

Tesi di Laurea

Progettazione e sviluppo di una web application per la consuntivazione di attività

Andrea De Boni

Facoltà di Scienze MM.FF.NN.
Università degli Studi di Padova

23/07/2009



L'azienda



IKS S.r.l. nasce nel 1999 dall'incontro di professionalità di alto profilo.

- Team in grado di affrontare le più complesse problematiche della moderna Information e Communication Technology.
- Sicurezza informatica.
- Gestione ed automazione di infrastrutture di ICT.
- Sviluppo di soluzioni software complesse aderenti agli standard Java/J2EE.



Scopo dello stage

Lo scopo dello stage è quello di progettare e realizzare una web application per la consuntivazione di attività.

- Analizzare e progettare le varie entità:
 - Progetti
 - Attività
 - Clienti
 - Team di lavoro
 - Utenti
- Definire la base dati associata.
- Implementare la soluzione.
- Eseguire i test.



Panoramica applicazione 1

Il prodotto consente di avere una visione globale sullo stato dei contratti stipulati in un'azienda.

I contratti possono essere suddivisi in contratti "a consuntivo" e in contratti "a progetto".

- Contratti "**a consuntivo**": si basano su un budget a disposizione del cliente, terminato il budget, il progetto può considerarsi concluso.
- Contratti "**a progetto**": sono caratterizzati da una scadenza, e da un importo iniziale, si concludono quando l'attività è stata completata.



Panoramica applicazione 2

- In entrambe le tipologie, un contratto coincide con l'inizio di un **progetto**.
- Un progetto comprende l'insieme delle **attività** definite dal contratto, la somma del budget e le scadenze delle attività.
- Un progetto è assegnato ad un **team** di lavoro, composto da persone che possono ricoprire uno o più **ruoli** nel progetto stesso.
- Gli **utenti** che partecipano al progetto devono inserire le attività svolte per il progetto, con dettaglio giornaliero.



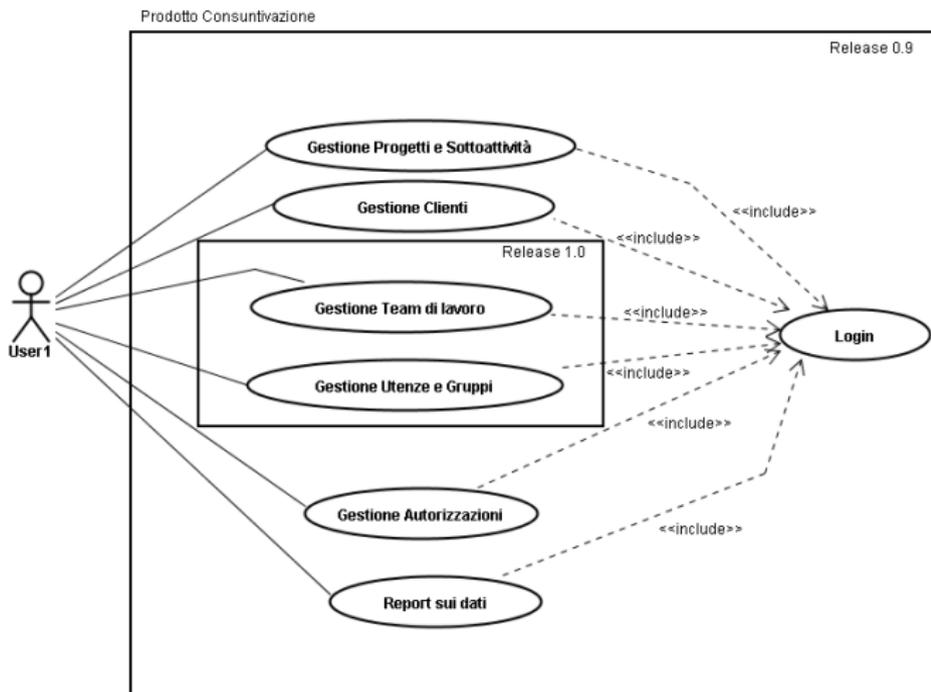
Funzionalità dell'applicazione

Il prodotto fornisce ad un utente dell'applicazione, le seguenti funzionalità:

- Gestione **progetti** e **attività**.
- Gestione **clienti**.
- Gestione **team** di lavoro.
- Gestione **utenze** e relativa appartenenza a **gruppi**.
- **Reportistica** sui dati.
- **Autorizzazioni** utenti.



