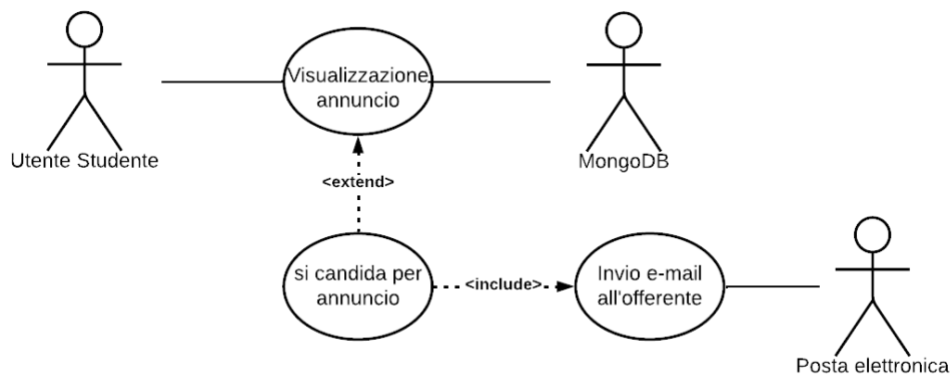
Titolo: Visualizzazione catalogo annunci

Riassunto: questo use case descrive la visualizzazione del catalogo annunci.

Descrizione:

- Il sistema fornisce allo studente un catalogo annunci in cui sono evidenziati:
 - tipologia di lavoro
 - nome offerente
 - la prima parte della descrizione del lavoro offerto
- Lo studente può cliccare su un annuncio specifico per visualizzarne ulteriori informazioni.
- Lo studente può visualizzare l'account dell'offerente (vedi [RF29](#)) cliccando la foto profilo.
- *MongoDB* è il database che si occupa di gestire e fornire i dati relativi al catalogo degli annunci dell'offerente.

RF6 Lista di attesa, RF21 Contatto con lo studente



Titolo: Visualizzazione annuncio

Riassunto: questo use case descrive cosa avviene quando viene visualizzato un singolo annuncio.

Descrizione:

- Lo studente può visualizzare in maniera più chiara tutte le informazioni riguardanti l'annuncio (specifiche annuncio), in particolare:
 - Tipo di lavoro

- Nome offerente
 - Contatti offerente
 - Descrizione del lavoro
 - Modalità di pagamento
- Lo studente può inviare la sua candidatura all'annuncio cliccando su “*Sono interessato*”. In questo modo entrerà a far parte della lista di attesa specifica di quell'annuncio visibile dall'offerente (vedi [Lista candidati](#)). Inoltre, all'offerente sarà automaticamente inviata un'e-mail di avviso in cui sarà segnalata la candidatura dello studente all'annuncio.
 - Lo studente può visualizzare l'account dell'offerente cliccando sulla sua foto profilo.
 - Lo studente può salvare l'annuncio tra i “*Preferiti*” per una visualizzazione futura.
 - *MongoDB* è il database che si occupa di gestire e fornire i dati relativi ad uno specifico annuncio.

RF7 Stato di richiesta degli annunci



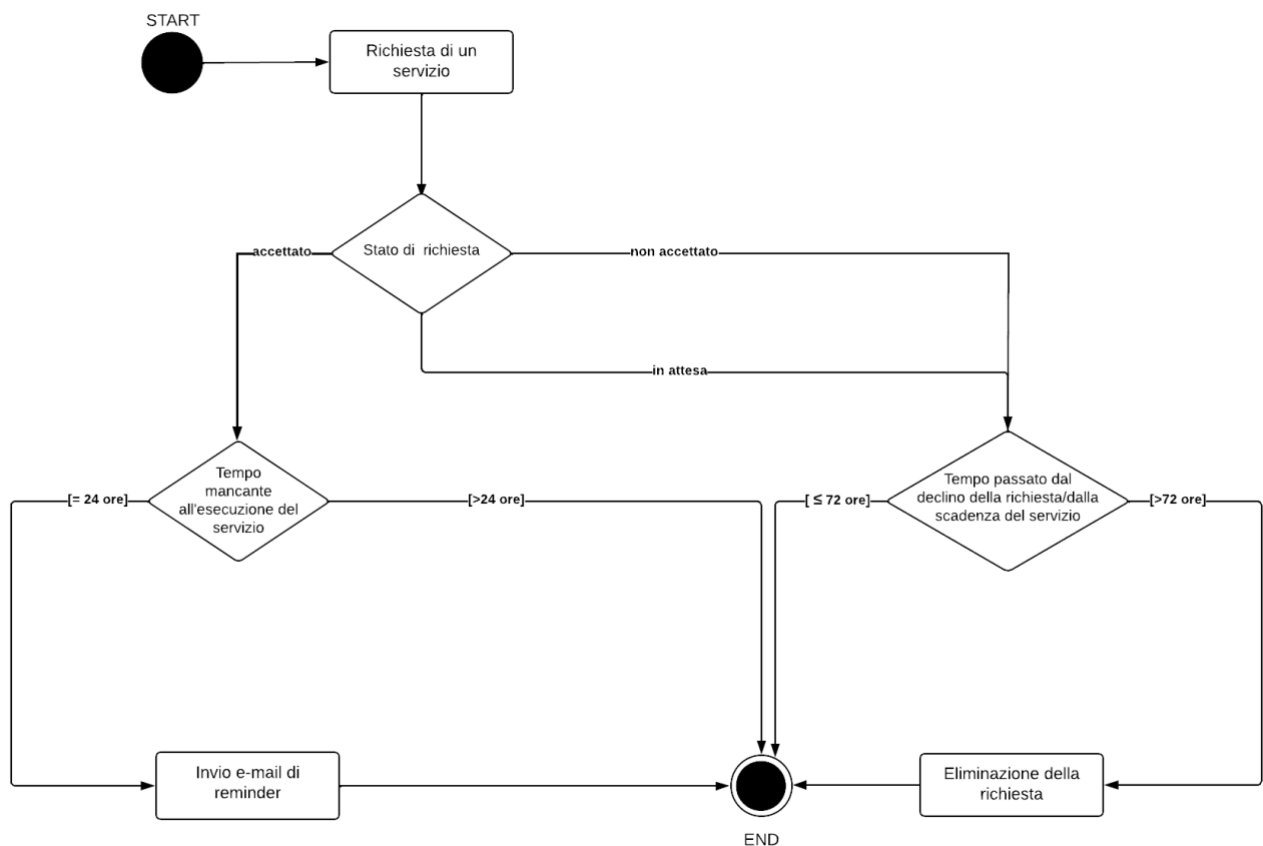
Titolo: Visualizzazione stato annunci

Riassunto: questo use case descrive la visualizzazione dello stato degli annunci per cui lo studente si è candidato.

Descrizione:

- Lo studente visualizza tutti gli annunci per cui si è precedentemente offerto.
- Il sistema mostra allo studente lo stato delle richieste degli annunci (*accettato/rifiutato/in attesa*).
- Cliccando su uno specifico annuncio, lo studente può visualizzarne ulteriori informazioni.
- Lo studente può visualizzare l'account dell'offerente (vedi [RF29](#)) cliccando la foto profilo.
- *MongoDB* è il database che si occupa di gestire e fornire i dati relativi allo stato degli annunci a cui lo studente si è candidato.

RF8 Invio e-mail di reminder



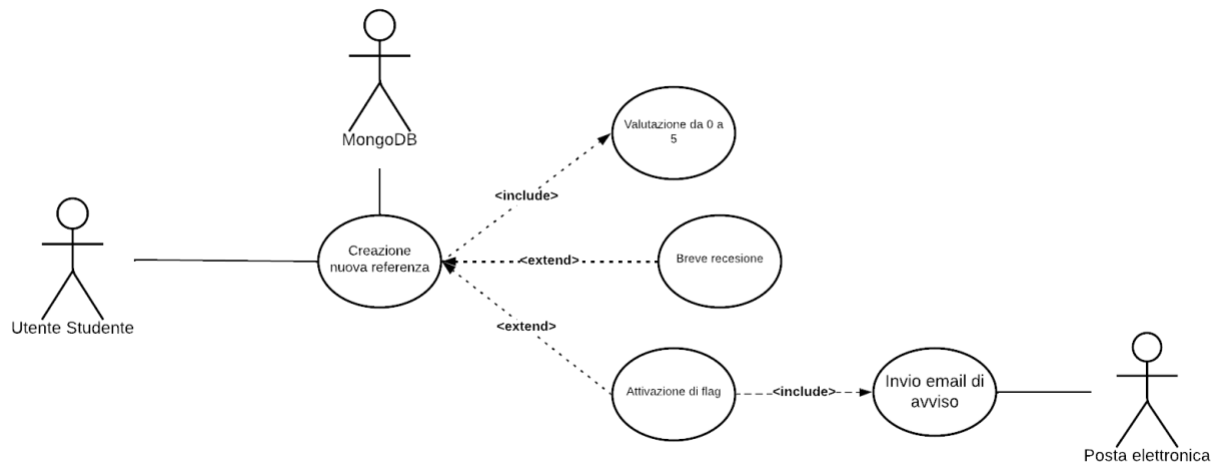
Titolo: Invio e-mail di reminder

Riassunto: questo state diagram descrive la gestione dell'invio di e-mail di reminder allo studente che è stato accettato per un lavoro.

Descrizione:

- Se lo studente si offre per un servizio e la sua richiesta viene rifiutata, questa viene eliminata dalla sezione dedicata alla raccolta dello stato degli annunci dell'utente studente dopo 72 ore dal momento in cui l'offerente l'ha declinata.
- Se lo studente si offre per un servizio e la sua richiesta viene accettata, il giorno antecedente all'inizio del lavoro, riceve un'e-mail in cui gli viene ricordato lo svolgimento del servizio che ha concordato con l'offerente.
- Se lo studente si offre per un servizio e la sua richiesta rimane in attesa, ovvero nell'eventualità in cui la data in cui è richiesto il servizio sia passata, la richiesta è eliminata nello stesso modo della richiesta rifiutata.

RF9 Creazione referenza



Titolo: Creazione referenza

Riassunto: questo use case descrive cosa avviene quando uno studente crea una referenza per l'offerente.

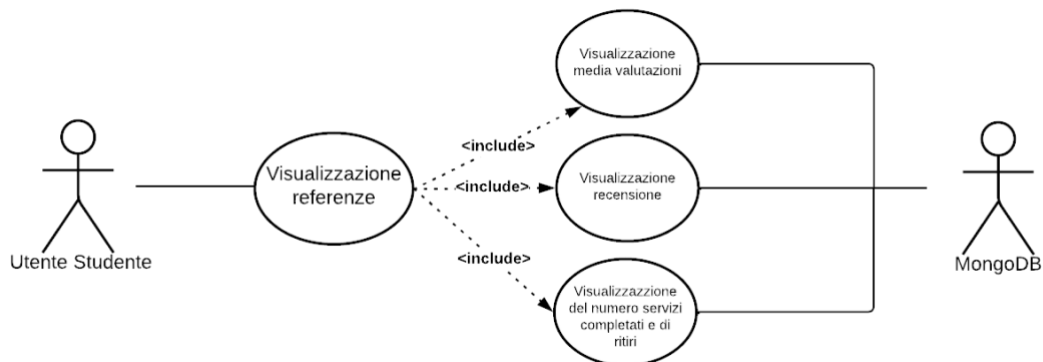
Descrizione:

- Completato un servizio, lo studente deve lasciare una valutazione da 0 a 5 all'offerente.
- Lo studente può scrivere una breve recensione all'offerente.
- Lo studente può attivare un flag di notifica grazie al quale riceverà, in futuro, una notifica e-mail per ogni annuncio – il cui orario è contenuto nelle disponibilità specificate dallo studente – creato da tale offerente [exception 1].
- La gestione dell'invio notifiche è affidata al servizio di *Posta elettronica*.
- La gestione e il salvataggio dei dati relativi alle nuove referenze create dall'offerente per valutare un servizio svolto dallo studente è affidata a *MongoDB*.

Exceptions:

[exception 1] se lo studente non è disponibile in quella fascia oraria non riceverà nessuna notifica.

RF10 Visualizzazione referenze



Titolo: Visualizzazione referenze personali

Riassunto: questo use case descrive come avviene la visualizzazione delle referenze personali da parte dello studente.

Descrizione:

- Il sistema mostra allo studente la media delle valutazioni che ha ricevuto, sia in maniera grafica – tramite delle stelline – sia in maniera numerica – tramite la media effettiva.
- Il sistema mostra allo studente la lista di recensioni inserite dagli offerenti [exception 1].
- Lo studente ha la possibilità di visualizzare il numero di servizi completati e il suo numero di ritiri.
- La gestione dei dati relativi referenze personali dello studente – in particolare la media delle valutazioni, la recensione e il numero di servizi completati e ritiri - è affidata a *MongoDB*.

Exceptions:

[exception 1] se lo studente non ha recensioni, queste non vengono visualizzate.

RF11 Cronologia lavorativa



Titolo: Visualizzazione cronologia dei servizi effettuati

Riassunto: questo use case descrive come avviene la visualizzazione della cronologia dei servizi effettuati dallo studente.

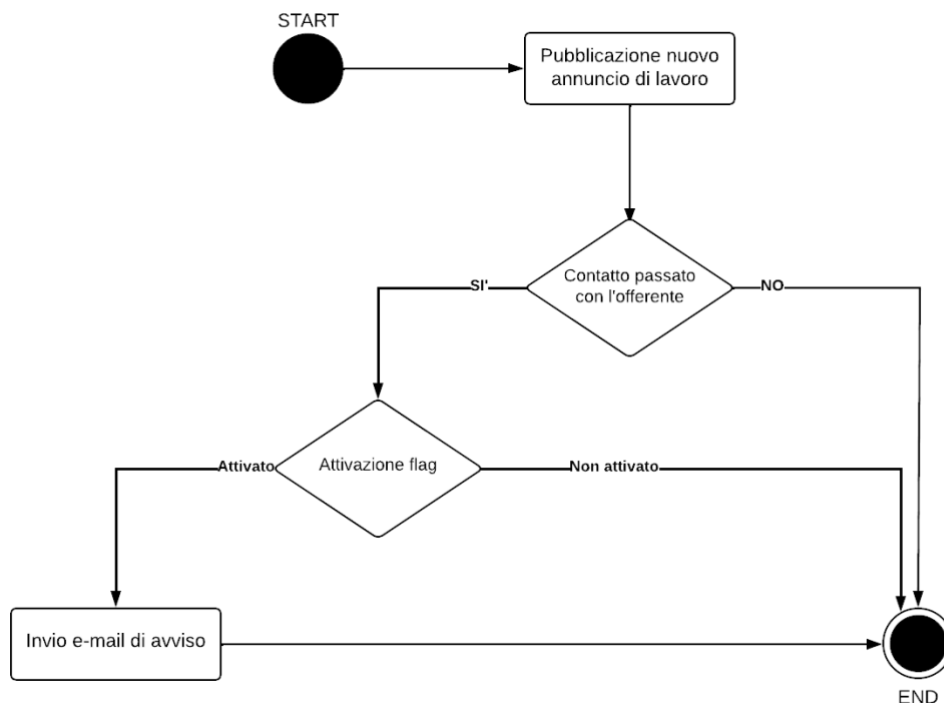
Descrizione:

- Lo studente visualizza la lista dei servizi effettuati in passato in ordine cronologico.
- Lo studente può visualizzare la recensione che l'offerente ha creato [exception 1].
- Lo studente può visualizzare l'account dell'offerente.
- La gestione dei dati della cronologia lavorativa personale dello studente è affidata a *MongoDB*.

