



Money Expense

Documento di specifica dei requisiti funzionali e non funzionali
con diagrammi use case.

Scopo del Documento

Questo documento avrà il compito di fornire una descrizione dettagliata dei molteplici requisiti funzionali e non funzionali indicati nel documento **D1-Gog**, integrandoli con ulteriori informazioni.

In particolare, volendo svolgere un confronto con il documento **D1-Gog**, nel seguente documento:

- Verrà effettuata una modellazione mediante diagrammi **UML** dei vari **requisiti funzionali**, a cui verrà accompagnata una descrizione in linguaggio naturale oppure espressa mediante diagrammi di stato, diagrammi delle attività o swimlanes.
- Verrà fornita inoltre una rappresentazione in formato tabellare dei **requisiti non funzionali**. In quest'ultima sarà possibile visualizzare le misure per verificare l'effettivo adempimento dello specifico requisito nel nostro sistema.
- Inoltre, presentiamo un **diagramma del contesto** per definire chiaramente i diversi attori esterni con i quali il nostro sistema interagisce e le modalità di scambio di informazioni.
- Verrà infine presentato un **diagramma dei componenti**, il quale mostrerà i diversi componenti interni al sistema e i flussi di scambio d'informazione tra di essi.

Requisiti funzionali

In questa sezione del documento riportiamo i requisiti funzionali del progetto attraverso l'utilizzo dei diagrammi dei casi d'uso (use-case diagram) di UML e/o una descrizione in linguaggio naturale o mediante ulteriori diagrammi UML.

Registrazione



Descrizione Use Case "Utente si registra"

Titolo: Utente si registra

Riassunto: questo **use case** descrive il procedimento di registrazione dell'utente che porta all'effettiva creazione di un nuovo account (vedi **RF 1**).

Descrizione:

1. L'utente anonimo seleziona il bottone **Sign Up** dalla schermata di login per procedere con la creazione di un nuovo account.
2. L'utente anonimo inserisce nome, cognome, email e password (**vedi exception 1, extension 1**).
3. Nella fase di registrazione viene svolta la richiesta al database che porterà alla creazione di un nuovo record per l'utente.
4. Il sistema informa l'utente dell'avvenuta registrazione e lo reindirizza alla pagina del proprio profilo.

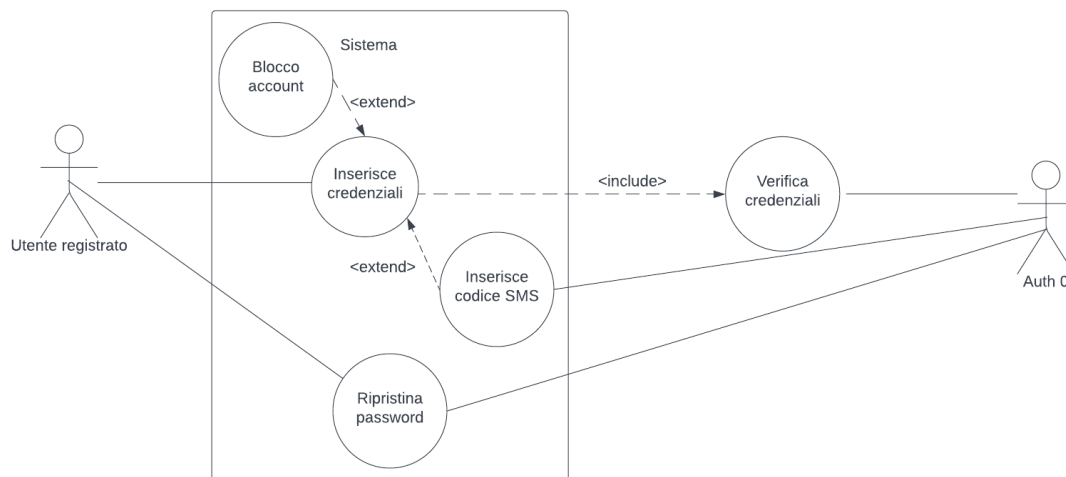
Exception:

1. Se l'indirizzo email è già associato a un account o se la password scelta non è conforme (**vedi RNF 1**), il sistema visualizza un messaggio di errore per informare l'utente sulla natura dell'errore, incoraggiandolo a inserire un'email e/o una password diverse.

Extension:

1. Su base volontaria, l'utente può inserire un proprio contatto telefonico in modo da poter attivare l'autenticazione a due fattori aggiungendo così un layer di sicurezza all'accesso al suo profilo (vedi **RF 3**).

Login



Descrizione Use Case "Accedere sito"

Titolo: Accedere sito.

Riassunto: questo use case illustra il processo di inserimento delle credenziali per accedere al proprio account sul sito (vedi **RF 4**).

Descrizione:

1. L'utente viene indirizzato alla schermata di login nei casi in cui il token di sessione sia assente o scaduto oppure l'utente accede a delle funzioni riservate solamente agli utenti registrati (e.g. visualizzare la dashboard) oppure nel momento in cui richiede esplicitamente di effettuare l'accesso al proprio account.
2. L'utente inserisce la propria email e la propria password (vedi [exception 1](#)).
3. Il sistema esterno [Auth0](#) effettua il controllo delle credenziali e della password (vedi [extension 1](#)).
4. Il sistema conferma l'accesso dell'utente nel proprio account e ne mostra la schermata col profilo.

Exception:

1. Nel caso di username errato o password errata il sistema mostrerà un messaggio di errore all'utente invitandolo al reinserimento delle credenziali.

Extension:

1. Nel caso in cui l'utente abbia abilitato l'autenticazione a due fattori, dopo aver inserito correttamente le credenziali riceverà un codice via SMS al numero specificato in fase di registrazione e provvederà a inserirlo.

Descrizione Use Case "Ripristina password"

Titolo: Ripristina password.

Riassunto: questo use case illustra il procedimento attraverso il quale l'utente può ripristinare la propria password (vedi **RF 2**).

Descrizione:

1. Nel caso di password dimenticata, l'utente seleziona la voce di ripristino password.
2. L'utente specifica il proprio username/email.
3. Il sistema invia una mail all'utente con un link dove può modificare la propria password.
4. L'utente inserisce la nuova password ([vedi exception 1](#)).
5. Il sistema esterno [Autho](#) gestisce il ripristino della password dell'utente.
6. Una volta modificata la password, l'utente effettua l'accesso al proprio account.

Exception:

1. Se la password inserita non risulta conforme con il **RNF 1**, il sistema mostra un messaggio di errore e chiede all'utente di inserire una nuova password, possibilmente conforme alle richieste. Questo processo viene iterato finché la password decisa non rispetta questi limiti.

Descrizione Use Case "Blocco account"

Titolo: Blocco account.

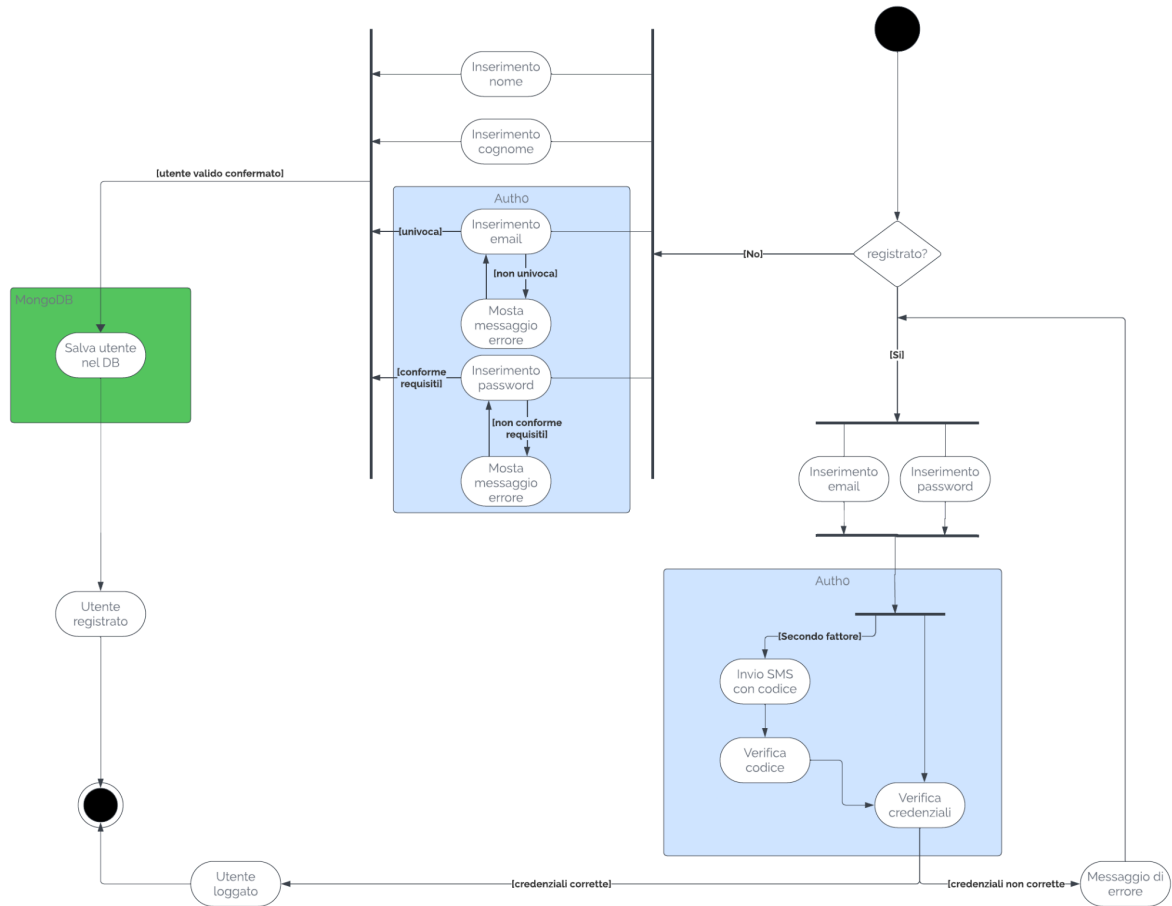
Riassunto: questo use case specifica le modalità di blocco dell'account nel caso di tentativi di accesso senza successo da parte di utenti non autorizzati o soggetti terzi.

