

Progetto Ingegneria del Software 2008-09

“New-Totosì”

Documento di System Design

Team Leader:
Francesco Lauria 26334
Team:
Donato Sabia 17104
Antonio Genovese 21368
Savino Sasso 21731

Coordinatori del progetto:

Nome	Matricola
Francesco Lauria	26334
Savino Sasso	21731

Partecipanti al progetto:

Nome	Matricola
Antonio Genovese	21368
Donato Sabia	17104

Revision History

Data	Versione	Descrizione	Autore
20-01-2009	1.0	Elaborazione del documento	Team
23-01-2009	1.1	Modifica Diagrammi Sottosistemi	Team
26-01-2009	2.0	Modifica nel modello dei Dati Persistenza	Team
27-01-2009	2.1	Correzione modello Scomposizione dei Sottosistemi	Team

Documento di SYSTEM DESIGN
Progetto di un portale per una società
di servizi dedicati alle scommesse sportive on-line

1.	Introduzione	2
1.1	Scopo del sistema	2
1.2	Obiettivi di Design	2
1.2.1	Criteri di Performance	2
1.2.2	Criteri di Design	3
1.2.3	Criteri di Manutenzione	4
1.3	Definizioni, acronimi ed abbreviazioni	4
1.4	Riferimenti	4
2.	Architettura Software corrente	5
3.	Architettura Software proposta	6
3.1	Overview	6
3.2	Scomposizione dei Sottosistemi	6
3.3	Mappatura Hardware / Software	8
3.4	Gestione dei dati persistenti	16
3.5	Controllo degli accessi e Sicurezza	18
3.5.1	Controllo degli accessi	18
3.5.2	Sicurezza	19
3.6	Global Software Control	19
3.7	Condizioni dei boundary	20
3.7.1	Configurazione	20
3.7.2	Casi d'uso Accesso / Uscita	21
3.7.3	Condizioni eccezionali	22
4.	Servizi del sottosistema	23

1. Introduzione

1.1 Scopo del Sistema

L'obiettivo del progetto è quello di re-ingegnerizzare il portale di scommesse "Totosì" (per maggiori dettagli vedi RAD). Il sistema software presenta dei vincoli importanti che devono essere rispettati nella fase di design del sistema, fra questi troviamo vincoli sulle prestazioni del sistema, vincoli di architettura, vincoli di portabilità ed altri che sono specificati nel seguito del documento. Nel documento di SDD ci si ripropone di fronteggiare i vari problemi e vincoli del sistema riscontrati fornendo un'architettura competitiva e vantaggiosa.

1.2 Obiettivi di Design

Il sistema proposto cerca di rispettare la maggior parte degli obiettivi di design proponendo dei buoni compromessi (trade – off) fra quelli che sono in contrasto fra loro.

In particolare abbiamo:

1.2.1 Criteri di Performance

Criterio di Design	Definizione
Tempo di risposta	Mediamente una richiesta da parte del client è soddisfatta in un tempo di risposta inferiore ai 5 secondi. Queste prestazioni sono garantite dagli algoritmi di programmazione non molto complessi e da un ottimo sistema hardware che permette di eseguire il sistema in modo fluido.
Throughput	Il sistema riesce a soddisfare almeno 12 task in un'unità di tempo fissata di 1 minuto.
Memoria	La memoria di massa richiesta per il corretto funzionamento del sistema è dettata, oltre che dalle dimensioni dei moduli dell'applicazione, anche dalle dimensioni dello spazio necessario alla memorizzazione e all'archiviazione dei dati nei vari database presenti. Prevedendo un immagazzinamento dei dati quotidiano si stima, quindi, siano necessari almeno 100 Gigabyte di memoria di massa. Per un'esecuzione ottimale del sistema, sono richiesti, inoltre, almeno 2 Gigabyte di memoria RAM.

1.2.2 Criteri di Design

Criterio di design	Definizione
Robustezza	<p>Garantire la robustezza significa verificare la correttezza dei dati mediante routine, che impediscono di inviare input sbagliati ai vari moduli dell'applicazione.</p> <p>Inoltre all'interno dei moduli verrà verificata la correttezza dei dati di maggiore rilevanza, assicurando così un livello di robustezza abbastanza elevato.</p>
Affidabilità	<p>In caso di crash, il sistema deve essere in grado di lasciare i valori dei dati o nella situazione precedente al crash, o comunque in uno stato consistente.</p> <p>Invece, nel caso i cui si verifichino temporanei problemi di connessione alla rete, i client, che non hanno possibilità di lavorare in locale, devono attendere il ripristino della rete per inoltrare le richieste al server.</p>
Tolleranza ai crash	<p>Al verificarsi di un crash il lavoro in corso verrà salvato ed il sistema software terminerà la sua esecuzione lasciando la possibilità all'utente di decidere in seguito se terminare il lavoro o annullarlo.</p>
Sicurezza	<p>Essendo un applicazione web-based è necessario garantire agli utenti ed al sistema software un alto livello di protezione e sicurezza.</p> <p>Verranno utilizzati gli algoritmi di cifratura più complessi per l'invio e la ricezione dei dati.</p>

1.2.3 Criteri di Manutenzione

Criterio di Design	Definizione
Estendibilità	Il sistema software è stato diviso in moduli semi-indipendenti che lo rendono facilmente comprensibile ed estendibile per l'aggiunta di nuove funzionalità.
Adattabilità	Il sistema può essere facilmente riadattato ad un nuovo dominio applicativo che richieda la gestione e la memorizzazione di dati.
Portabilità	Il prodotto software è composto da moduli scritti utilizzando il linguaggio JSP, pertanto l'adattamento su altre piattaforme è davvero semplice.
Leggibilità	Dalla lettura del codice, la comprensione del sistema risulta facile grazie alla progettazione modulare del software e all'aggiunta di linee di commento a supporto del codice stesso.
Tracciabilità dei requisiti	La tracciabilità dei requisiti a partire dal codice è relativamente semplice per via della modularità delle componenti software.

1.3 Definizioni, acronimi ed abbreviazioni

RAD : Requirements Analysis Document

SDD : System Design Document

DBMS : Database Management System

JDBC : Java DataBase Connectivity

JSP : Java Server Page

SQL : Structured Query Language

1.4 Riferimenti

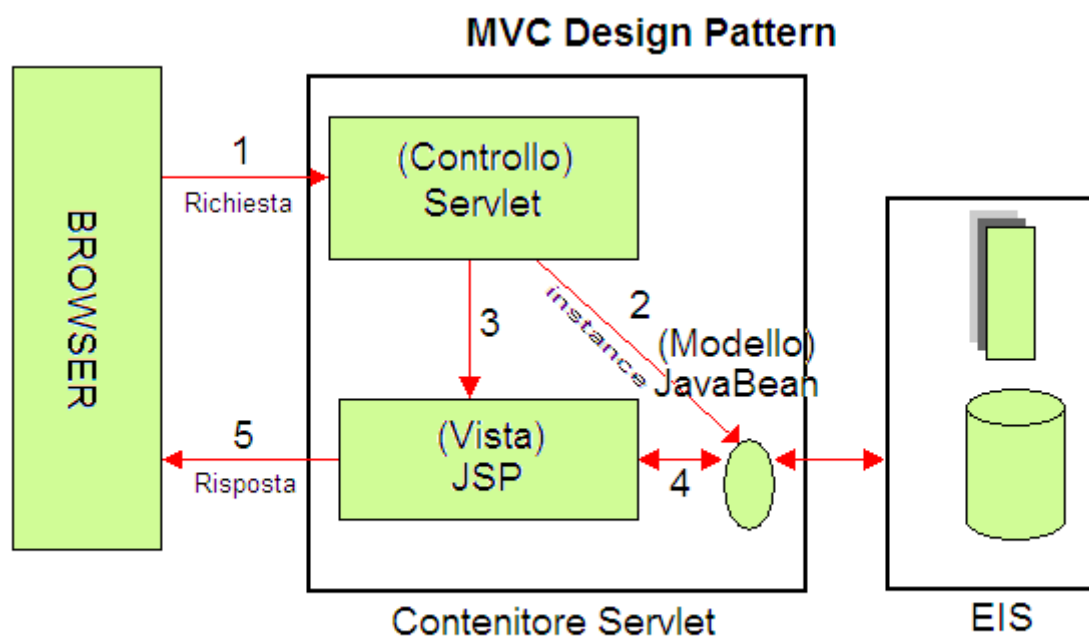
Per maggiori informazioni, consultare il RAD del progetto, contenente i dettagli aggiuntivi che per motivi di vario genere sono stati omessi in questo documento, in quanto rappresentavano dati non rilevanti per l'SDD.

2. Architettura Software corrente

L'architettura proposta nel seguito rimpiazza l'architettura preesistente .

Presenterà uno stile di decomposizione ben noto strutturato su tre livelli logici principali denominato anche *MVC (Modello – Vista – Controllo)*, modello di architettura ampiamente utilizzato per applicazioni di tipo web e particolarmente adatto per l'utilizzo di tecnologia JSP e SERVLET. I tre livelli logici utilizzati nello specifico sono:

1. **Livello Vista:** include tutti gli oggetti boundary, finestre, form, pagine web e così via.
2. **Livello Controllo:** include tutti gli oggetti control ed entity, realizzando così tutti i processi logici e di controllo forniti dal sistema.
3. **Livello Modello:** responsabile della gestione e della memorizzazione degli oggetti permanenti del sistema.



