

# PROBLEM STATEMENT




## Situazione attuale

Per molti utenti nasce la necessità di avere uno spazio su internet dove pubblicare le proprie esperienze ed opinioni. I sistemi che offrono tali servizi sono spesso complessi e difficili da utilizzare, quindi l'obiettivo che ci siamo posti è la creazione di una piattaforma web che consenta a chiunque abbia minime conoscenze in ambito web di creare il proprio spazio.







Il sistema dovrà permettere agli utenti di creare blog e scrivere o modificare le proprie pagine tramite l'utilizzo del linguaggio [Markdown](#) e garantire un sistema di comunicazione minimale tra gli utenti.

## Requisiti Funzionali

Legenda:

-  - Priorità alta
-  - Priorità media
-  - Priorità bassa





L'utente dovrebbe poter:

- RF1.**  visitare blog creati da altri utenti
- RF2.**  visualizzare i migliori blog
- RF3.**  visualizzare gli ultimi articoli di un particolare blog
- RF4.**  ricercare utenti specifici
- RF5.**  ricercare blog specifici
- RF6.**  registrarsi alla piattaforma

L'utente registrato dovrebbe poter:

- RF7.**  effettuare il login

L'utente autenticato dovrebbe poter:

- RF8.**  creare blog
- RF9.**  cancellare i propri blog
- RF10.**  caricare contenuti nei propri blog
- RF11.**  rimuovere contenuti nei propri blog

- RF12.** ■ visualizzare i contenuti dei propri blog
- RF13.** ■ visualizzare le chat iniziate con altri utenti
- RF14.** ■ creare articoli in Markdown
- RF15.** ■ modificare articoli in Markdown
- RF16.** ■ cancellare articoli in Markdown
- RF17.** ■ effettuare il logout
- RF18.** ■ segnalare messaggi inadeguati
- RF19.** ■ segnalare contenuti inadeguati nei blog
- RF20.** ■ commentare i contenuti dei blog
  
- RF21.** ■ inviare messaggi ad altri utenti

Il moderatore dovrebbe poter:

- RF22.** ■ visualizzare le segnalazioni effettuate

Il moderatore dei contenuti dovrebbe poter:

- RF23.** ■ rimuovere contenuti se la segnalazione viene confermata

Il moderatore delle chat dovrebbe poter:

- RF24.** ■ rimuovere messaggi se la segnalazione viene confermata

Il gestore degli utenti dovrebbe poter:

- RF25.** ■ modificare il ruolo di altri utenti
- RF26.** ■ bandire un utente dalla piattaforma

La piattaforma dovrebbe:

- RF27.** ■ memorizzare i dati dell'utente registrato (username, password)
- RF28.** ■ crittografare la password

# Requisiti non funzionali

## Usability

- RNF1.** ■ tutti gli utenti dovranno essere in grado di creare pagine in Markdown tramite un editor di testo
- RNF2.** ■ verranno utilizzate label e testi popup descrittivi quando necessario

Nella parte superiore delle pagine saranno previsti:

- RNF3.** ■ barra di ricerca
- RNF4.** ■ link alla home
- RNF5.** ■ link al proprio profilo

## Reliability

- RNF6.** ■ La robustezza è garantita da pagine di errore o messaggi a schermo che vengono visualizzati laddove ci sia l'inserimento di un input scorretto
- RNF7.** ■ I dati sensibili degli utenti saranno protetti tramite crittografia e saranno ridotti al minimo per impedire usi impropri
- RNF8.** ■ Il sistema dovrà resistere ai più comuni attacchi alle applicazioni web-based quali XSS e SQL injection