Descrizione:

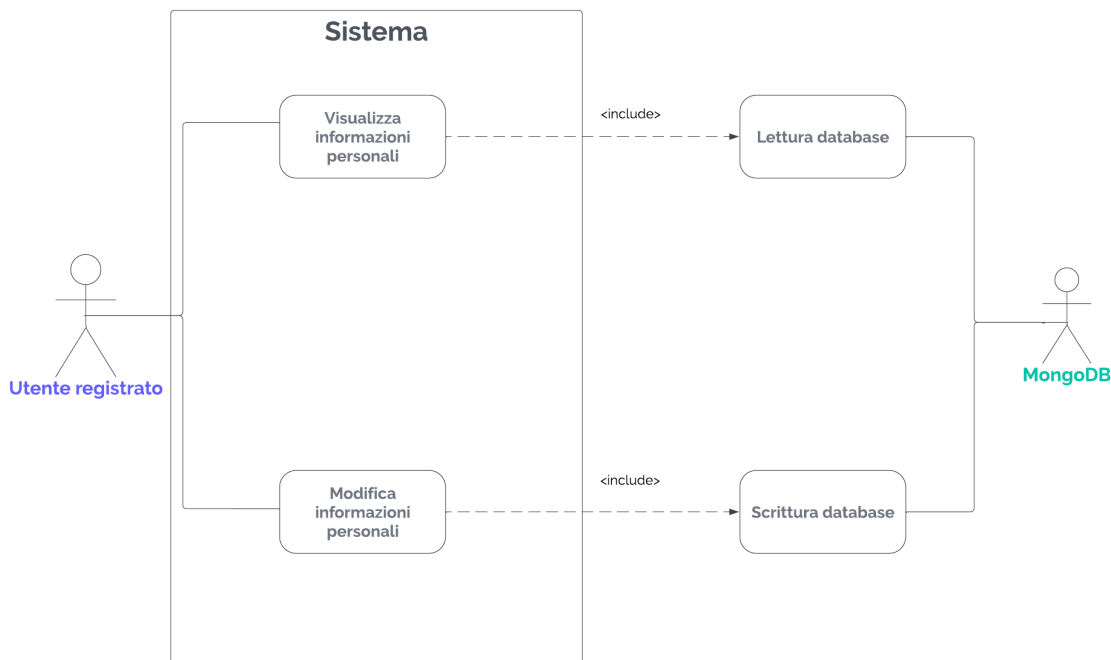
1. Il sistema notifica l'utente del blocco dell'account e dell'impossibilità di accedervi nel caso in cui l'utente sbaglia a digitare la password di accesso per 5 volte.
2. Il sistema invia automaticamente all'utente un'email con un link per la modifica della password dell'utente.
3. Una volta modificata la password, l'utente effettua l'accesso al proprio account.

Activity diagram per registrazione e login

Nel seguente activity diagram viene dettagliato ulteriormente tutto il processo di autenticazione e registrazione che un utente deve seguire per poter usufruire della piattaforma Money Expense.



Settings



Descrizione Use Case "Visualizza informazioni personali"

Titolo: Visualizza informazioni personali.

Riassunto: questo use case specifica i modi con cui avviene la visualizzazione dei dati personali dell'utente registrato (vedi **RF 5**).

Descrizione:

1. Una volta effettuato l'accesso al sistema, l'utente viene reindirizzato nella schermata [Dashboard](#).
2. L'utente accede alla pagina delle impostazioni mediante il bottone nel menu di sinistra, cliccando la voce [Settings](#).
3. L'utente registrato avrà la possibilità di visualizzare tutti i dati personali inseriti nella fase di registrazione.
4. Viene eseguita una richiesta al database per la ricezione dei dati dell'utente.

Descrizione Use Case "Modifica informazioni personali"

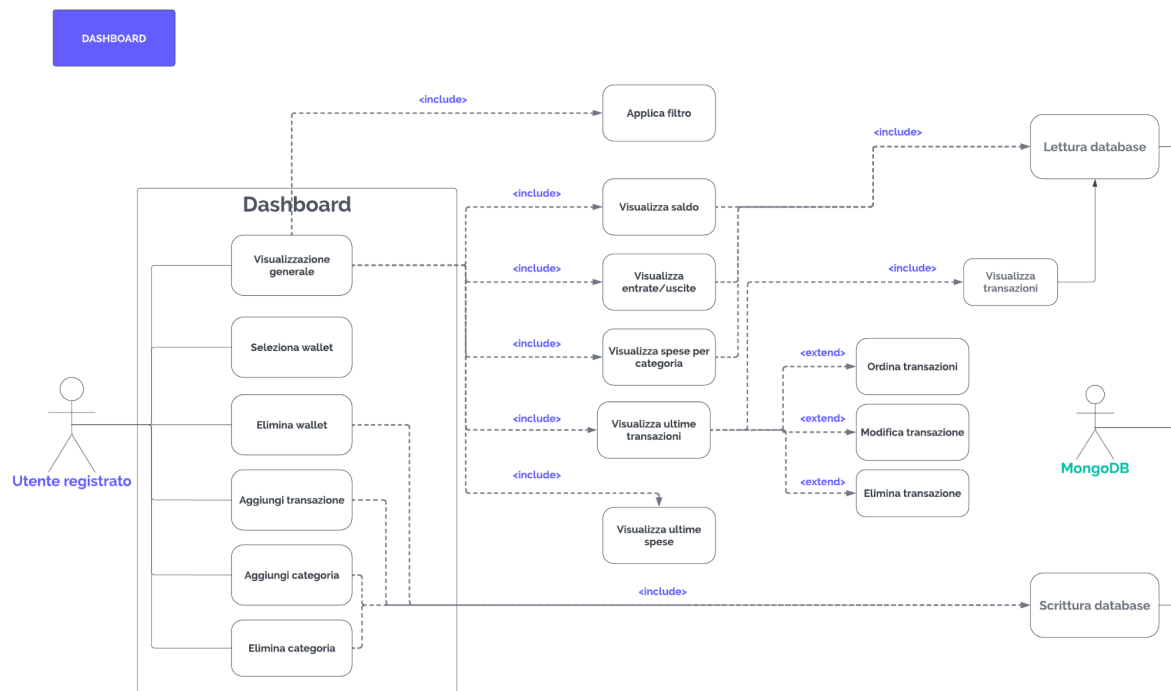
Titolo: Modifica informazioni personali.

Riassunto: questo use case specifica i modi con cui avviene la modifica dei dati personali dell'utente registrato (vedi **RF 5**).

Descrizione:

1. Una volta registrato, l'utente seleziona la schermata [Settings](#).
2. L'utente registrato avrà la possibilità di modificare tutti i dati personali eccetto l'indirizzo email.
3. Viene eseguita una richiesta al database per la modifica del record dell'utente.

Dashboard



Descrizione Use Case "Visualizzazione Generale"

Titolo: Visualizzazione Generale.

Riassunto: questo use case delinea i modi con cui l'utente potrà visualizzare la sua situazione finanziaria personale in tempo reale (vedi **RF 6**).

Descrizione:

1. Una volta effettuato l'accesso al sistema, l'utente viene reindirizzato nella schermata **Dashboard**, dove si trova un riassunto con i dati dell'attuale situazione.
2. Per ricevere i dati relativi all'utente verrà svolta una richiesta al database.
3. L'utente può ora visualizzare la sua **Dashboard** personale in piattaforma.

Descrizione Use Case "Selezione wallet"

Titolo: Selezione wallet.

Riassunto: questo use case delinea i modi con cui avviene la visualizzazione dell'informazione riguardante il saldo di una determinata wallet dell'utente (vedi **RF 10**).

Descrizione:

1. Una volta effettuato l'accesso al sistema, l'utente viene reindirizzato nella schermata **Dashboard**.
2. Nel menu di sinistra, l'utente clicca sulla voce **Wallets**, e viene reindirizzato nella pagina che mostra tutti i wallet presenti sulla piattaforma.
3. L'utente registrato seleziona un wallet tra quelli registrati in piattaforma per visualizzarne il contenuto.
4. L'utente visualizza i dati registrati all'interno del wallet, quindi il saldo e i movimenti che questo ha avuto di recente.

Descrizione Use Case "Elimina wallet"

Titolo: Elimina wallet.

Riassunto: questo use case delinea i modi con cui avviene l'eliminazione di un wallet dell'utente (vedi **RF 11**).

Descrizione:

1. Una volta effettuato l'accesso al sistema, l'utente viene reindirizzato nella schermata **Dashboard**.
2. Nel menu di sinistra, l'utente clicca sulla voce **Wallets**, e viene reindirizzato nella pagina che mostra tutti i wallet presenti sulla piattaforma.
3. L'utente visualizza una lista con tutti i wallet. L'utente sceglie il wallet da eliminare e clicca sull'icona di eliminazione a fianco al nome del wallet.
4. La piattaforma crea un menu pop up che richiede la conferma. L'utente preme il pulsante di conferma.
5. Viene eseguita una richiesta al database per l'eliminazione della wallet.
6. Viene automaticamente selezionata la prima wallet dell'elenco delle wallet rimaste.

Descrizione Use Case "Visualizza saldo"

Titolo: Visualizza saldo.

Riassunto: questo use case delinea i modi con cui avviene la visualizzazione dell'informazione riguardante il saldo che l'utente ha registrato complessivamente in piattaforma, tenendo conto di tutta la liquidità e le passività da esso inserite dell'utente (vedi **RF 6**).

Descrizione:

1. Una volta effettuato l'accesso al sistema, l'utente viene reindirizzato nella schermata **Dashboard**.
2. Viene eseguita una richiesta al database per la ricezione dell'informazione sul saldo registrato.
3. La pagina Dashboard mostrerà, tra le altre cose, in sovraimpressione il saldo totale che risulta attualmente registrato in piattaforma dall'utente da quando si è registrato.

Descrizione Use Case "Visualizza entrate/uscite"

Titolo: Visualizza entrate/uscite.

Riassunto: questo use case delinea i modi con cui avviene la visualizzazione dell'informazione riguardante le entrate/uscite finanziarie totali dell'utente che ha registrato dal momento di iscrizione alla piattaforma dell'utente (vedi **RF 13**).

Descrizione:

1. Una volta effettuato l'accesso al sistema, l'utente viene reindirizzato nella schermata **Dashboard**.
2. Viene eseguita una richiesta al database per la ricezione dell'informazione sulle transazioni avvenute.

3. Verranno eseguite delle operazioni di aggregazione delle transazioni per il calcolo di somme parziali ([vedi extension 1](#)).
4. L'utente registrato avrà la possibilità di visualizzare le entrate/uscite finanziarie mediante delle card mostranti i valori calcolati.

Extension:

1. L'informazione sulle entrate/uscite si aggiornerà in base al periodo selezionato sulla schermata (1D, 1M, 3M, 6M, 1Y).

Descrizione Use Case "Visualizza spese per categoria"

Titolo: Visualizza spese per categoria.

Riassunto: questo use case descrive i modi con cui avviene la visualizzazione dell'informazione riguardante le spese suddivise per categoria dal conto registrato dell'utente in piattaforma (vedi **RF 17**).

Descrizione:

1. Una volta effettuato l'accesso al sistema, l'utente viene reindirizzato nella schermata [Dashboard](#).
2. Viene eseguita una richiesta al database per la ricezione dell'informazione sulle transazioni avvenute.
3. L'utente sceglie una o più categorie di selezione per il filtraggio dei dati che verranno da lui visualizzati.
4. Verranno eseguite delle operazioni di filtraggio e di aggregazione per categoria in modo da calcolare le somme parziali ([vedi extension 1](#)).
5. L'utente registrato avrà la possibilità di visualizzare le spese suddivise per categorie, in base a quanto da lui espresso nei passaggi precedenti, mediante delle card mostranti i valori calcolati per ogni categoria.
6. Il processo viene iterato finché l'utente non decide di interrompere la visualizzazione.

Extension:

1. L'informazione sulle spese si aggiornerà in base al periodo selezionato sulla schermata (1D, 1M, 3M, 6M, 1Y), oltre che alla categoria desiderata.

Descrizione Use Case "Visualizza transazioni"

Titolo: Visualizza transazioni.

Riassunto: questo use case specifica come avverrà la visualizzazione dell'informazione riguardante tutte le transazioni effettuate dall'utente sulla wallet selezionata (vedi **RF 16**, **RF 17**).

Descrizione:

1. Una volta effettuato l'accesso al sistema, l'utente viene reindirizzato nella schermata [Dashboard](#).
2. L'utente accede alla pagina delle transazioni mediante il bottone nel menu di sinistra, cliccando la voce [Transactions](#).

3. Viene eseguita una richiesta al database per la ricezione dell'informazione su tutte le transazioni effettuate dall'utente.
4. Il sistema a questo punto mostrerà un elenco delle transazioni avvenute attraverso un'adeguata paginazione.

