

# La ruota della fortuna

Obiettivo del progetto *La ruota della fortuna (RdF)* è la realizzazione di una piattaforma di gioco ispirata al celebre programma televisivo.

## Le regole del gioco

Una partita de *La ruota della Fortuna* è strutturata in 5 manche di gioco, a cui partecipano 3 concorrenti. L'obiettivo per i concorrenti, in ogni manche, è di indovinare una frase misteriosa, su un tema comunicato. La frase è celata da un tabellone che inizialmente mostra una cella vuota per ogni lettera della frase (separando le parole con spazi), e che incrementalmente, in base alle chiamate effettuate dai giocatori, rivela le lettere che la compongono. Per ogni consonante individuata i concorrenti ricevono dei punti che si accumulano in un deposito relativo alla manche di gioco. Il concorrente che individua la frase misteriosa vince la manche e trasferisce i punti guadagnati nel deposito dell'intera partita. Al termine dell'ultima manche, il concorrente con il punteggio più alto nel deposito della partita viene proclamato vincitore. In ogni manche di gioco, i 3 concorrenti giocano a turno, sequenzialmente, effettuando in ogni turno di gioco una o più mosse. L'alternanza dei turni dipende dalla bravura dei giocatori nell'individuare le lettere che compongono la frase, e dalla fortuna. I concorrenti infatti si affidano ad un generatore di numeri casuale che regola l'andamento dei turni e i punti guadagnabili. Nella versione televisiva il generatore era realizzato da un dispositivo meccanico, denominato *ruota della fortuna*, un grosso cerchio suddiviso in settori colorati, vedi Figura 1. La ruota veniva fatta girare dai concorrenti, e l'esito dell'estrazione corrispondeva all'etichetta riportata nel settore in corrispondenza del concorrente una volta che la ruota aveva concluso la rotazione. La ruota può determinare: 1) la quantità di punti acquisiti per ogni consonante individuata dal concorrente (da 300 a 1000), 2) il passaggio del turno di gioco al concorrente successivo (*Passa*), 3) l'azzeramento del punteggio parziale del concorrente (*Perde*), e 4) un eventuale bonus per mantenere il turno di gioco (*Jolly*). Qui nel seguito, per analogia con il gioco televisivo, riferiremo l'azione di estrazione effettuata dal concorrente con l'espressione *gira la ruota*. Inoltre, denotiamo come *mosse*, le seguenti azioni: i) girare la ruota, ii) chiamare una consonante, iii) chiamare una vocale, iv) spendere un bonus (*Jolly*), e v) provare a dare la soluzione.



Figure 1 – La ruota della fortuna

In ogni manche, ogni giocatore, come prima mossa deve necessariamente girare la ruota. Il concorrente che acquisisce il primo turno di gioco nella prima manche è determinato casualmente. Il primo turno sarà assegnato a rotazione a tutti i concorrenti nelle manche successive. All'inizio di ogni manche viene comunicato, come indizio, il tema della frase misteriosa da indovinare. Il primo concorrente gira la ruota. Se la ruota indica *Passa*, il concorrente perde il turno di gioco, che passa al secondo concorrente, mentre se indica *Perde*, il concorrente, oltre a perdere il turno, perde tutti i punti acquisiti nella manche. Se la ruota indica *Jolly*, il concorrente riceve un bonus che potrà decidere di spendere in seguito per evitare di perdere il turno di gioco, e mantiene il turno. Infine, se la ruota riporta un punteggio numerico, il concorrente deve chiamare, entro 5 secondi, una consonante. Se la consonante chiamata non è presente nella frase, o se erroneamente viene chiamata una vocale, oppure se il concorrente non comunica la consonante in tempo, oppure se chiede di dare la soluzione, il turno di gioco passa al concorrente successivo. Mentre, se la consonante è presente, le caselle del tabellone che nascondono la lettera richiesta mostrano la consonante individuata e il concorrente acquisisce il punteggio indicato dalla ruota moltiplicato per il numero di occorrenze della consonante nella frase. Il punteggio acquisito si somma al punteggio parziale del concorrente per la manche in corso. A questo punto, lo stesso concorrente può effettuare una nuova mossa, scegliendo, entro 5 secondi, di:

