Catalogo connessioni per switch di rete

[1 Introduzione 3](#_Toc59108287)

[1.1 Informazioni sul progetto 3](#_Toc59108288)

[1.2 Abstract 3](#_Toc59108289)

[1.3 Scopo 3](#_Toc59108290)

[2 Analisi 4](#_Toc59108291)

[2.1 Analisi del dominio 4](#_Toc59108292)

[2.2 Analisi e specifica dei requisiti 4](#_Toc59108293)

[2.3 Use case 5](#_Toc59108294)

[2.4 Pianificazione 6](#_Toc59108295)

[2.5 Analisi dei mezzi 7](#_Toc59108296)

[2.5.1 Software 7](#_Toc59108297)

[2.5.2 Hardware 7](#_Toc59108298)

[3 Progettazione 8](#_Toc59108299)

[3.1 Design dell’architettura del sistema 8](#_Toc59108300)

[3.1.1 Schema di rete 8](#_Toc59108301)

[3.1.2 Sitemap 8](#_Toc59108302)

[3.2 Design dei dati e database 9](#_Toc59108303)

[3.2.1 Database iniziale 9](#_Toc59108304)

[3.2.2 Database finale 9](#_Toc59108305)

[3.3 Design delle interfacce 10](#_Toc59108306)

[3.4 Design procedurale 18](#_Toc59108307)

[4 Implementazione 19](#_Toc59108308)

[4.1 Prerequisiti 19](#_Toc59108309)

[4.1.1 VM 19](#_Toc59108310)

[4.1.2 Software 19](#_Toc59108311)

[4.2 Programmazione 19](#_Toc59108312)

[4.2.1 Database 19](#_Toc59108313)

[4.2.2 Sito 19](#_Toc59108314)

[4.2.3 Certificati 20](#_Toc59108315)

[4.3 Guida installazione 20](#_Toc59108316)

[5 Test 21](#_Toc59108317)

[5.1 Protocollo di test 21](#_Toc59108318)

[5.2 Risultati test 22](#_Toc59108319)

[6 Consuntivo 23](#_Toc59108320)

[7 Conclusioni 24](#_Toc59108321)

[7.1 Sviluppi futuri 24](#_Toc59108322)

[7.2 Considerazioni personali 24](#_Toc59108323)

[8 Bibliografia 24](#_Toc59108324)

[8.1 Sitografia 24](#_Toc59108325)

[8.2 Indice delle figure 25](#_Toc59108326)

[9 Allegati 25](#_Toc59108327)

# Introduzione

## Informazioni sul progetto

Allievi coinvolti nel progetto: Samuele Ganci.

Classe: 3AC, Scuola Arti e Mestieri, sezione Informatica.

Docenti responsabili: Geo Petrini.

Data inizio: 03.09.2020.

Data di consegna: 17.12.2020.

## Abstract

*Sei un sistemista che si occupa di gestire le connessioni di dispositivi all’interno di un’azienda o scuola? Hai bisogno qualcosa di versatile e veloce per poter fare l’inventario dei dispositivi? Se si, sei nel posto giusto. Questo progetto di nome “Catalogo connessioni per switch di rete”, non solo si occupa di aiutarti nel gestire i dispositivi, ma anche di gestire gli switch al quale essi sono collegati. Ma non finisce qui, il prodotto si basa sul web, ciò vuol dire che non solo si ha un catalogo elettronico, ma può essere possibile accederci da qualunque posto con connessione ad internet se voluto, perfino da un dispositivo mobile.*

## Scopo

Lo scopo di questo progetto, è principalmente di semplificare il lavoro dei sistemisti che si occupano di gestire i dispositivi di un luogo, più precisamente, fornisce un catalogo elettronico dove è possibile catalogare gli switch secondo il modello, la posizione, eventualmente l’etichetta ed infine il numero di porte. Per ognuno di essi, si possono gestire i dispositivi connessi mediante quale cavo. Lo scopo, è di avere uno strumento versatile con il quale poter gestire i collegamenti, essere accessibile da qualsiasi posto e da qualsiasi dispositivo connesso ad internet se desiderato.

# Analisi

## Analisi del dominio

È stato richiesto lo sviluppo di un’applicazione utile a sostituire i metodi analogici di catalogare i dispositivi di una rete, con quelli digitali. L’applicazione deve essere basata sul web e responsive, per poter essere facilmente accessibile da qualunque dispositivo e per poter essere utilizzata ovunque. Il programma deve permettere di catalogare degli Switch, specificando le loro caratteristiche, oltre ai quali, deve essere possibile specificare quali dispositivi ci sono connessi mediante quale tipo di cavo. Il sistema deve anche essere munito di un sistema di login secondo 3 utenti, amministratore, operatore e viewer. A dipendenza del ruolo dell’utente, deve essere possibile aggiungere, modificare o rimuovere Switch, dispositivi, cavi, utenti oppure semplicemente visualizzare e stampare i dati.

## Analisi e specifica dei requisiti

L’applicazione deve essere munita di sistema di login, senza la possibilità di accesso anonimo secondo 3 ruoli:

* Amministratore, gestisce i dispositivi, i cavi e gli utenti, aggiungendone, modificandoli o eliminandoli.
* Operatore, si occupa di aggiungere, modificare o eliminare gli switch, per ognuno di esso è in grado di aggiungere alla porta selezionata un collegamento con i cavi e i dispositivi messi a disposizione dall’amministratore.
* Viewer, può solo visualizzare gli switch e i collegamenti, eventualmente stamparne le informazioni.

Inoltre il prodotto deve essere web-based e responsive, con relativo DB dove salvare i dati.

|  |  |
| --- | --- |
| **ID: REQ-001** | |
| **Nome** | Applicativo web based. |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** |  |
| **Sotto requisiti** | |
| **001** | L’accesso al sito deve avvenire in HTTPS. |
| **002** | Il sito deve essere responsive. |
| **003** | Il sito deve essere munito di un DB nel quale salvare i dati. |

|  |  |
| --- | --- |
| **ID: REQ-002** | |
| **Nome** | Accesso con login |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Non deve essere possibile l’accesso anonimo. |
| **Sotto requisiti** | |
| **001** | Pagina di login secondo 3 tipi di utenti (ruoli): amministratore, operatore e viewer.   * Amministratore: può aggiungere, rimuovere o modificare gli utenti, i cavi e i dispositivi. * Operatore: può aggiungere, rimuovere o modificare gli switch, per ognuno di esso può aggiungere, rimuovere o modificare i collegamenti, specificando la porta dello switch, il dispositivo e il cavo messi a disposizione dall’amministratore. * Viewer: solo visione ed esportazione dei dati di ogni singolo switch. |

|  |  |
| --- | --- |
| **ID: REQ-003** | |
| **Nome** | Catalogo Switch |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** |  |
| **Sotto requisiti** | |
| **001** | Possono essere catalogati più Switch fornendo i seguenti attributi: Modello. Numero di porte. Posizione (luogo dove giace lo Switch). ID (identificativo fornito dall’utente, altrimenti dal software). |
| **002** | Per ogni porta dello Switch catalogato, è possibile selezionare tramite dei menu il tipo di dispositivo collegato e mediante quale cavo. |

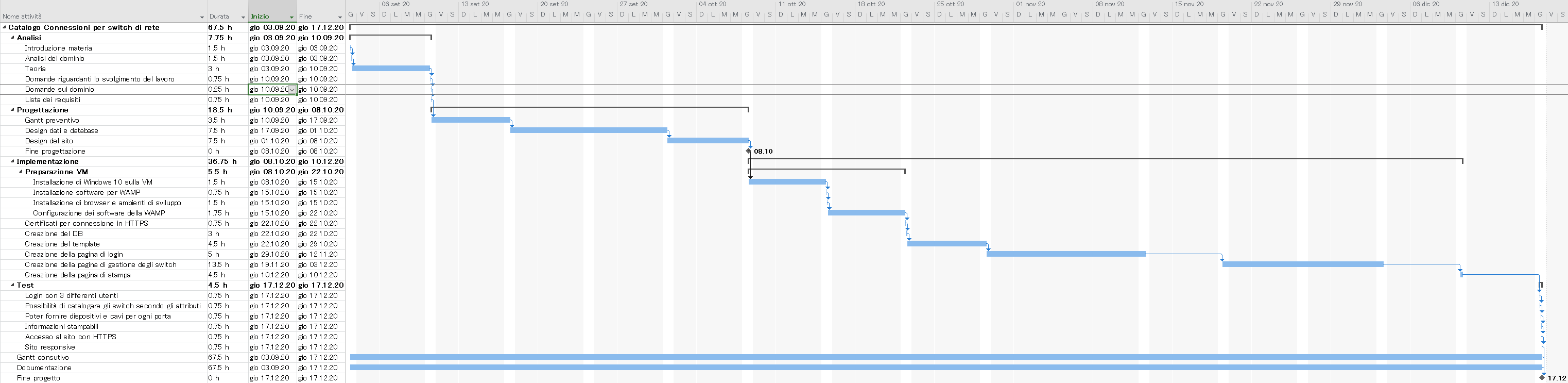
|  |  |
| --- | --- |
| **ID: REQ-004** | |
| **Nome** | Possibilità di stampare una lista con tutte le informazioni di uno Switch. |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** |  |

