

DESIGN DOCUMENT

Dario Casula, Alessandro Chetta

Prof. Luca Mottola

Progetto di Ingegneria del Software 2

A.A. 2013-2014 Politecnico di Milano



*TravelDream*

# Indice Generale

---

1.	Introduzione.....	3
2.	Architettura dell'applicazione.....	4
2.1.	Livelli Architeturali.....	4
3.	Struttura del Database.....	5
3.1.	Schema Entità Relazioni.....	5
3.2.	Schema Logico.....	7
4.	User Experience nella navigazione.....	9
4.1.	Utente non Registrato.....	9
4.2.	Utente Registrato.....	11
4.3.	Impiegato.....	13
5.	Schemi BCE.....	16
5.1.	Home Generale.....	16
5.2.	Utente Registrato.....	17
5.3.	Impiegato.....	19
6.	Sequence Diagrams.....	20
6.1.	Utente non Registrato.....	20
6.2.	Utente Registrato.....	20
6.3.	Impiegato.....	21
	Indice delle figure.....	22



# 1 Introduzione

---

Questo documento descrive le scelte di design effettuate per la creazione del portale TravelDream, sulla base dell'analisi dei requisiti documentata nel RASD.

Sono state individuate quattro principali aree di lavoro che si rifletteranno nella struttura del presente documento di design: descrizione dell'architettura del sistema, struttura del database (schema ER), user experience nella navigazione tra le schermate del portale (schema UX), distinzione tra elementi di interfaccia, di controllo e di gestione dei dati (diagramma BCE).

Ogni area è stata analizzata e realizzata mediante una fusione delle strategie di top down e bottom up: da una visione d'insieme del sistema si è passati ad una analisi dettagliata degli elementi dedicati ai singoli attori, utente generico, utente registrato e impiegato.

Il documento si chiude con sequence diagram generali sull'utilizzo del sistema da parte degli attori, in modo da chiarire ulteriormente le scelte effettuate, eliminando ogni possibile ambiguità.

Il Design Document rappresenterà la base per l'implementazione del sistema, fornendo indicazioni precise riguardo alla realizzazione, senza dare una definizione già completa e dettagliata sulle modalità di scrittura del codice.

## 2 Architettura dell'applicazione

### 2.1 Livelli Architetturali

L'applicazione si svilupperà in diversi livelli. E' previsto il livello Web Tier e Business Tier, Data Tier, Client Tier.

Per l'implementazione dell'applicazione web è stato scelto il linguaggio java. Java Enterprise Edition 7 è il più adatto per lo sviluppo di grandi applicazioni di rete a più livelli e garantisce scalabilità, affidabilità e sicurezza. Pertanto Java EE 7 sarà la nostra piattaforma software per lo sviluppo dell'applicazione web TravelDream. Come Application Server sarà adottato GlassFish Open Source Edition 4.0 che offre pieno supporto alle specifiche Java EE 7.

Per quanto riguarda la gestione dei dati si utilizza il Relational database management system MySQL.

Gli utenti del sistema che sarà sviluppato potranno accedere ad esso tramite un browser. Le pagine saranno elaborate dal server e inviate al browser usando il protocollo http.

In questo modo ogni operazione potrà essere svolta da qualsiasi piattaforma che consente l'utilizzo di un browser. Di seguito è riportata l'architettura del sistema a livello grafico finalizzata principalmente a comunicare la funzione di ogni Tier coinvolto.

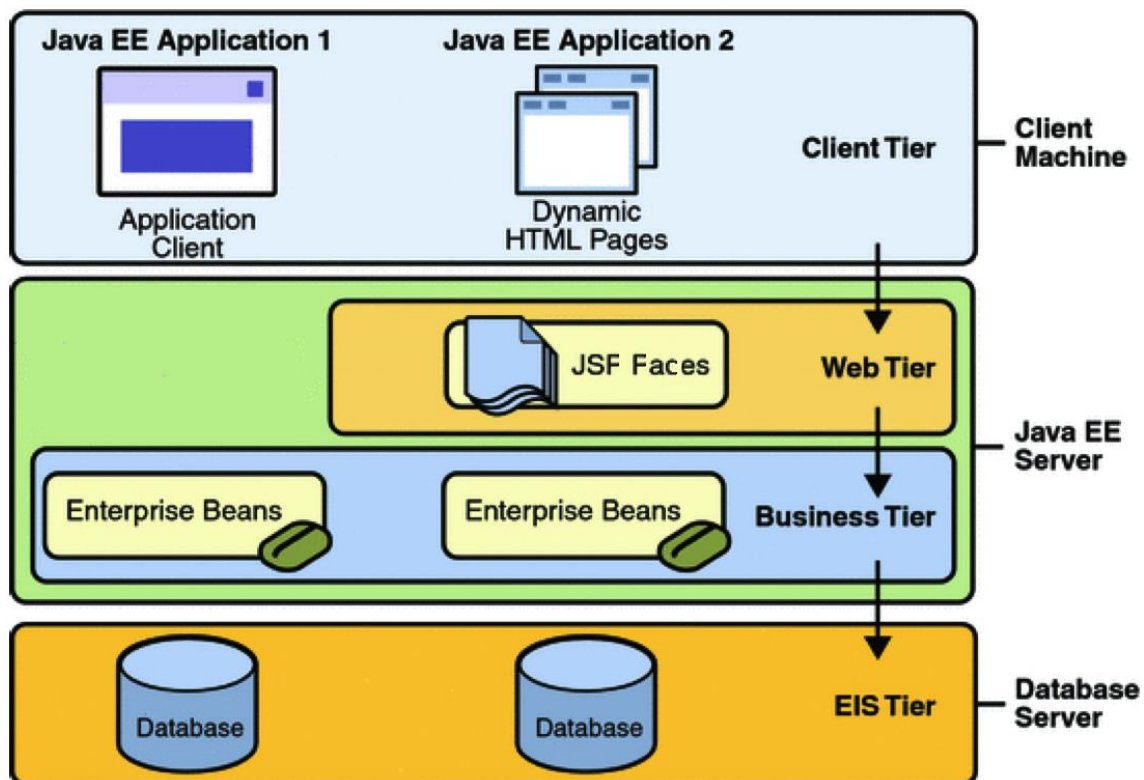


Figura 1: Livelli architetturali

## 3 Struttura del database

---

Il database relazionale di TravelDream è rappresentato di seguito con un diagramma concettuale Entità-Relazione.

Successivamente sarà proposto lo schema logico relativo allo schema ER, che sarà necessario per esplicitare nel dettaglio le tabelle che saranno effettivamente implementate nel database e i cammini di join tra esse.

### 3.1 Schema Entità Relazioni

In primo luogo si è scelto di rappresentare l'utente e l'impiegato con due entità diverse e indipendenti.

