

Diario di lavoro

Luogo	Canobbio
Data	03.09.2019

Lavori svolti

**Nella prima giornata del progetto abbiamo letto e analizzato tutti i quaderni dei compiti.
Alla fine ho scelto il progetto di Geo Petrini di sviluppare un deduplicatore di files multipiattaforma.**

Dopo aver analizzato il QDC mi sono giunte delle domande:

- Avrò a disposizione un pc mac per poter testare la mia app?
- Il rapporto in che formato dovrà essere generato?

Problemi riscontrati e soluzioni adottate

-

Punto della situazione rispetto alla pianificazione

-

Programma di massima per la prossima giornata di lavoro

-

Diario di lavoro

Luogo	Canobbio
Data	05.09.2019

Lavori svolti

Mi sono informato su cosa era REST e come applicarlo nel mio progetto.

Ho guardato i seguenti siti:

- <https://www.javaworld.com/article/2077920/java-app-dev-rest-for-java-developers-part-1-it-s-about-the-information-stupid.html>
- <https://www.codecademy.com/articles/what-is-rest>

Mi sono anche informato sul framework spring e come utilizzarlo con REST

<https://spring.io/guides/gs/rest-service/>

Ho capito come funziona il sistema REST e ho iniziato a scrivere i primi capitoli della documentazione.

Ho finito di scrivere il capitolo 1.1, 1.2, 1.3, 1.4

Ho preparato una macchina virtuale di windows 10 per poter utilizzare microsoft Project per creare il gantt come richiesto visto che il gantt fatto con GanttProject non va bene.

Problemi riscontrati e soluzioni adottate

-

Punto della situazione rispetto alla pianificazione

Al passo con il gantt

Programma di massima per la prossima giornata di lavoro

Iniziare a scrivere I capitoli dell'analisi e scrivere i requisiti.

Diario di lavoro

Luogo	Canobbio
Data	06.09.2019

Lavori svolti

**Le prime due ore ho finito il gantt preventivo con Microsoft Project.
Il rimanente del tempo ho iniziato a fare l'analisi dei requisiti.**

Mi sono consultato con il docente Valsangiacomo su come rappresentare e stampare il gantt.

Sto ancora pensando se usare il framework <https://spring.io/guides/gs/rest-service/> in java per gestire le richieste REST.

Ho creato i primi requisiti nella documentazione

Problemi riscontrati e soluzioni adottate

-

Punto della situazione rispetto alla pianificazione

Sono al passo con il Gantt

Programma di massima per la prossima giornata di lavoro

Finire di scrivere i requisiti

Convertire i diari in versione word

Fadil Smajilbasic I4AC

Diario di lavoro

Luogo	Canobbio
Data	10.09.2019

Lavori svolti

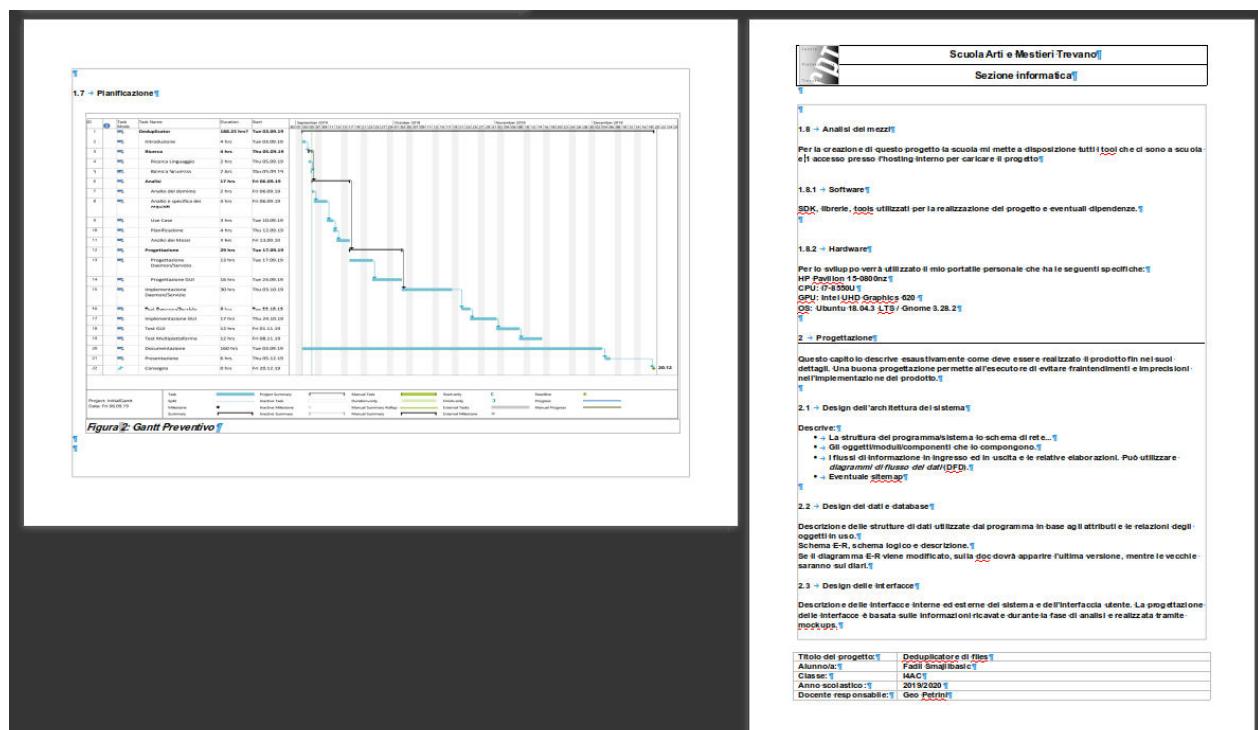
Ho finito di scrivere i requisiti analizzando ancora una volta in dettaglio il qdc, inoltre ho trasferito tutti i diari precedenti in formato word dal formato markdown.

Ho inserito un immagine del gantt preventivo all'interno della documentazione, in più ho creato il diagramma del Use Case, ma devo ancora verificare con il docente se va bene.

