

Raccolta ed analisi dei requisiti

[Task: [link](#)]

Analisi Del problema

Il progetto che ci viene richiesto di sviluppare si può considerare come un'applicazione di gestione del ciclo di vita dei progetti di ricerca destinati a essere presentati all'Unione Europea per il finanziamento. Si tratta di un sistema di valutazione interna che permette ai ricercatori di creare e sottoporre i loro progetti, comprensivi di documenti specifici in formato PDF, a una serie di valutazioni periodiche. Ogni progetto attraversa vari stati nel suo ciclo di vita e la piattaforma deve essere in grado di gestire e registrare tutte queste transizioni. La piattaforma deve inoltre fornire un sistema di comunicazione tra ricercatori e valutatori, conservando la storia delle versioni dei documenti e delle valutazioni.

Obiettivi del Progetto

- **Funzionalità di Creazione e Sottomissione del Progetto:** Fornire un sistema che permetta ai ricercatori di creare e sottoporre progetti per la valutazione. Questo dovrebbe includere la possibilità di caricare uno o più documenti in formato PDF relativi al progetto.
- **Gestione dello Stato del Progetto:** Implementare un sistema che gestisca i vari stati di un progetto: "Approvato", "Sottomesso per valutazione", "Richiede modifiche", "Non approvato". La piattaforma deve consentire ai valutatori di aggiornare lo stato del progetto a seguito della loro valutazione.
- **Implementazione del Sistema di Valutazione:** I valutatori devono essere in grado di accedere ai progetti, scaricare i documenti da valutare, creare un report di valutazione e aggiungere commenti direttamente sui documenti PDF.
- **Sistema di Messaggistica:** Implementare un sistema di messaggistica che permetta ai ricercatori e ai valutatori di comunicare tra loro. Questo dovrebbe includere la possibilità per i ricercatori di chiedere chiarimenti sui report di valutazione e per i valutatori di rispondere in maniera anonima.
- **Storico delle Versioni:** Mantenere uno storico delle diverse versioni di ogni documento del progetto e dei relativi report di valutazione. Questo storico deve essere accessibile sia ai ricercatori che ai valutatori.
- **Sicurezza e Privacy:** Assicurarci che tutte le interazioni sulla piattaforma siano sicure e che i dati sensibili siano protetti. Questo include garantire l'anonimato dei valutatori durante le interazioni con i ricercatori.

Tecnologie individuate per lo sviluppo

- **Database:** PostgreSQL adatto per gestire i dati strutturati del progetto. Potrebbe essere necessario un database NoSQL come MongoDB per gestire i dati semi-strutturati o non strutturati, come i messaggi di testo tra valutatori e ricercatori.
- **Cloud Storage:** Google Cloud Storage, Dropbox, Google Drive o Amazon S3 possono essere utilizzati per memorizzare i documenti in formato PDF.
- **Back-end Development:** Node.js o Django
- **Front-end Development:** React.js o Angular, Bootstrap, W3Css, Alla maniera di andre.
- **PDF Manipulation Libraries:** Librerie come PDF.js o PyPDF2 possono essere utilizzate per gestire l'interazione con i documenti PDF, come l'aggiunta di commenti.
- **Data Security and Privacy:** SSL/TLS per la crittografia dei dati in transito, e AES per la crittografia dei dati a riposo. Inoltre, per garantire l'anonimato, potrebbero essere implementate soluzioni di pseudonimizzazione.

- **Version Control System:** Git
- **Hosting:** Heroku (va con Flask e Postgres ma a limite di storage)
- **[PLUS] - Testing:** Jest o Mocha per i test unitari ad esempio, potrebbe essere creato un test per verificare che il codice che gestisce l'upload dei documenti PDF funzioni correttamente. OWASP ZAP per testing sulla sicurezza che può essere utilizzato per eseguire test di penetration testing e verificare la robustezza del sistema contro attacchi informatici.
- **[PLUS] - Data Warehouse/Business Intelligence:** (se vogliamo) potrebbe essere utile uno strumento come Google BigQuery o Amazon Redshift per l'analisi dei dati. Ad esempio, potrebbe essere necessario analizzare i progetti di ricerca presentati in un determinato periodo, identificare i documenti più frequentemente richiesti dai valutatori per valutazione o scoprire il motivo più comune per cui un progetto non viene approvato. potrebbe anche essere utilizzato per l'analisi predittiva, come la previsione della probabilità di approvazione di un progetto sulla base delle sue caratteristiche e delle valutazioni precedenti.
- **[PLUS] - Messaging/Notification Service:** Un servizio come RabbitMQ o Google Cloud Pub/Sub può essere utilizzato per implementare il sistema di messaggistica e notifiche.