Dal momento che nel sistema non è necessario tenere traccia delle azioni compiute da un impiegato, l'entità che lo rappresenta non è connessa tramite alcuna relazione ad altre entità. In altre parole, non è necessario conoscere quale impiegato ha compiuto una modifica al sistema, anche se la modifica è memorizzata in maniera persistente. L'impiegato è identificato univocamente da un id, che sarà utilizzato per l'accesso al portale (back-end).

L'utente è identificato univocamente dall'email nel database. In questo caso è indispensabile memorizzare le relazioni tra utenti e pacchetti acquistati o personalizzati da essi, in modo tale da rendere sempre accessibile all'utente i suoi pacchetti comprati e i suoi pacchetti personalizzati. Questo è rappresentato nello schema mediante le due relazioni *Purchases* e *Personalizes*.

Si è scelto di rappresentare il *Personalized Package* come sotto entità di *Package*, in questo modo è possibile rendere il pacchetto personalizzabile dall'utente senza modificare quello di partenza.

Il *Package* possiede degli attributi che lo caratterizzano tra i quali il suo id che lo identifica in maniera univoca all'interno della base di dati. Il *Package* è abinato ai suoi prodotti base tramite delle relazioni *Has*. I prodotti base *Trip*, *Accommodation* e *Transportation Mean* risultano parte di una gerarchia la cui entità padre è *BasicProduct*, dalla quale i prodotti base ereditano tutti gli attributi.

Nello schema sono state omesse le cardinalità per favorire la velocità di comprensione delle entità e delle relazioni fra esse, tuttavia le cardinalità risultano essere:

User (1) Purchases (0,*) Package
User (1) Personalizes (0,*) PersonalizedPackage
Package (1,*) Has (1,*) Trip
Package (1,*) Has (1,*) Accommodation
Package (1,*) Has (1,*) TransportationMean

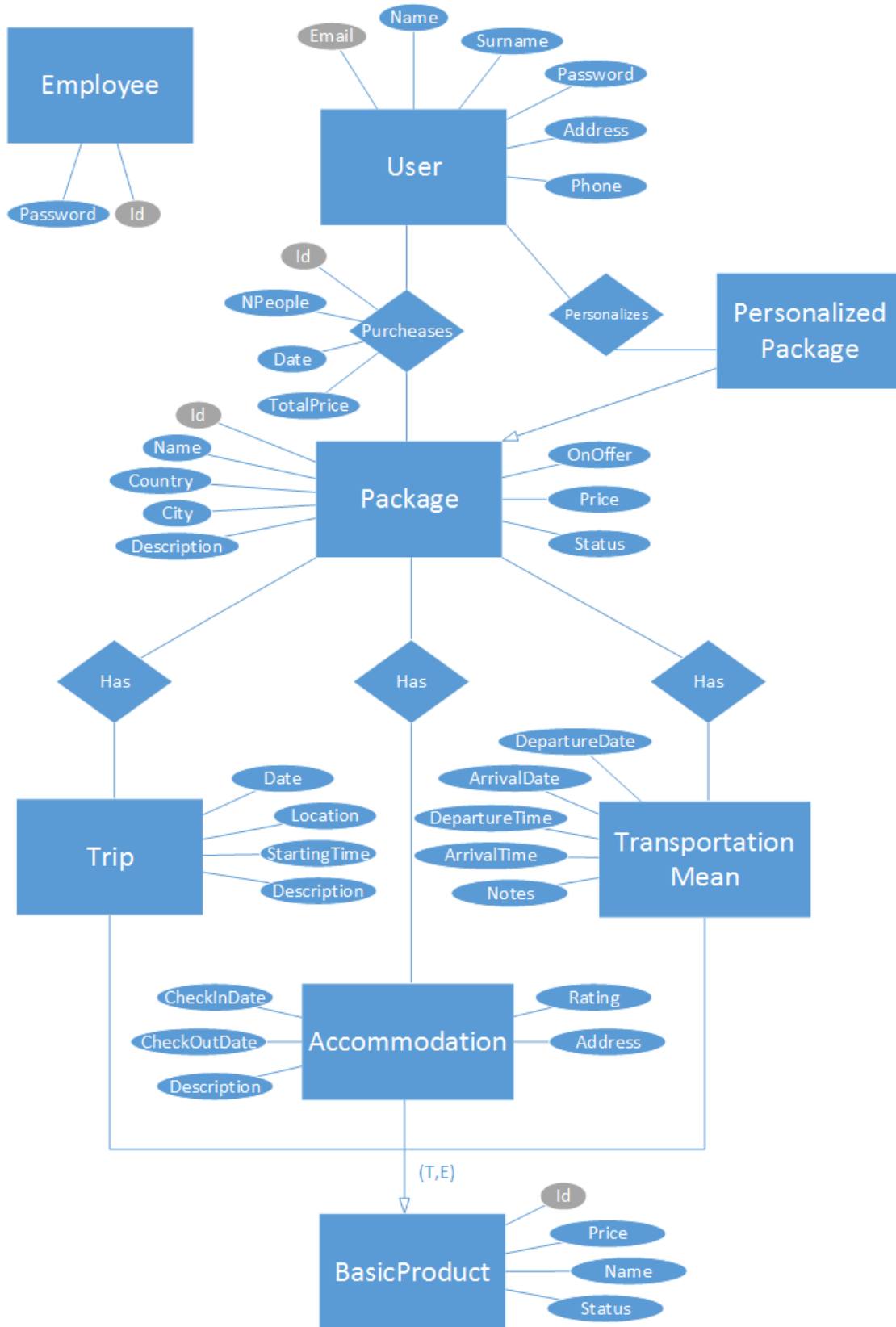


Figura 2: Schema ER

### 3.2 Schema Logico

Il modello logico descrive esattamente le tabelle relazionali che saranno presenti nel database, pertanto è stato necessario ristrutturare il modello concettuale per eliminare le gerarchie e tradurre le relazioni in tabelle relazionali.

Le entità *Package* e *PersonalizedPackage* sono state accorpate, quindi sono stati aggiunti due ulteriori attributi oltre a quelli presenti nell'entità *Package*. Gli attributi aggiunti sono *Personalized* e *UserEmail*. *Personalized* è un attributo booleano che indica se la tupla si riferisce ad un pacchetto personalizzato o meno. *UserEmail* è la chiave esterna che serve a mantenere in memoria la relazione uno a molti tra *User* e *PersonalizedPackage* (o tra *User* e *Package*).

La relazione *Purchases* è stata rappresentata con la tabella *PurchasedPackage* che oltre agli attributi della relazione contiene anche le chiavi esterne *UserEmail* e *PackageId*, che indicano rispettivamente l'utente che ha acquistato il pacchetto e il pacchetto che è stato acquistato.

L'entità *Employee* è rimasta invariata.