Ho iniziato a scrivere l'analisi dei mezzi

Problemi riscontrati e soluzioni adottate

Ho riscontrato un problema usando LibreOffice Writer, per inserire l'immagine del gantt dovevo girare una pagina in orizzontale, ma questo mi cambiava il footer e header della pagina dopo quella girata:



A sinistra la pagina girata e a destra la pagina dopo con il footer e header della prima pagina del documento

Soluzione: Alla fine della documentazione, aproro il file con Microsoft Word e mettero apposto l'header e il footer

Punto della situazione rispetto alla pianificazione

Sono al passo con il gantt

Programma di massima per la prossima giornata di lavoro

Consultarsi con il docente riguardo il diagramma UseCase e se è il caso mettere apposto il diagramma

Finire di scrivere l'analisi dei mezzi

Diario di lavoro

Luogo	Canobbio
Data	12.09.2019

Lavori svolti

Ho corretto il Use case come richiesto dal docente Geo Petrini rendendolo più completo, inoltre abbiamo discusso su come gestire l'accesso al servizio, poiché al momento non è prevista la gestione degli utenti.

Questo potrebbe essere un requisito che sarà aggiunto in futuro.

Chiedendo consiglio al docente responsabile ho deciso di usare MySQL per il salvataggio dei dati.

Il gantt consuntivo lo modificherò man mano che vado avanti con il progetto, quindi oggi l'ho creato e modificarlo per la prima volta.

Ho iniziato a creare il diagramma ER.

Problemi riscontrati e soluzioni adottate

Soluzione al problema dello stile delle pagine (riscontrato nel diario del 10.09.2019):

Girare l'immagine al posto di girare l'intera pagina. Questo permette di mantenere lo stesso stile del footer e del header.

Punto della situazione rispetto alla pianificazione

Sono una giornata di lavoro in avanti rispetto al Gantt preventivo

Programma di massima per la prossima giornata di lavoro

Finire il diagramma ER e fare ricerca sul software che utilizzerò

Diario di lavoro

Luogo	Canobbio
Data	13.09.2019

Lavori svolti

Ho finito il diagramma ER del database e il diagramma di flusso su come funzionerà il servizio.

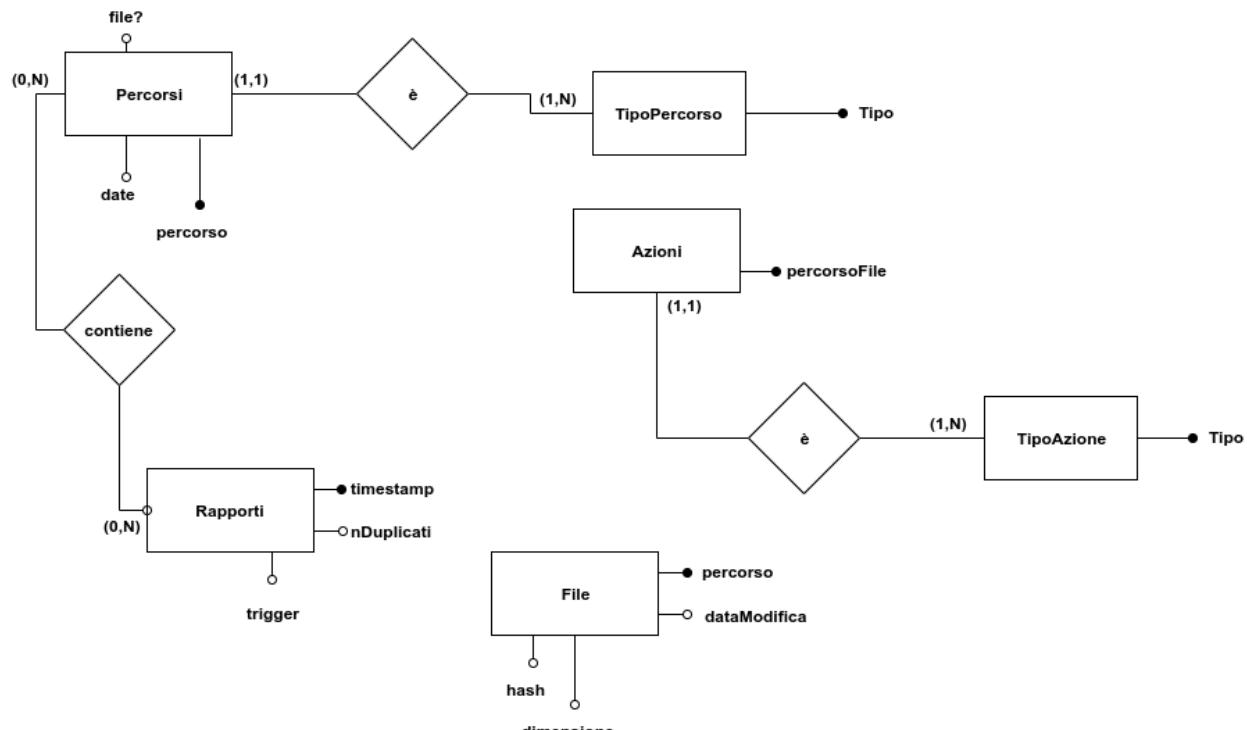


Diagramma ER

Il docente Guido Montalbetti ci ha detto che il diagramma di flusso non deve essere dettagliato.

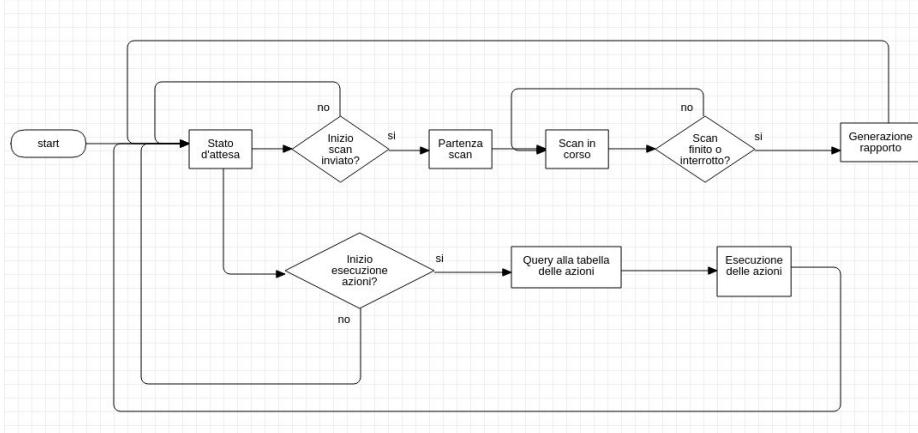


Diagramma di flusso

Ho inserito il diagramma ER nella documentazione è l'ho descritto.

