Gestione whitelist di un proxy

1 Introduzione 3

1.1 Informazioni sul progetto 3

1.2 Abstract 3

1.3 Scopo 3

2 Analisi 3

2.1 Analisi del dominio 3

2.2 Analisi e specifica dei requisiti 4

2.3 Use Case 5

2.4 Pianificazione 6

2.4.1 Analisi 1

2.4.2 Progettazione 1

2.4.3 Implementazione 1

2.4.4 Documentazione 1

2.5 Analisi dei mezzi 2

2.5.1 Software 2

2.5.2 Hardware 2

3 Progettazione 3

3.1 Design dell’architettura di sistema 3

3.2 Design dei dati e database 3

3.3 Design procedurale 4

4 Implementazione 5

5 Test 5

5.1 Protocollo di test 5

5.2 Mancanze/limitazioni conosciute 5

6 Consuntivo 5

7 Conclusioni 5

7.1 Sviluppi futuri 5

7.2 Considerazioni personali 5

8 Bibliografia 5

8.1 Sitografia 5

Allegati 5

# Introduzione

## Informazioni sul progetto

Autore: Carlo Pezzotti

Scuola: Arti e Mestieri Trevano

Classe: I4AC

Anno scolastico: 2019-2020

Sezione: Informatica

Docenti responsabili: Ivan Raimondi

Data di inizio: 03.09.2019

Data di consegna: 20.12.2019

## Abstract

----

## Scopo

Lo scopo di questo progetto è quello di creare un’applicazione web che semplifichi la gestione dei siti nella whitelist di un proxy in ogni suo aspetto.

# Analisi

## Analisi del dominio

Il dominio per questa applicazione è pressochè inesistente, in quanto l'applicazione per la gestione di un proxy deve venir fatta da capo partendo dai requisiti del cliente e non esistono applicazioni che potremmo usare come modello dalle quali ispirarci.

Il prodotto dovrà funzionare in contesto scolastico o lavorativo, dove un proxy è fondamentale per la sicurezza informatica siccome.

Gli utenti che utilizzerano il prodotto saranno sicuramente di due tipi,

gli informatici che si occuperanno della parte amministrativa e l'utente medio che lo utilizzerà. Siccome si basa per una fascia di utenti variabile, il suo funzionamento dovrà essere semplificato al massimo.

## Analisi e specifica dei requisiti

Il committente ha bisogno di un prodotto che gli fornisca la possibilità di gestire un proxy da un pagina web. Dovrà gestire,inoltre , gli utenti che possono accedere a questa gestione del proxy, perchè, come specificato durante gli incontri, per accedere alla gestione bisogna passare da una pagina di login.

Il prodotto finale quindi dovrà essere sia i sistema che ospiterà il prodotto e il proxy che verrà gestito.

Il prodotto, sotto richiesta, avrà delle interfaccie con un grafica minima per renderlo leggero e trasportabile in caso di necessità.

Le prestazione minima che il prodotto dovrà fornire è la gestione del proxy.

Siccome il proxy server per garantire sicurezza all'interno di una rete, il grado di sicurezza dovrà essere elevato.

|  |  |
| --- | --- |
| **ID: REQ-001** | |
| Nome | Ubuntu server |
| Priorità | 1 |
| Versione | 1.0 |
| Note |  |
| *Sotto-requisiti* | |
| 001 | Proxy Squid |
| 002 | PHP |
| 003 | Apache |

|  |  |
| --- | --- |
| **ID: REQ-002** | |
| Nome | Proxy Squid |
| Priorità | 1 |
| Versione | 1.0 |
| Note |  |
| *Sotto-requisiti* | |
| 001 | Deve girare dulla porta 8080 |
| 002 | Non deve avere limitazioni |

|  |  |
| --- | --- |
| **ID: REQ-003** | |
| Nome | Interfaccia login |
| Priorità | 2 |
| Versione | 1.0 |
| Note |  |
| *Sotto-requisiti* | |
| 001 | Dovrà essere presente un capo per l’username e uno per la password |

|  |  |
| --- | --- |
| **ID: REQ-004** | |
| Nome | Interfaccia di gestione utenti |
| Priorità | 2 |
| Versione | 1.0 |
| Note |  |
| *Sotto-requisiti* | |
| 001 | Pagina per poter modificare/rimuovere/aggiungere utenti. |
| 002 | Ogni utente dovrà avere un grado di permesso (user/admin) |