Exceptions:

[exception 1] se l'offerente non ha creato la recensione, questa non viene visualizzata.

RF12 Avviso di richiesta



Titolo: Ricezione notifica di avviso richiesta

Riassunto: questo state-diagram descrive la gestione dell'invio della notifica e-mail per la creazione nuovi annunci.

Descrizione:

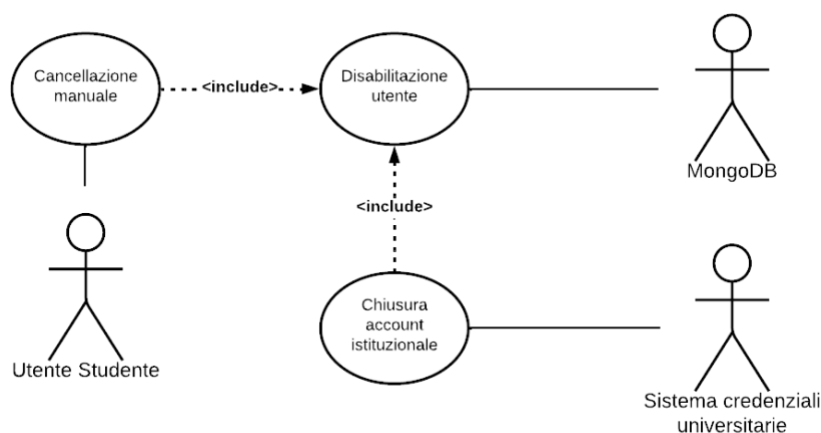
- L'utente che in passato ha prestato servizio ad un offerente può decidere di attivare il flag di richiesta per gli annunci creati da quest'ultimo [exception 1] (vedi [creazione referenza](#)).
- Se l'utente ha precedentemente attivato il flag di avviso richiesta, nel momento in cui l'offerente pubblica un nuovo annuncio, lo studente riceve una notifica via e-mail [exception 2].

Exceptions:

[exception 1] se lo studente non ha mai prestato servizio presso un offerente non può richiedere che gli sia notificato l'avviso di richiesta al momento della pubblicazione di un nuovo annuncio da parte di tale offerente.

[exception 2] se lo studente non ha mai attivato il flag di avviso richiesta, non riceverà la notifica e-mail di avviso nuovo annunci dell'offerente.

RF13 Cancellazione manuale dell'account, RF14 Cancellazione automatica dell'account



Titolo: Cancellazione dell'account

Riassunto: questo use case descrive come viene gestita la cancellazione dell'account dell'utente studente.

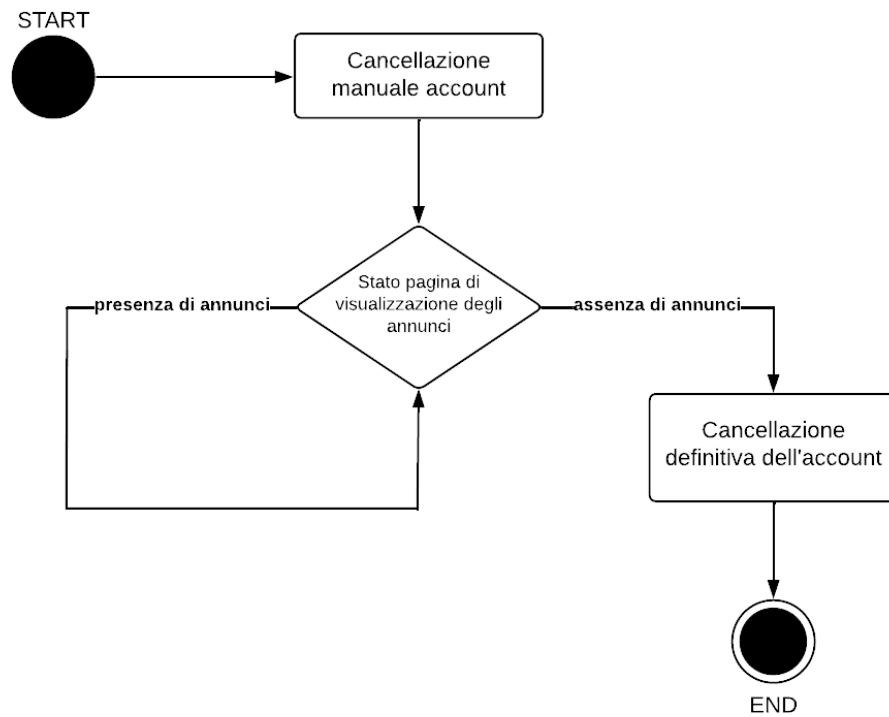
Descrizione:

- Lo studente ha la possibilità di cancellare il proprio account personale manualmente solo nel caso in cui non sia candidato ad alcun servizio, ovvero solo nell'eventualità in cui la pagina di visualizzazione degli annunci sia vuota [exception 1] (vedi [Stato annunci](#), vedi [Cancellazione manuale dell'account – State Diagram](#)).
- In seguito al conseguimento del titolo di studio e alla disattivazione dell'e-mail istituzionale, il sistema effettua automaticamente la cancellazione dell'account dello studente.
- Conseguentemente alla cancellazione del profilo, viene eseguita la disabilitazione dell'account dello studente.
- La gestione della disabilitazione dell'account dello studente è affidata a *MongoDB*. Inoltre, in seguito al conseguimento del titolo di studio, il *Sistema Credenziali Universitarie* provvede alla chiusura dell'account istituzionale mentre *MongoDB* si occupa dell'eliminazione dell'account dello studente.

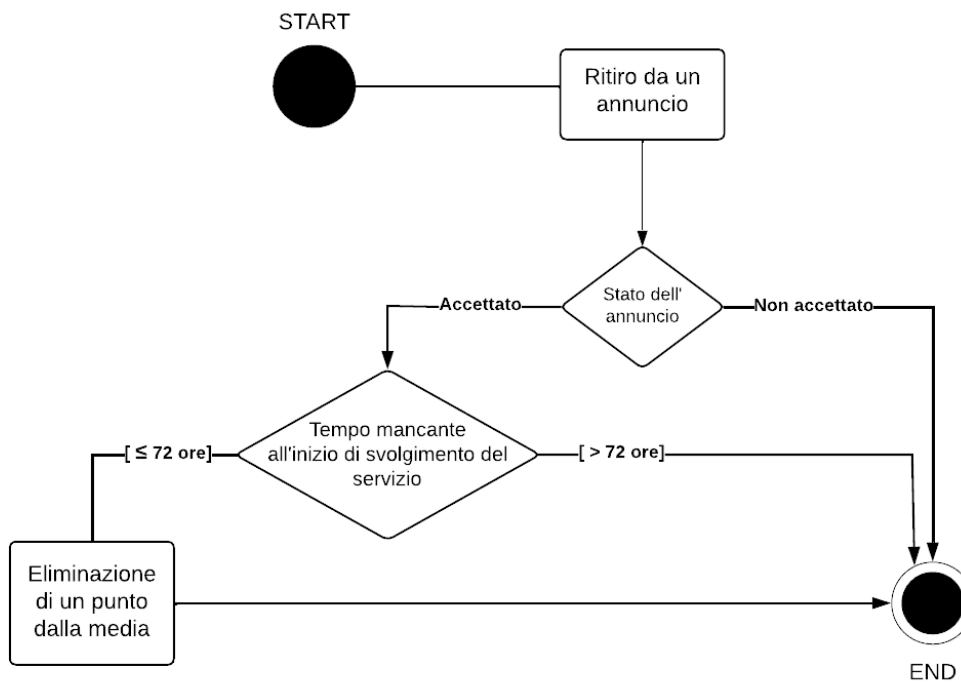
Exceptions:

[exception 1] se nella sezione di visualizzazione degli annunci del profilo dell'utente sono presenti annunci per cui lo studente si è proposto, il sistema non permette l'eliminazione dell'account.

Cancellazione manuale dell'account, State Diagram



RF15 Ritiro da un annuncio



Titolo: Ritiro da un annuncio

Riassunto: questo state-diagram descrive l'operazione di ritiro da un annuncio da parte di uno studente.

Descrizione:

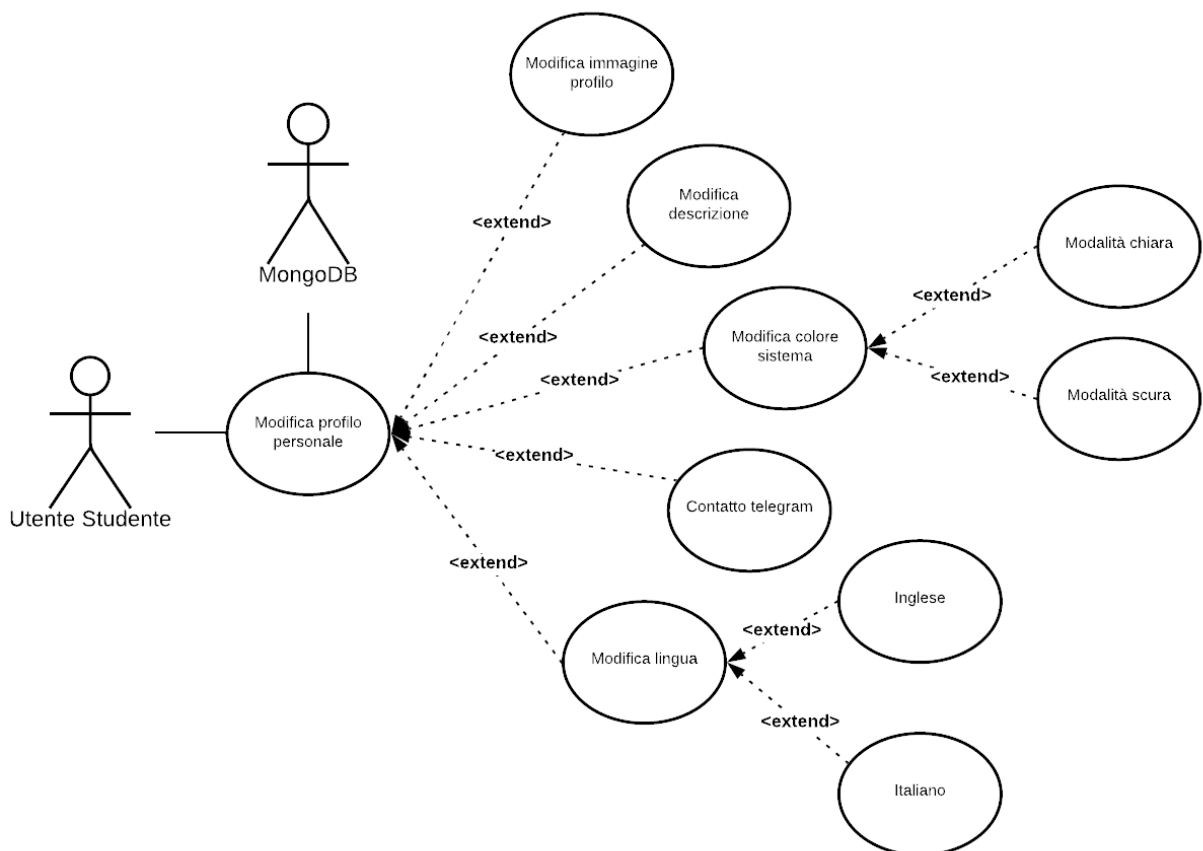
- Nel caso in cui nella sezione di visualizzazione dello stato di richiesta degli annunci (vedi [RF7](#)) siano presenti uno o più annunci a cui lo studente ha fatto domanda che presentano lo stato “*Accettato*” [exception 1], lo studente ha la possibilità di ritirarsi da uno specifico annuncio solo nell'eventualità in cui il tempo mancante dallo svolgimento del servizio richiesto è maggiore di 72 ore [exception 2].

Exceptions:

[exception 1] se l'annuncio presenta stato “*in attesa*”, lo studente può ritirarsi in qualsiasi momento.

[exception 2] se il tempo mancante allo svolgimento del servizio richiesto è minore o uguale a 72 ore, verrà accettata la richiesta di ritiro dello studente ma quest'ultimo perderà un punto dalla propria media personale delle valutazioni.

RF16 Profilo personale



Titolo: Visualizzazione e modifica del profilo personale

Riassunto: questo use case descrive la visualizzazione e la modifica del profilo personale dell'utente studente.

Descrizione: accedendo alla propria area personale, lo studente ha la possibilità di modificare in qualsiasi momento alcune delle informazioni personali visibili nel sito e alcune funzionalità del sistema:

- L'utente può aggiungere/modificare l'immagine profilo.
- L'utente può modificare la descrizione personale inserita obbligatoriamente al momento della registrazione.
- L'utente può modificare il proprio contatto *Telegram* attraverso il quale avverrà la comunicazione con l'offerente.
- L'utente può modificare la lingua di sistema, scegliendo tra italiano (default) e inglese.
- L'utente può modificare la modalità di colore del sistema, scegliendo tra modalità chiara (default) e modalità scura.
- La gestione di tutti i dati personali dello studente è affidata a *MongoDB*, database che si occupa della loro archiviazione e, qualora venissero modificati, del loro aggiornamento.

RF17 Ricerca di annunci



Titolo: Ricerca di un annuncio nel sistema

Riassunto: questo use case descrive la ricerca di annunci da parte dello studente.

Descrizione:

- Lo studente ha la possibilità di effettuare una ricerca di annunci inserendo solo delle parole chiave all'interno della barra di ricerca fornita [exception 1] [extension 1].
- *MongoDB* è il database che gestisce e fornisce tutti i dati riguardanti un annuncio.