## Performance

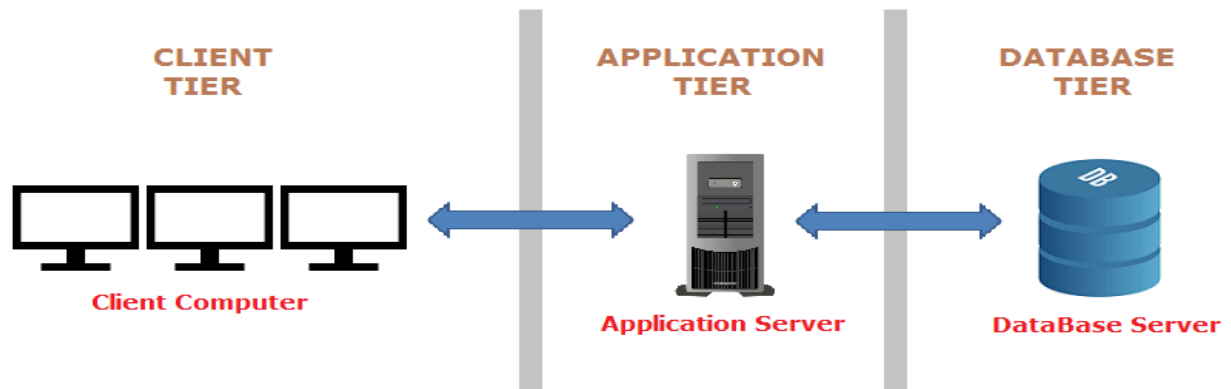
- RNF9.** ■ Le pagine dei blog devono caricarsi in meno di un secondo
- RNF10.** ■ Il sistema dovrebbe essere in grado di servire almeno 200 utenti in contemporanea senza particolari rallentamenti nelle singole richieste
- RNF11.** ■ Il server dovrebbe garantire un uptime di almeno il 95%

## Packaging

- RNF12.** ■ Il sistema dovrà essere installabile mediante il sistema di virtualizzazione Docker.

# Target Environment

Per il sistema si è proposta un'architettura a tre livelli.



## Deliverables attese dal cliente

- Requisiti e casi d'uso: 28 ottobre
- Requirements Analysis Document: 11 novembre
- System Design Document: 25 novembre
- Object Design: 9 dicembre
- Piano di test e specifica interfacce dei moduli di sistema: 16 dicembre
- Implementazione: 31 dicembre
- Report sul testing: 10 gennaio