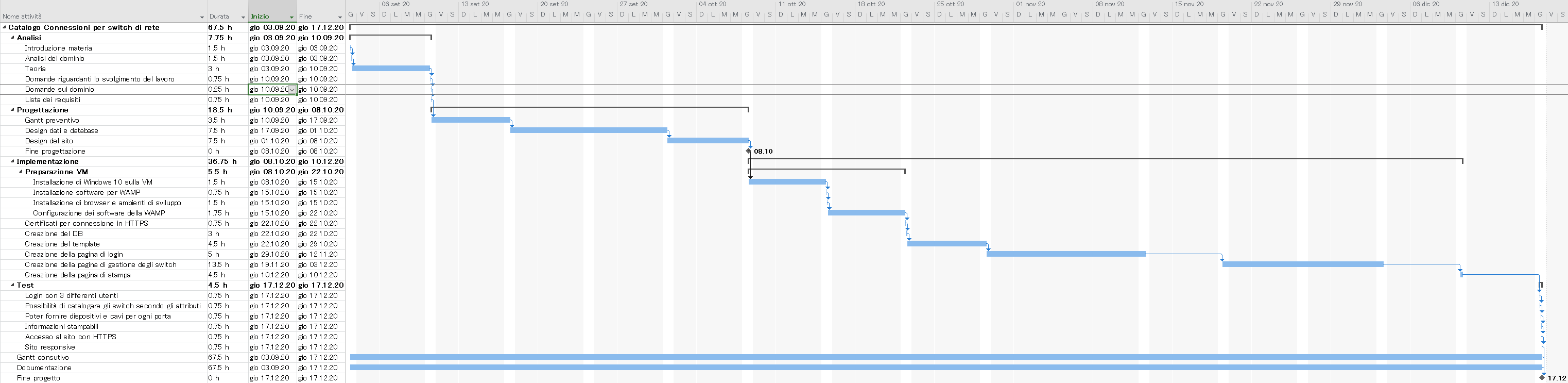
## Use case

L’applicazione si basa secondo 3 tipi di utenti, dati i 3 ruoli di Amministratore, Operatore e Viewer, l’use case si presenta con 3 attori. L’amministratore gestisce tutto ciò che sono i cavi, gli utenti e i dispositivi, più precisamente ha la facoltà di creare, modificare o eliminare qualsiasi tra le 3 cose. L’operatore invece si occupa di gestire gli switch e i collegamenti per ognuno di esso, come per l’amministratore la gestione consiste nella creazione, la modifica e l’eliminazione dei dati esistenti. Infine c’è il viewer che ha la semplice facoltà di visualizzare la lista di switch e i collegamenti per ognuno di essi, eventualmente può stampare la lista dei collegamenti di ogni singolo switch.

*Figura 1 - Use Case*

## Pianificazione





*Figura 2 - GANTT Preventivo*

## Analisi dei mezzi

Per fare il progetto, è stato dato in dotazione un solo PC, caratterizzato da medie/alte prestazioni, sul quale, si è provveduto a installare una macchina virtuale per poter lavorare su un sistema operativo pulito interamente dedicato al progetto. Su quest’ultima è stata scaricata una XAMPP per poter avere tutti gli strumenti necessari per sviluppare un sito web in un solo posto, inoltre, sono stati installati altri strumenti per essere in grado di visualizzare il codice e il risultato finale del prodotto.

### Software

Elenco di software utilizzati per la realizzazione dell’applicazione:

* Google Chrome 87.0.4280.88
* Visual Studio Code 1.51.1
* MySQL Workbech 8.0 CE 8.0.22
* Notepad++ 7.8.8
* XAMPP 7.4.11-0 con le ultime versioni disponibili degli strumenti integrati.
* Visual Studio C++ 2015-2019 32 bit 14.27.29112.0
* Visual Studio C++ 2015-2019 64 bit 14.27.29112.0

Sistema ospitante della virtual machine:

* Vmware Workstation

### Hardware

Il progetto è stato eseguito su una macchina virtuale di 2 GB di RAM, utilizzando 1 solo core del processore.

Il PC ospitante era caratterizzato da:

* 16 GB di RAM
* Intel Core i7-7700

Perciò, è consigliato di avere un PC con almeno

* 20 GB di disco fisso a disposizione
* 2 GB di RAM
* Un processore qualsiasi abbastanza performante

# Progettazione

## Design dell’architettura del sistema

### 3.1.1 Schema di rete

*Figura 3 - Schema di rete*

Il client che vuole accedere al sito, deve essere collegato alla rete del server che ospita quest’ultimo, essa può essere tipo pubblico (internet) oppure locale, è da premettere che se si vuole raggiungere da qualsiasi posto è obbligatorio metterlo online. Il progetto in questo caso è eseguito su una virtual machine ad uso locale.

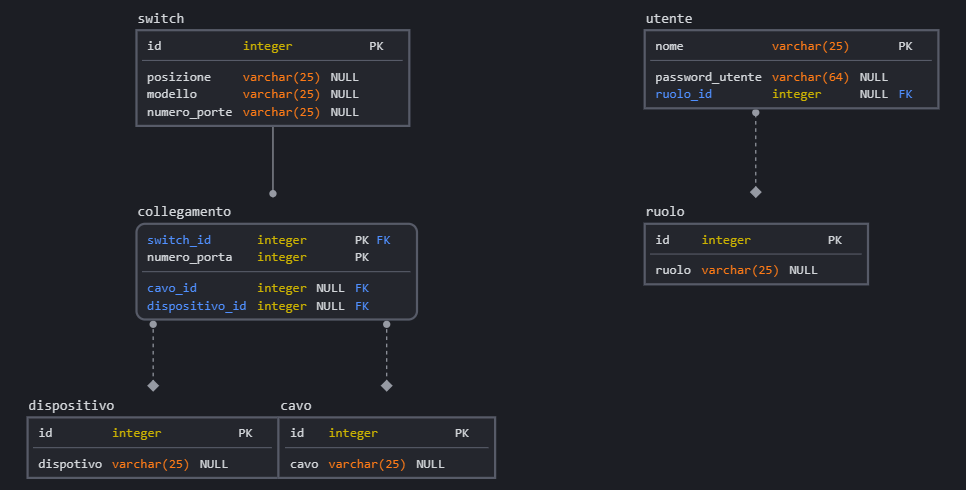
### 3.1.2 Sitemap

*Figura 4 - Sitemap*

Il sito è costruito da molte pagine, principalmente si suddivide in 3, che sono le pagine per l’amministratore, per l’operatore e per il viewer, per l’amministratore ci sono altre 3 gruppi di pagine sottostanti che raggruppano tutte le pagine per la visualizzazione, la gestione e la modifica degli utenti, dei dispositivi e dei cavi. L’operatore invece ha un gruppo di pagine per la gestione degli switch, che include le pagine per l’aggiunta, la visualizzazione e la modifica degli switch, inoltre ha le il gruppo di pagine utili alla gestione dei collegamenti per ognuno di essi. Il viewer invece ha semplicemente le pagine utili alla visualizzazione degli switch e dei relativi collegamenti.