La gerarchia tra *BasicProduct* e *Trip*, *Accommodation* e *TransportationMean* viene accorpata eliminando l'entità *BasicProduct* che di fatto non avrà nessuna istanza nel sistema. Pertanto gli attributi di *BasicProduct* vengono ripetuti in tutte le entità figlie. A questo punto è stato necessario stabilire le relazioni tra i prodotti base e i *Package* e tra i prodotti base e i *PersonalizedPackage*.

Per i *Package* è stato deciso di rappresentare le opzioni di scelta con una tabella che mette in relazione due chiavi esterne, la prima indica il l'id del pacchetto a cui si riferisce l'opzione e la seconda indica il prodotto base che è stato messo a disposizione per il pacchetto. Si fa notare che, dal momento che non ha senso abbinare due volte lo stesso prodotto base allo stesso pacchetto, le due chiavi esterne compongono la chiave primaria della tabella appena descritta.

Per i *PersonalizedPackage* lo scopo della relazione tra essi e i prodotti base è quello di mantenere in memoria i prodotti base scelti da un utente per un determinato pacchetto. Quindi sono state create delle tabelle analoghe a quelle descritte per la relazione pacchetto - prodotto base con l'aggiunta di un id, perchè in questo caso le due chiavi esterne non costituiscono la chiave primaria, inquanto un pacchetto può essere personalizzato da più utenti.

I cammini di join tra tutte le tabelle consentono di creare tutte le views necessarie.



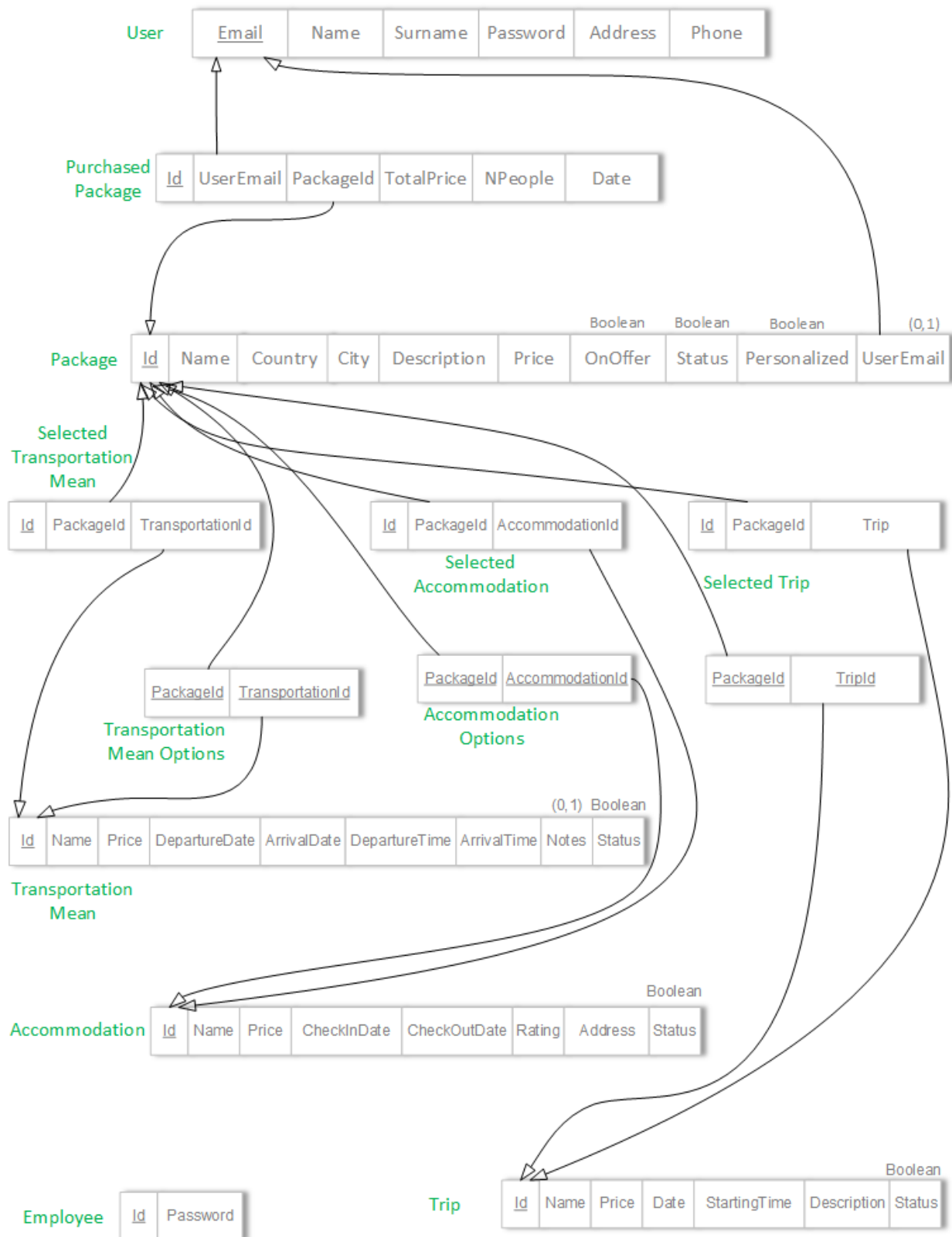


Figura 3: Schema Logico del Database

## 4 User Experience nella navigazione

In questa sezione verranno mostrati i diagrammi UX (User eXperience) che descrivono la navigazione nel portale. Sono mostrate le schermate di navigazione e i form di input nei quali si evidenzia l'interazione da parte dell'utente.

---

### 4.1 Utente non Registrato

Nel momento della connessione al portale TravelDream viene visualizzata l'Homepage generale del sito. In questa schermata sono presenti le opzioni base utilizzabili da un utente non registrato al sistema: la visualizzazione di un pacchetto condiviso e la registrazione al portale. Entrambe le funzioni visualizzano dei form per consentire l'interazione con l'utente. L'opzione di registrazione richiede le informazioni personali necessarie a creare un nuovo utente registrato nel database, mentre la visualizzazione di un pacchetto condiviso richiede l'inserimento del codice univoco relativo al pacchetto.

L'Homepage generale offre inoltre la possibilità ad un utente già registrato di effettuare il login al sistema.

Il sistema, all'atto del login, riconosce in base alle credenziali inserite se ad effettuare l'accesso è un utente registrato o un impiegato, visualizzando quindi la relativa homepage. Nel caso di una nuova registrazione da parte di un cliente dell'agenzia, viene visualizzata l'homepage di un utente registrato.

