

Specifiche dei Requisiti per NoProfitHub di Imperatore Luigi A13002767

Requisiti Funzionali

1. Autenticazione e gestione utenti
2. Gestione progetti e candidature
3. Gestione turni
4. Matching progetti-volontari
5. Attestati e ore di volontariato
6. Dashboard utente e associazione
7. Notifiche e comunicazioni
8. Valutazioni e feedback
9. Storico attività e tracciamento

Requisiti Non Funzionali (RNF) – NoProfitHub

- Prestazioni e Scalabilità
- Sicurezza e accesso
- Usabilità e UX
- Manutenibilità e modularità
- Compatibilità
- Monitoraggio e logging

Progettazione Logica

Piattaforma di Sviluppo

Consegna

Requisiti Funzionali

1. Autenticazione e gestione utenti

- **RF1.1:** Gli utenti possono registrarsi come volontari o associazioni.
 - **RF1.2:** Gli utenti possono effettuare il login con email e password.
 - **RF1.3:** Gli utenti possono modificare le proprie informazioni personali (profilo, email, password).
 - **RF1.4:** I volontari possono selezionare competenze da un elenco predefinito.
 - **RF1.5:** I volontari possono indicare disponibilità settimanale (giorni/orari).
-

2. Gestione progetti e candidature

- **RF2.1:** Le associazioni possono creare, modificare e cancellare progetti.
 - **RF2.2:** Ogni progetto può avere requisiti di competenze, orari e numero massimo di volontari.
 - **RF2.3:** I volontari possono visualizzare l'elenco dei progetti pubblici e candidarsi.
 - **RF2.4:** Le associazioni possono accettare o rifiutare le candidature dei volontari.
 - **RF2.5:** I volontari possono vedere lo stato della propria candidatura (in attesa / accettata / rifiutata).
-

3. Gestione turni

- **RF3.1:** Le associazioni possono creare turni per i propri progetti (data, orario, max volontari).

- **RF3.2:** Le associazioni possono assegnare manualmente i volontari accettati ai turni.
 - **RF3.3:** I volontari possono visualizzare solo i turni a cui sono assegnati.
 - **RF3.4:** Il sistema impedisce l'assegnazione se un turno è pieno.
 - **RF3.5:** Il sistema controlla che non ci siano sovrapposizioni di turni per uno stesso volontario.
-

4. Matching progetti-volontari

- **RF4.1:** Il sistema suggerisce progetti compatibili ai volontari in base a competenze e disponibilità.
 - **RF4.2:** Il sistema suggerisce volontari compatibili alle associazioni per ogni progetto.
-

5. Attestati e ore di volontariato

- **RF5.1:** Le associazioni possono generare attestati PDF per i volontari dopo la conclusione di un progetto.
 - **RF5.2:** I volontari possono visualizzare e scaricare i propri attestati.
 - **RF5.3:** Il sistema traccia le ore di volontariato svolte da ciascun utente.
-

6. Dashboard utente e associazione

- **RF6.1:** I volontari hanno una dashboard con:

- Profilo, progetti compatibili, candidature, turni assegnati, ore svolte, attestati.
 - **RF6.2:** Le associazioni hanno una dashboard con:
 - Progetti, volontari candidati, gestione turni, attestati, statistiche.
 - **RF6.3:** L'amministratore ha una dashboard con statistiche globali: ore totali, numero volontari, progetti, top associazioni.
-

7. Notifiche e comunicazioni

- **RF7.1:** Il sistema invia email per le azioni chiave:
 - Nuova candidatura, accettazione, rifiuto, assegnazione turno.
 - **RF7.2:** I volontari ricevono notifiche interne sul portale per questi eventi.
-

8. Valutazioni e feedback

- **RF8.1:** Alla fine di ogni progetto, l'associazione può valutare i volontari.
 - **RF8.2:** I volontari possono valutare l'associazione al termine del progetto.
 - **RF8.3:** I feedback possono essere consultati solo dalle controparti coinvolte.
-

9. Storico attività e tracciamento

- **RF9.1:** I volontari hanno uno storico con l'elenco dei progetti svolti, ore, turni, attestati.

