Gestione Web Ritardi Allievi SAMT

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

2.3 Use case 8

2.4 Pianificazione 9

2.4.1 Analisi 10

2.4.2 Progettazione 10

2.4.3 Implementazione 11

2.4.4 Testing 11

2.4.5 Consegna 11

2.5 Analisi dei mezzi 12

2.5.1 Software 12

2.5.2 Hardware 12

3 Progettazione 13

3.1 Design dell’architettura del sistema 13

3.2 Design dei dati e database 13

3.2.1 Schema ER 13

3.2.2 Descrizioni delle tabelle 14

3.2.3 Schema logico 16

3.3 Design delle interfacce 17

3.3.1 Pagina di accesso 17

3.3.2 Pagina cambio password 17

3.3.3 Pagina di recupero password 17

3.3.4 Pagina di aggiunta ritardi 18

3.3.5 Pagina di aggiunta recuperi 18

3.3.6 Pagina di gestione utenti 19

3.3.7 Pagina impostazioni 19

4 Implementazione 19

5 Test 19

5.1 Protocollo di test 19

5.2 Risultati test 20

5.3 Mancanze/limitazioni conosciute 20

6 Consuntivo 20

7 Conclusioni 20

7.1 Sviluppi futuri 20

7.2 Considerazioni personali 20

8 Sitografia 20

9 Allegati 21

# Introduzione

## Informazioni sul progetto

**Titolo:** Gestione Web Ritardi Allievi SAMT  
**Allievo coinvolto nel progetto:** Filippo Finke, [filippo.finke@samtrevano.ch](mailto:filippo.finke@samtrevano.ch)

**Classe:** I4AC Scuola Arti e Mestieri Trevano, Informatica

**Formatore:** Fabrizio Valsangiacomo, [fabrizio.valsangiacomo@edu.ti.ch](mailto:fabrizio.valsangiacomo@edu.ti.ch)

**Perito:** Gianluca Costante, [gianluca.costante@gmail.com](mailto:gianluca.costante@gmail.com)

**Data inizio:** 11.05.2020

**Data fine:** 29.05.2020

**Durata:** 82 ore

DA\_REVISIONARE

## Abstract

*The aim of the project "SAMT Student Web Delays Management" is to simplify, speed up and automate the process of the assignment of the students delays at the SAMT. At the moment the delays are marked mainly on paper or with other methods that do not guarantee the integrity of the data themselves and do not allow to perform quick searches, controls and other. The project in question therefore deals with digitizing what is done manually and creating a website that allows you to manage everything in an automated way. This will speed up the work to be done by the teachers. Thanks to this system it will also be possible to keep a record of the accumulated and recovered delays of all the students who have been registered. You will also be able to create a PDF file with the history of a student.*

DA\_REVISIONARE

## Scopo

Lo scopo del progetto “Gestione Web Ritardi Allievi SAMT” è quello di semplificare, velocizzare ed automatizzare il processo di assegnazione dei ritardi all’interno della Scuola d’Arti e Mestieri di Trevano. Al momento i ritardi vengono segnati principalmente su carta o con altre metodologie che però non garantiscono l’integrità dei dati stessi e non permettono di eseguire ricerche velocemente, controlli e altro. Il progetto in questione si occupa dunque di digitalizzare ciò che viene fatto manualmente e di creare un sito web che permetta di gestire il tutto in modo automatizzato. Questo permetterà di velocizzare il lavoro che dovrà essere svolto dai docenti. Grazie a questo sistema si potrà inoltre tenere uno storico dei ritardi accumulati e recuperati di tutti gli studenti che sono stati registrati. Sarà inoltre possibile creare un file PDF con lo storico di uno studente.