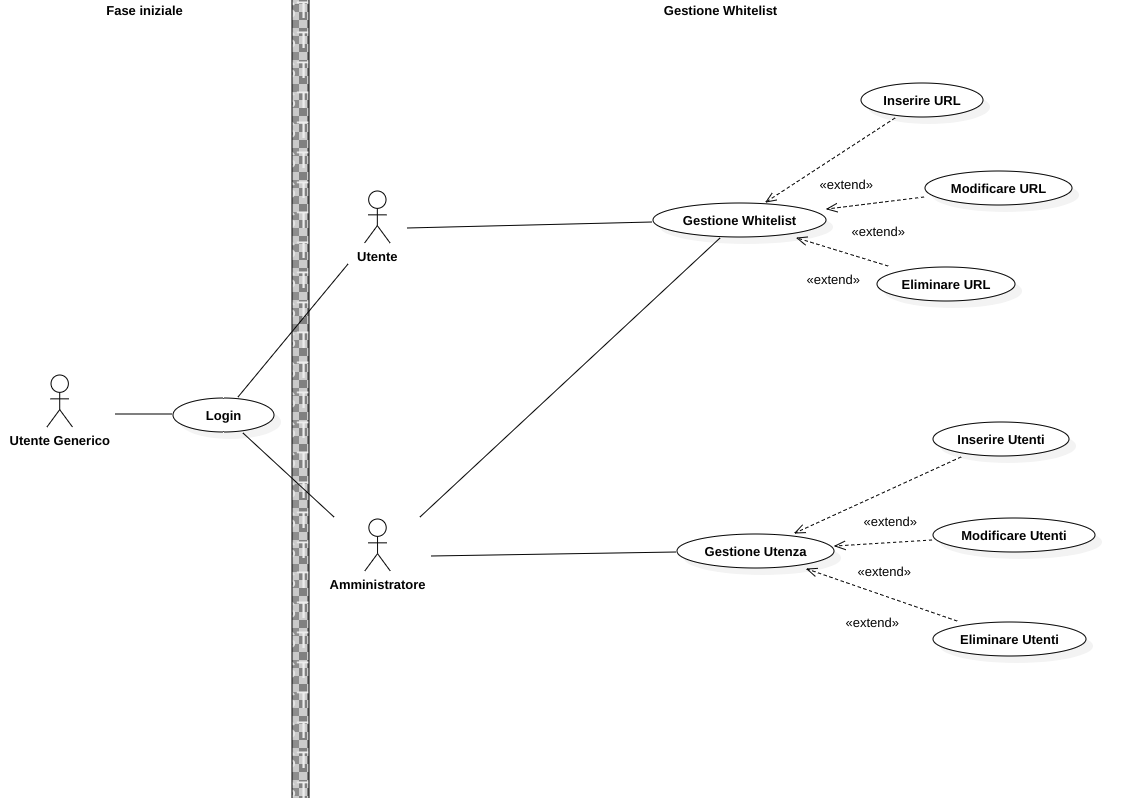
|  |  |
| --- | --- |
| **ID: REQ-005** | |
| Nome | Gestione lista whiteslist |
| Priorità | 1 |
| Versione | 1.0 |
| Note |  |
| *Sotto-requisiti* | |
| 001 | Controllo esistenza sito |
| 002 | Controllo sintassi |
| 003 | Scrittura corretta su file |
| 004 | Auto riavvio alla modifica della lista |

|  |  |
| --- | --- |
| **ID: REQ-006** | |
| Nome | Grafica minima |
| Priorità | 2 |
| Versione | 1.0 |
| Note |  |
| *Sotto-requisiti* | |
| 001 | Pagine con grafica minimale |

|  |  |
| --- | --- |
| **ID: REQ-007** | |
| Nome | Database |
| Priorità | 2 |
| Versione | 1.0 |
| Note |  |
| *Sotto-requisiti* | |
| 001 | Tabella per gestione utenti |