Ho iniziato a creare il mockup delle interfacce

Problemi riscontrati e soluzioni adottate

-

Punto della situazione rispetto alla pianificazione

Sono al passo con il gantt

Programma di massima per la prossima giornata di lavoro

Finire il mockup delle interfacce

Diario di lavoro

Luogo	Canobbio
Data	17.09.2019

Lavori svolti

Ho finito il capitolo 3.3 Design delle interfacce creando 5 mockup e inserendoli nella documentazione

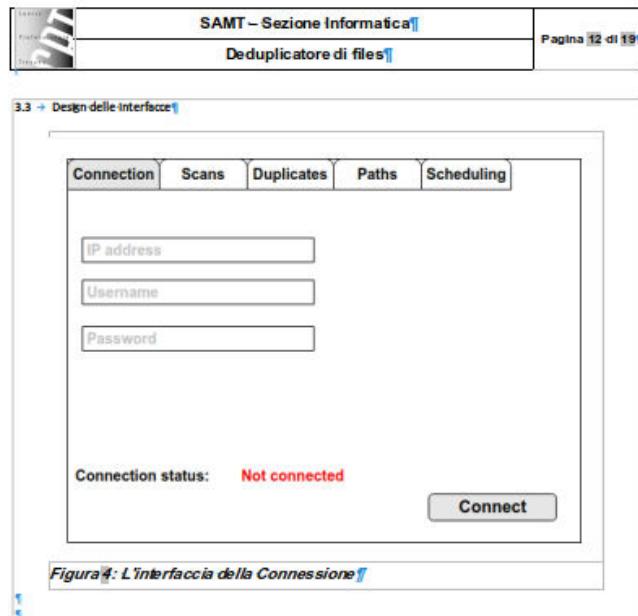




Figura 5: L'interfaccia che contiene le informazioni sulle scansioni

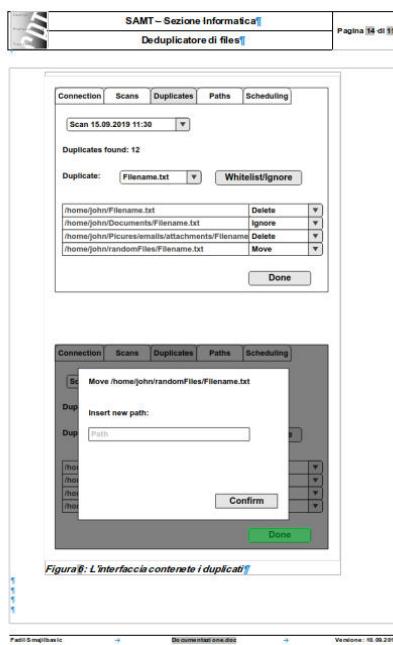


Figura 6: L'interfaccia contenente i duplicati

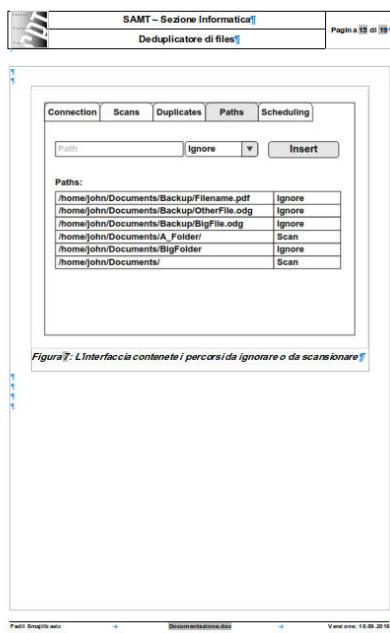


Figura 7: L'interfaccia contenete i percorsi da ignorare o da scansionare

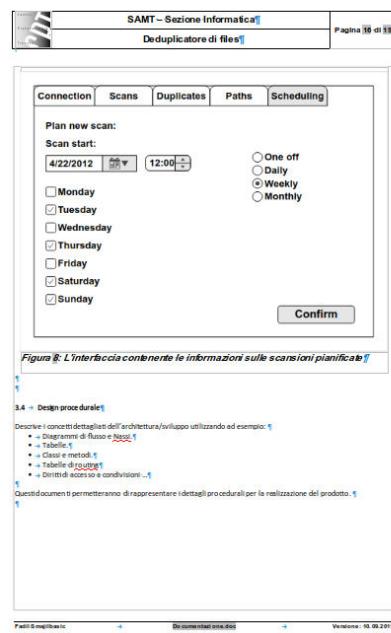


Figura 8: L'interfaccia contenente le informazioni sulle scansioni pianificate

Ho fatto un po' di ricerca su docker e nginx e altre funzionalità che offre il framework spring per avere una migliore idea su come si interfaccieranno tutte le parti del progetto.

Problemi riscontrati e soluzioni adottate

-

Punto della situazione rispetto alla pianificazione

Nome Progetto: Deduplicatore di files

2

Sono avanti di 3 ore di lavoro rispetto il gantt

Programma di massima per la prossima giornata di lavoro

Iniziare a sviluppare il database

Diario di lavoro

Luogo	Canobbio
Data	19.09.2019

Lavori svolti

Ho scritto la sintassi sql per creare il database e l'ho eseguito, facendo questo mi sono accorto di alcuni errori sul diagramma che ho corretto.

Il mio docente responsabile mi ha fatto notare delle mancanze nello schema ER e nei mockup delle interfacce, le modifiche le applicherò la prossima giornata di lavoro.

Dopo le modifiche dovrò ricreare il database.

Problemi riscontrati e soluzioni adottate

Non riuscivo ad accedere all'utente root di mysql, anche dopo aver configurato il servizio tramite il seguente comando: sudo mysql_secure_installation utility.

Soluzione:

Eseguire il client mysql usando sudo → sudo mysql -u root

Punto della situazione rispetto alla pianificazione

Sono avanti di 4 ore rispetto al gantt

Programma di massima per la prossima giornata di lavoro

Coreggere l'ER

Correggere le mockup

Diario di lavoro

Luogo	Canobbio
Data	20.09.2019

Lavori svolti

Ho corretto gli errori fatti notare dal mio docente responsabile, gli errori riguardavano il diagramma ER e i mockup.