## Design dei dati e database

### 3.2.1 Database iniziale

**

*Figura 5 - Database iniziale*

### 3.2.2 Database finale

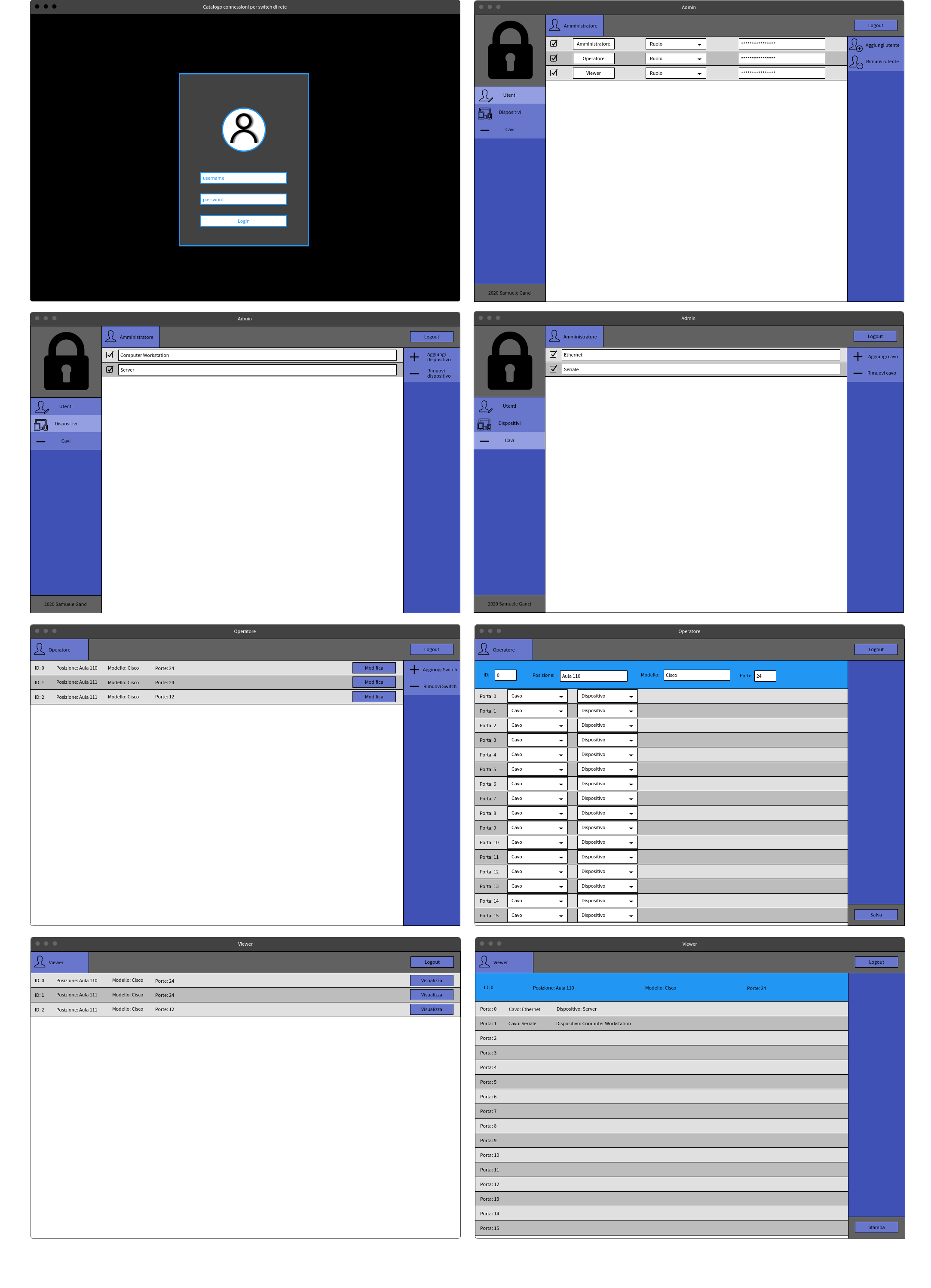
*Figura 6 - Database finale*

Rispetto al database inziale, sono state aggiunte altre opzioni alle colonne principali, ad esempio ai campi id sono state aggiunte le opzioni AUTO\_INCREMENT e a molte altre sono stati aggiunti i NOT NULL per non permettere di creare record con campi vuoti. Il campo password invece viene salvato con la cifratura MD5.

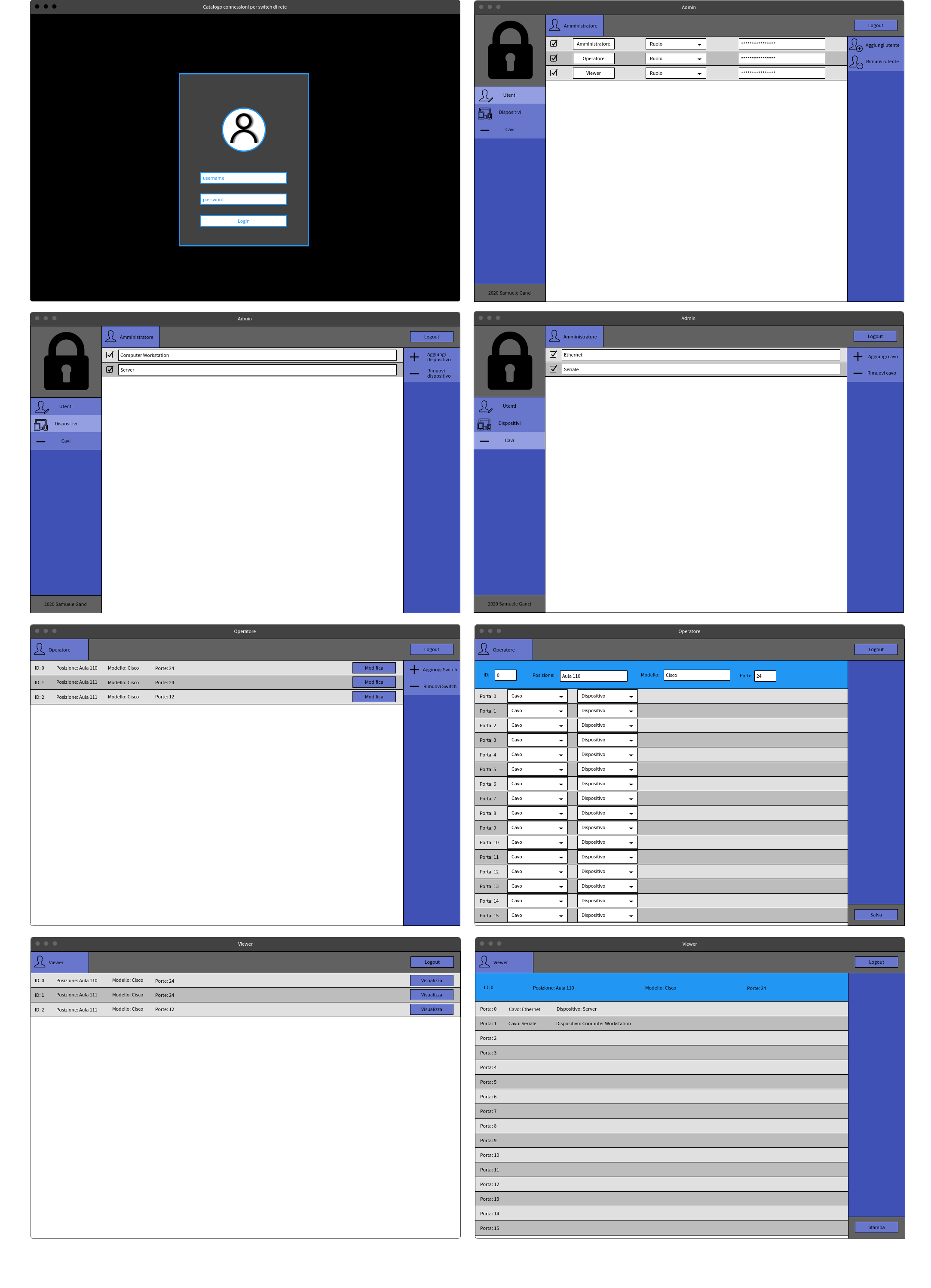
## Design delle interfacce

*Figura 7 - Pagina di login*

Semplice pagina di login, si occupa di verificare le informazioni inserite per consentire o meno l’accesso.