1. girare nuovamente la ruota, riapplicando i criteri precedentemente illustrati
2. dare la soluzione, ossia comunicare, entro 10 secondi, la frase misteriosa. Se la frase comunicata corrisponde alla frase misteriosa il concorrente vince la manche acquisendo nel deposito della partita il punteggio parziale collezionato fino a quel momento, se invece la frase non è corretta, o la soluzione non è comunicata per tempo, il turno di gioco passa al concorrente successivo.
3. comprare una vocale, al costo di 1000 punti, che vengono sottratti dal deposito della manche del concorrente. Se il concorrente, nel deposito della manche, non dispone di un quantitativo di punti sufficiente all'acquisto, o se la vocale richiesta non è presente nella frase, o se erroneamente viene chiamata una vocale già mostrata dal tabellone oppure una consonante, oppure se la vocale non è chiamata in tempo, il concorrente cede il turno di gioco. Se invece la vocale è presente, le relative occorrenze vengono mostrate sul tabellone di gioco, e il concorrente può effettuare una nuova mossa.

Un concorrente in possesso del bonus (*Jolly*) che effettua una chiamata di consonante o di vocale errata, o che non effettua una mossa nei tempi stabiliti, o che estrae *Passa* può spendere il bonus acquisito per evitare di perdere il turno. Il bonus non è spendibile quando il concorrente estrae *Perde*.

Il bonus, una volta acquisito, può essere conservato ed usato nelle manche successive.

Infine, ai concorrenti viene comunicato quando tutte le consonanti che compongono la frase misteriosa sono state chiamate, in seguito alla notifica, per quella manche, possono provare a dare la soluzione o comprare vocali, ma non possono più girare la ruota, e nel caso lo facessero, perderebbero il turno di gioco.

## Requisiti

La piattaforma di gioco *Ruota della Fortuna (RdF)* dovrà essere utilizzata da utenti con ruoli distinti: 1) il *concorrente* di gioco, che organizza e partecipa a delle partite; 2) l'*osservatore*, che monitora l'andamento di una partita in corso osservando le mosse dei concorrenti; e 3) l'*amministratore*, che gestisce le frasi misteriose usabili per il gioco.

La piattaforma consiste di: 1) un modulo *serverRdF*, che interfacciandosi con un DBMS relazionale (PostgreSQL), fornisce servizi di back-end; 2) un modulo *playerRdF*, che fornisce servizi designati per concorrenti e osservatori; e 3) un modulo *adminRdF*, che fornisce servizi di gestione della piattaforma RdF per utenti amministratori. La realizzazione dei servizi offerti richiede una

opportuna suddivisione di funzionalità tra le componenti, e la definizione di protocolli di interazione tra queste parti. RdF dovrà essere realizzata in modo tale da supportare l'interazione in parallelo con più utenti connessi alla piattaforma da postazioni differenti. RdF dovrà essere realizzata in Java utilizzando tecnologie che permettano un'implementazione efficiente dei servizi.

## Avvio del server

All'invocazione di *serverRdF* deve essere richiesto di specificare: i) le credenziali per accedere a *dbRdF*, un database di supporto all'esecuzione dei servizi della piattaforma *RdF*, e ii) l'host del database. Se *dbRdF* non contiene un profilo di utente amministratore, *serverRdF* deve avviare un processo di registrazione, richiedendo all'utente amministratore che ha invocato l'esecuzione: il cognome, il nome, un nickname, l'email, e una password. Le credenziali registrate (email e pwd) potranno essere utilizzate per effettuare l'autenticazione dell'utente amministratore tramite *adminRdF* e *serverRdF*. Al termine della registrazione, o nel caso in cui in *dbRdF* sia già presente un profilo di amministratore, *serverRdF* rimane in attesa di richieste di connessione da parte di client *playerRdF* e *adminRdF*.

## Registrazione utenti

Un utente che intende accedere a *RdF* deve necessariamente registrarsi.

La registrazione di concorrenti e osservatori è gestita da *playerRdF*, mentre la registrazione di amministratori è gestita da *adminRdF*. Il processo di registrazione richiederà all'utente di fornire il nome, il cognome, un nickname, e un indirizzo di posta elettronica. L'indirizzo di posta elettronica e il nickname non devono essere già stati usati da altri utenti. Prima di completare la registrazione la piattaforma invierà all'utente un codice di attivazione da immettere per finalizzare la registrazione. Se l'utente non completa la registrazione entro 10 minuti dalla spedizione del codice il profilo dovrà essere cancellato.

