



Università degli Studi di Salerno
Anno Accademico 2016/2017

Corso di Ingegneria del Software

Object Design Document
V 1.2

ITECH

Introduzione.....	4
<i>Scelte di design degli Oggetti.....</i>	<i>4</i>
Interfaccia vs. Usabilità.....	4
Tempo di rilascio vs Tolleranza ai fault.....	4
Sicurezza vs. Efficienza.....	4
Comprensibilità vs. Tempo	4
Costi vs. Mantenimento	4
Interfaccia vs. Tempo di risposta	4
Package	5
<i>Descrizione dei layer</i>	<i>5</i>
Interface layer	5
Application logic layer	5
Storage layer.....	6
<i>Comunicazione tra package.....</i>	<i>6</i>
Class interfaces	7
<i>Gestione utente.....</i>	<i>7</i>
Creazione account	7
Modifica account	7
Eliminazione account.....	7
<i>Gestione magazzino.....</i>	<i>7</i>
Creazione catalogo	7
Eliminazione catalogo.....	7
Controllo quantità	7
Ricerca categoria	8
<i>Gestione prodotti.....</i>	<i>8</i>
Creazione prodotto.....	8
Eliminazione prodotto	8
Carico prodotto	8
Ricerca prodotto	8
<i>Gestione servizi</i>	<i>8</i>
Creazione servizio	8
Eliminazione servizio	8
Ricerca servizio	9
<i>Gestione vendite</i>	<i>9</i>
Carrello	9
Storico vendite.....	9
<i>Gestione interfaccia</i>	<i>9</i>
Class diagram	9
Glossario	10

Top Manager:

Prof. De Lucia Andrea

Team di sviluppo:

Nome e Cognome	Matricola
<i>Stefano Foresta</i>	0512100504
<i>Gennaro Franzese</i>	0512100270

Revision History:

Autore	Data	Descrizione	Versione
<i>Stefano Foresta</i>	<i>10/12/16</i>	<i>Struttura documento</i>	<i>v 0.1</i>
<i>Gennaro Franzese</i>	<i>12/12/16</i>	<i>Stesura primo capitolo</i>	<i>v 0.2</i>
<i>Stefano Foresta</i>	<i>13/12/16</i>	<i>Stesura secondo capitolo</i>	<i>v 0.3</i>
<i>Stefano Foresta</i>	<i>13/12/16</i>	<i>Stesura terzo capitolo, correzioni capitoli precedenti</i>	<i>v 0.6</i>
<i>Gennaro Franzese</i>	<i>13/12/16</i>	<i>Stesura quarto capitolo, revisione documento</i>	<i>v 1.0</i>
<i>Stefano Foresta</i>	<i>13/12/16</i>	<i>Stesura glossario</i>	<i>v 1.0.1</i>
<i>Gennaro Franzese</i>	<i>14/12/16</i>	<i>Inserimento Class Diagram</i>	<i>v 1.2</i>

Introduzione

Lo scopo dell'Object Design Document è quello di specificare i servizi che ogni sottosistema, che è stato presentato nel System Design Document, offre in termini di classi includendo operazioni, tipi, argomenti e signature in modo da avere una specifica completa. Questo documento serve come base per l'implementazione del progetto software.

Scelte di design degli Oggetti

Interfaccia vs. Usabilità

L'interfaccia, grazie all'utilizzo di un'impostazione semplice e intuitiva, permette un uso facile (*Easy-Use*) della gestione del sistema e del relativo database di prodotti, da parte di chi amministra il negozio, e della navigazione all'interno del negozio per la ricerca e il relativo acquisto di prodotti, da parte dei clienti.

Tempo di rilascio vs Tolleranza ai fault

Una volta creata la struttura principale verrà dedicato tempo in più per la creazione di controllori in grado di gestire ogni eventuale errore della web application e verrà creato un modulo che gestirà i backup (giornalieri/settimanali/mensili) che permetteranno di avere un'istantanea del database che, in caso di guasti, sarà subito disponibile per la rimessa in funzione dell'intera applicazione.

