



Ingegneria del Software

AA 2014/2015

Progetto *Book Sharing* di Marco Strambini

documentazione e sorgenti presso repository GitHub:

<https://github.com/marcostrambini/ProgettoIngegneria.git>



Introduzione ed Obiettivi

Per condividere un proprio patrimonio librario, le principali soluzioni per un utente privato ricadono principalmente nel doversi appoggiare a piattaforme online che per tale servizio, spesso pongono dei limiti sul materiale condiviso utilizzando un approccio “royalty free”, e/o richiedono il pagamento di commissioni (d’ora in avanti “fees”) al buon esito di un’operazione di condivisione.

L’obiettivo è allora proporre un alternativa realizzando un software indipendente che sia privo di fees e di limitazioni nel suo utilizzo, che permetta comunque ad un privato di condividere il proprio patrimonio librario con facilità attraverso la rete.



Ciclo di Vita e Processo di Sviluppo

Ciclo di Vita

- Analisi di fattibilità
- Progettazione
- Assegnazione delle risorse alle varie attività
- Definizioni di scadenze e Milestone
- Produzione
- Testing e Verifica
- Rilascio



Ciclo di Vita e Processo di Sviluppo

Processo di Sviluppo

Modello utilizzato: Modello a Spirale

- Il progetto ha richiesto una prima fase di analisi generica
- Sviluppo in base ai bisogni emersi nelle prime fasi
- Creazione di layout temporanei per verificarne la bontà
- Scrittura di codice ad hoc per le fasi implementate
- Buon modello per effettuare test consistenti nel breve periodo
- Permette flessibilità al prodotto rilasciando versioni in tempi differenti



Documento di Vision

In Sintesi

Il problema di	condividere patrimonio librario privato
Interessa	privati
Il cui impatto è	indipendenza e assenza di limitazioni e fees
Una soluzione sarebbe	Un sistema client/server per la gestione e la condivisione dei dati dei patrimoni privati, con l'ausilio di un database

Documento di Vision

Chi utilizzerà questo prodotto

Chi	privato
Per	condividere patrimonio librario
Prodotto	sistema di condivisione patrimonio librario privato
Che	indipendente e privo di fees e limitazioni
Diversamente da	Amazon, ebay

Documento di Vision

Perché si dovrebbe utilizzare questo prodotto?

Per evitare di doversi appoggiare a servizi a pagamento e dover subire limitazioni di utilizzo e/o a pagare fees per l'effettivo utilizzo totale o parziale di un articolo.

Il servizio è destinato agli utenti privati.



Documento di Vision

Parti interessate

Stakeholder	Descrizione	Responsabilità
privato	gestisce il proprio patrimonio librario	utilizza funzionalità del sistema
system manager	gestisce il sistema di condivisione	responsabile dell'intera infrastruttura



Documento di Vision

Attori del sistema

Nome	Descrizione	Stakeholder
amministratore	persona responsabile dell'intera struttura	system manager
utente base	la persona che utilizza le funzionalità del sistema	privato



Documento di Vision

Business Needs

Nome	Descrizione
Gestione patrimonio librario	si deve permettere la gestione del proprio patrimonio librario esistente
Ricerca libri e condivisione	si deve permettere la ricerca di nuovi libri e la gestione della condivisione con altri utenti utilizzatori



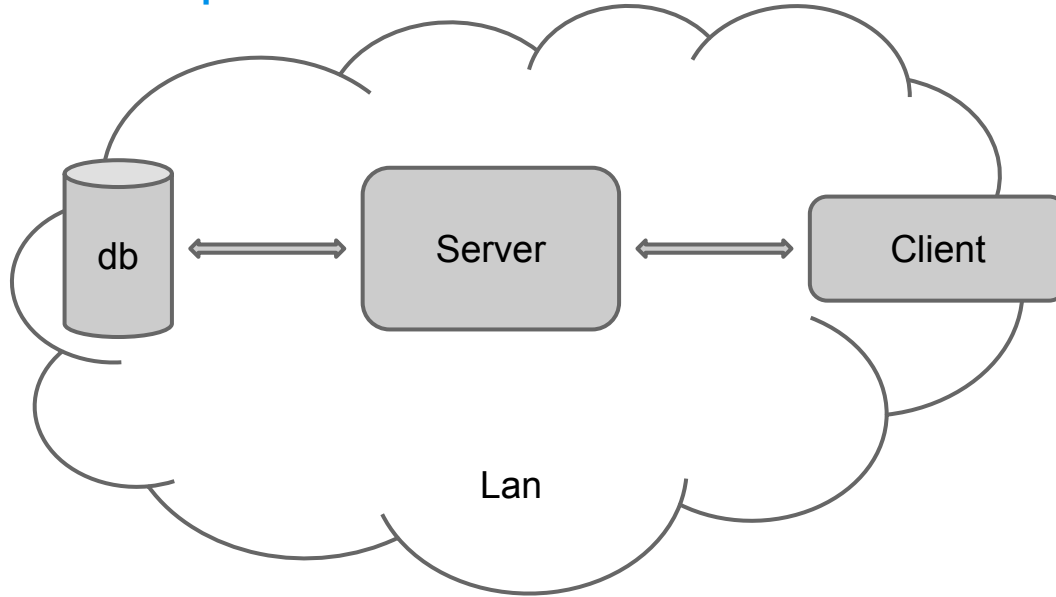
Documento di Vision

Requisiti utente

Nome	Descrizione
Gestione del profilo personale	La possibilità di pubblicare il proprio patrimonio librario e di gestire una pagina personale
Suddivisione contenuti per categorie	La possibilità di ricercare contenuti in base a criteri scelti dall'utente
Individuazione utenti limitrofi	La possibilità di ricercare utenti geograficamente vicini
Gestione delle condivisioni	La possibilità di richiedere il prestito del libro ad altri utenti
Statistiche di utilizzo	La possibilità di evincere informazioni sull'utilizzo del sistema