## Autenticazione

L'accesso ad ogni funzionalità offerta dalla piattaforma tramite *playerRdF* ed *adminRdF* (ad eccezione del servizio di registrazione utenti e di ripristino password dimenticata) richiede l'autenticazione dell'utente.

Un utente effettua il login mediante *playerRdF* / *adminRdF*, specificando come identificativo l'indirizzo di posta e la password associata al proprio profilo.

Un utente autenticato mediante *playerRdF*, può:

1. organizzare una nuova partita,
2. visualizzare le partite organizzate per la quale non si sono ancora chiuse le iscrizioni dei concorrenti,
3. visualizzare le partite in corso di svolgimento,
4. richiedere la partecipazione, come concorrente, ad una partita organizzata da altri utenti per la quale non sono ancora state chiuse le iscrizioni (dei concorrenti)
5. richiedere la partecipazione, come osservatore, ad una partita per la quale non sono ancora chiuse le iscrizioni (dei concorrenti), o che è in corso di svolgimento
6. abbandonare, da concorrente, o da osservatore, una partita per la quale non sono ancora state chiuse le iscrizioni, oppure che è in corso di svolgimento
7. modificare i dati del proprio profilo
8. analizzare le statistiche di utilizzo della piattaforma di gioco

Un utente autenticato mediante *adminRdF*, può:

1. visualizzare le partite organizzate per la quale non si sono ancora completate le iscrizioni dei concorrenti,
2. visualizzare le partite in corso di svolgimento,
3. richiedere di partecipare, come osservatore, ad una partita in corso di svolgimento, o per la quale non sono ancora completate le iscrizioni dei concorrenti

4. abbandonare, da osservatore, una partita per la quale sono ancora state chiuse le iscrizioni, oppure che è in corso di svolgimento
5. modificare i dati del proprio profilo,
6. gestire il catalogo di frasi misteriose utilizzabili nelle manche di gioco
7. analizzare statistiche di utilizzo della piattaforma di gioco.

## **Organizzare una nuova partita**

Un utente autenticato può organizzare una nuova partita. Una partita, strutturata in 5 manche di gioco, ha associato un nome, una data, e un'ora di creazione, un elenco dei concorrenti e di osservatori.

L'utente che organizza la partita viene automaticamente iscritto come concorrente alla partita organizzata.

## **Visualizzare la lista delle partite**

I client *playerRdF* e *adminRdF* permettono agli utenti autenticati di visualizzare l'elenco delle partite organizzate con iscrizioni dei concorrenti ancora aperte, e l'elenco delle partite già in corso di svolgimento.

## **Richiesta di partecipazione ad una partita da concorrente**

Un utente autenticato, mediante *playerRdF* può richiedere di partecipare, come *concorrente*, ad una partita organizzata con iscrizioni ancora aperte.

*RdF* autorizza la partecipazione se il numero di concorrenti già iscritti è inferiore a 3. Al raggiungimento di 3 iscrizioni *RdF* chiude le iscrizioni, e verifica che nel catalogo di frasi misteriose gestite dalla piattaforma, siano presenti almeno 5 frasi che non siano state usate in manche a cui i 3 concorrenti hanno partecipato come osservatori o come concorrenti. Se tale condizione è verificata, *RdF* seleziona 5 frasi che soddisfano questo vincolo da usare come frasi misteriose nelle manche della partita in corso, e notifica l'inizio imminente del gioco ai concorrenti e agli osservatori registrati mediante i rispettivi moduli *playerRdF*. Se invece il vincolo non è soddisfatto, *RdF* annulla la partita e tramite *playerRdF* avvisa i concorrenti e gli osservatori registrati dell'impossibilità di attivare la partita per la mancanza di frasi non note ai concorrenti, inoltre, *RdF* manda una mail agli amministratori di gioco registrati, informandoli della necessità di caricare nuove frasi. La partita annullata è rimossa dall'elenco delle partite organizzate.

## **Richiesta di partecipazione ad una partita da osservatore**

Tramite *playerRdF* e *adminRdF* un utente autenticato può richiedere di partecipare, come *osservatore*, ad una partita organizzata con iscrizioni dei concorrenti ancora aperte, oppure già in corso di svolgimento. L'osservatore è aggiunto alla lista degli utenti che monitorano l'andamento della partita. Non sono previsti limiti al numero di osservatori che partecipano ad una partita.