DA\_REVISIONARE

# Analisi

## Analisi del dominio

È stato richiesto lo sviluppo di un gestionale web per la gestione dei ritardi della Scuola d’Arti e Mestieri di Trevano. Il prodotto dovrà essere un applicativo web accessibile attraverso la rete utilizzando browser moderni (Esempio: Chrome). Gli utenti che accederanno a questo applicativo saranno principalmente docenti di classe che andranno ad inserire e gestire i ritardi dei propri alunni. Saranno disponibili ulteriori permessi che permetteranno di distinguere gli utenti in utenti amministratori e utenti normali. Gli utenti amministratori potranno gestire gli utenti presenti all’interno dell’applicativo. Gli utenti normali avranno dei permessi limitati (inserimento dei ritardi, visione dei dati o creazione PDF) che permetteranno a questo utente di eseguire solamente determinate azioni limitate. Sarà presente una pagina di amministrazione che permetterà agli amministratori di modificare alcune impostazioni dell’applicativo come ad esempio la soglia limite dei ritardi prima del recupero e altre opzioni. I docenti di classe potranno inserire studenti ed i relativi ritardi all’interno del sistema, sarà inoltre possibile inserire quando uno studente ha recuperato un determinato ritardo in modo da poterne tenere uno storico. Vi sarà inoltre la possibilità di inserire dei ritardi giustificati, ovvero dei ritardi che vengono mostrati nel conteggio totale ma non dovranno essere recuperati dallo studente. Una volta raggiunta la soglia massima di ritardi un sistema automatico invierà una e-mail di notifica allo studente informandolo che verrà contattato per eseguire il recupero del ritardo. Ogni semestre verrà eseguito un reset automatico del numero di ritardi degli studenti ma non dei ritardi da recuperare. L’applicativo dovrà inoltre essere caricato su un hosting.

DA\_REVISIONARE

## Analisi e specifica dei requisiti

È richiesto da parte del committente lo sviluppo di un applicativo web che sia accessibile tramite la rete e che sia web. Il prodotto dovrà possedere un sistema di autenticazione con un sistema di permessi integrato. Il sistema dovrà dunque distinguere i vari utenti in base ai loro permessi. È dunque richiesta una pagina di login attraverso la quale gli utenti andranno ad accedere all’applicativo. Gli amministratori del software potranno aggiungere ed eliminare gli utenti, alla creazione di un utente le credenziali per l’accesso verranno inviate in modo automatico attraverso un messaggio di posta elettronica e al primo accesso verrà richiesto all’utente di cambiare la propria password. In caso di perdita della propria password essa potrà essere recuperata attraverso il proprio indirizzo e-mail. Gli amministratori potranno aggiungere utenti amministratori oppure utenti limitati i quali potranno avere i seguenti permessi:

* Inserimento dei ritardi
* Visione dei dati
* Creazione dei PDF

Dovrà essere presente una pagina di amministrazione attraverso la quale sarà possibile configurare le date di inizio e di fine di ogni semestre, i nomi delle sezioni scolastiche, gli anni scolastici e la soglia di ritardi massimi dopo i quali essi andranno recuperati. I docenti avranno a disposizione una pagina attraverso la quale potranno registrare gli allievi che hanno cominciato ad accumulare dei ritardi e di conseguenza sarà possibile anche aggiungere ritardi ad allievi già registrati. Sarà disponibile un'altra pagina che permetterà ai docenti di inserire i ritardi recuperati da parte degli studenti. Di ogni studente verrà inoltre tenuto lo storico dei ritardi. Il prodotto dovrà essere caricato su un hosting.

DA\_REVISIONARE

|  |  |
| --- | --- |
| **ID: REQ-000** | |
| **Nome** | Piattaforma dell’applicativo |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Si necessita di un applicativo web. |
| **Sotto requisiti** | |
| **001** | Il sito deve funzionare su browser moderni (Esempio: Chrome). |

|  |  |
| --- | --- |
| **ID: REQ-001** | |
| **Nome** | Pagina di accesso |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Si necessita di una pagina che permetta di inserire le proprie credenziali per accedere all’applicativo web. |
| **Sotto requisiti** | |
| **001** | Si necessita di una maschera di login. |
| **002** | Si necessita di un modale di cambio password per nuovi utenti. |

|  |  |
| --- | --- |
| **ID: REQ-002** | |
| **Nome** | Pagina di recupero password |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Si necessita di una pagina che permetta di recuperare la propria password. |
| **Sotto requisiti** | |
| **001** | Dovrà essere inviata una e-mail di recupero password se l’utente che la richiede è esistente. |