Documento di Vision

Concetto Operativo



Documento di Caratteristiche

Architettura

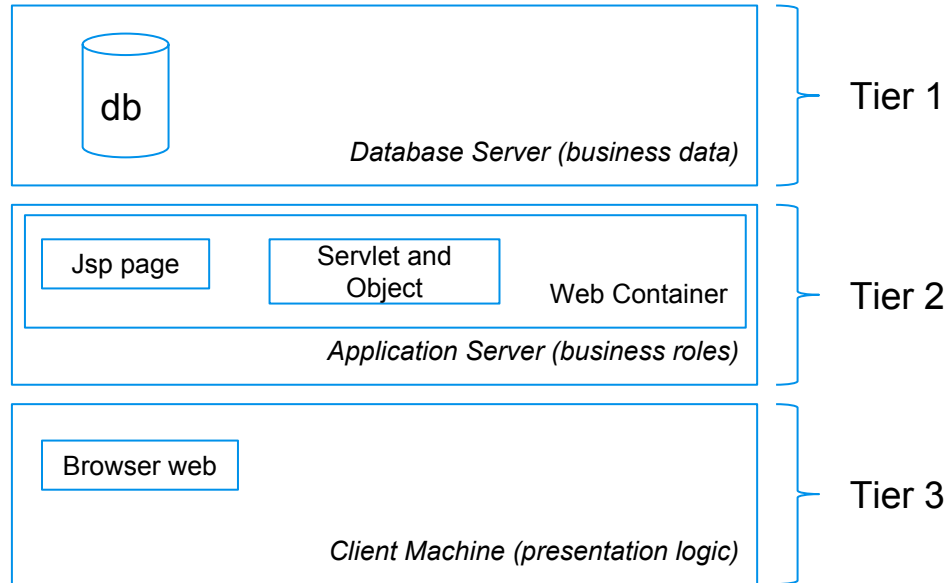
Il prodotto è interamente software, con struttura a tre livelli (“three tiers”):

- livello 1: database management system (DBMS)
- livello 2: implementazione di tutte le logiche di interazione tra utente e database
- livello 3: interfaccia utente client



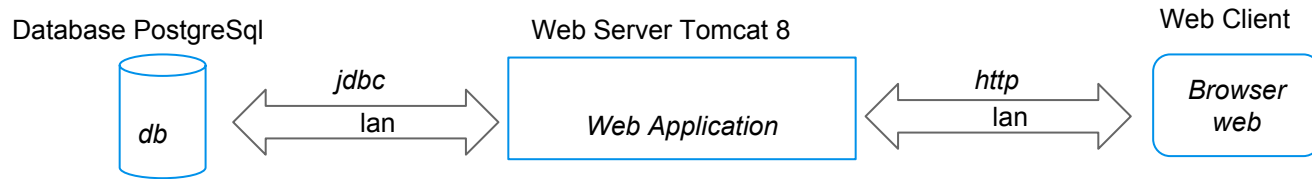
Documento di Caratteristiche

Modello Logico



Documento di Caratteristiche

Modello Fisico



Documento di Caratteristiche

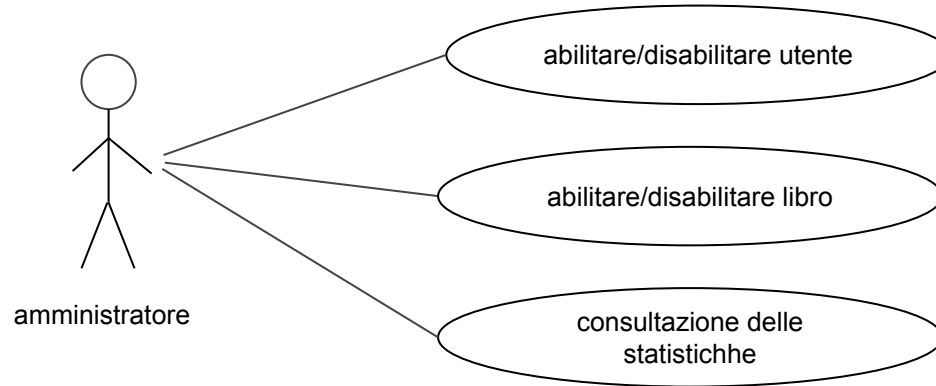
Tecnologie

Protocollo di rete	TCP/IP
Ambiente	Java 2 Enterprise Edition
Sistema Operativo	Windows 7
Motore di Database	PostgreSql 8.0
Application Server	Apache Tomcat 8



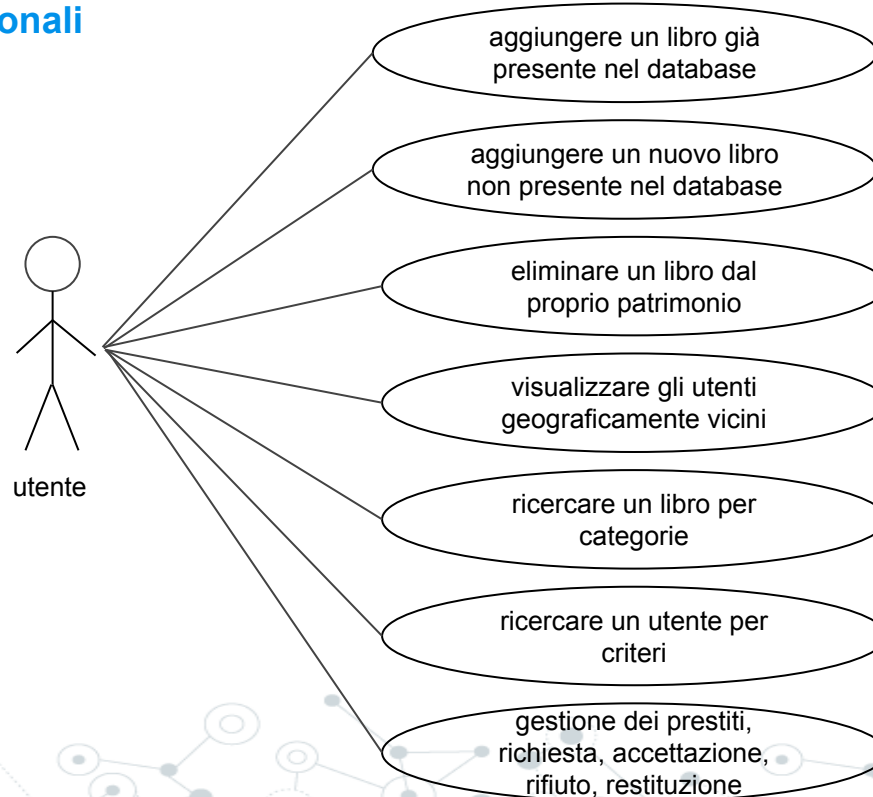
Documento di Caratteristiche

Requisiti Funzionali



Documento di Caratteristiche

Requisiti Funzionali



Documento di Caratteristiche

Requisiti Non Funzionali

- **Robustezza** (vengono gestite le eventuali eccezioni)
- **Sicurezza** (viene effettuato un controllo sulla validità del login)
- **Prestazioni** (ottimizzate operazioni di richiesta al db con l'uso di bean)
- **Interoperabilità** (html visualizzabile da dispositivi che supportano il protocollo. Responsive)
- **Scalabilità** (in base al carico di utilizzo è possibile estendere il web server e dbms. La tipologia di scelta può dipendere dai costi)
- **Privacy** (i dati di login caratteristici di un utente sono conoscibili solo ed unicamente allo specifico utente che li ha inseriti. L'amministratore di sistema può disabilitare un utente, ma non può comunque accedere comunque alle sue credenziali.

