



POLITECNICO DI MILANO

INGEGNERIA DEL SOFTWARE 2

**TravelDream**

---

## Design Document

---

816820 Federico BADINI  
818196 Alessandro BALZI  
816512 Massimo DE MARCHI

*Prof.*  
Luca MOTTOLA

20 dicembre 2013

# Indice

<b>1</b>	<b>Introduzione</b>	<b>2</b>
1.1	Obiettivo . . . . .	2
1.2	Ambito . . . . .	2
1.3	Riferimenti . . . . .	2
1.4	Glossario . . . . .	3
<b>2</b>	<b>Panoramica</b>	<b>4</b>
2.1	Composizione del documento . . . . .	4
2.2	Contesto e progettazione . . . . .	5
2.3	Funzionalità . . . . .	6
<b>3</b>	<b>Architettura del sistema</b>	<b>8</b>
3.1	Design Architetturale . . . . .	8
3.2	Analisi della User eXperience . . . . .	10
3.2.1	Impiegato . . . . .	10
3.2.2	Utente . . . . .	13
3.3	Analisi dei livelli logici (BCE) . . . . .	17
3.3.1	Impiegato . . . . .	17
3.3.2	Utente . . . . .	20
<b>4</b>	<b>Analisi dei dati</b>	<b>24</b>
4.1	Progettazione concettuale . . . . .	24
4.1.1	Entità e attributi . . . . .	25
4.1.2	Relazioni . . . . .	36
4.2	Progettazione logica . . . . .	37
<b>5</b>	<b>Appendice</b>	<b>39</b>
5.1	Modifiche al RASD . . . . .	39

# 1 | Introduzione

## 1.1 Obiettivo

All'interno del presente documento viene descritta l'architettura software del portale e-commerce TravelDream. L'obiettivo del seguente documento è quello di fornire un dettaglio del sistema sufficiente a guidare la successiva fase di implementazione. La descrizione del sistema procede secondo un approccio top-down. La scelta di seguire tale approccio è dettata dalla volontà di realizzare un sistema il più possibile aderente ai requisiti di alto livello formalizzati nel documento di analisi. Nel seguito verranno descritti innanzitutto gli aspetti di navigazione utente, poi verrà fornita una scomposizione del sistema tramite diagrammi BCE e poi si procederà ad analizzare l'architettura dal punto di vista dei dati. Il presente documento si rivolge principalmente agli sviluppatori che dovranno implementare il sistema, i quali saranno tenuti a seguire modalità e scelte di progettazione qui descritte.

## 1.2 Ambito

TravelDream è un servizio online di commercializzazione dei pacchetti, la cui realizzazione è stata commissionata da Travel Dream Co.. Le funzionalità, gli obiettivi e le specifiche dettagliate di tale piattaforma sono descritte all'interno del documento di analisi elaborato nella fase precedente del progetto. Ogni eventuale modifica a tale documento sarà annotata e descritta in appendice.

## 1.3 Riferimenti

- Requirement Analysis and Specification Document
- Software design document template  
[http://www.ceng.metu.edu.tr/\\_media/course/ceng490/sdd\\_template.pdf?rev](http://www.ceng.metu.edu.tr/_media/course/ceng490/sdd_template.pdf?rev)

- IEEE Draft Standard for Software Design Descriptions  
<http://www.iso-architecture.org/ieee-p1016/IEEE-P1016-d5.0b.pdf>

## 1.4 Glossario

Sono in seguito elencate le principali definizioni, utili ad una migliore comprensione del documento:

- **Top-down**: strategia di progettazione software in cui si fornisce dapprima una visione generale del sistema, per poi entrare nel dettaglio delle singole parti che lo compongono.
- **Layer**: livello logico che racchiude un insieme di funzionalità dell'applicazione
- **Server**: componente informatico che fornisce un servizio ad altri componenti detti Client.
- **Client**: componente informatico che richiede un servizio ad un altro componente detto Server.
- **Package**: Macro-partizione del sistema software.

Analogamente forniamo ora il significato degli acronimi e delle abbreviazioni presenti nel documento.

- **RASD**: Requirement Analysis and Specification Document
- **GUI**: Graphical User Interface
- **JSF**: Java Server Faces
- **EJB**: Enterprise Java Bean
- **DBMS**: Database Management System
- **UX**: User eXperience model
- **BCE**: Boundary Control Entity model
- **ER**: Entity Relationship model

## 2 | Panoramica

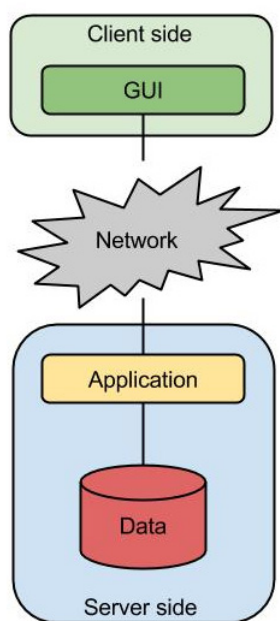
### 2.1 Composizione del documento

Il presente documento è suddiviso nelle seguenti sezioni:

- **Introduzione:** sezione che definisce aspetti generali relativi al design del sistema oltre a definire scopi e destinatari per il presente documento.
- **Panoramica del sistema:** sezione che riassume i principali aspetti di carattere funzionale e di integrazione per il sistema.
- **Architettura del sistema:** sezione che descrive approfonditamente struttura, navigazione e scomposizione logica dell'applicativo.
- **Design dei dati:** sezione che descrive come vengono memorizzate e raggruppate le informazioni gestite dal sistema.

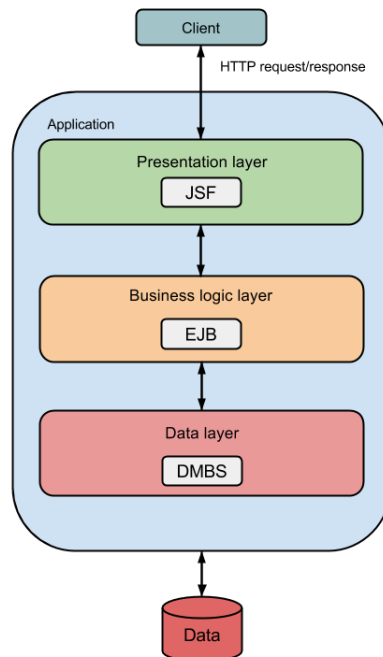
## 2.2 Contesto e progettazione

Il sistema che deve essere sviluppato non prevede nè interazioni con altri sistemi, nè lo sviluppo di una logica di processing e gestione dei pagamenti. La piattaforma fornisce servizi tramite Internet e viene implementata con un'architettura thin-client. Lato utente l'applicazione necessiterà unicamente di un browser; il sistema risiederà su un unico server.



L'applicativo sarà suddiviso su tre livelli logicamente separati ovvero il presentation layer, il business logic layer e il data layer. Viene ora fornita una breve descrizione dei tre layer:

- **Presentation Layer:** livello che gestisce l'interazione con l'utente. In modo particolare il presentation layer è costituito da componenti che ricevono le richieste utente via protocollo http e producono risposte per l'utente sulla base di un dialogo con il sottostante livello di business logic.
- **Business logic Layer:** Il livello di business logic si occupa del corretto processing delle operazioni del sistema. Occupa una posizione centrale nella scomposizione a livelli così da poter ricevere richieste dal livello di presentazione, interrogare il livello dati e fornire delle risposte adeguate.
- **Data Layer:** livello che incorpora la specifica dei dettagli relativi all'interazione con le strutture dati e fornisce al business logic layer un insieme di primitive semplificate per potervi accedere.



I diversi livelli logici verranno, come detto, mappati su un'unica macchina fisica. Il dialogo fra utente e portale prevede un'interazione tramite protocollo http tra il browser lato client ed il presentation layer lato server. Ricevuta la richiesta il livello di presentation la evade tramite un dialogo con i livelli sottostanti, ovvero il livello che controlla la logica di business ed il livello di gestione dei dati. Il modello strutturato a livelli prevede unicamente comunicazione fra livelli adiacenti.

## 2.3 Funzionalità

Si riporta qui un breve riassunto delle principali funzionalità del sistema, esposte in maniera approfondita nel precedente documento di analisi. La piattaforma permetterà agli utenti correttamente autenticati di effettuare operazioni di ricerca, modifica, composizione ed acquisto di pacchetti viaggio. Gli utenti autenticati potranno poi condividere pacchetti viaggio con amici e creare una gift list personale, ovvero una lista dei pacchetti di viaggio che si desidera acquistare. La gift list personale sarà accessibile all'interno pagina personale di ogni utente registrato. Gli utenti non autenticati potranno prendere visione dei pacchetti viaggio che vengono condivisi con loro da utenti registrati. Ogni altra funzionalità del portale richiede l'autenticazione. La piattaforma, dal punto di vista della gestione, dovrà poi garantire agli impiegati di TravelDream Co. tutti gli strumenti necessari ad una corretta gestione delle soluzioni di viaggio commercializzate. In modo particolare gli

impiegati dovranno poter gestire pacchetti e risorse in termini di creazione, modifica ed eliminazione.

Di seguito per maggiore chiarezza si riportano i suddetti requisiti funzionali in maniera schematica.

#### **Impiegato**

- RF1: autenticazione al sistema (login).
- RF2: inserimento di una risorsa.
- RF3: eliminazione di una risorsa.
- RF4: modifica di una risorsa.
- RF5: inserimento di un pacchetto.
- RF6: eliminazione di un pacchetto.
- RF7: modifica di un pacchetto.

#### **Utente Non Autenticato**

- RF8: registrazione al sistema.
- RF9: autenticazione al sistema (login).
- RF10: visione di pacchetti condivisi.

#### **Utente Autenticato**

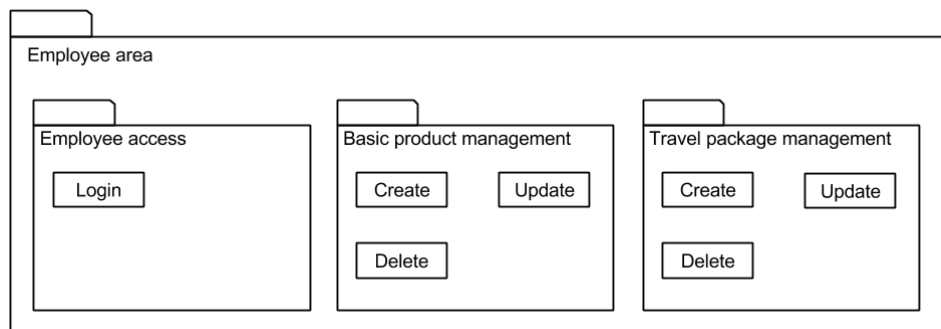
- RF11: selezione e personalizzazione di una pacchetto.
- RF12: acquisto di una pacchetto.
- RF13: condivisione di un pacchetto.
- RF14: aggiunta di un pacchetto in gift list.
- RF15: ricerca di un utente.
- RF16: regalo di un pacchetto.