Extension:

1. Si ha la possibilità di filtrare le transazioni in base al periodo selezionato sulla schermata (1D, 1M, 3M, 6M, 1Y).

Descrizione Use Case "Visualizza ultime transazioni"

Titolo: Visualizza ultime transazioni.

Riassunto: questo use case descrive i modi con cui avviene la visualizzazione dell'informazione riguardante le ultime transazioni effettuate dall'utente nel periodo desiderato, ordinando quindi il tutto in ordine cronologico (vedi **RF 16**, **RF 17**).

Descrizione:

1. Una volta effettuato l'accesso al sistema, l'utente viene reindirizzato nella schermata [Dashboard](#).
2. Viene eseguita una richiesta al database per la ricezione dell'informazione sulle ultime transazioni effettuate dall'utente ([vedi extension 1](#)).
3. Il sistema a questo punto mostrerà il risultato su una card contenente l'elenco delle ultime transazioni effettuate in base al periodo selezionato.

Extension:

1. Si ha la possibilità di filtrare le transazioni in base al periodo selezionato sulla schermata (1D, 1M, 3M, 6M, 1Y).

Descrizione Use Case "Applica filtro"

Titolo: Applica filtro.

Riassunto: questo use case descrive come l'utente applica dei filtri all'elenco delle transazioni, per mostrare solo quelli di reale interesse (vedi **RF 16**, **RF 17**).

Descrizione:

1. Una volta effettuato l'accesso al sistema, l'utente viene reindirizzato nella schermata [Dashboard](#).
2. L'utente accede alla pagina delle transazioni mediante il bottone nel menu di sinistra, cliccando la voce [Transactions](#).
3. L'utente applica diversi parametri (categorie, tipologia transazione (spesa o entrata), periodo temporale (1D, 1M, 3M, 6M, 1Y) per visualizzare solo le transazioni che soddisfano tali parametri.
4. Il sistema ricalcola la lista delle transazioni e mostra solo le transazioni che soddisfano tali parametri.

Descrizione Use Case "Ordina transazioni"

Titolo: Ordina transazioni.

Riassunto: questo use case descrive come l'utente applica l'ordinamento per visualizzare le transazioni in un ordine specifico (vedi **RF 16**, **RF 17**).

Descrizione:

1. Una volta effettuato l'accesso al sistema, l'utente viene reindirizzato nella schermata [Dashboard](#).
2. L'utente accede alla pagina delle transazioni mediante il bottone nel menu di sinistra, cliccando la voce [Transactions](#).
3. L'utente applica uno dei parametri (data crescente, data decrescente, prezzo crescente, prezzo decrescente) per visualizzare le transazioni nell'ordine selezionato.
4. Il sistema riordina la lista delle transazioni e lo mostra secondo l'ordine scelto.

Descrizione Use Case "Visualizza ultime spese"

Titolo: Visualizza spese.

Riassunto: questo use case specifica i modi con cui avviene la visualizzazione dell'informazione riguardante le spese per una determinata categoria che l'utente ha effettuato dalla liquidità registrata in piattaforma (vedi **RF 7**).

Descrizione:

1. Una volta effettuato l'accesso al sistema, l'utente viene reindirizzato nella schermata [Dashboard](#).
2. Viene eseguita una richiesta al database per la ricezione dell'informazione sulle ultime spese ([vedi extension 1](#)).
3. L'utente ha la possibilità di visualizzare le spese suddivise per categorie mediante delle cards.

Extension:

1. L'informazione sulle ultime spese si aggiornerà in base al periodo selezionato sulla schermata (1D, 1M, 3M, 6M, 1Y).

Descrizione Use Case "Aggiungi transazione"

Titolo: Aggiungi transazione.

Riassunto: questo use case descrive i modi con cui avviene la registrazione di una nuova transazione sulla wallet selezionata (vedi **RF 13**).

Descrizione:

1. Una volta effettuato l'accesso al sistema, l'utente viene reindirizzato nella schermata [Dashboard](#).
2. L'utente accede alla pagina delle transazioni mediante il bottone nel menu di sinistra, cliccando la voce [Transactions](#).
3. L'utente clicca sul bottone [Add transaction](#).
4. Il sistema mostra una schermata popup.
5. Il sistema chiede all'utente di inserire i seguenti dati:

- Wallet, su cui avverrà la transazione.
 - Type, il tipo di transazione (e.g. entrata, uscita, investimento).
 - Import/amount, il valore della transazione (e.g. 150€).
 - Categoria, la categoria a cui sarà associata la spesa (vedi [exception 1](#)).
 - Description/notes, descrizione della transazione da effettuare.
6. L'utente inserisce i dati sopra indicati.
 7. L'utente clicca sul bottone [Add transaction](#) nel popup (vedi [extension 1](#)).
 8. Viene eseguita una richiesta al database per registrare la transazione avvenuta.
 9. Le modifiche applicate dall'utente avranno ripercussioni immediate sui dati visualizzabili nella [Dashboard](#).

Exception:

1. In caso di investimento non sarà prevista alcuna categoria.

Descrizione Use Case "Modifica transazione"

Titolo: Modifica transazione.

Riassunto: questo use case descrive i modi con cui avviene la modifica di una transazione sulla wallet selezionata (vedi **RF 19**).

Descrizione:

1. Una volta effettuato l'accesso al sistema, l'utente viene reindirizzato nella schermata [Dashboard](#).
2. L'utente accede alla pagina delle transazioni mediante il bottone nel menu di sinistra, cliccando la voce [Transactions](#).
3. L'utente visualizza la lista delle varie transazioni, sceglie quella che desidera modificare, e clicca sul bottone con l'icona [Edit](#) ad essa corrispondente.
4. L'utente modifica i dati mediante il form che si apre, e poi clicca il bottone [Confirm](#).
5. Viene eseguita una richiesta al database per registrare le modifiche della transazione.
6. Le modifiche applicate dall'utente avranno ripercussioni immediate sui dati visualizzabili nella [Dashboard](#).

Descrizione Use Case "Elimina transazione"

Titolo: Elimina transazione.

Riassunto: questo use case descrive i modi con cui avviene l'eliminazione di una transazione sulla wallet selezionata (vedi RF 19).

Descrizione:

1. Una volta effettuato l'accesso al sistema, l'utente viene reindirizzato nella schermata [Dashboard](#).
2. L'utente accede alla pagina delle transazioni mediante il bottone nel menu di sinistra, cliccando la voce [Transactions](#).
3. L'utente visualizza la lista delle varie transazioni, sceglie quella che desidera eliminare, e clicca sul bottone con l'icona [Cestino](#) ad essa corrispondente.
4. L'utente clicca il pulsante [Confirm](#) sul popup di conferma apertosi in piattaforma.

5. Viene eseguita una richiesta al database per registrare l'eliminazione della transazione.
6. Le modifiche applicate dall'utente avranno ripercussioni immediate sui dati visualizzabili nella [Dashboard](#).

Descrizione Use Case "Aggiungi categoria"

Titolo: Aggiungi Categoria.

Riassunto: questo use case espone le modalità attraverso le quali un utente può generare una nuova categoria di spesa all'interno del proprio account, migliorando la classificazione delle transazioni in uscita (vedi **RF 8**).

Descrizione:

1. Una volta effettuato l'accesso al sistema, l'utente viene reindirizzato nella schermata [Dashboard](#).
2. L'utente accede alla pagina delle categorie mediante il bottone nel menu di sinistra, cliccando la voce [Categories](#).
3. Si apre un menu pop up con tutte le correnti categorie esistenti nell'account. L'utente preme il tasto [Add category](#).
4. L'utente inserisce il nome della categoria, e clicca [Confirm](#).
5. Viene eseguita una richiesta al database per registrare l'aggiunta della nuova categoria.
6. Le modifiche applicate dall'utente avranno ripercussioni immediate sui dati visualizzabili nella [Dashboard](#).

Descrizione Use Case "Elimina categoria"

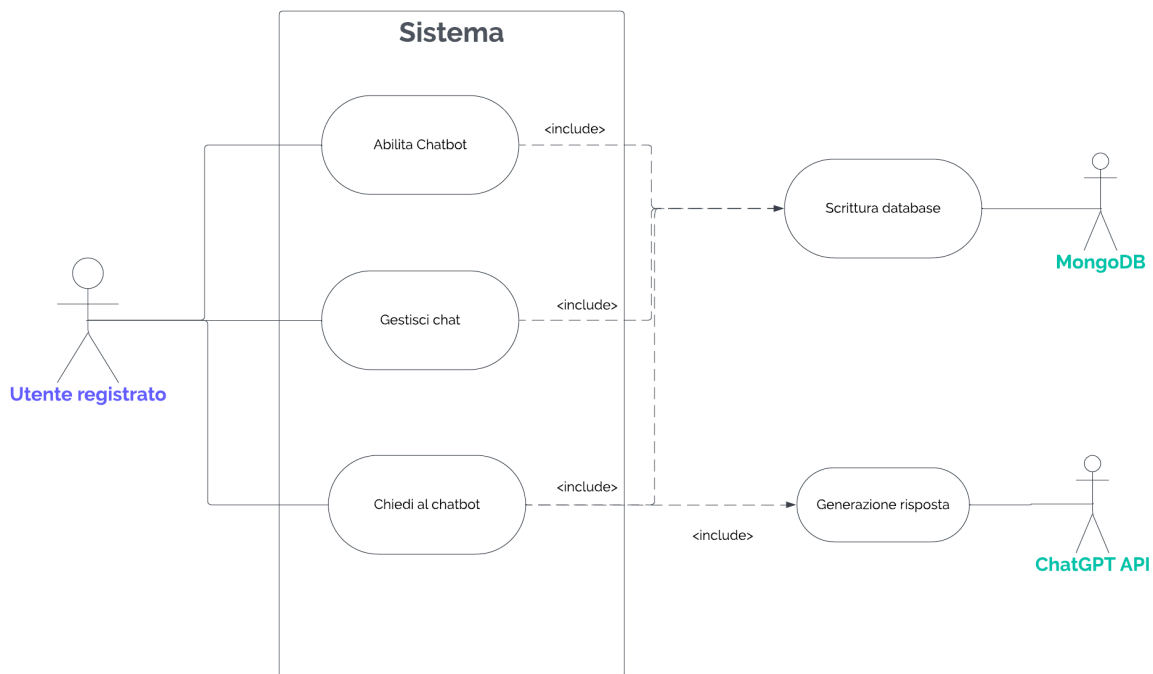
Titolo: Elimina Categoria.

Riassunto: questo use case delinea i modi con cui un utente potrà eliminare una categoria di spesa all'interno del proprio account (vedi **RF 9**).

Descrizione:

1. Una volta effettuato l'accesso al sistema, l'utente viene reindirizzato nella schermata [Dashboard](#).
2. L'utente accede alla pagina delle categorie mediante il bottone nel menu di sinistra, cliccando la voce [Categories](#).
3. Si apre un menu pop up con tutte le correnti categorie esistenti nell'Account. L'utente preme il tasto con l'icona di eliminazione all'interno della card della categoria.
4. La piattaforma crea un menu pop up che richiede la conferma. L'utente preme [Confirm](#).
5. Viene eseguita una richiesta al database per registrare l'eliminazione della categoria.
6. Le modifiche applicate dall'utente avranno ripercussioni immediate sui dati visualizzabili nella [Dashboard](#).

Chatbot



Descrizione Use Case "Abilita chatbot"

Titolo: Abilita chatbot.