I dati sulle ricerche dello specifico cliente non sono visualizzabili all'esterno, ma unicamente al cliente.

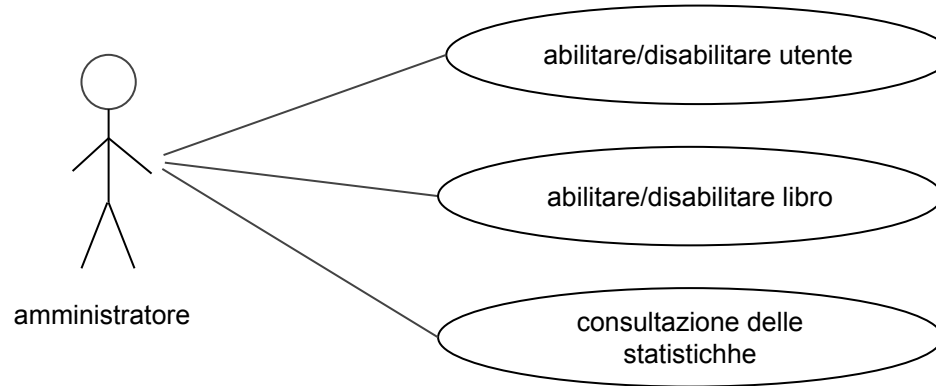
Tali non sono girate a terzi a fini commerciali, ma vengono dal software solo per calcolare le statistiche sui contenuti, non sul comportamento utente)



Specifica dei Casi d'Uso

Il sistema prevede due attori, l'amministratore e l'utente

Funzioni per l'attore Amministratore



Specifica dei Casi d'Uso

Funzioni per l'attore Utente



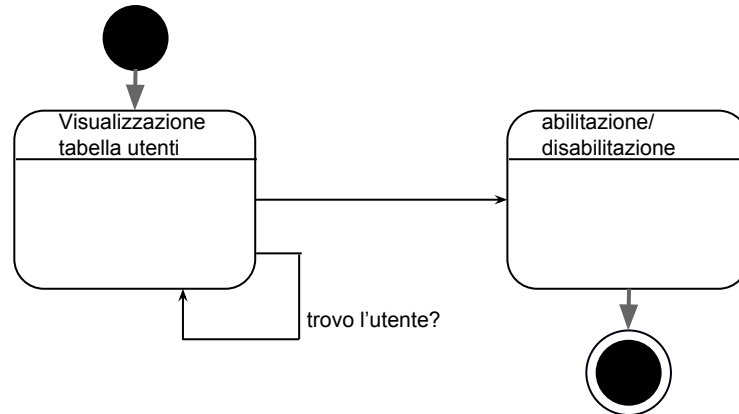
Specifica dei Casi d'Uso

Specifica Caso d'Uso Amministratore

abilitare/disabilitare utente

L' amministratore accede alla propria pagina dove può scegliere di visualizzare una tabella contenente tutta la lista utenti e il loro stato di attivazione corrente. Tramite collegamenti presenti nella tabella è possibile attivare/disattivare l'utente.

Flusso



Pre-condizioni

Database popolato correttamente, nello specifico la tabella utenti.

Post-condizioni

Se l' amministratore abilita/disabilita un utente, il sistema aggiorna lo stato dell'utente nella tabella.

Qualora l'utente sia disabilitato, il suddetto non potrà fare il login.

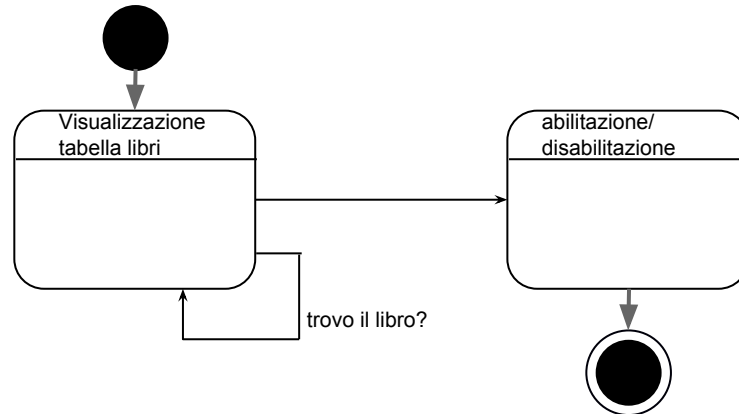
Specifica dei Casi d'Uso

Specifica Caso d'Uso Amministratore

abilitare/disabilitare libro

L' amministratore accedere alla propria pagina dove può scegliere di visualizzare una tabella contenente tutta la lista dei libri e il loro stato di attivazione corrente. Tramite collegamenti presenti nella tabella è possibile attivare/disattivare il libro.

Flusso



Pre-condizioni

Database popolato correttamente, nello specifico la tabella libro.

Post-condizioni

Se l' amministratore abilita/disabilita un libro, il sistema aggiorna lo stato del libro nella tabella.

Qualora il libro sia disabilitato, il suddetto non sarà più disponibile.

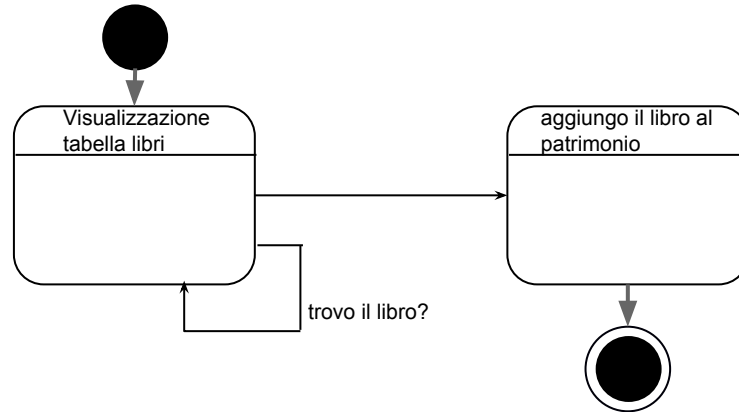
Specifica dei Casi d'Uso

Specifica Caso d'Uso Utente

aggiungere un libro già presente nel database

L'utente ha la possibilità di selezionare tramite una tabella, un libro già esistente nel database. Viene creata dunque una relazione tra l'utente e il libro selezionato.