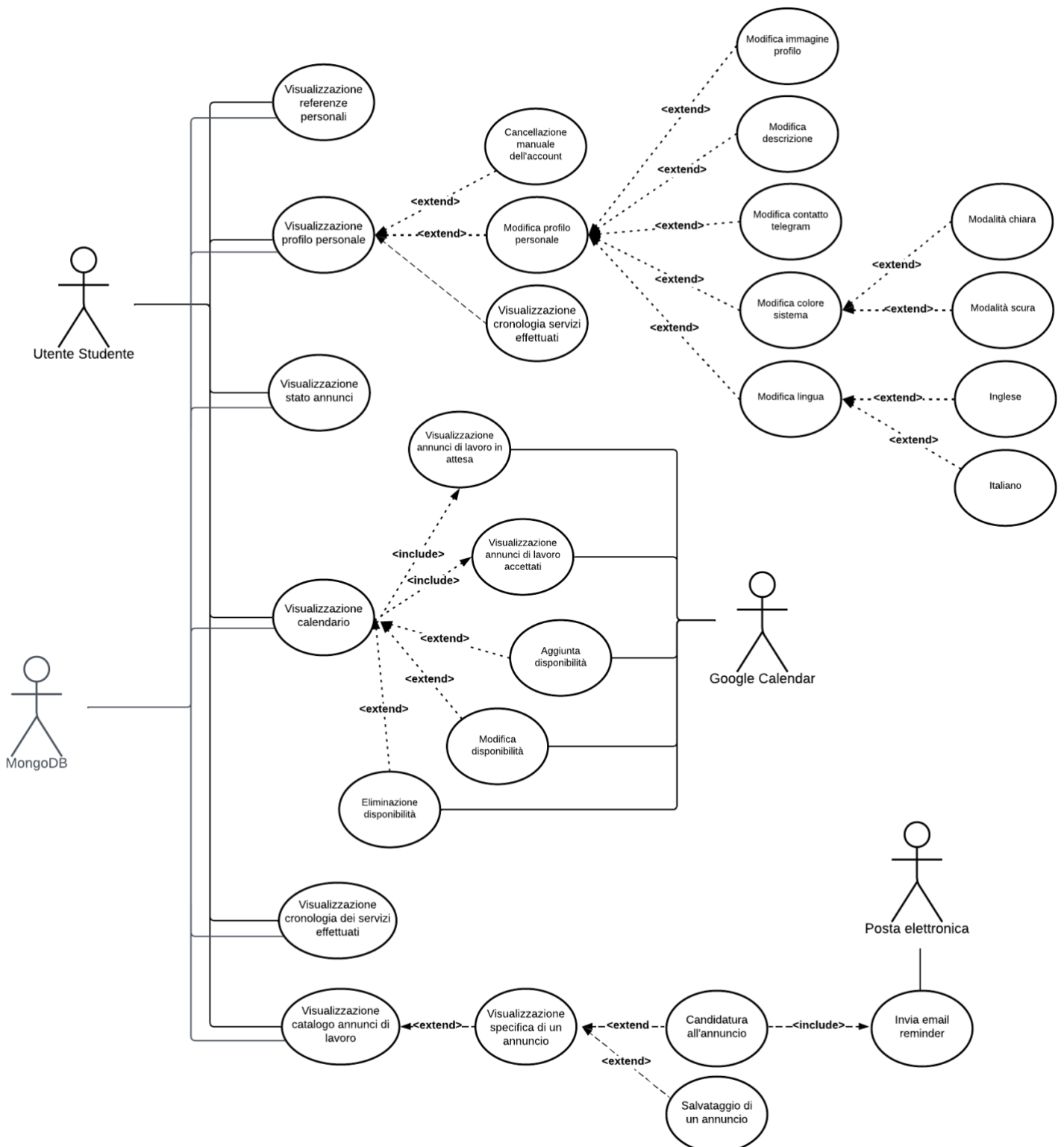
Exceptions:

[exception 1] se le parole chiave inserite non vengono trovate dal sistema, lo studente si ritroverà quale esito della ricerca una pagina bianca con dicitura "*Error 404 – Not found*". Per ritornare alla pagina home del sito, è richiesto allo studente di fare un refresh della pagina.

Extensions:

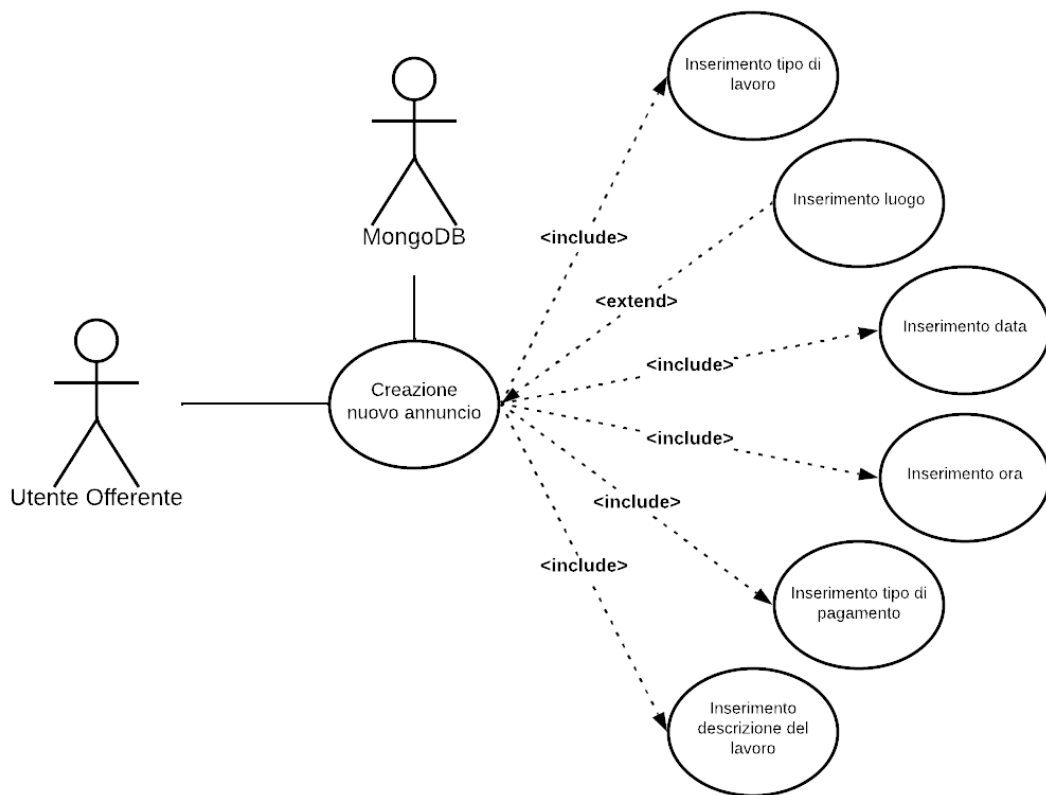
[extension 1] la ricerca potrà essere ancora più performante attraverso l'utilizzo di filtri (luogo, distanza, orari).

2.2.1 UML generale utente studente



2.3 Utente offerente

RF18 Creazione di un annuncio



Titolo: Creazione nuovo annuncio

Riassunto: questo use case descrive la creazione di un nuovo annuncio da parte dell'utente offerente.

Descrizione:

- Al momento della creazione di un nuovo annuncio, l'offerente deve inserire opportunamente le informazioni relative a:
 - tipologia di lavoro
 - data del lavoro
 - fascia oraria del lavoro (ora inizio – ora fine)
 - tipologia di pagamento (contati/PayPal)
 - descrizione del lavoro

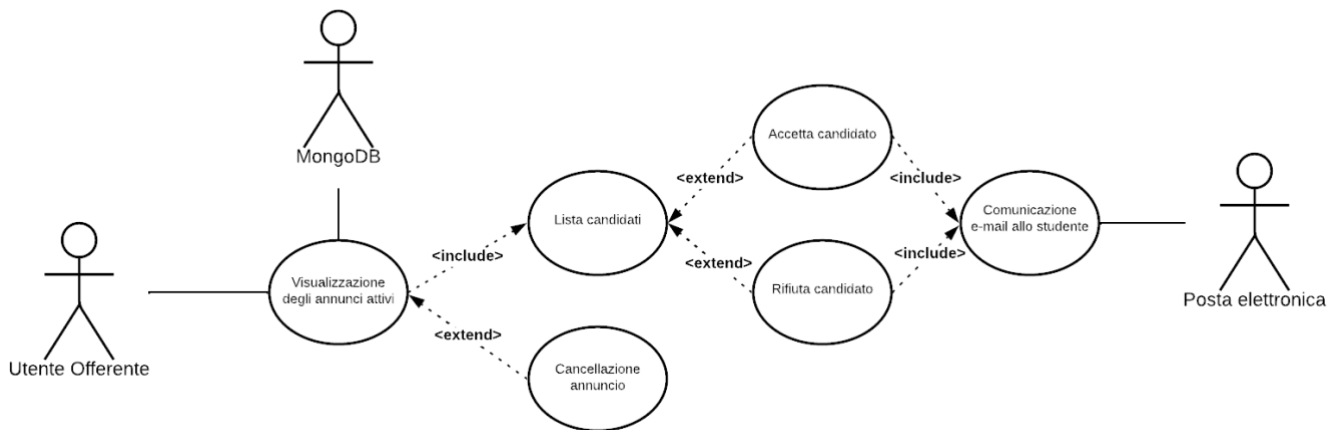
[exception 1]

- L'offerente ha la possibilità di inserire il *luogo* in cui il servizio deve essere svolto.
- *MongoDB* è il database che gestisce e fornisce tutti i dati riguardanti un annuncio – in particolare tipo di lavoro, luogo, data, ora, inserimento del tipo di pagamento e descrizione dettagliata del tipo di lavoro.

Exceptions:

[exception 1] nel caso in cui le informazioni elencate precedentemente non venissero inserite, il sito non permette la creazione dell'annuncio.

RF19 Visualizzazione degli annunci attivi, RF20 Lista candidati



Titolo: Visualizzazione degli annunci attivi

Riassunto: questo use case descrive la visualizzazione degli annunci attivi da parte dell'offerente e l'eventuale funzionalità di eliminazione di un annuncio attivo.

Descrizione:

- L'offerente visualizza tutti gli annunci in quel momento attivi [extension 1]
- Di ogni annuncio è possibile vedere:
 - tipologia di lavoro
 - data
 - ora
 - (eventuale) luogo
 - descrizione
 - metodo di pagamento

Il sito mostra lo *stato dell'annuncio* (assegnato/non assegnato) e la *lista candidati*. Quest'ultima permette di visualizzare:

- (nel caso in cui l'annuncio attivo sia in stato "non ancora assegnato") la lista di studenti – e relative informazioni personali – candidati per svolgere il lavoro [exception 1] (vedi [Lista candidati](#));
- (nel caso l'annuncio attivo sia in stato "assegnato") lo studente assegnato all'annuncio.
- L'offerente ha la possibilità di eliminare gli annunci attivi (vedi [Cancellazione annunci – State diagram](#)).
- *MongoDB* è il database che permette di gestire e conservare i dati degli annunci dell'utente offerente ancora attivi.

Exceptions:

[exception 1] nel caso in cui nessuno studente si sia ancora candidato per tale annuncio, la lista risulterà vuota.

Extensions:

[extension 1] gli annunci archiviati, ovvero non attivi, vengono visualizzati nella schermata degli annunci archiviati (vedi [Archiviazione annunci](#)).

Titolo: Lista candidati

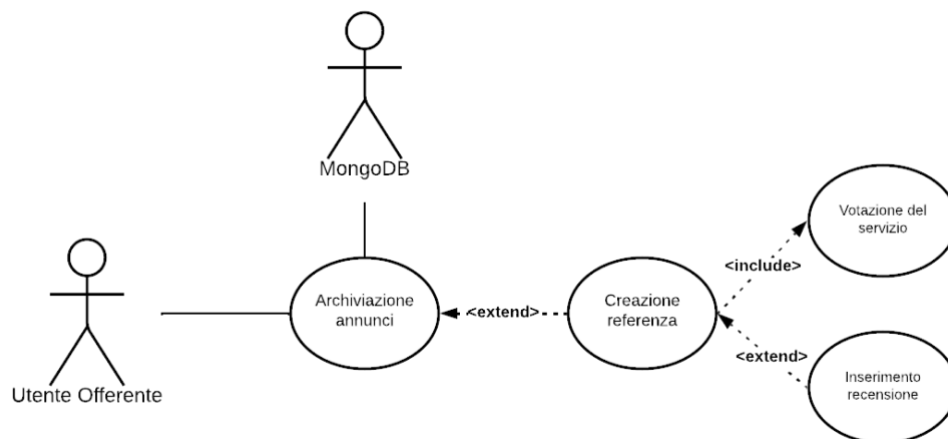
Riassunto: questo use case descrive la visualizzazione della lista di candidati ad uno specifico servizio che presenta lo stato “*non ancora assegnato*”.

Descrizione:

- L’offerente visualizza una lista di tutti gli studenti che si sono candidati per effettuare il servizio richiesto [exception 1]. Nello specifico, è possibile visualizzare un’anteprima di ogni studente contenente *nome, cognome, e-mail* e (eventuale) *contatto Telegram*.
- L’offerente può decidere di visualizzare il profilo personale dello studente (contenente, oltre alle informazioni precedentemente citate, anche la *descrizione*, la sezione di *statistiche profilo e valutazione* e quella di *Feedback e recensioni*) selezionandone uno dalla lista.
- L’offerente ha la possibilità di rifiutare la richiesta di uno specifico studente, senza dover necessariamente accettarne un altro. Allo studente verrà notificato il rigetto della richiesta via notifica e-mail.
- L’offerente può decidere di accettare uno studente. Quando questo avviene:
 - o lo stato dell’annuncio verrà impostato su “*assegnato*”;
 - o allo studente selezionato verrà comunicato via notifica e-mail l’accettazione della sua richiesta e il suo assegnamento al lavoro;
 - o agli studenti non selezionati verrà comunicato via notifica e-mail il rigetto della richiesta.
- La comunicazione all’utente studente dell’accettazione o del rifiuto di una sua richiesta ad un servizio avviene mediante l’utilizzo del servizio di *Posta Elettronica*.

Exceptions:

[exception 1] nel caso in cui nessuno studente si sia ancora candidato per tale annuncio, la lista risulterà vuota.



Titolo: Archiviazione annunci e creazione referenze

Riassunto: questo use case descrive la visualizzazione della sezione di “*archiviazione annunci*”, ovvero la sezione in cui è possibile visualizzare tutti gli annunci la cui data di lavoro è antecedente alla data di visualizzazione. Per ogni annuncio visualizzato è inoltre possibile inserire una referenza allo studente che ha effettuato il servizio.

Descrizione:

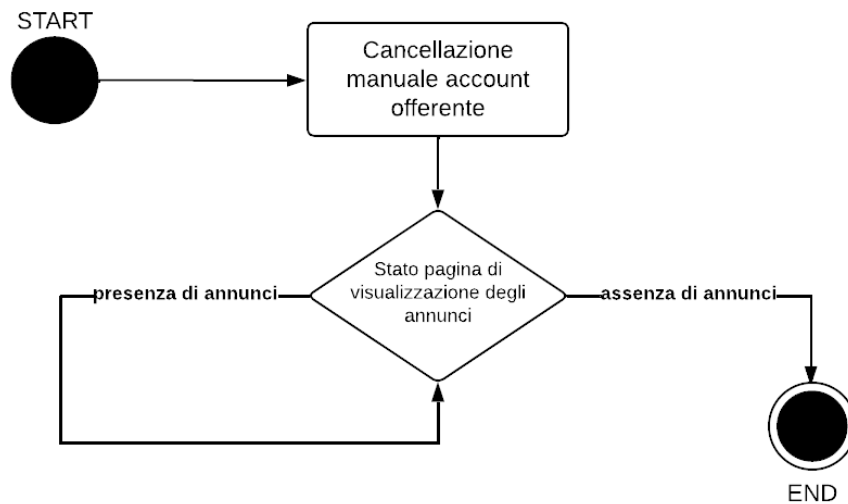
- L’offerente visualizza tutti gli annunci archiviati, ovvero tutti gli annunci la cui data e ora di lavoro sono antecedenti alla data e ora di visualizzazione dell’archivio [exception 1]
- Di ogni annuncio è possibile vedere *tipologia di lavoro, data, ora, (eventuale) luogo, descrizione e metodo di pagamento*. Viene inoltre visualizzato lo studente che ha effettuato il servizio e le sue informazioni personali.
- Per ogni annuncio archiviato, l’offerente deve lasciare una referenza per lo studente che ha effettuato il servizio. Questa prevede
 - o la *votazione del servizio*, ovvero una valutazione del servizio ricevuto e dello studente che lo ha effettuato su una scala di valori da 0 a 5;
 - o (facoltativo) la creazione di una recensione testuale allo studente [exception 2].
- *MongoDB* è il database utilizzato dal sito per la gestione dei dati relativi alle referenze create dall’offerente e per l’archiviazione degli annunci.