Sicurezza vs. Efficienza

Verrà implementato un modulo dedicato interamente all'autenticazione degli utenti che permetterà di mantenere privati gli account dei clienti/amministratori e che permetterà, solo dopo effettuata l'autenticazione, l'acquisto di beni/servizi (da parte dei clienti) e la modifica del database (da parte degli amministratori).

Comprensibilità vs. Tempo

La stesura del codice sarà suddivisa in parti e commentata per rendere il più leggibile possibile lo stesso e permettere a terzi eventuali modifiche strutturali.

Costi vs. Mantenimento

Grazie a un uso di materiale open source e l'utilizzo di commenti, come descritto al punto precedente, il codice sarà facilmente modificabile (implementazione di nuove funzioni o correzioni di errori) con costi contenuti. Al contrario, l'utilizzo di un'architettura three-tier, avrà dei costi superiori alla media per via della presenza di più macchine su cui girerà la web application e il database utilizzato.

Interfaccia vs. Tempo di risposta

Il tempo di risposta tra server e interfaccia sono più che sufficienti a soddisfare le esigenze dei vari clienti che si collegheranno al Sistema per ricercare e/o acquistare beni/servizi. Ovviamente maggiore sarà la grandezza del database e maggiore sarà il tempo di risposta e ricerca nel database che ovviamente sarà indicizzato per abbassare ulteriormente questi tempi.

Package

Il software ha un'architettura three-tier e i livelli che si distinguono sono i seguenti:

1. **Interface Layer:** oggetti con cui l'utente interagisce (form, button, textbox, ecc.).
2. **Application Logic Layer:** livello che si occupa di gestire moduli che girano su un application server che genera i moduli per i contenuti dinamici e che riceve le richieste e le elabora.
3. **Storage Layer:** questo livello gestisce l'archiviazione persistente dei dati.

I layer sono suddivisi in package ed ognuno di questi contiene degli oggetti che andranno ad implementare le varie funzionalità del sistema che andiamo a sviluppare. I package vengono utilizzati per riunire le classi, logicamente correlate, che forniscono dei servizi simili.

Descrizione dei layer

Interface layer

Nome	<i>Interface layer</i>
Descrizione	Offre all'utente servizi di interazione uomo-macchina, ovvero click, submit etc.
Moduli presenti	
Modulo	Descrizione
<i>Interfaccia web</i>	Questo modulo descrive l'interfaccia grafica con cui l'utente interagisce e che invia al sistema le varie richieste.

Application logic layer

Nome	<i>Application logic layer</i>
Descrizione	livello che si occupa di gestire moduli che girano su un application server che genera i moduli per i contenuti dinamici e che riceve le richieste e le elabora.
Moduli presenti	
Modulo	Descrizione
<i>Gestione utente</i>	Il modulo si occupa della gestione generale dell'utente. Creazione, modifica, gestione, cancellazione.
<i>Gestione magazzino</i>	Il modulo si occupa della gestione generale del magazzino.
<i>Gestione prodotti</i>	Il modulo gestisce i prodotti.
<i>Gestione servizi</i>	Il modulo gestisce i servizi.
<i>Gestione vendite</i>	Il modulo gestisce le vendite.
<i>Gestione interfaccia</i>	Il modulo si occupa di ricevere e interpretare gli input.
<i>Gestione database</i>	Il modulo si occupa della comunicazione con il database.