Flusso



Pre-condizioni

Database popolato correttamente, nello specifico la tabella libro.

Post-condizioni

Creata la relazione di possesso tra utente e libro.

Specifica dei Casi d'Uso

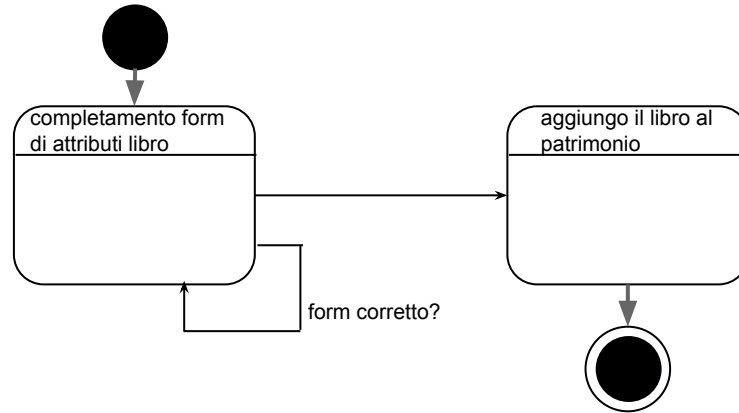
Specifica Caso d'Uso Utente

aggiungere un libro non presente nel database

L'utente ha la possibilità di aggiungere manualmente le informazioni relative al libro non contenuto nel database: nome, autore, immagine.

Viene creata dunque una relazione tra l'utente e il libro aggiunto.

Flusso



Pre-condizioni

Nessuna

Post-condizioni

crea relazione tra l'utente e il libro aggiunto

aggiunge un record nella tabella dei libri per il suddetto libro, ponendo lo stato "attivo".

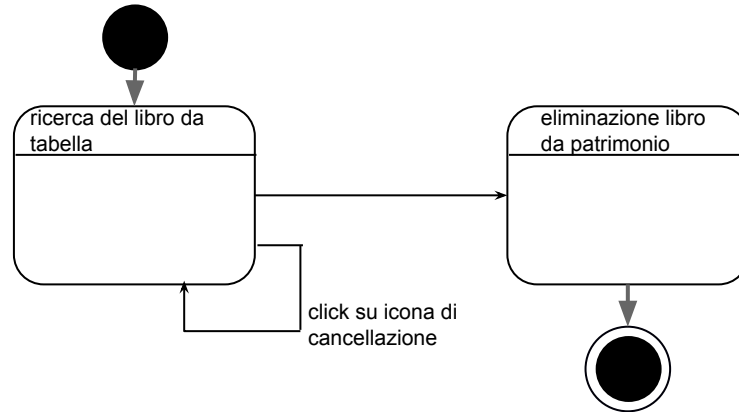
Specifica dei Casi d'Uso

Specifica Caso d'Uso Utente

eliminare un libro dal proprio patrimonio

L'utente ha la possibilità di eliminare un libro dal proprio patrimonio

Flusso



Pre-condizioni

Il libro deve essere presente nel patrimonio librario dell'utente

Post-condizioni

viene eliminata la relazione tra l'utente e il libro aggiunto

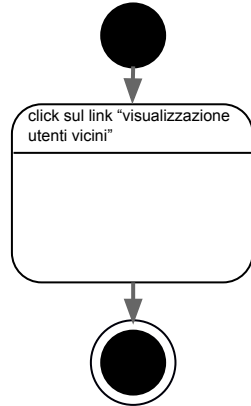
Specifica dei Casi d'Uso

Specifica Caso d'Uso Utente

visualizzazione utenti geograficamente vicini

L'utente ha la possibilità di consultare una tabella contenente gli utenti ordinati per distanza geografica dal più vicino al più lontano

Flusso



Pre-condizioni

la tabella degli utenti deve essere popolata correttamente

Post-condizioni

la selezione dal menu mostra tutti gli utenti con indicazione delle distanze

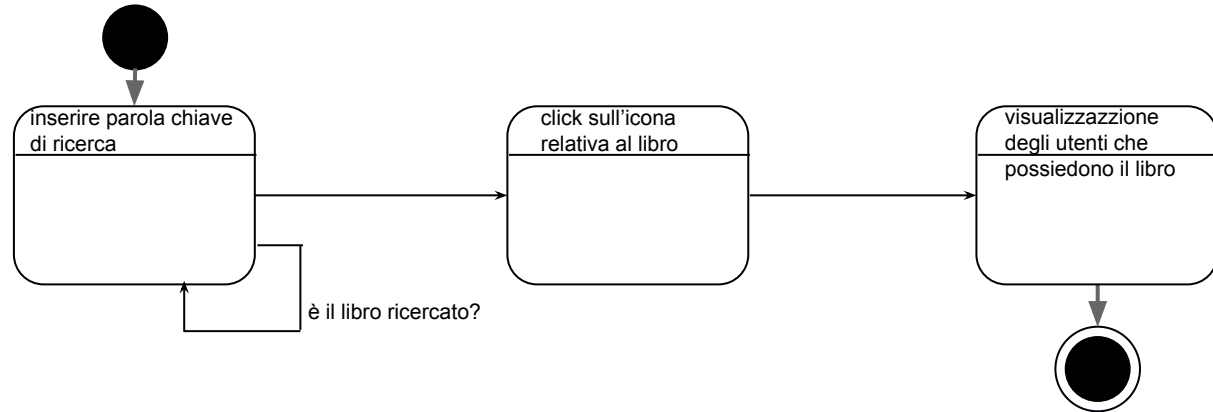
Specifica dei Casi d'Uso

Specifica Caso d'Uso Utente

ricercare un libro per categorie

L'utente ha la possibilità di ricercare un libro in base a delle categorie o criteri a sua scelta

Flusso



Pre-condizioni

il libro deve essere correttamente inserito nel database

Post-condizioni

viene ritornata la lista degli utenti che possiedono il libro scelto e che non lo abbiamo dato in prestito

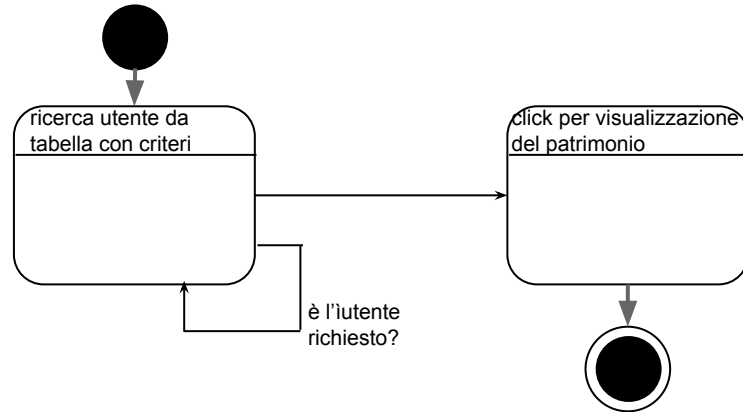
Specifica dei Casi d'Uso

Specifica Caso d'Uso Utente

ricerca di un utente

L'utente ha la possibilità di ricercare uno specifico utente e visualizzare il suo patrimonio librario