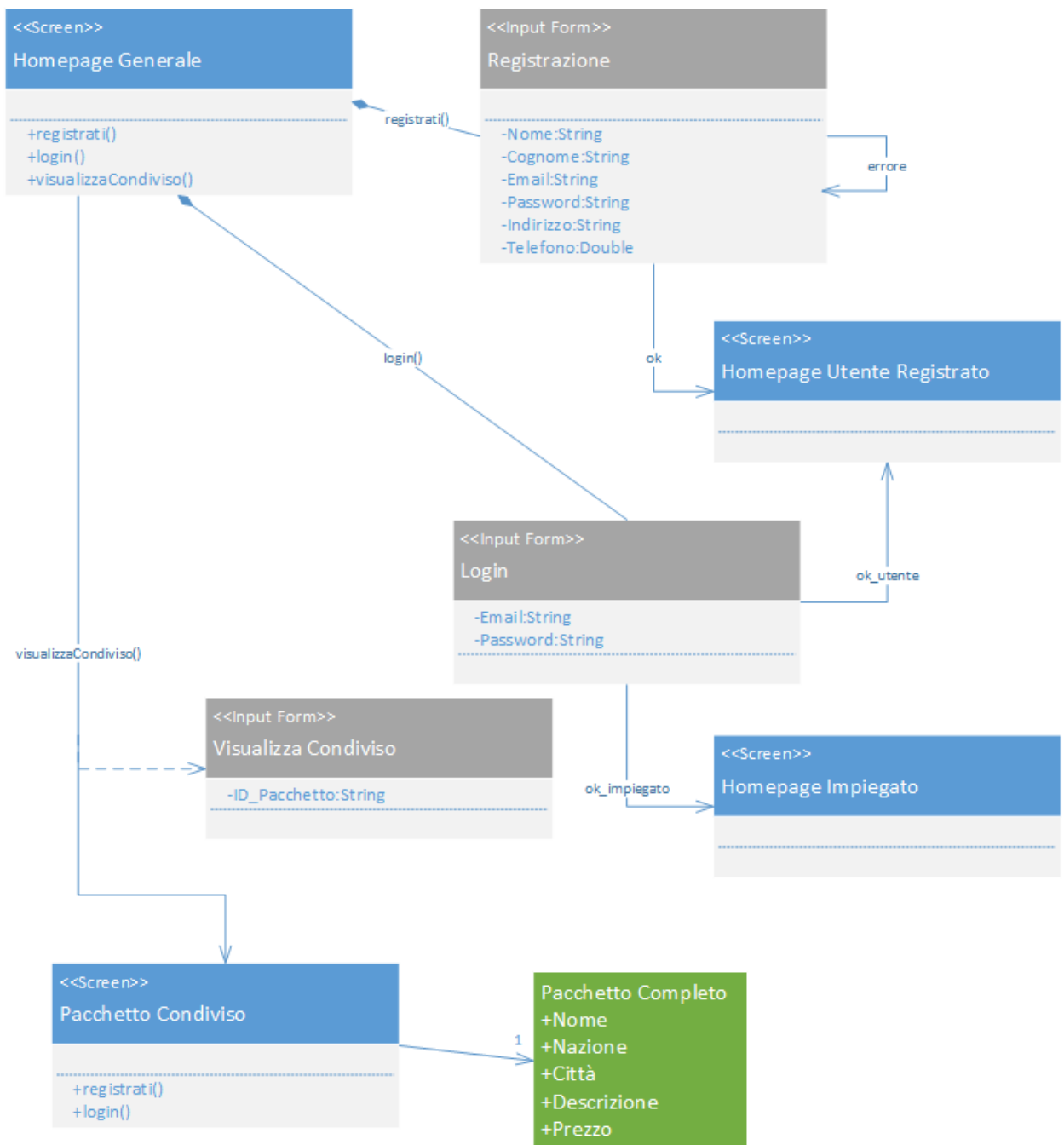


Figura 4: Diagramma UX Homepage Generale e Utente non Registrato

## 4.2 Utente Registrato

Dopo il login, l'utente registrato riconosciuto viene indirizzato alla propria homepage personale, nella quale sono immediatamente mostrati i pacchetti "in offerta", dichiarati tali dall'impiegato che li ha inseriti.

Questa schermata presenta un menù di opzioni base, il quale sarà presente in ogni schermata della navigazione. Sarà così possibile accedere in ogni momento alla propria homepage, alla WishList, allo storico acquisti o al proprio profilo.

Dalla homepage è possibile stabilire quali pacchetti devono essere visualizzati, grazie a termini chiave inseriti tramite apposite finestre a tendina che mostrano le opzioni di ricerca disponibili. Dalla schermata iniziale è possibile visualizzare un pacchetto nel dettaglio accedendo alla schermata di visualizzazione oppure procedere direttamente all'acquisto, indirizzandosi alla schermata di definizione dei dettagli di vendita. È inoltre possibile, direttamente dalla homepage, inserire un codice univoco per la visualizzazione di un pacchetto condiviso; sarà poi possibile salvarlo nella WishList o procedere all'acquisto.

La funzione di personalizzazione di un pacchetto permette l'accesso ad una schermata nella quale sono visualizzati i dettagli del pacchetto in questione e tutte le opzioni disponibili di modifica dei prodotti base contenuti. L'utente può selezionare i prodotti desiderati tramite bottoni di selezione (radiobutton o tecniche analoghe) e procedere ad aggiornare la visualizzazione dei dettagli del pacchetto, il quale può essere salvato o acquistato.

La schermata di visualizzazione della WishList permette l'accesso ai dettagli di un pacchetto o all'acquisto diretto, è possibile quindi definire i dettagli della vendita, selezionando il numero di pacchetti da acquistare e visualizzandone il prezzo totale. È importante notare che i dettagli di un pacchetto presente nella WishList comprendono un codice univoco che l'utente registrato può condividere tramite un servizio esterno al portale, in modo da condividere il pacchetto con i propri amici.

Dallo storico acquisti è possibile visualizzare i dettagli di un pacchetto acquistato precedentemente e procedere ad un nuovo acquisto dello stesso.

La pagina dedicata al profilo riassume tutte le informazioni personali dell'utente e permette un aggiornamento delle stesse tramite un apposito form di input.

