Documento di Analisi dei Requisiti

[**Scopo del Progetto 1**](#_79cwg5c6qf6s)

[**Specifica dei Requisiti 1**](#_jop8ahbtvd4s)

[Requisiti Funzionali 1](#_oew2a5fuiccu)

[Requisiti Non Funzionali 2](#_cf84hppg4oc9)

[Requisiti di Sistema 3](#_vjljeljxdb62)

[**Progettazione 4**](#_ddk3r0pbpcff)

[Casi D’Uso: 4](#_thpy4jwo7qmh)

[Elenco dei File da Realizzare 7](#_8s7wdjcnh7e7)

[**Librerie da Utilizzare 7**](#_6wb5c1t7z9qs)

# **Scopo del Progetto**

Come scopo del progetto si vuole realizzare una piattaforma basata su web, che permetta la creazione, modifica, approvazione, visualizzazione e ripristino di documenti in modo sicuro, in maniera da poterli esporre in maniera chiara e uniforme.

Il sistema sarà basato su due mini-PC, uno che svolgerà la funzione di hosting del sito web, mentre l’altro funzionerà da Database, questi comunicheranno fra di loro in modo sicuro attraverso protocolli di sicurezza come HTTPS, firewall e OpenSSL (TLS).

Il sistema dovrà avere diversi ruoli utente (Admin, Evaluatore, Editor, Utente) con permessi speciali per ogni ruolo.

Il layout del sito sarà fisso e simile a quello di Wikipedia, con una struttura che include titolo, abstract, immagini, testo, e voci correlate.

Mentre se utilizzassimo “Media Wiki” ne dovremmo utilizzare solamente 8 di giorni uomo, di cui 6 giorni per impostare il servizio e i vari ambienti e i rimanenti 2 per allestire la pagina.

# **Specifica dei Requisiti**

## **Requisiti Funzionali**

***Gestione Utenti e Ruoli***:

* Registrazione e login degli utenti.
* Assegnazione e revoca dei ruoli (Admin, Evaluatore, Editor, Utente) da parte dell'Admin.
* Possibilità per un utente di avere più ruoli e scegliere il ruolo al momento del login.

***Gestione Documenti***:

* Creazione di bozze da parte degli Editor.
* Approvazione o rifiuto delle bozze da parte degli Evaluatori.
* Visualizzazione delle versioni approvate dei documenti da parte degli Utenti.
* Salvataggio e ripristino delle versioni precedenti dei documenti.

***Sicurezza e Autenticazione***:

* Autenticazione sicura tramite HTTPS.
* Configurazione di firewall e chiusura delle porte non necessarie.
* Utilizzo di tunnel OpenSSL (TLS) per la comunicazione sicura tra server web e database.

***Layout e Interfaccia Utente***:

* Layout fisso simile a Wikipedia, con titolo, abstract, immagini, testo, e voci correlate.
* Slider per scorrere fino a tre immagini.
* Collegamenti ipertestuali tra pagine diverse.

## **Requisiti Non Funzionali**

***Performance***:

* Il programma permette un utilizzo veloce ed intuitivo al fine di garantire un’esperienza piacevole all’utente.

***Disponibilità***:

* La disponibilità del programma garantisce il funzionamento del programma al 99.9%. Questo significa che l’applicazione funzionerà sempre tranne che in casi di futuri cambiamenti.

***Sicurezza***:

* Tutte le comunicazioni devono essere criptate.
* Il sistema deve essere protetto da attacchi comuni come SQL injection.

***Manutenibilità***:

* Lo sviluppo dell’applicazione è stato fatto al fine di rendere più semplici sviluppi e risoluzioni di problemi in futuro, e che tutto sia ben documentato in caso di modifiche.

## **Requisiti di Sistema**

***Hosting***:

* Due mini-PC su cui è installata l’ultima versione di Ubuntu, uno per il server web (Apache/HTTP) e uno per il database.
* Configurazione di un router/switch per collegare le due macchine.

***Software***:

* Utilizzo di PHP per il lato server e JavaScript per il lato client.
* Configurazione di Apache per l'hosting del sito web.
* Utilizzo di database relazionale (MySQL/MariaDB) per il salvataggio dei dati.

***Sicurezza***:

* Implementazione di un sistema di autenticazione sicuro (es. Radius).
* Configurazione di firewall e chiusura delle porte non necessarie.
* Utilizzo di protocolli di sicurezza come HTTPS e TLS.
* Il server sarà situato all’interno della DMZ (zona demilitarizzata)

# **Progettazione**

## **Casi D’Uso**:

**Caso 1 - Visualizza articoli (UC1)**:

* ***ID***: UC1
* ***Attori***: Utente generico, Evaluatore, Scrittore, Admin
* ***Precondizioni***: Accesso all’applicazione web
* ***Sequenza eventi***:
  + L’utente (loggato o non loggato) accede all’applicazione web.
  + L’applicazione mostra le pagine visualizzabili da chiunque senza necessità di login.
* ***Postcondizioni***: Le pagine dell’applicazione vengono visualizzate correttamente.

**Caso 2 - Scrive i contenuti (UC2)**:

* ***ID***: UC2
* ***Attori***: Scrittore
* ***Precondizioni***: Lo scrittore si è loggato per scrivere contenuti.
* ***Sequenza eventi***:  
  + L’utente si logga come scrittore.
  + Lo scrittore scrive i contenuti.
  + Una volta terminato, i contenuti vengono salvati come bozza.
* ***Postcondizioni***: I contenuti scritti vengono salvati come bozza e non sono modificabili da altri editor fino all’approvazione.