|  |  |
| --- | --- |
| **ID: REQ-003** | |
| **Nome** | Pagina di gestione utenti |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Si necessita di una pagina che permetta di gestire gli utenti che avranno accesso all’applicativo web. |
| **Sotto requisiti** | |
| **001** | Dovrà essere possibile creare utenti. |
| **002** | Le credenziali verranno inviate per posta elettronica. |
| **003** | Dovrà essere possibile eliminare utenti. |
| **004** | Dovrà essere possibile modificare i permessi di ogni singolo utente. |
| **005** | Dovrà essere sempre presente almeno un utente amministratore. |
| **006** | Non dovrà essere possibile eliminare il proprio utente. |

|  |  |
| --- | --- |
| **ID: REQ-004** | |
| **Nome** | Pagina di amministrazione |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Si necessita di una pagina che permetta di gestire le impostazioni dell’applicativo web. |
| **Sotto requisiti** | |
| **001** | Dovrà essere possibile inserire le date di inizio e fine di ogni semestre. |
| **002** | Dovrà essere possibile inserire i nomi delle sezioni. |
| **003** | Dovrà essere possibile inserire gli anni scolastici. |
| **004** | Dovrà essere possibile modificare la soglia di ritardi massimi. (Di base tre compreso) |

|  |  |
| --- | --- |
| **ID: REQ-005** | |
| **Nome** | Pagina di aggiunta ritardi |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Si necessita di una pagina che permetta di aggiungere ritardi agli studenti. |
| **Sotto requisiti** | |
| **001** | Dovrà essere possibile inserire nuovi studenti. |
| **002** | Dovrà essere possibile aggiungere un ritardo ad uno studente. |
| **003** | Al raggiungimento della soglia massima dovrà essere inviata una e-mail di notifica. |
| **004** | Dovrà essere tenuto un istoriato dei ritardi. |

|  |  |
| --- | --- |
| **ID: REQ-006** | |
| **Nome** | Pagina di gestione recuperi |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Si necessita di una pagina che permetta di aggiungere i ritardi recuperati da parte degli studenti. |
| **Sotto requisiti** | |
| **001** | Dovrà essere possibile inserire il ritardo recuperato da parte dello studente. |

|  |  |
| --- | --- |
| **ID: REQ-007** | |
| **Nome** | Messa in produzione |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | L’applicativo dovrà essere messo in produzione su un hosting. |

DA\_REVISIONARE

## Use case

Gli agenti di questo schema si dividono in docente ed amministratore. Entrambi gli agenti hanno la possibilità di accedere alla pagina di accesso attraverso la quale sarà possibile accedere all’applicativo. Attraverso la pagina di login sarà possibile recarsi in un'altra pagina dedicata al recupero password. Entrambi gli agenti potranno eseguire azioni di gestione di recuperi e ritardi ovvero potranno creare degli studenti ed assegnarci dei ritardi. Ulteriormente potranno anche gestire i recuperi dei ritardi eseguiti dagli studenti. Successivamente solo gli amministratori potranno accedere alle pagine di gestione degli utenti dove sarà possibile aggiungere, modificare ed eliminare gli utenti presenti nell’applicativo. L’amministratore potrà accedere ad una ulteriore pagina di amministrazione che permetterà di modificare le impostazioni dell’applicativo come: i semestri, gli anni scolastici, le sezioni e la soglia massima dei ritardi. In questo caso l’utente docente ha tutti i permessi per accedere alle pagine di gestione dei ritardi, prima di tutte queste pagine viene eseguito un controllo dei permessi dell’utente per decidere se esso può accedere oppure no ad una determinata pagina. I permessi controllati saranno:

* Inserimento dei ritardi
* Visione dei dati
* Creazione dei PDF

DA\_REVISIONARE

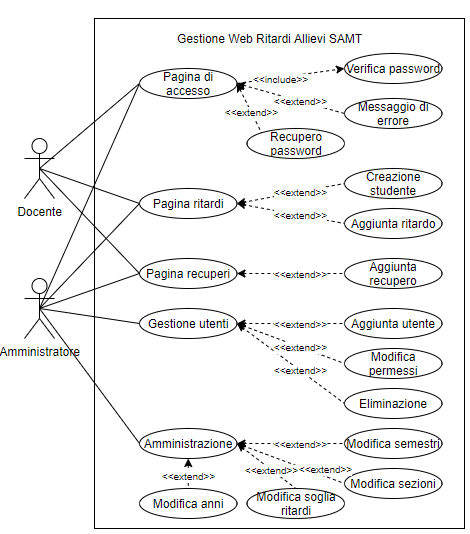


Figura 1 Schema caso d'uso.

## Pianificazione

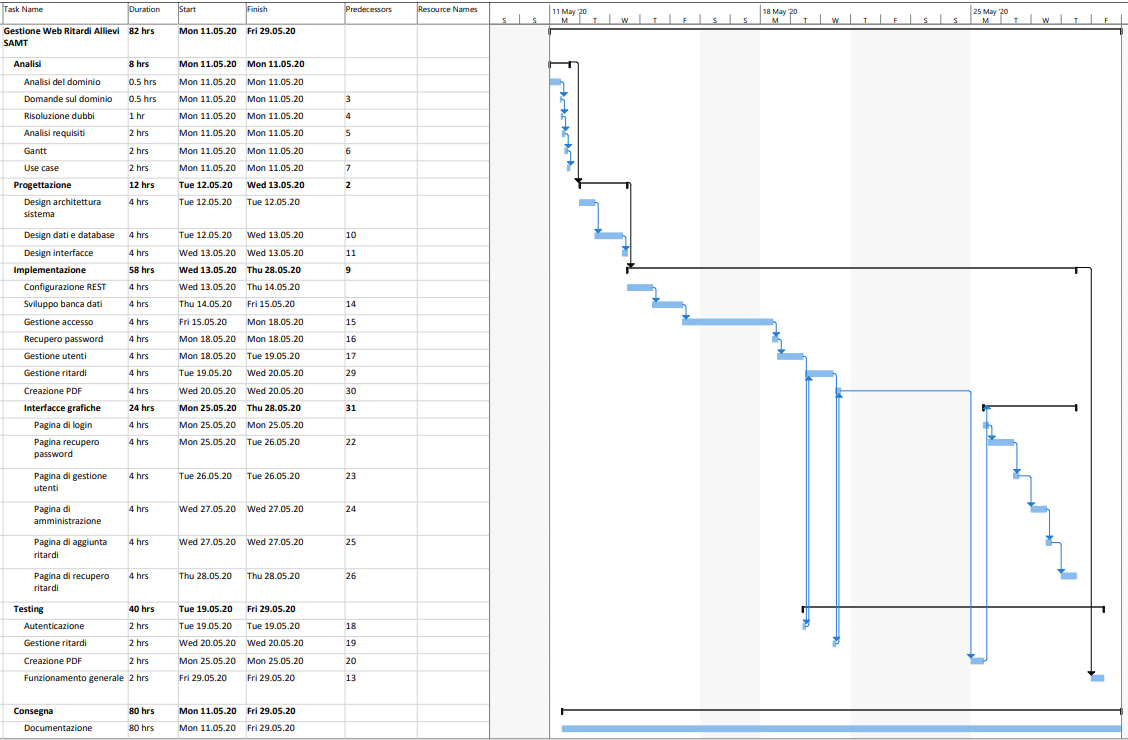


Figura 2 Diagramma di Gantt preventivo.

### Analisi

Ho suddiviso la fase di analisi in sei attività. Durante questa fase mi sono occupato di capire di cosa tratta il progetto e quali sono le richieste da parte del committente.

DA\_REVISIONARE

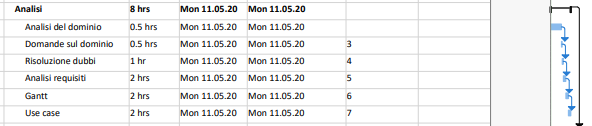


Figura 3 Diagramma di Gantt, Analisi.