## **Partecipazione ad una partita da osservatore**

Un utente autenticato, registrato come *osservatore* ad una partita, deve poter monitorare la partita mediante *playerRdF/adminRdF*. Se l'iscrizione non è ancora chiusa, *playerRdF/adminRdF* deve mettere in attesa dell'inizio del gioco l'osservatore. Durante il corso del gioco *playerRdF/adminRdF* deve visualizzare: l'identificativo della partita e della manche in corso, il tabellone che nasconde la frase misteriosa, il punteggio nel deposito della manche e della partita di ogni concorrente, l'eventuale possesso di un bonus. Deve essere inoltre mostrato il concorrente che mantiene il turno di gioco, e le mosse da lui compiute.

1. Se il concorrente gira la ruota dovrà essere mostrato l'esito della rotazione, e gli eventuali effetti sul suo deposito di punti della manche.

2. Se chiama una consonante o una vocale, dovrà essere mostrata la lettera chiamata e gli effetti della mossa sul tabellone e sul suo deposito di punti della manche.
3. Se il concorrente prova a dare la soluzione, dovrà essere mostrata la soluzione proposta, e la sua eventuale correttezza. Se la soluzione è corretta dovrà essere mostrata la frase sul tabellone, dovrà essere evidenziato che il concorrente risulta vincitore della manche, trasferendo i punti acquisiti durante manche nel deposito della partita, e azzerando al contempo i depositi di punti della manche di tutti i concorrenti. Se la soluzione non è corretta dovrà essere evidenziato che il concorrente ha commesso un errore
4. Se il concorrente spende il proprio bonus, dovrà essere mostrata la scelta operata, rimuovendo il bonus spendibile dalla rappresentazione delle informazioni riguardanti il concorrente.

Infine, poiché le mosse hanno vincoli temporali, *playerRdF/adminRdF* dovrà mostrare il count down dei secondi rimanenti al concorrente per effettuare ogni mossa.

## **Partecipazione ad una partita da concorrente**

*playerRdF* deve permettere a un utente autenticato, iscritto come concorrente ad una partita, di giocare la partita a cui è iscritto. Il concorrente deve essere messo in attesa dell'inizio della partita. Una volta iniziata la partita, *playerRdF* deve mostrare al concorrente le stesse informazioni che sono mostrate agli osservatori (vedi il caso d'uso precedente), e permettere al concorrente, durante i suoi turni di gioco, ed entro i vincoli di tempo specificati dal regolamento, di girare la ruota, chiamare una consonante, chiamare una vocale, giocare un jolly, e provare a dare la soluzione della frase misteriosa.

## **Abbandono di una partita**

*playerRdF* deve permettere ai concorrenti di una partita di abbandonare la partita. Se un concorrente abbandona la partita in corso di svolgimento, la partita viene annullata, tutti i concorrenti ed osservatori iscritti ricevono notifica dell'annullamento a causa dell'abbandono del concorrente, e la partita viene rimossa dalla lista delle partite organizzate. Se invece l'abbandono avviene prima della chiusura delle iscrizioni, il giocatore viene rimosso dalla lista dei concorrenti iscritti. Se in seguito all'abbandono la lista dei concorrenti iscritti è vuota, la partita viene annullata, gli osservatori eventualmente iscritti ricevono comunicazione dell'annullamento, e la partita viene rimossa dalla lista delle partite organizzate.

*playerRdF* e *adminRdF* devono anche permettere agli osservatori di una partita con iscrizioni aperte o in corso di svolgimento di abbandonare la partita. L'abbandono degli osservatori non ha conseguenze sulla gestione della partita. La partita può essere giocata anche se la lista degli osservatori è vuota.

*RdF* deve tener traccia della partecipazione di concorrenti ad ogni manche, comprese quelle iniziate ma non interamente giocate, e dei vincitori di ogni manche interamente giocata.

## **Modifica dei dati di profilo**

Un utente, mediante *playerRdF / adminRdF* deve poter modificare i dati del proprio profilo. In particolare, l'utente potrà correggere nome e cognome, e cambiare nickname, e password. L'impostazione di una nuova password è soggetta all'inserimento della vecchia password. *RdF* dovrà inviare una mail all'indirizzo di posta registrato informandolo del cambio avvenuto (senza mostrare le password).