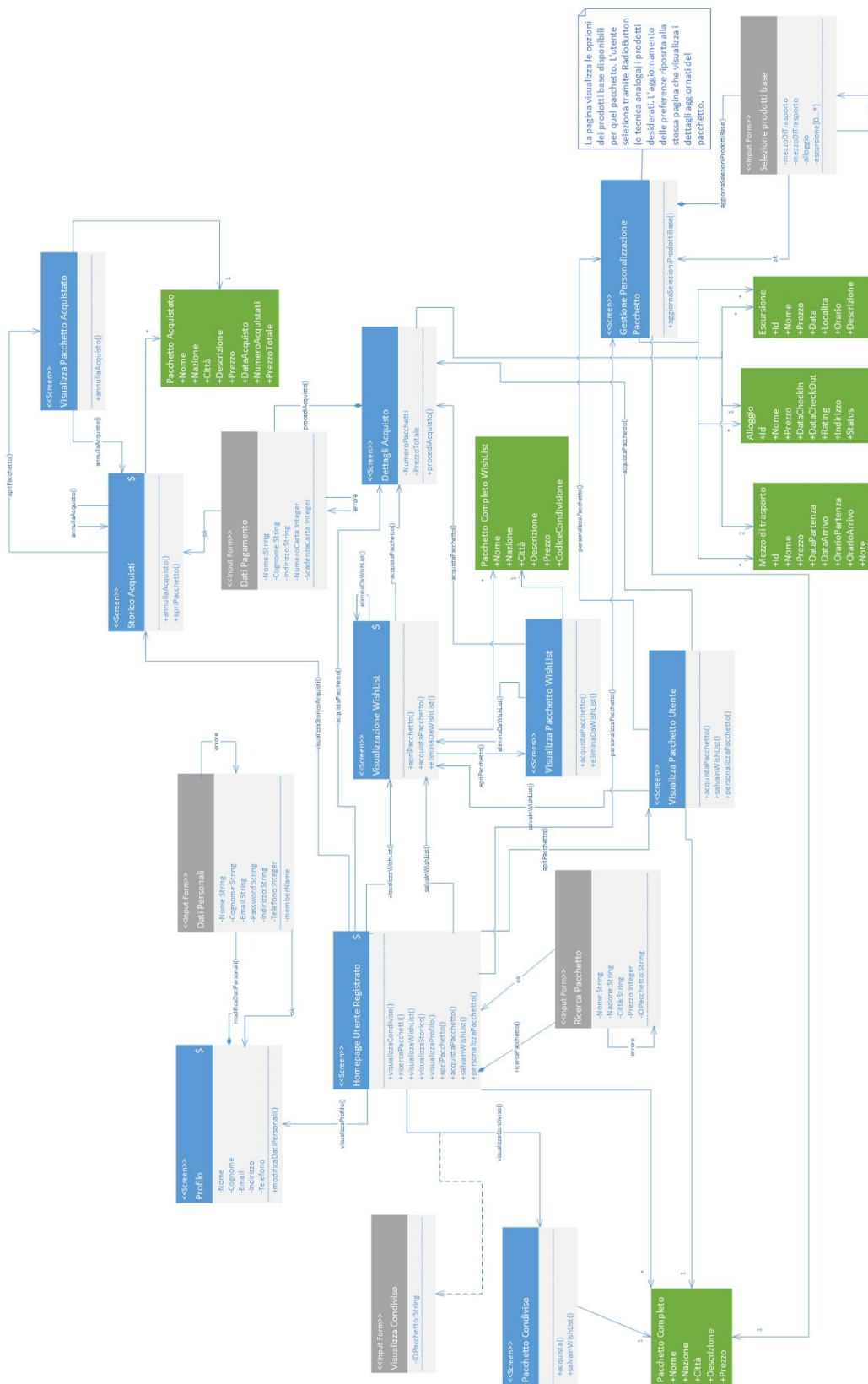


Figura 5: Diagramma UX Utente Registrato

### 4.3 Impiegato

L'impiegato dopo aver inserito le sue credenziali viene reindirizzato alla *Homepage Impiegato*. In questa screen è presente il menù che sarà accessibile da ogni screen e form dell'impiegato per rendere veloce la navigazione tra le principali attività che può svolgere. Il menu comprende una voce *homepage impiegato* (non riportata nello schema), e due voci che portano l'impiegato alle due screen che raccolgono le principali attività dell'impiegato: *crea pacchetto* e *gestione prodotti base*.

Nella homepage, oltre al menù, sono presenti tutti i pacchetti contenuti nel sistema opportunamente impaginati. L'operazione di ricerca contenuta in questa pagina consiste nella definizione da parte dell'impiegato di alcuni filtri sugli attributi dei pacchetti, l'impiegato può infatti scegliere se visualizzare solo i pacchetti ancora in vendita, quelli in offerta o quelli relativi ad una determinata meta turistica. In questo modo la ricerca e i suoi risultati sono compresi nella *homepage impiegato*. Per ogni pacchetto mostrato nell'elenco l'impiegato può selezionare tre opzioni: elimina, modifica, visualizza. Tali funzioni hanno come parametro l'id del pacchetto in modo tale da poterlo identificare lato server e generare l'opportuno output. Nel caso di elimina il pacchetto viene reso non più disponibile e viene visualizzata nuovamente la *homepage impiegato*.

Il form *Crea Pacchetto* è di fondamentale importanza per la gestione del sistema da parte dell'impiegato. Essa infatti consente all'impiegato di inserire nel sistema un nuovo pacchetto con tutti i suoi attributi. L'impiegato è tenuto a inserire i prodotti base che saranno disponibili all'utente al momento della personalizzazione del pacchetto. La coerenza di questi dati non è controllata dal sistema ma il controllo è lasciato all'impiegato.

Il form *Modifica Pacchetto* è accessibile da due screen diverse. Infatti, oltre alla *home page* dove è presente la funzione modifica per ogni pacchetto dell'elenco, la stessa funzione è viene resa disponibile nella screen *Visualizza Pacchetto*. Il form *Modifica Pacchetto* consente all'impiegato di modificare gli attributi del pacchetto o di aggiungere o eliminare le relazioni che esso ha con i vari prodotti base ad esso associati.

La screen *Visualizza Pacchetto* rende possibile una esibizione approfondita di tutti gli attributi e le caratteristiche del pacchetto, inoltre sono calcolati lato server dei dati derivati riguardanti il pacchetto come il numero di volte in cui il pacchetto è stato venduto. (Nello Ux *NumeroPacchettiAcquistati* è stato riportato come attributo dell'entità pacchetto, ma in realtà esso è generato dinamicamente dal server analizzando il database).

Anche in questa posizione l'impiegato può scegliere di modificare il pacchetto o eliminarlo, nel primo caso viene visualizzato il form *Modifica Pacchetto* nel secondo il pacchetto viene disabilitato e l'impiegato sarà reindirizzato sulla *homepage impiegato*.

La screen *Gestione Prodotti Base* rappresenta il pannello di controllo dei prodotti base. Da questa screen l'impiegato può creare un prodotto base, modificarlo ed eliminarlo. La ricerca è gestita analogamente alla ricerca dei pacchetti nella *homepage impiegato*. Nello schema sono stati omessi degli screen che visualizzassero in dettaglio gli attributi del singolo prodotto base (funzione analoga al *Visualizza Pacchetto*) per questioni di leggibilità.





## 5 Schemi BCE

In questa sezione verranno mostrati gli schemi BCE (Boundary Control Entity) che rappresentano la divisione dell'applicazione in tre livelli distinti: elementi di interfacciamento con l'utente, elementi di gestione della logica applicativa ed elementi di gestione dei dati contenuti nel sistema.