### Progettazione

La progettazione è la seconda fase del progetto “Gestione Web Ritardi Allievi SAMT” ed è composta da tre attività. Questa fase contiene le attività di design della struttura del progetto. Considero questa fase molto importante in quanto le fasi successive saranno basate su di essa.

DA\_REVISIONARE

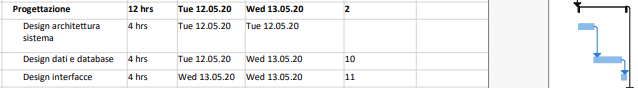


Figura 4 Diagramma di Gantt, Progettazione.

### Implementazione

La fase di implementazione è la più lunga all’interno del progetto. Questa fase consiste nella vera scrittura del codice che andrà a comporre il prodotto finale. Questa fase è strettamente collegata all’analisi e alla progettazione in quanto si basa su di esse per lo sviluppo.

DA\_REVISIONARE

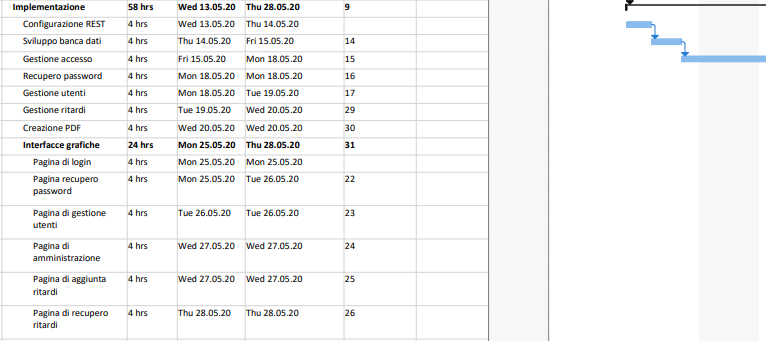


Figura 5 Diagramma di Gantt, Implementazione.

### Testing

Un’altra fase molto importante è il testing. All’interno di questa fase sono presenti i test che verranno eseguiti alle varie sezioni del progetto. Le attività di testing sono state generalizzate in modo tale da rendere il gantt leggibile, durante il corso del progetto verranno eseguiti test più approfonditi.

DA\_REVISIONARE



Figura 6 Diagramma di Gantt, Testing.

### Consegna

L’ultima fase del diagramma è la consegna del prodotto, all’interno di essa vi è una sola attività che riguarda la documentazione. Questa attività ha la durata dell’interno progetto perché verrà aggiornata nel corso dello sviluppo di esso.

DA\_REVISIONARE



Figura 7 Diagramma di Gantt, Consegna.

## Analisi dei mezzi

### Software

I software utilizzati per la realizzazione del progetto sono:

- Google Chrome 76.0

- Microsoft Word 265 18.2004

- Microsoft Project 2019

- Microsoft VS Code 1.37.1

- MySQL 8.0.13

- PHP 7.3.5

- Draw.io (<https://draw.io>)

- PaperCut 5.7.0

Librerie utilizzate:

- jQuery 3.4.1 (https://jquery.com/)

- Bootstrap 4.3.1 (https://getbootstrap.com/)

- php-rest (<https://github.com/filippofinke/php-rest>)

- FPDF (<http://www.fpdf.org/>)

Template utilizzati:

- SB Admin 2 (https://github.com/BlackrockDigital/startbootstrap-sb-admin-2)

Gestore librerie utilizzato:

- Composer 1.7.3 (<https://getcomposer.org/download/>)

DA\_REVISIONARE

### Hardware

Il progetto è stato sviluppato su un computer fisso.