Duplicate 1:
Hash: 7ead3c169f4b3b2dac7ab45e6c32f6e3 Size: 19KB

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

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 1: La nuova interfaccia dei duplicati

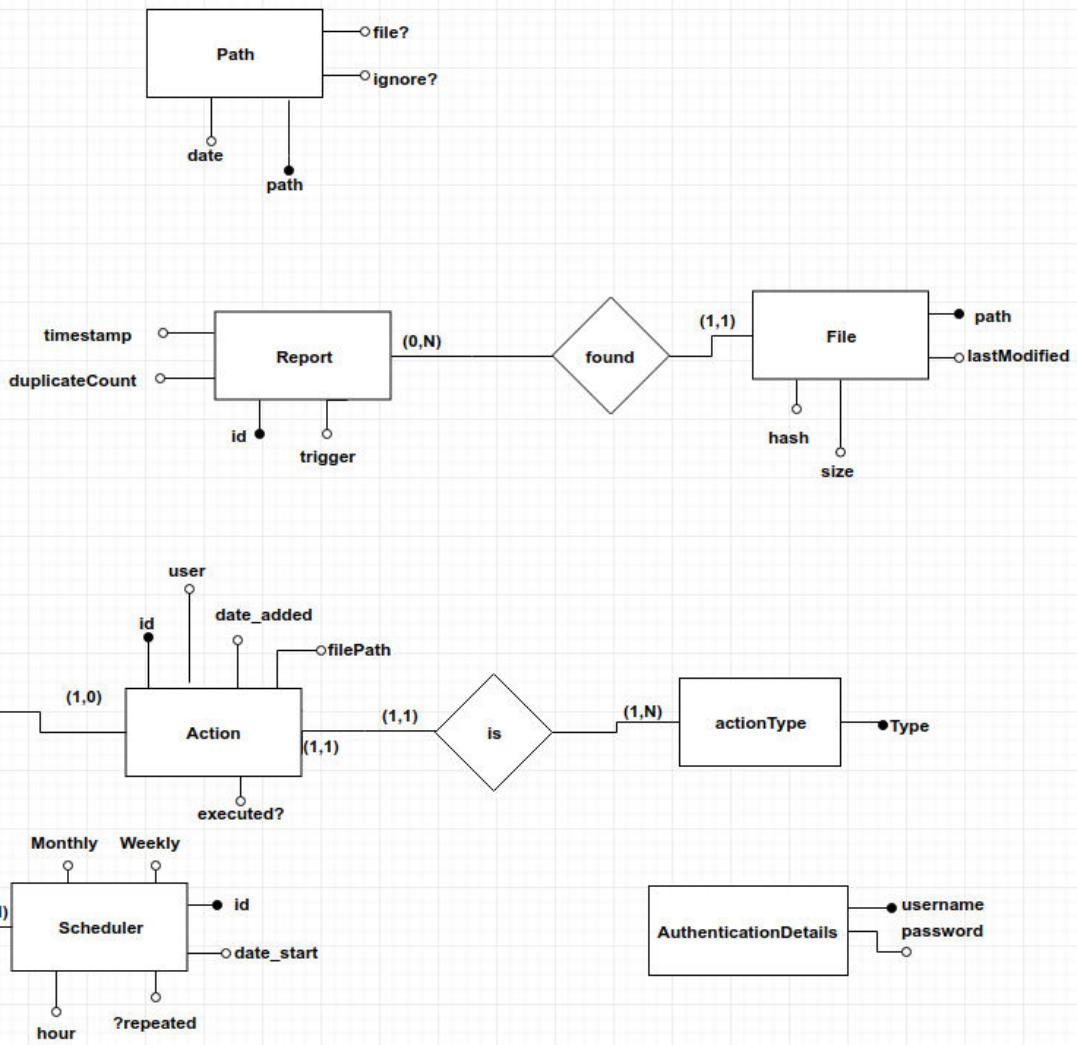


Figura 2: Nuovo diagramma ER

Ho anche descritto per metà il diagramma ER nella documentazione

Problemi riscontrati e soluzioni adottate

-

Punto della situazione rispetto alla pianificazione

Sono avanti di 2 ore di lavoro rispetto al gantt

Programma di massima per la prossima giornata di lavoro

Finire di descrivere il diagramma ER nella documentazione

Diario di lavoro

Luogo	Canobbio
Data	24.09.2019

Lavori svolti

Ho modificato leggermente il diagramma ER, ed ho finito di descrivere il digramma nella documentazione.

Ho creato di database sul mio server mysql locale.

Problemi riscontrati e soluzioni adottate

Punto della situazione rispetto alla pianificazione

Sono avanti di 6 ore di lavoro rispetto al gantt

Ho stimato male il tempo che mi serve a progettare il servizio (e il database)

Programma di massima per la prossima giornata di lavoro

Diario di lavoro

Luogo	Canobbio
Data	26.09.2019

Lavori svolti

Ho corretto alcuni errori nelle mockup delle gui e ho pensato un po' sulle query che dovrò fare.

Ho fatto ricerca su come interfacciare il framework spring con mysql

Problemi riscontrati e soluzioni adottate

Punto della situazione rispetto alla pianificazione

Sono avanti rispetto al gantt

Programma di massima per la prossima giornata di lavoro

Iniziare l'implementazione

Diario di lavoro

Luogo	Canobbio
Data	27.09.2019

Lavori svolti

Ho iniziato l'implementazione.

Ho creato diverse classi che rappresentano le tabelle del database per poter comunicare con quest'ultimo.

Per la comunicazione tra il programma e il database uso java persistance api che spring fornisce.

Problemi riscontrati e soluzioni adottate

Il progetto non mi compilava perché al posto di usare la classe **javax.persistence.Id** io per sbaglio usavo la seguente:

import org.springframework.data.annotation.Id;

E perciò la compilazione del gradle non riusciva a finire in modo giusto.

Punto della situazione rispetto alla pianificazione

Sono avanti rispetto al gantt di 6 ore di lavoro

Programma di massima per la prossima giornata di lavoro

Finire di creare le classi che rappresentano le tabelle

Diario di lavoro

Luogo	Canobbio
Data	01.10.2019

Lavori svolti

Ho creato tutte le classi che rappresentano le entità del database con le relazioni usando spring boot hibernate e java persistence api e le apposite repository.

Problemi riscontrati e soluzioni adottate

Avevo difficoltà nel capire come funzionassero le relazioni ma cercando online ho trovato il seguente sito <https://www.baeldung.com/jpa-join-column> che spiega bene le notazioni e come funzionano

Punto della situazione rispetto alla pianificazione

Sono avanti rispetto al gantt di 3 ore di lavoro

Programma di massima per la prossima giornata di lavoro

Completare le classi delle entità

Diario di lavoro

Luogo	Canobbio
Data	03.10.2019

Lavori svolti

Ho finito l'implementazione del controller e tutta la logica che ci sta dietro con tutti i controlli per la gestione dei percorsi da ignorare o da scanerizzare.

Ho scritto anche la logica per fare la scansione che funziona in modo multithreaded, am è ancora da testare e revisionare.

Ho scoperto il tool <https://www.getpostman.com/downloads/>, serve a mandare richieste di qualsiasi tipo con qualsiasi parametro su un server per fare testing.