Flusso



Pre-condizioni

l'utente deve essere correttamente inserito nel database

Post-condizioni

viene ritornata la pagina pubblica dell'utente selezionato

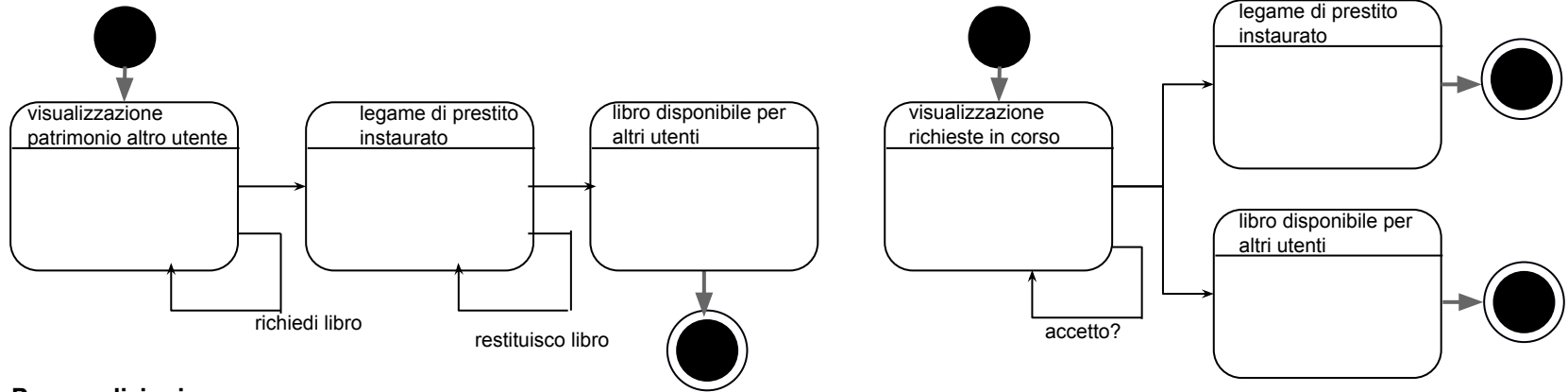
Specifica dei Casi d'Uso

Specifica Caso d'Uso Utente

gestione dei prestiti libro

L'utente ha la possibilità di gestire i prestiti con funzionalità di: richiesta, respinta, accettazione,

Flusso



Pre-condizioni

nessuna

Post-condizioni

viene instaurato il legame di prestito o annullato

Project Plan

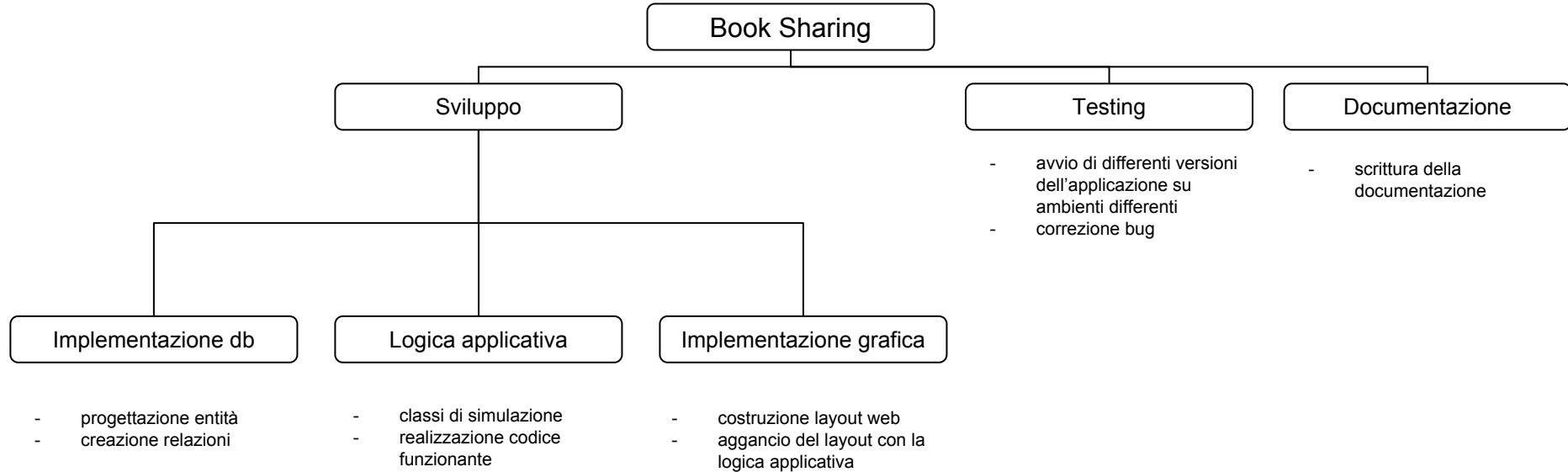
Gestione del lavoro collaborativo

- Il carico di lavoro è stato suddiviso a gruppi di due persone dove ogni persona si occupa di almeno 2 moduli differenti della WBS
- Generate intersezioni di attività in modo da garantire la business continuity in caso di indisponibilità di una risorsa umana, ad eccezione di singolari attività, dove le skill del collaboratore permettevano di guadagnare notevole tempo. Il tempo guadagnato è stato investito nella ottimizzazione del software e nella fase di test
- Effettuati check point da pianificazione e check point in caso di bisogno
- Condivisione delle conoscenze all'interno del gruppo
- Caricamento del materiale online su repository GitHub consultabile dai componenti del gruppo



Project Plan

WBS (Work Breakdon Structure)



Project Plan

OBS (*Organization Breakdon Structure*)

<i>Realizzazione db</i>	Marco Strambini	Luke Skywalker	
<i>Sviluppo software</i>	Marco Strambini	Luke Skywalker	Leila Skywalker
<i>Testing</i>	Marco Strambini	Leila Skywalker	
<i>Documentazione</i>	Marco Strambini		

Project Plan

RAM (*Responsability Assigment Matrix*)