## 3 | Architettura del sistema

### 3.1 Design Architetture

Il design del software è stato affrontato in maniera modulare, tramite scomposizione dell'applicazione in package. Tale divisione tiene conto in primo luogo della tipologia di utente che dialoga con il portale ed in secondo luogo delle funzionalità offerte dal portale per la tipologia di utente in questione. Ciascun package applicativo racchiuderà come detto un sottoinsieme di funzionalità del sistema e tale sottoinsieme verrà ripartito tra i vari componenti presenti nel package. La scomposizione relativa agli utenti prevede di partizionare il sistema in una parte dedicata alle funzionalità amministrative proprie degli impiegati e una parte dedicata alle funzionalità utente. Presentiamo ora la prima partizione, ossia quella relativa alle funzionalità lato impiegato:

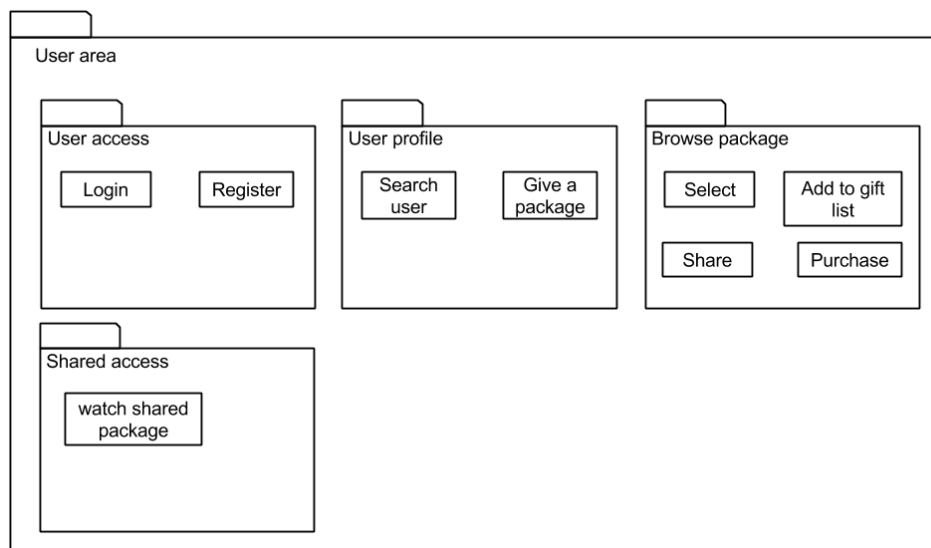


Le funzionalità del portale lato impiegato sono suddivise in tre macroaree, ossia:

- **Accesso:** sezione che prevede la gestione dell'autenticazione al portale da parte degli impiegati.
- **Gestione dei prodotti base:** sezione che è composta dall'insieme di funzionalità relative alla gestione del ciclo di vita dei prodotti base (creazione, aggiornamento ed eliminazione della risorsa dalle disponibilità).

- **Gestione dei pacchetti viaggio:** sezione che è composta dall'insieme di funzionalità relative alla gestione del ciclo di vita di un pacchetto (creazione, aggiornamento, eliminazione del pacchetto dalle disponibilità).

La partizione del sistema che modella le funzionalità utente è invece la seguente:



Le funzionalità del portale lato utente sono suddivise in quattro macroaree, ossia:

- **Accesso:** sezione che prevede la gestione dell'autenticazione al portale da parte degli utenti.
- **Profilo utente:** sezione che raggruppa le funzionalità di ricerca utente e di regalo di un pacchetto inserito nella gift list di un utente.
- **Browsing:** sezione che raggruppa le funzionalità di ricerca, personalizzazione, acquisto condivisione o aggiunta alla gift list del pacchetto personalizzato.
- **Accesso da condivisione:** sezione che contiene le funzionalità legate all'interazione con un soggetto che ha ricevuto la condivisione di un pacchetto da un utente registrato. Fase questa che prevede il recupero sul sistema dei dati relativi al pacchetto che era stato condiviso, la presentazione del contenuto dello stesso all'utente generico e la possibile riconciliazione con il flusso di acquisto previa autenticazione al portale.

## 3.2 Analisi della User eXperience

Per quanto riguarda il livello di presentazione si vogliono mostrare i percorsi di navigazione forniti dall'interfaccia grafica all'utente finale. A questo scopo sono stati realizzati opportuni modelli di navigazione (User eXperience) organizzati per gruppi di funzionalità.

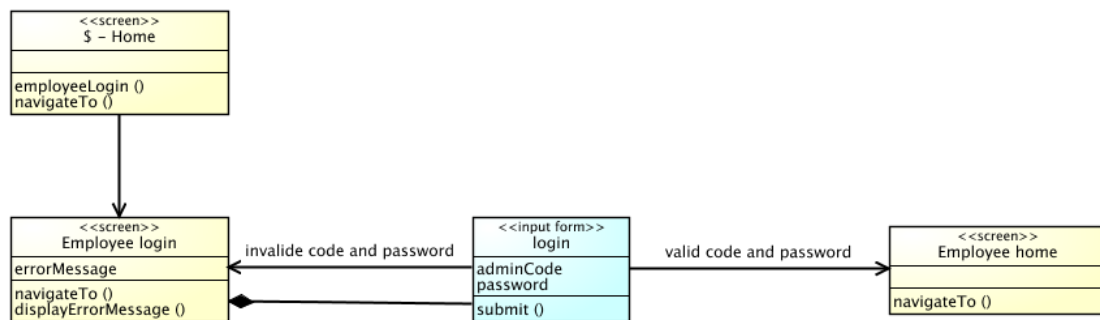
Il modello User eXperience o UX-model è caratterizzato da una serie di schermate (definite tramite lo stereotipo «screen») collegate tra loro mediante metodi o funzionalità. In ogni schermata è sempre presente il metodo statico `navigateTo` che permette la navigazione fra le pagine. Gli screen possono contenere parti dati e form che permettono l'inserimento di dati da parte dell'utente (definiti tramite lo stereotipo «input form»).

La notazione \$, presente in alcuni screen, denota il fatto che lo screen annotato in tale modo risulta sempre raggiungibile dagli altri screen del diagramma. Per esigenze di sintesi e di miglior comprensione dei grafi si è scelto di inserire all'interno dei vari screen solo le funzionalità necessarie a descrivere per ciascun grafo la specifica interazione.

### 3.2.1 Impiegato

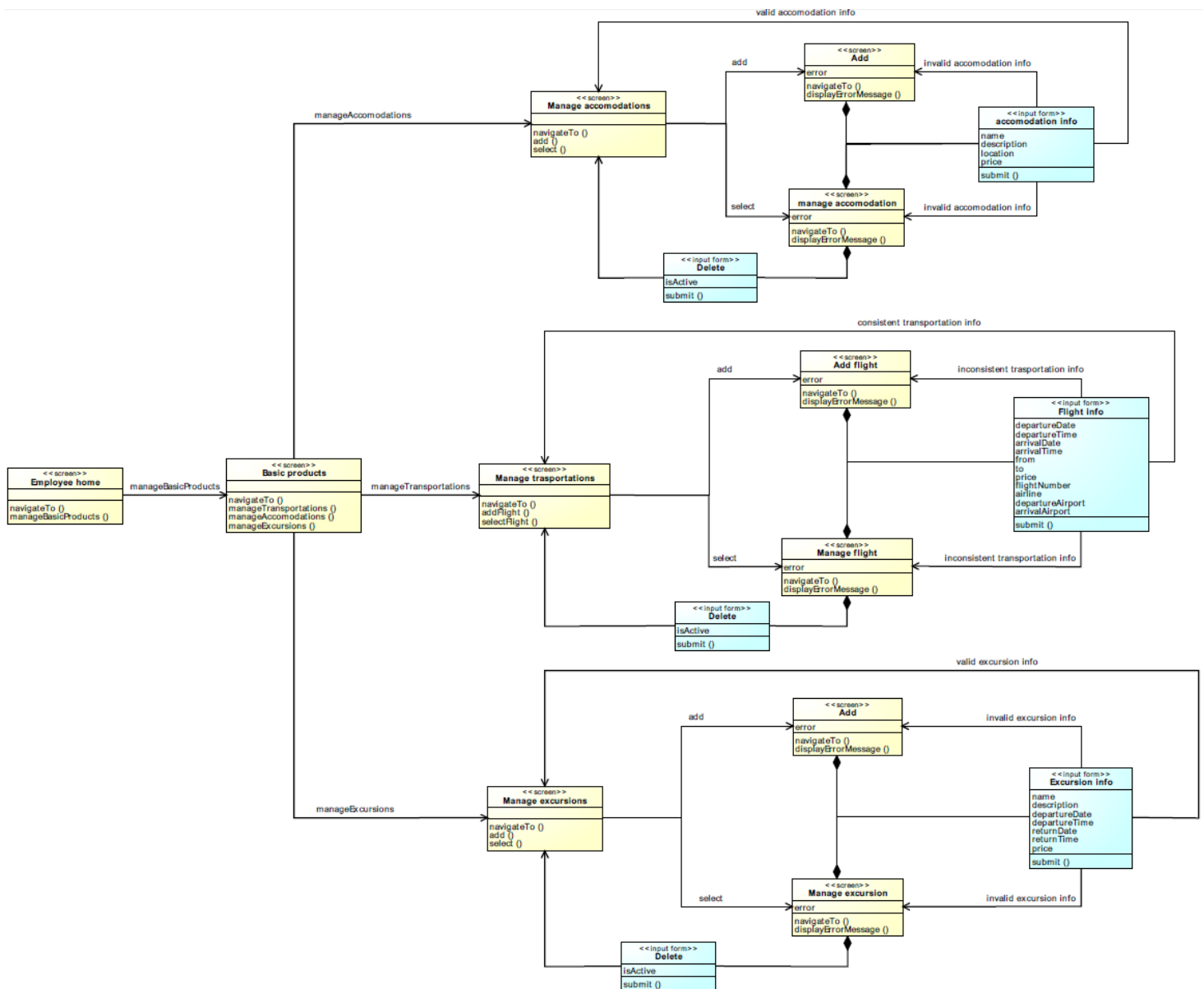
#### Accesso

Nel primo UX-model è descritto lo schema di navigazione relativo all'accesso lato impiegato. Se l'autenticazione ha esito positivo si viene reindirizzati alla pagina di gestione dell'impiegato, altrimenti si rimane sulla pagina di login.