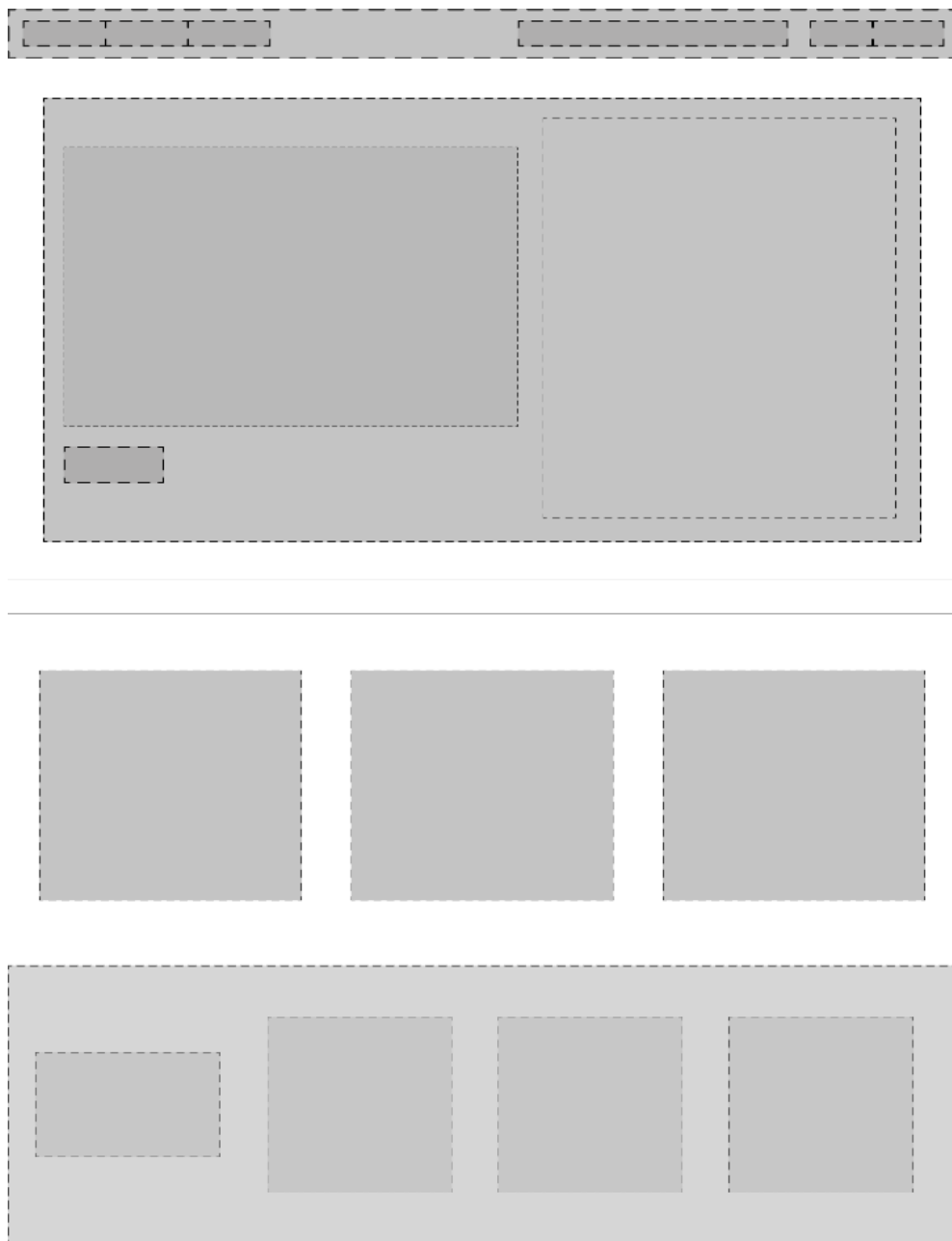
## Criteri di accettazione

Tutti i requisiti funzionali devono necessariamente essere soddisfatti e verificati in accordo con il cliente.