	Marco Strambini	Luke Skywalker	Leila Skywalker
<i>Progettazione entità db</i>	x	x	
<i>Creazione relazioni</i>		x	
<i>Classi di simulazione</i>	x		
<i>Realizzazione codice funzionante</i>	x	x	x
Costruzione layout web			x
Aggancio del layout con la logica applicativa		x	x
Avvio di differenti versioni dell'applicazione su ambienti differenti		x	
Correzione bug	x	x	x
<i>Documentazione</i>	x		

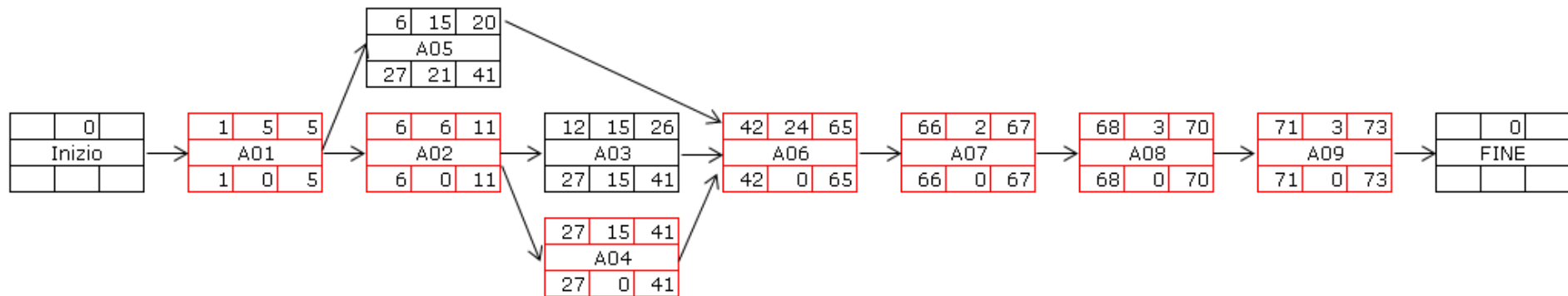
Project Plan

Reticolo di Progetto

[illegible]

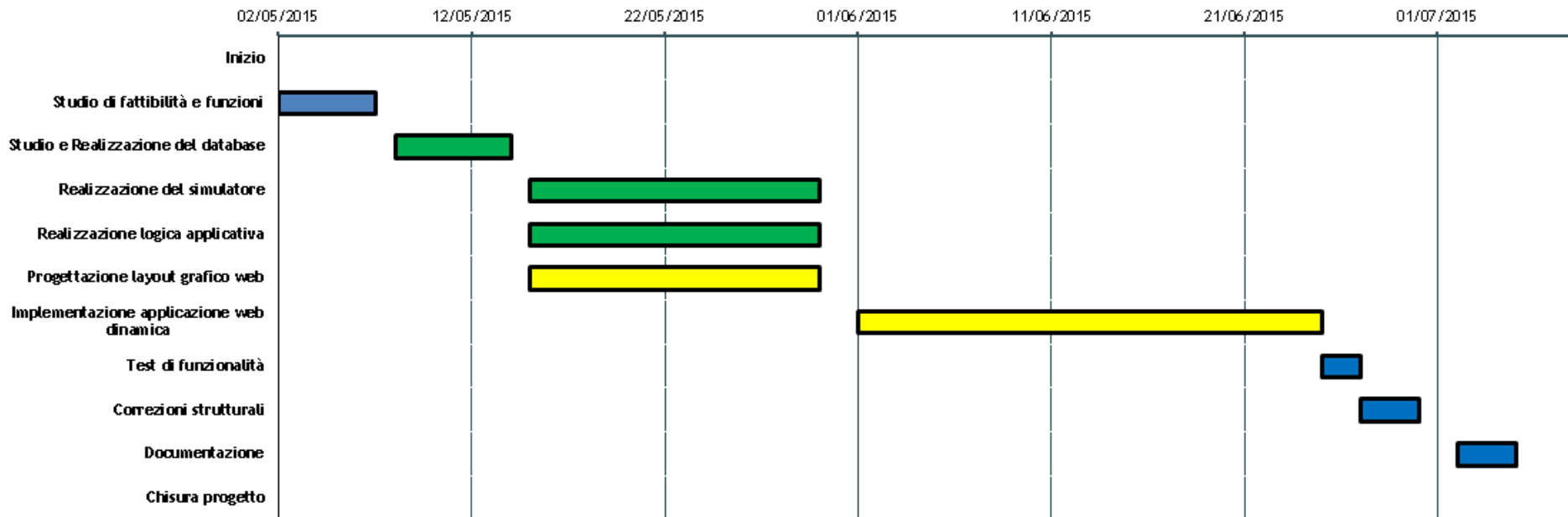
Project Plan

CPM (Critical Path Method)



Project Plan

Diagramma di Gantt



Infrastruttura di Progetto

Ambienti

- Workspace: gestione di package locali su notebook
- Repository: online tramite GitHub
- Test: notebook degli sviluppatori, tabelle gemelle di test
- Collaudo: notebook degli sviluppatori, pc terzi
- Produzione: server centrale online



Gestione dei Rischi

Rischio	Gravità	Descrizione
R01: Occupazione delle risorse umane su progetti terzi urgenti	Dannoso	E' possibile che eventi esterni impediscano lo sviluppo del progetto nei tempi previsti
R02: Crash dell'infrastruttura hardware	Dannoso	E' possibile che il comparto tecnico a disposizione possa subire danni
R03: Mancanza di conoscenze di sviluppo su alcune fasi del progetto	Medio	E' possibile che per alcune parti di sviluppo manchino delle conoscenze



Gestione dei Rischi

R01

Rischio	Occupazione delle risorse umane su progetti terzi urgenti
Gravità	Dannoso
Descrizione	E' possibile che eventi esterni impediscano lo sviluppo del progetto nei tempi previsti
Impatto	I componenti del team rallentano i propri sviluppi per ricoprire la parte mancante
Mitigazione	Aumentare la frequenza checkpoint e condivisione delle conoscenze
Contingency Plan	I componenti del team hanno conoscenze basilari su ogni aspetto del progetto



Gestione dei Rischi

R02

Rischio	Crash dell'infrastruttura hardware
Gravità	Dannoso
Descrizione	E' possibile che il comparto tecnico a disposizione possa subire danni
Impatto	Perdita del codice scritto e degli ambienti di test
Mitigazione	Ausilio di repository online per il codice, salvataggi frequenti online e su dischi esterni
Contingency Plan	Importare i progetti dal repository online o dai backup esterni.