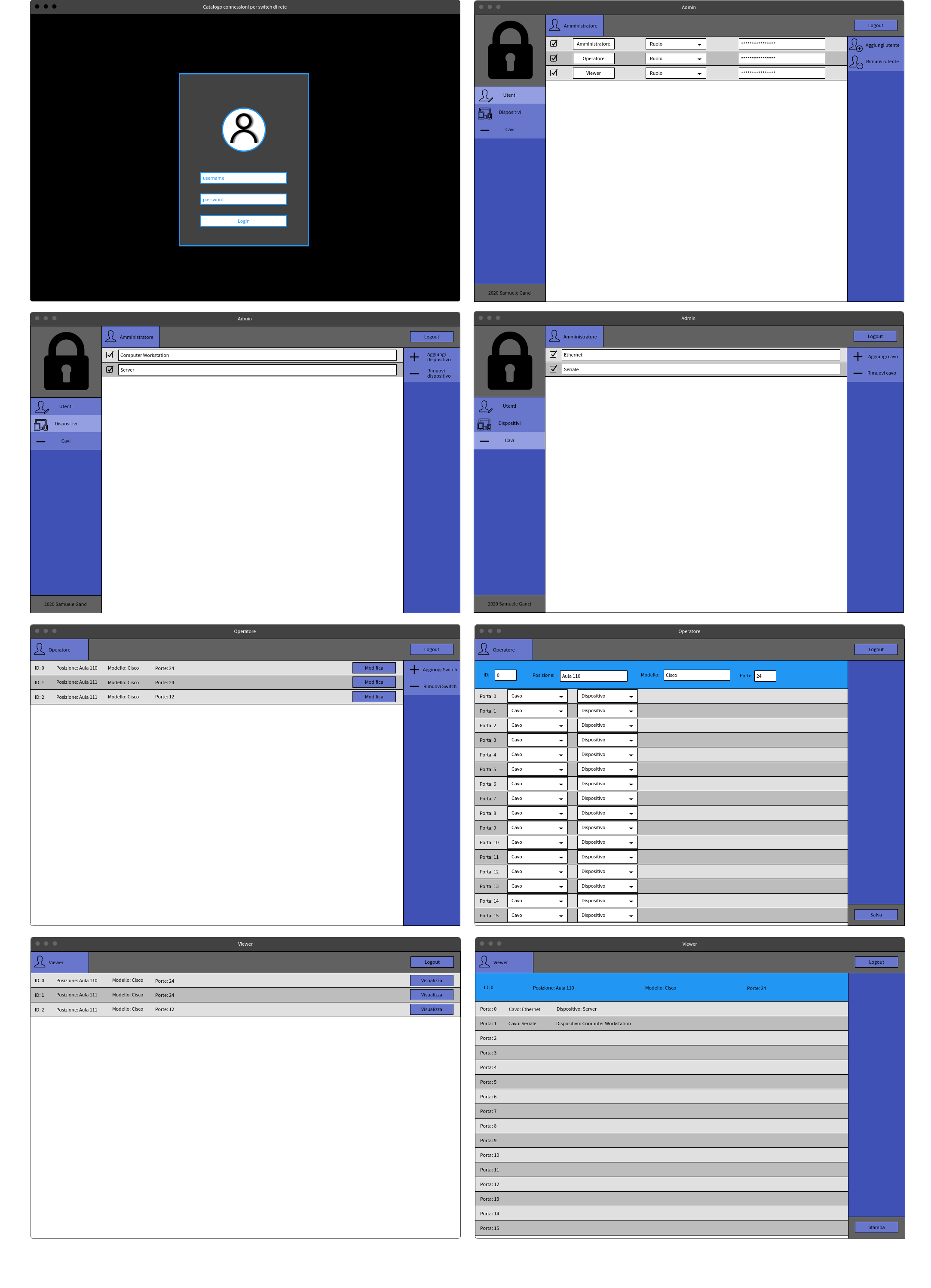
Pagina di amministratore, dal quale si possono raggiungere le pagine per gestire gli utenti, i dispositivi e i cavi, più nello specifico in questa pagina si possono gestire gli utenti.

*Figura 8 - Pagina di gestione degli utenti*



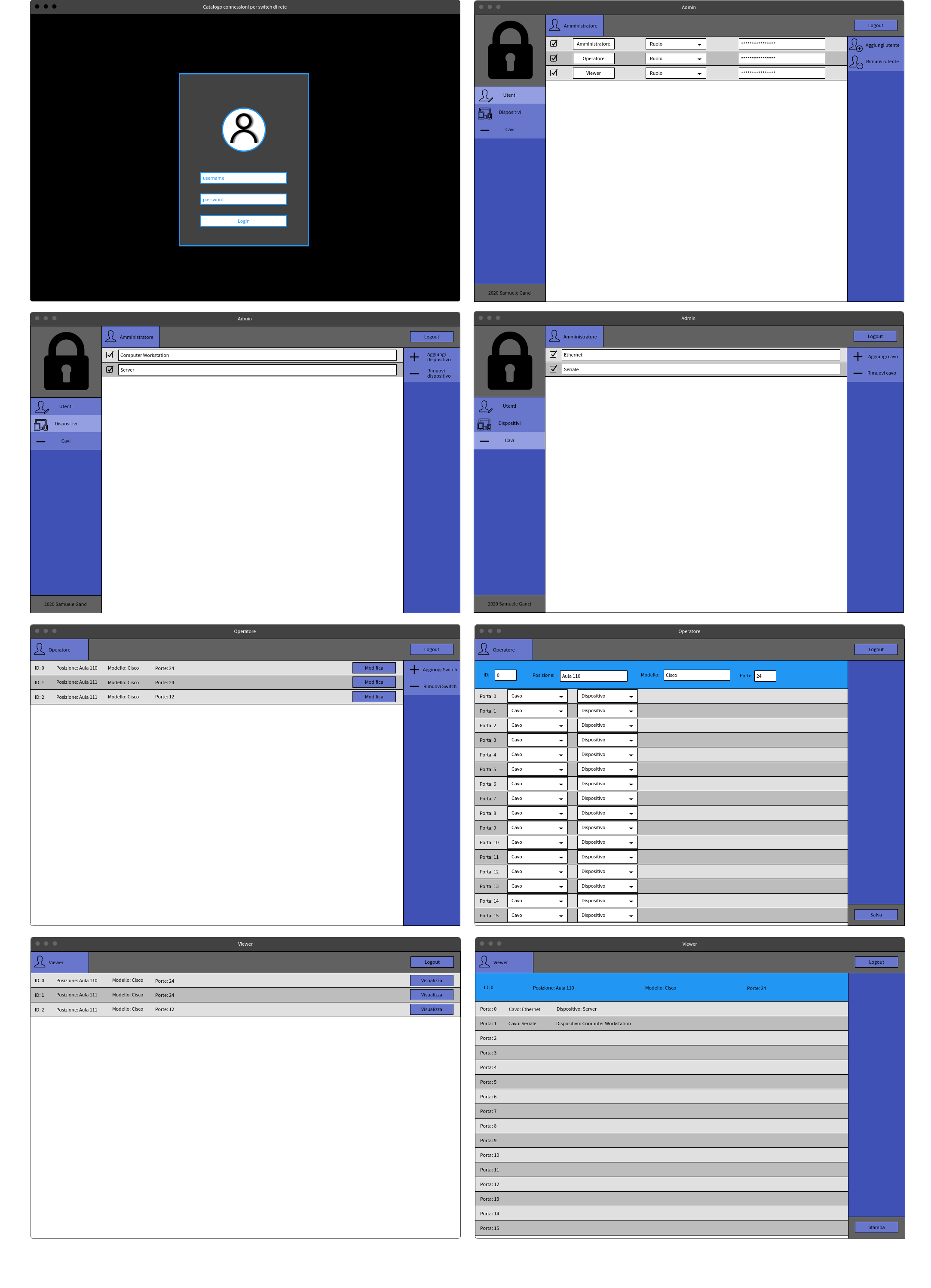
*Figura 9 - Pagina di gestione dei dispositivi*

Pagina di amministratore utile a gestire i dispositivi.