Exceptions:

[exception 1] gli annunci che non rispettano tale condizione sono visibili nella schermata di annunci attivi (vedi [Visualizzazione annunci attivi](#)).

[exception 2] se vuota, la creazione di una recensione può anche essere effettuata in un secondo momento rispetto alla votazione del servizio.

RF23 Cancellazione account



Titolo: Cancellazione account offerente

Riassunto: questo state-diagram descrive la procedura di cancellazione account da parte di un offerente.

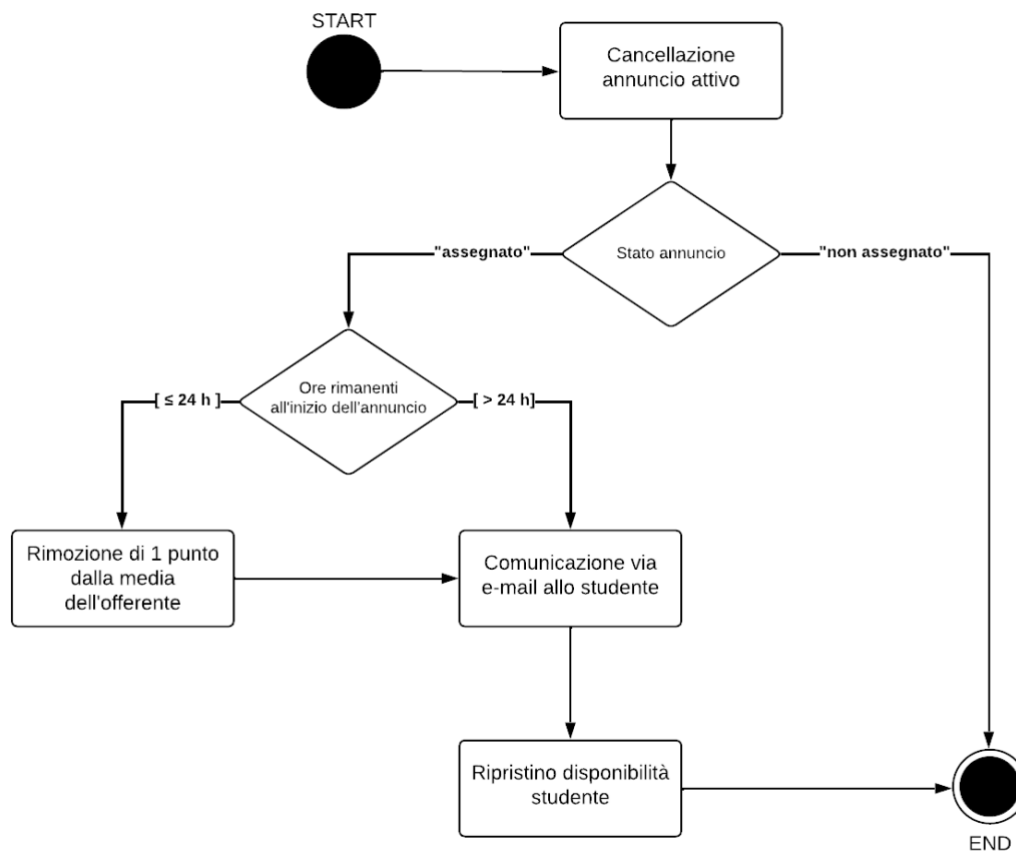
Descrizione:

- L'offerente ha la possibilità di cancellare il proprio account solamente se non ha annunci attivi [exception 1].

Exceptions:

[exception 1] nel caso di annunci attivi, l'offerente deve eliminarli personalmente uno ad uno.

RF24 Cancellazione annuncio



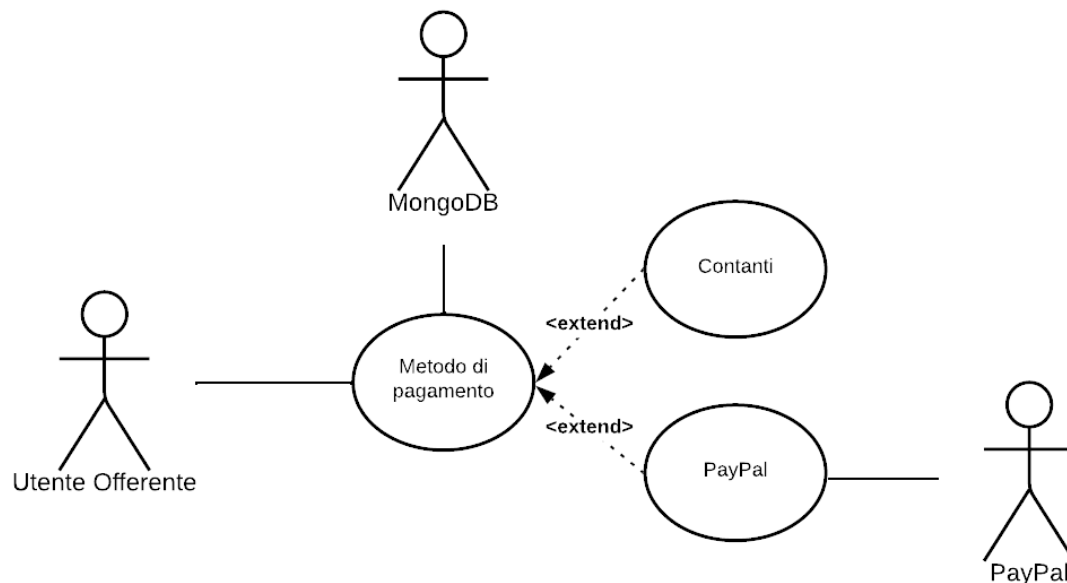
Titolo: Cancellazione annuncio

Riassunto: questo state-diagram descrive la procedura di cancellazione di un annuncio attivo.

Descrizione: L'offerente ha la possibilità di cancellare un proprio annuncio attivo in qualsiasi momento. Questa procedura di eliminazione potrebbe però avere delle ripercussioni sull'offerente. Di seguito una descrizione nello specifico:

- Quando l'offerente decide di eliminare in maniera definitiva un annuncio, il sistema controlla lo stato dell'annuncio:
 - o (stato "non assegnato") il sistema procede con l'eliminazione dell'annuncio nel database;
 - o (stato "assegnato") il sistema
 - calcola il periodo di tempo mancante al momento del servizio: se questo è inferiore a 24 h, la media del profilo offerente risentirà di un calo di una unità;
 - il sistema comunicherà allo studente, via notifica e-mail, l'eliminazione dell'annuncio;
 - la disponibilità dello studente per la fascia oraria occupata dall'annuncio viene ripristinata.
- Il sistema procede con l'eliminazione dell'annuncio dal database.

RF26 Pagamento



Titolo: Pagamento dei servizi ricevuti

Riassunto: questo use case descrive il meccanismo di pagamento da parte dell'offerente ad uno studente a servizio effettuato.

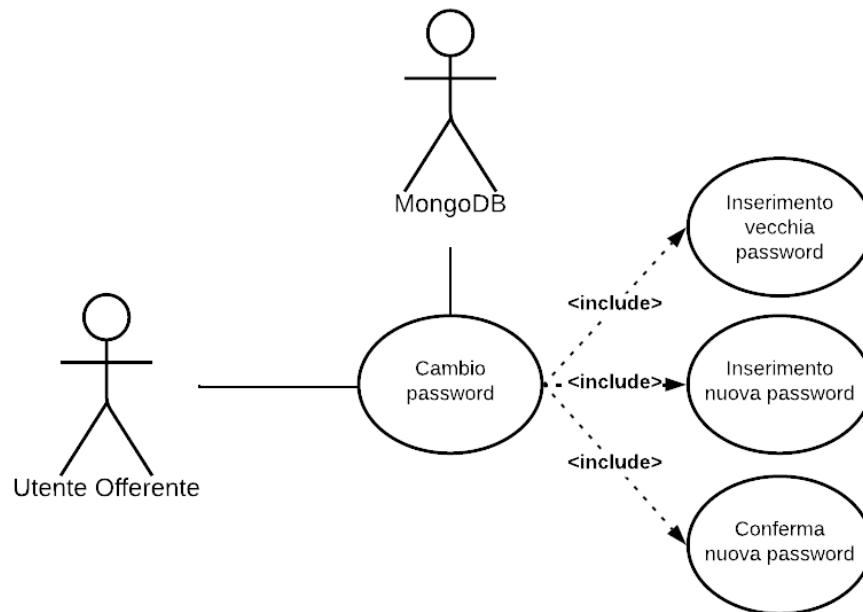
Descrizione:

- Al momento di creazione dell'annuncio, l'offerente specifica il metodo di pagamento che utilizzerà per pagare lo studente al termine del servizio [extension 1] (vedi [Creazione di un annuncio](#)). Questa informazione sarà infatti visibile nelle specifiche dell'annuncio (vedi [Specifiche annuncio](#)).
- Il pagamento verrà gestito:
 - o (contanti) autonomamente tra lo studente e l'offerente;
 - o (PayPal) tramite il sito, che permetterà – al termine del servizio – il pagamento tramite il sistema esterno *PayPal*.
- *MongoDB* è il database utilizzato per il salvataggio del metodo di pagamento scelto dall'offerente al momento di creazione dell'annuncio.

Extensions:

[extension 1] alla creazione di un annuncio, il sistema permette anche la selezione contemporanea di entrambi i metodi di pagamento, ovvero sia *contanti* sia *PayPal*. In questo caso sarà poi lo studente ad esprimere la propria preferenza per il metodo di pagamento all'offerente.

RF27 Cambio password



Titolo: Cambio password

Riassunto: questo use case descrive la procedura di cambio password di accesso al sito per l'utente offerente.

Descrizione:

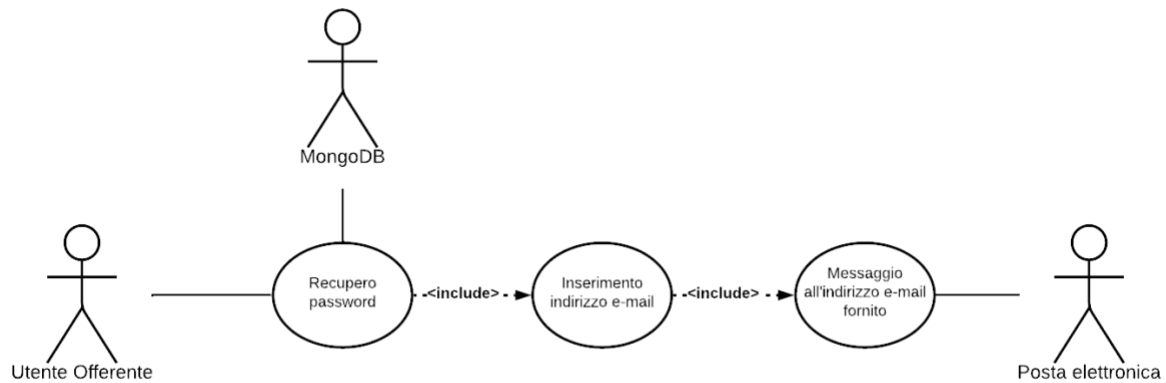
- La procedura di cambio password prevede l'inserimento, da parte dell'offerente, della *vecchia password* [exception 1].
- L'offerente deve quindi inserire una *nuova password*, confermandola ri-inserendola nell'apposito campo *conferma nuova password* [exception 2].
- *MongoDB* è il database utilizzato per verificare l'inserimento corretto della vecchia password, per modificare e salvare la nuova password scelta e per confermare l'inserimento corretto dell'ulteriore password di convalidazione.

Exceptions:

[exception 1] se l'inserimento della *vecchia password* non coincide con la password presente nel database, il sistema non permette di terminare il processo.

[exception 2] se i due campi *nuova password* e *conferma nuova password* non coincidono, il sistema non permette di terminare il processo.

RF28 Recupero password



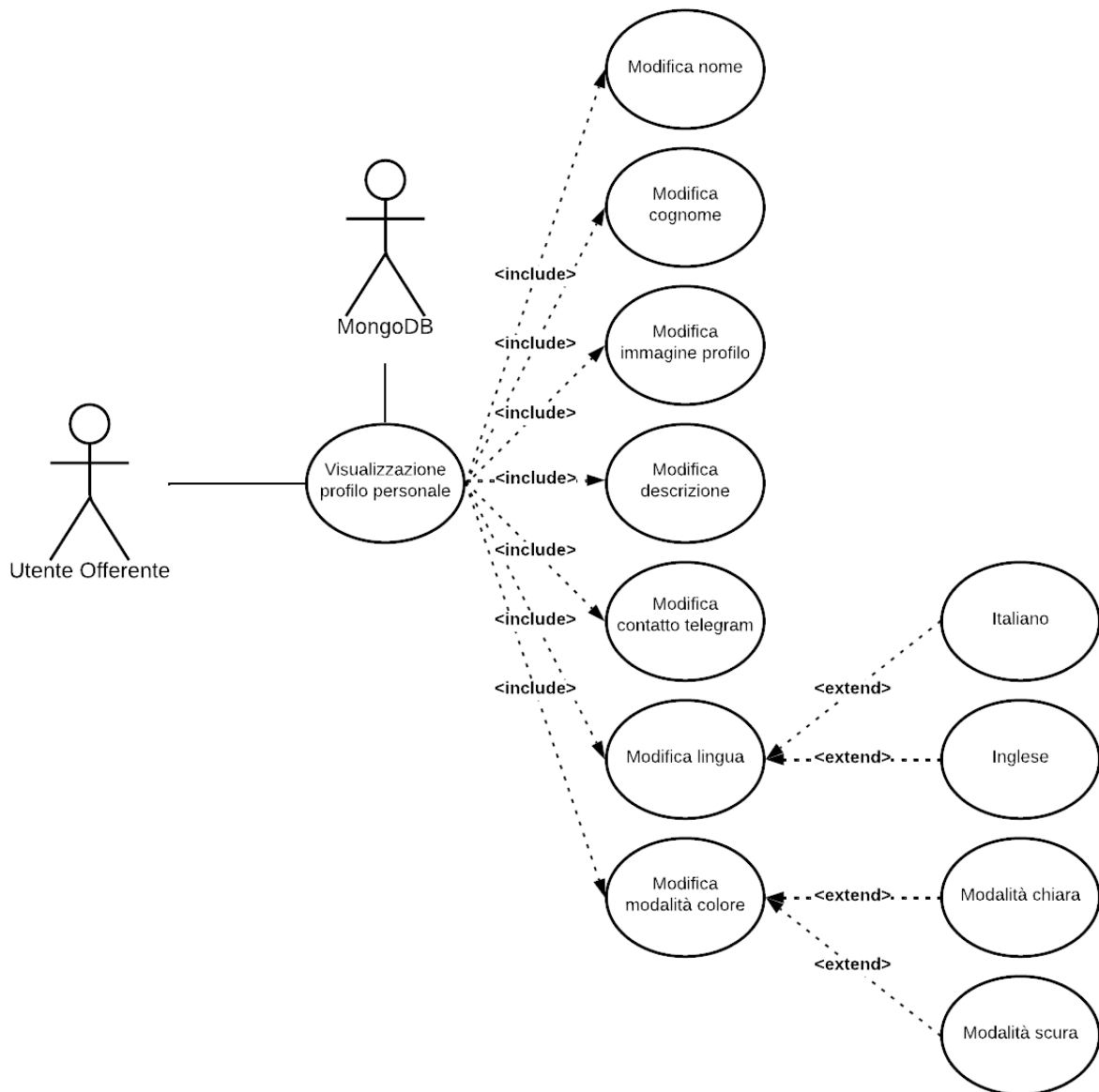
Titolo: Recupero password

Riassunto: questo use case descrive la procedura di recupero password per l'accesso al sito da parte dell'utente offerente.