3. Architettura Software proposta

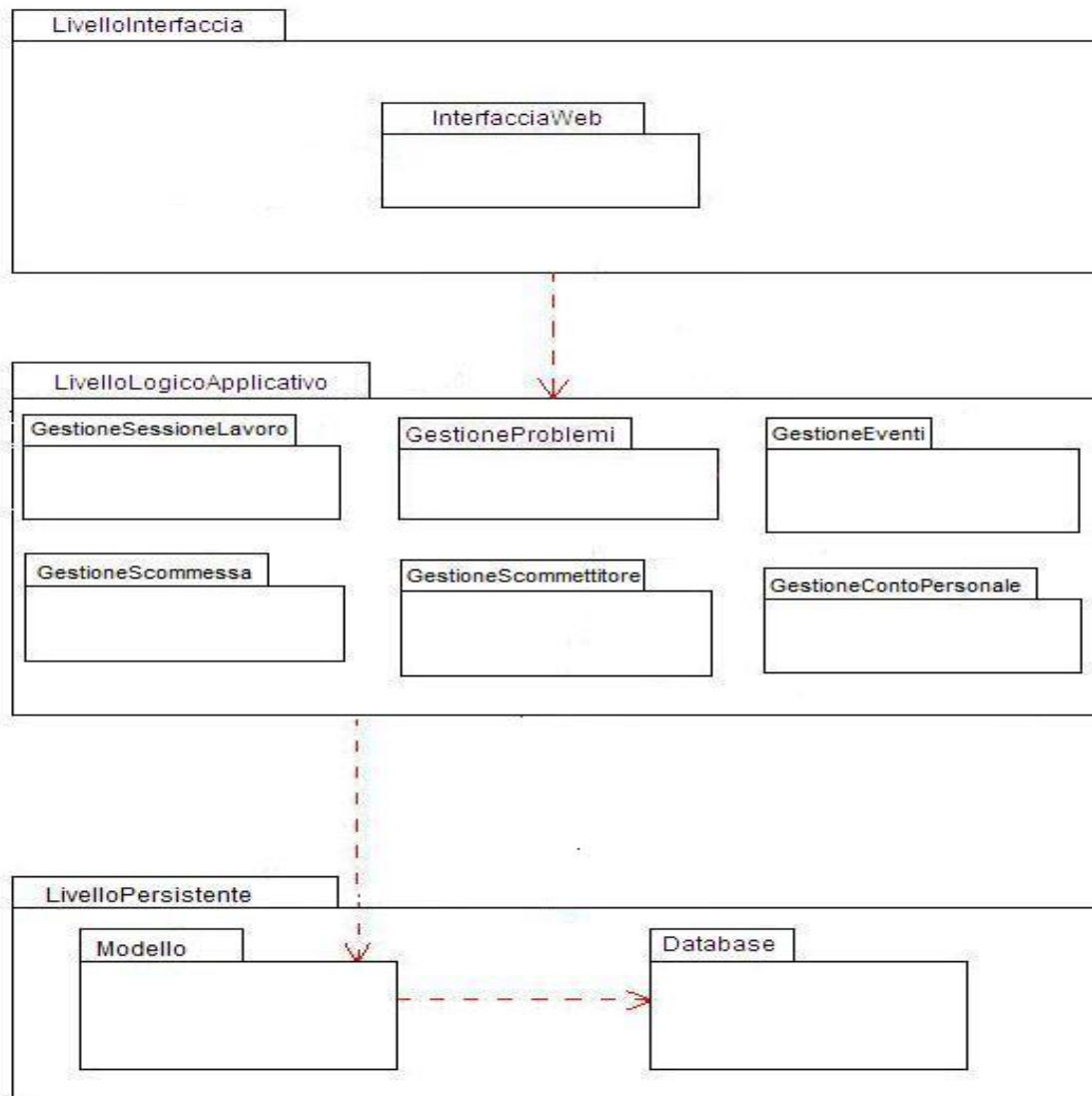
3.1 Overwiev

Come anticipato in precedenza, l'architettura del nostro sistema si presenterà decomposta in tre livelli logici principali. Tale architettura, denominata anche *MVC*, è ampiamente utilizzata per applicazioni di tipo web. Il sistema software è sarà decomposto in diversi sottosistemi per facilitarne l'implementazione e per ridurre la complessità generale. La decomposizione si propone inoltre di minimizzare le dipendenze (coupling) fra i sottosistemi identificati e di massimizzare la coesione all'interno degli stessi.

L'unico sottosistema fondamentale individuato è "Amministrazione".

3.2 Scomposizione dei Sottosistemi

Sottosistema "**Amministrazione**":



Il sottosistema sarà suddiviso in tre livelli:

- “*Livello Interfaccia*”
- “*Livello Logico Applicativo*”
- “*Livello Persistente*”

L'utente interagirà, mediante la componente WebServer, con il “*Livello Interfaccia*”, il quale rappresenta l'interfaccia utente del sottosistema. Tale modulo interagirà a sua volta con il “*Livello Logico Applicativo*” che espletterà le funzionalità richieste.

Il “*Livello Persistente*” potrà, quindi, utilizzare le componenti JDBC™ che permettono l'interfacciamento con i servizi sottostanti di DBMS.

Per quanto descritto sopra, tutto il flusso di informazioni avviene utilizzando un'architettura Client-Server.

Ogni livello conterrà al suo interno diversi moduli, nel seguito si presenta una loro breve descrizione.

“Livello Interfaccia”:

InterfacciaWeb	Modulo che rappresenta l'interfaccia grafica con cui l'utente si ritrova ad interagire inoltrando tramite esso le richieste al sistema e visualizzandone i risultati.
-----------------------	---

“Livello Logico Applicativo”:

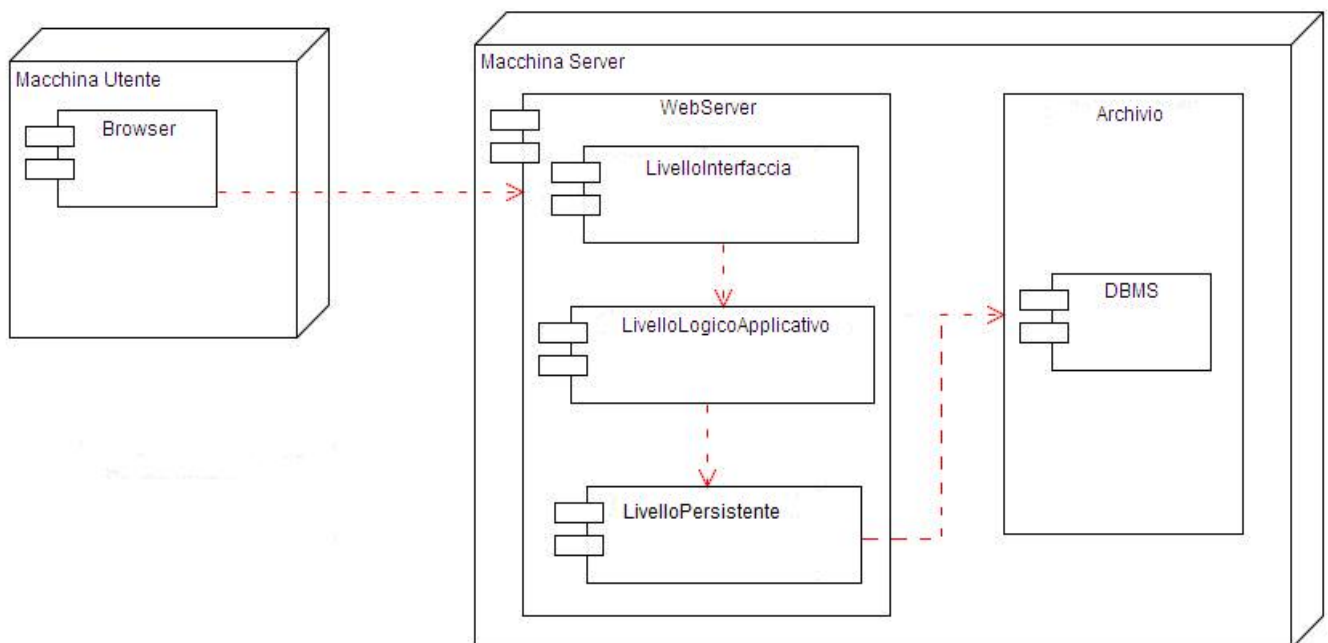
GestioneSessioneLavoro	Modulo che regola gli accessi al sistema con le operazioni di login e logout.
GestioneScommettitore	Modulo che si occupa di gestire il profilo dello scommettitore cioè le operazioni di registrazione, cancellazione, modifica e ricerca.
GestioneEventi	Modulo che si occupa della gestione degli eventi cioè le operazioni di ricerca, registrazione, cancellazione e inserimento di un risultato.
GestioneContoPersonale	Modulo che si occupa di gestire le operazioni sul conto personale cioè prelievo, versamento e riepilogo del conto.
GestioneScommessa	Modulo che si occupa di gestire la scommessa cioè le operazioni registrazione, ricerca e di calcolo vincite.
GestioneProblemi	Modulo che si occupa di fornire l'help on-line.