Gestione dei Rischi

R03

Rischio	Mancanza di conoscenze di sviluppo su alcune fasi del progetto
Gravità	Medio
Descrizione	E' possibile che per alcune parti di sviluppo manchino delle conoscenze
Impatto	Rallentamento dello sviluppo per l'integrazione delle conoscenze ad hoc
Mitigazione	Dopo la definizione degli obiettivi nel modello a spirale, una risorsa si preoccupa di cercare le informazioni adeguate prima della fase di sviluppo interessata
Contingency Plan	Allocazione di una risorsa sullo studio delle soluzioni nel minor tempo possibile



Riuso

- Generatori di codice tramite funzionalità dell'IDE utilizzato
- Framework Bootstrap per la creazione dei layout web e caratteristiche responsive
- Applicazione COTs come dbms PostgreSQL, web server Tomcat
- Pagine jsp per la creazione di sezioni comuni come il menu
- Classi java già scritte in altri progetti precedenti
- Ausilio del livello di trasporto standard tcp/ip

Gli effetti del riuso comprendono notevoli agevolazioni nello sviluppo in termini di tempo, ergonomia e scelte strategiche di utilizzo delle classi.

In prossime release dell'applicazione di prevederà:

- Utilizzo di design pattern conosciuti.
- Ausilio di file di configurazione esterni per limitare le compilazioni