Specifiche delle operazioni

Tabelle principali identificate a prima lettura:

- **Ricercatori:** Conserva informazioni sui ricercatori come ID_ricercatore, nome, email, ecc.
- **Valutatori:** Conserva informazioni sui valutatori come ID_valutatore, nome, ecc.
- **Progetti:** Conserva informazioni sui progetti come ID_progetto, ID_ricercatore, stato, descrizione, ecc.
- **Documenti:** Conserva informazioni sui documenti come ID_documento, ID_progetto, tipo (data management plan, ethics deliverable, ecc.), percorso del file (in Google Cloud Storage, Dropbox, ecc.)
- **Versioni:** Conserva informazioni sulle versioni dei documenti come ID_versione, ID_documento, data_creazione, commenti, ecc.
- **Report_valutazione:** Conserva informazioni sui report di valutazione come ID_report, ID_valutatore, ID_progetto, commenti, data_creazione, ecc.
- **Messaggi:** Conserva informazioni sui messaggi inviati tra ricercatori e valutatori come ID_messaggio, ID_ricercatore, ID_valutatore, testo, data_invio, ecc. oppure usare MongoDB

Alcune delle operazioni comuni potrebbero includere:

- Inserire un nuovo progetto: Questa operazione potrebbe essere implementata come una procedura in PostgreSQL.

```

1 CREATE OR REPLACE PROCEDURE inserisci_progetto(_ID_ricercatore INT, _descrizione TEXT,
2   _stato TEXT)
3 AS $$
4 BEGIN
5   INSERT INTO Progetti (ID_ricercatore, descrizione, stato)
6   VALUES (_ID_ricercatore, _descrizione, _stato);
7 END;
8 $$ LANGUAGE plpgsql;
```

- Cambiare lo stato di un progetto: Questa operazione potrebbe essere implementata come una funzione in PostgreSQL.

```

1 CREATE OR REPLACE FUNCTION cambia_stato_progetto(_ID_progetto INT, _stato TEXT) RETURNS
  VOID AS $$
2 BEGIN
3   UPDATE Progetti
4   SET stato = _stato
5   WHERE ID_progetto = _ID_progetto;
6 END;
7 $$ LANGUAGE plpgsql;

```

- Ottenere i progetti da valutare: Questa operazione potrebbe essere implementata come una query.

```

1
2 SELECT * FROM Progetti WHERE stato = 'sottomesso per valutazione';

```

- Inserire un nuovo report di valutazione: Questa operazione potrebbe essere implementata come una procedura.

```

1
2 CREATE OR REPLACE PROCEDURE inserisci_report(_ID_valutatore INT, _ID_progetto INT,
  _commenti TEXT)
3 AS $$
4 BEGIN
5   INSERT INTO Report_valutazione (ID_valutatore, ID_progetto, commenti)
6   VALUES (_ID_valutatore, _ID_progetto, _commenti);
7 END;
8 $$ LANGUAGE plpgsql;

```

- Ottenere i report di valutazione per un determinato progetto: Questa operazione potrebbe essere implementata come una query.

```

1
2 SELECT * FROM Report_valutazione WHERE ID

```

Autenticazione ed autorizzazione al DB

L'ereditarietà o la delega dei permessi dovrebbe essere presa tenendo in considerazione il principio del minimo privilegio, che sostiene che a un utente o a un ruolo dovrebbe essere concesso solo il minimo livello di permessi necessario per svolgere le proprie funzioni. Considerando le tabelle definite precedentemente, possiamo creare i seguenti ruoli:

- Ricercatore: Può creare e modificare i propri progetti e documenti, può visualizzare i propri progetti e documenti, può visualizzare i report di valutazione dei propri progetti, e può inviare messaggi.
- Valutatore: Può visualizzare i progetti sottomessi per la valutazione, può creare report di valutazione, può cambiare lo stato dei progetti, può visualizzare e commentare i documenti dei progetti, e può inviare messaggi.
- Amministratore: Ha accesso a tutte le tabelle e può eseguire tutte le operazioni.