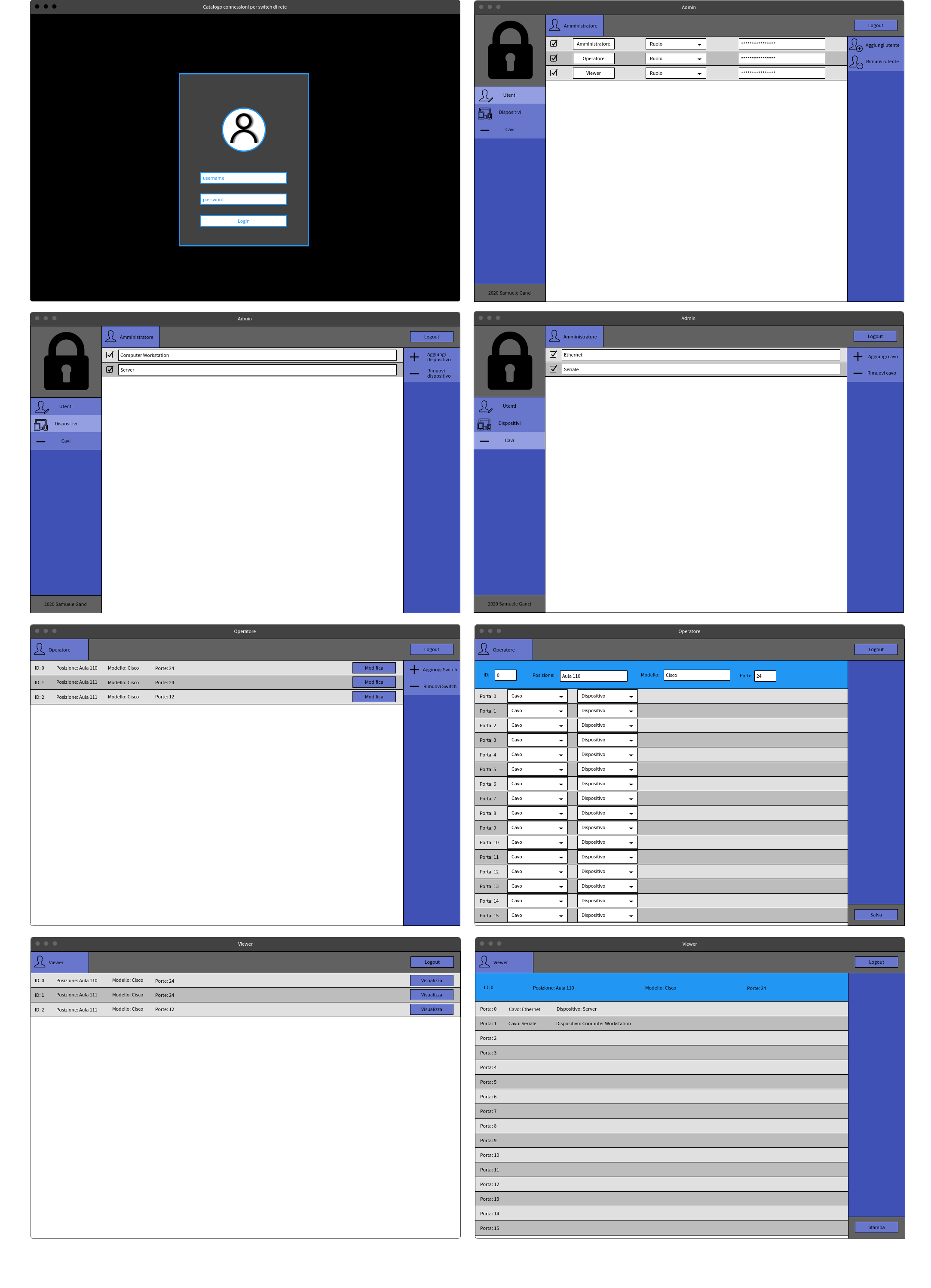


*Figura 10 - Pagina di gestione dei cavi*

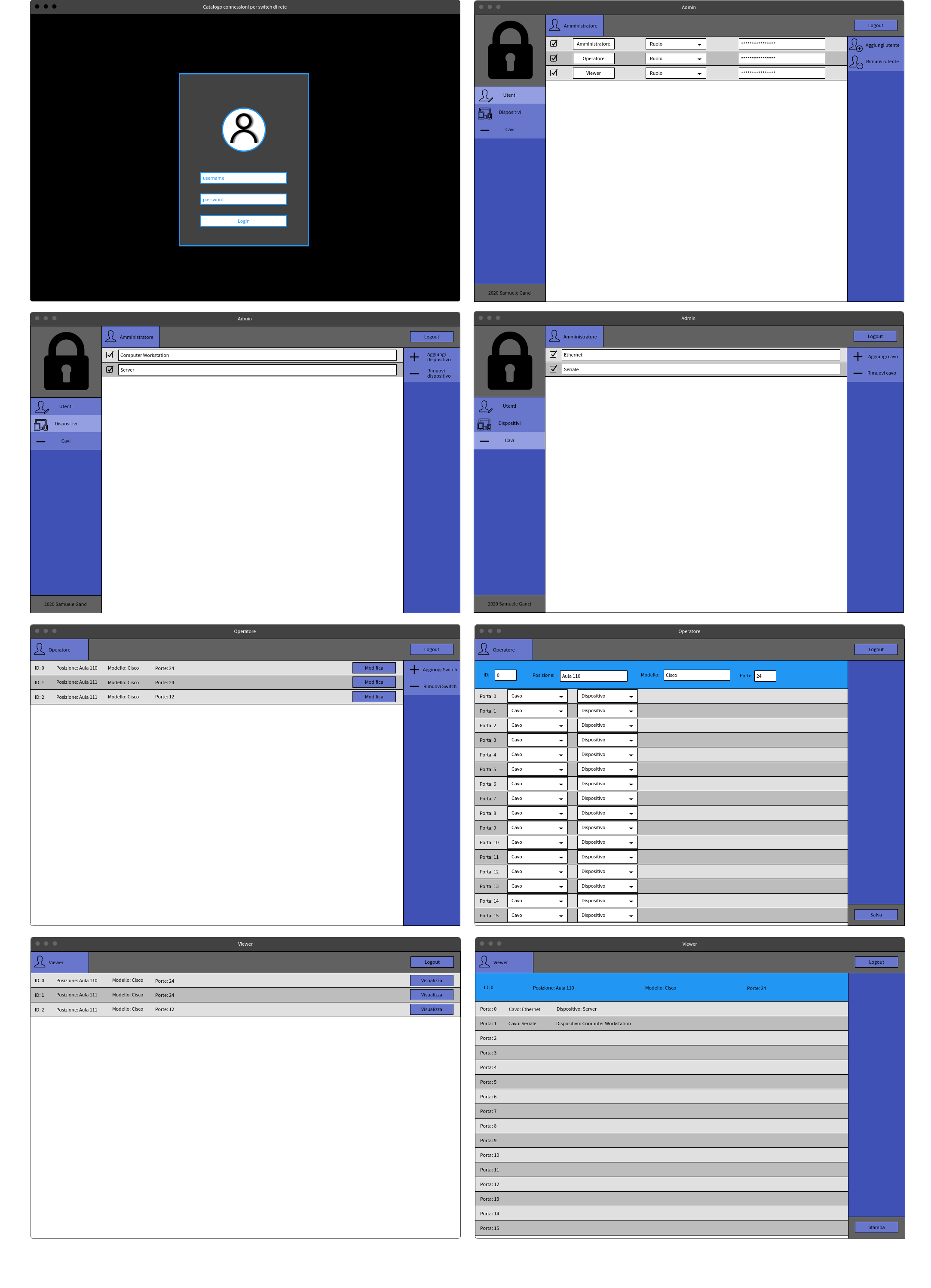
Pagina di amministratore utile a gestire i cavi.

 Pagina di gestione degli switch da parte del’operatore.