Riassunto: questo use case spiega come l'utente dovrà comportarsi qualora volesse attivare (o disattivare) la funzionalità di chat all'interno della suddetta piattaforma (vedi **RF 20**).

Descrizione:

1. Una volta effettuato l'accesso al sistema, l'utente avrà la possibilità di cliccare il pulsante **Settings** (oppure l'icona con l'ingranaggio, in base alla tipologia di dispositivo su cui si trova) per accedere alle impostazioni.
2. Qui ci troverà una voce **Chatbot**, con un bottone di toggle, che permetterà di attivare o disattivare la funzionalità.
3. L'utente potrà attivare o meno il bottone, per garantire la funzionalità del tool nella piattaforma (vedi **extension 1**).

Extension:

1. La prima volta che l'utente richiede l'attivazione, gli viene richiesto di dare disponibilità relativamente alla condivisione dei propri dati con il chatbot, atti a garantire un migliore servizio.

Descrizione Use Case "Chiedi al chatbot"

Titolo: Chiedi chatbot.

Riassunto: questo use case spiega come l'utente potrà interagire con il chatbot una volta abilitato (vedi **RF 21 e RF 22**).

Descrizione:

1. Una volta effettuato l'accesso al sistema, l'utente che ha già attivato la funzionalità chatbot (vedi [Abilita chatbot](#)), potrà entrare nella pagina [Chatbot](#) mediante l'apposito bottone presente nella [Dashboard](#).
2. Da qui gli verrà presentata una chat vuota, e una riga di input per poter scrivere la sua query al [Chatbot](#).
3. L'utente scrive dunque la domanda, specificando contesto e risultato desiderato, preme [Invio](#), oppure l'apposito bottone [Invio](#) sull'interfaccia.
4. Il chatbot provvederà a questo punto a fornire la sua risposta in formato testuale nella storia della chat corrente.
5. L'utente potrà poi reiterare la domanda fatta, aggiungendo testo. Il processo di reiterazione della domanda sarà una ripetizione dei passi dal 2 al 4, fino a che l'utente sarà soddisfatto.

Descrizione Use Case "Gestisci chat"

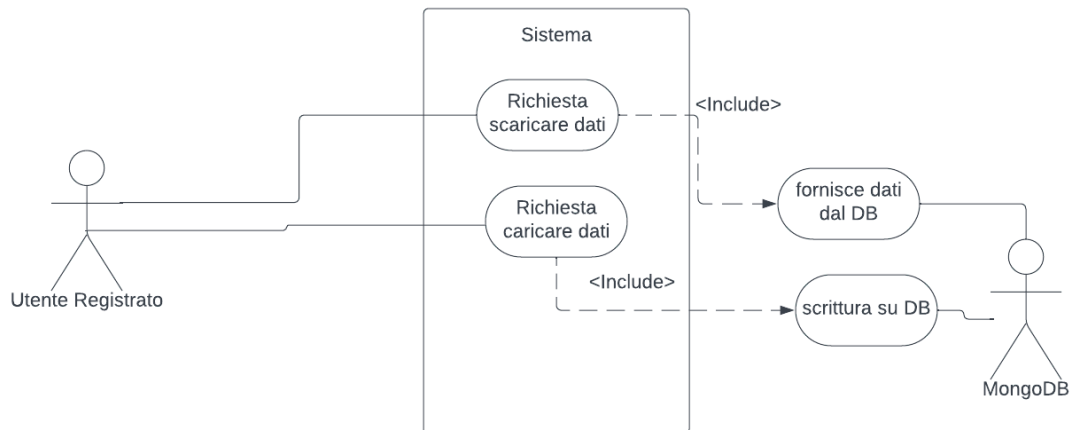
Titolo: Gestisci chat.

Riassunto: questo use case spiega come l'utente potrà gestire, quindi rivisitare e cancellare vecchie chat avute con il Chatbot della piattaforma (vedi **RF 23**).

Descrizione:

1. Una volta effettuato l'accesso al sistema, l'utente che ha già attivato la funzionalità chatbot (vedi [Abilita chatbot](#)), potrà entrare nella pagina [Chatbot](#) mediante l'apposito bottone presente nella [Dashboard](#).
2. Nella colonna di sinistra della nuova chat al centro della schermata, ci saranno elencate tutte le passate conversazioni, ordinate per data, dalla più recente a quella meno.
3. L'utente, cliccando su una di esse, potrà visualizzare una delle conversazioni avute con il Chatbot, per rivisitare le vecchie risposte.
4. L'utente potrà cancellare le vecchie conversazioni cliccando sul pulsante con l'icona di eliminazione, posta all'interno del bottone che identifica ogni chat passata.

Gestione Dati



Descrizione Use Case "Esporta dati"

Titolo: Esporta dati

Riassunto: questo use case spiega come un utente si muoverà qualora dovesse voler esportare i propri dati al di fuori della piattaforma, in un formato universale (vedi **RF 24 e RF 25**).

Descrizione:

- Una volta effettuato l'accesso al sistema, l'utente avrà la possibilità di accedere alle impostazioni, cliccando il pulsante **Settings** (oppure l'icona con l'ingranaggio, in base alla tipologia di dispositivo su cui si trova).
- Qui ci troverà una voce **Esporta**, con un apposito bottone con scritto **Esporta Dati in CSV**.
- L'utente clicca sul suddetto bottone.
- La piattaforma allora chiederà all'utente che dati desidera esportare. Le possibilità sono:
 - Tutti i dati in piattaforma.
 - I dati dopo una certa data (inserita dall'utente sul momento) (vedi **exception 1**)
 - I dati relativi ad una categoria specifica registrata (solo spese, solo investimenti...) (vedi **exception 2**).
- L'utente effettuerà dunque la sua scelta.
- La piattaforma provvederà a scaricare nella memoria locale del dispositivo dell'utente un file **csv**, popolato con i dati richiesti.

Exception:

- Nel caso in cui la data selezionata si trovi nel futuro, la piattaforma non procederà nella compilazione del file, ma avviserà l'utente che non ci sono dati disponibili per tale filtro.

2. Nel caso in cui la categoria selezionata sia vuota o senza dati, la piattaforma non procederà nella compilazione del file, ma avviserà l'utente che non ci sono dati disponibili per tale filtro.

Descrizione Use Case "Importa dati"

Titolo: Importa dati

Riassunto: questo use case spiega come un utente potrà importare i propri dati all'interno della piattaforma, per popolarla automaticamente (vedi **RF 26**).

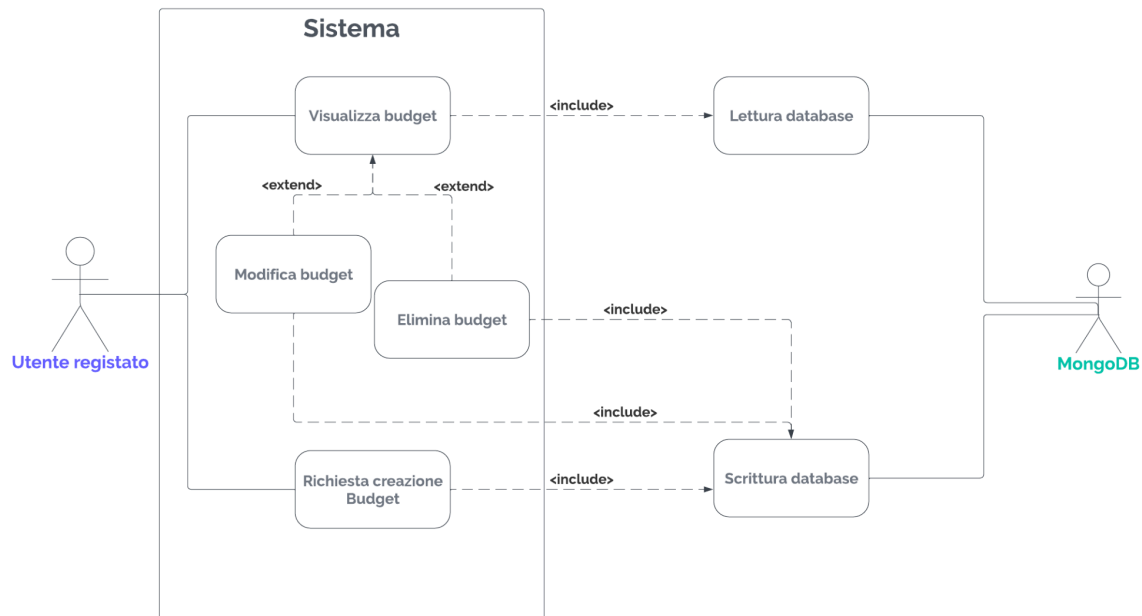
Descrizione:

1. Una volta effettuato l'accesso al sistema, l'utente avrà la possibilità di accedere alle impostazioni, cliccando il pulsante [Settings](#) (oppure l'icona con l'ingranaggio, in base alla tipologia di dispositivo su cui si trova).
2. Qui ci troverà una voce [Importa](#), con un apposito bottone con scritto [Importa CSV](#), e dovrà cliccare su di esso.
3. La piattaforma chiederà di caricare dalla memoria locale un file [csv](#) contenente i dati che l'utente vuole importare. L'utente selezionerà il file [csv](#) desiderato.
4. Il file viene caricato in piattaforma e letto dalla stessa ([vedi exception 1](#)), e i dati verranno poi aggiunti a quelli già presenti nella piattaforma.
5. Visitando la propria [Dashboard](#), l'utente potrà vedere la versione aggiornata dei propri dati.

Exception:

1. Nel caso in cui la struttura dei dati contenuta nel file non sia conforme a quella che la piattaforma si aspetta, verrà fornito un messaggio di errore in linguaggio naturale, che spiegherà a che linea del file si trova il problema, provando a dare suggerimenti su come sistemarlo.

Budget



Descrizione Use Case "Crea Budget"

Titolo: Crea budget

Riassunto: questo use case spiega come un utente potrà creare un budget di spesa nella piattaforma, corrispondente al suo account (vedi **RF 27**).

Descrizione:

1. Una volta effettuato l'accesso al sistema, l'utente viene reindirizzato nella schermata [Dashboard](#).
2. Nel menu di sinistra, clicca sul bottone chiamato [Budget](#), per entrare nella pagina dedicata e visualizzare tutti i budget già esistenti.
3. L'utente clicca il bottone [Crea Budget](#).
4. La piattaforma apre un form che richiede all'utente la categoria di spesa di cui si vuole tenere traccia, il budget massimo, il budget corrente che ci si pone e il periodo del budget (settimanale o mensile).
5. L'utente inserisce i dati e clicca [Conferma](#) (vedi [exception 1](#)).

Exception:

1. Nel caso in cui l'utente abbia lasciato un campo vuoto, la piattaforma provvederà a richiedere esplicitamente all'utente di completare tale sezione.

Descrizione Use Case "Visualizza Budget"

Titolo: Visualizza budget

Riassunto: questo use case spiega come un utente potrà visualizzare i propri budget di spesa nella piattaforma (vedi **RF 28**).

Descrizione:

1. Una volta effettuato l'accesso al sistema, l'utente viene reindirizzato nella schermata **Dashboard**.
2. Nel menu di sinistra, clicca sul bottone chiamato **Budget**, per entrare nella pagina dedicata e visualizzare tutti i budget già esistenti.
3. L'utente ora visualizza nella pagina i suoi budget, con la percentuale a cui è arrivato nella sessione attuale (settimanale o mensile)

Descrizione Use Case "Modifica Budget"

Titolo: Modifica budget

Riassunto: questo use case spiega come un utente potrà modificare un proprio budget di spesa nella piattaforma (vedi **RF 29**).