Nel database PostgreSQL, possiamo creare questi ruoli e assegnare i permessi appropriati utilizzando comandi come i seguenti:

```

1 CREATE ROLE ricercatore;
2 GRANT SELECT, INSERT, UPDATE ON Progetti, Documenti, Messaggi TO ricercatore;
3 GRANT SELECT ON Report_valutazione TO ricercatore WHERE ID_ricercatore = USER;
4
5 CREATE ROLE valutatore;
6 GRANT SELECT, INSERT, UPDATE ON Progetti, Documenti, Messaggi, Report_valutazione TO
  valutatore;
7
8 CREATE ROLE amministratore;
9 GRANT ALL PRIVILEGES ON DATABASE mio_database TO amministratore;

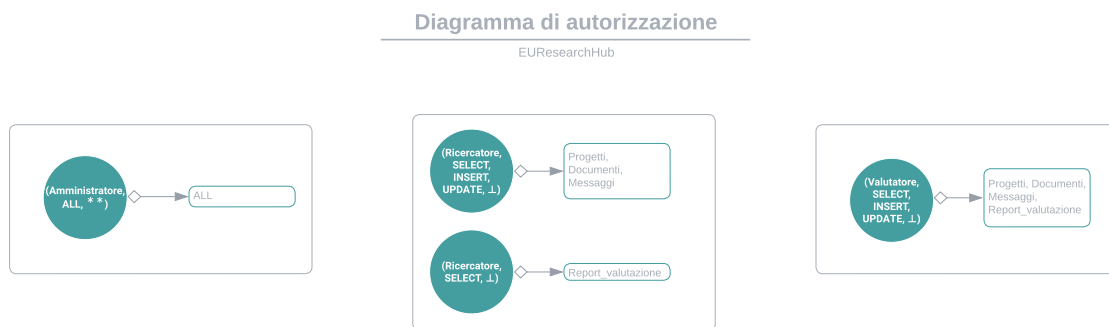
```

Si noti che per la tabella Report_valutazione, abbiamo concesso al ricercatore solo il permesso di SELECT sui report di valutazione che appartengono a loro stessi. Questo è un esempio di politica di confidenzialità: i ricercatori non dovrebbero essere in grado di vedere i report di valutazione di altri ricercatori.

Riguardo all'integrità dei dati, utilizziamo il controllo delle autorizzazioni per garantire che solo i ruoli appropriati possano modificare specifici dati. Ad esempio, solo i valutatori possono modificare lo stato di un progetto. Per assegnare i ruoli agli utenti, possiamo utilizzare il comando GRANT:

```
1 GRANT ricercatore TO utente1;
2 GRANT valutatore TO utente2;
3 GRANT amministratore TO utente3;
```

Diagramma di autorizzazione



Nel contesto descritto, non sembra necessario inserire ereditarietà o delega di permessi tra i ruoli. motivo per cui i nodi non sono legati tra di essi. Questo perché ogni ruolo (Ricercatore, Valutatore, Amministratore) ha responsabilità e permessi molto specifici e distinti, e non sembra esserci un caso d'uso in cui un ruolo avrebbe bisogno di "ereditare" i permessi di un altro ruolo.

Tuttavia, potrebbero esserci scenari in cui l'ereditarietà o la delega dei permessi potrebbe avere senso. Ad esempio, se ci fosse un ruolo di "Capo Valutatore" che ha tutti i permessi di un Valutatore, ma ha anche il permesso di approvare o rifiutare definitivamente un progetto, allora il ruolo di Capo Valutatore potrebbe ereditare i permessi del ruolo di Valutatore e avere permessi aggiuntivi ma nel nostro problema non viene richiesto.