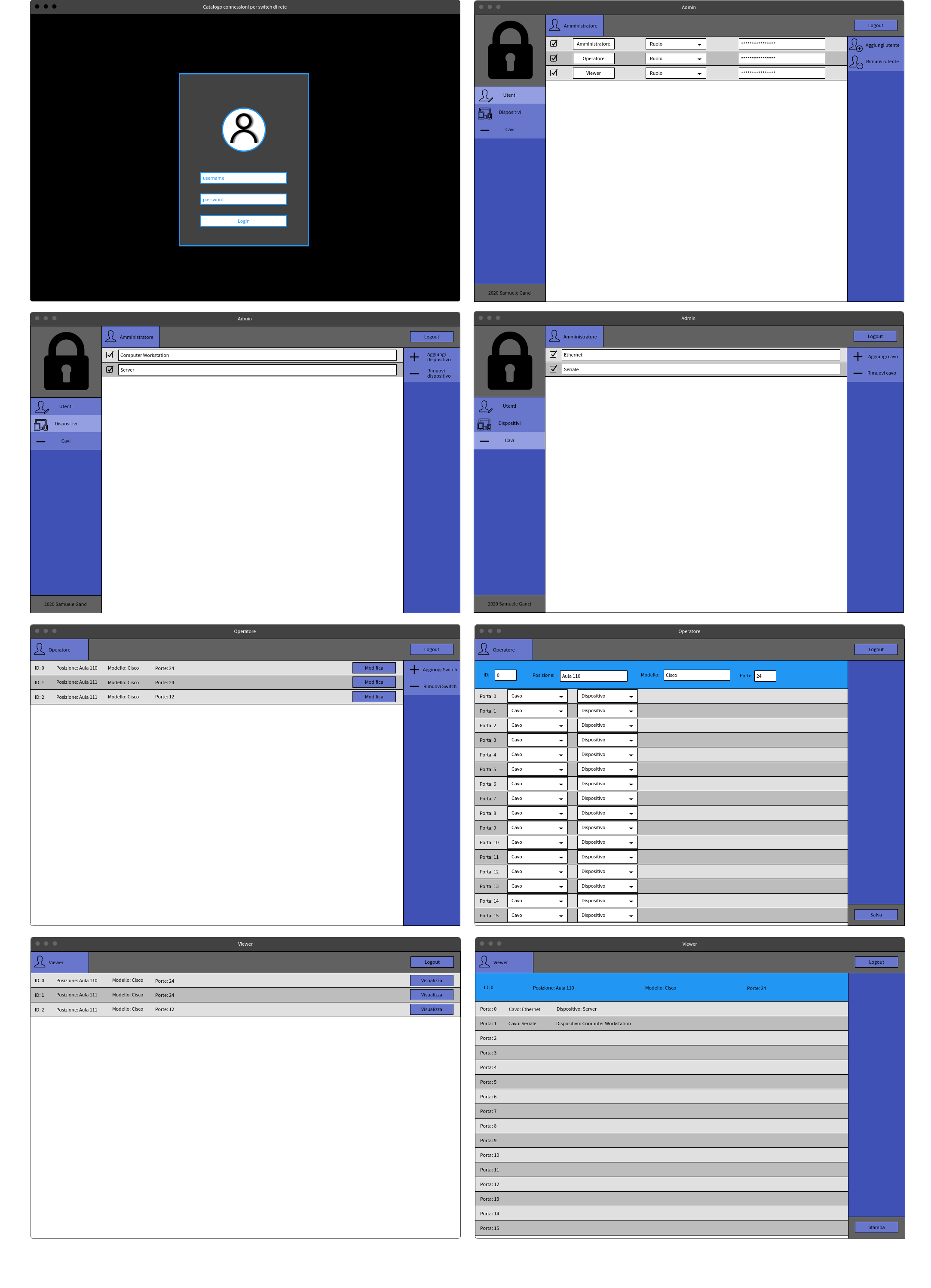
*Figura 11 - Pagina di gestione degli switch*

Pagina utile a gestire i collegamenti dello switch selezionato. Naturalmente è accessibile solo da parte dell’operatore.

*Figura 12 - Pagina di gestione dei collegamenti*

Pagina per l’utente viewer, utile a visualizzare tutti gli switch salvati all’interno dell’applicazione.

*Figura 13 - Pagina di lettura degli switch*

Questa pagina, sempre per l’utente viewer, si occupa della visualizzazione dei collegamenti per lo switch selezionato.

*Figura 14 - Pagina di lettura dei collegamenti*

## Design procedurale



*Figura 15 - Struttura MVC*

In questo progetto, il sito si baserà su una struttura MVC più precisamente Model View Controller. Ogni tipo di utente (ruolo) avrà un controller dedicato con il rispettivo file model e le relative view. In questo modo ogni utente è suddiviso e tutti quanti i file vengono gestiti meglio, senza contare che la separazione del codice può portare solo ad avere una protezione in più riguardante la sicurezza.

# Implementazione

## Prerequisiti

### VM

Come già detto in precedenza, il progetto è stato eseguito su una virtual machine, creata con il software di virtualizzazione VMware, le specifiche principali utilizzate sono state:

* 4 GB di RAM
* 2 Core
* 30 GB di Hard Disk
* Windows 10 Pro

### Software

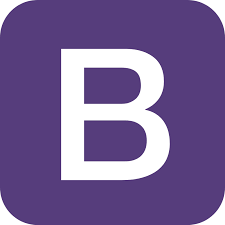
Prima di poter cominciare con la vera e propria implementazione, cominciando a digitare il codice, è stato necessario installare tutti i software necessari, in questo progetto è stato utilizzato principalmente l’editore di testo VSCode per la programmazione. Come ambiente di sviluppo invece, è stato installato il software XAMPP con tutti i valori di default e i relativi prerequisiti, inoltre, per interagire al meglio con il server di MySQL, è stato installato l’apposito software MySQL Workbench. Per l’accesso all’applicazione invece è stato installato il browser Google Chrome.

## Programmazione

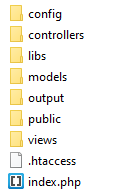
### Database

Il database ha più o meno rispettato la progettazione, ma ha subito qualche modifica durante il progetto, dapprima è stato aggiunto un campo “Etichetta” agli switch per poter tenere traccia da parte dei sistemisti delle etichette attaccate ai dispositivi, inoltre gli id sono stati modificati in AUTO\_INCREMENT, in questo modo l’applicazione non si deve preoccupare di gestire gli id

### Sito

Inizialmente, prima della scrittura del codice, ancora non sapevo che framework utilizzare, quale MVC, come funzionasse una MVC. Non avevo le conoscenze, fino a che, non mi sono state date dal modulo 133. Dopodiché, ho cominciato a programmare utilizzando la MVC fornita dal docente Sartori per la sua lezione, ho cominciato dai controller, suddividendoli in uno per rogni ruolo come da progettazione, ma rimaneva il problema delle view responsive, così dopo essermi documentato a dovere e aver chiesto al docente Petrini, ho deciso di utilizzare Bootstrap, del codice CSS e Javascript precompilato che nelle mie view ha l’onore di gestire tutto ciò che è responsive in modo autonomo e il relativo stile. Naturalmente senza l’allenamento fatto nel modulo di Php non sarei mai stato in grado di sviluppare e gestire le view e i relativi file di modifica e di aggiunta di nuovi elementi, infatti come base ho utilizzato quelle fatte in quest’ultimo, adattandole al progetto e a ciò che era richiesto.

*Figura 16 – Bootstrap logo*

Dapprima ho cominciato a sviluppare la pagina di login del sito, successivamente dopo aver fatto i relativi controlli ho cominciato a creare le cartelle che avrebbero contenuto i models, le views e i controllers di ogni tipo di utente. Fatto ciò, ho cominciato a sviluppare i controller per ognuno di essi verificando che l’utente che ci accedesse fosse del ruolo corretto, altrimenti sarebbe stato reindirizzato al controller adatto, e da qui in avanti, tutto ciò che ho fatto (views, models, metodi dei controllers), si sono basati sugli esercizi fatti nel modulo 133, facendo per ogni opzione (eliminazione, modifica, aggiunta di un record), il solito standard.

Per l’eliminazione di un record viene passato al controller l’id di ciò che deve essere eliminato, successivamente viene passato a model e quest’ultimo elimina il record di riferimento.

Per la modifica viene passato l’id su una nuova view, che permette all’utente di reinserire i dati e successivamente quest’ultimi vengono passati con metodo POST nuovamente al controller, che li controlla e li passa al model che aggiorna il record.

Figura 17 - Struttura cartelle

L’aggiunta invece, reindirizza l’utente su una nuova pagina dove può immettere i dati, dopodiché essi vengono passati con metodo POST al controller che lì tramanda al model, che grazie all’auto incremento del Database, inserisce i dati in un nuovo record senza dover gestire l’id che viene automaticamente creato dal Database.