## **Reset di password**

*playerRdF* e *adminRdF* devono fornire un servizio per reimpostare una password dimenticata, accessibile anche da utenti non autenticati. Il servizio deve richiedere all'utente di specificare

l'email associata al profilo, e dovrà rimpiazzare la password del profilo con una nuova generata automaticamente, inoltrandola all'indirizzo di posta elettronica associato al profilo.

## Gestione delle frasi misteriose

L'utente amministratore (autenticato), mediante *adminRdF*, può aggiungere / rimuovere e modificare le frasi misteriose da indovinare, e, per ogni frase, il tema associato. Una frase è una stringa alfanumerica con lunghezza fino a 60 caratteri, mentre il tema è una stringa alfanumerica di lunghezza fino a 60 caratteri, che rappresenta un indizio per indovinare la frase misteriosa. Per fini di praticità *adminRdF* dovrà permettere di importare le frasi da file testuali in formato csv.

## Monitoraggio

*playerRdF* e *adminRdF* devono offrire servizi di analisi per i rispettivi utenti. In particolare:

1. Mostrare il concorrente
  - che detiene il primato di punteggio raggiunto per manche e per partita
  - che ha giocato più manche in assoluto
  - con la media più alta di punti acquisiti per manche
  - che ha dovuto cedere più volte il turno di gioco in seguito ad errori
  - che ha perso tutto in seguito ad un giro di ruota per il maggior numero di volte
2. Mostrare la chiamata di consonante, riferita ad una manche già giocata, che ha portato all'acquisizione della maggior quantità punti, la frase misteriosa associata, e l'utente che ha fatto la mossa.
3. Mostrare il numero medio mosse per manche con le quali viene indovinata una frase.
4. Mostrare le statistiche di gioco relative ad un utente:
  - il numero di manche e di partite giocate,
  - il numero di manche e di partite osservate,
  - il numero di manche e partite vinte,
  - il punteggio medio vinto per partita
  - Il numero medio di volte, per manche e partita, che ha dovuto cedere il turno di gioco
  - Il numero medio di volte, per manche e partita, che ha perso tutto

## Requisiti dati

*serverRdF* fornisce servizi di supporto alla memorizzazione ed analisi dei dati dell'intera piattaforma di gioco. E' richiesto di progettare la base di dati che *serverRdF* utilizzerà per gestire informazioni relative: ai concorrenti, agli osservatori, agli amministratori, alle manche e alle partite giocate, e alle frasi misteriose. In particolare, sono di interesse le seguenti informazioni:

1. il profilo dei concorrenti, degli osservatori e degli amministratori regolarmente registrati, caratterizzato dal cognome, il nome, un nickname, un indirizzo email, e una password.

2. Lo storico delle manche di gioco e delle partite che sono state giocate e che si stanno giocando, tenendo traccia di tutte le mosse fatte dai concorrenti, e dei punti acquisiti e persi con ogni mossa.
3. La collezione di frasi misteriose e dei relativi temi. Per ogni manche di gioco, RdF potrà selezionare solo frasi misteriose che non sono state già usate in manche in cui hanno giocato i concorrenti coinvolti nella manche considerata, o a cui hanno assistito come osservatori.

## Svolgimento del progetto

L'obiettivo del progetto consiste nella 1) progettazione e 2) sviluppo di un sistema software e della relativa base di dati che soddisfi i requisiti proposti, 3) utilizzando un processo di sviluppo strutturato, e 4) nella produzione di artefatti (quali, diagrammi UML e schemi ER) che descrivano quanto realizzato nelle varie fasi dello sviluppo, in conformità alle metodologie seguite a lezione. Le attività di analisi e progettazione devono essere adeguatamente documentate facendo uso del linguaggio UML per l'applicazione software e del modello Entity-Relationship (ER) per il database. È richiesto di progettare l'applicazione avvalendosi dove possibile dell'uso di design patterns, e di realizzare l'applicazione con un'opportuna interfaccia grafica, usando il linguaggio di programmazione Java, e progettare e realizzare un database utilizzando PostgreSQL per la sua implementazione (<http://www.postgresql.org>) e JDBC per l'accesso alla base di dati da programma Java (<http://jdbc.postgresql.org/download.html>)

È opportuno evidenziare che i servizi dell'applicazione vengono erogati in parallelo a più utenti, e che possono verificarsi accessi concorrenti a risorse condivise. L'applicazione deve quindi essere definita in modo tale da gestire opportunamente l'accesso concorrente alle risorse accedute.