Gli schemi sono riportati in tre differenti paragrafi: Homepage Generale, Utente Registrato e Impiegato. Questa ulteriore divisione permette un'analisi più dettagliata senza creare una inutile confusione. È necessario però sottolineare il fatto che elementi appartenenti a schemi diversi non sono necessariamente elementi diversi, al contrario, elementi con lo stesso nome sono da considerarsi lo stesso elemento. È il caso, per esempio, del Controller "Gestione Pacchetti" o delle Entity "User" e "Package", nonché del Boundary "Visualizza Pacchetto". Questi elementi comuni sono coinvolti quindi in tutte le funzionalità degli attori nei quali compaiono, funzioni che però sono esplicitate solo nei relativi paragrafi.

---

### 5.1 Home Generale

Lo schema mostra la divisione in livelli delle funzioni presenti nella homepage generale, comprendendo le funzioni accessibili da un utente generico, anche non registrato.

L'homepage generale costituisce un boundary di interfacciamento con l'utente, permettendo il login, la registrazione o la visualizzazione di un pacchetto condiviso.

Nel momento del login, l'homepage comunica con la logica applicativa che gestisce l'accesso degli utenti e li indirizza alla homepage personale relativa. Questo livello di controllo interviene anche nel caso di una nuova registrazione al sistema, accedendo al livello in cui sono presenti i dati e aggiungendo un nuovo utente al database.

Nel caso in cui venga richiesto di visualizzare un pacchetto condiviso interviene la logica che gestisce i pacchetti, la quale ricerca il pacchetto desiderato e lo trasferisce alla homepage generale del livello superiore.



Figura 7: Schema BCE Homepage Generale e Utente non Registrato

## 5.2 Utente Registrato

Lo schema mostra la divisione in livelli delle funzionalità accessibili da parte di un utente registrato che ha effettuato il login.

Dalla homepage dell'utente è possibile accedere alle funzioni del sistema, le quali saranno fornite grazie al livello della logica applicativa. Questo livello è stato suddiviso nei gestori del profilo, dei pacchetti e degli acquisti.

Il primo si occupa di tutte le funzionalità inerenti il profilo utente e la sua modifica. Il gestore del profilo gestisce quindi i dati riguardanti gli utenti, in quanto deve averne accesso per la visualizzazione a video e nel caso di modifica dei dati personali da parte dell'utente.

Il secondo gestisce tutto ciò che richiede l'uso di informazioni riguardanti i pacchetti, è il mediatore tra le boundary e i dati dei packages.

Il gestore dei pacchetti ne permette la visualizzazione, la personalizzazione, il salvataggio in WishList e l'inserimento nello storico acquisti. Il Controller ha quindi accesso ai dati riguardanti i Packages e ai PurchasedPackages. Questo gestore si occupa inoltre di tutte le funzionalità necessarie all'attore impiegato, le quali sono analizzate in modo dettagliato al punto "5.3 Impiegato" di questo documento.

Il terzo gestisce gli acquisti, permettendo la scelta del numero di pacchetti da acquistare e gestendo l'inserimento dei dettagli di pagamento. Questo Controller ha accesso ai dati riguardanti i pacchetti acquistati in quanto svolge le funzioni di inserimento in tabella degli stessi.

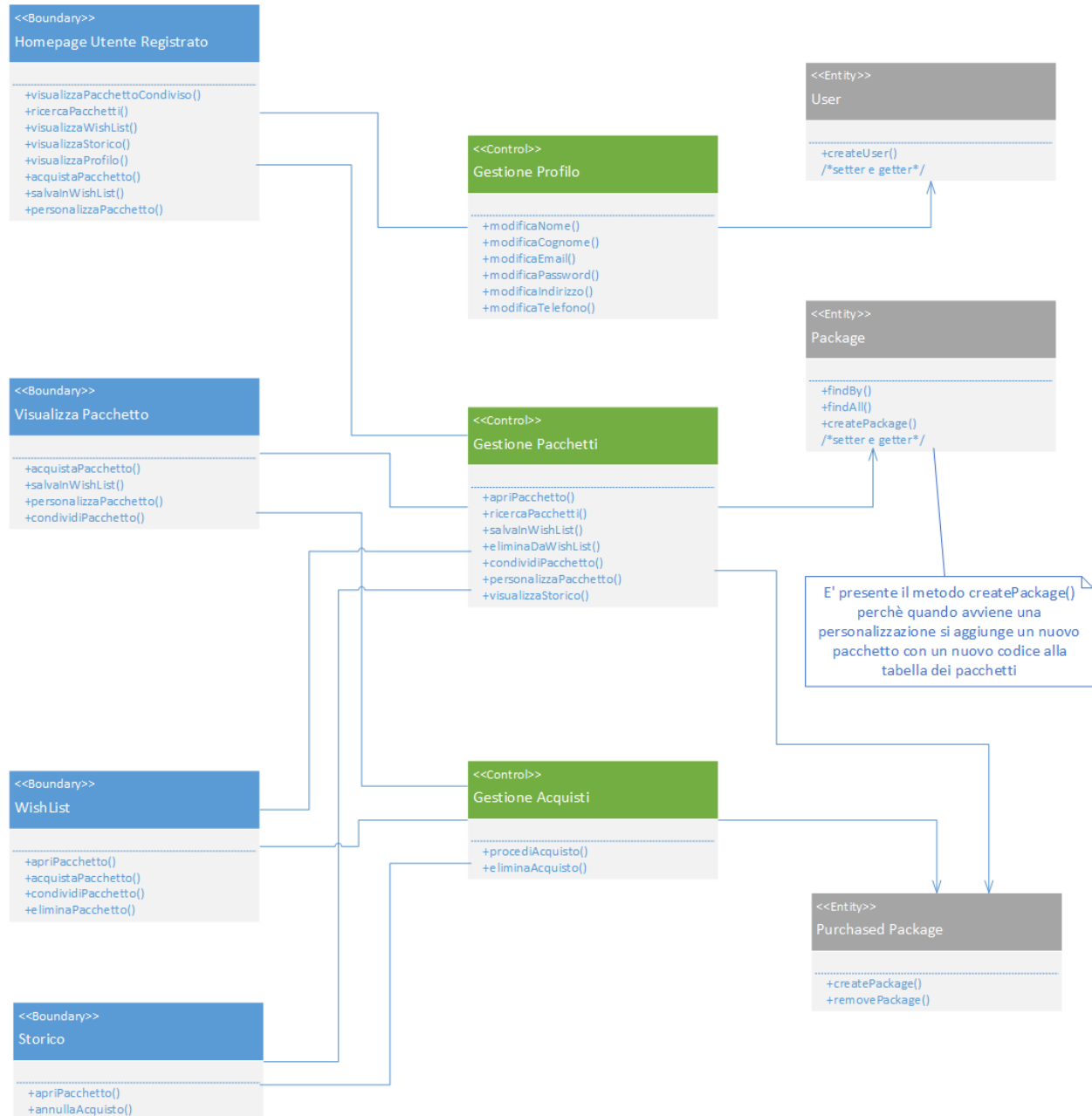


Figura 8: Schema BCE Utente Registrato

### 5.3 Impiegato

In questa sezione mostriamo l'organizzazione architeturale che gestisce le principali pagine (boundary) riservate agli impiegati.