**Caso d’Uso: Approva, rifiuta, ripristina (UC3)**:

* ***ID***: UC3
* ***Attori***: Evaluatore
* ***Precondizioni***: L’utente si logga come evaluatore.
* ***Sequenza eventi***:  
  + L’evaluatore accede alla sezione dell’applicazione che contiene la lista di contenuti in fase di approvazione.
  + L’evaluatore esamina i contenuti.
  + L’evaluatore può approvare, rifiutare o ripristinare la versione precedente del contenuto.
* ***Postcondizioni***: Il contenuto viene processato e restituito nella sua versione esaminata dall’evaluatore.

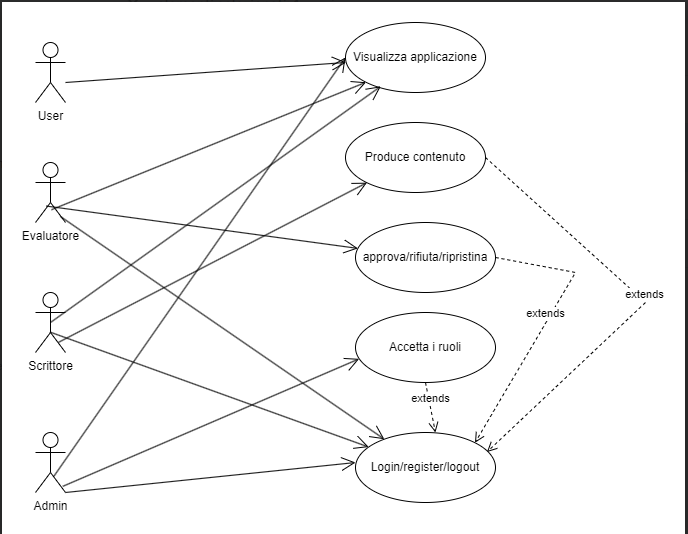
**Caso d’Uso: Accetta i ruoli (UC4)**:

* ***ID***: UC4
* ***Attori***: Admin
* ***Precondizioni***: Un utente fa richiesta per un ruolo specifico.
* ***Sequenza eventi***:  
  + Un utente richiede i permessi per un ruolo specifico.
  + L’admin esamina la richiesta.
  + L’admin decide se approvare o rifiutare la richiesta.

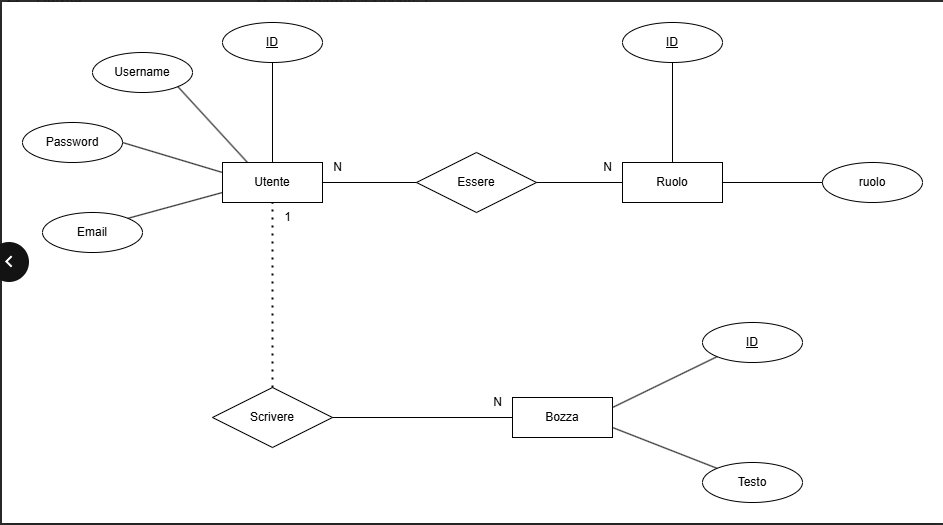
* ***Postcondizioni***: L’utente mantiene il ruolo precedente o ottiene il nuovo ruolo richiesto.

**Caso d’Uso: Registrarsi/loggarsi/sloggarsi (UC5)**:

* ***ID***: UC5
* ***Attori***: Evaluatore, Scrittore, Admin
* ***Precondizioni***: Accesso all’applicazione web.
* ***Sequenza eventi***:  
  + Un utente non loggato accede all’applicazione web.
  + L’applicazione mostra la pagina di registrazione con i ruoli disponibili.
  + Se l’utente è già registrato, può loggarsi scegliendo un ruolo specifico (evaluatore, scrittore, admin).
  + L’utente loggato accede alle aree dell’applicazione specifiche per il ruolo scelto.
  + L’utente può sloggarsi in qualsiasi momento.
  + Postcondizioni: L’utente è loggato o sloggato correttamente, con accesso alle funzionalità del ruolo scelto.
* ***Postcondizioni***: Il contenuto viene processato e restituito nella sua versione esaminata dall’evaluatore



## **Schema E/R**



## **Elenco dei File da Realizzare**

Per realizzare questa applicazione abbiamo pensato che sarebbe stato il caso di dividerla nei seguenti componenti:

* **Navbar**, gestione di tutto ciò che riguarda la navbar.
* **Form**, gestione dell’invio della form e delle eventuali azioni successive all’invio
* **Database**, gestione di tutto cio’ che riguarda le query per salvare e creare il DB
* **Server**, gestione delle porte per permettere lo scambio di dati fra i due server.
* **Table**, gestione delle tabelle che mostrano i contenuti della pagina