## Use Case

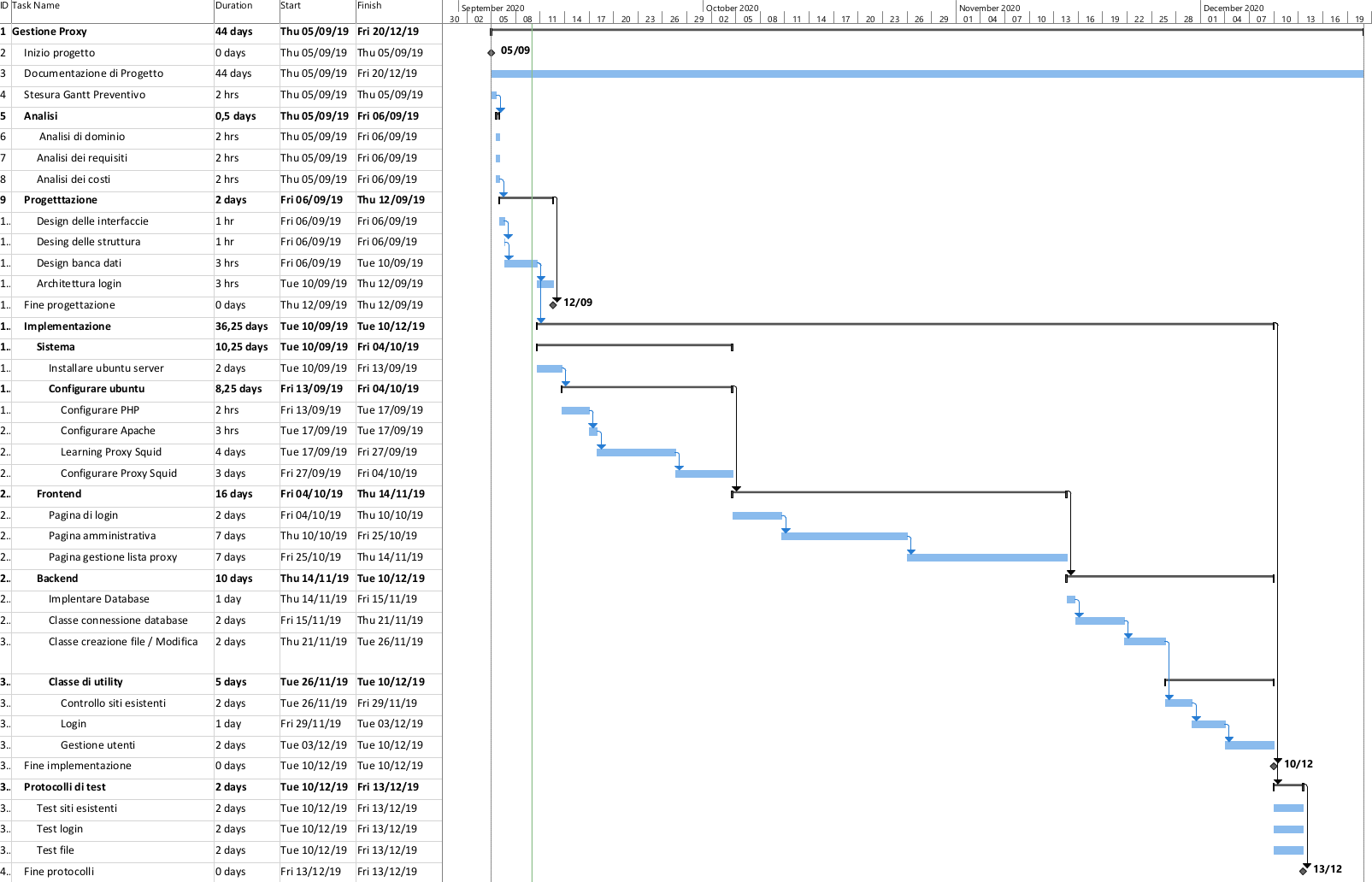
Un utente generico fa il login alla pagina.  
Gli utenti si dividono in 2 categorie  
Utenti normali e Amministratori.



Gli **utenti normali**, possono solo interragire con la gestione della whitelist.

Gli **amministratori** possono sia fare quello che possono fare gli utenti normale e in più gestire la utenza del sito.

## Pianificazione



.

### Analisi

L’attività delle analisi si divide in 3 sotto attività:

**Analisi di dominio:** ovvero l’analisi della situazione prima della realizzazione del progetto per considerare la validità di iniziare un progetto oppure no.

**Analisi dei requisiti:** ovvero l’analisi più approfondita di quello che il commitente chiede che venga realizzato come progetto, nel mio caso hoo dovuto valutare che sistema operativo scegliere, che versione di proxy squid usare e come realizzare (bozza) le varie interfaccie presenti.

**Analisi dei costi:**ovvero il costo completo complessivo che questo progetto verrà a costare al mio commitente in questo caso nullo.

### Progettazione

La progettazione è consistita nell’organizzare l’implementazione del progetto in modo da dividere il lavoro all’interno del team e non avere problemi in cui due componenti stanno lavorando su cose contrastanti. Essa si divide in:

* Design banca dati, ovvero la progettazione del database tramite la realizzazione di un diagramma Entità/Relazioni e del relativo schema logico.
* Infrastruttura del server, ossia la progettazione di ogni componente che verrà poi installato sul server di produzione.