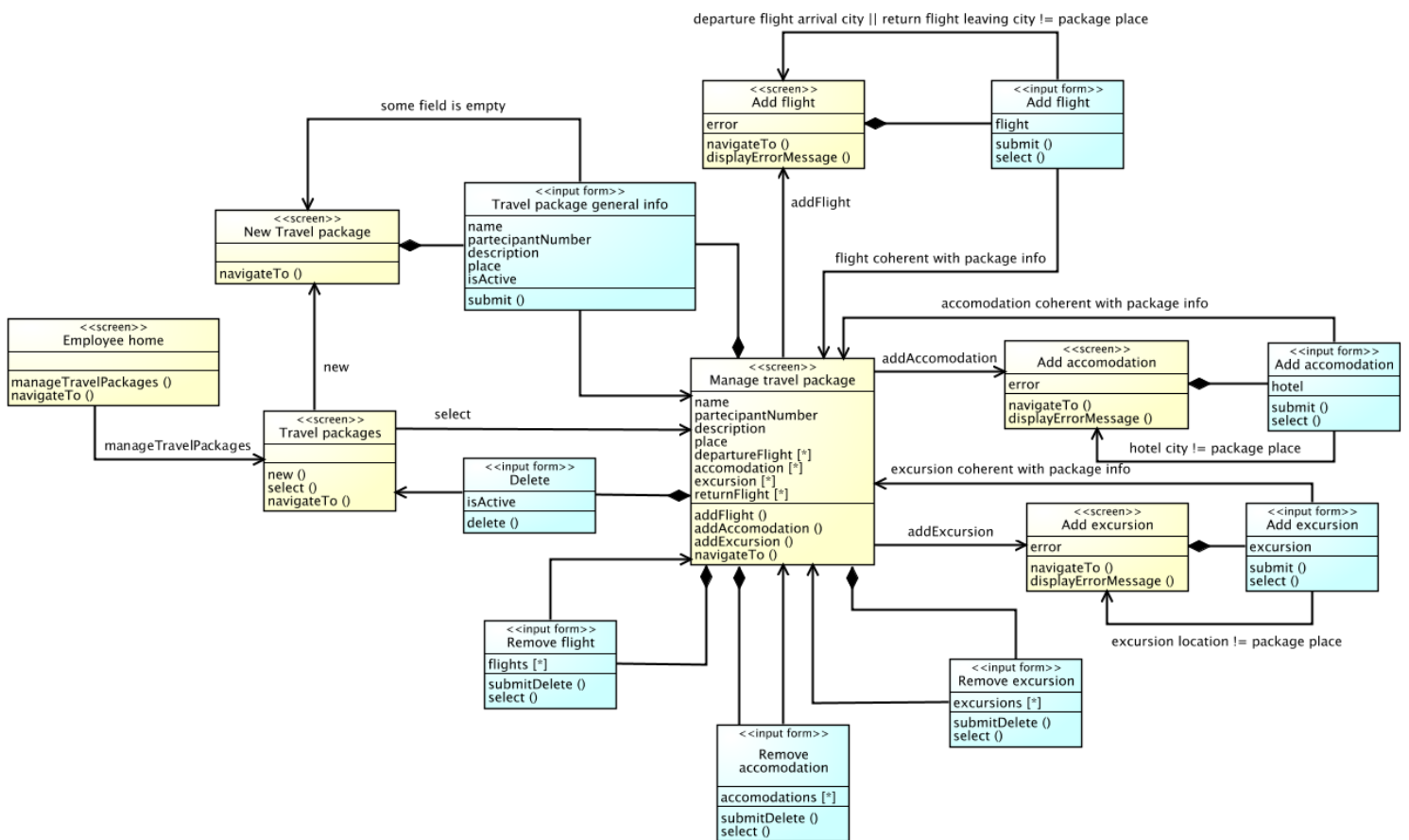
## Gestione dei prodotti base

Nel seguente diagramma viene descritta l'interazione dell'impiegato con il sistema al fine di inserire modificare o rimuovere dalle disponibilità uno specifico prodotto base. In modo particolare si evidenziano due flussi distinti ossia il flusso relativo all'introduzione di una nuova risorsa e quello relativo alla selezione di una risorsa che può poi proseguire con la cancellazione o con la modifica della stessa.



## Gestione dei pacchetti di viaggio

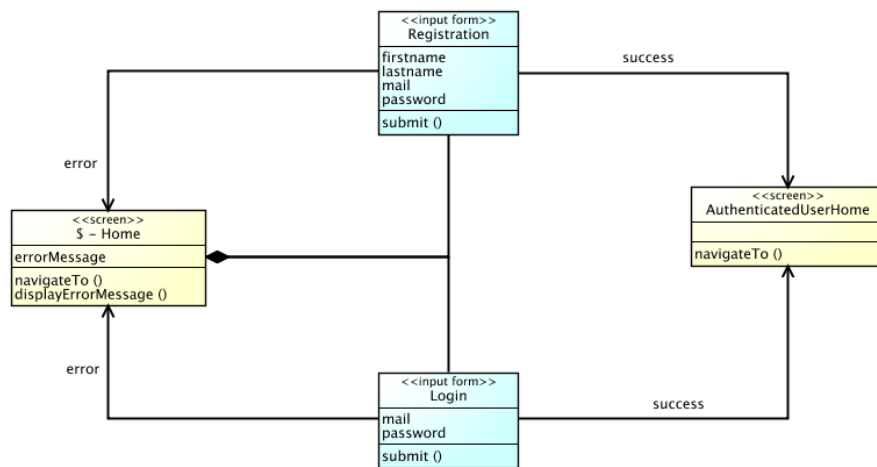
Nel seguente diagramma viene descritta l'interazione dell'impiegato con il sistema al fine di inserire modificare o rimuovere dalle disponibilità uno specifico prodotto base. Sono visibili due flussi distinti nel grafo ossia il flusso di creazione di un nuovo pacchetto ed il flusso di selezione di un pacchetto. I due flussi si unificano quando si passa alla fase di inserimento dei prodotti base all'interno dei pacchetti.



### 3.2.2 Utente

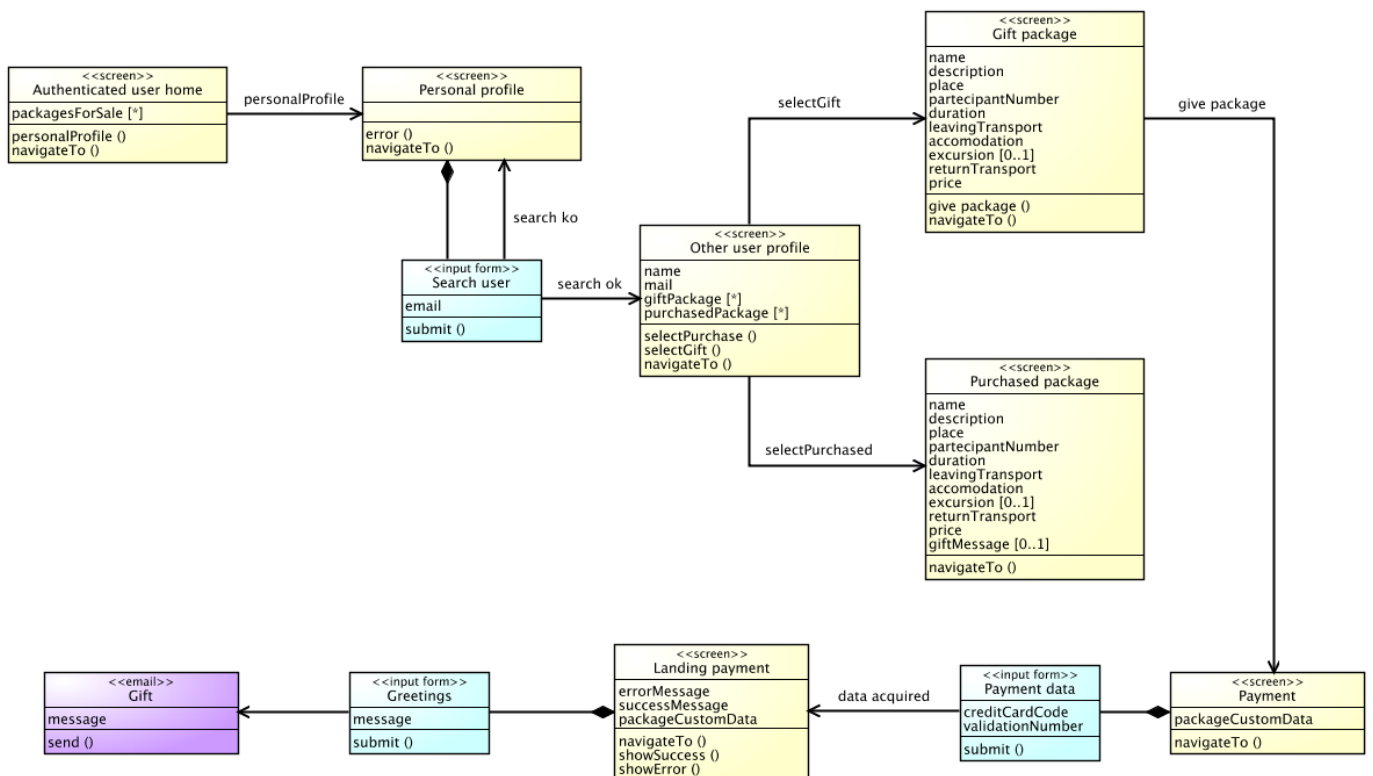
#### Accesso

Nel seguente diagramma Ux si descrive l'interazione lato utente relativa all'autenticazione/registrazione al portale. L'utente sarà accolto da una pagina di benvenuto dove saranno già presenti i form relativi al login e alla registrazione. Per accedere ai servizi offerti dal sito dovrà procedere a registrarsi o ad autenticarsi.



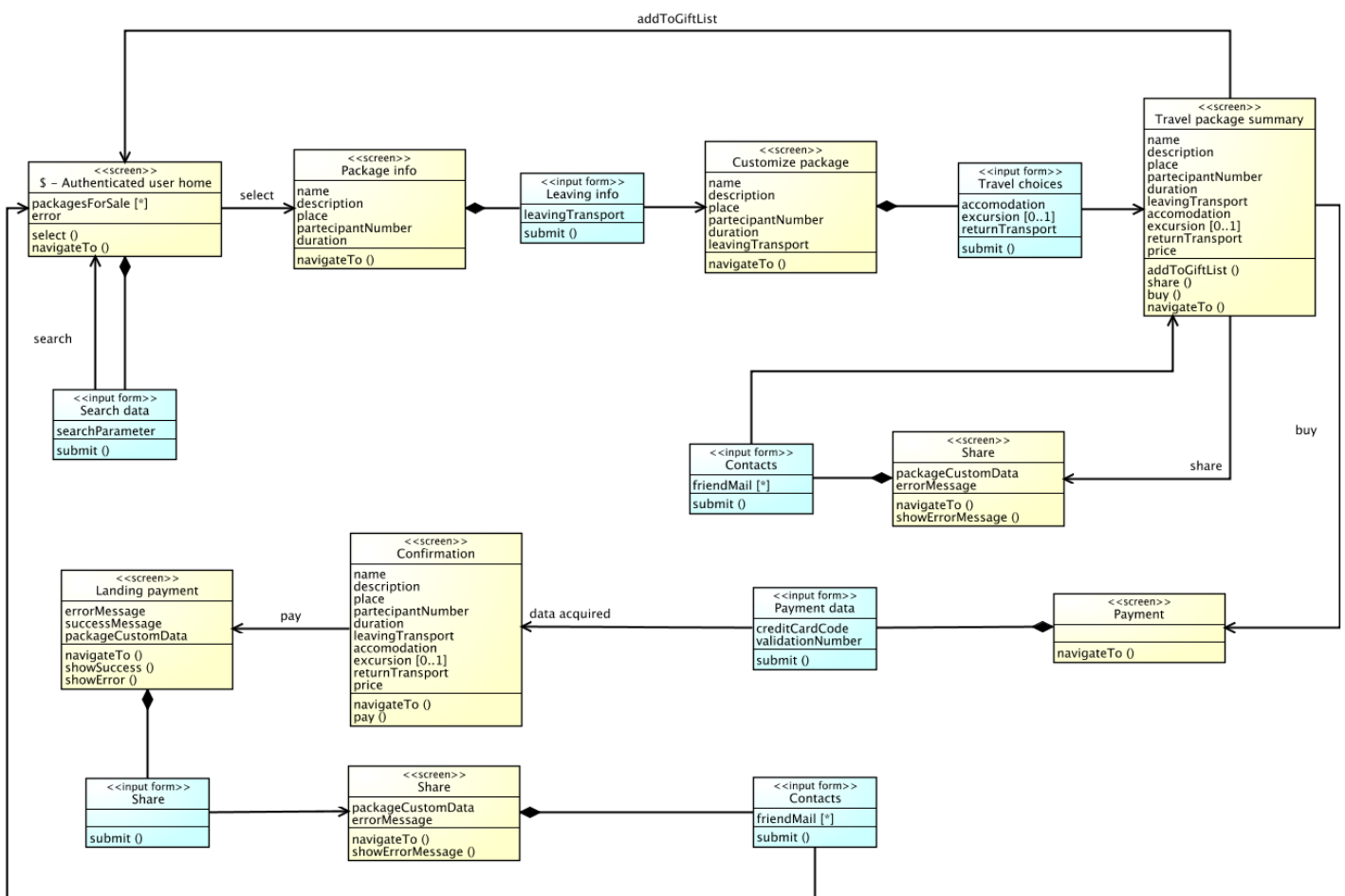
## Profilo utente

Il diagramma descrive le possibili funzionalità legate alla gestione dei profili, disponibili ad un utente autenticato. L'utente dalla propria pagina profilo potrà raggiungere tramite ricerca la pagina profilo di un altro utente, visitarne il dettaglio della giftList e anche procedere all'acquisto di pacchetti in giftList.