# Scenari

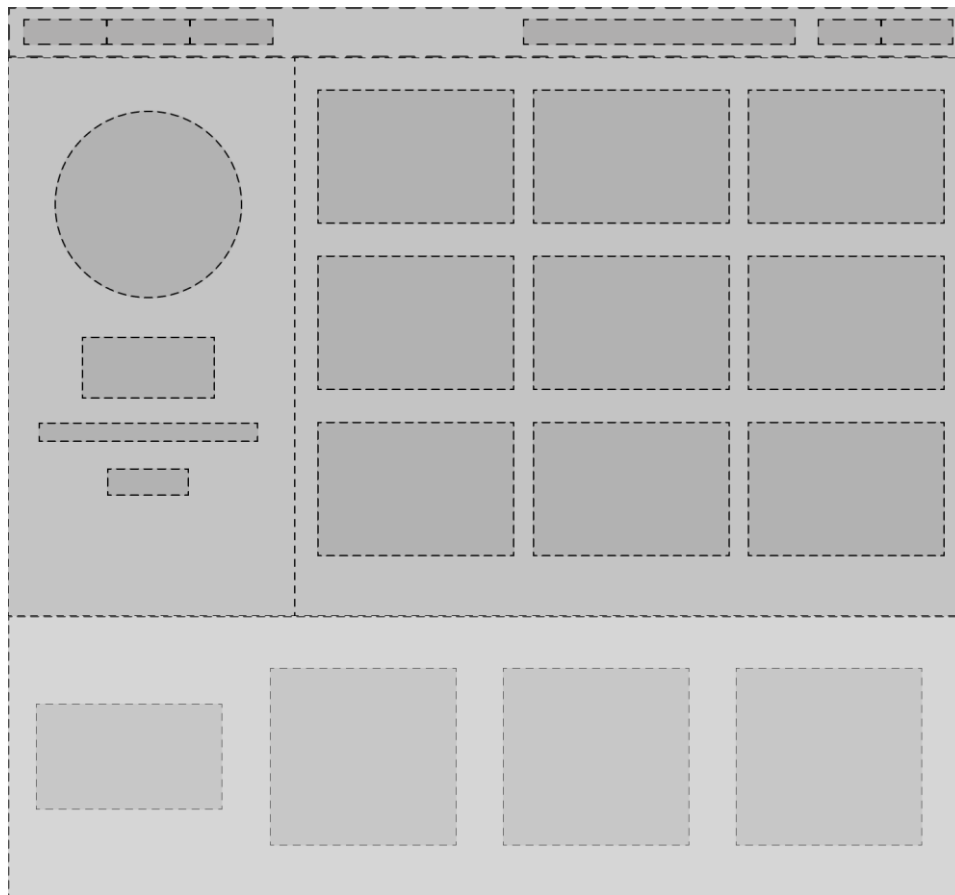
## Scenari utente

ScU1. Anastasia è appassionata di lettura e da sempre vorrebbe avere il suo spazio sul Web per condividere la propria passione e le proprie opinioni sui libri che più ama. Navigando in rete si imbatte in TomMASO, un servizio completamente gratuito e open source che le permette di fare ciò che ha sempre voluto. Una volta sul sito nota subito il link alla pagina che mostra i migliori blog, decide quindi di aprire il primo e le viene mostrata un'anteprima degli ultimi articoli creati. Vedendo la qualità del blog e avendo letto della semplicità con cui è possibile crearli, decide di registrarsi alla piattaforma. Mentre si registra inserisce due password diverse e il sistema le segnala l'errore. Sistemato l'errore, Anastasia riesce finalmente a registrarsi e apre la pagina del suo profilo. Decide quindi di creare subito il suo primo blog usando l'apposito pulsante e le viene chiesto di inserire un nome. Inserisce un nome poco originale ed il sistema la avvisa che un blog con questo nome è già presente, pertanto decide di dargliene uno più distintivo. Il blog viene creato e nel suo profilo compare l'icona ad esso associata. A questo punto si presentano diverse scelte per il blog: visualizzare gli articoli principali, gestire i contenuti e cancellare il blog. Sceglie di gestire i contenuti e decide di caricare un'immagine del suo libro preferito, usa l'apposito pulsante e le vengono mostrate due possibilità: caricare un file e creare un articolo in Markdown. Sceglie di caricare un file e si apre una finestra popup che le permette di selezionare il file dal suo dispositivo, lo sceglie e preme su "Carica". L'immagine viene aggiunta ai contenuti del blog. A questo punto decide di creare un articolo tramite l'editor offerto dalla piattaforma. Inserisce il titolo del suo articolo e le viene mostrato l'editor Markdown, il quale presenta tasti per: grassetto, corsivo, intestazione, citazione, elenco puntato, elenco numerato, link, immagine e nota che è possibile visualizzare una preview dell'articolo in tempo reale. Finita la modifica del documento, lo salva e ne visualizza il risultato.