Le specifiche hardware sono:

- 32 GB di RAM

- Intel Core I7-7700K 4 core

Il progetto potrà essere messo in produzione su una qualsiasi macchina con più di:

- 512MB di RAM

- 2GB di disco

DA\_REVISIONARE

# Progettazione

## Design dell’architettura del sistema

Questo è lo schema dell’architettura del sito web molto semplificato, l’applicativo web interagisce con il database in due modi. Il primo modo è diretto tramite l’utilizzo dei metodi messi a disposizione da PHP per l’interrogazione della banca dati MySQL. Mentre il secondo modo per interrogare la banca dati è attraverso delle richieste HTTP eseguite in modo asincrono da JavaScript. Per implementare questo sistema vengono dunque utilizzate due stili architetturali: REST e MVC. Vengono ricavati i dati in modo diretto in azioni che non vanno ad influenzare una buona esperienza utente, mentre dove vengono eseguite delle azioni da parte dell’utente (ad esempio l’aggiunta di un utente) vengono eseguite delle chiamate alle API REST in modo da offrire una migliore esperienza all’utente.

DA\_REVISIONARE

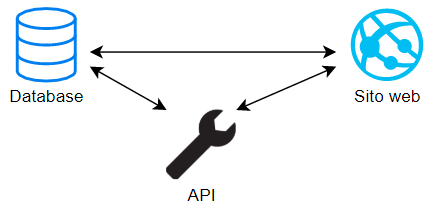


Figura 8 Architettura del sistema semplificata.

## Design dei dati e database

Tutti i dati riguardanti l’applicativo verranno salvati all’interno di un database MySQL.

DA\_REVISIONARE

### Schema ER

Questo è lo schema ER utilizzato dall’applicativo web. È composto da sette tabelle. La tabella “USER” viene utilizzata per salvare tutte le informazioni relative agli utenti che potranno utilizzare l’applicativo web. La tabella “TOKEN” viene utilizzata per immagazzinare i codici di recupero password che vengono generati quando un utente dimentica la propria password. La tabella “STUDENT” contiene tutti gli studenti che hanno fatto almeno un ritardo e che dunque sono stati inseriti dai propri docenti di classe. Successivamente in correlazione con gli studenti sono presenti due tabelle, la prima “SECTION” rappresenta la sezione e classe di uno studente, mentre la seconda tabella “DELAY” contiene tutti i ritardi che sono stati fatti dai vari alunni con varie informazioni. Vi sono ulteriori due tabelle a sé rispettivamente “YEAR” la quale contiene i vari semestri per i vari anni e la tabella “SETTING” che contiene le impostazioni del sito web in formato: nome impostazione e valore.

DA\_REVISIONARE

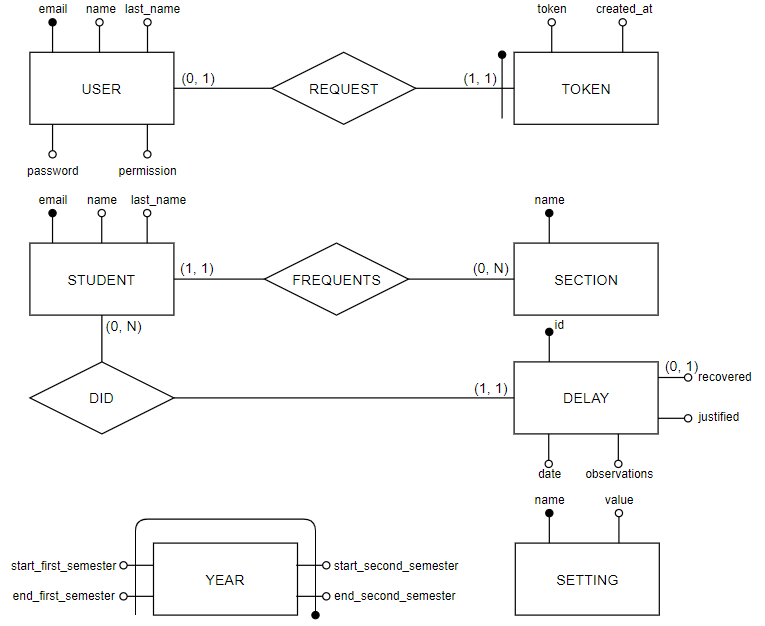


Figura 9 Schema ER banca dati.