## Scelta del pacchetto

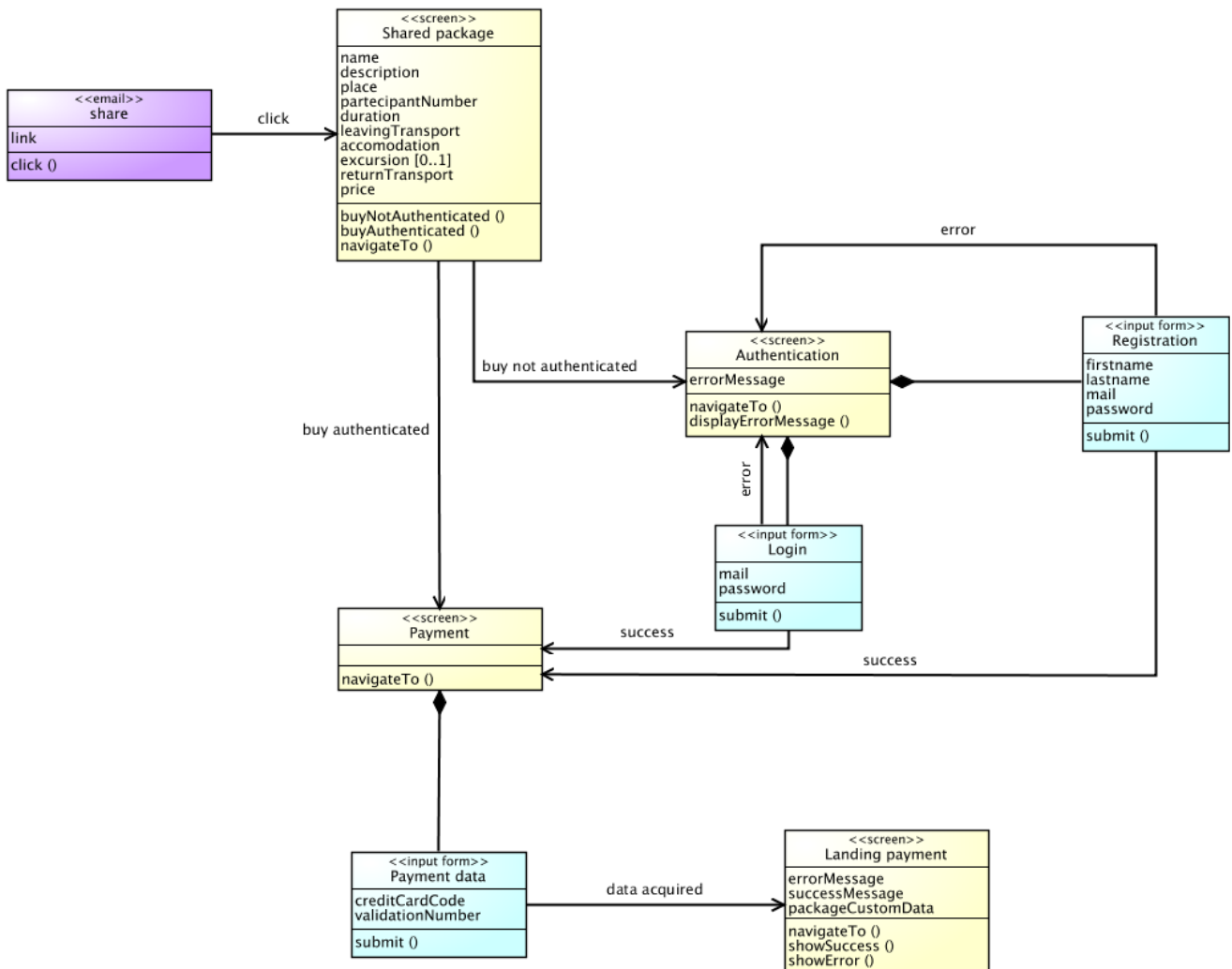
Il diagramma descrive l'interazione tra l'utente ed il sistema in caso di ricerca, personalizzazione, acquisto, condivisione ed inserimento in giftList di un pacchetto. In particolare durante la fase di personalizzazione il sistema richiederà innanzitutto la scelta del viaggio di andata tra le opzioni disponibili. Nella fase successiva il sistema provvederà poi a proporre all'utente scelte consistenti riguardanti il volo di ritorno, il pernottamento e l'eventuale escursione.





## Visione di un pacchetto condiviso

Il diagramma descrive come il sistema risponde a fronte di una richiesta da parte di un utente generico di prendere visione di un pacchetto di cui ha ricevuto la condivisione. Il sistema fornirà il dettaglio del pacchetto e il cliente, nel caso in cui lo ritenga necessario, potrà procedere all'acquisto del pacchetto previa autenticazione.



### 3.3 Analisi dei livelli logici (BCE)

I diagrammi BCE affrontano il problema del design secondo una prospettiva differente rispetto ai diagrammi Ux. Non hanno più come obiettivo quello di mostrare la navigazione utente, pertanto non si limitano a modellare il livello di presentazione del sistema ma lo analizzano anche negli aspetti di logica applicativa e presentazione dati.

Al fine di una maggiore comprensione dei modelli di seguito riportati è bene sottolineare che un diagramma BCE presenta tre stereotipi fondamentali: Boundary, Control ed Entity.

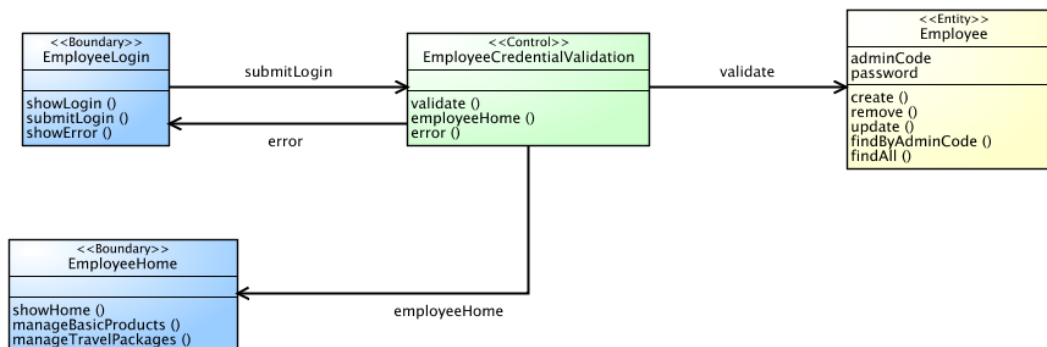
- Lo stereotipo Boundary serve per rappresentare in maniera molto sintetica l'interazione con l'utente.
- Lo stereotipo Control rappresenta un elemento della logica applicativa. Oggetti di questo tipo gestiscono lo stato degli oggetti applicativi utilizzati per fornire le funzionalità descritte attraverso i boundary.
- Lo stereotipo Entity rappresenta un elemento della logica di accesso ai dati.

I seguenti grafici andranno a analizzare in profondità sui tre livelli i concetti visti nei precedenti diagrammi di navigazione.

#### 3.3.1 Impiegato

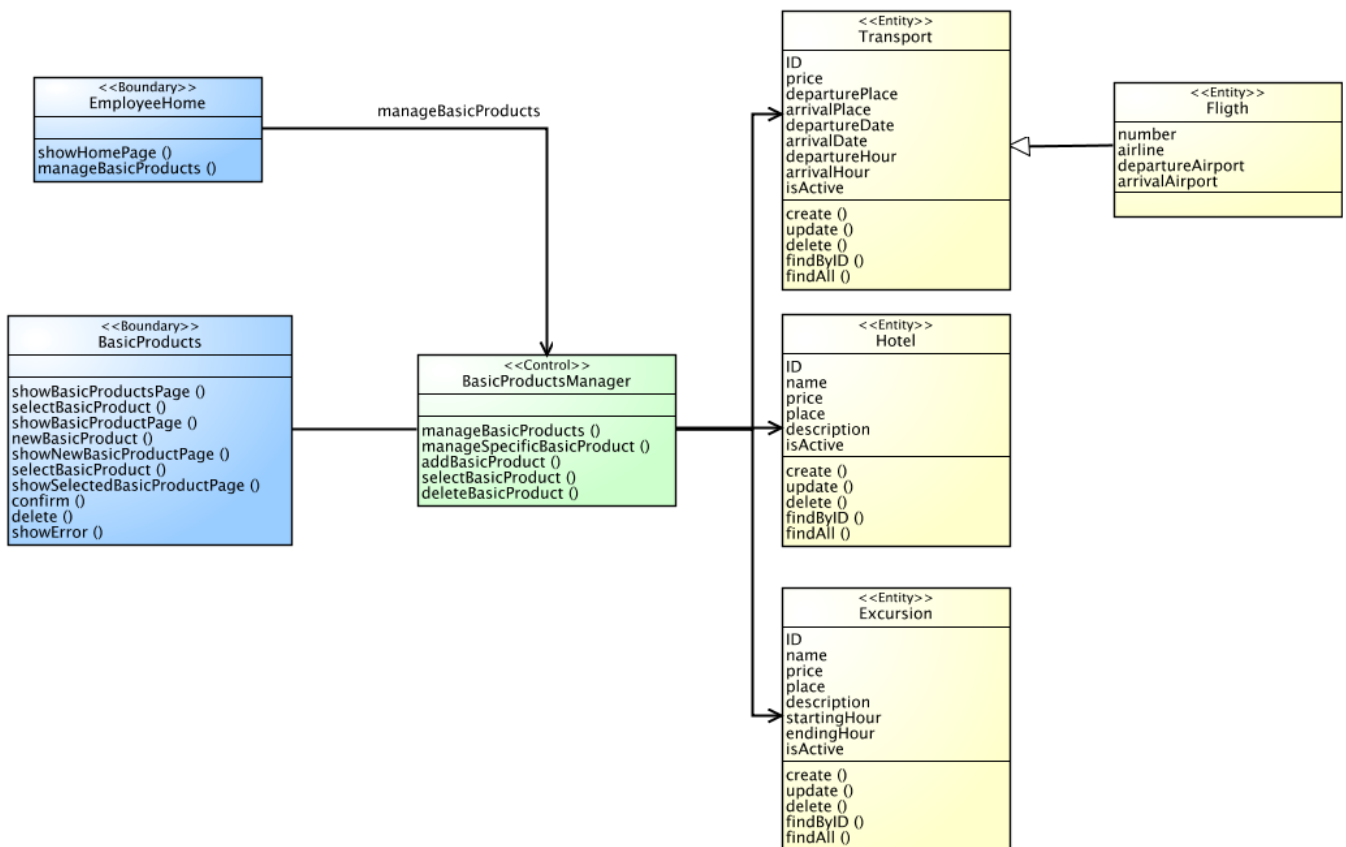
##### Accesso

Il seguente diagramma descrive il login di un impiegato. A livello presentazione il sistema prevede il display delle pagine e la ricezione degli input; a livello di controllo il sistema dovrà essere in grado di ricevere le richieste provenienti dal livello presentazione e fornire una risposta positiva o negativa all'utente in base all'esito di un confronto fra i dati immessi e quelli raggiungibili tramite le entity.