Una volta eseguito il login l'impiegato può usufruire di tutte le funzioni comprese nella boundary *HomePagImpiegato* che sono gestite da il control *GestionePacchetti* il quale si servirà dei dati dell'entity *Package*. Lo stesso control si occupa anche di tutte le operazioni disponibili dai boundary *VisualizzaPacchetto* e *Crea-ModificaPacchetto*. L'altro control che gestisce tutte le funzioni relative alla gestione dei prodotti base è *GestioneProdottiBase* che si serve delle entity relatie ai prodotti base e serve la boundary *GestioneProdottiBase*.

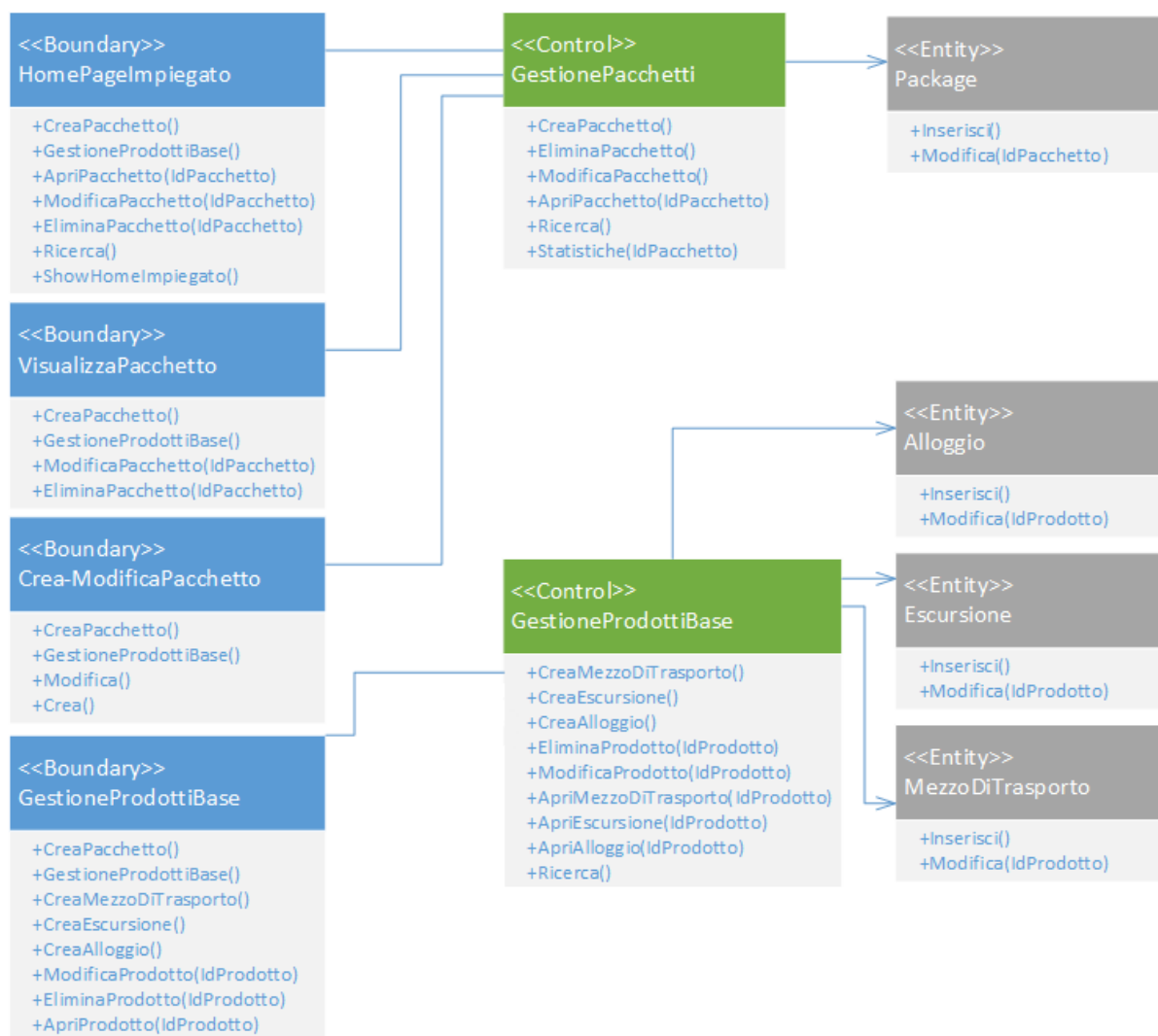


Figura 9: Schema BCE Impiegato

## 6 Sequence Diagrams

In questa sezione si riportano i sequence diagrams che descrivono esempi significativi di uso del sistema. L'analisi delle sequenze di eventi è stata divisa in Utente non Registrato, Utente Registrato, Impiegato.

### 6.1 Utente non Registrato

Il sequence diagram mostra la successione delle seguenti azioni:

- Collegamento al portale TravelDream;
- Visualizzazione di un pacchetto condiviso;
- Registrazione;
- Salvataggio in WishList.