## Realizzazione del Database

La realizzazione del database prevede svariate fasi. In seguito si riportano le attività ed i relativi artefatti che dovranno essere prodotti nelle varie fasi.

- Si ristrutturi, se necessario, secondo le metodologie di progettazione i requisiti descritti. Si scelgano le metodologie per la costruzione dello schema ER, motivando le scelte fatte.
- Si definisca lo schema concettuale ER per il database, evidenziando le entità e le associazioni di interesse, nonché i vincoli di cardinalità e di identificazione, motivando le scelte effettuate. Altri eventuali vincoli devono essere espressi in linguaggio naturale.
- Si effettui la ristrutturazione dello schema ER motivando le scelte effettuate.

*E' richiesto di produrre un documento di analisi dei requisiti ristrutturato e documentazione associata allo schema ER (ristrutturato e non), con eventuale specifica di vincoli in linguaggio naturale.*

- Si effettui la traduzione dello schema ER ristrutturato in un equivalente schema relazionale.

*E' richiesto di produrre la documentazione associata allo schema relazionale derivato dallo schema concettuale.*

- Si realizzi il database utilizzando PostgreSQL, e SQL per la definizione dei dati, l'implementazione dei vincoli identificati, e la manipolazione dei dati, secondo le operazioni previste dall'applicazione.

*Documentare gli script SQL necessari alla creazione della base di dati e dei vincoli definiti sui dati e le query SQL a supporto dei servizi erogati da RdF.*

## Codifica

Il codice sorgente dell'applicazione dovrà essere definito in modo da facilitarne la comprensione e la manutenibilità. Per favorire questo obiettivo vengono in seguito riportate alcune linee guida:

- Utilizzare la lingua inglese per etichettare classi, metodi, attributi e variabili nel codice sorgente java. I commenti possono essere espressi sia in inglese che in italiano.
- Ogni volta che si identificano due estratti di codice sufficientemente simili, il codice deve essere reingegnerizzato, e il codice duplicato eliminato portando a fattore comune le funzionalità condivise. Si consiglia l'utilizzo di strumenti di refactoring forniti da ambienti di sviluppo quali Eclipse / NetBeans.
- E' opportuno notare che esistono delle "regole di stile" per ogni linguaggio di programmazione, quindi per indentazione, nomi, parentesi, commenti, etc.

## Documentazione del codice

Generare documentazione javadoc documentando i metodi e gli attributi con visibilità *public* o *protected* di ogni classe con visibilità *public*. La documentazione generata deve spiegare come usare una determinata classe o interfaccia, non come questa sia stata effettivamente implementata, se questo può essere tenuto nascosto.

Prima di commentare il codice perché altrimenti incomprensibile, tentare le seguenti strade:

- Se si sono usati nomi poco (o del tutto non) chiari, sostituirli con nomi che spieghino lo scopo dell'entità (variabile locale, parametro, attributo, metodo, classe, package...)
- Se il corpo di un metodo è troppo grande, scomporre il metodo in più metodi (eventualmente con visibilità private)
- Se i metodi o gli attributi di una classe sono troppo numerosi, scomporre la classe in classi con meno responsabilità (eventualmente con visibilità package)
- Se la classe svolge uno specifico ruolo di un pattern, specificare il ruolo, il pattern e gli altri partecipanti (quali altre classi e che ruolo hanno); non descrivere il funzionamento del pattern.
- Se il codice rimane comunque di difficile comprensione, inserire commenti in linguaggio naturale.