Problemi riscontrati e soluzioni adottate

Ho seguito la seguente guida per capire come funziona la gestione delle eccezioni:

<https://www.baeldung.com/exception-handling-for-rest-with-spring>

Ho guardato il seguente sito per coreggere alcuni erori che avevo nelal gestione dei parametri in entrata:

<https://www.baeldung.com/spring-request-param>

Punto della situazione rispetto alla pianificazione

Sono avanti rispetto al gantt di 3 ore di lavoro

Programma di massima per la prossima giornata di lavoro

Finire il controller della scansione e la logica che ci sta dietro la scansione.

Diario di lavoro

Luogo	Canobbio
Data	03.10.2019

Lavori svolti

Ho implementato la funzione del controller per le scansioni per eseguire le scansioni e per fermarle, questo è da finire.

Ho completato alcune parti del controller per i percorsi e ho fatto altri test.

Problemi riscontrati e soluzioni adottate

All'inizio della scansione il programma tira una null pointer exception output:

Exception in thread "Thread-3" java.lang.NullPointerException at
samt.smajilbasic.deduplicator.scanner.ScanManager.run(ScanManager.java:47)

Questo problema è dovuto al framework spring che non inizializza la variabile per la comunicazione tra il servizio e il database anche se usa la notazione corretta usando @Autowired.

```
31     @Autowired
32     GlobalPathRepository gpr;
```

L'errore occorre alla riga 47:

```
45     public void run() {
46
47         System.out.println("count: " + gpr.count());
48         paths = gpr.findAll().iterator();
```

Punto della situazione rispetto alla pianificazione

Sono al passo con il gantt

Programma di massima per la prossima giornata di lavoro

Finire il controller della scansione e la logica che ci sta dietro la scansione.

Diario di lavoro

Luogo	Canobbio
Data	08.10.2019

Lavori svolti

Problemi riscontrati e soluzioni adottate
Ho risolto il problema del scorso diario trovando la soluzione su questo sito:
https://stackoverflow.com/questions/19896870/why-is-my-spring-autowired-field-null
Ho modificato il codice nel seguente modo nella classe ScanController:
<pre>44 45 // currentScan = new ScanManager(reportRepository.findById(report.getId()).get()); 46 currentScan.setReport(reportRepository.findById(report.getId()).get()); 47 currentScan.start();</pre>
Al posto di istanziare un nuovo ScanManager dovevo lasciare quel lavoro al @Autowired di spring.

Punto della situazione rispetto alla pianificazione
Sono al passo con il gantt

Programma di massima per la prossima giornata di lavoro
Fare test sulle nuove funzionalità aggiunte e continuare con l'implementazione di altre parti del programma.

Diario di lavoro

Luogo	Canobbio
Data	10.10.2019

Lavori svolti

Ho aggiunto un controllo al controller delle scansioni per impedire l'inizializzazione di uno scan se non ci sono percorsi da scansionare.

Ho aggiunto il codice per trovare i duplicati e corretto alcuni errori nella classe ScanManager.java

Ho fatto testing e debugging delle parti del codice aggiunte in precedenza e quelle aggiunte oggi.

Problemi riscontrati e soluzioni adottate

Mysql non permetteva l'uso di group by con più parametri.

Soluzione trovata su stackoverflow: <https://stackoverflow.com/questions/23921117/disable-only-full-group-by>

Bisognava inserire le seguenti righe nel file di configurazione di mysql, che si trova sotto /etc/mysql/my.cnf, e riavviare il servizio:

```
[mysqld]
sql_mode =
"STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,
NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION"
```

Punto della situazione rispetto alla pianificazione

Sono al passo con il gantt

Programma di massima per la prossima giornata di lavoro

Fare test sulle nuove funzionalità aggiunte e continuare con l'implementazione di altre parti del programma.

Diario di lavoro

Luogo	Canobbio
Data	10.10.2019

Lavori svolti

Le prime due ore di lezione abbiamo ascoltato la presentazione sulla valutazione dei progetti.

Ho aggiunto la possibilità di eliminare un percorso dalla tabella dei percorsi.

Ho eliminato le eccezioni custom inutili

Problemi riscontrati e soluzioni adottate

-

Punto della situazione rispetto alla pianificazione

Sono al passo con il gantt

Programma di massima per la prossima giornata di lavoro

Devo controllare la query per la ricerca dei duplicati

Diario di lavoro

Luogo	Canobbio
Data	15.10.2019

Lavori svolti

Ho corretto la query per la ricerca dei duplicati e ho fatto test su quello sia dal progetto sia connettendomi direttamente a mysql da terminale.

Ho aggiunto delle query custom al file FileRepository.java, utili per ricerca di file grazie solo al hash del file o solo grazie al id del report.

Al FileController.java ho aggiunto la funzionalità di ricavare i file dal id di un report.

Ho aggiunto la possibilità di ricavare tutti i file che sono rilevati come duplicati in un rapporto e tutti i duplicati in un rapporto

Ho migliorato il codice in vari punti del progetto.

Problemi riscontrati e soluzioni adottate

- Il progetto non riusciva a compilare a causa di un problema con la mia query custom per trovare i file duplicati:

```
@Query("SELECT f.size,f.hash,count(*) as count "+  
       "FROM file f "+  
       "WHERE report=?1 "+  
       "GROUP BY f.hash,f.size "+  
       "HAVING count > 1 "+  
       "ORDER BY hash,size DESC")  
List<Duplicate> findDuplicatesFromReport(Report report);
```

La compilazione non finiva a buon fine perché il compilatore non riusciva a validare il tipo di ritorno della query.

La soluzione l'ho trovata sul seguente sito:

<https://stackoverflow.com/questions/44647630/validation-failed-for-query-for-method-jpql>

Bisognava aggiungere il parametro *nativeQuery* così:

```
@Query( value = "SELECT f.size,f.hash,count(*) as count "+  
       "FROM file f "+  
       "WHERE report=?1 "+  
       "GROUP BY f.hash,f.size "+  
       "HAVING count > 1 "+  
       "ORDER BY hash,size DESC",  
       nativeQuery = true)  
List<Duplicate> findDuplicatesFromReport(Report report);
```

Punto della situazione rispetto alla pianificazione

Sono al passo con il gantt

Programma di massima per la prossima giornata di lavoro

Controllare i requisiti e vedere se sono sulla strada giusta.

Fare altre prove sul codice scritto

Aggiungere controlli di dati dove richiesto

Controllare i metodi pause e stop

Diario di lavoro

Luogo	Canobbio
Data	17.10.2019

Lavori svolti

Ho cambiato la filosofia della scansione dei percorsi, adesso l'aggiunta al database succede contemporaneamente con la scansione dei percorsi, questo velocizza tutto il processo.