Descrizione:

1. Una volta effettuato l'accesso al sistema, l'utente viene reindirizzato nella schermata **Dashboard**.
2. Nel menu di sinistra, clicca sul bottone chiamato **Budget**, per entrare nella pagina dedicata e visualizzare tutti i budget già esistenti.
3. L'utente clicca sull'icona **Edit** del budget che intende modificare.
4. L'utente inserisce i nuovi dati all'interno della form pop up che si apre, e preme **Confirm** (vedi exception 1).

Exception:

1. Nel caso in cui l'utente abbia lasciato un campo vuoto, la piattaforma provvederà a richiedere esplicitamente all'utente di completare tale sezione.

Descrizione Use Case "Elimina Budget"

Titolo: Elimina budget

Riassunto: questo use case spiega come l'utente potrà eliminare un proprio budget di spesa nella piattaforma (vedi **RF 30**).

Descrizione:

1. Una volta effettuato l'accesso al sistema, l'utente viene reindirizzato nella schermata **Dashboard**.
2. Nel menu di sinistra, clicca sul bottone chiamato **Budget**, per entrare nella pagina dedicata e visualizzare tutti i budget già esistenti.
3. L'utente clicca sull'icona dell'eliminazione del budget che intende modificare.
4. L'utente clicca **Confirm** nel menu pop up che richiede la conferma dell'eliminazione del Budget in questione.

Requisiti non funzionali

In questa sezione del documento riportiamo tutti i requisiti non funzionali per Money Expense in una tabella contenente **titolo del requisito**, allegato ad una [descrizione](#) ed una [misura](#) deterministica che ne determini l'attuazione con successo da parte della piattaforma.

Sicurezza

| Titolo | Descrizione | Misura |
|-------------------------------|---|--|
| Password forte | La password deve garantire un elevato livello di sicurezza e resistenza. Questo per permettere all'utente di avere maggiore certezza che il suo profilo non venga utilizzato da terzi. | La password deve essere composta da un minimo di 8 caratteri, inclusi almeno una lettera maiuscola, una lettera minuscola, un numero e uno dei seguenti caratteri speciali: !, ", £, \$, %, &, ?, €, ^, o =. Il sistema, oltre a obbligare questi limiti, incoraggia l'utente con un apposito banner ad evitare di utilizzare parole di senso compiuto, per rendere la password meno indovinabile da utenti terzi. |
| Cambio password sicuro | La procedura di modifica della password deve essere eseguita in modo sicuro e conforme a una strutturata e rigorosa procedura, al fine di garantire la sicurezza di questo processo. | È necessario inserire la password precedente due volte prima di procedere con la scelta della nuova password. La nuova password deve rispettare i requisiti specificati precedentemente. |
| Blocco account | Ostacolare con tutti i mezzi possibili l'accesso all'account da parte di utenti che non sono i reali creatori di quell'account, al fine di garantire la privacy. | Dopo 5 tentativi di accesso con password errata da parte di un utente, l'account risulterà bloccato e non sarà più possibile tentare di inserire la password. La piattaforma manderà un'email al mail account, contenente un link per ripristinare la password e riguadagnare l'accesso al servizio. |
| Autenticazione a due fattori. | La piattaforma provvede facoltativamente un layer ulteriore di sicurezza, mediante un altro fattore di autenticazione. Questo per rendere ulteriormente complicato per utenti non possessori dell'account di accedervi in maniera illecita. | Nel processo di accesso, l'utente ha l'opzione di inserire un codice di conferma a 6 cifre dopo aver fornito le sue credenziali. Questo codice viene inviato al numero di telefono indicato dall'utente solo se ha abilitato questa funzionalità nel suo account. |
| Durata sessione | La sessione di accesso al sito deve | Dopo 12 ore di inattività, il token di |

| | | |
|------------|---|---|
| d'accesso. | avere una scadenza, in modo da ostacolare l'accesso da dispositivi già autenticati da parte di terzi, con tecniche di social engineering. | accesso generato durante il login di un utente diventerà inutilizzabile e sarà necessario effettuare un nuovo login nella piattaforma per accedere alle funzionalità riservate agli utenti autenticati. |
|------------|---|---|

Privacy e informazioni personali

| Titolo | Descrizione | Misura |
|----------------------------------|---|---|
| Verifica dell'identità. | La piattaforma verifica l'identità dei propri utenti, e si impegna a non divulgare a terze parti i dati che questi gli forniscono. | L'identificazione avviene in seduta di Login, con l'inserimento dei propri dati e la password, che deve essere posseduta solo dall'utente registrato. La piattaforma non inoltra a nessun altro sistema alcun dato dell'utente senza la diretta approvazione. |
| Regolamentazione protezione dati | La piattaforma si impegna a rispettare le norme standard sulla protezione dei dati dei propri utenti, seguendo le normative proposte dall'Unione Europea, precisamente il regolamento n. 2016/679 in materia del trattamento, noto come GDPR. | Il sistema deve essere conforme al GDPR nel CAPO II, CAPO III (riguardante i diritti dell'interessato) e nel CAPO IV (riguardante il titolare del trattamento), garantendo con trattamento dei dati personali e di particolare attenzione i seguenti articoli: articoli 5, 6 e 7 (riguardanti i principi del trattamento dei dati e il consenso dell'interessato), articolo 15 (l'utente ha diritto di ottenere informazioni relative ai dati conferiti e alle modalità d'uso di essi), articolo 16 (Diritto di rettifica, ovvero l'utente ha il diritto di ottenere la rettifica dei dati personali inesatti), articolo 17 (Diritto all'oblio, ovvero l'utente ha il diritto di ottenere la cancellazione dei dati personali e di ottenere comunicazioni in merito, come previsto dall'articolo 19). |

Performance

| Titolo | Descrizione | Misura |
|-------------------------|---|---|
| Avvio applicazione web. | La piattaforma deve impiegare tempi ristretti per caricarsi ed avviarsi | Il login, ovvero la verifica delle credenziali, sarà effettuato in un |

| | | |
|---------------------------|---|--|
| | correttamente. | massimo di 2 secondi dall'inserimento dei dati. |
| Transizioni tra schermate | L'applicazione web deve risultare fluida e il più possibile performante nel passaggio da una schermata alla successiva. | La transizione tra schermate deve impiegare non più di 2 secondi, a meno che tra queste schermate non vi sia la dashboard o la pagina delle transazioni. In quel caso fino a 5 secondi sono accettati, a causa della mole di dati consistente che è necessario elaborare a priori. |

Portabilità

| Titolo | Descrizione | Misura |
|--|---|---|
| Design responsive multi piattaforma | La piattaforma deve essere in grado di adattarsi agilmente alle dimensioni di schermo del dispositivo su cui viene utilizzata, sia il dispositivo Desktop, Mobile o Tablet. | Il sistema deve essere adattivo, ovvero rispettare le proporzioni dei vari elementi dell'interfaccia a seconda del dispositivo in questione. È cruciale che tutti gli elementi, qualsiasi sia il dispositivo, siano visibili e fruibili dall'utente in maniera chiara e semplice. |
| Versioni supportate di Google Chrome | La piattaforma dovrà funzionare correttamente sul browser Google Chrome. | La piattaforma sarà compatibile e funzionante a partire dalla versione 97.0.4692 del 4 gennaio 2022. |
| Versioni supportate di Mozilla Firefox | La piattaforma dovrà funzionare correttamente sul browser Mozilla Firefox. | La piattaforma sarà compatibile e funzionante a partire dalla versione Firefox 96.0.3 del 27 gennaio 2022. |
| Versioni supportate di Safari | La piattaforma dovrà funzionare correttamente sul browser Safari. | La piattaforma sarà compatibile e funzionante a partire da ogni versione Safari rilasciata a partire da inizio 2022. |
| Versioni supportate di Opera | La piattaforma dovrà funzionare correttamente sul browser Opera. | La piattaforma sarà compatibile e funzionante a partire dalla versione Opera 93 del 19 gennaio 2022. |

Usabilità

| Titolo | Descrizione | Misura |
|----------------|---|--|
| Facilità d'uso | L'usabilità del sistema deve essere semplice ed intuitiva. Un utente deve poter essere in grado di usare autonomamente la piattaforma, anche se non particolarmente esperto | Per utenti che si stanno registrando nel sistema senza precedente esperienza, ci si aspetta che il processo di creazione di un profilo richieda un tempo medio non |

| | | |
|----------------------------------|--|---|
| | nell'utilizzo di tali tecnologie. | superiore a 5 minuti, mentre per l'inserimento delle informazioni finanziarie dovrebbe essere completato in circa 15 minuti. |
| Comunicazione errori all'utente. | In caso di errore o di mal utilizzo, la piattaforma deve fornire messaggi chiari e dettagliati su cosa l'utente ha fatto di sbagliato o cosa sta causando il problema posto in essere. | In situazioni in cui gli utenti inseriscono dati in modo errato, il sistema deve generare messaggi di testo informativi che dettagliano il problema, identificano le cause possibili e offrono suggerimenti per la risoluzione. L'utente deve quindi essere in grado di capire l'errore che ha commesso, se causato da lui, e agire di conseguenza. |
| Supporto iniziale e tour | Il sistema deve introdurre in maniera graduale e naturale l'utente al workflow in piattaforma. | La pagina Dashboard e la pagina Wallets devono contenere dei messaggi a comparsa sequenziali che descrivono le funzionalità, la prima volta che un utente accede al sistema. |

Robustezza

| Titolo | Descrizione | Misura |
|--|--|---|
| Tempistiche risoluzioni e manutenzioni | In caso di necessarie manutenzioni o di problemi di infrastruttura che la piattaforma dovesse affrontare, dovranno essere sistemati nel minor tempo possibile. | In caso di indisponibilità del sistema causata da problemi, si garantirà il ripristino entro un massimo di 2 ore. |
| Backup dati | È necessario garantire il salvataggio sicuro e possibilità di ripristino dati degli utenti utilizzando il sito. | Il sistema possiede copie di backup dei dati per ogni utente, che vengono aggiornati costantemente, così da evitare perdite non preventivabili. |

Capacità

| Titolo | Descrizione | Misura |
|------------------------|--|--|
| Supporto carico utenti | La piattaforma deve mantenere prestazioni buone ed accettabili anche se sottoposta a grandi traffici e se utilizzata da molti utenti contemporaneamente. | La seguente proprietà è garantita fino a 200 utenti simultaneamente collegati e che la utilizzano. |

Accessibilità

| Titolo | Descrizione | Misura |
|----------------------|--|--|
| Input in piattaforma | La piattaforma accetta diverse modalità di input da parte dei propri utenti. | È possibile navigare ed usare le funzionalità usando la tastiera, il mouse e il touch screen, qualora il proprio device lo supporti. |
| Maggiore visibilità | La piattaforma deve essere fruibile anche da utenti con eventuale deficit o disabilità visiva parziale. | L'applicazione web è conforme alla sezione 1.1.1 nelle WCAG 2.1, i contenuti dovranno essere compresi anche tramite screen-reader (JAWS for Windows e screen reader Apple) e le parti testuali del sistema possono venire ingrandite fino al 200%. |
| Interfaccia chiara | È necessario che tutte le parti dell'interfaccia siano chiaramente riconoscibili e distintamente separate. | Tutte le componenti rilevanti per il corretto funzionamento del sistema devono aderire ai requisiti di contrasto di almeno il livello AA, come specificato nella sezione 1.4.3 delle linee guida WCAG 2.1. |

Analisi del contesto

In questa sezione del documento viene effettuata l'analisi del contesto per **Money Expense**.

