


Deduplicatore di files

Titolo del progetto: Deduplicatore di files
Alunno/a: Fadil Smajilbasic
Classe: I4AC
Anno scolastico: 2019/2020
Docente responsabile: Geo Petrini

1	Introduzione.....	3
1.1	Informazioni sul progetto.....	3
1.2	Abstract.....	3
1.3	Scopo.....	3
2	Analisi.....	4
2.1	Analisi del dominio.....	4
2.2	Analisi e specifica dei requisiti.....	4
2.4	Use casePianificazione.....	9
2.5	Analisi dei mezzi.....	10
2.5.1	Software.....	10
2.5.2	Hardware.....	10
3	Progettazione.....	10
3.1	Design dell'architettura del sistema.....	10
3.2	Design dei dati e database.....	11
3.3	Design delle interfacce.....	12
3.4	Design procedurale.....	17
4	Implementazione.....	18
5	Test.....	18
5.1	Protocollo di test.....	18
5.2	Risultati test.....	19
5.3	Mancanze/limitazioni conosciute.....	19
6	Consuntivo.....	19
7	Conclusioni.....	19
7.1	Sviluppi futuri.....	19
7.2	Considerazioni personali.....	19
8	Bibliografia.....	19
8.1	Bibliografia per articoli di riviste.....	19
8.2	Bibliografia per libri.....	19
8.3	Sitografia.....	20
9	Allegati.....	20

	SAMT – Sezione Informatica	Pagina 3 di 23
	Deduplicatore di files	

1 Introduzione

1.1 Informazioni sul progetto

Si tratta di creare un'applicazione che serve a eliminare i duplicati di file in uno o più percorsi definiti dall'utente. L'intenzione è di creare una applicazione che lavora in background e un'interfaccia grafica per permettere la configurazione all'utente. Il docente responsabile è Geo Petrini, il termine del progetto è il 20 dicembre.

1.2 Abstract

As the quantity and diversity of files created by an average user increases it becomes hard to keep track of all of the files a user has on their PC. This program has the task to simplify the search of duplicate files in order to save the users available space on disk. The project can be separated in 2 parts: the actual deduplicator that will run in the background as a service and a GUI made for the user to configure and manage the service. The program might also help the user realize how many files he has. The GUI part of the program can also be used by a SysAdmin to manage a small local network of PC's that have the service running. The scanning operation can be executed on demand or it can be set on a schedule.

1.3 Scopo

Lo scopo del progetto è di creare un programma che elimina i file duplicati e che ha la possibilità di essere eseguito su multiple piattaforme (Windows/Linux/macOS). Il progetto sarà diviso in 2 parti: La parte del servizio che lavora in background e la parte della GUI dove verrà eseguita la configurazione del servizio da parte dell'utente.

2 Analisi

2.1 Analisi del dominio

Attualmente sul mercato esistono dei tool per la deduplicazione, i più conosciuti sono: *CloneSpy*, *Duplicate Cleaner Pro/Free*, *Dupscout*, *Advanced Duplicates Finder*, *Duplicate Finder*, *Auslogistics Duplicate File Finder*, *Fast Duplicate File Finder*, *Anti-Duplicate* e altri, ma la maggior parte di loro è a pagamento oppure non si adegua ai requisiti imposti, cioè lavorare in background e essere configurabili da remoto.

Il programma deve inoltre poter essere utilizzato sia per scopo personale che per scopo di piccole aziende dove un sysadmin gestisce una piccola rete.

Per usare il prodotto l'utente avrà bisogno di conoscenze minime su come utilizzare un pc.

2.2 Analisi e specifica dei requisiti

I requisiti per questo progetto sono stati definiti dal docente responsabile Geo Petri.

Il programma sarà suddiviso in 2 parti: la parte del servizio che accetterà i comandi e che infine eseguirà le scansioni e la parte della GUI che sarà utile all'utente per configurare e usare la parte del servizio. Il servizio deve eseguire la scansione in modo Multithreaded basandosi su un file di configurazione per sapere che percorsi scansionare. Il servizio alla fine della scansione deve produrre un rapporto tramite quale l'utente potrà visualizzare i file duplicati e decidere cosa fare. Non è prevista una gestione degli accessi / utenti al servizio, anche se in futuro potrebbe essere aggiunta. La comunicazione tra il servizio e la GUI deve avvenire in modo sicuro utilizzando HTTPS con autenticazione.

ID: REQ-01	
Nome	Comunicazione sicura
Priorità	1
Versione	1.0
Note	La comunicazione tra il servizio e la GUI deve avvenire in modo sicuro tramite richieste HTTPS + autenticazione
Sotto requisiti	
001	Si necessita un webserver HTTPS dal lato del servizio
002	Si necessita un coppia di chiave pubblica e privata per l'autenticazione

ID: REQ-02	
Nome	Comunicazione usando il protocollo REST
Priorità	1
Versione	1.0
Note	La comunicazione tra il servizio e la gui deve avvenire usando il protocollo REST
Sotto requisiti	
001	Si necessita di un approccio MVC

ID: REQ-03	
Nome	Creazione del rapporto
Priorità	1
Versione	1.0
Note	Alla fine della scansione il programma deve generare un rapporto

ID: REQ-04	
Nome	L'esecuzione delle azioni
Priorità	1
Versione	1.0
Note	Questo requisito dipende dal requisito REQ-04, perché solo dopo che un rapporto sia stato generato si possono impostare le azioni da eseguire per i file duplicati trovati. Si avrebbe anche bisogno dei permessi di root, a dipendenza dei percorsi, per poter eseguire alcune operazioni
Sotto requisiti	
001	La possibilità di eseguire le azioni programmaticamente
002	La messa in coda delle azioni da eseguire

ID: REQ-05	
Nome	GUI per utente per il controllo
Priorità	1
Versione	1.0
Note	Creare una GUI per l'utente
Sotto requisiti	
001	Possibilità di inserire i percorsi
002	Possibilità di inserire dei percorsi da escludere (whitelist)
003	Possibilità di gestire le scansioni in corso (vedi REQ-07)