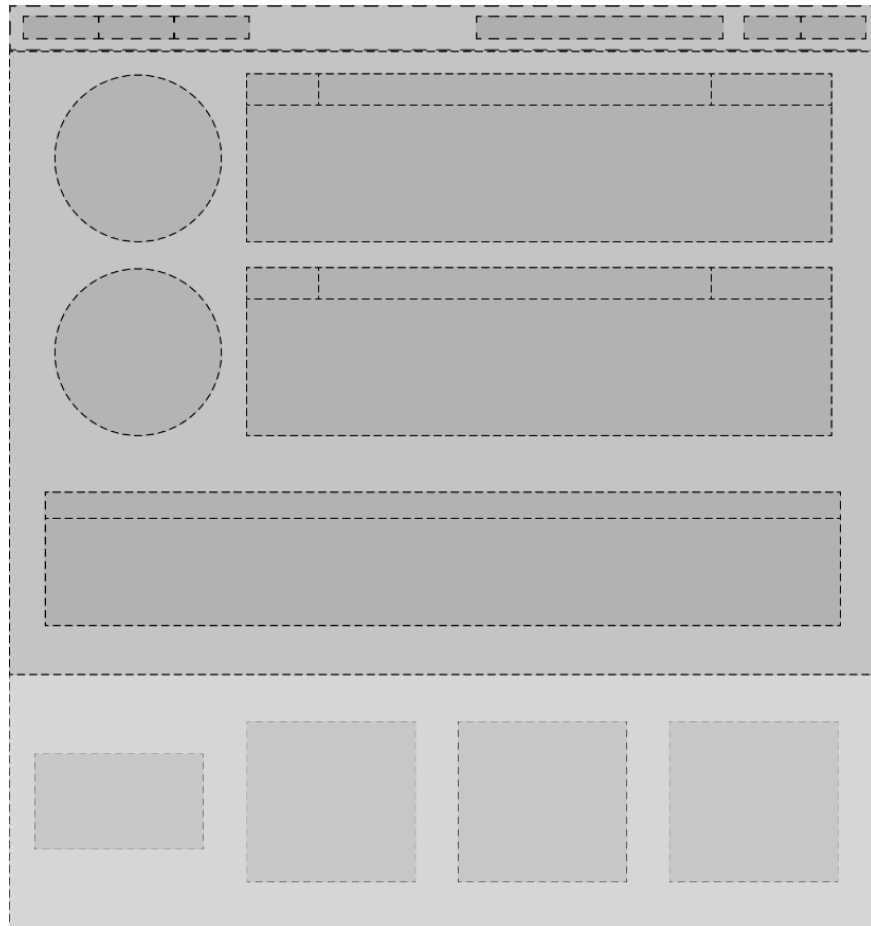
*Mockup schermata home*

ScU2. Bartolomeo è un utente registrato al sito e si imbatte nel blog di Anastasia. Da grande appassionato di lettura, decide di leggere l'articolo sul libro preferito di Anastasia. Lo trova molto interessante e decide di lasciare un commento con le proprie impressioni sul libro. Si dirige nell'area commenti dell'articolo e, tramite la casella di testo, scrive le sue impressioni. Una volta lasciato il commento è ancora molto entusiasta e decide di visualizzare il profilo di Anastasia per controllare se gestisce altri blog.



*Mockup profilo*

ScU3. Carolina è registrata alla piattaforma e guardando l'articolo sul libro preferito di Anastasia si imbatte nel commento lasciato da Bartolomeo. Carolina rimane affascinata dal punto di vista di Bartolomeo e cliccando sulla sua immagine viene reindirizzata al profilo. Tramite l'icona per creare la chat, avvia una conversazione con Bartolomeo e viene reindirizzata alla schermata dedicata alla chat tra i due. A questo punto, scrive in un'area di testo il suo messaggio e lo invia. Il messaggio viene quindi visualizzato sulla chat per entrambi gli utenti.