- **RF9.2:** Le associazioni vedono la cronologia dei progetti completati e delle attività svolte.
-

Requisiti Non Funzionali (RNF) – NoProfitHub

I Requisiti Non Funzionali definiscono come il sistema deve comportarsi, al di là delle funzioni operative.

Prestazioni e Scalabilità

- **RNF1:** Il sistema deve rispondere alle richieste dell'utente entro 2 secondi nella maggior parte delle operazioni.
 - **RNF2:** La piattaforma deve supportare almeno 10.000 utenti registrati senza degrado delle prestazioni.
 - **RNF3:** Il caricamento delle dashboard non deve superare i 3 secondi.
-

Sicurezza e accesso

- **RNF4:** Le password devono essere memorizzate in formato hash (es. bcrypt).
 - **RNF5:** Ogni utente può accedere solo ai dati per cui ha i permessi (volontario, associazione, admin).
 - **RNF6:** Le operazioni CRUD devono essere protette da autenticazione e autorizzazione.
-

Usabilità e UX

- **RNF7:** L'interfaccia utente deve essere intuitiva e navigabile da utenti non tecnici.
 - **RNF8:** I messaggi di errore e conferma devono essere chiari e localizzati in italiano.
-

Manutenibilità e modularità

- **RNF9:** Il codice deve essere modulare, separando logica di business, interfaccia e accesso al database.
 - **RNF10:** Il sistema deve permettere l'aggiunta di nuove funzionalità (es. nuove tipologie di progetti) con minime modifiche.
-

Compatibilità

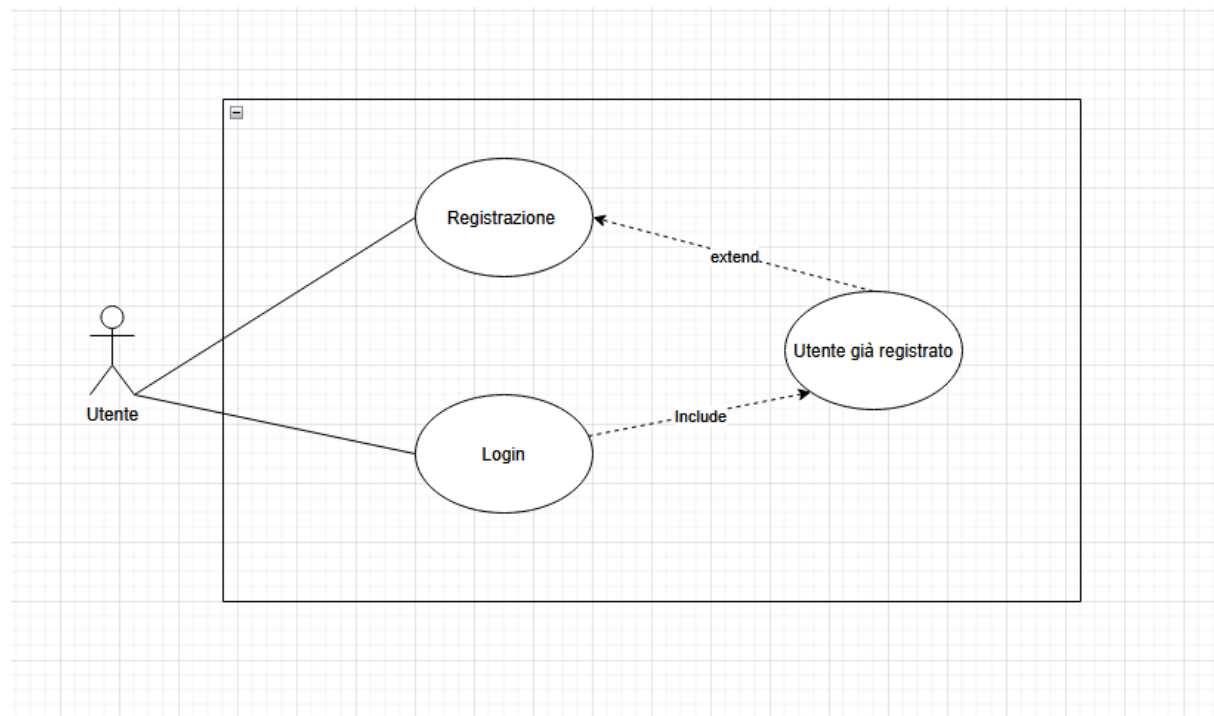
- **RNF11:** Il sistema deve funzionare correttamente con i browser: Chrome, Firefox, Edge.
 - **RNF12:** I file PDF (es. attestati) devono essere generabili e scaricabili senza necessità di plugin esterni.
-