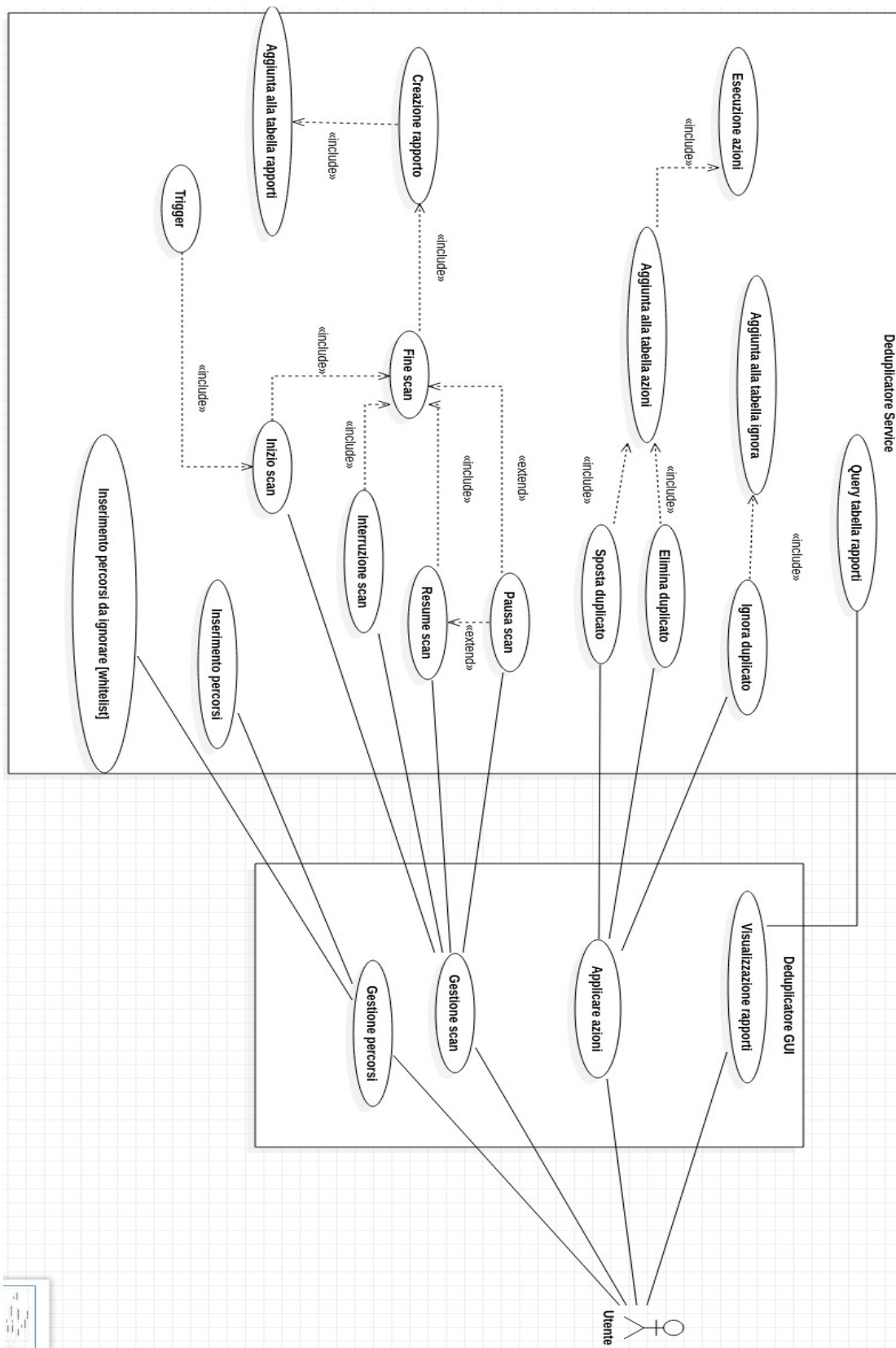
ID: REQ-06	
Nome	Messa in pausa della scansione
Priorità	2
Versione	1.0
Note	La possibilità di fermare o mettere in pausa una scansione.

ID: REQ-07	
Nome	Gestione dei rapporti
Priorità	2
Versione	1.0
Note	La possibilità di salvare, ricaricare un rapporto passato e la continuazione di scansione di un rapporto non finito.

ID: REQ-08	
Nome	Un scheduler delle scansioni
Priorità	2
Versione	1.0
Note	La possibilità di pianificare l'esecuzione delle scansioni.

ID: REQ-09	
Nome	Rilevazione in tempo reale
Priorità	3
Versione	1.0
Note	Usare i trigger di sistema per trovare nuovi duplicati in tempo reale.