### 4.2.3 Certificati

Nella creazione dei certificati, che sono giunti per ultimi, ho riscontrato parecchi problemi e complicanze prima del corretto funzionamento, ad esempio subito dopo la creazione il server MySQL non accennava più a partire, ed è stato necessario ripristinarlo con dei backup. Infine per creare i certificati corretti in HTTPS ho seguito la seguente guida riadattandola ai miei bisogni (Ad esempio timezone), molti campi invece li ho lasciati vuoti come indicava. <https://gist.github.com/adnan360/ad2b1cfc44114ac6f91fbb668c76798d>

## 4.3 Guida installazione

Per installare l’applicazione, è sufficiente spostare il contenuto della cartella htdocs allegata all’interno della cartella contenente i file pubblici del proprio sito. Inoltre, è necessaria la creazione del database, per farlo è sufficiente aprire MySQL Workbench sul proprio server e aprire il file gestionale.sql con esso. Una volta fatto ciò si possono modificare i parametri desiderati, ad esempio il nome del Database, il nome e password dell’utente e molto altro, ma **ricordarsi che le modifiche vanno eseguite anche al file database.php all’interno di htdocs/config.** Infine basta premere il fulmine giallo in alto a sinistra. **Se si vuole che il sito funzioni anche su un sito pubblico e non localhost, inserire l’URL all’interno del file Path.php contenuto in htdocs/config.**

**Le credenziali per eseguire l’accesso iniziale per configurare i record sono:**

**Amministratore: Password&1**

**Operatore: Password&1**

**Viewer: Password&1**

Se si vuole accedere il HTTPS a localhost, basta trascinare il contenuto della cartella certificati all’interno della propria cartella di XAMPP/Apache/conf/ssl.crt/server.crt. Altrimenti, se il sito funziona su una rete pubblica, seguire la guida riportata sopra.

# Test

## Protocollo di test

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:** | TC-001 | **Nome:** | Pagina di login |
| **Descrizione:** | Verificare che la pagina di login sia funzionante. | | |
| **Prerequisiti:** | Nessuno. | | |
| **Procedura:** | 1. Tentare di immettere dati errati o nulli.  2. Tentare il login con utente amministratore, operatore e viewer. | | |
| **Risultati attesi:** | Login sul controller giusto solo con I dati immessi correttamente. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:** | TC-002 | **Nome:** | Sito responsive |
| **Descrizione:** | Verificare che il sito sia responsive su tutte le pagine esistenti. | | |
| **Prerequisiti:** | Nessuno. | | |
| **Procedura:** | 1. Loggarsi con utente amministratore, verificare tutte le singole pagine dedicate.  2. Loggarsi con utente operatore, verificare tutte le singole pagine dedicate.  3. Loggarsi con utente viewer, verificare tutte le singole pagine dedicate. | | |
| **Risultati attesi:** | Tutte le pagine devono essere responsive. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:** | TC-003 | **Nome:** | Amministratore |
| **Descrizione:** | Verificare che tutte le pagine dell’amministratore funzionino correttamente. | | |
| **Prerequisiti:** | Login come amministratore. | | |
| **Procedura:** | 1. Creare un nuovo utente testando che tutte le combinazioni non diano errori (campi vuoti, numeri al posto di caratteri, injection ecc).  2. Modificare l’utente tentando nuovamente tutte le combinazioni.  3. Eliminare l’utente.  4. Ripetere la stessa procedura per i cavi e i dispositivi. | | |
| **Risultati attesi:** | Il sito non genera mai errori. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:** | TC-004 | **Nome:** | Operatore |
| **Descrizione:** | Verificare che tutte le pagine dell’operatore funzionino correttamente. | | |
| **Prerequisiti:** | Login come operatore. | | |
| **Procedura:** | 1. Creare due nuovi switch testando che tutte le combinazioni non diano errori(campi vuoti, numeri al posto di caratteri, injection ecc).  2. Modificare uno switch tentando nuovamente tutte le combinazioni.  3. Selezionare “Gestisci” e creare nuovi collegamenti testando tutte le combinazioni su uno switch.  3. Modificare un collegamento casuale, tentando sempre tutte le combinazioni.  4. Eliminare un collegamento casuale.  5. Eliminare lo switch con i collegamenti.  6. Eliminare lo switch senza collegamenti. | | |
| **Risultati attesi:** | Il sito non genera mai errori. | | |

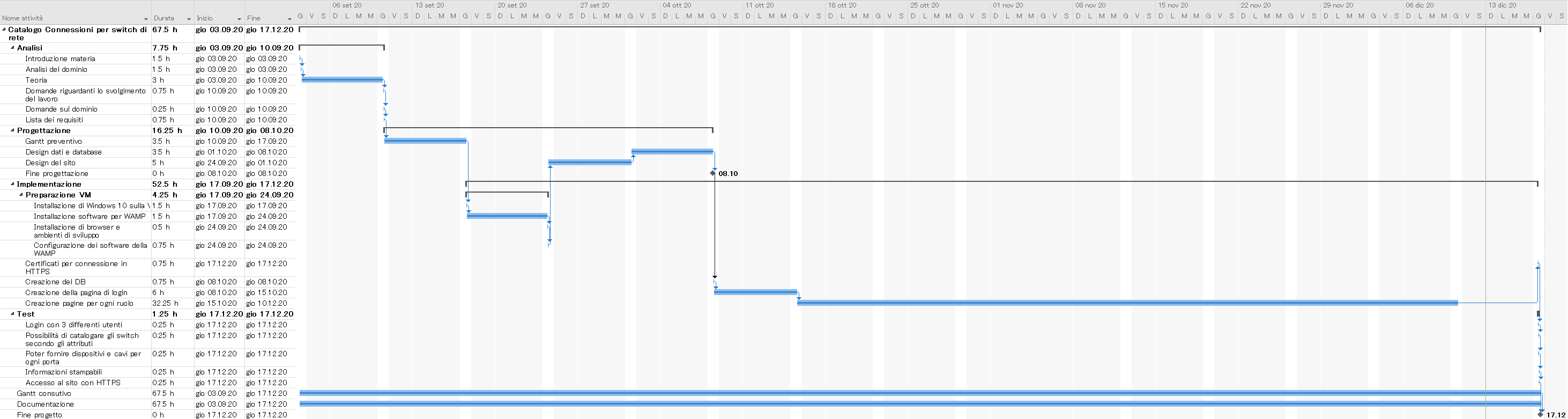
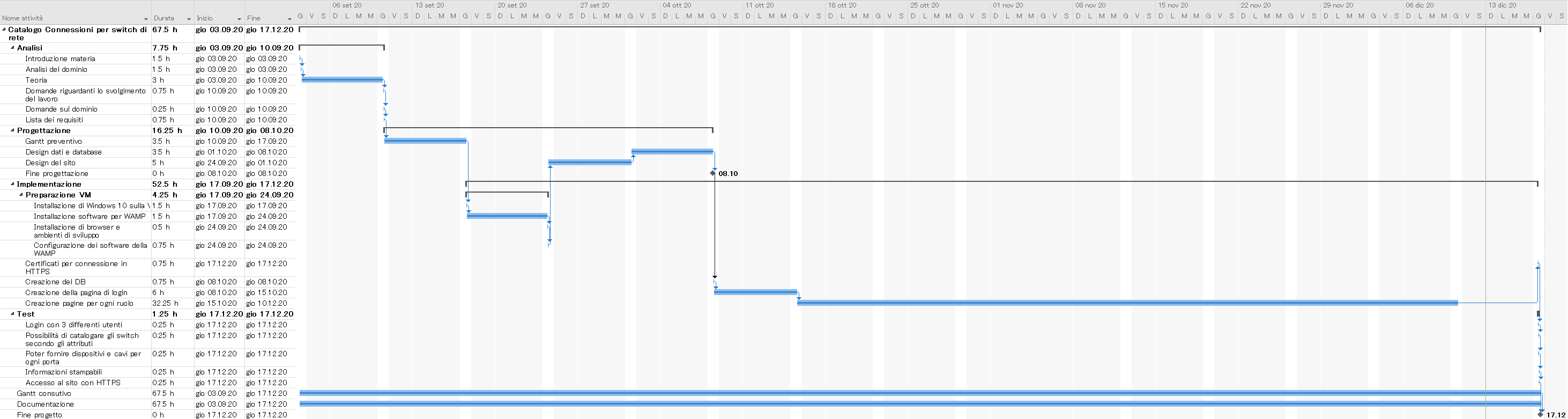
|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:** | TC-005 | **Nome:** | Viewer |
| **Descrizione:** | Verificare che tutte le pagine dell’utente viewer funzionino correttamente. | | |
| **Prerequisiti:** | Aver creato degli switch e dei collegamenti con un utente operatore. Login come utente viewer. | | |
| **Procedura:** | 1. Visualizzare gli switch.  2. Visualizzare i collegamenti di uno switch vuoto e uno con collegamenti.  4. Testare la stampa delle informazioni di ogni switch (con e senza collegamenti). | | |
| **Risultati attesi:** | Il sito non genera mai errori. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:** | TC-006 | **Nome:** | Database |
| **Descrizione:** | Verificare che lo script del database funzioni correttamente | | |
| **Prerequisiti:** | Nessuno. | | |
| **Procedura:** | 1. Aprire Workbench.  2. Connettersi a localhost.  3. Far partire lo script.  4. Verificare nuovamente tutti i test precedenti. | | |
| **Risultati attesi:** | Lo script non genera errori ed è corretto. | | |