Monitoraggio e logging

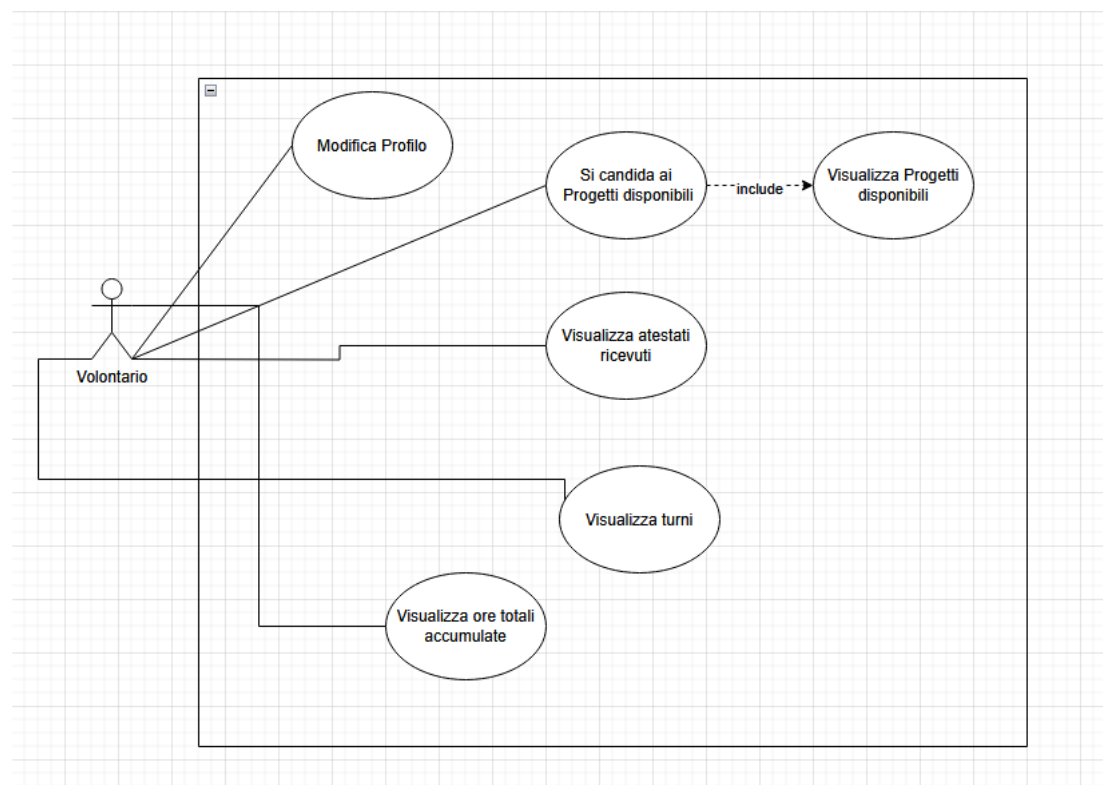
- **RNF13:** Il sistema deve registrare in un log le azioni importanti: login, creazione/modifica di progetti, assegnazione turni.