## Nota implementativa

Come descritto, l'interazione tra RdF e l'utente finale può avvenire anche per mezzo di email composte e inoltrate automaticamente. La realizzazione di tale funzionalità richiede l'utilizzo di servizi di invio di messaggi di posta elettronica forniti da un server SMTP. JavaMail (<https://java.net/projects/javamail>), è una libreria che supporta l'interfacciamento di applicazioni Java con vari servizi di gestione della posta elettronica, tra cui quelli forniti da server SMTP. E' pertanto richiesto di utilizzare JavaMail (<https://maven.java.net/content/repositories/releases/com/sun/mail/javax.mail/1.5.6/javax.mail-1.5.6.jar>) per supportare l'interazione tra RdF e un server SMTP. In allegato alla presente documentazione viene fornito il codice sorgente di una classe che effettua la spedizione di una mail usando il server SMTP usato in ateneo (smtp.office365.com). Per implementare la spedizione di email è suggerito di ispirarsi a tale codice, riutilizzandolo, ed eventualmente modificandolo. Il server SMTP considerato consente l'invio di email esclusivamente ad utenti autenticati. Le credenziali di autenticazione sono le stesse usate per il login al servizio di web mail dell'università (ad es. username: nome.cognome@studenti.uninsubria.it pwd: mypassword). La spedizione è autorizzata solo se le credenziali sono valide e il mittente del messaggio corrisponde allo username.

## Suddivisione del lavoro

Si richiede di indicare:

- come è stato suddiviso il prodotto software. Ad es. il programma è stato diviso in "moduli": GUI, gestione/elaborazione dati, etc.
- come è stato organizzato il processo di sviluppo (ad es. in quali fasi è stato suddiviso): analisi del problema, modellazione UML, implementazione, progettazione DB, realizzazione DB, e stesura della documentazione.



- A chi sono state allocate le diverse attività rispetto alle varie parti del prodotto. Ad es. l'analisi è stata fatta da X, la progettazione UML da X e Y in coppia, l'implementazione della GUI da X, l'implementazione della gestione dati da Y, etc.

## Valutazione del progetto

È auspicabile che il progetto possa essere un'occasione formativa di lavoro in team. Gli studenti sono pertanto invitati a svolgere il progetto in gruppi di lavoro composti da 3 / 4 studenti.

La valutazione del progetto avviene singolarmente per ciascuno studente. Il giorno dell'appello verrà comunicato tramite avviso sulla pagina del corso nella piattaforma di elearning. Durante la valutazione è richiesto allo studente di saper argomentare in modo opportuno le scelte progettuali, algoritmiche, e implementative adottate. In fase di discussione orale verrà verificata l'effettiva padronanza delle tecniche utilizzate attraverso una serie di domande.

La valutazione terrà conto dei seguenti fattori: l'aderenza del sistema realizzato ai requisiti proposti, i documenti di analisi e progettazione prodotti sia per la realizzazione del software che per il database (correttezza sintattica, semantica, completezza e leggibilità, minimalità dello schema logico), le scelte algoritmiche e di progettazione effettuate (design pattern), la qualità del codice sorgente prodotto (funzionalità, correttezza, facilità d'uso).

## Consegna

Il progetto deve essere consegnato entro il giorno stabilito per ogni appello d'esame, utilizzando l'apposita form disponibile sulle pagine del Laboratorio Interdisciplinare B della piattaforma di e-learning. Alla voce commenti, specificare il nome e le matricole degli studenti del gruppo coinvolto. Il progetto deve essere allegato in formato compresso ZIP.

I progetti devono essere corredati da:

1. documentazione
2. codice sorgente · file di build per Apache Ant (<http://ant.apache.org/>) / Maven (<https://maven.apache.org/>) per compilare il progetto, lanciare il server e i client, creare il database, creare la documentazione javadoc, etc.
3. eventuali librerie necessarie alla compilazione e/o all'esecuzione
4. file README con indicazioni sull'installazione e sulla compilazione, specificando i comandi Ant/Maven da utilizzare, ed indicazioni di particolari librerie, usate in modo non standard.

Il progetto deve compilare correttamente (tramite il file di build), una volta espanso in una directory.

I progetti con errori di compilazione e/o di esecuzione non verranno valutati.

Una volta effettuata la consegna, agli studenti verrà comunicata una data in cui potranno sostenere la discussione orale del progetto e la valutazione del lavoro svolto.

## Domande sul Progetto

Per qualsiasi dubbio durante lo svolgimento del progetto: [pietro.colombo@uninsubria.it](mailto:pietro.colombo@uninsubria.it), l'oggetto della mail deve essere: "Laboratorio B: Domande"

Progetto valido a partire dall'appello di Giugno 2019