“Livello Persistente”:

Modello	Modulo che permette di minimizzare il coupling fra il modulo del database e quello del LivelloLogicoApplicativo, è composto da un'interfaccia che consente di chiamare le operazioni sul database.
Database	Modulo che inoltra le richieste di recupero dati al database fisico sul disco e ne recupera i dati per restituirli al livello superiore.

3.3 Mappatura Hardware / Software

Per quanto riguarda il flusso d'informazioni, l'architettura sarà scomposta in due livelli, un livello CLIENT ed un livello SERVER. Rappresentiamo nel seguito la distribuzione delle componenti Hardware e Software sui due nodi.

**Descrizione della componente WebServer**

Il WebServer che sarà utilizzato è Apache Tomcat 6.0.18 che sarà residente su di una macchina con sistema operativo Windows NT/XP/VISTA oppure piattaforme UNIX.

Descrizione della componente “Livello Interfaccia”

Il “Livello Interfaccia”, direttamente dipendente dal WebServer, sarà posizionato lato server ed i suoi contenuti, ossia le JSP, sono visualizzati tramite la componente Browser dal client.

Descrizione della componente “Livello Logico Applicativo”

Il “Livello Logico Applicativo” sarà interamente dedicato alla logica di controllo del sistema ed inoltre funge da mediatore tra l’interfaccia utente, fornita da “Livello Interfaccia”, e il livello sottostante ossia “Livello Persistente”.

Descrizione della componente “Livello Persistente”

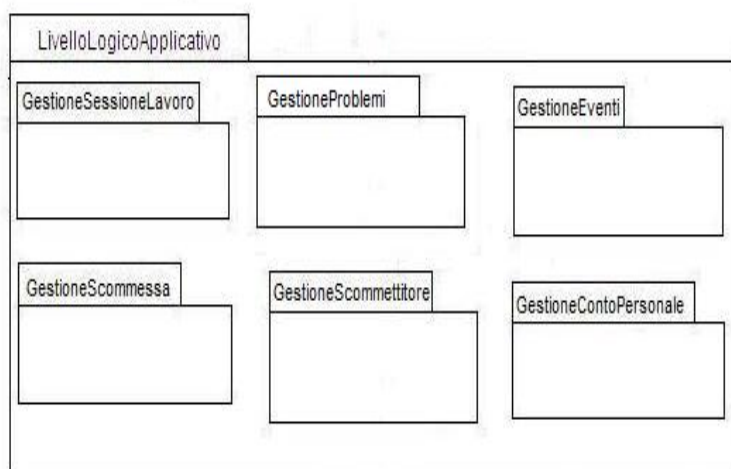
“Livello Persistente” è un’insieme di classi e interfacce scritte in Java che forniscono agli sviluppatori la possibilità di scrivere applicazioni robuste per interfacciarsi con i database usando esclusivamente tecnologia Java. All’interno del sottosistema verrà utilizzata la tecnologia JDBC. Per usare le API di JDBC con un particolare DBMS occorre un driver basato su tecnologia JDBC che ha il compito di mediare tra JDBC e il database.

Descrizione della componente DBMS

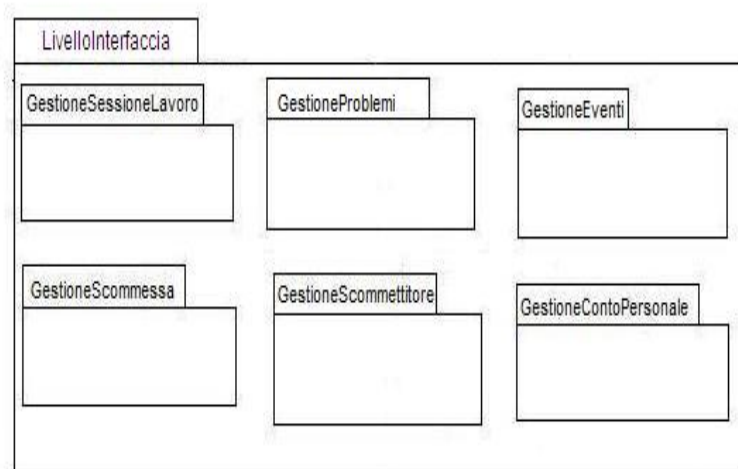
Vista la problematica degli accessi multipli è stato deciso di affidare la gestione dei dati persistenti ad un database relazionale ed in particolare al DBMS PostgreSQL. Nel nostro sistema verrà usato il framework JDBC di Java e il driver postgresqlDriver.jar per l’interfacciamento con il DBMS.

Di seguito riportiamo la suddivisione in moduli delle componenti “**Livello Interfaccia**” e “**Livello Logico Applicativo**”.

“Livello Logico Applicativo”

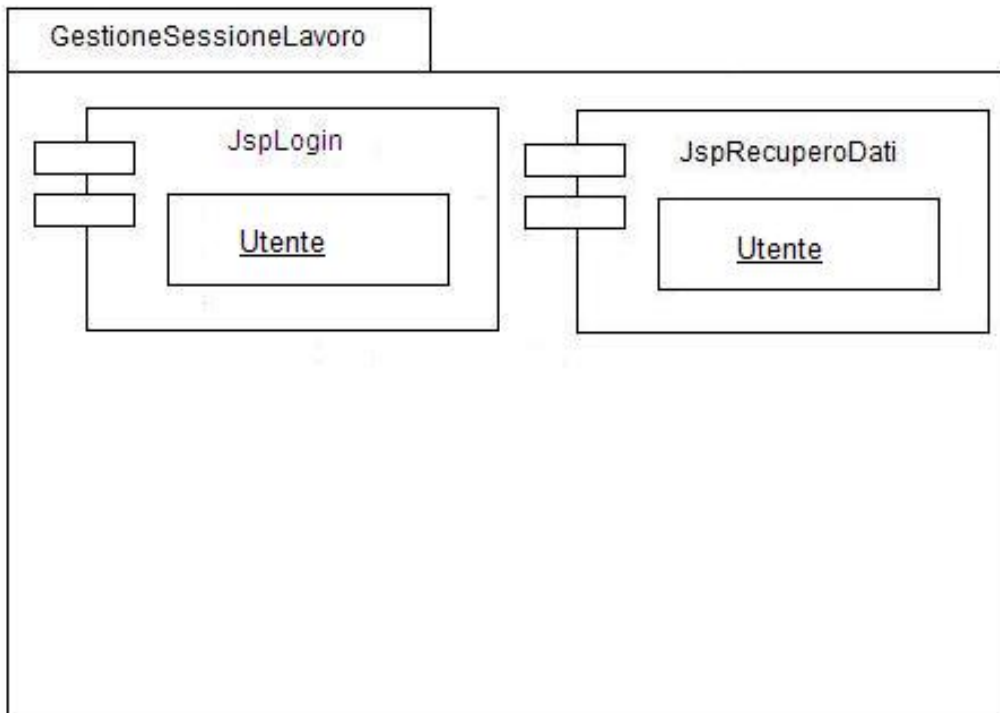


“Livello Interfaccia”