Ho aggiunto il controllo dei percorsi da ignorare.

Ho aggiornato il mio docente responsabile su dove sto con il sviluppo del progetto e ho ricevuto alcuni consigli.

Problemi riscontrati e soluzioni adottate

Alla richiesta dei rapporti delle scansioni il programma ritornava un json infinito dove un rapporto contiene la referenza ai file, i file avevano però la referenza al rapporto, e quel rapporto conteneva la referenza ai file.... e così si creava una riposta ricorsiva.

Per mettere apposto questo problema dovevo mettere dalla parte della entità Report un tag che indica che il campo file è un json. E dalla parte della entità File un'altro tag che indica che la proprietà associata fa parte del collegamento bidirezionale tra i campi; e che il suo ruolo nel collegamento è quello di essere "figlio" ..

```
@Entity  
public class Report {  
  
    @OneToMany(fetch = FetchType.LAZY, mappedBy = "report")  
    @JsonManagedReference  
    private List<File> file;
```

```
@Entity  
public class File {  
  
    @ManyToOne(fetch = FetchType.LAZY)  
    @JoinColumn(name="report")  
    @JsonBackReference  
    private Report report;
```

Soluzione trovata su: <https://stackoverflow.com/questions/3325387/infinite-recursion-with-jackson-json-and-hibernate-jpa-issue/18288939#18288939>

Punto della situazione rispetto alla pianificazione

Sono al passo con il gantt

Programma di massima per la prossima giornata di lavoro

Controllare i metodi pause e stop

Testare il funzionamento dei percorsi da testare

Diario di lavoro

Luogo	Canobbio
Data	18.10.2019

Lavori svolti

Durante un controllo dei metodo pause e resume mi sono accorto che quelli non funzionavano in modo corretto in modo corretto.

L'implementazione del metodo pause e resume mi ha preso più tempo che previsto.

Ho fatto altre piccole correzioni nel codice.

Problemi riscontrati e soluzioni adottate

All'avvio di una nuova scansione il programma usava il vecchio scan manager, questo non andava bene ma non potevo semplicemente chiamare il costruttore con new ScanManager(...) perché questo crea un oggetto di java e non fa tutti i passaggi per poter essere utilizzato con le classi di spring (non è una Bean), in più facendo così non si stanziano le variabili @Autowired nella classe ScanManager.

Per far creare la casse a spring in modo che diventi una Bean ho trovato la seguente soluzione:

<https://codippa.com/how-to-autowire-objects-in-non-spring-classes/>

Che ho implementato nella classe ScanController

Per far aspettare le thread di scansione mi sono accorto che manca un oggetto monitor che sul quale le thread aspetteranno e dal quale si risveglieranno anche.

Questo ho risolto dichiarando una semplice variabile di tipo Object nella classe ScanManager e passandogli questo oggetto come parametro alle thread.

Punto della situazione rispetto alla pianificazione

Sono al passo con il gantt

Programma di massima per la prossima giornata di lavoro

Controllare metodo stop

Implementare l'autenticazione

Diario di lavoro

Luogo	Canobbio
Data	22.10.2019

Lavori svolti

Ho fatto dei test sul metodo che ferma la scansione. Dopo un po' di test ho scoperto che la scansione non si fermava in modo corretto e quindi non lascivia partire una nuova scansione

Ho corretto questo problema

Problemi riscontrati e soluzioni adottate

-

Punto della situazione rispetto alla pianificazione

Sono indietro di 4 ore rispetto al gantt preventivo

Programma di massima per la prossima giornata di lavoro

Implementare la gestione delle azioni da fare

Implementare l'autenticazione

Diario di lavoro

Luogo	Canobbio
Data	24.10.2019

Lavori svolti

Ho aggiunto la gestione delle azioni da applicare ai duplicati grazie al ActionController.java

Ho aggiunto la creazione automatica del database se questo non esistesse.

Ho iniziato a scrivere la logica che ci sta dietro all'eliminazione e spostamento dei file.

Problemi riscontrati e soluzioni adottate

-

Punto della situazione rispetto alla pianificazione

Sono al passo con il gantt

Programma di massima per la prossima giornata di lavoro

Finire l'implementazione della parte che elimina e sposta i file

Implementare l'autenticazione

Diario di lavoro

Luogo	Canobbio
Data	25.10.2019

Lavori svolti

Ho finito la classe ActionManager che permette di eseguire le azioni impostate all'eliminazione e spostamento dei file.

Ho creato il ScheduleController per l'inserimento di scansioni pianificate, è ancora da finire.

L'autenticazione la impleggerò di seguito.

Problemi riscontrati e soluzioni adottate

-

Punto della situazione rispetto alla pianificazione

Sono al passo con il gantt

Programma di massima per la prossima giornata di lavoro

Finire di implementare il ScheduleController

Implementare un modo per iniziare una scansione una volta che sia scattato un Schedule.

Implementare la gestione dei utenti

Diario di lavoro

Luogo	Canobbio
Data	05.11.2019

Lavori svolti

Ho finito la classe ScheduleController e ho implementato un modo per avviare le scansioni e l'esecuzione delle azioni secondo uno schedule impostato, inoltre ho implementato anche l'autenticazione BASIC e la gestione dei utenti. Tutto questo è da testare e correggere eventuali errori.

Problemi riscontrati e soluzioni adottate

Non ero in chiaro su come implementare l'autenticazione quindi mi sono consultato con i seguenti siti:

<https://www.baeldung.com/spring-security-basic-authentication>

<https://www.baeldung.com/spring-security-with-maven>

<https://www.devglan.com/spring-security/spring-boot-security-rest-basic-authentication>

Punto della situazione rispetto alla pianificazione

Sono al passo con il gantt

Programma di massima per la prossima giornata di lavoro

Test delle funzioni implementate

Diario di lavoro

Luogo	Canobbio
Data	07.11.2019

Lavori svolti

Ho cercato di risolvere i problemi che sono apparsi all'aggiunta dell'autenticazione.

Problemi riscontrati e soluzioni adottate

L'autenticazione non poteva essere eseguita a causa di un id mancante che definisce il tipo di hash usato, questa funzione è stata implementata con spring security 5.X

La soluzione l'ho trovata sul seguente sito: <https://www.javacodemonk.com/spring-security-5-there-is-no-passwordencoder-mapped-for-the-id-b0503f3d>

L'autenticazione avviene correttamente, ma l'applicazione non ritorna nessuna risposta né usando i metodi GET né POST né PUT

Punto della situazione rispetto alla pianificazione

Sono indietro di 4 ore rispetto al gantt.