Use Case d'inizio applicazione



Architettura della web application

L'applicazione sviluppata consta di 3 livelli :

- **Presentation Layer:** Java Server Faces per costruire lo strato di presentazione, permette di creare delle GUI ricche per applicazioni web, in questo strato convivono pagine Jsp e componenti JSF.
- **Business Logic Layer:** JavaBeans di servizio e classi Java che implementano la logica vanno a creare lo strato Business.
- **Data Access Layer:** Lo strato di accesso ai dati gestisce la logica per salvare e recuperare i dati dal database.
Hibernate gestisce la logica per salvare e recuperare i DAO (Data Access Objects).



I JavaBeans

I **JavaBeans** sono classi scritte in linguaggio di programmazione Java secondo particolari convenzioni.

Vengono usati come *contenitori per le informazioni* passate:

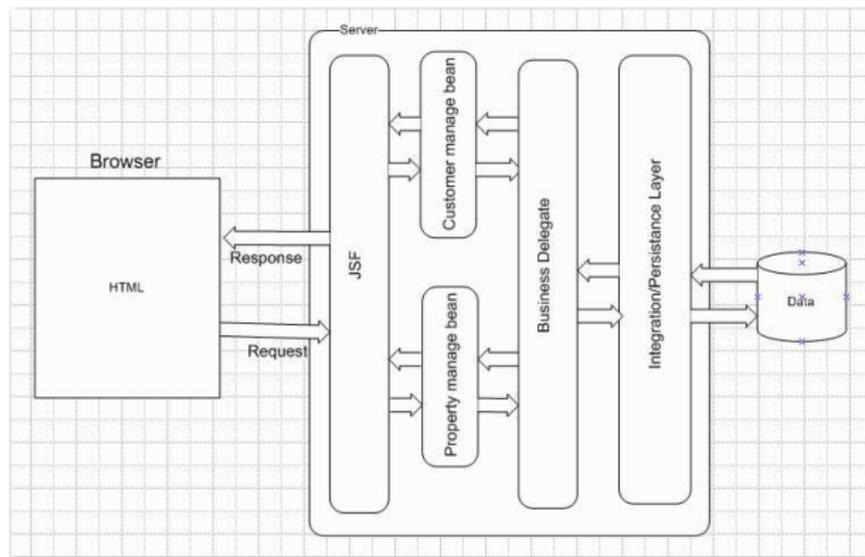
HTML <—> Codice Java <—> DataBase

Convenzioni:

- La classe deve avere un costruttore senza argomenti.
- Le sue proprietà devono essere accessibili usando get, set e altri metodi (detti metodi accessori), seguendo una convenzione standard per i nomi.
- La classe deve essere serializzabile (capace di salvare e ripristinare il suo stato in modo persistente).



Schema architetturale dell'applicazione



Java Server Faces

Il framework, utilizzato ai fini della realizzazione della logica di presentazione, è **Java Server Faces**.

- Tecnologia Java basata sul design pattern architetturale Model-View-Controller (MVC).
- Descritta da un documento alla cui stesura hanno partecipato aziende quali IBM, Oracle, Siemens e Sun Microsystems.
- Semplifica lo sviluppo dell' interfaccia utente di una applicazione Web.
- Framework per componenti lato server di interfaccia utente



Vantaggi del framework JSF

Vantaggi:

- Grosse aziende (Oracle, IBM, JBoss) supportano l'implementazione JSF, c'è quindi un buon livello di supporto.
- Permette di creare componenti riutilizzabili, aumenta la consistenza e la produttività.
- Esistono molti componenti pronti all'uso (Apache, Richfaces, Oracle ...).
- Il codice JavaScript è integrato nei componenti, causa meno confusione agli sviluppatori e permette un maggiore riutilizzo del codice.



Richfaces

- È una libreria di componenti per JSF.
- Framework avanzato per realizzare una facile integrazione di funzionalità AJAX nello sviluppo di applicazioni web-based.
- Da supporto orientato alle pagine al posto del supporto orientato agli oggetti (AJAX).
- È possibile definire un evento in una pagina che invoca una particolare richiesta AJAX.