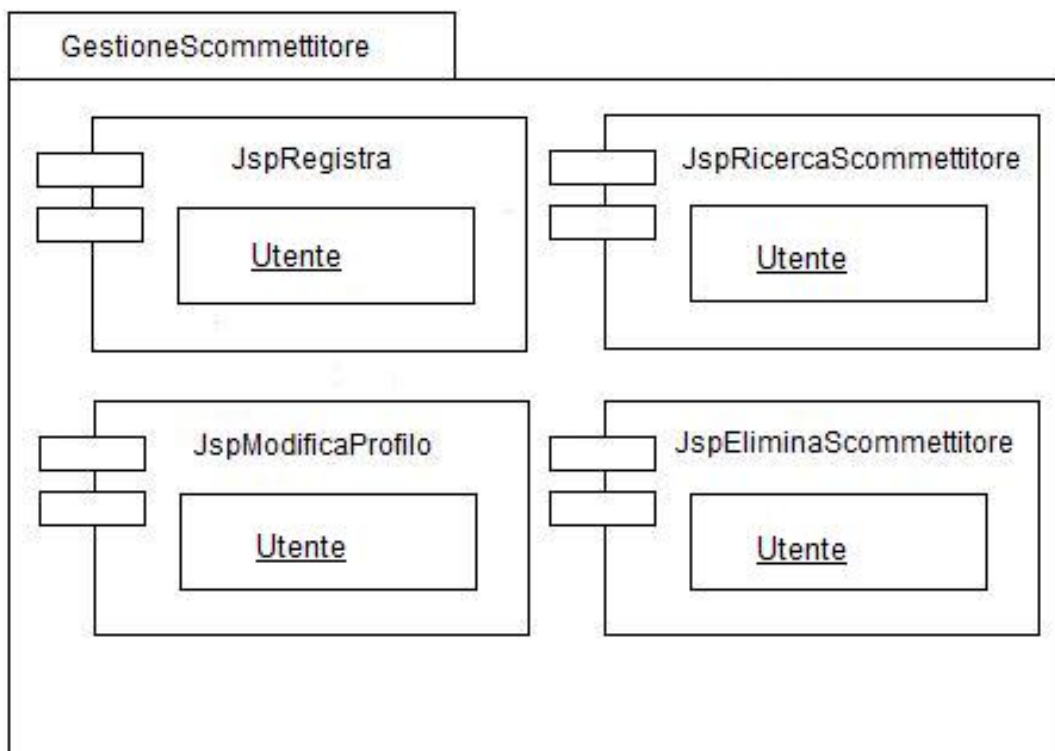


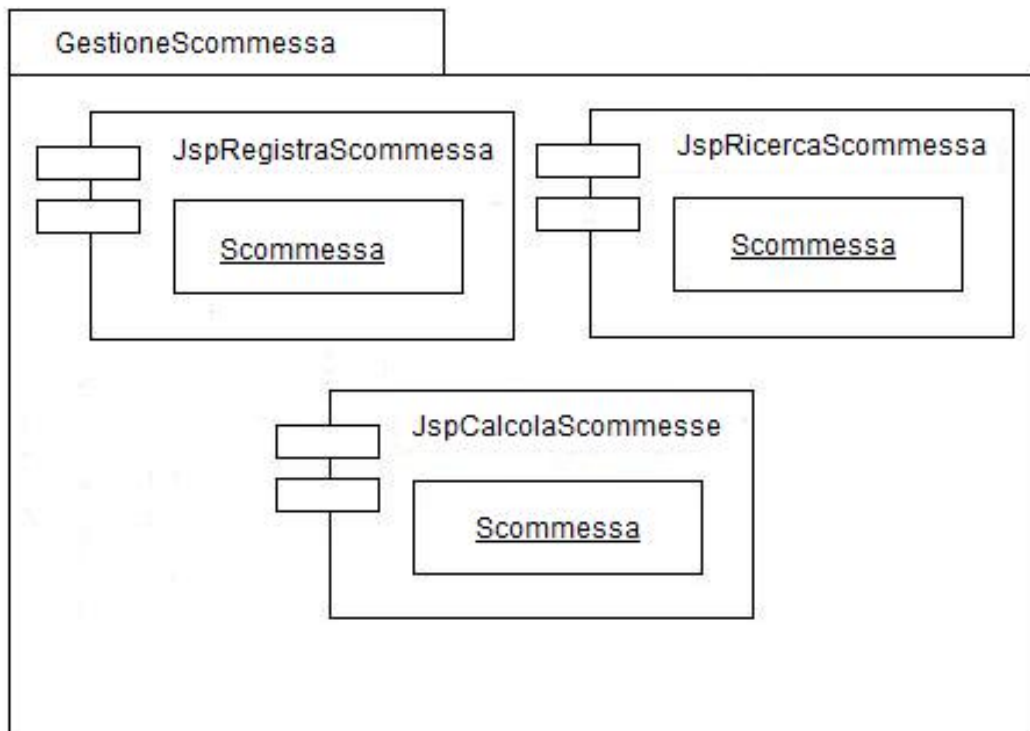
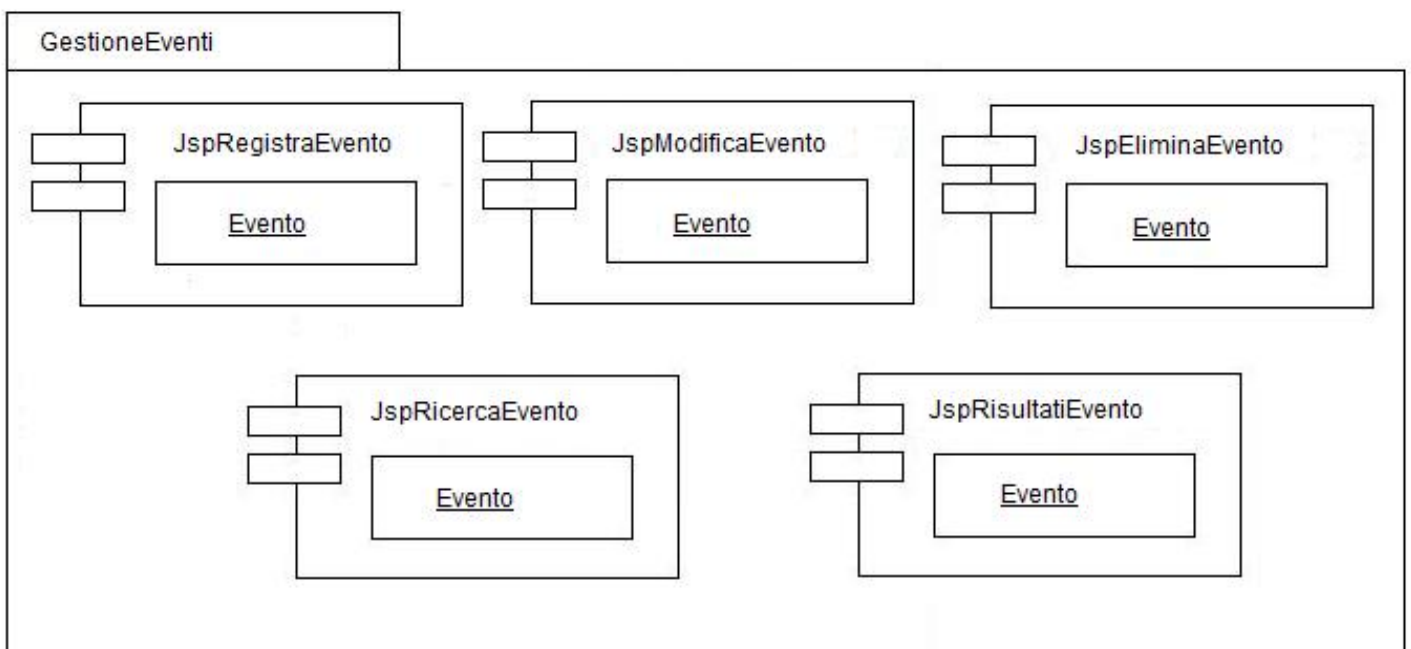
Per ogni modulo indicato riportiamo le JSP e le Servlet contenute all'interno:

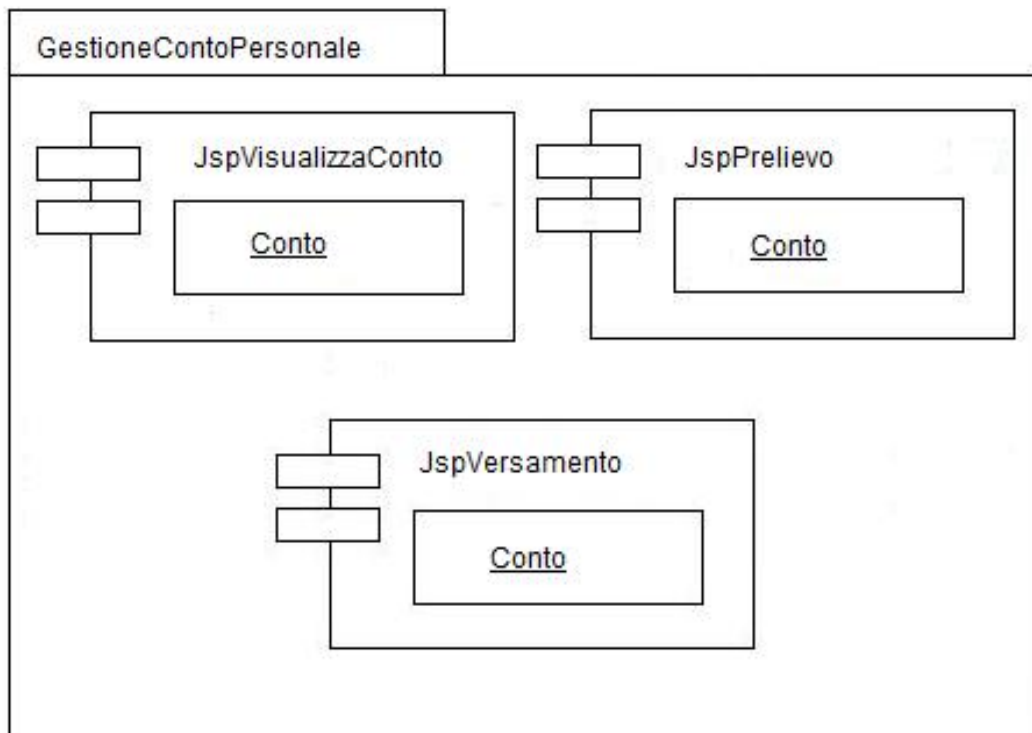
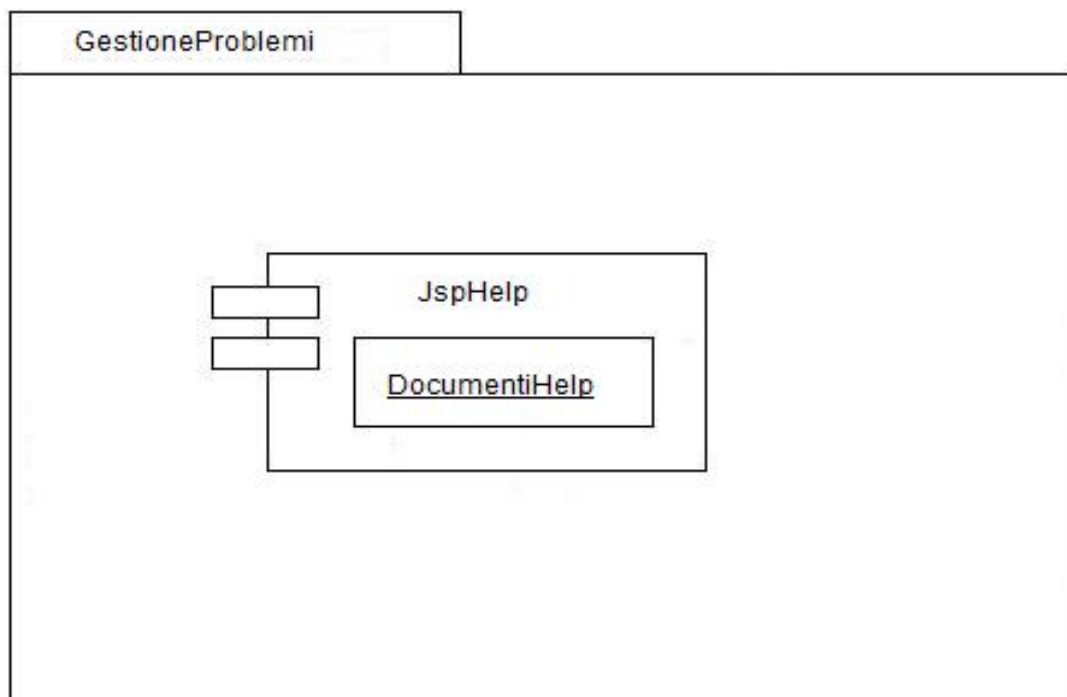
JSP GestioneSessioneLavoro – “Livello Interfaccia”