Descrizione: per effettuare la procedura di recupero password:

- L'offerente deve inserire, nell'apposito campo, l'indirizzo e-mail utilizzato per effettuare la registrazione al sito.
- Il sito invia, tramite e-mail, un modulo per reimpostare la password di accesso al sito.
- *MongoDB* è il database utilizzato per salvare la password di accesso al sistema nel caso in cui l'utente offerente richieda di poterla recuperare. Nello specifico, *MongoDB* si occupa di verificare l'inserimento corretto dell'indirizzo e-mail dell'offerente e solo successivamente fornisce la password archiviata che viene inviata all'offerente mediante il servizio di *Posta elettronica*.

RF29 Profilo personale



Titolo: Visualizzazione e modifica del profilo personale

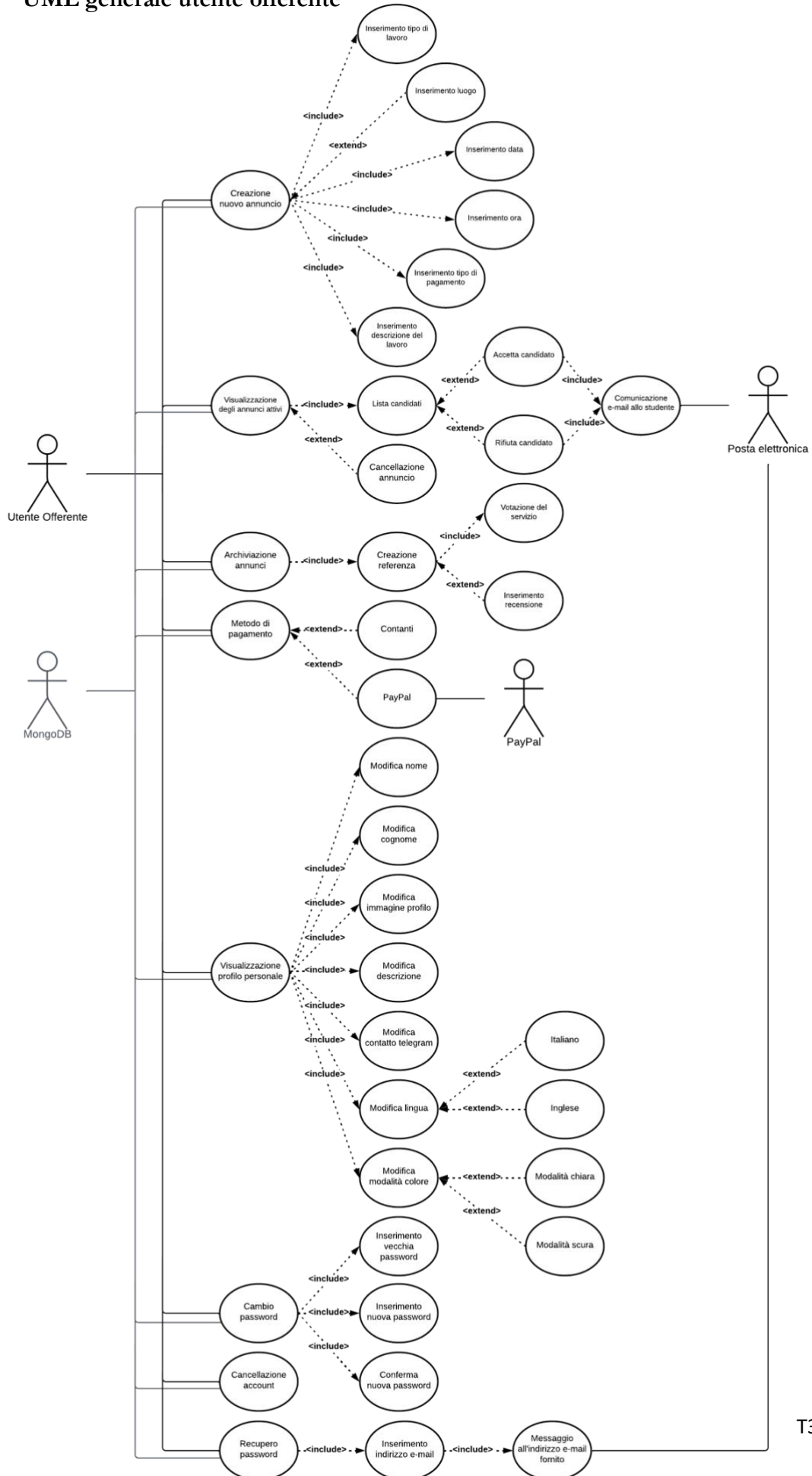
Riassunto: questo use case descrive la visualizzazione e la modifica del profilo personale dell'utente offerente.

Descrizione: accedendo alla propria area personale, l'offerente ha la possibilità di modificare in qualsiasi momento alcune delle informazioni personali visibili nel sito e alcune funzionalità del sistema:

- L'offerente può modificare il nome.
- L'offerente può modificare il cognome.
- L'offerente può aggiungere/modificare l'immagine profilo.
- L'offerente può modificare la descrizione personale.

- L'utente può modificare il proprio contatto *Telegram* attraverso il quale avverrà la comunicazione con l'offerente.
- L'utente può modificare la lingua di sistema, scegliendo tra italiano (default) e inglese.
- L'utente può modificare la modalità di colore del sistema scegliendo tra modalità chiara (default) e modalità scura.
- La gestione di tutti i dati personali dell'offerente è affidata a *MongoDB*, database che si occupa della loro archiviazione e, qualora venissero modificati, del loro aggiornamento.

2.3.1 UML generale utente offerente



3. Requisiti Non Funzionali

RNF1 Sicurezza

PROPRIETÀ	DESCRIZIONE	MISURA
<u>Inserimento “Strong Password”</u>	Requisiti di accettazione delle password inserite dall’utente offerente	La password deve presentare almeno 8 caratteri, di cui almeno una lettera maiuscola (A ... Z), un numero (0 ... 9) e un carattere speciale (!, #, \$, %, ^, &, -)
<u>Controllo di correttezza e-mail</u>	Controllo che l’indirizzo e-mail inserito dall’utente offerente durante la compilazione del modulo di registrazione sia valido	Nel caso in cui l’indirizzo e-mail inserito durante la registrazione non è corretto appare un messaggio di alert in cui viene segnalato l’errore
<u>Controllo di correttezza password</u>	Controllo che la password e la conferma password inserite durante la compilazione del modulo di registrazione rispettino i requisiti di accettazione password (vedi RNF1 – Strong Password) e corrispondano	Nel caso in cui password e conferma password sono differenti non è possibile continuare la registrazione appare un messaggio di alert in cui viene segnalata la non corrispondenza
<u>Controllo di inserimento dati non duplicati</u>	Controllo che l’indirizzo e-mail non sia già stato usato nella registrazione di un altro account	Nel caso in cui i dati inseriti appartengono ad un account già registrato appare un messaggio di alert in cui viene segnalato l’errore
<u>Interazione sicura client-server</u>	Modalità di trasmissione dei dati tra il client e il server	La trasmissione di dati tra il client e il server utilizza il protocollo HTTPS

RNF2 Tempistiche di promemoria

PROPRIETÀ	DESCRIZIONE	MISURA
<u>Tempistiche di promemoria</u>	Invio di un’e-mail di promemoria all’utente studente per ogni servizio che deve effettuare	L’e-mail di promemoria verrà inviata alle 00:00 del giorno precedente al servizio da effettuare

RNF3 **E-mail di conferma**

PROPRIETÀ	DESCRIZIONE	MISURA
<u>E-mail di conferma</u>	Tempistica dell'e-mail di conferma	Il sistema invia un'e-mail di conferma all'utente (studente/offerdente) entro 20 secondi dal completamento della registrazione

RNF4 **Privacy**

PROPRIETÀ	DESCRIZIONE	MISURA
<u>Tipologia di informazioni raccolte e modalità d'utilizzo</u>	Descrizione delle informazioni raccolte e il modo in cui sono utilizzate	Nel sito vengono raccolte informazioni sull'utente, quali informazioni di contatto e informazioni personali, utilizzate come supporto per favorire l'interazione offerdente-studente e per fornire altri servizi (<i>home page personalizzata</i>)
<u>Modalità di condivisione delle informazioni</u>	Utilizzo da parte del sito delle informazioni che l'utente inserisce	Il sito non utilizza le informazioni di contatto dell'utente a scopo di marketing. L'utente studente però, rispondendo ad un annuncio segnalando l'interesse per un'opportunità di lavoro, acconsente alla divulgazione delle proprie informazioni all'offerdente e ad essere contattato per le finalità correlate all'impiego offerto. Il sistema, inoltre, non rilascia ai suoi operatori nessuna informazione personale relativa ai clienti tranne nominativo e identificatore
<u>Accesso alle informazioni personali dell'utente</u>	Funzionalità che il sistema offre allo studente per gestire i propri dati personali	All'utente è fornita la possibilità di rivedere, correggere, aggiornare o cancellare il profilo in qualsiasi momento accedendo al proprio account
<u>Accettazione trattamento di autenticazione</u>	Rispetto da parte dell'utente dei "Termini e condizioni, Privacy e Policy" del sito	Nella parte finale del modulo di registrazione è presente una casella che deve essere spuntata dall'utente per poter procedere con la registrazione. In tale casella si richiede all'utente di accettare i termini e le vigenti disposizioni di legge di tutela della privacy. Tali disposizioni sono presenti in allegato pdf con il fine di permettere all'utente di leggerle

RNF5 Usabilità

PROPRIETÀ	DESCRIZIONE	MISURA
<u>Usabilità</u>	Interfaccia utente implementata in modo da garantire la massima operabilità e un apprendimento rapido delle operazioni disponibili	Un utente medio, dopo i primi 30 minuti di utilizzo, è in grado di usufruire di tutte le funzionalità che il sistema offre.

RNF6 Modularità

PROPRIETÀ	DESCRIZIONE	MISURA
<u>Modularità</u>	Suddivisione delle operazioni da svolgere in aree funzionali	Ogni funzionalità che il sistema offre è un modulo a sé stante

RNF7 Efficienza

PROPRIETÀ	DESCRIZIONE	MISURA
<u>Efficienza</u>	Capacità di produrre il massimo output con il minimo spreco di tempo, fatica, denaro, energia e materie prime a partire da un input dato	Il tempo di risposta massima per ogni pagina è di 4 secondi

RNF8 Supportabilità

PROPRIETÀ	DESCRIZIONE	MISURA
<u>Adattabilità schermo</u>	Tipologie di schermate sulle quali il sito funziona senza problemi	Il sito deve essere supportato nei formati di monitor, laptop, tablet e smartphone
<u>Impostazione modalità colore</u>	Tipologie di colorazione schermo che il sito prevede	Nel proprio profilo personale, ogni tipologia di utente ha in qualsiasi momento la possibilità di impostare la modalità colore (diurna/notturna) desiderata
<u>Lingua di sistema</u>	Lingue previste su tutte le schermate del sito	L'applicazione è fornita sia in lingua italiana sia in lingua inglese

RNF9 Affidabilità

PROPRIETÀ	DESCRIZIONE	MISURA
<u>Pagamenti sicuri</u>	Certezza di corretto funzionamento della transizione	Attraverso le API di Paypal si offre ad entrambe le tipologie di utente una garanzia al momento dei pagamenti. Queste sono garantite anche nel caso vi siano problemi di connettività

RNF10 Portabilità

PROPRIETÀ	DESCRIZIONE	MISURA
<u>Portabilità</u>	Capacità del sito di poter funzionare su più sistemi	Il sito è sviluppato con HTML5 e deve essere supportato nello specifico da Mozilla Firefox, Google Chrome, Internet Explorer, Safari

RNF11 Cambio password

PROPRIETÀ	DESCRIZIONE	MISURA
<u>Cambio password</u>	Metodo applicato per permettere all'utente di cambiare la password	La password da inserire deve rispettare il requisito di sicurezza "Strong Password", delineato in dettaglio nel RNF1

RNF12 Password dimenticata

PROPRIETÀ	DESCRIZIONE	MISURA
<u>Password dimenticata</u>	Metodo applicato per la segnalazione di password personale dimenticata	All'utente viene inviata un'e-mail per il recupero password dell'account entro 2 minuti dalla sua richiesta

RNF13 Inserimento errato della password

PROPRIETÀ	DESCRIZIONE	MISURA
<u>Inserimento errato della password</u>	Regole da seguire nel caso l'utente inserisca una password errata	In caso di inserimento errato della password per due volte consecutive, la possibilità di login per l'utente viene bloccata per 1 ora. Se al tentativo successivo l'utente sbaglia nuovamente, il sistema invia all'utente un'e-mail di identificazione

PROPRIETÀ	DESCRIZIONE	MISURA
<u>Richiesta rifiutata/scaduta</u>	Eliminazione di richieste rifiutate/scadute	Se l'offerente rifiuta una richiesta, l'annuncio viene eliminato dalla sezione dedicata dopo 72 ore dal declino della domanda di servizio, ma rimane comunque nel database del sistema. Ciò avviene anche nell'eventualità in cui la data in cui è richiesto il servizio sia passata
<u>Richiesta accettata</u>	Eliminazione di richieste accettate	Se l'offerente accetta la richiesta, questa viene eliminata dalla sezione dedicata dopo che l'utente offerente ha creato una referenza per il lavoro svolto, ovvero fino a 48 ore dopo lo svolgimento del servizio

4. Analisi del Contesto

In questo capitolo viene analizzato il contesto di funzionamento del sistema attraverso una descrizione testuale e una rappresentazione grafica basata su *Context Diagram*.