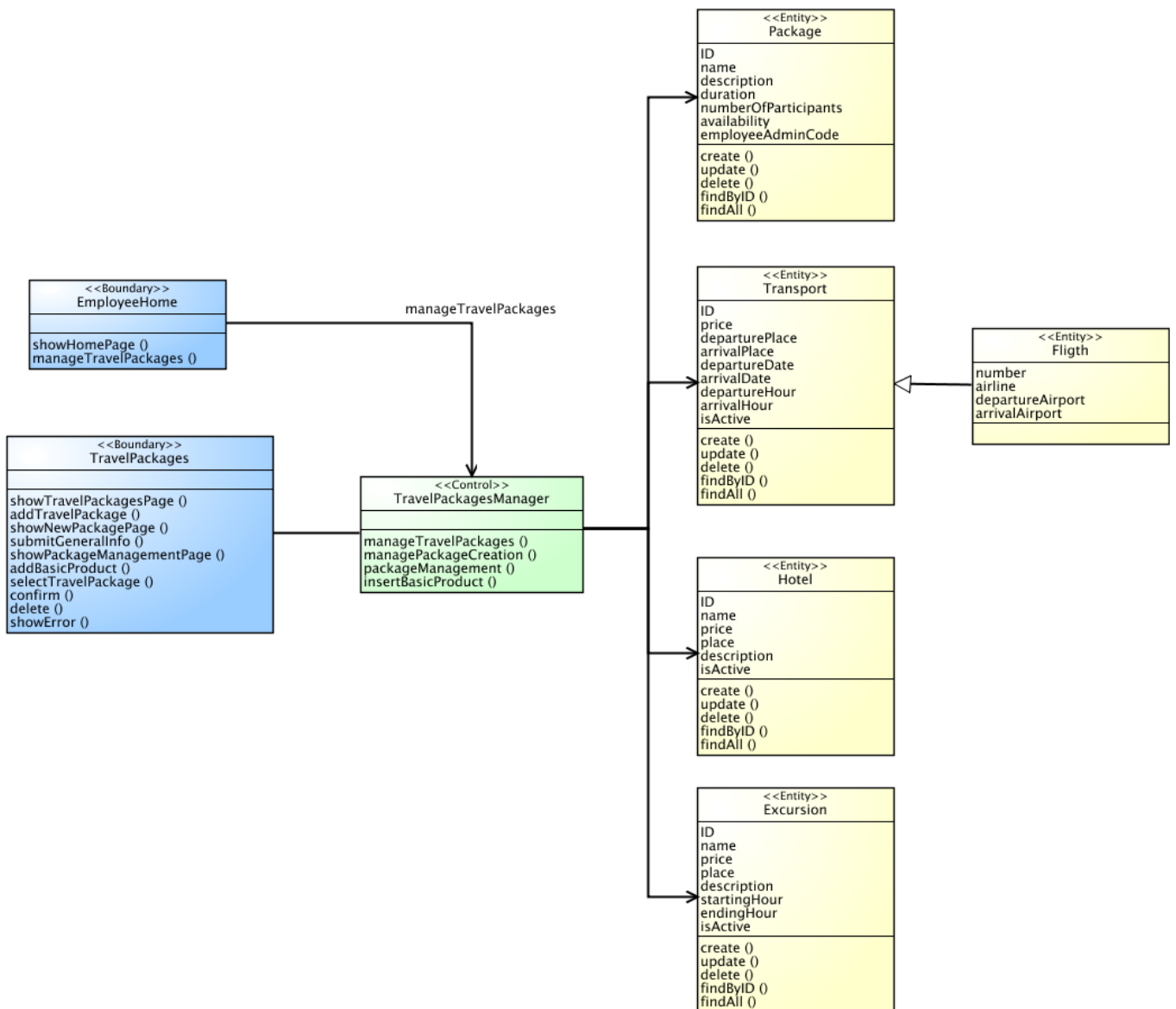
## Gestione dei prodotti base

Il diagramma mostra le funzionalità legate al livello presentazione, ossia il display delle pagine lato utente e la ricezione degli input, le funzionalità di controllo come la gestione delle modifiche all'insieme di una particolare tipologia di prodotti base, e le principali entità coinvolte.



## Gestione dei pacchetti viaggio

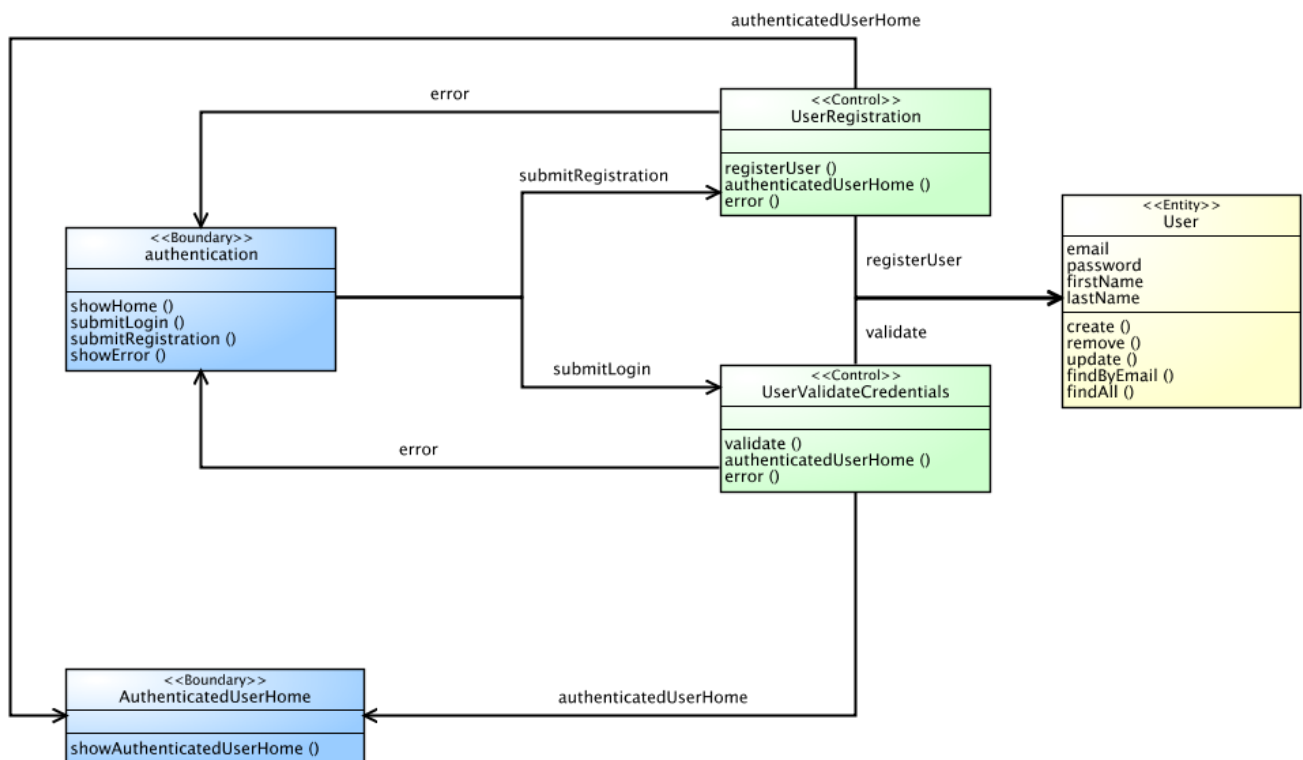
Come per il diagramma precedente si assiste ad una divisione dell'interazione nei tre livelli applicativi. Il controllore, come per il caso dei prodotti base, incanala il flusso proveniente dall'utente e provvede all'interazione con le entità coinvolte.



### 3.3.2 Utente

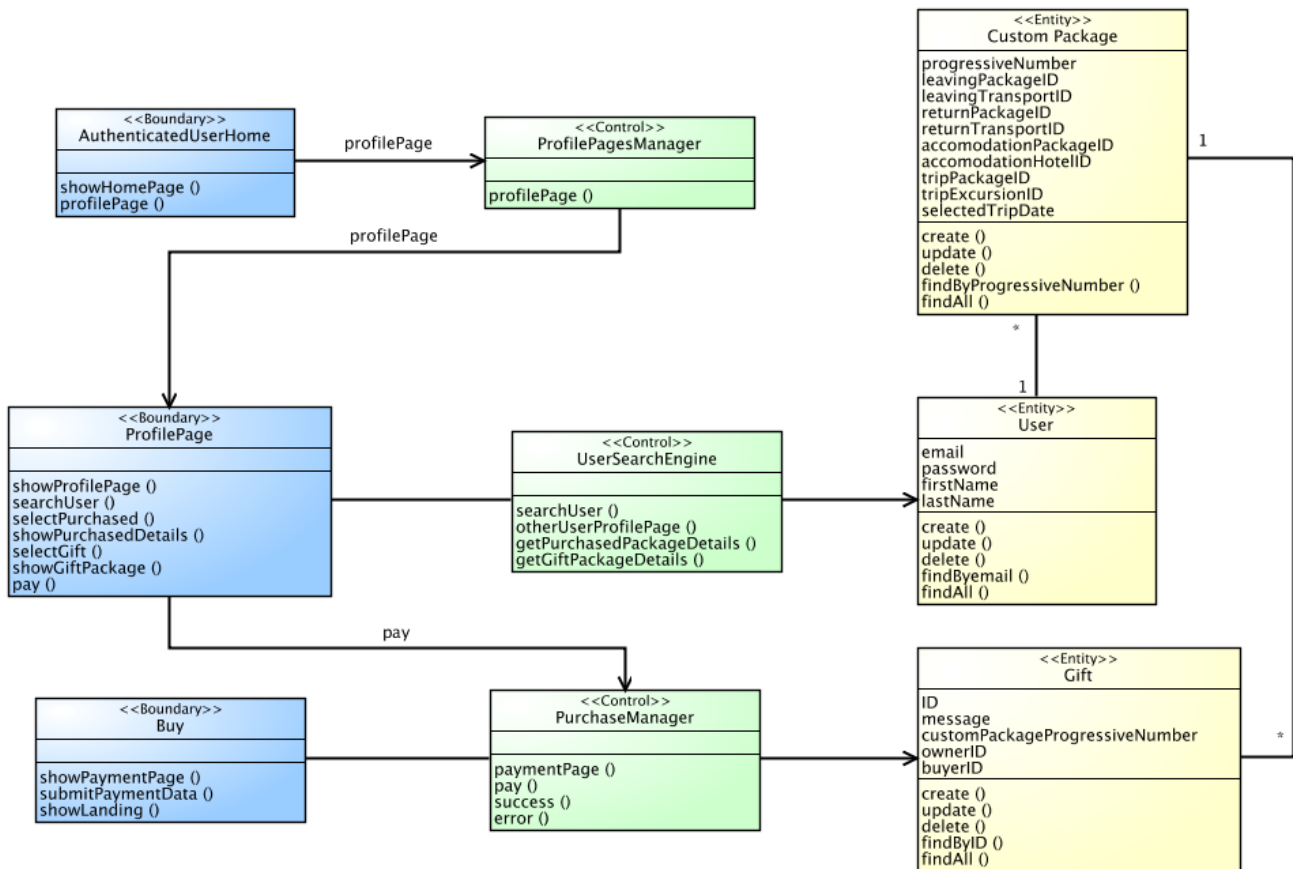
#### Accesso

Il diagramma mostra come l'autenticazione utente procede a partire da un contesto liberamente accessibile, rappresentato dalla boundary home, verso un contesto visibile solo per gli utenti correttamente autenticati al portale (Authenticated User Home). L'accesso a questa sezione del portale è mediato da due controllori: il controllore di login, che provvede a verificare le credenziali inserite sul database, ed il controllore di registrazione, che inserisce i dati relativi al nuovo utente nel database.