2.3 Use case



2.4 Pianificazione

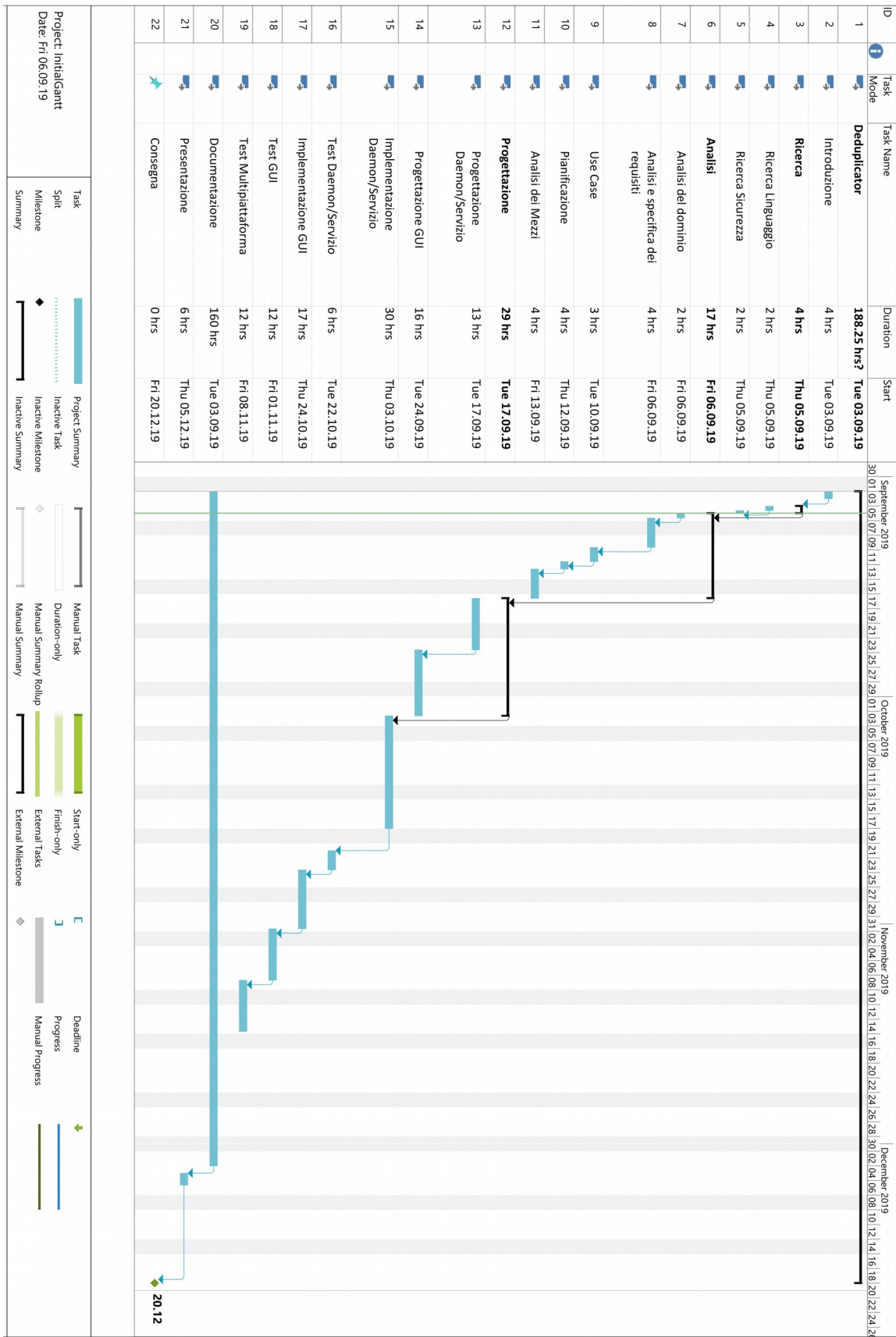



Figura 1: Gantt preventivo

	SAMT – Sezione Informatica	Pagina 10 di 23
	Deduplicatore di files	

2.5 Analisi dei mezzi

Per la creazione di questo progetto la scuola mi mette a disposizione tutti i tool che sono disponibili a scuola e 1 accesso presso l'hosting interno per caricare il progetto.

2.5.1 Software

Per lo sviluppo userò il framework Spring Boot 2.1.8, MySQL 8.0.17, VSCode 1.38.0.

Per i test Postman 7.8.0.

Il framework che utilizzerò dipende da Java 11 e Gradle 4.4.1 per la compilazione del progetto
Star UML

2.5.2 Hardware

Per lo sviluppo verrà utilizzato il mio portatile personale che ha le seguenti specifiche:

HP Pavilion 15-0800nz

CPU: i7-8550U

RAM: 16 GB DDR4

GPU: Intel UHD Graphics 620

OS: Ubuntu 18.04.3 LTS / Gnome 3.28.2

3 Progettazione

3.1 Design dell'architettura del sistema

TODO: Fare UML

3.2 Design dei dati e database

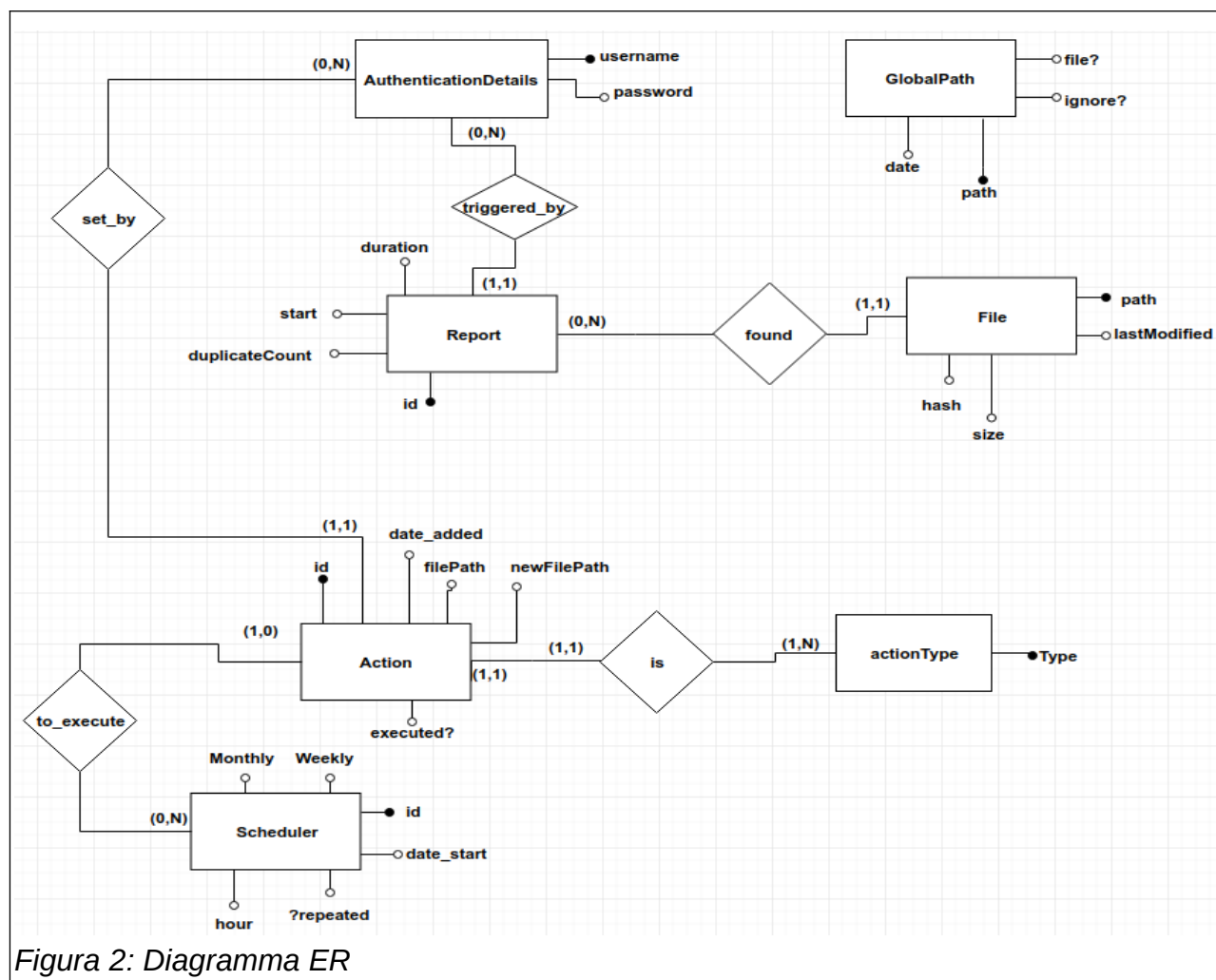


Figura 2: Diagramma ER

Il diagramma ER è fatto da 7 entità e 4 relazioni.

L'entità GlobalPath conterrà i percorsi da scansionare o da ignorare che il servizio userà durante la scansione.

L'entità Report contiene informazioni sui report creati alla fine della scansione, contiene il numero di duplicati,

da chi è stato eseguito (utente o scheduler), il timestamp di quando è stato eseguito e un id.

L'entità File verrà usata per salvare informazioni sui file scansionati, contiene il percorso con nome del file, la data dell'ultima modifica, il hash in MD5 (32 Byte) del file e la grandezza del file. Inoltre contiene l'informazione in quale report è stato scoperto il file.

L'entità ActionType contiene il tipo di azione che verrà applicata ai file (ignora, elimina, sposta)

L'entità Action contiene le azioni da eseguire dopo che l'utente abbia revisionato un report, ogni azione ha un id, una data d'aggiunta, il percorso del file, se l'azione è stata eseguita e dopo essere eseguita viene aggiunto un riferimento segnalando quale schedule nella tabella Scheduler abbia fatto eseguire l'azione e l'utente che ha scelto quella azione.

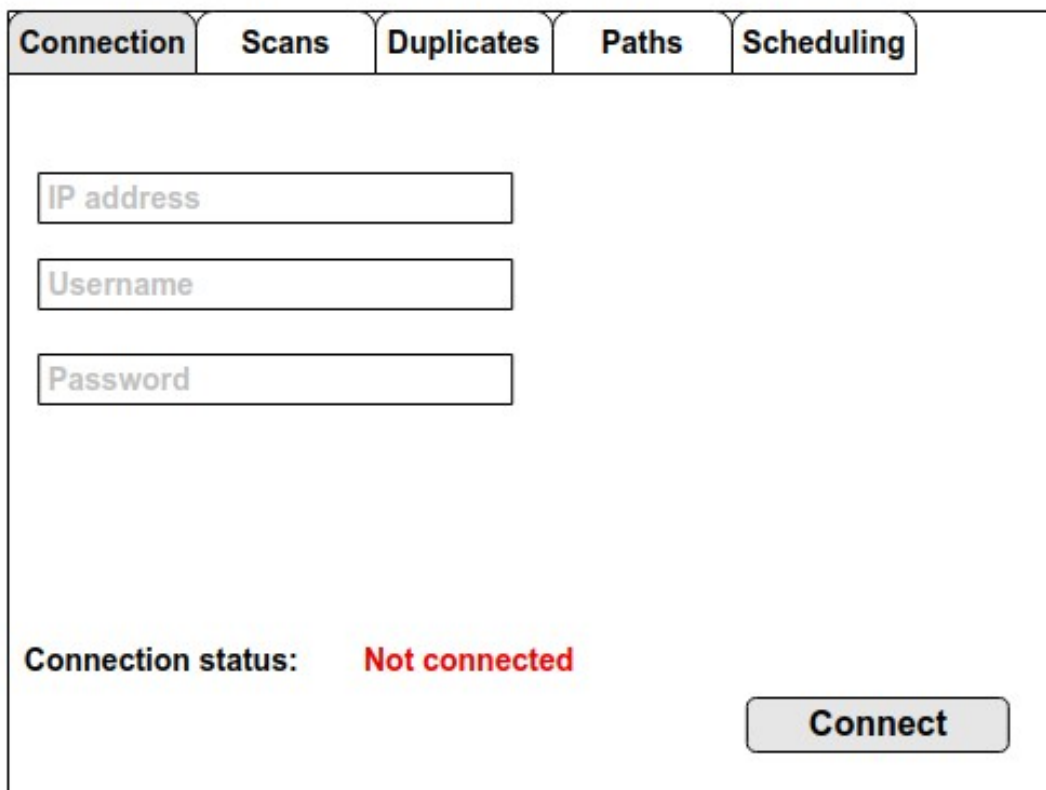
L'entità Scheduler contiene informazioni sulle operazioni programmate che dovranno essere eseguite, verrà utilizzato da una parte del servizio per controllare quali schedule sono da eseguire e quali no.