Nella parte seguente della sezione vengono invece presentati gli attori e i sistemi esterni con cui l'applicazione *StayBusy* si interfaccia.

3.1 Utenti e sistemi esterni

4.1.1 Utente studente

L'utente studente - identificato da **RF4** a **RF17** - utilizza l'applicazione per visualizzare una lista di offerte di servizi retribuiti, filtrati dal sito in base alle personali disponibilità e all'ubicazione dell'ateneo di studio, ed eventualmente per proporsi ad uno o più di questi.

4.1.2 Utente offerente

L'utente offerente - identificato da **RF18** a **RF30** - utilizza l'applicazione per creare annunci di lavoro e comunicare con gli studenti candidati per offrire a questi ultimi l'opportunità di svolgere il lavoro.

4.1.3 Posta elettronica

La posta elettronica è il servizio utilizzato dal sistema per confermare l'identità dell'utente che sta effettuando la registrazione, come da **RF3**, per permettere all'offerente di recuperare la password in caso di dimenticanza, come da **RF28** e per notificare lo studente, come da **RF8**, **RF9**, **RF12**, **RF21**.

4.1.4 PayPal

PayPal è il servizio di pagamento del sito, come da **RF26**, attraverso cui è possibile pagare, inviare denaro e accettare pagamenti in modo più rapido, semplice e sicuro senza dover immettere ogni volta i propri dati finanziari.

4.1.5 Google Calendar

Google Calendar, come da **RF4**, è il servizio utilizzato dal sistema per fornire allo studente la possibilità di memorizzare le proprie disponibilità e gestire gli annunci presi in carico.

4.1.6 Sistema credenziali universitarie

Sistema credenziali universitarie, come da **RF1**, **RF2**, è un sistema attraverso cui è possibile effettuare l'autenticazione degli studenti, tramite le credenziali universitarie già in possesso.

4.1.7 MongoDB

MongoDB è il database locale utilizzato dal sistema per contenere e gestire tutti i dati (es. utenze, annunci, prenotazioni).

4.2 Diagramma di contesto

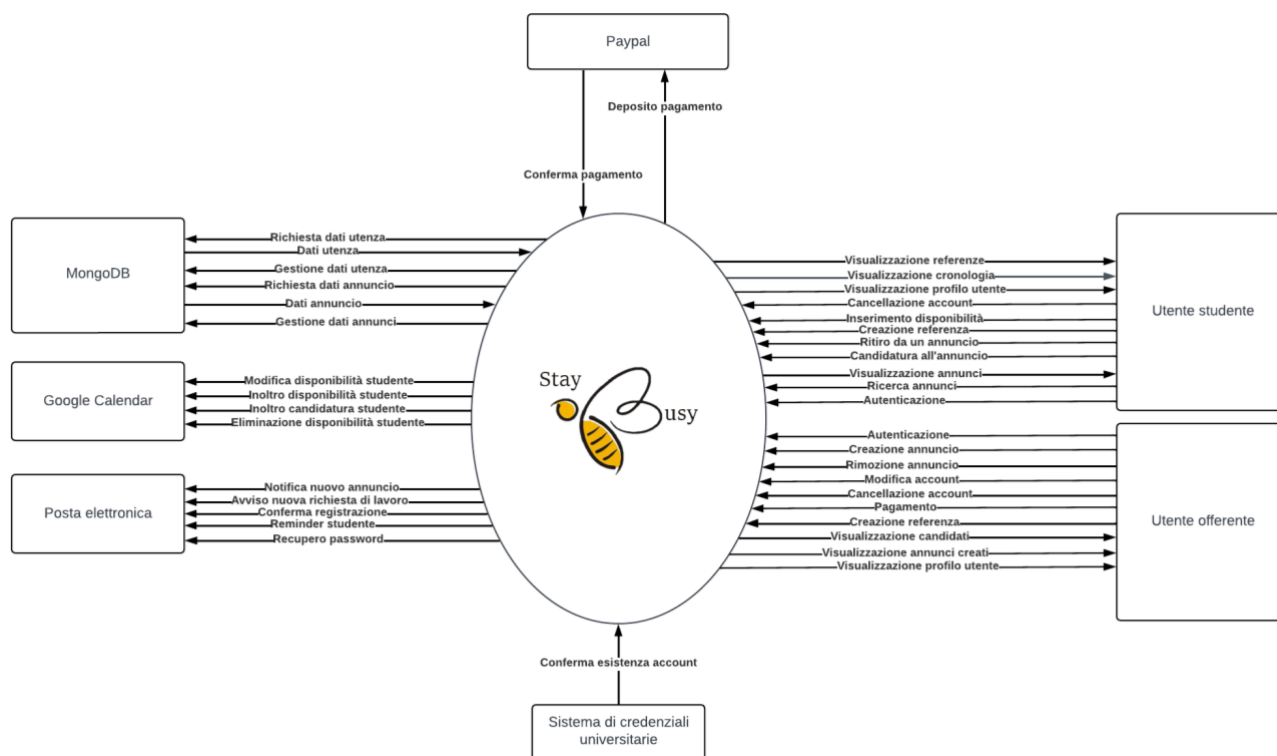


Figura 1. Diagramma di contesto per l'applicazione StayBusy 🐛

La Figura 1 mostra le principali interazioni tra attori o sistemi esterni e l'applicazione StayBusy. Di seguito è presente una descrizione delle interazioni evidenziate dal diagramma.

Per poter accedere al sito, l'utente studente deve svolgere la fase di login, come indicato da **RF2**. Effettuato l'accesso, lo studente deve necessariamente inserire nel calendario fornito le fasce orarie di disponibilità al lavoro giornaliere, settimanali oppure mensili che possono essere modificate in qualsiasi momento, come da **RF4**. Oltre a ciò, ha la possibilità di ricercare annunci attraverso la barra di ricerca fornita, come da **RF17**, di candidarsi ad uno specifico servizio, come da **RF6** e di cancellare manualmente e in qualsiasi momento il proprio account, come da **RF13**. Inoltre, una volta effettuato un servizio, allo studente è richiesto di dare una valutazione da 0 a 5 rispetto all'attività svolta e al rapporto instaurato con l'offerente, come da **RF9**. Il sito web offre in aggiunta la possibilità di visualizzare una cronologia dei servizi svolti nel passato, come da **RF11**, di vedere ed eventualmente modificare il proprio profilo personale, come da **RF17**, di visualizzare la lista di annunci e la storia di tutte le referenze ottenute dalle attività passate

Per effettuare l'accesso al sito, l'utente offerente deve eseguire l'autenticazione con l'applicazione come indicato da RF2, come da **RF2**. Effettuato l'accesso, l'offerente può usufruire di una serie di funzioni offerte dal sistema quali creare un nuovo annuncio inserendo la descrizione del servizio richiesto e definendo prezzo e tempistiche, come da **RF18**, rimuovere un annuncio precedentemente creato come da **RF24**, cancellare, modificare ed eliminare in ogni istante il proprio account, come da **RF29** e **RF23**. Infine, al termine di un lavoro, è richiesto obbligatoriamente all'offerente di effettuare il pagamento, come da **RF26** e di dare una votazione da 0 a 5 al servizio svolto, come da **RF25**. Oltre a ciò, il sistema offre la possibilità di visualizzare la lista di tutti gli studenti che si sono candidati per un servizio, come da

RF20, l'elenco di tutti gli annunci archiviati, come da **RF22**, e del proprio profilo personale, come da **RF29**.

Attraverso il sistema credenziali universitarie è possibile effettuare l'autenticazione degli studenti, utilizzando le credenziali universitarie personali già in possesso dagli studenti.

Il servizio di posta elettronica permette l'interazione tra utente e sistema. Ci sono infatti molti casi in cui il sistema invia e-mail automatiche sia a studenti che a offerenti. In particolare, l'utente offerente riceve una notifica e-mail per la conferma della registrazione, come da **RF3**, e, nel caso in cui faccia richiesta, per il recupero password, come da **RF28**. Inoltre, attraverso questo servizio viene inviata allo studente un'e-mail di reminder per ricordargli del servizio che ha concordato con l'offerente, come da **RF8**, e, se lo studente ha precedentemente attivato il flag di richiesta, riceve un'e-mail di avviso di richiesta di prestazione da parte di un offerente presso cui ha già prestato servizio in passato, come da **RF12**.

Il pagamento elettronico, utilizzato per remunerare lo studente per il lavoro svolto, avviene tramite PayPal, servizio che riceve i dati del pagamento dal sito e che conferma l'avvenuta transazione correttamente, come da **RF26**.

La gestione del calendario fornito all'utente studente è affidata a Google Calendar. Attraverso questo servizio lo studente ha la possibilità di inserire, modificare ed eliminare le sue disponibilità in ogni istante.

Il sito utilizza MongoDB come database locale al fine di contenere tutti i dati presenti all'interno del sistema. In particolare, MongoDB gestisce tutti i dati personali e di accesso riguardanti l'utente offerente, mentre quelli dell'utente studente vengono gestiti autonomamente dal sistema credenziali universitarie. Il database contiene inoltre tutti i dati relativi agli annunci pubblicati dall'offerente, alle referenze di utente studente e offerente e, in generale, tutti i dati presenti e salvati nel sito. Ogni qualvolta questi dati vengono modificati dall'utente, il sito demanda a MongoDB la loro gestione.

5. Analisi dei Componenti

In questo capitolo viene presentata l'architettura – in termini di componenti interni al sistema definiti sulla base dei requisiti precedentemente analizzati – minimizzando il livello di coesione. Vengono inoltre rappresentate le interconnessioni tra i vari componenti – identificando le interfacce presenti verso sistemi esterni – tramite l'uso di Component Diagram e viene valutato il livello di unione tra i vari componenti.

5.1 Definizione dei componenti

5.1.1 Home page studente

Motivazione. Nel mock-up **5.3 Vista studente, Home page** viene espressa l'esigenza di un componente **Home page studente** che permetta di interfacciarsi e scambiare dati con altre componenti quali **Annunci vista studente, Profilo studente, Calendario**, e altre.

5.1.2 Credenziali universitarie studente

Motivazione. Il **RF1 modalità studente** esprime l'esigenza di un componente **Credenziali universitarie studente** che si interfaccia con il sistema di gestione delle credenziali di ateneo. L'autenticazione dell'utente studente si appoggia infatti al sistema di credenziali universitarie. In particolare, il componente gestisce l'autenticazione dell'utente tramite il sistema di credenziali di ateneo, mantenendo poi attiva la sessione utente all'interno dell'applicazione.

5.1.3 Interfaccia registrazione studente

Motivazione. Nel **RF1 modalità studente** e nei mock-up **5.2 Registrazione account, Registrazione account studente** viene espressa l'esigenza di un componente **Interfaccia registrazione studente** che permetta all'utente studente di creare un account di accesso al sito e di interfacciarsi con il sistema **Credenziali universitarie studente** per acquisire i dati utente necessari. Il componente gestisce quindi l'intera procedura di registrazione dell'utente studente al sito.

5.1.4 Interfaccia login studente

Motivazione. Il **RF2** e il mock-up **5.1 Fase di login, Login account studente** esprimono l'esigenza di un componente **Interfaccia login studente** che permetta all'utente studente di accedere al sito fornendo l'e-mail universitaria. Questo componente si interfaccia infatti con il componente **Credenziali universitarie studente** al fine di permettere allo studente di autenticarsi correttamente.

5.1.5 Annunci vista studente

Motivazione. Nel **RF4** e nel mock-up **5.3 Vista studente, Home page** viene espressa l'esigenza di un componente **Annunci vista studente** per gestire la visualizzazione di un “catalogo” annunci a cui lo studente può candidarsi in base alle proprie disponibilità espresse e gestite dal componente **Calendario**. Questo si occupa di interfacciarsi con i componenti **Visualizza annunci** e **Ricerca annunci**.

5.1.6 Visualizza annuncio

Motivazione. Il mock-up **5.3 Vista studente, Visualizzazione specifiche annuncio** specifica l'esigenza di un componente **Visualizza annuncio** adibito alla visualizzazione dei dettagli di un annuncio attivo specifico e alla gestione dell'eventuale richiesta di candidatura da parte dello studente.

5.1.7 Ricerca annuncio

Motivazione. Il **RF17** e il mock-up **5.3 Vista studente, Home page** specificano l'esigenza di un componente **Ricerca annuncio** al fine di gestire la ricerca degli annunci per parole chiave da parte dell'utente studente. Il componente permette quindi un utilizzo migliore e più efficiente del sito da parte degli utenti studenti.

5.1.8 Profilo studente

Motivazione. Nel **RF16** viene espressa l'esigenza di un componente **Profilo studente** al fine di permettere all'utente studente la gestione delle proprie informazioni personali, del proprio account e del *setting* del sito. Questo permette inoltre, interfacciandosi con il componente **Referenze studente**, la visualizzazione delle referenze ottenute.

5.1.9 Referenze studente

Motivazione. Il **RF10** specifica l'esigenza di un componente **Referenze studente** che permetta all'utente studente di visualizzare le proprie referenze e valutazioni create, a seguito del completamento di un servizio, dagli utenti offerenti. Come si nota dal mock-up **5.4 Vista offerente, Visualizzazione profilo studente**, queste informazioni sono visibili anche lato offerente nel caso in cui lo studente invii una richiesta per un annuncio da quest'ultimo pubblicato.

5.1.10 Calendario

Motivazione. Nel **RF4** e nel mock-up **5.3 Vista studente, Modifica disponibilità settimanali** si evidenzia la necessità di un componente **Calendario** per la gestione delle disponibilità dell'utente studente. Questo permette – oltre alla visualizzazione – l'inserimento, la modifica e l'eliminazione di fasce orarie di disponibilità. Questo mostra inoltre allo studente le fasce orarie occupate da una candidatura attiva e quelle occupate da una candidatura in attesa. Questo componente è estremamente importante per il funzionamento del sistema: **Calendario** è il componente che permette la visualizzazione filtrata per fasce orarie di disponibilità nel componente **Annunci vista studente**.

5.1.11 Cronologia lavori effettuati

Motivazione. Il **RF11** esplicita l'esigenza di un componente **Cronologia lavori effettuati** al fine di permettere una sola visualizzazione dei servizi effettuati dall'utente studente.

5.1.12 Home page offerente

Motivazione. Nel mock-up **5.4 Vista offerente, Home page** viene espressa l'esigenza di un componente **Home page offerente** che permetta di interfacciarsi e scambiare dati con altre componenti quali **Annunci vista offerente, Profilo offerente**, e altre.

5.1.13 Interfaccia registrazione offerente

Motivazione. Nel **RF1 modalità offerente** e nei mock-up **5.2 Registrazione account, Registrazione account offerente** viene espressa l'esigenza di un componente **Interfaccia registrazione offerente** che permetta all'utente di creare un account di accesso al sito a seguito dell'inserimento di determinate informazioni personali, quali *nome, cognome, e-mail, password*, e altre. Il componente gestisce quindi l'intera procedura di registrazione dell'utente al sito.