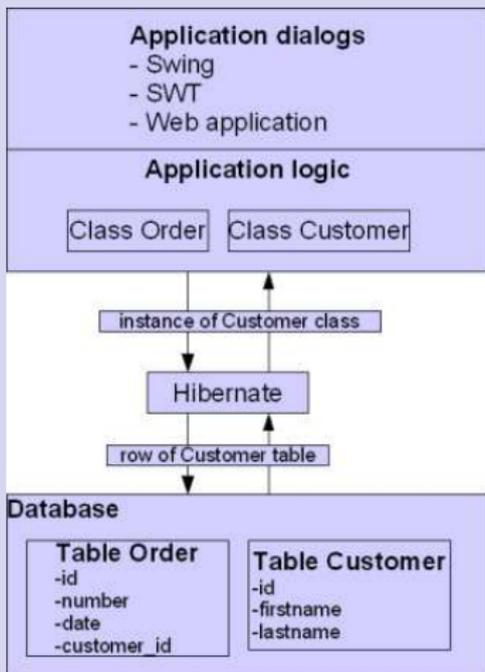
Hibernate

Hibernate è una soluzione ORM per la persistenza dei dati che consente di automatizzare le procedure CRUD del database.

Vantaggi:

- **Elevata portabilità** rispetto al database relazionale utilizzato, il programma non è quindi vincolato all'uso di un particolare RDBMS, nel caso di cambiamento della base dati basta cambiare poche righe nel file di configurazione del ORM.
- **Facilità d'uso**, non c'è bisogno di usare direttamente il linguaggio SQL.





Persistenza dei dati

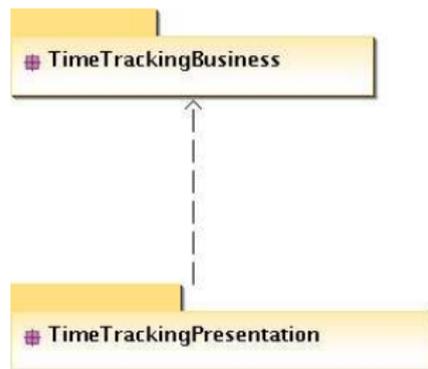
- Per gestire la persistenza dei dati si è optato per l'uso del RDBMS open-source **MySQL**
- È stato utilizzato anche il connettore per il database **MySQL Connector/J** che fornisce la connettività per le applicazioni scritte in Java al database relazionale.
- L'applicazione deve essere però **portabile** su diversi RDBMS, in modo da garantire all'azienda il suo funzionamento anche nel caso in cui quest'ultimo venga cambiato.



Primo livello di decomposizione

L'applicazione al suo livello più alto si divide in due "macro" package composti a loro volta da vari sottopackage:

- **TimeTrackingBusiness**: Gestisce le regole di business-logic e il prelievo/salvataggio su database.
- **TimeTrackingPresentation**: Gestisce tutti gli oggetti della GUI e le chiamate alla parte di business-logic.



Package di TimeTrackingBusiness

- **it.iks.timetrack.util**: contiene classi di varia utilità richiamate anche da classi di TimeTrackingPresentation (contiene la classe Logger, usata per la creazione di log delle transizioni con il database).
- **it.iks.timetrack.businesslogic.beans**: contiene tutte le classi che fungono da Bean per lo strato business-logic.
- **it.iks.timetrack.businesslogic.dao**: contiene i Data Access Objects (DAO) che operano nel Data Access Layer.
- **it.iks.timetrack.businesslogic.manager**: contiene le interfacce con i metodi accessibili dal lato Presentation (TimeTrackingPresentation).