### Implementazione

L’implementazione è stata la parte più lunga del progetto oltre alla documentazione. È l’attività durante la quale ho dovuto realizzare il progetto in ogni suo aspetto, basandomi sulla progettazione fatta nel punto precedente. Essa è divisa nelle seguenti operazioni:

* Installare ambiente, che a sua volta è categorizzato da:
  + Installare server, ovvero l’installazione e la configurazione di tutti i componenti di base necessari per l’utilizzo del sistema operativo.
* Configurare ambiente, che si divide in:
  + Installare MySQL
  + Installare PHP
  + Installare Apache2
  + Installare Proxy Squid
    - Configurare Proxy Squid.
* Realizzare Frontend, ovvero realizzare le seguenti pagine web:
  + Pagina di login
  + Pagina gestione utenti
  + Pagina gestione lista
* Realizzare Backend
  + Gestione delle connesioni con il database
  + Gestione del login
  + Getione dei siti dela whitelist
* Protocolli di test
  + Controllo che i siti siano esistenti
  + Controlli anti sql injection
  + Controllo sui file

### Documentazione

Infine c’è un’attività rimasta costante durante tutto il progetto: la documentazione. Infatti chi aveva dei tempi morti da riempire lo faceva documentando il progetto e aggiungendo informazioni al file che state leggendo in questo momento in modo da non doverla scrivere tutta in poco tempo alla fine

## Analisi dei mezzi

### Software

Macchina server:

* Proxy Squid
* Apache 2
* PHP 7.2

Macchina sviluppo:

* Apache 2
* PHP 7.2
* Visual Studio Code (Ide e Debugger)

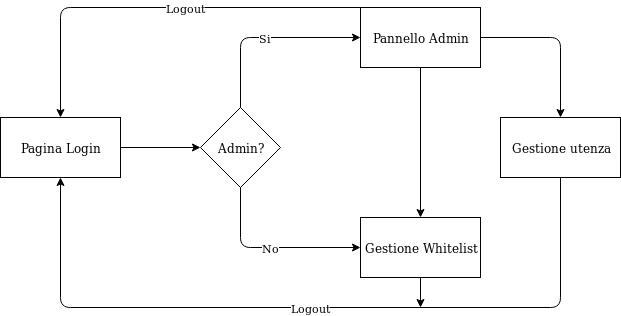
### Hardware

Macchina server:

* Ubuntu Server 18.4
* 1 GB di memoria RAM
* 25 GB di disco disponibili

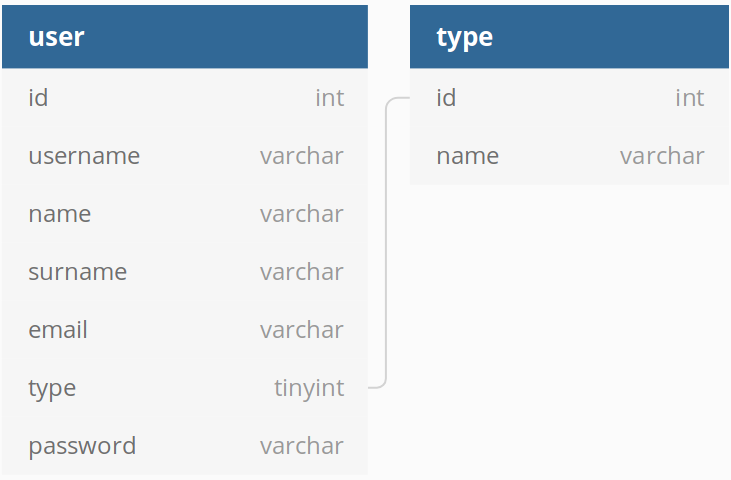
# Progettazione

## Design dell’architettura di sistema



## Design dei dati e database

Il database sviluppato per essere utilizzato con questa applicazione è stato progettato tramite il seguente schema E/R e logico:



**User:**

1. ID: L’identificatore del utente che verrà utilizzato per identificarlo nelle pagine
2. Username: L’username del’utente utilizzato per loggarsi nel sito
3. Name: nome dell’utente
4. Surname: cognome dell’utente
5. Email: email dell’utente
6. Type: tipo dell’utente che si collega alla tabella type
7. Password: password dell’utente

**Type:**

1. ID: L’identificare della tabella
2. Nome: nome (admin/user)

## Design procedurale

Le classi presenti in questo progetto sono le seguenti, divise per linguaggio:

# Implementazione

# Test

## Protocollo di test

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-001  REQ-001 | **Nome:** |  |
| **Descrizione:** |  | | |
| **Prerequisiti:** |  | | |
| **Procedura:** |  | | |
| **Risultati attesi:** |  | | |

## Mancanze/limitazioni conosciute

# Consuntivo

# Conclusioni

## Sviluppi futuri

Si potrebbe aggiungere oltre alla whitelist anche un blacklist dei siti..

## Considerazioni personali

# Bibliografia

## Sitografia

# Allegati

Elenco degli allegati, esempio:

* Diari di lavoro
* Codici sorgente/documentazione macchine virtuali
* Istruzioni di installazione del prodotto (con credenziali di accesso) e/o di eventuali prodotti terzi
* Eventuali guide utente / Manuali di utilizzo
* Mandato e/o Qdc
* Prodotto
* …