5.1.14 Interfaccia login offerente

Motivazione. Il **RF2** e il mock-up **5.1 Fase di login, Login account offerente** esprimono l'esigenza di un componente **Interfaccia login offerente** che permetta all'utente offerente di accedere al sito inserendo l'e-mail e la password fornite al momento di registrazione. Il componente si occupa anche di mantenere attiva la sessione utente all'interno dell'applicazione.

5.1.15 Crea annuncio

Motivazione. Il **RF18** e il mock-up **5.4 Vista offerente, Creazione nuovo annuncio** specificano l'esigenza di un componente **Crea annuncio** che gestisca la procedura di creazione nuovo annuncio effettuata dall'utente offerente. Nello specifico dovrà gestire i dati inseriti dall'offerente – quali *Tipo lavoro, Logo, Data*, e altri – e controllare che questi rispettino i vincoli e le restrizioni imposte dal sito.

5.1.16 Annuncio vista offerente

Motivazione. Nel **RF19** e nel mock-up **5.4 Vista offerente, Home page** viene espressa l'esigenza di un componente **Annunci vista offerente** per gestire la visualizzazione degli annunci attivi precedentemente creati. Questo si occupa di interfacciarsi con il componente **Annuncio singolo**.

5.1.17 Annuncio singolo

Motivazione. Il mock-up **5.4 Vista offerente, Visualizzazione specifiche annuncio** specifica l'esigenza di un componente **Annuncio singolo** al fine di permettere la visualizzazione all'offerente dei dettagli di un annuncio pubblicato e la lista di candidati ad esso proposti (**RF20**). Il componente si occupa anche della gestione della lista di tali candidati nel momento in cui l'offerente dovesse scegliere o rifiutarne uno (**RF7**).

5.1.18 Profilo offerente

Motivazione. Nel **RF29** viene espressa l'esigenza di un componente **Profilo offerente** al fine di permettere all'utente offerente la gestione delle proprie informazioni personali, del proprio account e del *setting* del sito. Questo permette inoltre, interfacciandosi con il componente **Referenze offerente**, la visualizzazione delle referenze ottenute.

5.1.19 Referenze offerente

Motivazione. Il **RF9** specifica l'esigenza di un componente **Referenze offerente** che permetta all'utente offerente di visualizzare le proprie referenze e valutazioni create dagli studenti che hanno completato un annuncio di lavoro pubblicato da tale offerente.

5.1.20 Pagamento

Motivazione. Il **RF26** specifica l'esigenza di un componente **Pagamento** adibito alla gestione dei pagamenti da parte dell'utente offerente. Questo può inoltre appoggiarsi al sistema esterno PayPal al fine di permettere l'esecuzione della transazione in modo sicuro e protetto.

5.1.21 Cronologia lavori completati

Motivazione. Il **RF22** specifica l'esigenza di un componente **Cronologia lavori completati** adibito alla visualizzazione degli annunci pubblicati la cui data è antecedente alla data di visualizzazione del sito.

5.1.22 Crea referenza

Motivazione. Il **RF25** e Il **RF9** specificano l'esigenza di un componente **Crea referenza** per la gestione della creazione di nuove referenze sia da offerenti verso studenti sia viceversa. Questo deve effettuare un controllo sul legame tra studente e offerente (un offerente può lasciare una recensione ad uno studente se e solo se quest'ultimo ha svolto un annuncio di lavoro pubblicato da un offerente, mentre uno studente può lasciare una recensione ad un offerente se e solo se ha svolto un lavoro per tale offerente) e gestire i dati inseriti in tale referenza. Questo si interfaccia con i componenti **Cronologia lavori effettuati** e **Cronologia lavori completati**.

5.1.23 Posta elettronica

Motivazione. Come citato nei **RF3**, **RF8**, **RF9**, **RF12**, **RF21** e **RF28** è necessario un componente **Posta elettronica** adibita alla gestione di notifica utenti che si interfaccia con il sistema **Posta elettronica**. Nello specifico, questo si occupa della gestione – rispettivamente ai RF citati – dell'invio di un'e-mail di conferma all'utente dopo aver effettuato la registrazione, dell'invio di un'e-mail di reminder allo studente il giorno prima dello svolgimento di un lavoro a cui si è candidato, dell'invio di un'email di notifica allo studente quando un offerente con cui ha già avuto contatto pubblica un nuovo annuncio, dell'invio di un'e-mail di notifica all'offerente quando uno studente si candida per un annuncio attivo e dell'invio di e-mail di notifica all'offerente per il ripristino password di accesso al sito.

5.2 Diagramma dei componenti

Al fine di rendere il diagramma dei componenti del sito *StayBusy* il più chiaro e semplice possibile, questo è stato diviso in un diagramma lato studente e in uno lato offerente:

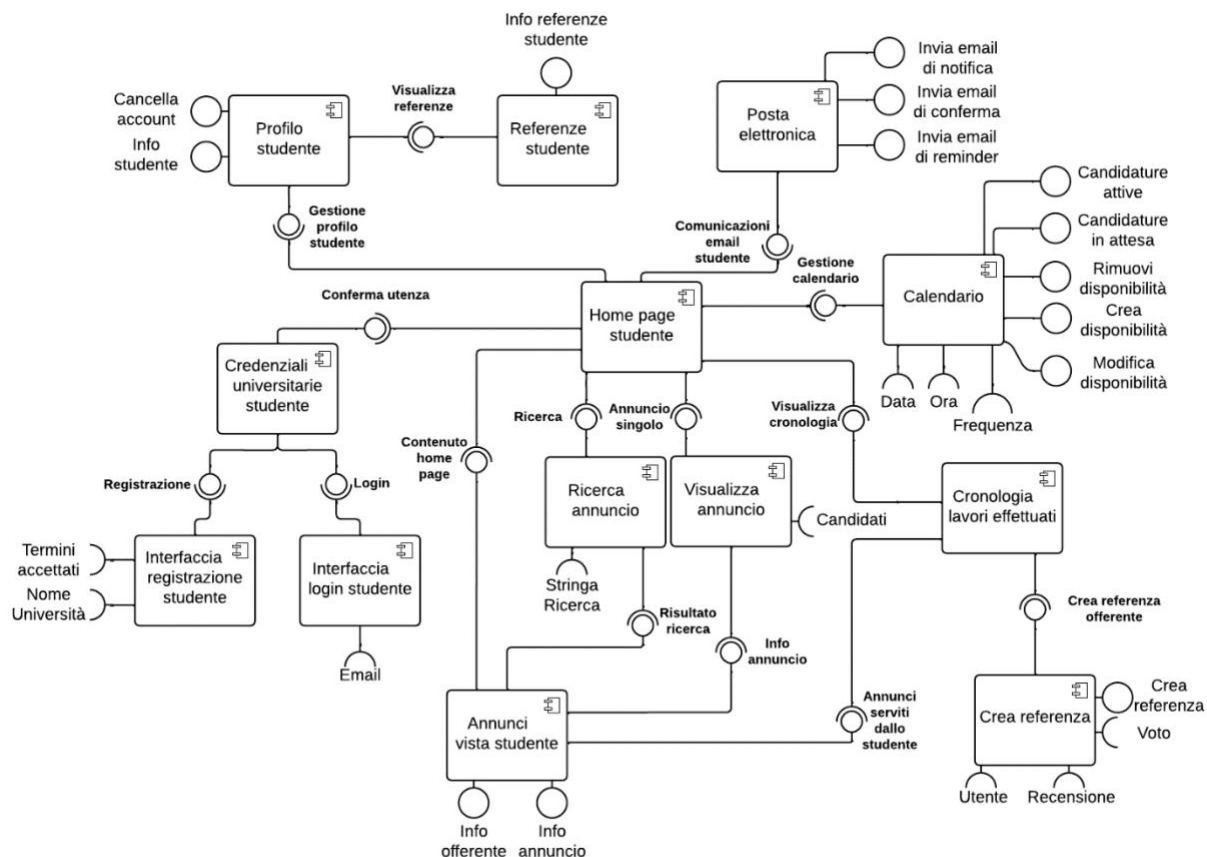


Figura 2. Diagramma dei componenti parte studente dell'applicazione StayBusy 🐼

La Figura 2 mostra i componenti di sistema (sezione studente) e la loro interconnessione; a seguire una descrizione dei componenti e delle interfacce presenti nel diagramma.

5.2.1 Home page studente

Interfaccia richiesta – Gestione profilo studente: è richiesta la gestione del profilo utente dal componente **Profilo studente**.

Interfaccia richiesta – Conferma utenza: richiede la correttezza delle credenziali universitarie fornite al componente **Credenziali universitarie studente**, permettendo quindi l'autenticazione al sito.

Interfaccia richiesta – Contenuto home page: è richiesta la visualizzazione degli annunci attivi dal componente **Annunci vista studente**.

Interfaccia richiesta – Ricerca: è richiesta la visualizzazione degli annunci attivi filtrati secondo i criteri inseriti dall'utente dal componente **Ricerca annuncio**.

Interfaccia richiesta – Annuncio singolo: sono richieste le informazioni e i dettagli relativi ad un singolo annuncio dal componente **Visualizza annuncio**.

Interfaccia richiesta – Visualizza cronologia: è richiesta la visualizzazione della cronologia dei lavori effettuati (e conclusi) dall'utente studente dalla componente **Cronologia lavori effettuati**.

Interfaccia richiesta – Gestione calendario: è richiesta la visualizzazione delle disponibilità date dall'utente dal componente **Calendario**.

Interfaccia richiesta – Comunicazioni e-mail studente: è richiesta la gestione delle comunicazioni e-mail con cui contattare e/o allertare lo studente dalla componente **Home page studente**.

5.2.2 Credenziali universitarie studente

Interfaccia fornita – Conferma utenza: fornisce la correttezza delle credenziali universitarie fornite al componente **Home page studente**, permettendo quindi l'autenticazione al sito.

Interfaccia fornita – Registrazione: fornisce la verifica delle credenziali al componente **Credenziali universitarie studenti** per confermare la registrazione dell'account.

Interfaccia fornita – Login: fornisce la verifica delle credenziali al componente **Interfaccia registrazione studente** per effettuare l'accesso al sito.

5.2.3 Interfaccia registrazione studente

Interfaccia richiesta – Registrazione: richiede la verifica delle credenziali dal componente **Credenziali universitarie studenti** per confermare la registrazione dell'account.

Interfaccia richiesta – Termini accettati: l'utente studente deve acconsentire ai *Termini e Condizioni* e *Privacy e Policy* imposti dal sito per effettuare la registrazione dell'account.

Interfaccia richiesta – Nome università: richiede l'inserimento, da parte dell'utente, del nome dell'università frequentata al fine di permettere l'indirizzamento verso il portale di ateneo.

5.2.4 Interfaccia login studente

Interfaccia richiesta – Login: è necessario ottenere la verifica delle credenziali dal componente **Credenziali universitarie studenti** per effettuare l'accesso al sito.

Interfaccia richiesta – E-mail: richiede l'inserimento, da parte dell'utente, dell'indirizzo e-mail universitario.

5.2.5 Annunci vista studente

Interfaccia fornita – Contenuto home page: fornisce la visualizzazione degli annunci attivi alla componente **Home page studente**.

Interfaccia fornita – Risultato ricerca: fornisce gli annunci attivi al componente **Ricerca annuncio**.

Interfaccia fornita – Info annuncio: fornisce le specifiche dell'annuncio – quali *Tipo lavoro*, *Luogo*, *Descrizione*, e altre – alla componente **Visualizza annuncio**.

Interfaccia fornita – Annunci serviti dallo studente:

Interfaccia fornita – Info offerente: fornisce le informazioni del profilo dell'offerente presenti nel sito, quali *Nome*, *Cognome*, *E-mail*, e altre.

Interfaccia fornita – Info annuncio: fornisce le specifiche dell'annuncio, quali *Tipo lavoro, Data, Ora, Luogo, Descrizione e Metodo di pagamento*.

5.2.6 Visualizza annuncio

Interfaccia fornita – Annuncio singolo: fornisce le informazioni e i dettagli relativi ad un singolo annuncio al componente **Home page studente**.

Interfaccia richiesta – Info annuncio: sono richieste le specifiche dell'annuncio – quali *Tipo lavoro, Luogo, Descrizione*, e altre – dalla componente **Annunci vista studente**.

Interfaccia richiesta – Candidati: l'utente studente può candidarsi all'annuncio.

5.2.7 Ricerca annuncio

Interfaccia fornita – Ricerca: fornisce la visualizzazione degli annunci attivi filtrati secondo i criteri inseriti dall'utente al componente **Home page studente**.

Interfaccia richiesta – Risultato ricerca: richiede gli annunci attivi al componente **Annunci vista studente** al fine di effettuare la visualizzazione filtrata secondo i criteri espressi dall'utente studente.

Interfaccia richiesta – Stringa ricerca: è necessario l'inserimento di una stringa di ricerca, contenente parole chiavi, secondo cui verrà applicato un filtro (momentaneo) agli annunci presenti nel sito.

5.2.8 Profilo studente

Interfaccia fornita – Gestione profilo studente: fornisce la gestione del profilo utente al componente **Home page studente**.

Interfaccia richiesta – Visualizza referenze: è necessaria la visualizzazione delle referenze dal componente **Referenze studente**.

Interfaccia fornita – Cancella account: l'utente studente può avviare la procedura di cancellazione account.

Interfaccia fornita – Info studente: l'utente studente può visualizzare le proprie informazioni personali presenti sul sito.

5.2.9 Referenze studente

Interfaccia fornita – Visualizza referenze: fornisce la visualizzazione delle referenze al componente **Profilo studente**.

Interfaccia fornita – Info referenze studente: permette la visualizzazione delle referenze personali lasciate dagli offerenti a cui si è prestato servizio.

5.2.10 Calendario

Interfaccia fornita – Gestione calendario: fornisce la visualizzazione delle disponibilità date dall'utente, delle disponibilità occupate da un annuncio a cui ci si è candidati e a cui si è stati accettati e delle disponibilità temporaneamente occupate da un annuncio a cui ci si è candidati e non si è ancora stati né accettati né rifiutati.

Interfaccia richiesta – Data: è necessario, per creare una nuova candidatura, l'inserimento della data in cui lo studente è disponibile.

Interfaccia richiesta – Ora: è necessario, per creare una nuova candidatura, l’inserimento della fascia oraria (ora fine e ora inizio) in cui lo studente è disponibile.

Interfaccia richiesta – Frequenza: lo studente può inserire la frequenza con cui tale disponibilità si ripete (es: ogni giorno, ogni mese, ogni lunedì).

Interfaccia fornita – Candidature attive: fornisce all’utente studente la visualizzazione delle candidature attive a lavori non ancora svolti.

Interfaccia fornita – Candidature in attesa: fornisce all’utente studente la visualizzazione delle candidature a lavori per cui non è ancora stato accettato.