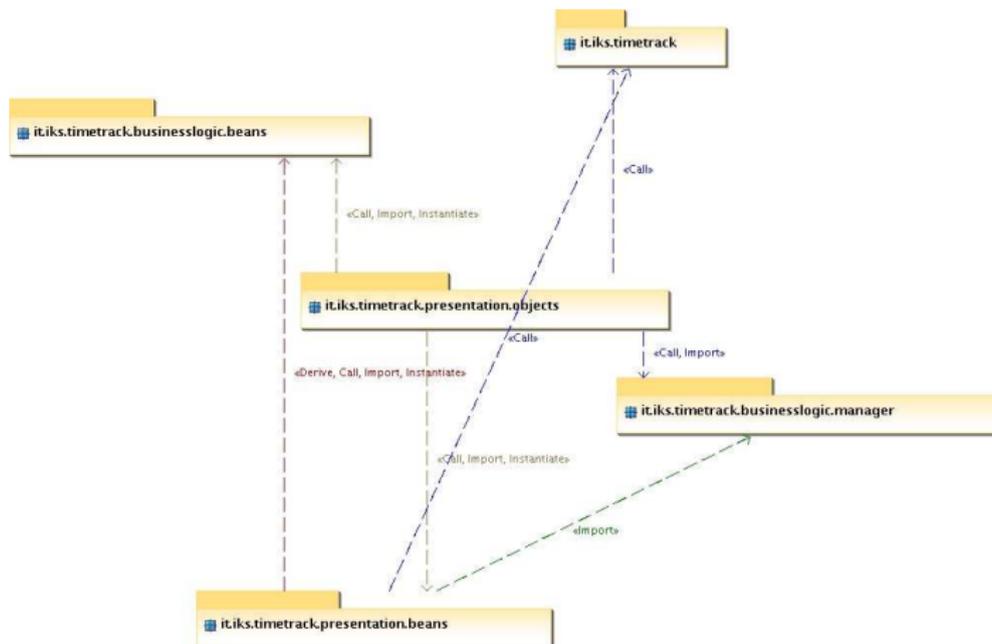
Package di TimeTrackPresentation

- **it.iks.timetrack.presentation.beans**: contiene tutte le classi che fungono da Bean per lo strato presentation.
- **it.iks.timetrack.presentation.objects**: contiene gli oggetti veri e propri della GUI.



Livelli di decomposizione

Relazione fra i package del sistema



Presentation Objects

Le classi denominate "**Presentation Objects**" vengono utilizzate dall'interfaccia grafica per interrogare il database e visualizzare i risultati prodotti.

A loro volta, queste classi utilizzano i metodi delle classi del package `it.iks.timetrack.businesslogic.manager`, questo separa la parte model dalla view.



Package it.iks.timetrack.businesslogic.manager

- Package che fornisce delle interfacce per la parte di presentation.
- Si sfrutta il pattern di design **Facade**, che permette, attraverso un'interfaccia più semplice, l'accesso a sottosistemi che espongono interfacce complesse e molto diverse tra loro.
- Le classi lato presentation, utilizzeranno i metodi forniti dall'interfaccia, senza preoccuparsi di come vengono realizzate tutte le operazioni da e verso il database.



Package it.iks.timetrack.businesslogic.dao

- Classi preposte all'interrogazione del database e alla lettura e scrittura dei JavaBeans sul DBMS.
- Hibernate:
 - Lega le funzionalità presenti di business-logic ai dati del database.
 - Permette un facile uso di tutta la logica di business sugli oggetti interagendo direttamente con il database.
 - Fornisce la soluzione per mappare le tabelle del database alle classi dell'applicazione.



Classi JavaBeans

Lato Business.logic - Data Access:

- Per comunicare con il database dobbiamo utilizzare i JavaBeans visti in precedenza del package **it.iks.timetrack.businesslogic.beans**, che mappano le informazioni relative alle tabelle del database.

Lato Presentation:

- Le classi del package **it.iks.timetrack.presentation.beans** contengono i dati da scambiare dal lato presentation verso lo strato data access.
- I presentation beans sono un'estensione dei beans lato business-logic, con delle funzionalità in più che risultano utili nello strato di presentation.



Screenshot dimostrativo

Pagina dove un utente può consuntivare del tempo impiegato per una certa attività:

Sezione Consuntivazione Attività

Info Progetto	Info Task
Nome :Quarta	Progettazione
Id :276	Id :464

Consuntiva Attività

Scegli la figura professionale

Progettista

Dettagli attività

Descrizione:
inserisci la descrizione....

Data:

1

[CREA](#)



Conclusioni e sviluppi futuri

- Il software sviluppato è utilizzabile dall'azienda nella sua prima versione 0.9 .
- Con il rilascio della versione 1.0, l'azienda comincerà ad usare l'applicazione al posto di quella già esistente per la consuntivazione delle attività, prima per alcuni settori dell'azienda e poi estendendone l'uso a tutti i dipendenti.
- L'applicazione verrà data in mano ai grafici dell'azienda, che costruiranno un foglio di stile CSS per renderla più presentabile (finora non è stato curato lo stile grafico).
- Successivamente, quando si sarà raggiunta una versione completamente stabile, il software verrà messo in produzione per essere venduto anche al di fuori del contesto aziendale.

