



Laurea Triennale in Informatica - Università degli Studi di Salerno
Corso di *Ingegneria del Software* - Prof. C.Gravino



Object Design Document ReVampAscent

Riferimento	ODD_V1.0
Versione	1.0
Data	10/12/2025
Destinatario	C.Gravino



Laurea Triennale in Informatica - Università degli Studi di Salerno
Corso di *Ingegneria del Software* – Prof. C.Gravino

Presentato da	Antonio Aliberti, Raffaella Di Pasquale, Vincenzo Martucci
---------------	--

Team composition

Ruolo	Nome	Acronimo	Contatti
Team Member	Aliberti Antonio	AA	a.aliberti56@studenti.unisa.it
Team Member	Di Pasquale Raffaella	RDP	r.dipasquale5@studenti.unisa.it
Team Member	Martucci Vincenzo	VM	v.martucci5@studenti.unisa.it

Sommario

Team composition.....	2
Revision History.....	3
1. Introduzione.....	3



2. Elementi di Riuso.....	3
2.1 Design Pattern usati.....	3

Revision History

Data	Versione	Descrizione	Autori
14/12/2025	0.1	Prima stesura del documento	[TEAM MEMBER]
15/12/2025	0.6	Design Pattern	AA
16/12/2025	1.0	Revisione finale documento	[TEAM MEMBER]

1. Introduzione

Il sistema ReVamp Ascent è progettato per consentire la compravendita di beni d’arredamento e complementi per la casa attraverso una piattaforma e-commerce moderna, sicura e centralizzata.

Lo scopo principale è facilitare l’incontro tra domanda e offerta mediante un’infrastruttura digitale che automatizza i processi di ricerca, selezione, ordine, pagamento e gestione post-vendita, garantendo un’esperienza d’acquisto fluida e personalizzata per l’utente finale.

2. Elementi di Riuso

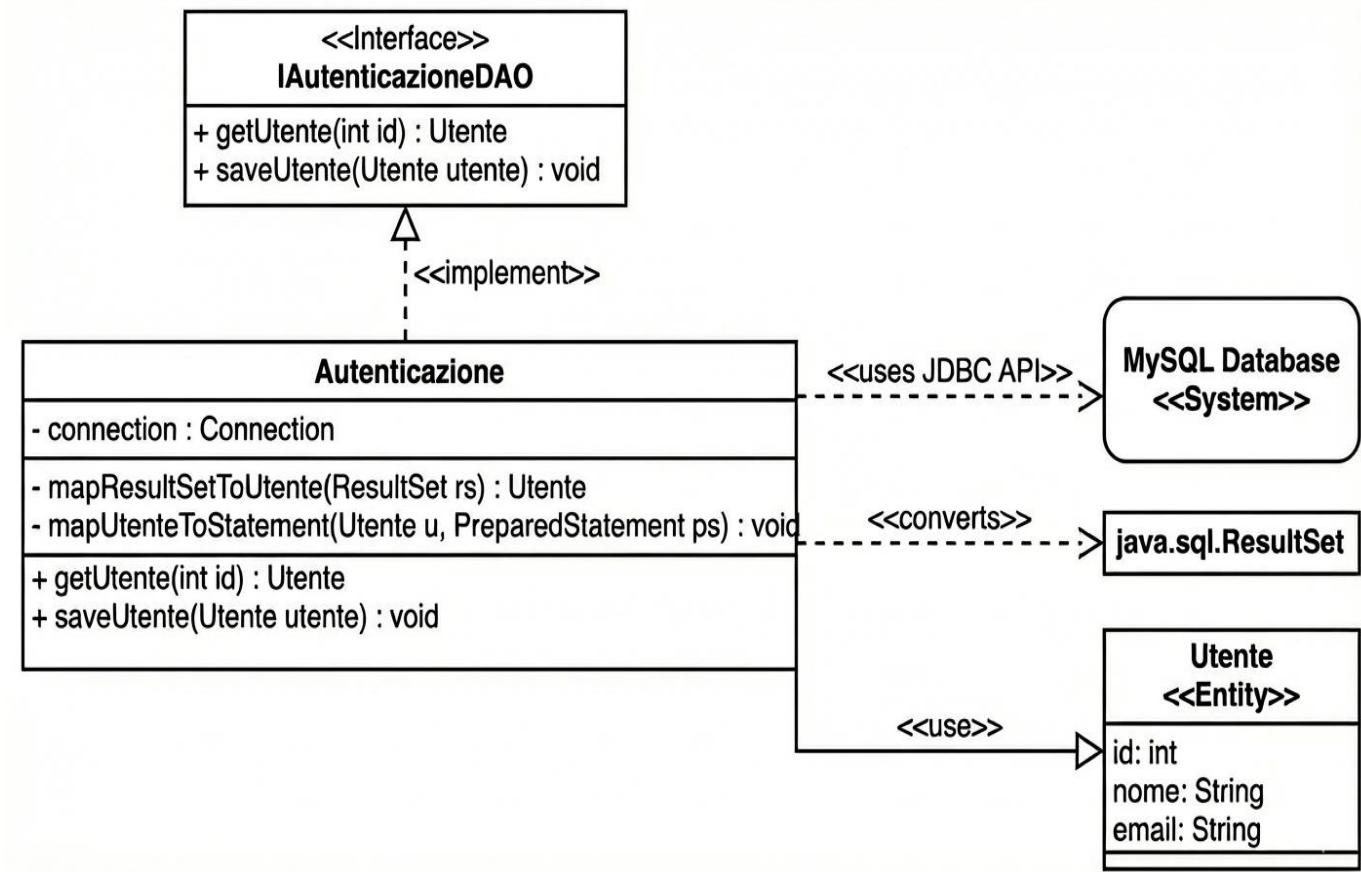
2.1 Design Pattern usati

All’interno del sistema sono stati usati dei pattern per risolvere alcuni problemi di programmazione e per rendere il codice più comprensibile.

Adapter Pattern (Gestione della Persistenza)



Il sistema gestisce la persistenza dei dati avvalendosi di MySQL, un database relazionale che organizza le informazioni in righe e colonne all'interno di tabelle, una struttura che differisce profondamente dal paradigma orientato agli oggetti utilizzato nello sviluppo dell'applicazione in Java. Per risolvere questo disallineamento strutturale, noto in ingegneria del software come Impedance Mismatch, e per soddisfare il requisito non funzionale riguardante la necessità di interfacciare il sistema con componenti esterni esistenti come il DBMS, è stato adottato il Design Pattern Adapter. L'implementazione pratica di tale pattern avviene tramite classi specifiche denominate Data Access Objects (DAO), le quali agiscono come veri e propri adattatori che si interpongono tra la logica di business e il database: il loro compito è incapsulare le API JDBC e convertire l'interfaccia dei dati grezzi provenienti dalle query SQL, sotto forma di ResultSet tabellari, nell'interfaccia a oggetti richiesta dal dominio applicativo, istanziando le relative classi Entity. In fase di salvataggio, il processo avviene inversamente mappando le proprietà degli oggetti Java in parametri per le istruzioni di manipolazione SQL, garantendo così che il core dell'applicazione rimanga completamente disaccoppiato dalla specifica implementazione della persistenza e rispettando i principi di Information Hiding e flessibilità architetturale.





Laurea Triennale in Informatica - Università degli Studi di Salerno
Corso di *Ingegneria del Software* – Prof. C.Gravino