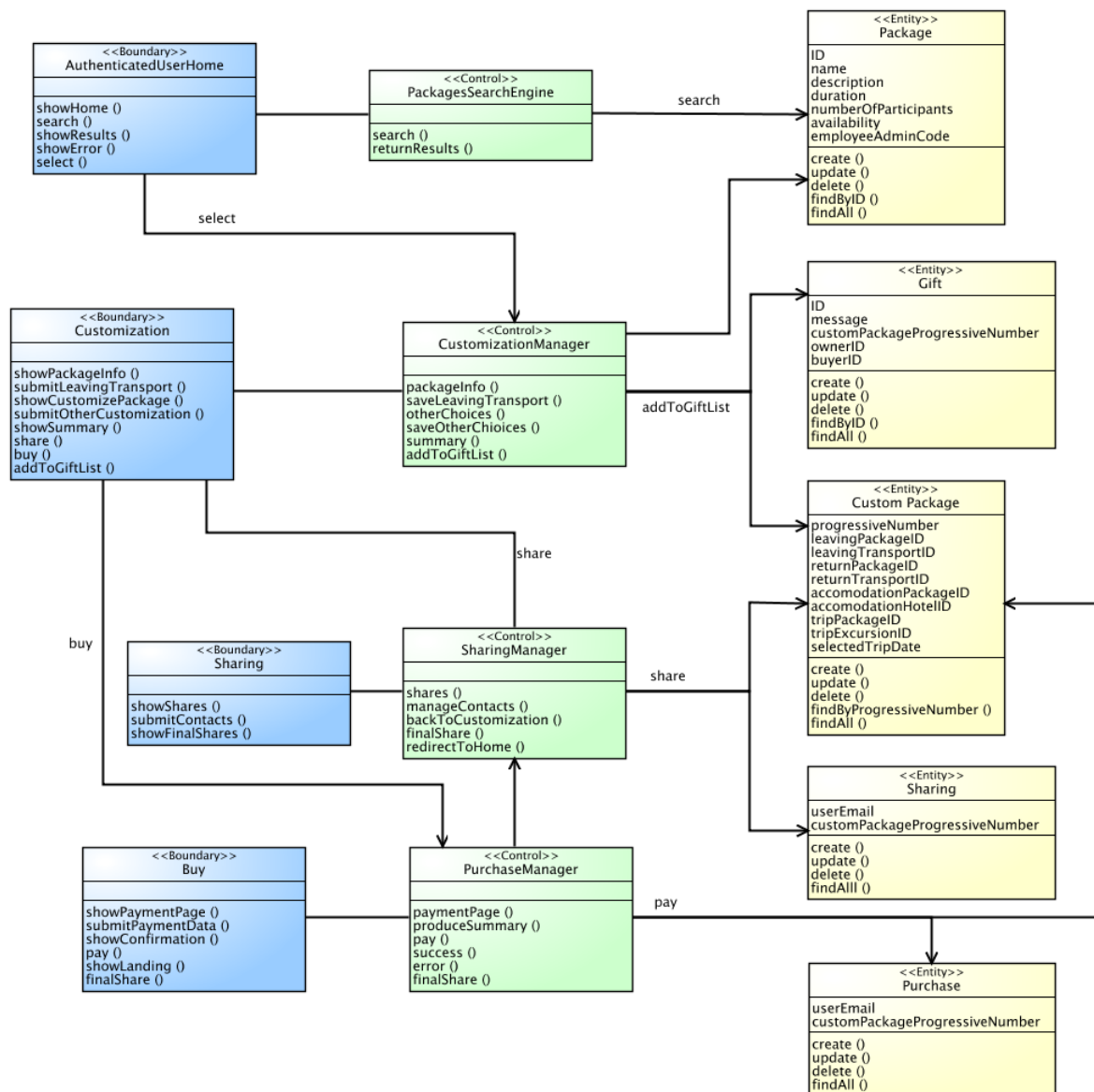
## Profilo utente

Il diagramma mostra le funzionalità offerte ad un utente autenticato focalizzandosi sugli aspetti relativi alla consultazione di profili utente. Le funzionalità principali che vengono descritte all'interno di questo grafo sono la ricerca utente e l'acquisto di un pacchetto inserito in GiftList; tali funzionalità sono gestite dallo UserSearchEngine e dal PurchaseManager.



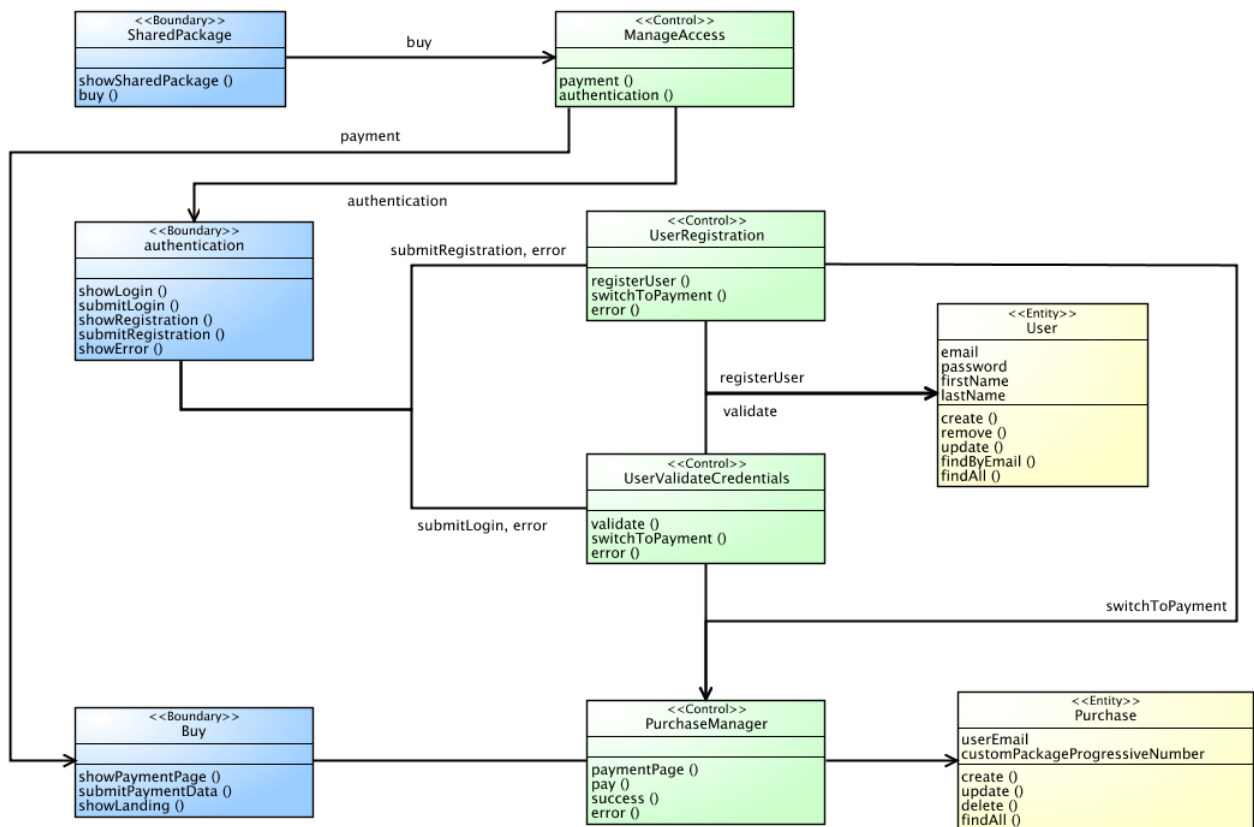
## Scelta del pacchetto

Il seguente grafico descrive le funzionalità che costituiscono il fulcro dell'applicazione web che si intende realizzare. Dal diagramma si può notare il flusso e l'insieme di componenti coinvolti all'interno di un processo di ricerca, selezione, personalizzazione, acquisto o inserimento in giftList e condivisione di un pacchetto viaggio. I controllori che compaiono all'interno del diagramma incarnano le principali funzionalità appena descritte.



## Visione di un pacchetto condiviso

Il diagramma mostra quali siano le entità coinvolte nei vari livelli in un processo di visione ed eventuale acquisto di un pacchetto di cui si è ricevuta la condivisione. Il sistema dovrà garantire all'utente gli aspetti di visualizzazione del pacchetto condiviso come evidenziato dalla boundary SharedPackage. Dovrà però garantire anche la possibilità di acquistare e dunque dovrà innanzitutto verificare che l'utente sia correttamente autenticato, e poi dovrà procedere alla gestione degli aspetti legati al pagamento.