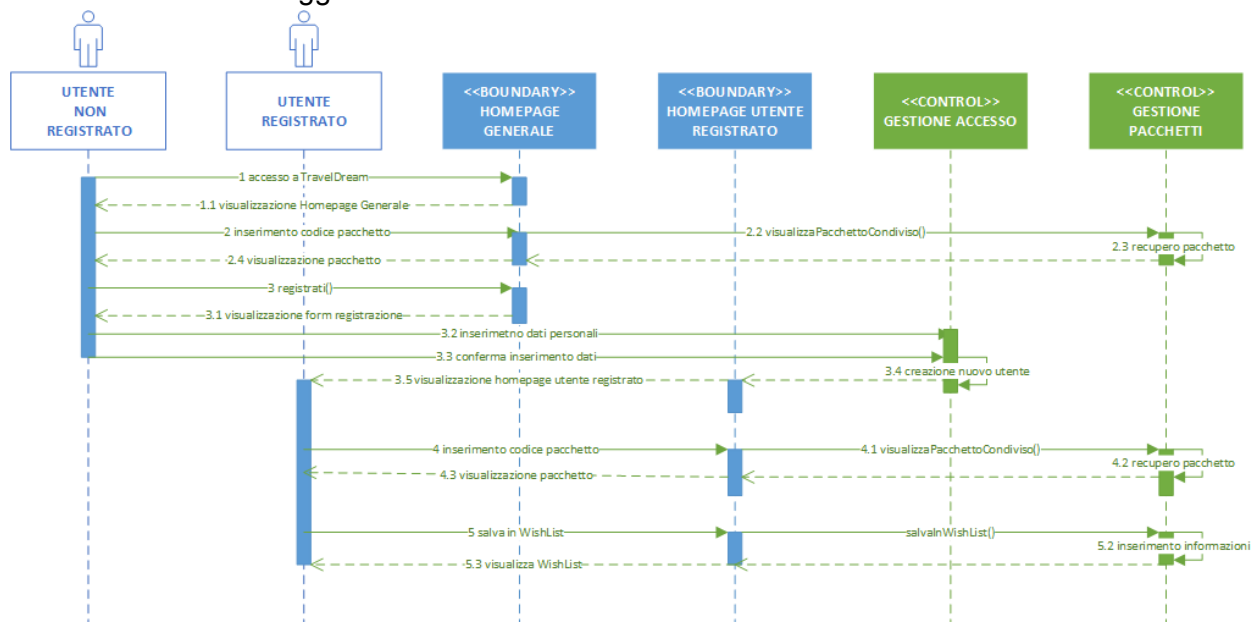


Figura 10: Sequence Diagram Utente non Registrato

### 6.2 Utente Registrato

Il sequence diagram mostra la successione delle seguenti azioni (l'utente ha già effettuato il login):

- Ricerca di pacchetti;
- Visualizzazione di un pacchetto;
- Acquisto del pacchetto.

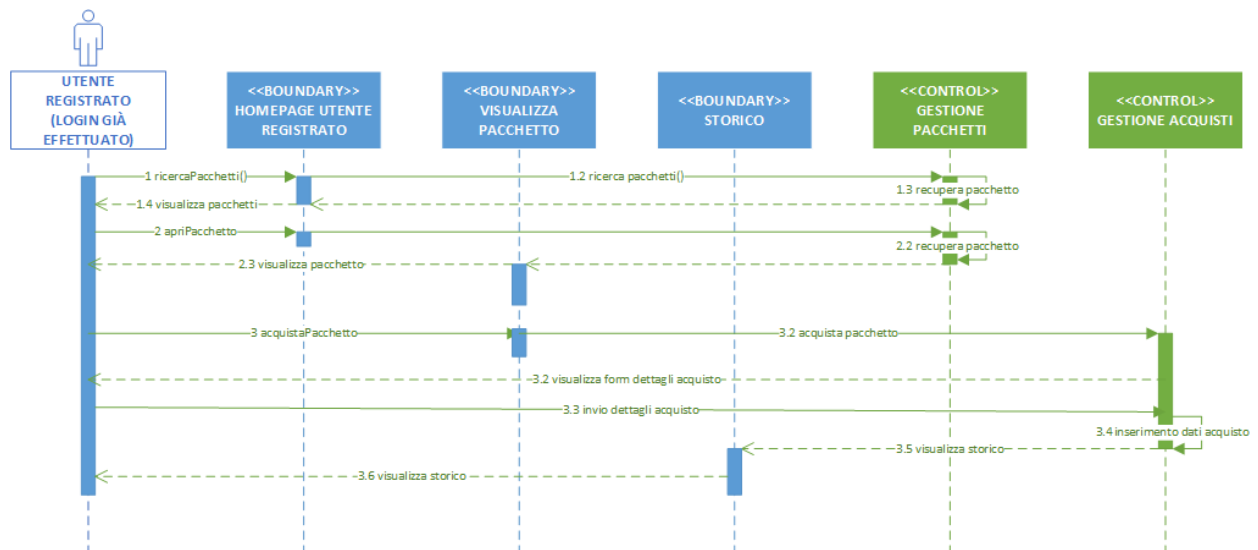


Figura 11: Sequence Diagram Utente Registrato

### 6.3 Impiegato

Il sequence diagram mostra la successione delle seguenti azioni (l'impiegato ha già effettuato il login):

- Crea un prodotto base (escursione);
- Crea un pacchetto.

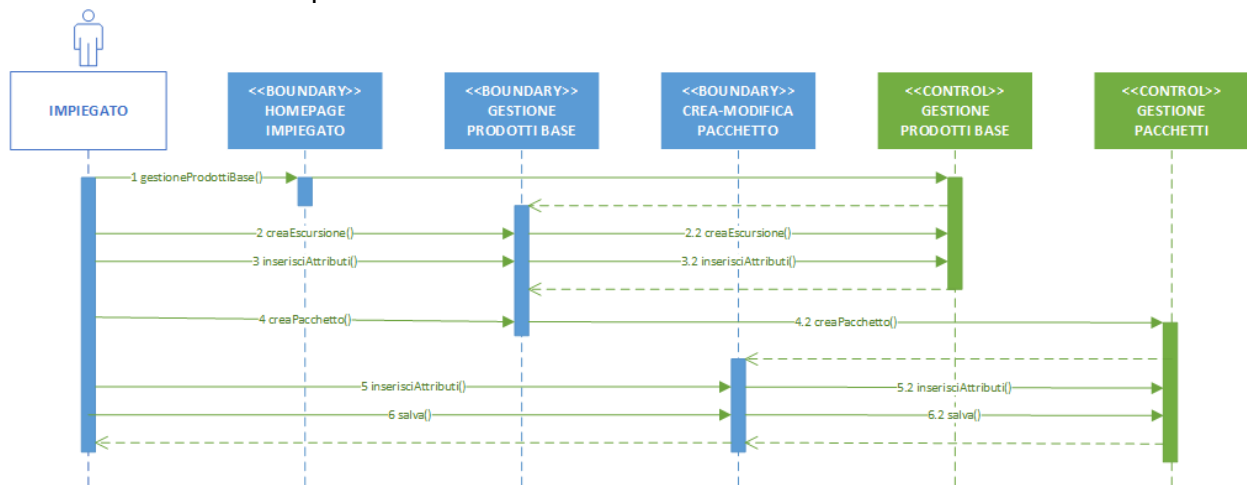


Figura 12: Sequence Diagram Impiegato

## Indice delle Figure

---

Figura 1: Livelli architetturali.....	4
Figura 2: Schema ER.....	6
Figura 3: Schema Logico del Database.....	8
Figura 4: Diagramma UX Homepage Generale e Utente non Registrato.....	10
Figura 5: Diagramma UX Utente Registrato.....	12
Figura 6: Diagramma UX Impiegato.....	15
Figura 7: Schema BCE Homepage Generale e Utente non Registrato.....	17
Figura 8: Schema BCE Utente Registrato.....	18
Figura 9: Schema BCE Impiegato.....	19
Figura 10: Sequence Diagram Utente non Registrato.....	20
Figura 11: Sequence Diagram Utente Registrato.....	21
Figura 12: Sequence Diagram Impiegato.....	21