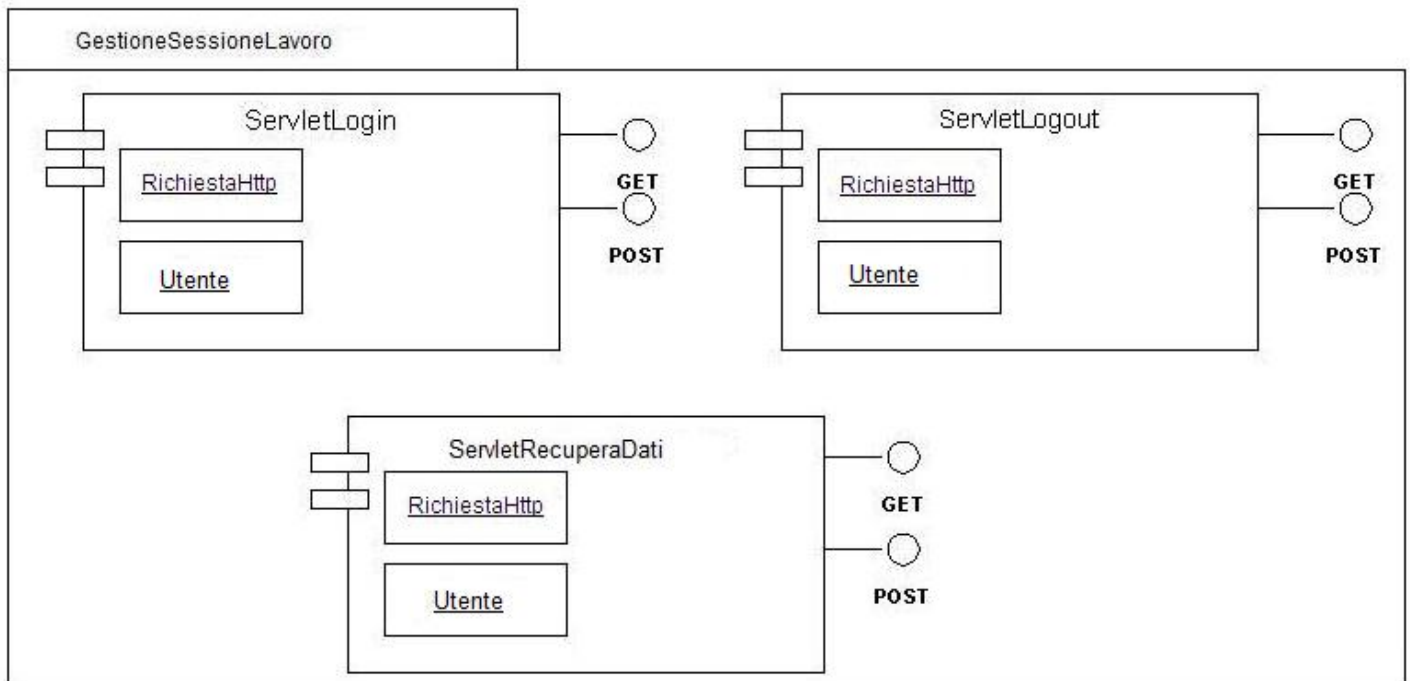
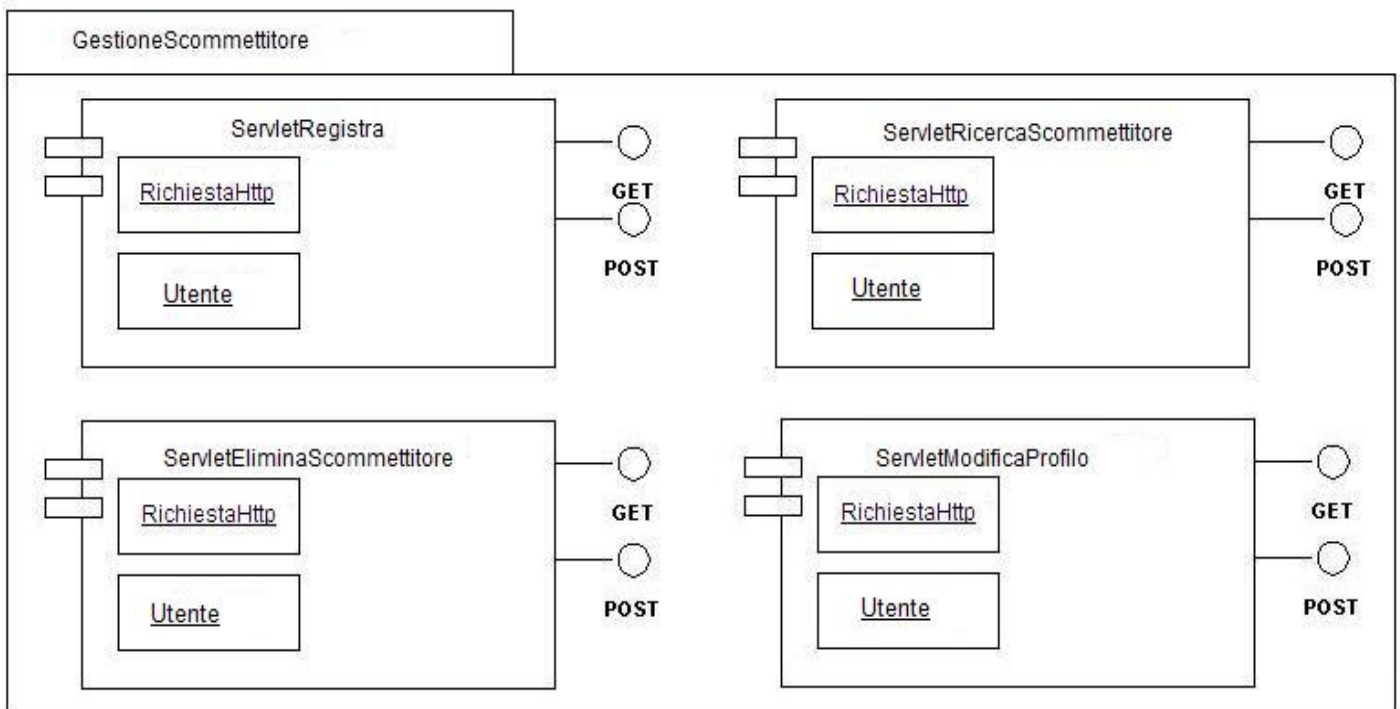