Storage layer

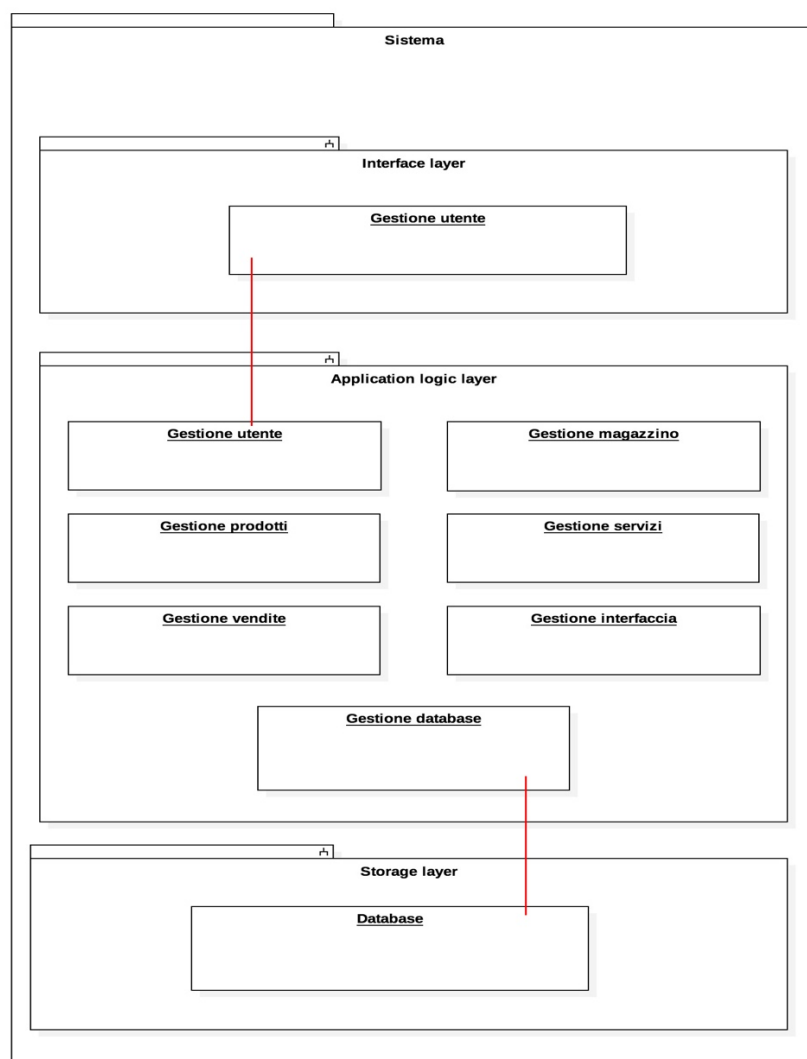
Nome	<i>Storage layer</i>
Descrizione	Questo livello gestisce l'archiviazione persistente dei dati.
Moduli presenti	
Modulo	Descrizione
<i>Database</i>	Gestisce le richieste di dati in entrata e in uscita. Si occupa del database e della sua gestione.

Comunicazione tra package

Il seguente diagramma mostra la comunicazione tra i vari package che compongono i layer del sistema che sono stati appena descritti. La scelta progettuale è quella di far passare ogni richiesta fatta dalle classi di interfaccia al Logic Application Layer.

All'interno del Logic Application Layer sono presenti i pacchetti che si occupano della logica applicativa del sistema e della sua gestione.

Infine, attraverso l'interfaccia Storage, il Logic Application Layer accederà allo Storage Layer, per recuperare, aggiornare o eliminare dati persistenti dalla base dati.



Class interfaces

Ad ogni classe viene assegnato un nome, significativo per il suo scopo e diverso da tutte le altre classi.

Gestione utente

Questo modulo si occupa della gestione generale dell'utente dal momento della sua creazione passando per le relative modifiche che possono essere apportate e finendo ad un'eventuale eliminazione dell'account.

Creazione account

Dipendenze	Nessuna
Eccezioni	<ul style="list-style-type: none">• Utente già presente nel database;• Dati inseriti non validi.

Modifica account

Dipendenze	Ricerca
Eccezioni	<ul style="list-style-type: none">• Dati già presenti;• Dati inseriti non validi.

Eliminazione account

Dipendenze	Ricerca
Eccezioni	<ul style="list-style-type: none">• Pagamento in sospeso;• Ordine in sospeso;

Gestione magazzino

A questo modulo viene affidata la gestione del magazzino.

Creazione catalogo

Dipendenze	Nessuna
Eccezioni	<ul style="list-style-type: none">• Catalogo già presente;• Dati inseriti non validi.

Eliminazione catalogo

Dipendenze	Ricerca
Eccezioni	<ul style="list-style-type: none">• Presenza prodotti nel catalogo.