Verranno:

- Descritti gli attori e i sistemi esterni in forma testuale con cui la piattaforma in questione si interfacerà.
- Fornito in seguito anche un diagramma del contesto o context diagram per scendere ulteriormente nel dettaglio.

Attori

Utente anonimo: Gli utenti anonimi sono individui o entità che si serviranno del nostro sistema e avranno accesso soltanto alle capacità legate alla creazione di un account..

I percorsi di dati riguardanti l'utente non autenticato sono:

- Registrarsi (vedi requisiti funzionali **RF 1**).

Utente registrato: La seconda categoria di utenti che interagisce con il sistema è rappresentata dagli utenti registrati. Un utente anonimo diventa un utente registrato una volta completata la procedura di registrazione e aver ottenuto un profilo confermato sulla piattaforma Money Expense.

I flussi di informazione relativi all'utente autenticato sono:

- Effettuare il login (vedi **RF 4**).
- Visualizzare e modificare il proprio profilo (vedi requisiti funzionali dal **RF 5**).
- Resettare e modificare la propria password (vedi **RF 3**).
- Visualizzare i propri dati in Dashboard (vedi da **RF 6** a **RF 19**).
- Utilizzare la funzionalità chatbot (vedi **RF 20, RF 21, RF 22, RF 23**).
- Esportare i dati all'esterno dell'applicazione (vedi **RF 24, RF 25 e RF 26**).
- Utilizzare il tool di budgeting (vedi **RF 27, RF 28, RF 29 e RF 30**).

Amministratore: è la terza ed ultima tipologia di attore che utilizza il sistema.

Rappresenta l'utente che gestisce e amministra l'intero sistema, intervenendo quando si sviluppano comportamenti inadeguati in piattaforma.

Il flussi di informazione relativo all'amministratore sono:

1. Bloccare gli utenti.
2. Sbloccare gli utenti bloccati.

Sistemi esterni

Autho: è un sistema ausiliario che offre servizi per effettuare l'accesso e la registrazione degli utenti, gestendo anche funzionalità come l'autenticazione a due fattori, il ripristino e la modifica delle password, come richiesto nei requisiti **RF 1, RF 2, RF 3, RF 4, RF 5 e RF 6**. Inoltre, [Autho](#) invierà messaggi email di conferma agli utenti e riceverà da loro email e password per supportare il processo di autenticazione. Sarà di fatto incaricato del riconoscimento di ogni attore agente sulla piattaforma.

Chat GPT: sistema subordinato che fornisce API per creazione di chatbot basati su intelligenza artificiale, che consentirà la corretta funzionalità della funzione Chatbot della piattaforma. Questa è la funzionalità dove l'utente ha la possibilità di interagire con l'intelligenza artificiale per ricevere sostegno e aiuto per quanto riguarda la gestione e la pianificazione della propria situazione finanziaria (come previsto da **RF 20, RF 21, RF 22 e RF 23**).

MongoDB: un sistema sottostante, utilizzato dall'applicazione web, è responsabile di memorizzare i dati personali degli utenti, le informazioni sugli oggetti pubblicati sul sito in un determinato momento e i dettagli del profilo, inclusi portafogli (wallets), categorie di spesa e i dati salvati in essi. Questo sistema si integra con [MongoDB](#) per eseguire operazioni di lettura dei dati. Durante le operazioni di scrittura, il sistema fornisce a [MongoDB](#) i dati da scrivere o sovrascrivere, mentre, durante le operazioni di lettura, [MongoDB](#) comunica al sistema i dati richiesti.

Yahoo Finance: sistema subordinato che fornisce informazioni relativamente a strumenti finanziari scambiabili su exchange pubbliche con dati in tempo reale. Fornisce inoltre il rapporto di prezzo che sta tra due diverse valute (EUR e USD per esempio). Consente di calcolare il valore in tempo reale del patrimonio totale di un utente in una determinata valuta a sua scelta, di modo che egli possa scegliere come meglio si trova, e personalizzare al meglio l'esperienza in piattaforma (come da **RF 6 e RF 7**).

Diagramma di contesto

Per rappresentare graficamente gli attori e i sistemi esterni con cui la piattaforma Money Expense interagisce verrà utilizzato un diagramma di contesto o context diagram.

Grazie al diagramma di contesto sarà possibile:

- Visualizzare i flussi di informazione e la loro direzionalità.
- Specificare quali particolari funzionalità fornisce all'applicazione web ogni sistema esterno.
- Specificare quali informazioni fornisce l'applicazione ai vari sistemi esterni.

Nel nostro caso, l'utente anonimo può registrarsi **RF1**, per diventare quindi un utente registrato.

L'utente registrato può dunque autenticarsi mediante richiesta di accesso **RF4**, e il sistema notificherà se la richiesta è avvenuta con successo.

L'utente potrà richiedere di modificare i propri dati inseriti in piattaforma e ricevere una notifica di avvenuta modifica **RF5**.

Può occuparsi della gestione dei propri wallet **RF10**, delle proprie transazioni **RF13**, delle categorie **RF8**, delle chat **RF20** e dei propri budget **RF27**. In ognuno di questi casi, il sistema restituisce la visualizzazione dei dati richiesti.

L'utente può inoltre domandare al sistema che siano esportati i propri dati **RF24**, e in questo caso il sistema risponderà con il file [csv](#) richiesto.

Infine, l'utente amministratore si occupa di comunicare al sistema un eventuale utente da bloccare.

Abbiamo poi il sistema [Autho](#), il quale si occupa dell'autenticazione degli utenti.

[Autho](#) spedisce al nostro sistema Money Expense la conferma dell'autenticazione quando un utente svolge il login **RF4** e spedisce le informazioni necessarie all'utente richiedente per il ripristino della password **RF2**.

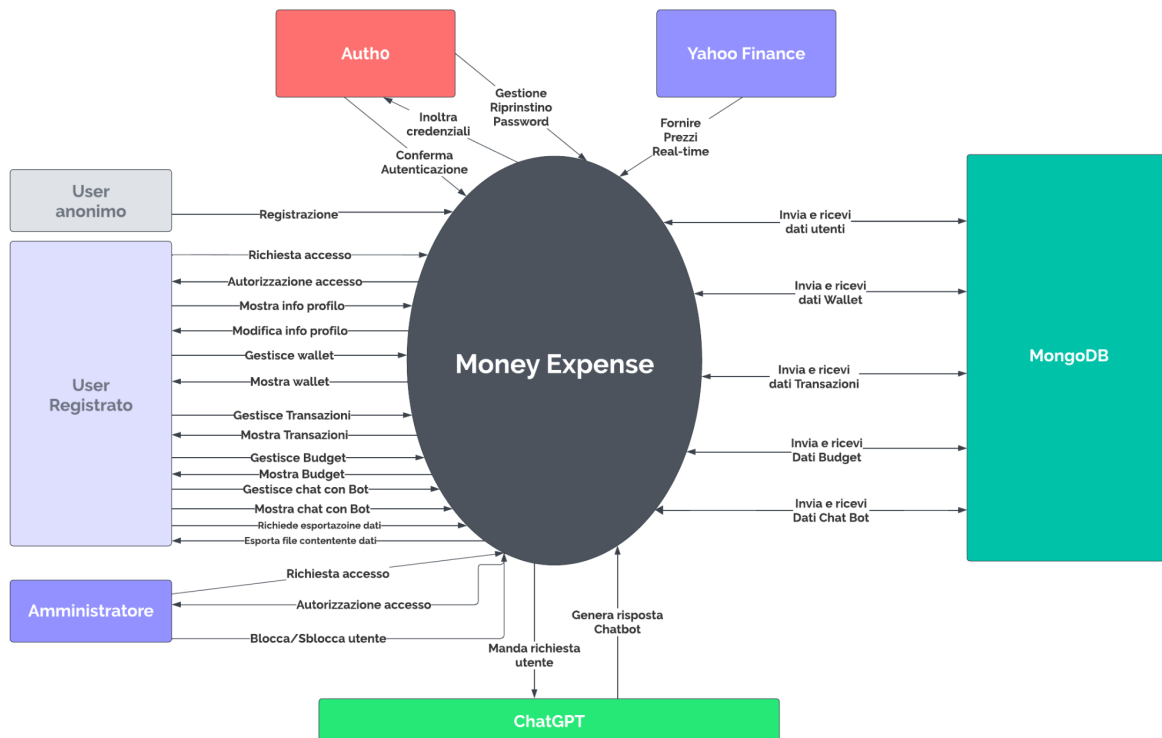
[Autho](#) riceve dal sistema le credenziali dell'utente che si vuole autenticare o registrare alla piattaforma **RF4**.

[Yahoo Finance](#) manda al nostro sistema i dati relativi ai prezzi degli investimenti che un utente ha registrato, o i dati sul cambio valutario, se questo fosse richiesto **RF6**, **RF7**, **RF10**, **RF12**.

[MongoDB](#) è il database utilizzato. Tutti i flussi ad esso sono bidirezionali, in quanto il database fornisce al sistema le informazioni richieste, ed eventualmente riceve anche dati nei momenti in cui il sistema vuole creare dei nuovi dati in piattaforma o modificare quelli esistenti.

In particolare, [MongoDB](#) si occupa di flussi di dati relativamente ai dati utente **RF1-RF5**, per le categorie **RF9**, per i wallet **RF10 RF11**, per le transazioni **RF13-RF19**, per il chatbot **RF20** e i budget **RF27-RF 30**.

ChatGPT infine è l'API che permette di avere un bot di intelligenza artificiale di sostegno all'utente. Il sistema di Money Expense fornisce al sistema esterno i messaggi dell'utente e il contesto in cui si trova **RF20**, mentre il sistema esterno fornisce risposte a tali query con frasi in linguaggio naturale, elaborate da AI **RF21 RF22**.



Analisi dei componenti

In questa sezione viene effettuata l'analisi dei componenti interni al nostro sistema. I componenti vengono definiti sulla base dei requisiti funzionali e sui sistemi esterni con cui comunica il nostro sistema.

La scelta dei componenti viene inoltre ponderata per cercare di massimizzare la coesione interna e minimizzare il grado di accoppiamento dei vari componenti (nei limiti delle possibilità ovviamente).

Definizione dei componenti

Qui descriviamo tutti i vari componenti del sistema di Money Expense, e il motivo per cui li abbiamo creati.

Gestione di registrazione

Motivazione: per l'accesso e l'utilizzo della piattaforma, è necessario essere registrati mediante un account valido **RF1**. Per questo motivo il componente **Gestione di registrazione** si occupa di interagire con il sistema esterno responsabile di registrare gli utenti (**Autho**) e permettergli di creare quindi l'account necessario ad utilizzare correttamente la piattaforma.

Coesione: livello 7 - funzionale

Spiegazione: il sistema gestisce la registrazione di un nuovo utente mediante l'inserimento di una email nuova e una password ritenuta sufficientemente robusta.

Gestione autenticazione

Motivazione: per poter accedere alla piattaforma, una volta creato un account, serve autenticarsi facendo il login in piattaforma **RF4**. Per questo **Gestione autenticazione** si occupa di verificare la correttezza delle credenziali inserite, mediante il sistema esterno, e poi concede all'utente di utilizzare la piattaforma a piacimento.

Coesione: livello 7 - funzionale

Spiegazione: il sistema autentica l'utente in piattaforma, confrontando i dati che esso provvede per l'accesso con quelli memorizzati nel database.