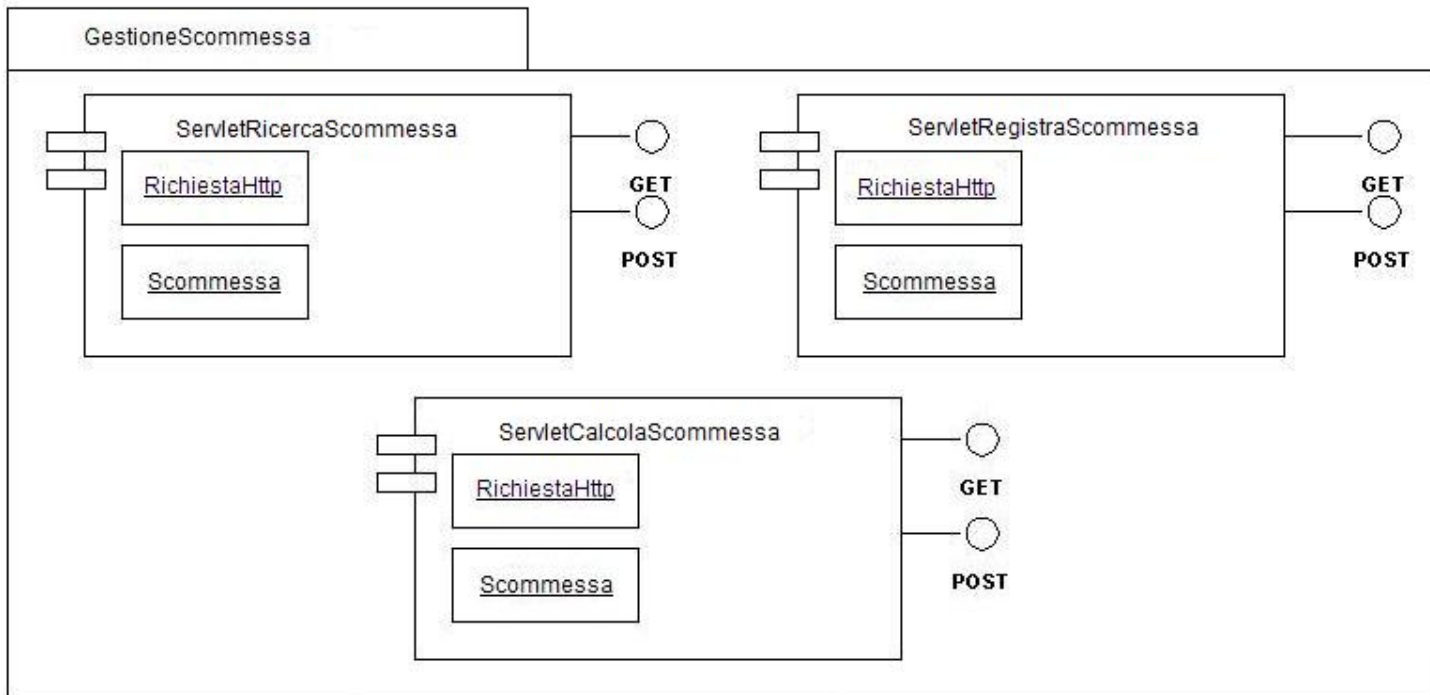
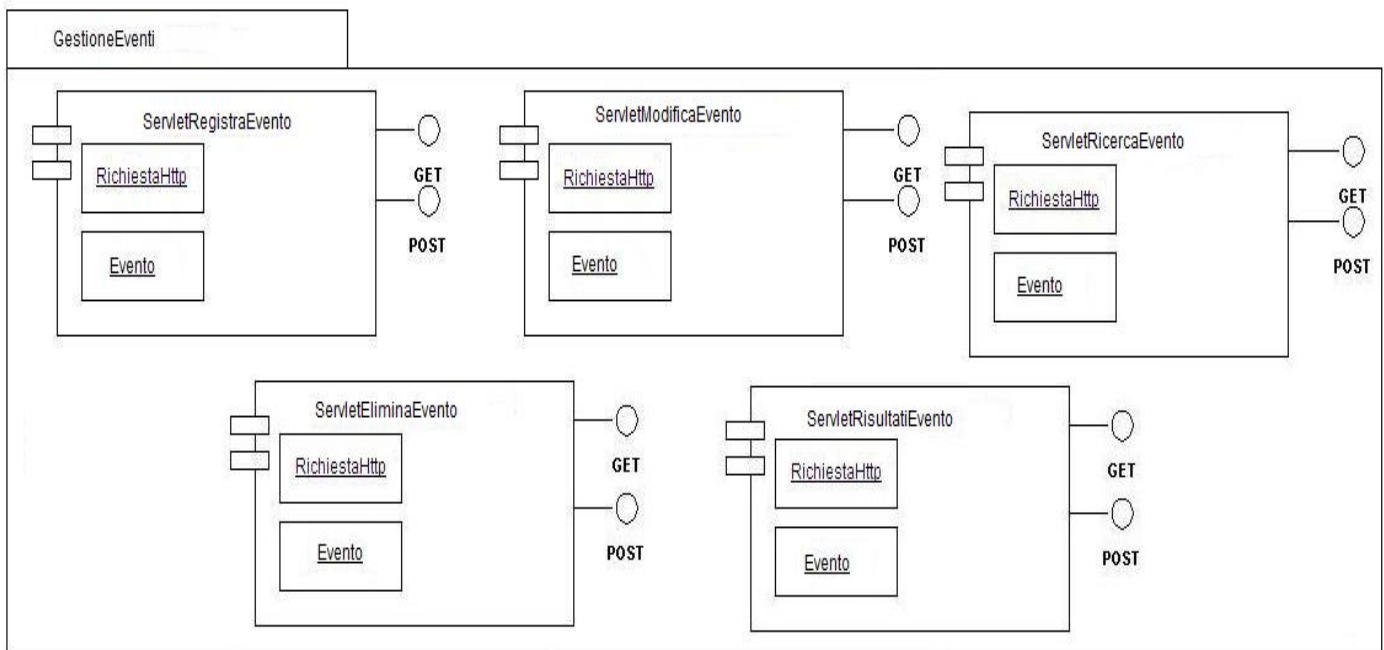
JSP GestioneScommettitore – “Livello Interfaccia”

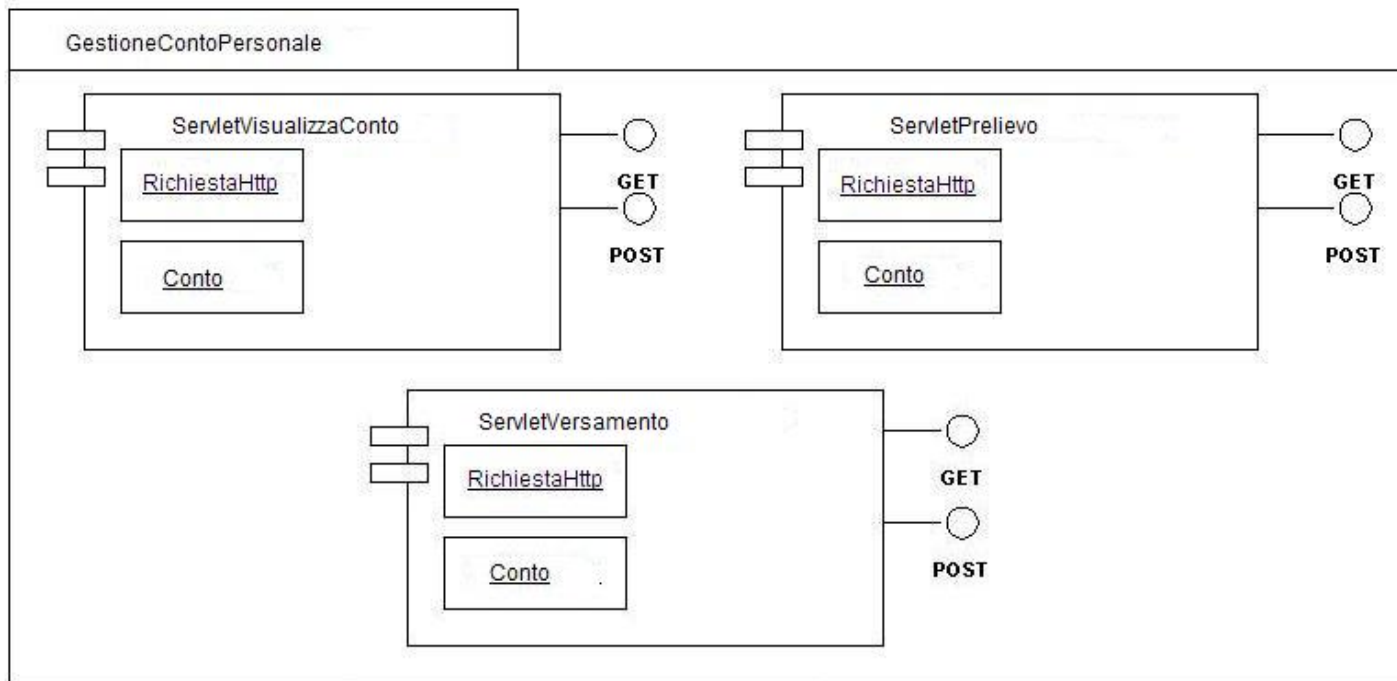
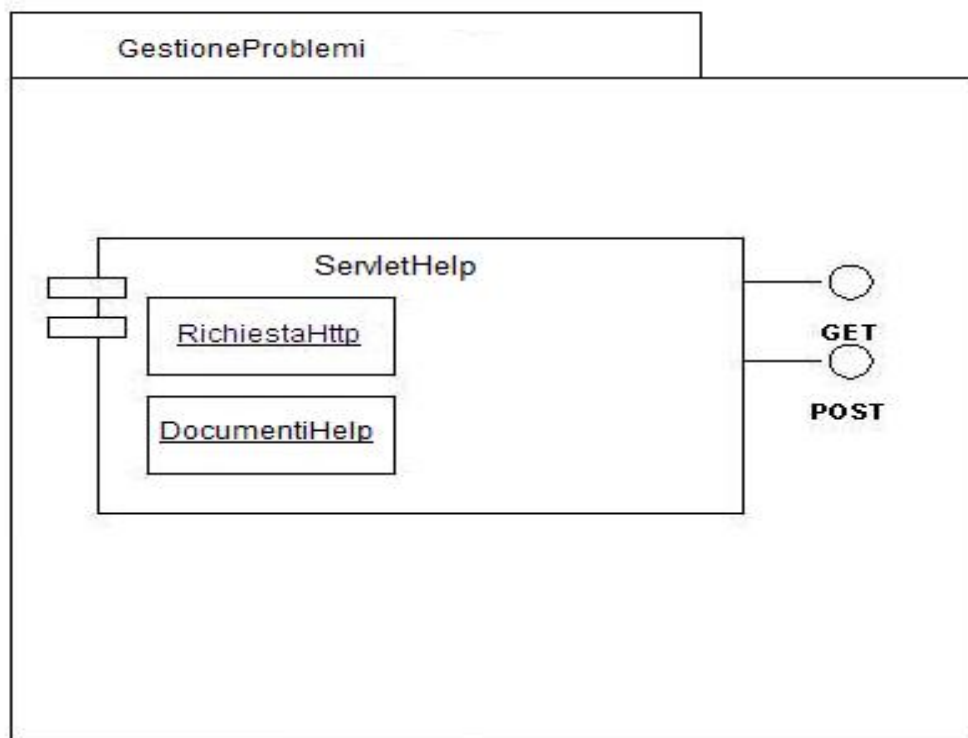