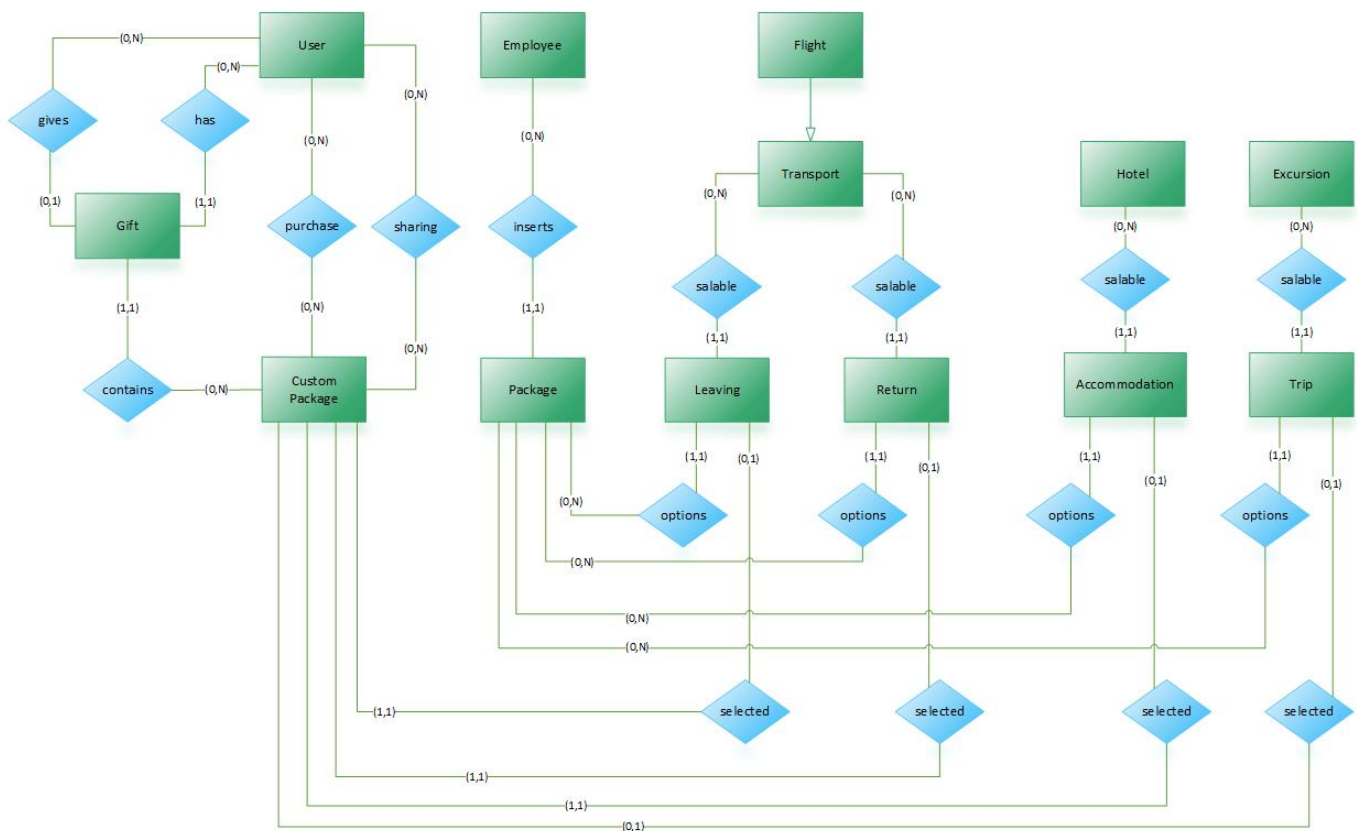




## 4 | Analisi dei dati

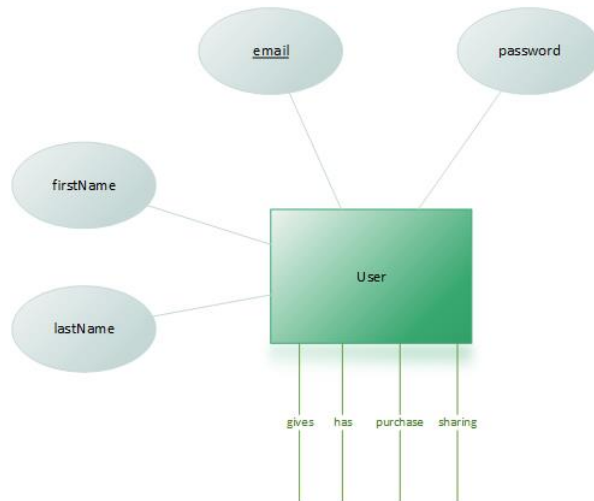
### 4.1 Progettazione concettuale

Viene di seguito riportato il diagramma ER che fornisce una panoramica ad alto livello di astrazione dello schema concettuale della base di dati. Per fornire una rappresentazione più concisa e tale da risultare leggibile nella pagina si è deciso di omettere gli attributi e mostrare inizialmente soltanto le relazioni tra entità. Nella successiva sezione seguirà il dettaglio degli attributi e delle scelte di progettazione.



### 4.1.1 Entità e attributi

#### User

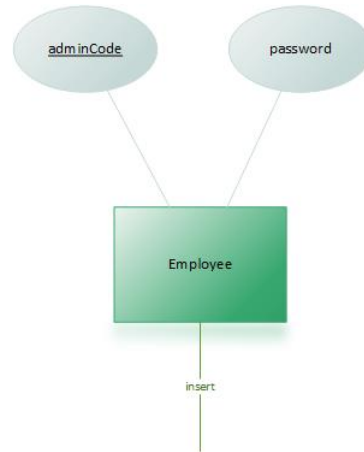


L'entità User rappresenta un cliente del portale TravelDream, ovvero un utente registrato al servizio. La base di dati di TravelDream non contiene riferimenti a utenti non registrati.

#### Attributi

- **email (chiave primaria):** stringa contenente l'indirizzo email dell'utente e che lo identifica univocamente. Viene fornita dall'utente stesso in fase di registrazione.
- **password:** stringa che contiene la password scelta dall'utente.
- **firstName:** stringa che contiene il nome dell'utente.
- **lastName:** stringa che contiene il cognome dell'utente.

## Impiegato

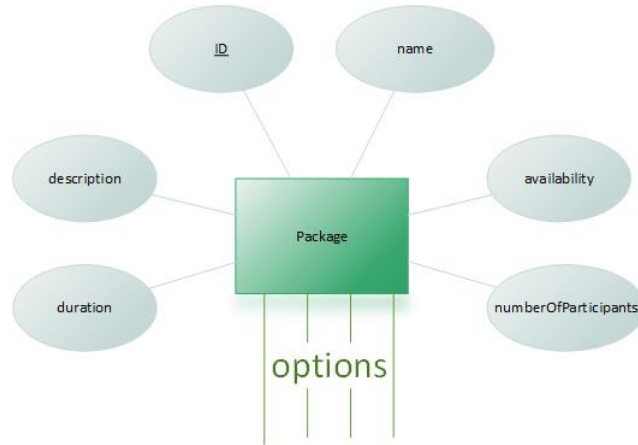


L'entità Employee rappresenta il generico impiegato della TravelDream Co..

### Attributi

- **adminCode (chiave primaria):** codice necessario all'impiegato per autenticarsi nel portale.
- **password:** stringa che contiene la password dell'impiegato.

## Package

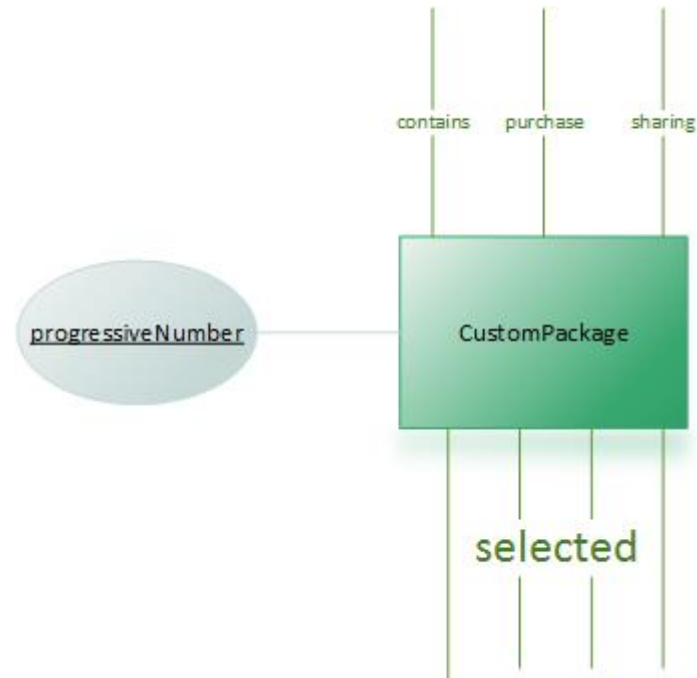


L'entità Package rappresenta una soluzione di viaggio venduta da Travel-Dream Co. comprendente varie opzioni per le tipologie di prodotti base offerti.

### Attributi

- **ID (chiave primaria):** codice alfanumerico che identifica univocamente un pacchetto.
- **name:** stringa che contiene il nome del pacchetto.
- **description:** descrizione della soluzione di viaggio offerta dal pacchetto.
- **duration:** numero che indica la durata (in giorni) della soluzione di viaggio offerta dal pacchetto.
- **availability:** booleano che indica se il pacchetto è disponibile o meno alla vendita.
- **numberOfParticipants:** numero che indica il numero di partecipanti per cui la soluzione di viaggio offerta dal pacchetto è stata pensata.

## Custom package

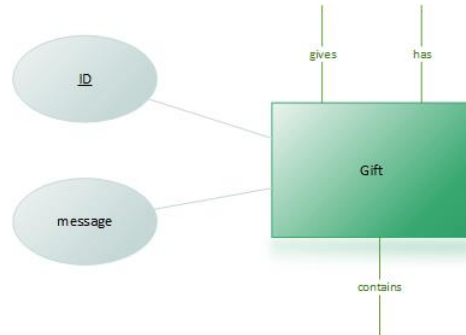


L'entità Custom Package rappresenta un viaggio offerto dalla TravelDream Co. in cui sono stati selezionati un trasporto di andata, un trasporto di ritorno, un hotel e opzionalmente un'escursione.

### Attributi

- **progressiveNumber** (chiave primaria): numero che enumera e identifica univocamente un pacchetto personalizzato.

## Gift

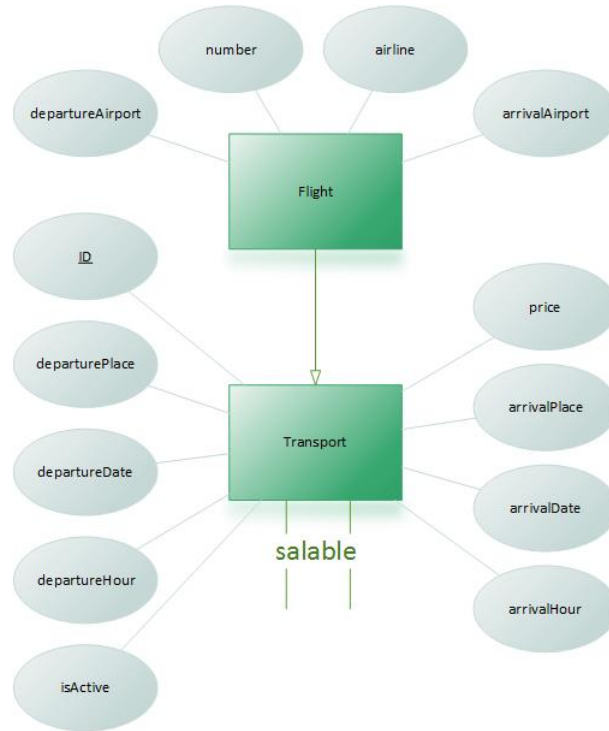


L'entità Gift rappresenta un regalo contenuto nella gift list di un utente.

### Attributi

- textttID (chiave primaria): codice alfanumerico (stringa) che identifica univocamente un regalo.
- textttmessage: stringa che contiene un messaggio di auguri associato al pacchetto.

## Transport



L'entità **Transport** rappresenta l'omonima risorsa base. L'entità **Flight** rappresenta un volo aereo, come particolare mezzo di trasporto. Per questo motivo la relazione è stata modellata tramite una gerarchia, esprimendo il fatto che **Flight** è specializzazione dell'entità **Transport**. Attualmente Travel-Dream supporta solo la vendita di voli aerei, ma sta progettando l'estensione del servizio ad altri mezzi di trasporto quali treni etc.. La base di dati risulta pertanto pronta ad affrontare il cambiamento con sole poche mirate modifiche.

### Attributi

#### Flight

- **number**: numero che identifica un volo aereo.
- **airline**: stringa che contiene il nome della compagnia aerea che fornisce il volo.
- **departureAirport**: stringa che contiene il nome dell'aeroporto di partenza del volo.

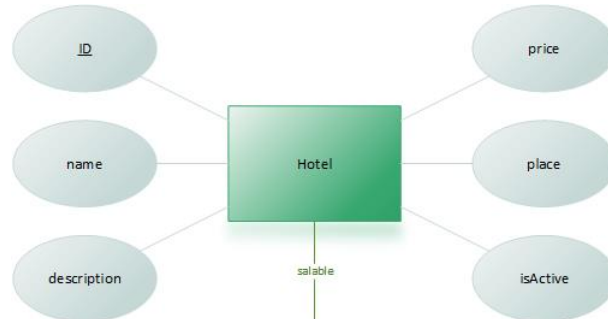
- **arrivalAirport**: stringa che contiene il nome dell'aeroporto di arrivo del volo.