Programma di massima per la prossima giornata di lavoro

Test delle funzioni implementate

Diario di lavoro

Luogo	Canobbio
Data	08.11.2019

Lavori svolti

Mentre eseguo i test sull'autenticazione ho scoperto che c'è un problema critico che mi ha impedito il proseguimento dell'implementazione del codice e ho mi sono concentrato nella risoluzione del problema. Ho creato un progetto da niente per poter eseguire i test cercando soluzioni online, al posto di pasticciare sul progetto principale.

Dopo 6 ore di lavoro e ricerca non ho trovato una soluzione perciò scritto un'email al mio docente responsabile per non cadere troppo indietro rispetto alla pianificazione.

La GUI sarà veloce da implementare, quindi non mi preoccupo tanto anche sono sono abbastanza indietro rispetto al gantt preventivo, e anche il gantt preventivo era stimato male.

Problemi riscontrati e soluzioni adottate

Il problema:

L'autenticazione avviene con successo,

se non si mettono credenziali nella richiesta, reindirizza la richiesta su una classe che poi da la risposta (messaggio d'errore in formato JSON, e come HTTPS status code: 401) all'utente,

se le credenziali mandate con la richiesta sono sbagliate risponde come nel caso che non ci fossero (quindi messaggio d'errore e 401),

se le credenziali sono giuste invece ritorna una pagina vuota con HTTP status code 200, e non importa cosa faccio, non cambia.

Punto della situazione rispetto alla pianificazione

Sono indietro di 8 ore rispetto al gantt iniziale

Programma di massima per la prossima giornata di lavoro

Nome Progetto: Deduplicatore di files

Risolvere il problema dell'autenticazione
Verificare il funzionamento dello scheduler
Iniziare il sviluppo della GUI

Diario di lavoro

Luogo	Canobbio
Data	08.11.2019

Lavori svolti

Ho trovato la soluzione al problema dell'autenticazione e ho risolto altri bug che ho scoperto testando i nuovi metodi per l'aggiunta delle azioni e per la pianificazione delle scansioni e dell'esecuzione delle azioni.

Problemi riscontrati e soluzioni adottate

Il problema:

Il file di configurazione, SecurityConfig.java, si trovava nella cartella samt.smajilbasic.deduplicator.security mentre doveva essere piazzato nella cartella samt.smajilbasic.deduplicator.config

Punto della situazione rispetto alla pianificazione

Sono indietro di 8 ore rispetto al gantt iniziale

Programma di massima per la prossima giornata di lavoro

Iniziare il sviluppo della GUI

Diario di lavoro

Luogo	Canobbio
Data	14.11.2019

Lavori svolti

Ho corretto alcuni errori dello scheduler e ho iniziato a creare il progetto con NetBeans per il sviluppo della GUI.

Problemi riscontrati e soluzioni adottate

Punto della situazione rispetto alla pianificazione

Sono indietro di 8 ore rispetto al gantt iniziale

Programma di massima per la prossima giornata di lavoro

Finire il layout di tutte le schermate della applicazione

Diario di lavoro

Luogo	Canobbio
Data	15.11.2019

Lavori svolti

Ho creato le interfacce della gui con tutti gli elementi necessari, inoltre ho inserito le classi che descrivono le entità dal progetto deduplicator in quello della GUI, è una soluzione temporanea, magari metterò assieme i due progetti. Ho anche craeto la classe client che comunicera con il servizio.

Problemi riscontrati e soluzioni adottate

Punto della situazione rispetto alla pianificazione

Sono indietro di 6 ore rispetto al gantt iniziale

Programma di massima per la prossima giornata di lavoro

Finire la logica dietro le schermate della GUI

Diario di lavoro

Luogo	Canobbio
Data	19.11.2019

Lavori svolti

Ho lavorato sulla classe della GUI che dovrà fare le richieste alle API e sulla interfaccia di login e quella che visualizza i percorsi da scansionare. Penso che sono su una buona strada

Problemi riscontrati e soluzioni adottate

Punto della situazione rispetto alla pianificazione

Sono indietro di 4 ore rispetto al gantt iniziale

Programma di massima per la prossima giornata di lavoro

Finire la logica dietro le schermate della GUI

Diario di lavoro

Luogo	Canobbio
Data	21.11.2019

Lavori svolti

Ho lavorato sulla interfaccia che rappresenta l'inserimento e cancellazione dei percorsi. Devo controllare il funzionamento di essa.

Problemi riscontrati e soluzioni adottate

Punto della situazione rispetto alla pianificazione

Sono indietro di 6 ore rispetto al gantt iniziale

Programma di massima per la prossima giornata di lavoro

Finire la logica dietro le schermate della GUI

Diario di lavoro

Luogo	Canobbio
Data	22.11.2019

Lavori svolti

Ho corretto alcuni bug nell'interfaccia dei percorsi e nel client, ho finito l'interfaccia dei percorsi con l'aggiunta e l'inserimento dei path, ho riscontrato alcuni errori e difficoltà nella creazione del client http che si interfaccia con le API.

Problemi riscontrati e soluzioni adottate

Non riuscivo a fare la richiesta DELETE sul controller PathController con un parametro del from perché l'HTTP client di java 11 non lo supporta quindi ho dovuto cambiare tutto il codice e usare quello dei Spring che è più ricco di funzionalità ed è più semplice da usare. Tutte le informazioni necessarie ho trovato su questi siti:

<https://www.baeldung.com/spring-rest-template-multipart-upload>

<https://code-examples.net/en/q/2498446>

<https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/client/RestTemplate.html#exchange-java.lang.String-org.springframework.http.HttpMethod-org.springframework.http.HttpEntity-java.lang.Class-java.lang.Object...->

<https://tamasgyorfi.net/2015/03/27/posting-multipart-requests-with-resttemplate/>

Punto della situazione rispetto alla pianificazione

Sono indietro di 8 ore rispetto al gantt iniziale

Programma di massima per la prossima giornata di lavoro

Finire la logica dietro le schermate della GUI

Diario di lavoro

Luogo	Canobbio
Data	26.11.2019

Lavori svolti

Ho scritto la logica che ci sta dietro la visuale dei report (DuplicatePanel) e quella dietro l'interfaccia dello scan

Problemi riscontrati e soluzioni adottate

-

Punto della situazione rispetto alla pianificazione

Sono indietro di 8 ore rispetto al gantt iniziale

Programma di massima per la prossima giornata di lavoro

Finire la logica dietro le schermate della GUI

Diario di lavoro

Luogo	Canobbio
Data	28.11.2019

Lavori svolti

Ho raffinato e corretto errori nell'hash rendendolo più efficace e versatile.