JSP GestioneScommessa – “Livello Interfaccia”**JSP GestioneEventi – “Livello Interfaccia”**

JSP GestioneContoPersonale – “Livello Interfaccia”**JSP GestioneProblemi – “Livello Interfaccia”**

Servlet GestioneSessioneLavoro – “Livello Logico Applicativo”**Servlet GestioneScommettitore – “Livello Logico Applicativo”**

Servlet GestioneScommessa – “Livello Logico Applicativo”**Servlet GestioneEventi – “Livello Logico Applicativo”**

Servlet GestioneContoPersonale – “Livello Logico Applicativo”**Servlet GestioneProblemi – “Livello Logico Applicativo”**

3.4 Gestione dei dati persistenti

La persistenza dei dati sarà effettuata dal sistema al verificarsi di pre-determinati eventi scatenati dall'utilizzatore del sistema.

Gli oggetti da memorizzare in modo persistente sono:

- Utente
- Evento
- Scommessa
- Evento Scommessa
- Conto
- Operazione

Per la memorizzazione dei dati viene scelto un database di tipo relazionale che provvederà alla memorizzazione dei dati in tabelle collegate tra di loro grazie ai valori comuni di alcuni campi delle entuple interessate.

La scelta del database è stata effettuata prendendo in considerazione la quantità e la complessità delle operazioni da effettuare e anche in base alla mole dei dati da memorizzare necessitando anche, trattandosi di un applicazione web-based, di una discreta velocità nel compiere il lavoro richiesto.

Di seguito la struttura fondamentale delle tabelle:

1. Tabella utente

ID_Utente	Si tratta della chiave primaria. Viene inserita dal sistema automaticamente ad ogni nuova registrazione di un utente
username	Il nick name dell'utente
password	La password scelta dall'utente
email	L'indirizzo e-mail dell'utente
nome	Il nome dell'utente
cognome	Il cognome dell'utente
tipo	Indica a che livello di permessi appartiene l'utente, quindi quali sono le operazioni che può e non può effettuare nel sistema

2. Tabella evento

ID_Evento	Si tratta della chiave primaria. Viene inserita dal sistema automaticamente ad ogni nuova registrazione di un evento
descrizione	La descrizione dell'evento inserito, quindi la data, l'ora e altri dettagli specifici
quota	Indica la quota in base alla quale si calcolerà la vincita in caso di un giusto pronostico da parte dell'utente
risultato	Il risultato finale dell'evento

3. Tabella scommessa

ID_Scommessa	Si tratta della chiave primaria. Viene inserita dal sistema automaticamente ad ogni nuova registrazione di una scommessa
data	La data in cui viene effettuata la scommessa
importo	L'importo di denaro che viene scommesso
esito	Contiene il risultato della scommessa
vincita	Contiene la quantità di denaro vinto

4. Tabella evento_scommessa

ID_Scommessa	Si tratta di una chiave esterna che fa riferimento alla tabella "scommessa". Insieme al campo "ID_Evento" costituisce la chiave primaria
ID_Evento	Si tratta di una chiave esterna che fa riferimento alla tabella "evento". Insieme al campo "ID_Scommessa" costituisce la chiave primaria

5. Tabella conto

ID_Conto	Si tratta della chiave primaria. Viene inserita dal sistema automaticamente ad ogni nuova apertura di un conto
saldo	Indica il saldo disponibile sul conto

6. Tabella operazione

ID_Operazione	Si tratta della chiave primaria. Viene inserita dal sistema automaticamente ad ogni nuova registrazione di un evento
descrizione	Descrive l'operazione effettuata
importo	Contiene l'importo dell'operazione

3.5 Controllo degli Accessi e Sicurezza

3.5.1 Controllo accessi

Al sistema potrà accedere l'Amministratore, lo scommettitore o l'utente e a seconda delle credenziali di accesso avrà diritto a leggere e modificare gli oggetti del sistema.

Attore	Utente	Evento	Scommessa	Conto
Amministratore	<<create>> EliminaScommettitore RicercaScommettitore	<<create>> Modifica Evento EliminaEvento RicercaEvento Registra Evento Inserimento Risultato Evento	Calcola Scommessa Vincente Ricerca Scommessa	
Scommettitore	ModificaProfilo	RicercaEvento	<<create>> Registra Scommessa	<<create>> EffettuaPrelievo Effettua Versamento VisualizzaConto
Utente	<<create>> Registrazione	RicercaEvento		

3.5.2 Sicurezza

Il sistema sarà protetto da accessi indesiderati grazie ad un sistema di autenticazione che, tramite una combinazione di login e password fornita ad ogni Amministratore o Scommettitore, regolerà gli accessi al sistema. L'utente visitatore non necessita di autenticazione in quanto avrà la possibilità di effettuare solo operazioni di consultazione.

Nello specifico i dettagli della combinazione login/password sono:

Login	È richiesta una lunghezza compresa fra un minimo di 6 e un massimo di 14 caratteri alfanumerici
Password	È richiesta una lunghezza compresa fra un minimo di 6 e un massimo di 14 caratteri alfanumerici

3.6 Global Software Control

Il sistema software New-Totosi sarà caratterizzato da un portale accessibile da un browser web e da un WebServer, attivo 24 ore su 24, che si occuperà di gestire gli accessi da parte degli utenti.

L'architettura del sistema software, essendo basata su un WebServer, utilizzerà un controllo di flusso di tipo "Event-driven control" secondo il quale un ciclo principale resta in attesa del verificarsi di eventi esterni.

Un insieme di classi Java svolgeranno la funzione di ricevitori di eventi e risponderanno alle richieste inoltrate dal client.

Il client genererà una richiesta che sarà associata ad un evento; la classe preposta a gestire l'evento associato alla richiesta del client, prenderà gli eventuali input inseriti dall'utente e trasferirà al controllo, inoltrando le richieste, alle classi designate per lo svolgimento dell'operazione.

Appena sarà prodotto il risultato, la classe gestore provvederà a inoltrare il risultato al client che aveva generato la richiesta.

Il server principale rimarrà in attesa delle richieste di un client (web browser) e nel momento in cui riceverà una richiesta si occuperà di smistarla alla servlet o alla JSP che avrà il compito di processare la richiesta in questione.

3.7 Boundary conditions

3.7.1 Configurazione

Per ogni oggetto persistente indicato nel punto 3.4 vengono di seguito definite le fasi in cui tale oggetto viene creato , distrutto e archiviato nel database

- **Utente**

L'oggetto utente viene creato dall'Amministratore del database che provvede ad assegnargli una username ed una password; l'archiviazione nella base di dati avviene fino al momento in cui l'Amministratore del database non decide di eliminarlo

- **Evento**

L'oggetto evento viene creato dall' Amministratore all'ingresso del caso d'uso "Registrazione evento" e l'archiviazione avviene all'uscita dello stesso caso d'uso. L'eliminazione dal sistema viene effettuata dall'Amministratore all'interno del caso d'uso intitolato "Cancellazione evento" e si concretizza all'uscita del caso d'uso.

- **Scommessa**

L'oggetto scommessa viene creato dallo Scommettitore all'ingresso del caso d'uso "Registrazione Scommessa" e l'archiviazione avviene al termine del caso d'uso. Lo Scommettitore una volta effettuata la scommessa non può più cancellarla e di conseguenza l'archiviazione avviene fino al momento in cui l'Amministratore non decide di eliminarlo.