## Risultati test

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:** | TC-001 | **Risultato:** | OK |
| **Test Case:** | TC-002 | **Risultato:** | OK |
| **Test Case:** | TC-003 | **Risultato:** | OK |
| **Test Case:** | TC-004 | **Risultato:** | OK |
| **Test Case:** | TC-005 | **Risultato:** | OK |
| **Test Case:** | TC-006 | **Risultato:** | OK |

# Consuntivo



*Figura 18 - GANTT Consuntivo*

La causa di tanta differenza tra GANTT consuntivo e preventivo, è principalmente non aver avuto gli strumenti giusti per poter fare una progettazione degna di nota, infatti, nel GANTT preventivo è stato immaginato prima di fare la progettazione, successivamente la virtual machine, eppure, si è rivelato prima necessario scaricare la virtual machine con tutti i software, in questo modo si è potuta fare una progettazione del DB dapprima con Workbench, successivamente con un sito online in quanto Workbench non era soddisfacente. Ancora prima del DB però, rispetto alla procedura organizzata in precedenza, si è rivelato ancora più utile dapprima partire dalla progettazione del sito, in questo modo personalmente sono riuscito a sviluppare meglio il DB avendo una base da cui partire (es query da eseguire a mio parere guardando le interfacce per organizzare meglio tabelle e dati). Inoltre, fare i certificati HTTPS si è rivelato molto più complicato del previsto, perciò è stato deciso di lasciarli per ultimi. A parte le procedure precedenti il resto è stato eseguito come da programma, incluso circa il tempo di ogni attività. Naturalmente, in tutto questo, l’inizio del progetto è partito a rilento in quanto al momento della commissione non avevo le conoscenze per fare un lavoro del genere.

# Conclusioni

In conclusione, il progetto è riuscito a creare quel prodotto, destinato alle persone che vogliono semplificarsi il lavoro della gestione dei propri dispositivi. Ha raggiunto gli obbiettivi di essere intuitivo da utilizzare e comodo, essendo in qualche modo “portatile”. Il prodotto finale è riuscito a rispettare tutti i requisiti richiesti, è web based, è responsive, e per salvare i propri dati fa affidamento ad un database dedicato. Infine È davvero utile e funzionale per fare le azioni per cui è stato pensato e commissionato.

## Sviluppi futuri

Attualmente non ci sono previsti sviluppi futuri, senz’altro questo lavoro sarà utile per apprendere meglio nuove conoscenze e magari riprenderlo in futuro per un nuovo progetto. Inoltre, qual’ora ci fossero errori di cui non ne conosco ancora l’esistenza, senza dubbio uno sviluppo futuro sarà la correzione di quest’ultimi.

## Considerazioni personali

Inizialmente credevo di non farcela, infatti sono rimasto qualche, lezione senza fare alcun tipo di progresso, ciò perché ero preoccupato e non riuscivo a capire come funzionavano le cose, ciò perché non avevo le conoscenze. Infatti, dopo che mi sono state accennate dalle lezioni di altre materie, ho cominciato a lavorare su qualcosa di concreto ed infine sono riuscito ad arrivare al termine, sono soddisfatto del risultato.

# Bibliografia

## 8.1 Sitografia

* https://getbootstrap.com/, *Bootstrap,* 17.12.2020
* https://stackoverflow.com/, *Stackoverflow*, 17.12.2020
* https://jquery.com/, *JQuery*, 17.12.2020
* https://popper.js.org/, P*opper*, 17.12.2020
* https://www.mockflow.com/, *Mockflow*, 17.12.2020
* https://app.sqldbm.com/OpenSample/, *SqlDBM*, 17.12.2020
* https://github.com/, *Github*, 17.12.2020

## 8.2 Indice delle figure

[Figura 1 - Use Case 5](file:///F:\SAMT\I3AC\MODULI\M306\CatalogoConnessioniRete\Repository\Documenti\Documentazione.docx#_Toc58960773)

[Figura 2 - GANTT Preventivo 1](file:///F:\SAMT\I3AC\MODULI\M306\CatalogoConnessioniRete\Repository\Documenti\Documentazione.docx#_Toc58960774)

[Figura 3 - Schema di rete 2](file:///F:\SAMT\I3AC\MODULI\M306\CatalogoConnessioniRete\Repository\Documenti\Documentazione.docx#_Toc58960775)

[Figura 4 - Sitemap 2](file:///F:\SAMT\I3AC\MODULI\M306\CatalogoConnessioniRete\Repository\Documenti\Documentazione.docx#_Toc58960776)

[Figura 5 - Database iniziale 3](#_Toc58960777)

[Figura 6 - Database finale 3](file:///F:\SAMT\I3AC\MODULI\M306\CatalogoConnessioniRete\Repository\Documenti\Documentazione.docx#_Toc58960778)

[Figura 7 - Pagina di login 4](file:///F:\SAMT\I3AC\MODULI\M306\CatalogoConnessioniRete\Repository\Documenti\Documentazione.docx#_Toc58960779)

[Figura 8 - Pagina di gestione degli utenti 5](file:///F:\SAMT\I3AC\MODULI\M306\CatalogoConnessioniRete\Repository\Documenti\Documentazione.docx#_Toc58960780)

[Figura 9 - Pagina di gestione dei dispositivi 6](#_Toc58960781)

[Figura 10 - Pagina di gestione dei cavi 7](file:///F:\SAMT\I3AC\MODULI\M306\CatalogoConnessioniRete\Repository\Documenti\Documentazione.docx#_Toc58960782)

[Figura 11 - Pagina di gestione degli switch 8](file:///F:\SAMT\I3AC\MODULI\M306\CatalogoConnessioniRete\Repository\Documenti\Documentazione.docx#_Toc58960783)

[Figura 12 - Pagina di gestione dei collegamenti 9](file:///F:\SAMT\I3AC\MODULI\M306\CatalogoConnessioniRete\Repository\Documenti\Documentazione.docx#_Toc58960784)

[Figura 13 - Pagina di lettura degli switch 10](file:///F:\SAMT\I3AC\MODULI\M306\CatalogoConnessioniRete\Repository\Documenti\Documentazione.docx#_Toc58960785)

[Figura 14 - Pagina di lettura dei collegamenti 11](file:///F:\SAMT\I3AC\MODULI\M306\CatalogoConnessioniRete\Repository\Documenti\Documentazione.docx#_Toc58960786)

[Figura 15 - Struttura MVC 12](#_Toc58960787)

[Figura 16 – Bootstrap logo 13](file:///F:\SAMT\I3AC\MODULI\M306\CatalogoConnessioniRete\Repository\Documenti\Documentazione.docx#_Toc58960788)

[Figura 17 - Struttura cartelle 14](file:///F:\SAMT\I3AC\MODULI\M306\CatalogoConnessioniRete\Repository\Documenti\Documentazione.docx#_Toc58960789)

[Figura 18 - GANTT Consuntivo 1](file:///F:\SAMT\I3AC\MODULI\M306\CatalogoConnessioniRete\Repository\Documenti\Documentazione.docx#_Toc58960790)

# Allegati

Allegati:

* Diari di lavoro
* Gantt preventivo
* Gantt consuntivo
* Schema Use Case
* Cartella con tutti i file del sito (htdocs), certificati e credenziali
* Script per la creazione del DB (gestionale.sql)
* Immagini della progettazione
* QdC