La tabella AuthenticationDetails contiene il username e la password per poter verificare le credenziali al momento della creazione della connessione.

Report contiene informazioni sulla scansione, quando è stata avviata, quanto è durata, i duplicati trovati e da chi è stata avviata, nel caso che essa venga avviata tramite lo scheduler, l'utente che l'ha avviata verrà impostato come quello di default.

3.3 Design delle interfacce

In seguito saranno rappresentati i mockup delle interfacce della GUI, il servizio non ha alcuna interfaccia utente.



Connection	Scans	Duplicates	Paths	Scheduling
<div style="margin-bottom: 10px;"> <input style="width: 100%;" type="text" value="IP address"/> </div> <div style="margin-bottom: 10px;"> <input style="width: 100%;" type="text" value="Username"/> </div> <div style="margin-bottom: 10px;"> <input style="width: 100%;" type="password" value="Password"/> </div> <div style="display: flex; justify-content: space-between; align-items: center; margin-top: 20px;"> <div> Connection status: Not connected </div> <div> <input style="border: 1px solid black; padding: 5px 15px;" type="button" value="Connect"/> </div> </div>				

Figura 3: L'interfaccia della Connessione

Connection	Scans	Duplicates	Paths	Scheduling
Scan in progress: True Objects scanned: 8284 Scan started: 17.09.2019 14:35				<input type="button" value="Refresh"/> <input type="button" value="Start scan"/> <input type="button" value="Pause scan"/> <input type="button" value="Stop scan"/>
Details: <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <pre> /var/lib/dpkg/info/xawtv-plugins.md5sums /var/lib/dpkg/info/xdg-utils.md5sums /var/lib/dpkg/info/libtheora0:amd64.triggers /var/lib/dpkg/info/libusbmuxd4:amd64.triggers /var/lib/dpkg/info/libharfbuzz-icu0:amd64.symbols </pre> </div>				

Figura 4: L'interfaccia che contiene le informazioni sulle scansioni

Connection

Scans

Duplicates

Paths

Scheduling

Scan 15.09.2019 11:30

info

Duplicates found: 12 Total files: 29

Duplicate 1:

Hash: 7ead3c169f4b3b2dac7ab45e6c32f6e3 Size: 19KB

Ignore all

▼ Path	▼ Name	▼ Last modified	▼ Action
/home/john/Documents/	filename.txt	11-09-2019	Delete ▼
/home/john/	nicknames.txt	04-05-2019	Ignore ▼
/home/john/Pictures/emails/attachments/	filename.txt	26-08-2019	Delete ▼
/home/john/randomFiles/	todo.txt	31-08-2019	Move ▼

Duplicate 2:

Hash: 20905d243ff3538d2082b4443ede9998 Size: 12KB

Ignore all

▼ Path	▼ Name	▼ Last modified	▼ Action
/home/john/Documents/	another.txt	08-07-2019	Delete ▼
/home/john/	dummy1.txt	05-04-2019	Move ▼

Apply date:

12/10/2019

Apply apply hour:

2:00

Apply

Figura 5: Interfaccia contenente i rapporti dei scan

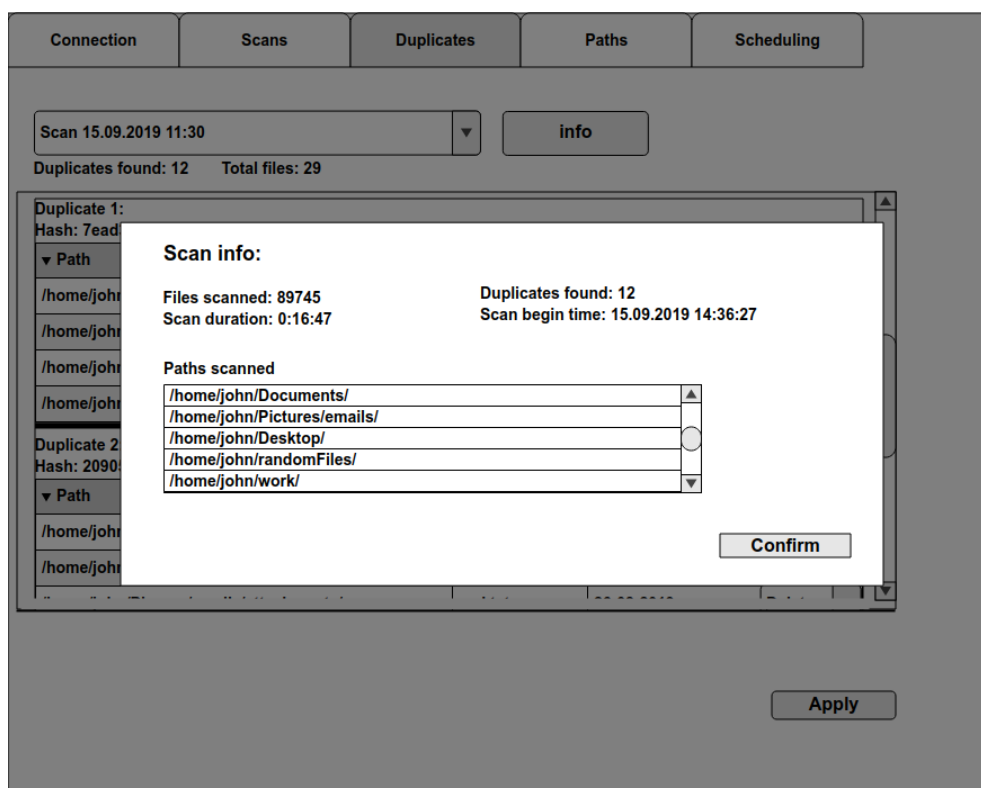


Figura 6: Interfaccia che si vede quando si clicca il bottone info

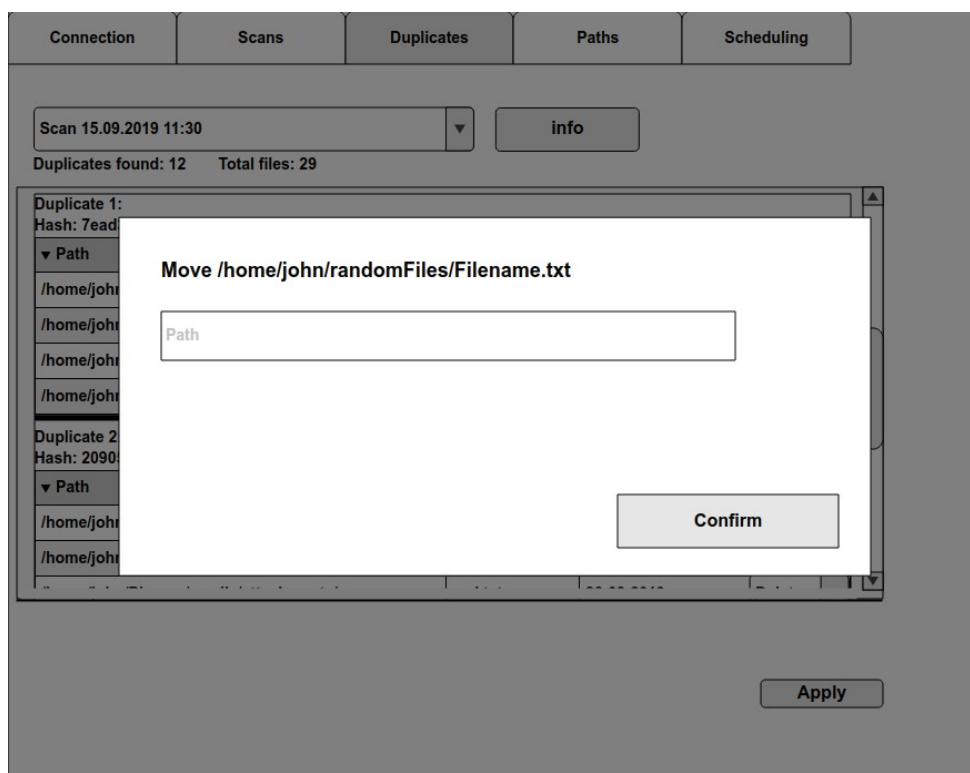


Figura 7: Interfaccia che si vede quando si sceglie di muovere un file

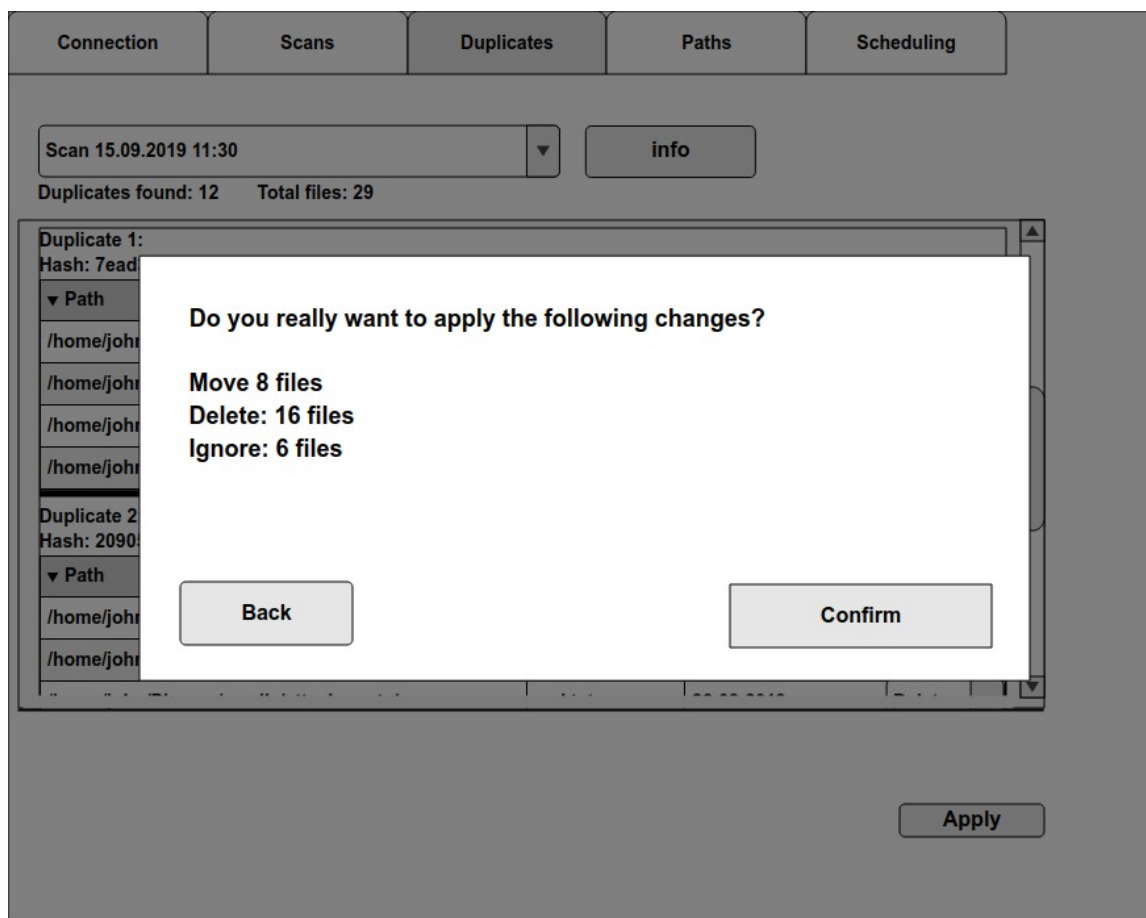


Figura 8: Il riassunto delle operazioni, quesat interfaccia è visibile quando si schiaccia il tasto apply

The 'Paths' tab is active. It contains a form with a 'Path' input field, an 'Ignore' dropdown menu, and an 'Insert' button. Below this is a table titled 'Paths:'.