Gestione profilo

Motivazione: un utente che crea un account in piattaforma deve poter essere in grado di visualizzare il proprio profilo **RF5** e modificare la sua password **RF2**. Questo componente **Gestione profilo** fornisce esattamente questa possibilità, dando quindi all'utente modo di aggiornare i propri dati personali salvati in piattaforma.

Coesione: livello 7 - funzionale

Spiegazione: l'utente può richiedere di visualizzare in piattaforma i dati salvati nel profilo, e richiedere di cambiare password qualora lo desiderasse. Perché questa procedura abbia successo è necessario che l'utente prima inserisca la password corrente, così da verificare che sia l'utente corretto a chiedere questa modifica.

Gestione Dashboard

Motivazione: da **RF6** l'utente deve poter fare l'accesso alla propria dashboard in modo da poter visualizzare tutti i dati relativi a ciò che è stato salvato sulla piattaforma. Da qui avrà anche modo di effettuare tutte le azioni necessarie in caso volesse creare e/o modificare un qualsiasi dato presente.

Coesione: livello 7 - funzionale

Spiegazione: l'utente accede ai propri dati registrati, mediante l'apposita pagina. Qui trova un digest di tutte le informazioni sulla situazione patrimoniale corrente.

Gestione lettura del database

Motivazione: per poter avere a disposizione dell'utente e del sistema in generale è necessario poter accedere a tali dati che sono registrati nel database. Quindi, il componente si occupa di fare le dovute interrogazioni al database per poter poi restituire agli altri componenti ciò di cui hanno bisogno. Il componente Gestione Lettura del

database è di fatto un supporto necessario a tutti gli altri componenti, anche se non risolve direttamente alcun requisito funzionale.

Coesione: livello 4 - procedurale

Spiegazione: altri componenti facenti parte del sistema chiedono informazioni salvate nel database, e il componente **Gestione lettura del database** fornisce quanto richiesto. Per esempio, il componente **Gestione transazioni** richiede informazioni su tutte le transazioni di un utente, e il componente **Gestione lettura del database** si occupa di fornirle.

Gestione scrittura del database

Motivazione: quando un utente vuole aggiungere o modificare dati al database, che siano essi per categorie, wallet, transazioni, budget o chat, questo deve avvenire mediante un apposito componente che si occupa delle interazioni in scrittura con [MongoDB](#) come sistema esterno. Per questo il componente **Gestione scrittura del database** è stato creato ed utilizzato in questo documento.

Coesione: livello 4 - procedurale

Spiegazione: il componente riceve i dati da altri componenti che richiedono la modifica, la creazione o l'eliminazione di un'istanza memorizzata nel database, e questo componente viene incaricato di gestire la richiesta. Per esempio, il componente **Gestione wallet** richiede la creazione di un wallet "Banca XYZ", e il componente **Gestione scrittura del database** si interfaccia con il database affinché ciò avvenga.

Gestione risultati

Motivazione: l'utente deve poter eseguire delle ricerche nei propri dati per categorie, e verificare quindi il suo stato attuale, come da **RF7**. Per questo il componente **Gestione risultati** si occupa di gestire questi dati.

Coesione: livello 7 - funzionale

Spiegazione: il componente riceve la categoria di interesse, e poi si interfaccia con il componente **Gestione lettura del database** per poter avere i dati relativi all'utente di interesse per quella query specifica.

Gestione chatbot

Motivazione: il **RF20** afferma che l'utente avrà a disposizione un chatbot che gli permetterà di fare domande e ricevere sostegno per la propria vita finanziaria. Questo componente fornisce questa funzionalità all'utente finale.

Coesione: livello 7 - funzionale

Spiegazione: l'utente interagisce con il componente, fornendo il messaggio che vuole sottoporre al bot. Il componente **Gestione chatbot** si occupa di prendere in carico

questo messaggio e comunicarlo al sistema esterno di API di chat GPT, per poi restituire all'utente ciò che è stato ritornato.

Gestione transazioni

Motivazione: Da **RF13** l'utente deve poter salvare in piattaforma i movimenti di denaro che fa quotidianamente. Per questo il componente **Gestione transazioni** viene creato, in modo che quindi ci sia la possibilità di gestire tali casistiche.

Coesione: livello 7 - funzionale

Spiegazione: l'utente o altri componenti richiedono accesso o modifica ai dati presenti in piattaforma relativi a determinate transazioni e il componente **Gestione transazioni** viene incaricato di gestire tali richieste al loro posto. Questo componente si interfaccia anche con il sistema esterno [Yahoo finance](#) per avere i prezzi delle valute in tempo reale, qualora ce ne fosse la necessità.

Gestione categorie

Motivazione: da **RF9** l'utente deve avere la possibilità di creare categorie, che permettono a lui di meglio strutturare i fondi salvati in piattaforma. Per questo il componente **Gestione categorie** viene creato. Si occuperà in particolare della gestione, creazione ed eliminazione di ogni categoria per l'utente o per altri componenti.

Coesione: livello 7 - funzionale

Spiegazione: l'utente o altri componenti richiedono accesso o modifica ai dati presenti in piattaforma relativi a determinate categorie e il componente **Gestione categorie** viene incaricato di gestire tali richieste al loro posto.

Gestione wallet

Motivazione: per soddisfare il **RF10**, nel quale si afferma che ogni utente deve poter registrare in piattaforma le proprie fonti di liquidità, tale componente viene creato, così che sia in grado di gestire creazioni e modifiche delle varie fonti inserite dall'utente.

Coesione: livello 7 - funzionale

Spiegazione: l'utente o altri componenti richiedono accesso o modifica ai dati presenti in piattaforma relativi a determinati wallet e il componente **Gestione wallet** viene incaricato di gestire tali richieste al loro posto.

Gestione budget

Motivazione: il **RF27** chiede la possibilità di creare i budget in piattaforma da parte degli utenti. Questo componente si occupa specificamente di questo compito, gestendo la loro creazione, modifica ed eliminazione, come richiesto da **RF 29** e **RF 30**.

Coesione: livello 7 - funzionale

Spiegazione: l'utente o altri componenti richiedono accesso o modifica ai dati presenti in piattaforma relativi a determinati budget, e il componente **Gestione budget** viene incaricato di gestire tali richieste al loro posto.

Diagramma dei componenti

Gestione di registrazione

Questo elemento avrà l'incarico di interagire direttamente con gli utenti, facilitando la registrazione sulla piattaforma e la creazione di un account valido.

Interfacce richieste:

- **Creazione utente:** il componente interagisce con il componente **Gestione scrittura del database** per salvare i dati dell'utente che si sta registrando all'interno del database.
- **Registrazione per autenticazione:** il componente interagisce con il componente esterno **Autho** per salvare l'utente che poi successivamente potrà essere autenticato.

Interfacce fornite

- **Registrazione:** il componente registra l'utente all'interno del database per poter poi in futuro estrarre i suoi dati per autenticarlo in piattaforma.
- **Conferma registrazione:** il sistema fornisce una risposta all'utente della avvenuta registrazione.

Gestione autenticazione

Questo componente funge da tramite tra l'utente e il sistema esterno di supporto **Autho**, il quale gestisce l'autenticazione. Si posiziona come un'interfaccia tra l'applicazione menzionata e l'utente che intende accedere alla piattaforma, fungendo da ponte tra l'ambiente interno e quello esterno della piattaforma.

Interfacce richieste

- **Richiesta modifica password:** il componente riceve dalla **Gestione profilo** la richiesta di cambiamento della password per uno specifico utente; questa richiesta verrà inoltrata ad **Autho** che effettuerà le necessarie procedure.

Interfacce fornite

- **Autenticazione:** il componente fornisce al sistema esterno di **Autho** email e password dell'utente ricevuti da altri componenti.
- **Inoltra richiesta modifica password:** il componente inoltra ad **Autho** la richiesta dell'utente di modificare la password. Sarà compito poi di **Autho** svolgere tutte le operazioni di modifica e notificare l'utente.

- **Conferma autenticazione:** il componente, dopo aver autenticato l'utente mediante **Autho**, restituisce la conferma dell'avvenuta autenticazione in piattaforma.

Gestione profilo

Questo elemento è responsabile di visualizzare le informazioni del profilo dell'utente ottenute tramite una richiesta di lettura al database.

Espone tutte le informazioni personali di un particolare utente e gestisce le diverse richieste ad esse relative, specialmente riguardo alla modifica dello stesso.

Interfacce richieste

- **Informazioni profilo:** il componente riceve i dati dell'utente e le informazioni del profilo sopra descritte dal componente di Gestione lettura dal database dopo che li ha esplicitamente richiesti al database.
- **Modifica profilo:** il componente riceve dall'utente i nuovi dati personali che si intendono modificare, se si intende aggiornare il proprio profilo personale.
- **Modifica password:** il componente invia al componente di Gestione autenticazione la richiesta da parte dell'utente di modifica della password, gestita da altri componenti.

Interfacce fornite

- **Informazioni utente:** il componente invia al componente Gestione lettura dal database la richiesta di lettura delle informazioni e dei dati del profilo dell'utente, inviando in concomitanza la email da cui ricevere le informazioni.
- **Modifica utente:** il componente inoltra al componente di Gestione scrittura sul database i nuovi dati inseriti dall'utente che andranno a modificare i vecchi.
- **Richiesta modifica password:** il componente si occupa di gestire la modifica della password di un utente.

Gestione Dashboard

Questo elemento gestisce la dashboard dell'applicazione, ovvero la schermata principale a cui si accede all'ingresso nell'applicazione web. Gli utenti interagiranno direttamente con questo componente, il quale riceverà varie richieste e le inoltrerà ai diversi componenti associati.

Interfacce richieste

- **Richiesta dati:** richiede i dati al componente Gestione risultati che fornisce tutti i dati relativi alle transazioni, budget e wallet relativi ad una data categoria.
- **Esportazione dati:** il sistema interagisce con il componente Gestione Lettura del database, per poter creare il set di dati da poter inserire nel CSV e da poter esportare dalla piattaforma al computer locale dell'utente.

Interfacce fornite

- **Mostra Dashboard:** il componente fornisce una rappresentazione visuale all'utente dei vari dati presenti nella Dashboard, contenenti i dati registrati in piattaforma dall'utente.
- **Richiesta esportazione dati:** il componente inoltra al componente Gestione esportazione dati tale richiesta per compilare il file di transizioni.

Gestione Lettura del database

Questo componente è incaricato di ricevere richieste di lettura al database da altri elementi e di eseguire interrogazioni specifiche per recuperare i dati desiderati. Successivamente, restituirà al componente richiedente le informazioni ottenute. Si interfaccia direttamente con [MongoDB](#) per condurre tutte le operazioni di lettura e restituire ai componenti richiedenti i dati e le informazioni ottenute dalle query.

Interfacce richieste

- **Lettura info profilo:** il componente effettua l'interrogazione sul database per ottenere i dati relativi al profilo di un utente.
- **Lettura Categorie:** il componente effettua l'interrogazione sul database per ottenere i dati relativi alle categorie previste dalla piattaforma.
- **Lettura Wallet:** il componente effettua l'interrogazione sul database per ottenere i dati relativi ai wallets previsti dalla piattaforma.
- **Lettura Transazioni:** il componente effettua l'interrogazione sul database per ottenere i dati relativi alle transazioni previste dalla piattaforma.
- **Lettura Chat:** il componente effettua l'interrogazione sul database per ottenere i dati relativi alle chat con chatbot previste dalla piattaforma.
- **Lettura Budget:** il componente effettua l'interrogazione sul database per ottenere i dati relativi ai budget previsti dalla piattaforma.