Controllo quantità

Dipendenze	Prodotti
Eccezioni	Nessuna

Ricerca categoria

Dipendenze	Storage
Eccezioni	<ul style="list-style-type: none">• Categoria non esistente.

Gestione prodotti

Il modulo gestisce i singoli prodotti che poi sono organizzati nel catalogo

Creazione prodotto

Dipendenze	Categoria
Eccezioni	<ul style="list-style-type: none">• Prodotto già esistente;• Dati inseriti non validi;• Categoria non selezionata.

Eliminazione prodotto

Dipendenze	Ricerca, Vendite
Eccezioni	<ul style="list-style-type: none">• Ordine in sospeso.

Carico prodotto

Dipendenze	Catalogo, Storage
Eccezioni	Nessuna.

Ricerca prodotto

Dipendenze	Storage, categoria.
Eccezioni	<ul style="list-style-type: none">• Prodotto non in magazzino;• Prodotto non esistente.

Gestione servizi

Il modulo gestisce i singoli servizi offerti che poi sono organizzati nel catalogo

Creazione servizio

Dipendenze	Categoria
Eccezioni	<ul style="list-style-type: none">• Servizio già esistente;• Dati inseriti non validi;• Categoria non selezionata.

Eliminazione servizio

Dipendenze	Ricerca, Vendite
Eccezioni	<ul style="list-style-type: none">• Ordine in sospeso.

Ricerca servizio

Dipendenze	Storage, categoria.
Eccezioni	<ul style="list-style-type: none">• Servizio non in magazzino;• Servizio non esistente.

Gestione vendite

Il modulo gestisce le vendite di beni/servizi e si interfaccia direttamente con gli utenti.

Carrello

Dipendenze	Storage, categoria, prodotti, servizi.
Eccezioni	Nessuna.

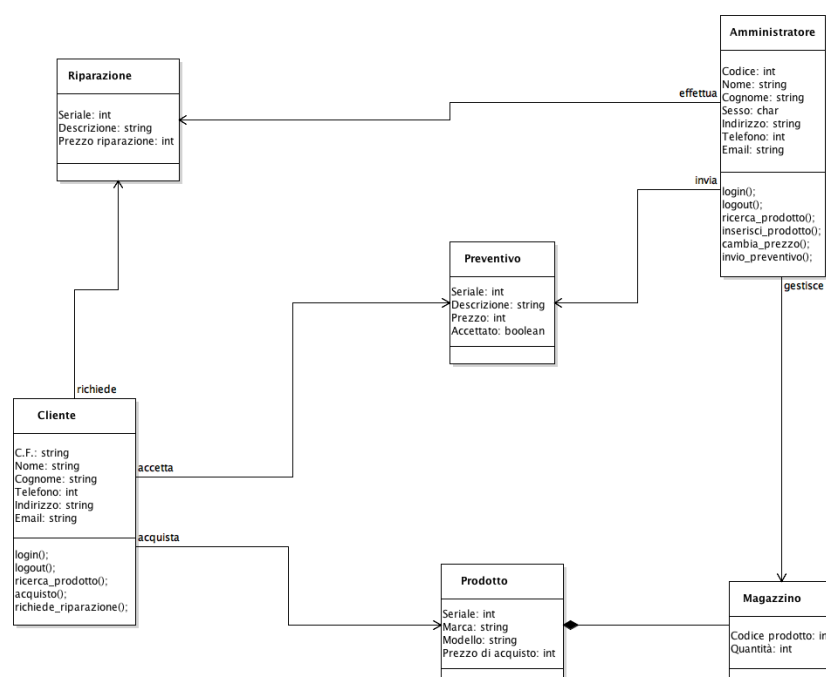
Storico vendite

Dipendenze	Storage, clienti.
Eccezioni	Nessuna.

Gestione interfaccia

Il modulo si occupa di ricevere e interpretare gli input che vengono dal layer interfaccia. Comprendono quindi tutte le varie componenti grafiche che servono per la gestione delle operazioni sulla web application.

Class diagram



Glossario

Termine	Descrizione
<i>Easy-Use</i>	Di facile utilizzo.
<i>Package</i>	Collezione di classi, interfacce.