Ho corretto alcuni errori del ScanController e del Mainframe.

Problemi riscontrati e soluzioni adottate

Le classe che si occupa delle scannerizzazioni tirava una Exception perchè ricavavo la grandezza del file, leggendo i suoi contenuti.

Soluzione:

Prendere la grandezza dal file usando il metodo length della classe RandomAccessFile

```
RandomAccessFile fileRAF = new RandomAccessFile(file.getAbsolutePath(), "r");
String hash = Hasher.getHash(fileRAF, "MD5");
long size = fileRAF.length();
```

Punto della situazione rispetto alla pianificazione

Sono indietro di 6 ore rispetto al gantt iniziale

Programma di massima per la prossima giornata di lavoro

Finire la logica dietro la schermata duplicates della GUI

Diario di lavoro

Luogo	Canobbio
Data	29.11.2019

Lavori svolti

Ho modificato il modo in cui vengono fatti partire le thread per creare le hash dei file, questo ha portato ad una riduzione drastica della durata della scansione

Nella immagine sotto possiamo vedere che per la stessa directory ci mette **50 sec** in meno.

```
{
    "duration": 72111,
    "start": 1575034929038,
    "filesScanned": 3552,
    "averageDuplicateCount": 0.224662,
    "user": null,
    "file": [],
    "id": 4
},
{
    "duration": 22467,
    "start": 1575039467379,
    "filesScanned": 3527,
    "averageDuplicateCount": 0.225971,
    "user": {
        "name": "Fadil Smajilbasic"
    }
}
```

Ho corretto alcuni bug della GUI.

Problemi riscontrati e soluzioni adottate

-

Punto della situazione rispetto alla pianificazione

Sono indietro di 4 ore rispetto al gantt iniziale

Programma di massima per la prossima giornata di lavoro

Finire la logica dietro la schermata duplicates della GUI

Controllare il ScanController

Diario di lavoro

Luogo	Canobbio
Data	03.12.2019

Lavori svolti

Ho cambiato il layout della schermata Duplicates, inoltre ho corretto alcuni errori nella schermata Scan.

Ho scritto una buona parte della logica dietro la schermata dei Duplicates

Ho corretto un errore nel ScanController, nel metodo getStatus dove veniva sollevata una NullPointerException.

Problemi riscontrati e soluzioni adottate

-

Punto della situazione rispetto alla pianificazione

Sono indietro di 4 ore rispetto al gantt iniziale

Programma di massima per la prossima giornata di lavoro

Finire la logica dietro la schermata duplicates della GUI

Diario di lavoro

Luogo	Canobbio
Data	05.12.2019

Lavori svolti

Oggi non ho lavorato perchè c'era la visita alla SUP di Friborgo organizzata dalla scuola.

Problemi riscontrati e soluzioni adottate

-

Punto della situazione rispetto alla pianificazione

-

Programma di massima per la prossima giornata di lavoro

-

Diario di lavoro

Luogo	Canobbio
Data	06.12.2019

Lavori svolti

Ho lavorato sulla schermata dei duplicati, adesso c'è la possibilità per l'utente di scegliere il rapporto che vuole visualizzare e i duplicati, dopodiché sullo schermo gli vengono mostrati i file duplicati di quel specifico duplicato selezionato. Ho aggiunto la possibilità di selezionare cosa fare con un file selezionato dalla lista.

Problemi riscontrati e soluzioni adottate

-

Punto della situazione rispetto alla pianificazione

Sono indietro di 8 ore rispetto al gantt iniziale

Programma di massima per la prossima giornata di lavoro

Finire la logica dietro la schermata duplicates della GUI

Iniziare a fare la parte d'implementazione della documentazione.

Diario di lavoro

Luogo	Canobbio
Data	10.12.2019

Lavori svolti

Ho aggiunto commenti javadoc alle classi del package
samt.smajilbasic.deduplicator.controller e corretto alcuni piccoli errori.

Ho aggiunto la funzionalità di applicare le operazioni da eseguire sui file, quello è ancora da controllare.

Ho corretto il Hasher che a volte generava delle hash di lunghezza diversa.

Problemi riscontrati e soluzioni adottate

-

Punto della situazione rispetto alla pianificazione

Sono indietro di 8 ore rispetto al gantt iniziale

Programma di massima per la prossima giornata di lavoro

Verificare il funzionamento dietro la schermata duplicates.

Diario di lavoro

Luogo	Canobbio
Data	12.12.2019

Lavori svolti

Ho finito la logica dietro la schermata duplicates e fatto alcuni controlli piccoli sul funzionamento.

Ho aggiunto i commenti alla classe AuthenticationDetails.

Problemi riscontrati e soluzioni adottate

-

Punto della situazione rispetto alla pianificazione

Sono indietro di 8 ore rispetto al gantt iniziale

Programma di massima per la prossima giornata di lavoro

Verificare funzionamento Scheduler e se le azioni impostate vengono eseguite

Cercare modo di implementare refresh durante la scansione nella schermata scan.

Diario di lavoro

Luogo	Canobbio
Data	13.12.2019

Lavori svolti

Ho controllato diverse parti del progetto e implementato la logica dietro il bottone info nella schermata duplicates. Ho implementato HTTPS nel progetto. Commentato alcune classi.

Problemi riscontrati e soluzioni adottate

-

Punto della situazione rispetto alla pianificazione

Sono indietro di 8 ore rispetto al gantt iniziale

Programma di massima per la prossima giornata di lavoro

Verificare funzionamento Scheduler e se le azioni impostate vengono eseguite

Diario di lavoro

Luogo	Canobbio
Data	17.12.2019

Lavori svolti

Mi sono portato avanti con la documentazione e ho commentato delle classi.

Problemi riscontrati e soluzioni adottate

-

Punto della situazione rispetto alla pianificazione

Sono indietro rispetto alla pianifica del gant e non finirò il progetto.

Programma di massima per la prossima giornata di lavoro

Stampare la documentazione e rilegare.

Diario di lavoro

Luogo	Canobbio
Data	19.12.2019

Lavori svolti

Ho completato la parte d'Implementazione della documentazione, esportato i diagrammi gantt e generato i diagrammi UML.

Problemi riscontrati e soluzioni adottate

-

Punto della situazione rispetto alla pianificazione

-

Programma di massima per la prossima giornata di lavoro

-