Diagramma dei package

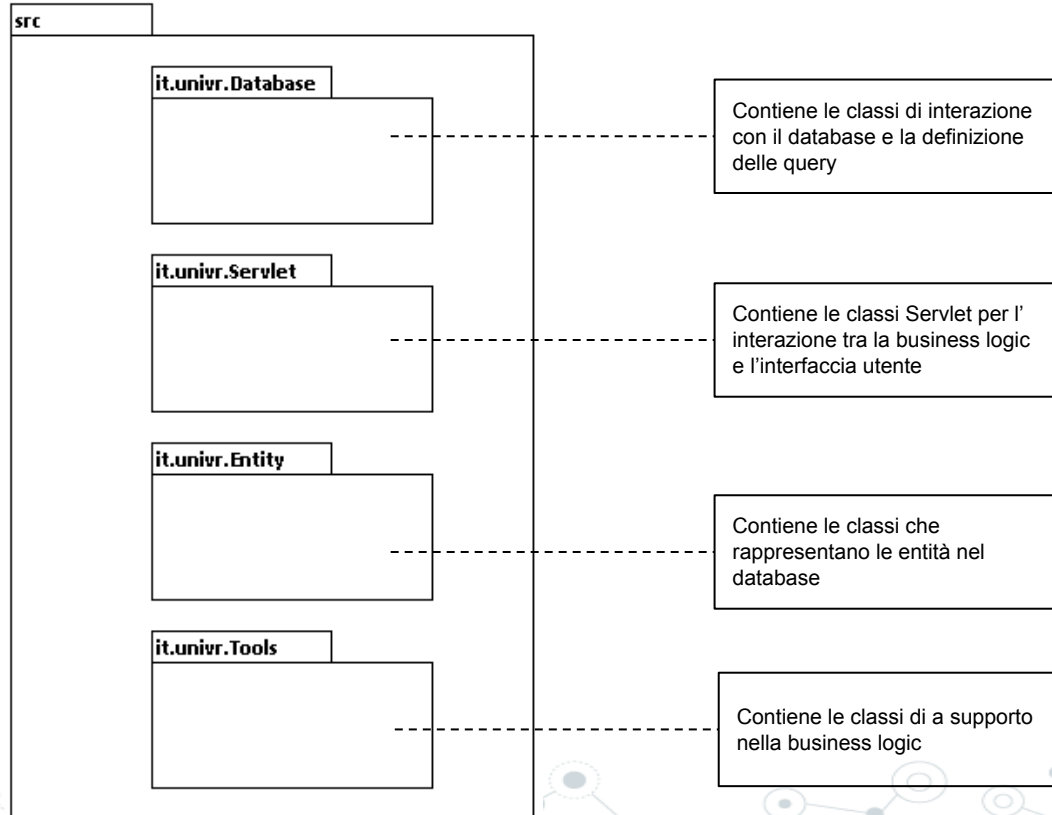


Diagramma delle classi

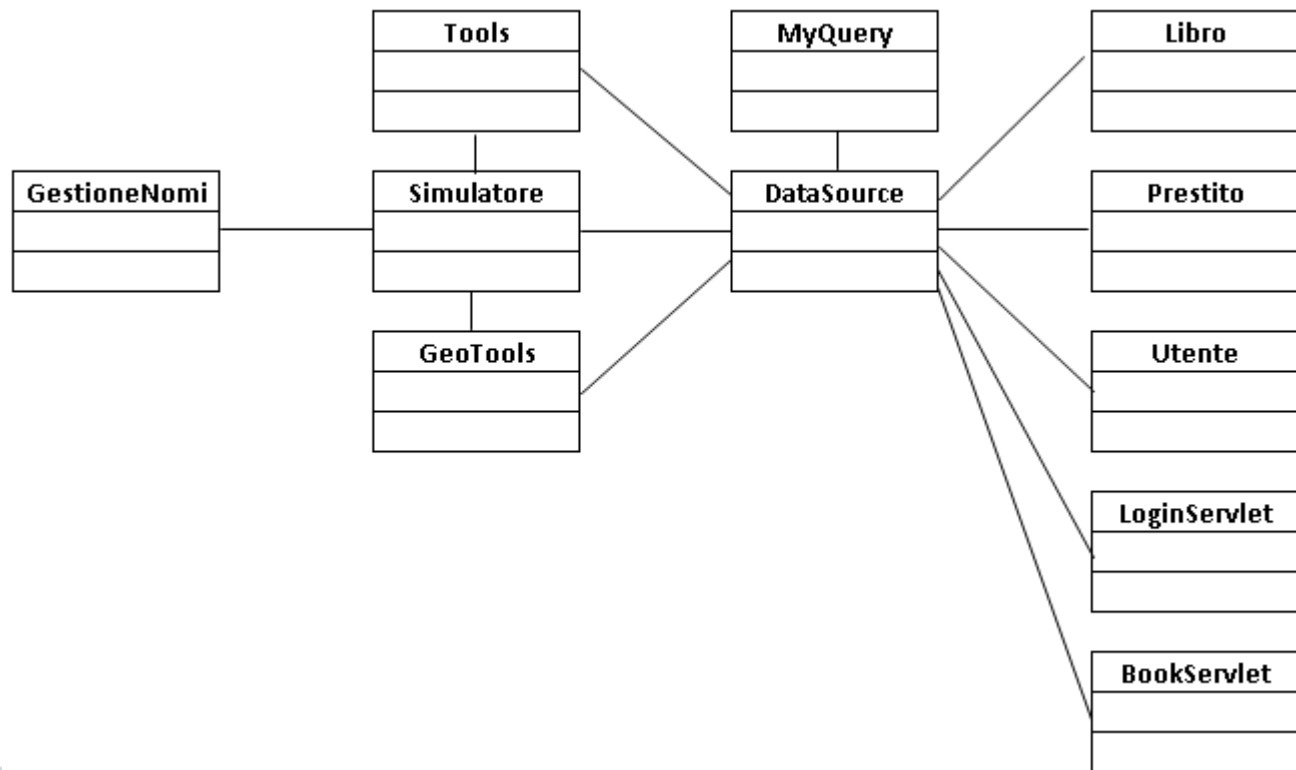
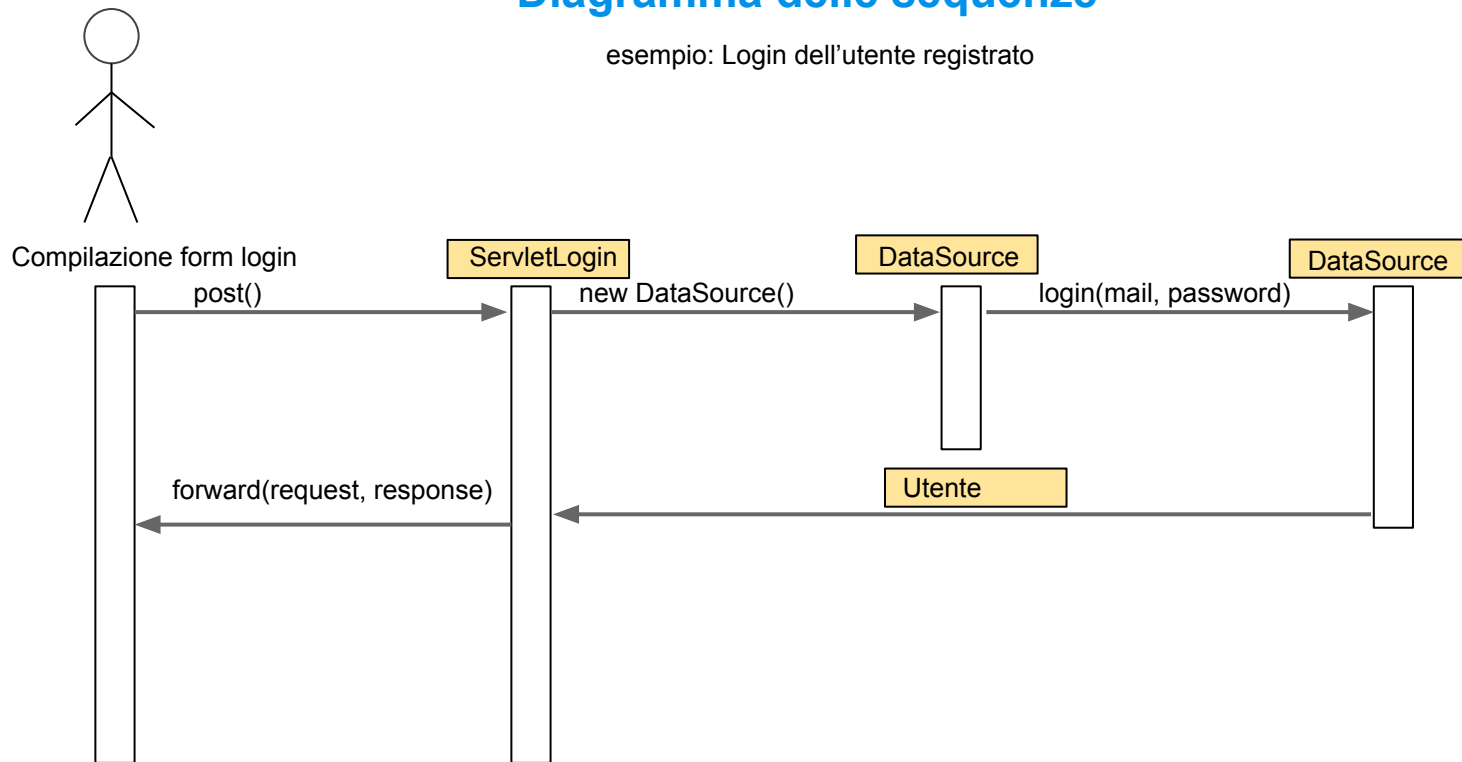


Diagramma delle sequenze

esempio: Login dell'utente registrato



Design Pattern

- Utilizzati bean maker
- ResultSet sostituiti da `ArrayList<T>` nelle pagine jsp



Principi SOLID

- **Single responsibility:** Una classe dovrebbe avere una sola ragione per cambiare
- **Open/close:** Le entità dovrebbe essere aperte per le estensioni, ma chiuse alle modifiche.
- **Liskov:** I sottotipi dovrebbero essere sostituibili per i supertipi; le classi figlie non devono mai rompere la definizione delle classi genitrici
- **Interface segregation:** I Client non dovrebbero essere costretti ad usare interfacce che non possono usare
- **Dependency inversion:** Un modulo ad alto livello non dovrebbe dipendere dai moduli a basso livello; entrambi dovrebbero dipendere dalle astrazioni. Le astrazioni non dovrebbe dipendere dai dettagli, sono i dettagli che devono dipendere dalle astrazioni

Il package `it.univr.Entity` comprende classi che non violano i principi di **Single responsibility** e **Open/Close**

Aggiornamenti futuri prevedono l'ausilio del principio di **Interface segregation** per eventualmente differenziare privilegi di accesso all'applicazione



Test

I test sono stati eseguiti secondo un approccio strutturale **“White Test”**.

Analizzati i cammini di codice che hanno riscontrato anomalie.

Unit Test: riscontrati errori poco significativi nelle fasi di interazione con db

Integration Test: riscontrati errori di passaggio variabili e prontamente risolti con gestione delle eccezioni

System Test: simulazioni di carico di lavoro e quantità di connessioni contemporanee al database tramite la classe Simulatore.





Conclusioni

Il progetto è stato implementato in tempi brevi, rilasciando una versione funzionante con caratteristiche in linea con le specifiche di startup, puntando sull'ergonomia e l'esperienza utente facilitata. Data la natura del progetto e il target a chi è rivolto, sarà interessante raccogliere feedback, da parte degli utenti utilizzatori, per permettere agli sviluppatori, l'integrazione di nuove features.