#### Transport

- **ID (chiave primaria)**: codice alfanumerico (stringa) che identifica univocamente un trasporto.
- **price**: numero che indica il prezzo del trasporto.
- **departureDate**: data che indica la data di partenza del trasporto.
- **departureHour**: ora che indica l'orario di partenza del trasporto.
- **arrivalDate**: data che indica la data di arrivo del trasporto.
- **arrivalHour**: ora che indica l'orario di arrivo del trasporto.
- **departurePlace**: stringa che contiene il nome del luogo di partenza del trasporto.
- **arrivalPlace**: stringa che contiene il nome del luogo di arrivo del trasporto.
- **isActive**: booleano che indica se il trasporto è disponibile o meno alla vendita.



## Hotel

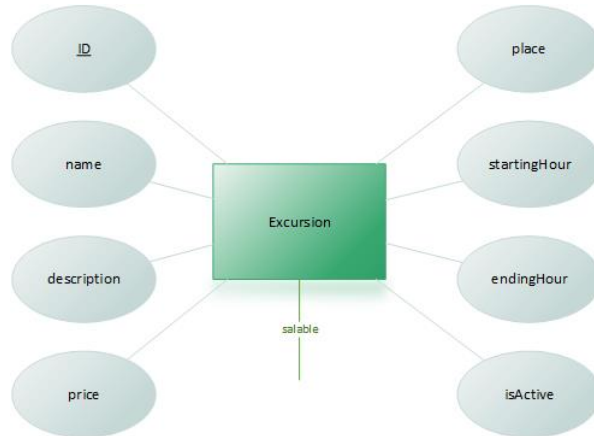


L'entità Hotel rappresenta l'omonima risorsa base.

### Attributi

- **ID (chiave primaria):** codice alfanumerico (stringa) che identifica univocamente un hotel.
- **name:** stringa che contiene il nome dell'hotel.
- **description:** stringa che contiene una descrizione dell'hotel.
- **price:** numero che indica il prezzo dell'hotel.
- **place:** stringa che contiene il nome del luogo di ubicazione dell'hotel.
- **isActive:** booleano che indica se l'hotel è disponibile o meno.

## Excursion

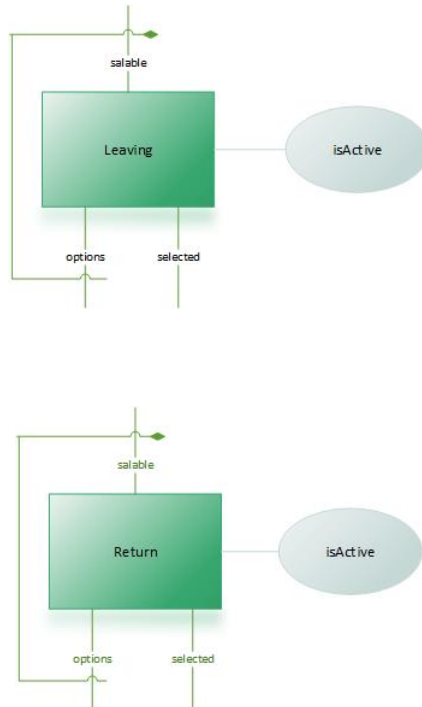


L'entità Excursion rappresenta l'omonima risorsa base.

### Attributi

- **ID (chiave primaria):** codice alfanumerico (stringa) che identifica univocamente un'escursione.
- **name:** stringa che contiene il nome dell'escursione.
- **description:** stringa che contiene una descrizione dell'escursione.
- **price:** numero che indica il prezzo dell'escursione.
- **place:** stringa che contiene il nome del luogo in cui si terrà l'escursione.
- **startingHour:** ora che indica l'orario di inizio dell'escursione.
- **endingHour:** ora che indica l'orario di fine dell'escursione.
- **isActive:** booleano che indica se l'escursione è disponibile o meno.

## Leaving, Return

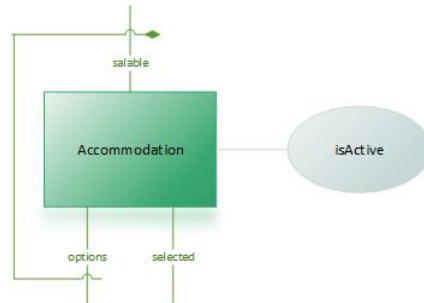


L'entità Leaving rappresenta l'associazione tra un pacchetto e uno specifico trasporto di andata. L'entità Return rappresenta l'associazione tra un pacchetto e uno specifico trasporto di ritorno. Le chiavi primarie di entrambe le entità sono per l'appunto la coppia pacchetto, mezzo di trasporto.

### Attributi

- **isActive**: booleano che indica se l'opzione di trasporto è attiva per il pacchetto a cui è associato.

## Accommodation

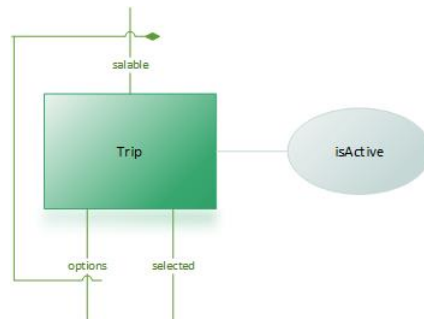


L'entità Accommodation rappresenta un' associazione tra un pacchetto ed un hotel. La chiave primaria di questa entità è per l'appunto la coppia pacchetto, hotel.

### Attributi

- **isActive**: booleano che indica se l'opzione di alloggio è attiva per il pacchetto a cui è associato.

## Trip



L'entità Trip rappresenta l'associazione tra un pacchetto ed una escursione. La chiave primaria di questa entità è la coppia pacchetto, escursione.

### Attributi

- **isActive**: booleano che indica se l'opzione di visita turistica è attiva per il pacchetto a cui è associata.

### 4.1.2 Relazioni

**inserts (Employee, Package):** Associa un impiegato ad i pacchetti da lui inseriti, ed i pacchetti all'impiegato che li ha inseriti.

**purchase (User, CustomPackage):** Associa un utente ad i pacchetti personalizzati da lui acquistati, ed un pacchetto personalizzato agli utenti che lo hanno acquistato.

**sharing (User, CustomPackage):** Associa un utente ad i pacchetti personalizzati da lui condivisi, ed un pacchetto personalizzato agli utenti che lo hanno condiviso.

**has (User, Gift):** Associa un utente ad i regali presenti nella propria gift list, ed un regalo all'utente che lo possiede in gift list.

**gives (User, Gift):** Associa un utente ad i regali da lui effettuati, ed un regalo all'utente che lo ha effettuato.

**constains (Gift, CustomPackage):** Associa un regalo al pacchetto personalizzato che contiene, ed un pacchetto personalizzato ad i regali che lo contengono.

**options (Package, Leaving):** Associa un pacchetto ad un insieme di partenze disponibili per quel pacchetto, ed una partenza ad un determinato pacchetto.

**options (Package, Return):** Associa un pacchetto ad un insieme di ritorni disponibili per quel pacchetto, ed un ritorno ad un determinato pacchetto.

**options (Package, Accommodation):** Associa un pacchetto ad un insieme di alloggi disponibili per quel pacchetto, ed un alloggio ad un determinato pacchetto.

**options (Package, Trip):** Associa un pacchetto ad un insieme di visite turistiche disponibili per quel pacchetto, ed una visita turistica ad un determinato pacchetto.

**selected (CustomPackage, Leaving):** Associa un pacchetto personalizzato alla partenza selezionata per quel pacchetto, ed una partenza ad un determinato pacchetto personalizzato.

**selected (CustomPackage, Return):** Associa un pacchetto personalizzato al ritorno selezionato per quel pacchetto, ed un ritorno ad un determinato

pacchetto personalizzato.

**selected (CustomPackage, Accommodation):** Associa un pacchetto personalizzato all'alloggio selezionato per quel pacchetto, ed un alloggio ad un determinato pacchetto personalizzato.

**selected (CustomPackage, Trip):** Associa un pacchetto personalizzato alla visita turistica selezionata per quel pacchetto, ed una visita turistica ad un determinato pacchetto personalizzato. Questa relazione avrà un attributo "date" indicante la data scelta dall'utente per la visita turistica selezionata.

**salable (Transport, Leaving):** Associa un trasporto ad un insieme di partenze che utilizzano quel trasporto, ed una partenza ad un determinato trasporto.

**salable (Transport, Return):** Associa un trasporto ad un insieme di ritorni che utilizzano quel trasporto, ed un ritorno ad un determinato trasporto.

**salable (Hotel, Accommodation):** Associa un hotel ad un insieme di alloggi, ed un alloggio ad un determinato hotel.

**salable (Excursion, Trip):** Associa un'escursione ad un insieme di visite turistiche, ed una visita turistica ad una determinata escursione.

## 4.2 Progettazione logica

La progettazione logica della base di dati consiste nella traduzione dello schema concettuale in uno schema logico atto ad organizzare i dati seguendo il modello relazionale. Di seguito è riportato lo schema logico derivato dal diagramma ER.

**User** (email, password, firstName, lastName)

**Employee** (adminCode, password)

**Package** (ID, name, description, duration, numberOfParticipants, availability, employeeAdminCode)

**CustomPackage** (progressiveNumber, leavingPackageID, leavingTransportID, returnPackageID, returnTransportID, accomodationPackageID, accomodationHotelID, tripPackageID, tripExcursionID, selectedTripDate)

**Purchase** (userEmail, customPackageProgressiveNumber)

**Sharing** (userEmail, customPackageProgressiveNumber)

**Gift** (ID, message, customPackageProgressiveNumber, ownerID, buyerID)

**Flight** (ID, number, airline, departureAirport, arrivalAirport, departurePlace, arrivalPlace, departureDate, arrivalDate, departureHour, arrivalHour, price, isActive)

**Hotel** (ID, name, description, place, price, isActive)

**Excursion** (ID, name, description, place, startingHour, endingHour, price, isActive)

**Leaving** (packageID, transportID, isActive)

**Return** (packageID, transportID, isActive)

**Accommodation** (packageID, hotelID, isActive)

**Trip** (packageID, excursionID, isActive)

## 5 | Appendice

### 5.1 Modifiche al RASD

Di seguito si riportano le rettifiche e le precisazioni ricavate in fase di design, rispetto a quanto espresso in precedenza all'interno del documento di analisi dei requisiti:

- Alla definizione di pacchetto è stato aggiunto un attributo luogo, assente nel documento di analisi.
- A ciascun pacchetto è stata associata una durata fissa.
- Nel RASD è stata indicata un'eccezione nel caso in cui la cancellazione di un prodotto base generasse dei conflitti con i pacchetti esistenti. In modo particolare si chiedeva l'azione da intraprendere per ciascun pacchetto coinvolto nella cancellazione. E' stato deciso che non si provvede alla visualizzazione dei pacchetti in conflitto ma semplicemente la risorsa viene disattivata: ciò significa che non è più selezionabile dall'utente in fase di personalizzazione del pacchetto ma rimane visibile nei pacchetti che sono già stati acquistati.
- In fase di design si è deciso di non considerare la disponibilità dei prodotti base, affidandone il controllo ad un sistema non facente parte la piattaforma TravelDream.

### 5.2 Ore di lavoro

Per la stesura del presente documento e la revisione del documento di analisi sono state necessarie 155 ore di lavoro equamente distribuite fra i membri del gruppo.