Interfaccia fornita – Rimuovi disponibilità: fornisce all’utente la possibilità di rimuovere una disponibilità data precedentemente.

Interfaccia fornita – Crea disponibilità: fornisce all’utente la possibilità di creare una nuova disponibilità. Questa prevede, come specificato sopra, l’inserimento di *luogo*, *data* ed eventuale *frequenza*.

Interfaccia fornita – Modifica disponibilità: fornisce all’utente la possibilità di modificare una disponibilità precedentemente inserita nel sistema.

5.2.11 Cronologia lavori effettuati

Interfaccia richiesta – Annunci serviti dallo studente: sono richiesti gli annunci già effettuati dallo studente del componente **Annunci vista studente**.

Interfaccia richiesta – Crea referenza offerente: è richiesta, per ogni lavoro effettuato, la referenza creata dal rispettivo offerente da parte del componente **Crea referenza**.

Interfaccia fornita – Visualizza cronologia: fornisce la visualizzazione della cronologia dei lavori effettuati (e conclusi) dall’utente studente alla componente **Home page studente**.

5.2.22 Crea referenza

Interfaccia fornita – Crea referenza offerente: fornisce, per ogni lavoro effettuato, la referenza creata dal rispettivo offerente alla componente **Cronologia lavori effettuati**.

Interfaccia richiesta – Utente: è necessario, per ogni referenza, specificare l’utente offerente a cui questa è associata.

Interfaccia richiesta – Recensione: è possibile, per ogni referenza, creare una recensione dell’utente studente.

Interfaccia richiesta – Voto: è necessario, per ogni referenza, indicare un voto da 0 a 5 per l’utente studente.

Interfaccia fornita – Crea referenza: permette allo studente di confermare i dati inseriti e creare la referenza.

5.2.23 Posta elettronica

Interfaccia fornita – Comunicazioni e-mail studente: fornisce la gestione delle comunicazioni e-mail con cui contattare e/o allertare lo studente alla componente **Home page studente**.

Interfaccia fornita – Invia e-mail di notifica: fornisce la gestione dell’invio delle e-mail di notifica allo studente.

Interfaccia fornita – Invia e-mail di conferma: fornisce la gestione dell'invio delle e-mail di conferma allo studente.

Interfaccia fornita – Invia e-mail di reminder: fornisce la gestione dell'invio delle e-mail di reminder allo studente.

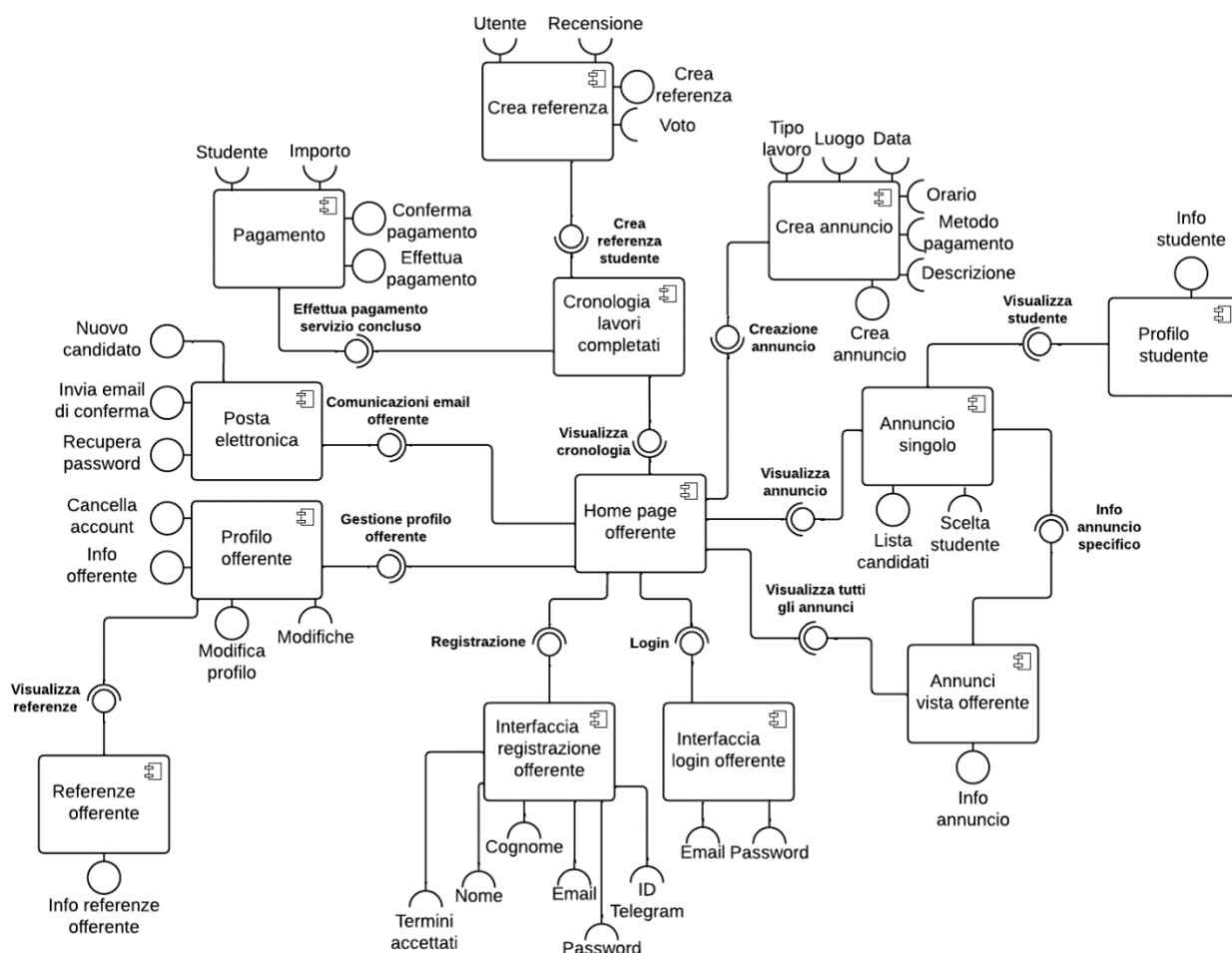


Figura 3. Diagramma dei componenti parte offerente dell'applicazione StayBusy

La Figura 3 mostra i componenti di sistema (sezione offerente) e la loro interconnessione; a seguire una descrizione dei componenti e delle interfacce presenti nel diagramma. Si noti che i componenti **Crea referenza** e **Posta elettronica** sono equivalenti tra i due diagrammi, ma per linearità sono stati riportati anche nella lista seguente.

5.2.12 Home page offerente

Interfaccia richiesta – Gestione profilo offerente: richiede la gestione del profilo utente al componente **Profilo offerente**.

Interfaccia richiesta – Comunicazioni e-mail offerente: richiede la gestione delle comunicazioni e-mail con cui contattare e/o allertare l'offerente alla componente **Posta elettronica**.

Interfaccia richiesta – Visualizza cronologia: richiede la visualizzazione della cronologia dei lavori completati al componente **Cronologia lavori completati**.

Interfaccia richiesta – Creazione annuncio: sono richieste le informazioni relative all'annuncio creato dal componente **Crea annuncio**.

Interfaccia richiesta – Visualizza annuncio: richiede le informazioni e i dettagli relativi ad un singolo annuncio al componente **Annuncio singolo**.

Interfaccia richiesta – Visualizza tutti gli annunci: richiede la visualizzazione degli annunci attivi al componente **Annunci vista offerente**.

Interfaccia richiesta – Registrazione: richiede la corretta creazione dell'account e l'autenticazione nel sistema dal componente **Interfaccia registrazione offerente**.

Interfaccia richiesta – Login: richiede la verifica delle credenziali al componente **Interfaccia login offerente** per effettuare l'accesso al sito.

5.2.13 Interfaccia registrazione offerente

Interfaccia fornita – Registrazione: fornisce la corretta creazione dell'account e l'autenticazione nel sistema al componente **Home page offerente**.

Interfaccia richiesta – Nome: è necessario fornire il proprio nome per la creazione di un account.

Interfaccia richiesta – Cognome: è necessario fornire il proprio cognome per la creazione di un account.

Interfaccia richiesta – E-mail: è necessario fornire il proprio indirizzo e-mail per la creazione di un account.

Interfaccia richiesta – Password: è necessario impostare una password di accesso, e confermarla inserendola una seconda volta, per la creazione di un account.

Interfaccia richiesta – ID Telegram: è possibile fornire il proprio ID Telegram per semplificare il contatto con gli utenti studenti.

Interfaccia richiesta – Termini accettati: è necessario accettare i *Termini e condizioni* e *Privacy Policy* per la creazione di un account.

5.2.14 Interfaccia login offerente

Interfaccia fornita – Login: fornisce la verifica delle credenziali al componente **Home page** per effettuare l'accesso al sito.

Interfaccia richiesta – E-mail: richiede l'inserimento, da parte dell'utente, dell'indirizzo e-mail fornito al momento della registrazione.

Interfaccia richiesta – Password: richiede l'inserimento, da parte dell'utente, della password di accesso al sito.

5.2.15 Crea annuncio

Interfaccia fornita – Creazione annuncio: fornisce le informazioni relative all'annuncio creato al componente **Home page offerente**.

Interfaccia richiesta – Tipo lavoro: richiede l'inserimento della tipologia di lavoro dell'annuncio.

Interfaccia richiesta – Luogo: richiede l'inserimento del luogo in cui sarà svolto il lavoro.

Interfaccia richiesta – Data: richiede l'inserimento della data in cui verrà svolto il lavoro.

Interfaccia richiesta – Orario: richiede l'inserimento della fascia oraria (ora inizio e ora fine) in cui il servizio deve essere svolto.

Interfaccia richiesta – Metodo pagamento: richiede l'inserimento della modalità di pagamento preferita dall'offerente (PayPal, contanti oppure entrambi).

Interfaccia richiesta – Descrizione: richiede l'inserimento della descrizione del servizio richiesto.

Interfaccia fornita – Crea annuncio: permette all'offerente di confermare i dati forniti e creare l'annuncio.

5.2.16 Annuncio vista offerente

Interfaccia fornita – Visualizza tutti gli annunci: fornisce la visualizzazione degli annunci attivi alla componente **Home page offerente**.

Interfaccia fornita – Info annuncio specifico: fornisce le specifiche dell'annuncio – quali *Tipo lavoro*, *Luogo*, *Descrizione*, e altre – alla componente **Annuncio singolo**.

Interfaccia fornita – Info annuncio: fornisce le specifiche dell'annuncio, quali *Tipo lavoro*, *Data*, *Ora*, *Luogo*, *Descrizione* e *Metodo di pagamento*.

5.2.17 Annuncio singolo

Interfaccia richiesta – Info annuncio specifico: sono richieste le specifiche dell'annuncio – quali *Tipo lavoro*, *Luogo*, *Descrizione*, e altre – dalla componente **Annuncio vista offerente**.

Interfaccia fornita – Visualizza annuncio: fornisce le informazioni e i dettagli relativi ad un singolo annuncio al componente **Home page offerente**.

Interfaccia fornita – Lista candidati: fornisce una lista di studenti, e relative informazioni personali, candidati per svolgere tale annuncio.

Interfaccia richiesta – Scelta studente: l'utente offerente può scegliere uno studente tra quelli presenti nella lista candidati per svolgere il lavoro.

5.2.18 Profilo offerente

Interfaccia richiesta – Visualizza referenze: è necessaria la visualizzazione delle referenze dal componente **Referenze offerente**.

Interfaccia fornita – Gestione profilo offerente: fornisce la gestione del profilo utente al componente **Home page offerente**.

Interfaccia richiesta – Modifiche: è possibile effettuare una modifica sui dati personali presenti nel sito.

Interfaccia fornita – Modifica profilo: l'utente offerente può confermare le modifiche effettuate sui propri dati personali.

Interfaccia fornita – Cancella account: l'utente offerente può avviare la procedura di cancellazione account.

Interfaccia fornita – Info offerente: fornisce la visualizzazione delle informazioni personali presenti nell'account del sito.

5.2.19 Referenze offerente

Interfaccia fornita – Visualizza referenze: fornisce la visualizzazione delle referenze al componente **Profilo offerente**.

Interfaccia fornita – Info referenze offerente: permette la visualizzazione delle referenze personali lasciate dagli studenti che hanno svolto un servizio all'offerente.

5.2.20 Pagamento

Interfaccia fornita – Effettua pagamento servizio concluso: fornisce la conferma di avvenuto pagamento al componente **Cronologia lavori completati**.

Interfaccia richiesta – Studente: richiede indicazioni a quale studente è necessario effettuare il pagamento.

Interfaccia richiesta – Importo: richiede l'importo da versare allo studente.

Interfaccia fornita – Effettua pagamento: l'offerente ha la possibilità di procedere al pagamento tramite sistema PayPal.

Interfaccia fornita – Conferma pagamento: fornisce la conferma di avvenuta transazione.

5.2.21 Cronologia lavori completati

Interfaccia richiesta – Effettua pagamento servizio concluso: richiede la conferma di avvenuto pagamento dal componente **Pagamento**.

Interfaccia richiesta – Crea referenza studente: richiede, per ogni lavoro completato, la referenza creata dal rispettivo studente dal componente **Crea referenza**.

Interfaccia fornita – Visualizza cronologia: fornisce la visualizzazione della cronologia dei lavori completati al componente **Home page offerente**.

5.2.22 Crea referenza

Interfaccia fornita – Crea referenza studente: fornisce, per ogni lavoro completato, la referenza creata dal rispettivo studente alla componente **Cronologia lavori completati**.

Interfaccia richiesta – Utente: è necessario, per ogni referenza, specificare l'utente offerente a cui questa è associata.

Interfaccia richiesta – Recensione: è possibile, per ogni referenza, creare una recensione dell'utente offerente.

Interfaccia richiesta – Voto: è necessario, per ogni referenza, indicare un voto da 0 a 5 per l'utente offerente.

Interfaccia fornita – Crea referenza: permette all'offerente di confermare i dati inseriti e creare la referenza.

5.2.23 Posta elettronica

Interfaccia fornita – Comunicazioni e-mail offerente: fornisce la gestione delle comunicazioni e-mail con cui contattare e/o allertare l'offerente alla componente **Home page offerente**.

Interfaccia fornita – Nuovo candidato: fornisce la gestione dell’invio delle e-mail di avviso nuovo candidato all’offerente.

Interfaccia fornita – Invio e-mail di conferma: fornisce la gestione dell’invio delle e-mail di conferma all’offerente.

Interfaccia fornita – Recupero password: fornisce la gestione dell’invio delle e-mail per il recupero password account all’offerente.

6 Note ed eventuali

I requisiti funzionali (RF) e i mock-up citati nella [sezione 5](#) fanno riferimento al documento “D1: Analisi dei Requisiti”.

Tutte le tipologie di diagramma sopra riportate – *Use case Diagram*, *State Diagram*, *diagramma di contesto*, *diagramma delle componenti* - sono state realizzate interamente dal nostro gruppo tramite il sito www.lucidchart.com. Rendiamo quindi disponibili tutte le immagini presenti nel documento [qui](#).