- **Evento_Scommessa**

L'oggetto evento_scommessa viene creato dallo Scommettitore all'interno del caso d'uso "Registrazione Scommessa" in cui all'oggetto scommessa vengono associati uno o più eventi. L'archiviazione avviene fino al momento in cui l'Amministratore non decide di eliminarlo.

- **Conto**

L'oggetto Conto viene creato dall'Utente all'interno del caso d'uso "Registrazione Scommettitore"; ogni volta che viene creato un nuovo scommettitore gli viene associato un oggetto Conto e l'archiviazione avviene all'uscita del caso d'uso; l'eliminazione del conto avviene contestualmente alla cancellazione dello Scommettitore da parte dell'Amministratore all'interno del caso d'uso "Cancellazione Scommettitore".

- **Operazione**

L'oggetto Operazione viene creato dallo Scommettitore all'interno del caso d'uso "EffettuaPrelievo" e "EffettuaVersamento"; ogni volta che viene effettuato un versamento o un prelievo viene creato un oggetto Operazione archiviato all'uscita del caso d'uso; l'eliminazione dell'Operazione avviene contestualmente alla cancellazione dello Scommettitore da parte dell'Amministratore all'interno del caso d'uso "Cancellazione Scommettitore".

3.7.2 Casi d'uso Accesso / Uscita

3.7.2.1 Start-up use case (Primo avvio del sistema):

Il sistema software New-Totosi sarà installato dallo sviluppatore sul WebServer. Al primo avvio il sistema crea in automatico l'account per l'Amministratore del sistema. Successivamente il sistema presenterà la visualizzazione della pagina principale.

3.7.2.2 Start-up use case (Avvio successivo al primo in seguito ad eventi eccezionali):

Il sistema dopo aver verificato l'integrità del sistema procederà a visualizzare la pagina principale con i campi per effettuare il login.

3.7.2.3 Start-up use case (Sessione di login):

Nome Use Case:	Autenticazione con successo
Attori Partecipanti:	Utente (Amministratore, Scommettitore)
Flusso degli eventi:	<ol style="list-style-type: none"> 1. Utente: seleziona il bottone "Login" nella sezione dedicata alla gestione delle sessione di lavoro 2. Sistema: presenta una schermata in cui inserire i dati necessari per l'autenticazione 3. Utente: inserisce l'username e la password e seleziona l'opzione "login" 4. Sistema: controlla le credenziali di accesso e se sono corrette invia l'utente alla pagina predefinita a seconda del suo gruppo di appartenenza
Condizione d'ingresso:	L'utente è già registrato nel sistema
Condizioni d'uscita:	Utente Autenticato
Eccezioni	Il sistema non riconosce l'utente e mostra un errore. La connessione viene persa durante il riconoscimento dei dati.
Requisiti di qualità	N/A

3.7.2.4 Shut down (del sistema)

Lo Shut Down del sistema, come descritto nella fase di analisi, non avverrà mai, in quanto il sistema resta attivo 24 ore su 24.

L'unico caso di shut down avverrebbe in caso di situazioni impreviste o di malfunzionamenti.

3.7.2.5 Shut down use case (Logout)

Nome Use Case:	Logout
Attori Partecipanti:	Utente (Amministratore, Scommettitore)
Flusso degli eventi:	<ol style="list-style-type: none"> 1. Utente: seleziona il bottone “Logout” nella sezione dedicata alla gestione della sessione di lavoro 2. Sistema: effettua il salvataggio della sessione di lavoro e disconnette l'utente dal sistema.
Condizione d'ingresso:	L'utente è connesso al sistema
Condizioni d'uscita:	Utente è disconnesso dal sistema
Requisiti di qualità	N/A

Nome Use Case:	Logout
Attori Partecipanti:	Utente (Amministratore, Scommettitore)
Flusso degli eventi:	<ol style="list-style-type: none"> 1. Utente: seleziona il bottone “Logout” nella sezione dedicata alla gestione della sessione di lavoro 2. Sistema: effettua il salvataggio della sessione di lavoro e disconnette l'utente dal sistema.
Condizione d'ingresso:	L'utente è connesso al sistema
Condizioni d'uscita:	Utente è disconnesso dal sistema
Requisiti di qualità	N/A

3.7.3 Condizioni eccezionali

3.7.3.1 WebServer disconnesso dalla rete o in crash

il WebBrowser visualizza un messaggio di errore che avvisa dell'impossibilità di contattare il WebServer. In tale caso tutti i servizi del sistema non saranno disponibili fino a quando il server centrale non sarà riconnesso alla rete o il crash risolto.

3.7.3.2 DBMS disconnesso dalla rete o in crash

I servizi forniti dal sistema software saranno solo in parte disponibili in quanto alcune componenti non possono interagire con il DBMS. Tutte le pagine dinamiche che effettuano ricerche e aggiornamenti del DBMS visualizzeranno un messaggio di errore.

4.Servizi del sottosistema

Di seguito vengono elencati e descritti i servizi offerti da ognuno dei sottosistemi in termini di operazioni.

Sottosistema “Livello Logico Applicativo”

Modulo	<i>GestioneSessioneLavoro</i>
Descrizione	Modulo che regola gli accessi al sistema con le operazioni di login e logout..

Servizi offerti

Servizio	Descrizione
Login	Il servizio di login permette all'utente di inserire le credenziali di accesso (username e password) per poter accedere al sistema software.
Logout	Il servizio di logout permette all'utente che si era precedentemente autenticato di uscire dal sistema.
RecuperaDati	Il servizio di recupero dei dati permette all'utente di recuperare le credenziali di accesso (username e password) per poter accedere al sistema software.

Modulo	<i>GestioneScommettitore</i>
Descrizione	Modulo che si occupa di gestire il profilo dello scommettitore cioè le operazioni di registrazione, cancellazione, modifica e ricerca.

Servizi offerti

Servizio	Descrizione
RegistrazioneScommettitore	Il servizio di Registrazione scommettitore permette di inserire all'interno del DBMS i dati e le credenziali di accesso di un nuovo scommettitore.
CancellazioneScommettitore	Il servizio di Cancellazione scommettitore permette di rimuovere dal DBMS i dati e le credenziali di accesso di uno scommettitore.
ModificaProfilo	Il servizio di Modifica scommettitore permette di modificare i dati e le credenziali di accesso di uno scommettitore.
RicercaScommettitore	Il servizio di RicercaScommettitore permette di ricercare all'interno del sistema uno o più scommettitori che corrispondono a determinati parametri di ricerca.

Modulo	<i>GestioneEventi</i>
Descrizione	Modulo che si occupa della gestione degli eventi cioè le operazioni di ricerca, registrazione, cancellazione e inserimento di un risultato.

Servizi offerti

Servizio	Descrizione
RicercaEvento	Il servizio di RicercaEvento permette di ricercare, tramite parametri di ricerca come quota, nome squadra ecc., uno o più eventi tra quelli precedentemente registrati nel sistema.
RegistrazioneEvento	Il servizio di Registrazione Evento permette inserire e memorizzare nell DBMS i dati di un nuovo evento.
CancellazioneEvento	Il servizio di Cancellazione Evento permette di rimuovere dal sistema software un evento precedentemente registrato.
RisultatoEvento	Il servizio di Inserimento risultato evento permette di inserire il risultato di un evento

Modulo	<i>GestioneContoPersonale</i>
Descrizione	Modulo che si occupa di gestire le operazioni sul conto personale cioè prelievo, versamento e riepilogo del conto.

Servizi offerti

Servizio	Descrizione
Prelievo	Il servizio di Prelievo conto permette di decrementare il credito a disposizione di uno scommettitore per effettuare scommesse e avviare la procedura di pagamento.
Versamento	Il servizio di Versamento conto permette di incrementare il credito a disposizione di uno scommettitore per effettuare scommesse.
VisualizzaConto	Il servizio di Riepilogo del Conto permette di visualizzare i dettagli di un conto di gioco: saldo, prelievi, versamenti, giocate vincenti e giocate non vincenti.

Modulo	<i>GestioneScommessa</i>
Descrizione	Modulo che si occupa di gestire la scommessa cioè le operazioni registrazione, ricerca e di calcolo vincite.

Servizi offerti

Servizio	Descrizione
RegistrazioneScommessa	Il servizio di Registrazione scommessa permette di inserire nel DBMS i dati relativi ad una nuova scommessa effettuata da un utente scommettitore.
RicercaScommessa	Il servizio di RicercaScommessa permette di ricercare , in base a determinati parametri di ricerca, una o più scommesse tra tutte quelle precedentemente registrate nel sistema software.
CalcoloScommessaVincente	Il servizio di Calcolo vincite permette di ricercare e verificare l'esito di tutte le scommesse registrate nel sistema e avviare la sincronizzazione che permetterà di aggiornare i saldi dei conti di gioco presenti nel sistema.

Modulo	<i>GestioneProblemi</i>
Descrizione	Modulo che si occupa della guida per la risoluzione dei problemi.

Servizi offerti

Servizio	Descrizione
Help-Online	Il servizio di help permette di consultare i documenti relativi alla risoluzione degli errori (la guida, e il regolamento).