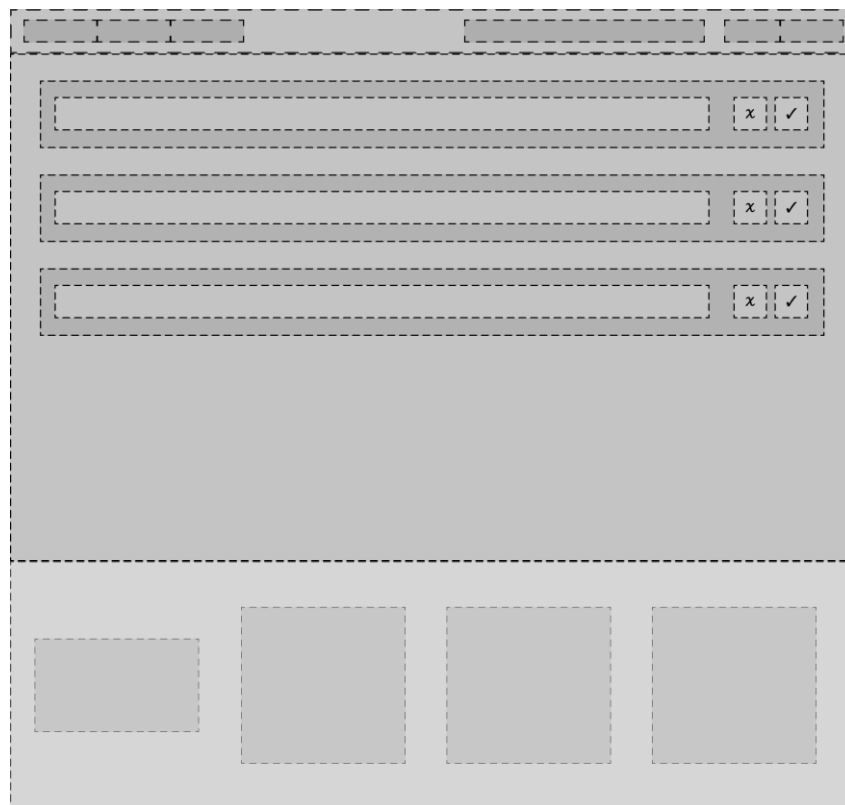


*Mockup schermata chat*



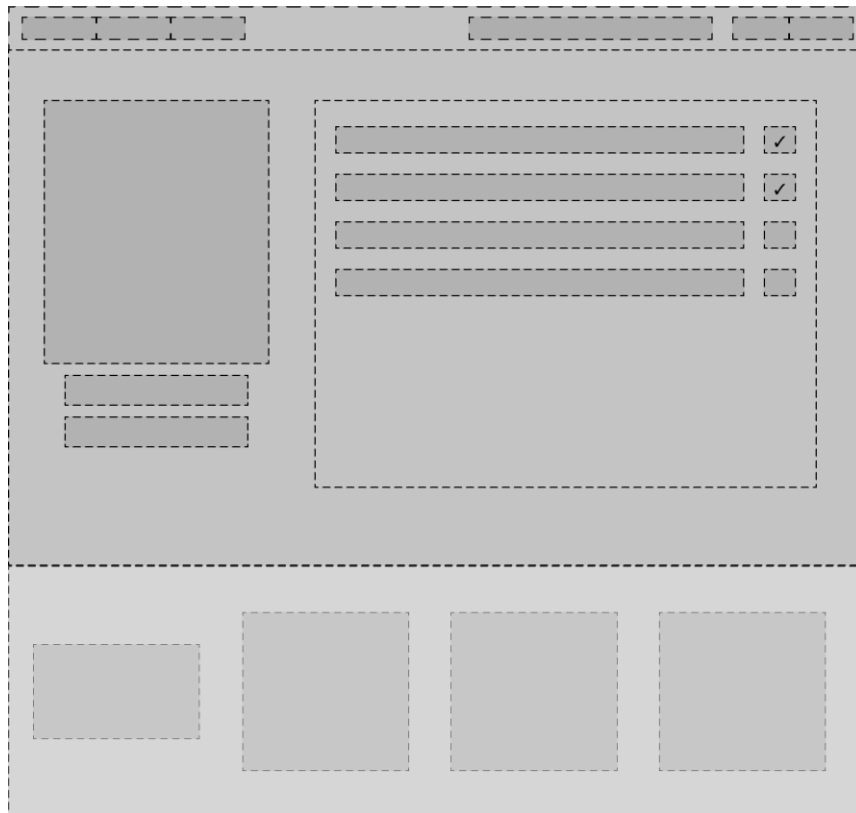
## Scenari d'amministrazione

ScA1. Damiano è un moderatore dei contenuti e delle chat, effettua il login e decide di controllare le segnalazioni ricevute dagli utenti. Clicca sulla propria icona e gli si presenta l'opzione "Vedi report", oltre alle opzioni da normale utente. Una volta entrato nella pagina apposita gli si presentano le segnalazioni consone ai suoi permessi e, poiché Damiano modera sia chat che contenuti, visualizza entrambi i tipi. Clicca sulla prima segnalazione e visualizza il contenuto segnalato. Dopo un controllo dei contenuti Damiano può decidere se far passare la segnalazione o cancellarla, sceglie di confermarla e il contenuto segnalato viene immediatamente rimosso.



*Mockup schermata segnalazioni*

ScA2. Erica è un gestore degli utenti, effettua il login e decide di rendere la sua amica Fabiana una moderatrice dei contenuti. Cerca il suo profilo inserendo l'username nella barra di ricerca apposita. Una volta sul profilo di Fabiana clicca il tasto per cambiare i permessi, aggiunge il permesso di moderare i contenuti e conferma la sua scelta



*Mockup schermata permessi*