Interfacce fornite

- **Informazioni profilo:** il componente invia al componente **Gestione profilo** le informazioni sul profilo dell'utente richiesto.
- **Informazioni Categorie:** il componente invia al componente richiedente le varie informazioni sulle categorie registrate dall'utente.
- **Informazioni Wallet:** il componente invia al componente richiedente le varie informazioni sui wallet registrati dall'utente.
- **Informazioni budget:** il componente invia al componente richiedente le varie informazioni sui budget registrati dall'utente.
- **Informazioni transazioni:** il componente invia al componente richiedente le varie informazioni sulle transazioni registrate dall'utente.
- **Informazioni chat:** il componente invia al componente richiedente le varie informazioni sulle chat con chatbot registrate dall'utente.
- **Crea esportazione dati:** il componente legge il database per fornire tutti i dati necessari per compilare il file [csv](#) affinché l'utente lo possa esportare.

Gestione scrittura del database

Questo modulo, invece, è destinato ad eseguire tutte le operazioni di scrittura, modifica e cancellazione dei dati nel database. Una volta ricevuti i dati da scrivere o modificare dai vari moduli, si collega direttamente al database e invia i dati per eseguire le operazioni di salvataggio o modifica all'interno del cluster.

Interfacce fornite

- **Inoltro Creazione utente:** il componente crea un nuovo utente con i dati forniti.
- **Inoltro Modifica utente:** il componente modifica l'utente con i nuovi dati forniti.
- **Inoltro Eliminazione utente:** il componente elimina l'utente che lo richiede.
- **Inoltro Blocca utente:** il componente si occupa di bloccare un utente dopo una segnalazione di un amministratore.
- **Inoltro Sblocca utente:** il componente si occupa di sbloccare un utente dopo una segnalazione di un amministratore.
- **Inoltro Creazione categoria:** il componente crea una nuova categoria con i dati forniti.
- **Inoltro Modifica categoria:** il componente modifica la categoria con i nuovi dati forniti.
- **Inoltro Eliminazione categoria:** il componente elimina la categoria che lo richiede.
- **Inoltro Creazione transazione:** il componente crea una nuova transazione con i dati forniti.
- **Inoltro Modifica transazione:** il componente modifica la transazione con i nuovi dati forniti.
- **Inoltro Eliminazione transazione:** il componente elimina la transazione che lo richiede.
- **Inoltro Creazione wallet:** il componente crea un nuovo wallet con i dati forniti.
- **Inoltro Modifica wallet:** il componente modifica il wallet con i nuovi dati forniti.
- **Inoltro Eliminazione wallet:** il componente elimina il wallet che lo richiede.
- **Inoltro Creazione chat:** il componente crea una nuova chat con i dati forniti.
- **Inoltro Modifica chat:** il componente modifica la chat con i nuovi dati forniti.
- **Inoltro Eliminazione chat:** il componente elimina la chat che lo richiede.
- **Inoltro Creazione budget:** il componente crea un nuovo budget con i dati forniti.
- **Inoltro Modifica budget:** il componente modifica il budget con i nuovi dati forniti.
- **Inoltro Eliminazione budget:** il componente elimina il budget che lo richiede.

Interfacce richieste

- **Creazione utente:** il componente interagisce con [MongoDB](#) per la creazione dell'utente.
- **Modifica utente:** il componente interagisce con [MongoDB](#) per la modifica dell'utente.
- **Eliminazione utente:** il componente interagisce con [MongoDB](#) per l'eliminazione dell'utente.
- **Blocca utente:** il componente interagisce con [MongoDB](#) per bloccare un utente.

- **Sblocca utente:** il componente interagisce con [MongoDB](#) per sbloccare un utente.
- **Creazione categoria:** il componente interagisce con [MongoDB](#) per la creazione della categoria.
- **Modifica categoria:** il componente interagisce con [MongoDB](#) per la modifica della categoria.
- **Eliminazione categoria:** il componente interagisce con [MongoDB](#) per l'eliminazione della categoria.
- **Creazione transazione:** il componente interagisce con [MongoDB](#) per la creazione della transazione.
- **Modifica transazione:** il componente interagisce con [MongoDB](#) per la modifica della transazione.
- **Eliminazione transazione:** il componente interagisce con [MongoDB](#) per l'eliminazione della transazione.
- **Creazione wallet:** il componente interagisce con [MongoDB](#) per la creazione del wallet.
- **Modifica wallet:** il componente interagisce con [MongoDB](#) per la modifica del wallet.
- **Eliminazione wallet:** il componente interagisce con [MongoDB](#) per l'eliminazione del wallet.
- **Creazione chat:** il componente interagisce con [MongoDB](#) per la creazione della chat.
- **Modifica chat:** il componente interagisce con [MongoDB](#) per la modifica della chat.
- **Eliminazione chat:** il componente interagisce con [MongoDB](#) per l'eliminazione della chat.
- **Creazione budget:** il componente interagisce con [MongoDB](#) per la creazione del budget.
- **Modifica budget:** il componente interagisce con [MongoDB](#) per la modifica del budget.
- **Eliminazione budget:** il componente interagisce con [MongoDB](#) per l'eliminazione del budget.

Gestione risultati

Questo componente si occupa di gestire la schermata che mostra all'utente le informazioni relative al proprio patrimonio (Dashboard) oltre che alle pagine di gestione delle categorie, wallets e transazioni. Gestisce anche le richieste degli utenti relative al filtraggio dell'elenco di tali tool nella pagina specifica.

Interfacce fornite

- **Mostra dati:** il componente mostra le informazioni relative ai dati che un utente ha in piattaforma, che rispondono ad una determinata categoria.

Interfacce richieste

- **Informazioni categorie:** il componente riceve da **Gestione lettura database** le informazioni sulle categorie interessate all'utente.
- **Informazioni transazioni:** il componente riceve da **Gestione transazioni** le informazioni sulle transazioni effettuate dall'utente.
- **Informazioni wallet:** il componente riceve da **Gestione lettura database** le informazioni sui portafogli dell'utente, inclusi il saldo corrente, il nome associato e altri dati rilevanti.
- **Informazioni budget:** il componente riceve da **Gestione lettura database** le informazioni sui budget dell'utente, incluso se li sta rispettando e il capitale rimasto in ognuno di essi.

Gestione chatbot

Questa componente si occupa di gestire la chat con il bot.

Interfacce fornite

- **Mostra chat:** il componente fornisce una rappresentazione visuale delle chat con il chatbot disponibile sulla piattaforma.
- **Invia messaggio:** il componente invia un messaggio al chatbot, inoltrando la richiesta al componente di **Gestione scrittura del database** per registrare la conversazione.

Interfacce richieste

- **Lettura chat:** il componente riceve dal componente **Gestione lettura database** le conversazioni passate.
- **Scrittura chat:** il componente aggiorna la chat corrente, con i nuovi messaggi dell'utente e le risposte del chatbot.
- **Ricevi risposta:** il componente, dopo aver mandato il messaggio al chatbot, deve poter ricevere la risposta.

Gestione transazioni

Questa componente si occupa di gestire le transazioni che l'utente registra in piattaforma, consentendo la loro creazione, modifica ed eliminazione.

Interfacce fornite

- **Visualizza transazioni:** il componente permette di visualizzare i dati relativi alle transazioni che l'utente ha registrato in piattaforma.
- **Creazione transazione:** il componente permette la creazione di una nuova transazione in piattaforma.
- **Modifica transazione:** il componente permette la modifica di una transazione in piattaforma.
- **Eliminazione transazione:** il componente permette l'eliminazione di una transazione in piattaforma.

Interfacce richieste

- **Inoltro Creazione transazione:** il componente interagisce con **Gestione scrittura del database** per la creazione di una transazione.
- **Inoltro Modifica transazione:** il componente interagisce con **Gestione scrittura del database** per la modifica di una transazione.
- **Inoltro Eliminazione transazione:** il componente interagisce con **Gestione scrittura del database** per l'eliminazione di una transazione.
- **Informazioni transazioni:** richiede al componente **Gestione lettura del database** informazioni relative alle transazioni registrate dall'utente in questione.
- **Informazioni valute:** il componente si interfaccia con il sistema esterno per avere informazioni relative ai tassi di cambio delle valute per poter correttamente fornire informazioni aggiornate al sistema.

Gestione categorie

Questa componente si occupa di gestire le categorie che l'utente crea nel sistema, gestendo quindi anche la loro modifica, eliminazione e utilizzo da parte di altri componenti.

Interfacce fornite

- **Visualizza categorie:** il componente permette di visualizzare i dati relativi alle categorie che l'utente ha registrato in piattaforma.
- **Creazione categoria:** il componente permette la creazione di una categoria in piattaforma.
- **Modifica categoria:** il componente permette la modifica di una categoria in piattaforma.
- **Eliminazione categoria:** il componente permette l'eliminazione di una categoria in piattaforma.

Interfacce richieste

- **Inoltro Creazione categoria:** il componente interagisce con **Gestione scrittura del database** per la creazione di una categoria.
- **Inoltro Modifica categoria:** il componente interagisce con **Gestione scrittura del database** per la modifica di una categoria.
- **Inoltro Eliminazione categoria:** il componente interagisce con **Gestione scrittura del database** per l'eliminazione di una categoria.
- **Informazioni categorie:** richiede al componente **Gestione lettura del database** informazioni relative alle categorie registrate dall'utente in questione.

Gestione wallet

Questa componente si occupa di gestire i wallet che l'utente crea nel sistema, gestendo quindi anche la loro modifica ed eliminazione.

Interfacce fornite

- **Visualizza wallet:** il componente permette di visualizzare i dati relativi ai wallet che l'utente ha registrato in piattaforma.
- **Creazione wallet:** il componente permette la creazione di un wallet in piattaforma.
- **Modifica wallet:** il componente permette la modifica di un wallet in piattaforma.
- **Eliminazione wallet:** il componente permette l'eliminazione di un wallet in piattaforma.

Interfacce richieste

- **Inoltro Creazione wallet:** il componente interagisce con **Gestione scrittura del database** per la creazione di un wallet.
- **Inoltro Modifica wallet:** il componente interagisce con **Gestione scrittura del database** per la modifica di un wallet .
- **Inoltro Eliminazione wallet:** il componente interagisce con **Gestione scrittura del database** per l'eliminazione di un wallet.
- **Informazioni wallet:** richiede al componente **Gestione lettura del database** informazioni relative ai wallet registrati dall'utente in questione.

Gestione budget

Questa componente si occupa di gestire i budget che l'utente crea nel sistema, gestendo quindi anche la loro modifica ed eliminazione.

Interfacce fornite

- **Visualizza budget:** il componente permette di visualizzare i dati relativi ai budget che l'utente ha registrato in piattaforma.
- **Creazione budget:** il componente permette la creazione di un budget in piattaforma.
- **Modifica budget:** il componente permette la modifica di un budget in piattaforma.
- **Eliminazione budget:** il componente permette l'eliminazione di un budget in piattaforma.

Interfacce richieste

- **Inoltro Creazione budget:** il componente interagisce con **Gestione scrittura del database** per la creazione di un budget.
- **Inoltro Modifica budget:** il componente interagisce con **Gestione scrittura del database** per la modifica di un budget.
- **Inoltro Eliminazione budget:** il componente interagisce con **Gestione scrittura del database** per l'eliminazione di un budget.
- **Informazioni budget:** richiede al componente **Gestione lettura del database** informazioni relative ai budget registrati dall'utente in questione.