/home/John/Documents/Backup/Filename.pdf	Ignore
/home/John/Documents/Backup/OtherFile.odg	Ignore
/home/John/Documents/Backup/BigFile.odg	Ignore
/home/John/Documents/A_Folder/	Scan
/home/John/Documents/BigFolder	Ignore
/home/John/Documents/	Scan

Figura 9: L'interfaccia contenete i percorsi da ignorare o da scansionare

The 'Scheduling' tab is active. It contains a 'Plan new scan:' section with a 'Scan start:' date and time picker. The date is 4/22/2012 and the time is 12:00. There are radio buttons for 'One off', 'Daily', 'Weekly' (selected), and 'Monthly'. Below this are checkboxes for days of the week: Monday (unchecked), Tuesday (checked), Wednesday (unchecked), Thursday (checked), Friday (unchecked), Saturday (checked), and Sunday (checked). A 'Confirm' button is at the bottom right.

Figura 10: L'interfaccia contenente le informazioni sulle scansioni pianificate

3.4 Design procedurale

Nel seguente diagramma di flusso si può vedere come lavorerà la parte del servizio che sta dietro alla GUI

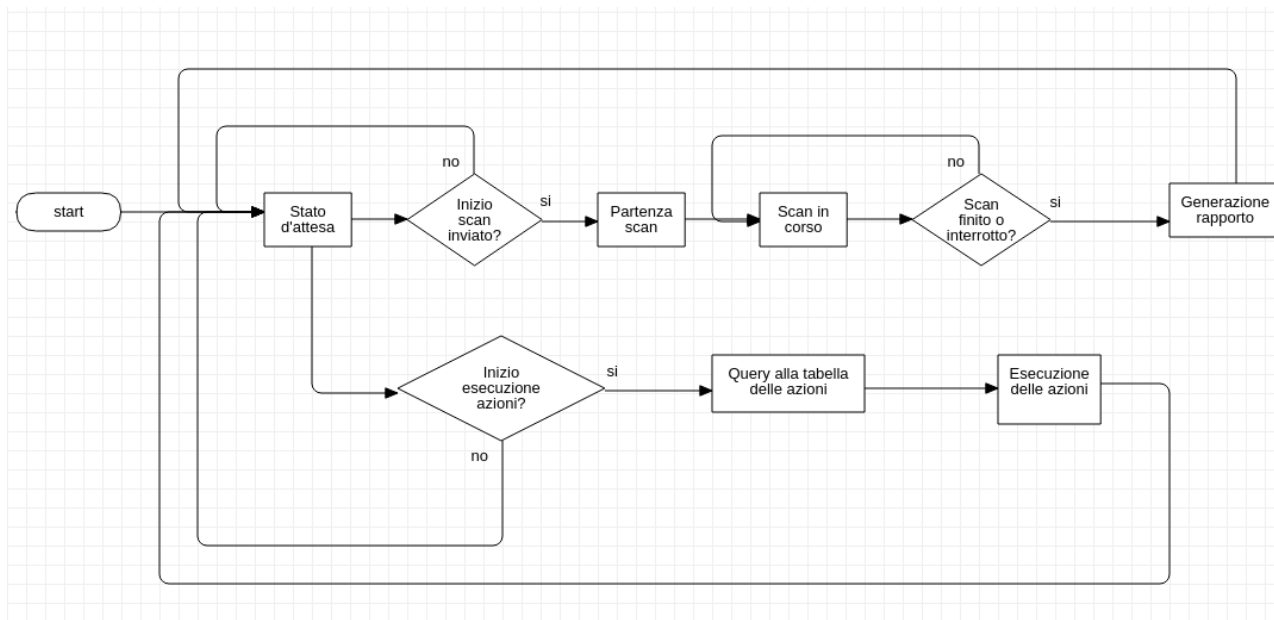


Figura 11: Diagramma di flusso servizio

4 Implementazione

In questo capitolo dovrà essere mostrato come è stato realizzato il lavoro. Questa parte può differenziarsi dalla progettazione in quanto il risultato ottenuto non per forza può essere come era stato progettato.

Sulla base di queste informazioni il lavoro svolto dovrà essere riproducibile.

In questa parte è richiesto l'inserimento di codice sorgente/print screen di maschere solamente per quei passaggi particolarmente significativi e/o critici.

Inoltre dovranno essere descritte eventuali varianti di soluzione o scelte di prodotti con motivazione delle scelte.

Non deve apparire nessuna forma di guida d'uso di librerie o di componenti utilizzati.

Eventualmente questa va allegata.

Per eventuali dettagli si possono inserire riferimenti ai diari.

5 Test

5.1 Protocollo di test

Tutti i test sono stati eseguiti tramite il client GUI del progetto.

Test Case:	TC-001	Nome:	La creazione dei rapporti.
Riferimento:	REQ-003		
Descrizione:	Far partire una scansione e controllare la generazione del rapporto per quella scansione.		
Prerequisiti:	<ul style="list-style-type: none"> • Il servizio deduplicator è in esecuzione • Eseguire il login tramite la GUI. • Impostare un percorso da scansionare. • La scansione sia finita 		
Procedura:	<ol style="list-style-type: none"> 1. Selezionare il tab duplicates 2. Selezionare un rapporto dal dropdown menu che si trova in alto a sinistra 3. Selezionare un duplicato (se ci sono) 		
Risultati attesi:	Una lista di file sotto i dropdown menu sui quali è possibile impostare un'azione da eseguire.		