Diagrammi dei casi D'uso

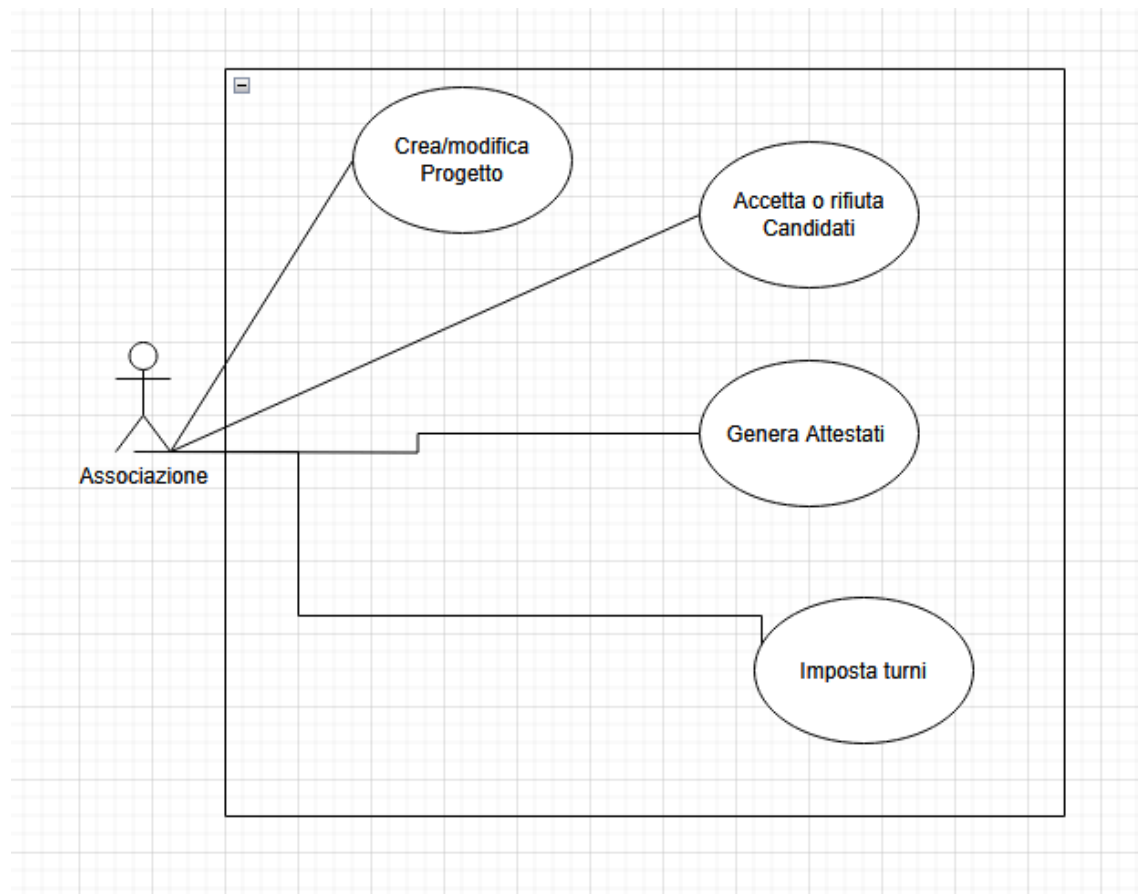
Registrazione/accesso



Dashboard Volontario



Dashboard Associazione

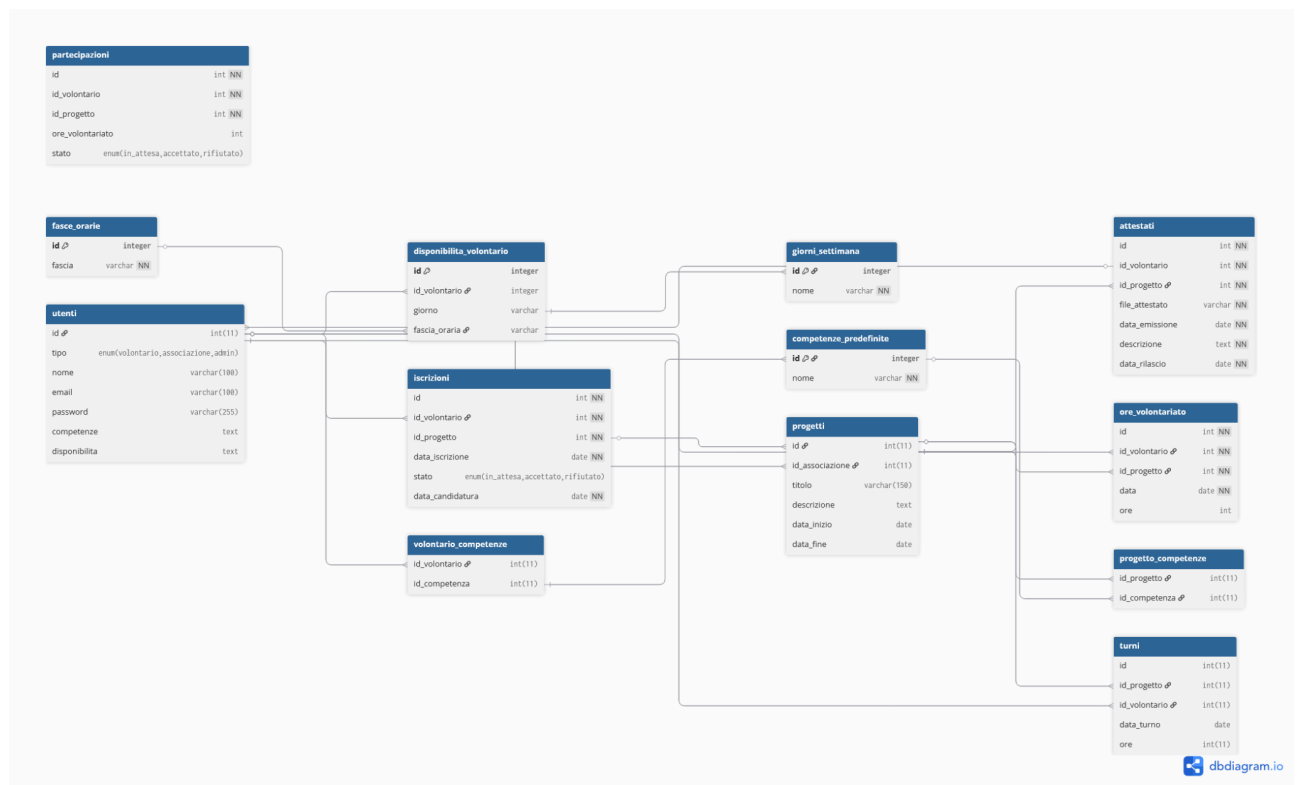


Modello dei Dati

Tabella	Funzione
utenti	Contiene volontari e associazioni (probabilmente con un campo "tipo")
progetti	Contiene i progetti creati dalle associazioni
iscrizioni	Probabilmente contiene le candidature ai progetti
turni	Turni associati ai progetti
partecipazioni	Associa i volontari ai turni (turni assegnati)
attestati	Attestati rilasciati ai volontari
ore_volontariato	Tiene traccia delle ore svolte (magari utile per statistiche)
competenze_predefinite	Elenco delle competenze disponibili

- volontario_competenze** Relazione volontari ↔ competenze
- progetto_competenze** Requisiti di competenze per ciascun progetto
- disponibilita_volontario** Disponibilità dei volontari (giorni e fasce)
- giorni_settimana** Supporto per la disponibilità settimanale
- fasce_orarie** Supporto per la disponibilità oraria

Progettazione Logica



Piattaforma di Sviluppo

- **Front-end:** HTML, CSS.
- **Back-end:** PHP per la logica server-side e la comunicazione con il database.
- **Database:** MySQL per la memorizzazione dei dati.

Consegna

- Applicazione completa con codice sorgente documentato.
- Istruzioni per l'installazione e configurazione del sistema.