Test Case: Riferimento:	TC-002 REQ-005	Nome:	L'esecuzione delle azioni impostate
Descrizione:	Verificare l'esecuzione delle azioni impostate alla revisione di un rapporto di una scansione.		
Prerequisiti:	<ul style="list-style-type: none"> Il servizio deduplicator è in esecuzione Esiste un rapporto 		
Procedura:	<ol style="list-style-type: none"> Selezionare il tab duplicates Selezionare un rapporto dal dropdown menu che si trova in alto a sinistra Selezionare un duplicato Selezionare un file dalla lista Scegliere l'opzione elimina Impostare la data e ora d'esecuzione Schiacciare il tasto apply Aspettare la data e ora d'esecuzione e poi rifare il scan Controllare la presenza del file eliminato nel nuovo scan 		
Risultati attesi:	Il file non deve essere più presente nella lista dei duplicati perché sarà stato eliminato dall'azione impostata.		

Test Case: Riferimento:	TC-003 REQ-007	Nome:	Messa in pausa della scansione
Descrizione:	Avere la possibilità fermare l'esecuzione di una scansione in corso.		
Prerequisiti:	<ul style="list-style-type: none"> Il servizio deduplicator è in esecuzione Almeno un percorso è inserito per la scansione 		
Procedura:	<ol style="list-style-type: none"> Cliccare il tab scans Avviare una scansione schiacciando il tasto Start Scan Dopo un paio di secondi schiacciare il tasto Pause Scan Controllare sul output del server che l'esecuzione si è fermata Schiacciare il tasto Resume Scan per riprendere la scansione 		
Risultati attesi:	Tutte le thread di scansione si fermano quando viene schiacciato il tasto Pause Scan, Schiacciando il tasto Resume Scan tutte le thread di scansione riprendono con la scansione.		

Test Case: Riferimento:	TC-004 REQ-008	Nome:	Gestione dei rapporti
Descrizione:	La possibilità di vedere i rapporti passati		
Prerequisiti:	<ul style="list-style-type: none"> • Il servizio è in esecuzione • Eseguire il login tramite la GUI. • Eseguire un paio di scansioni 		
Procedura:	<ol style="list-style-type: none"> 1. Selezionare il tab Duplicates e verificare che tutti i rapporti sono presenti nel dropdown menu 2. Selezionare i rapporti uno a uno e verificare i dati con il tasto info. 		
Risultati attesi:	Tutti i rapporti sono presenti e navigabili tramite il dropdown menu. I duplicati che si trovano in un rapporto vecchio, verranno spostati automaticamente sul rapporto più recente per avere le informazioni sui duplicati più recenti.		

Test Case: Riferimento:	TC-005 REQ-009	Nome:	Scheduler delle scansioni
Descrizione:	<p>La possibilità di impostare una scansione pianificata.</p> <p>Questo test è da eseguire con Postman oppure un altro tool per fare richieste HTTP/HTTPS perché non è stata implementata questa funziona nella GUI.</p>		
Prerequisiti:	<ul style="list-style-type: none"> • Il servizio è in esecuzione • I certificati sono impostati in Postman 		
Procedura:	<p>Fare una richiesta PUT sull'indirizzo <ip server>/scheduler/ con i parametri nel body (form-data)</p> <pre>monthly:null weekly:null timeStart:<data e ora d'inizio in formato timestamp> (possibilmente 5 min dall'ora attuale) repeated:false</pre> <p>inviare la richiesta e verificare che la risposta sia una con il header 200 OK</p>		
Risultati attesi:	Osservare l'output nel server e dopo 5 min dall'invio della richiesta dovrebbe partire una nuova scansione.		

5.2 Risultati test

Requisito	Soddisfatto Si/No	Note
REQ-01	Si	Autenticazione BASIC + HTTPS con certificato self-signed
REQ-02	Si	
REQ-03		
REQ-04		
REQ-05		
REQ-06		
REQ-07		
REQ-08		
REQ-09		

TEST	Riuscito Si/No	Note
TC-001		
TC-002		
TC-003		
TC-004		
TC-005		

5.3 Mancanze/limitazioni conosciute

Descrizione con motivazione di eventuali elementi mancanti o non completamente implementati, al di fuori dei test case. Non devono essere riportati gli errori e i problemi riscontrati e poi risolti durante il progetto.

6 Consuntivo

Consuntivo del tempo di lavoro effettivo e considerazioni riguardo le differenze rispetto alla pianificazione (cap 1.7) (ad esempio Gantt consuntivo).

7 Conclusioni

Quali sono le implicazioni della mia soluzione? Che impatto avrà? Cambierà il mondo? È un successo importante? È solo un'aggiunta marginale o è semplicemente servita per scoprire che questo percorso è stato una perdita di tempo? I risultati ottenuti sono generali, facilmente generalizzabili o sono specifici di un caso particolare? ecc

7.1 Sviluppi futuri

Migliorie o estensioni che possono essere sviluppate sul prodotto.

7.2 Considerazioni personali

Cosa ho imparato in questo progetto? ecc

8 Bibliografia

8.1 Bibliografia per articoli di riviste:

1. Cognome e nome (o iniziali) dell'autore o degli autori, o nome dell'organizzazione,
2. Titolo dell'articolo (tra virgolette),
3. Titolo della rivista (in italico),
4. Anno e numero
5. Pagina iniziale dell'articolo,

8.2 Bibliografia per libri

1. Cognome e nome (o iniziali) dell'autore o degli autori, o nome dell'organizzazione,
2. Titolo del libro (in italico),
3. ev. Numero di edizione,
4. Nome dell'editore,
5. Anno di pubblicazione,
6. ISBN.

8.3 Sitografia

1. URL del sito (se troppo lungo solo dominio, evt completo nel diario),
2. Eventuale titolo della pagina (in italico),
3. Data di consultazione (GG-MM-AAAA).

Esempio:

- <http://standards.ieee.org/guides/style/section7.html>, *IEEE Standards Style Manual*, 07-06-2008.

9 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
- Documentazione di prodotti di terzi
- Eventuali guide utente / Manuali di utilizzo
- Mandato e/o Qdc
- Prodotto
- ...