### Descrizioni delle tabelle

|  |  |
| --- | --- |
| **USER** | |
| Attributo | Descrizione |
| email | Rappresenta un indirizzo e-mail dell’utente. È un attributo di tipo stringa con limite di 255 caratteri. Non può essere nullo e deve essere univoco.  Esempio: filippo.finke@samtrevano.ch |
| name | Rappresenta il nome di un utente. È un attributo di tipo stringa con un limite di 20 caratteri. Può contenere solamente lettere dell’alfabeto e non può essere nullo.  Esempio: Filippo |
| last\_name | Rappresenta il cognome di un utente. È un attributo di tipo stringa con un limite di 20 caratteri. Può contenere solamente lettere dell’alfabeto e non può essere nullo.  Esempio: Finke |
| password | Rappresenta la password di un utente. È un attributo di tipo stringa con limite di 60 caratteri. Non può essere nullo. All’interno di questo attributo verrà salvata un hash della password dell’utente.  Esempio: $2y$10$NmiaiLmr3dhUg3ePIExyt.l2KvE7SK6le1UH67QVikBlyBjjTHgVG |
| permission | Rappresenta il permesso di un utente. È un attributo di tipo intero e non può esssere nullo. Può contenere i seguenti valori:  1 - Inserimento ritardi.  2 - Visione ritardi.  4 - Creazione PDF.  8 - Amministratore.  Esempio: 8 |

|  |  |
| --- | --- |
| **TOKEN** | |
| Attributo | Descrizione |
| token | Rappresenta un codice che verrà utilizzato per eseguire il recupero della password. Questo codice verrà generato in modo casuale. All’interno dell’attributo verrà salvata un hash in SHA256 del token di recupero password. Il campo è di tipo stringa con un limite di 64 caratteri e non può essere nullo.  Esempio: 4b9eb466ad2615314c8194b1c46e6ef8e910d0b41a682e32de73503883e09b58 |
| created\_at | Rappresenta la data di creazione del codice di recupero password. È un attributo di tipo DATETIME e non può essere nullo.  Esempio: 2020-05-12 14:16:59 |

|  |  |
| --- | --- |
| **STUDENT** | |
| Attributo | Descrizione |
| email | Rappresenta un indirizzo e-mail dell’utente. È un attributo di tipo stringa con limite di 255 caratteri. Non può essere nullo e deve essere univoco.  Esempio: filippo.finke@samtrevano.ch |
| name | Rappresenta il nome di un utente. È un attributo di tipo stringa con un limite di 20 caratteri. Può contenere solamente lettere dell’alfabeto e non può essere nullo.  Esempio: Filippo |
| last\_name | Rappresenta il cognome di un utente. È un attributo di tipo stringa con un limite di 20 caratteri. Può contenere solamente lettere dell’alfabeto e non può essere nullo.  Esempio: Finke |

|  |  |
| --- | --- |
| **SECTION** | |
| Attributo | Descrizione |
| name | Rappresenta il nome della sezione. È un attributo di tipo stringa, ha un limite di 10 caratteri e non può essere nullo.  Esempio: SAM I4AC |

|  |  |
| --- | --- |
| **DELAY** | |
| Attributo | Descrizione |
| id | Rappresenta l’identificatore di un ritardo. È un attributo di tipo intero e viene impostato in modo automatico dall’applicativo. Non può essere nullo e deve essere univoco.  Esempio: 1 |
| recovered | Rappresenta la data di recupero del ritardo. È un attributo di tipo DATE e può essere nullo.  Esempio: 2020-05-12 |
| justified | Flag che indica se il ritardo è giustificato oppure no.  Esempio: 0 |
| date | Rappresenta la data del ritardo. È un attributo di tipo DATE e non può essere nullo.  Esempio: 2020-05-12 |
| observations | Rappresenta le osservazioni correlate con il ritardo. È un attributo di tipo stringa con un limite di 255 valori, può contenere qualsiasi carattere e può essere nullo.  Esempio: Ritardo causa treni. |

|  |  |
| --- | --- |
| **SETTING** | |
| Attributo | Descrizione |
| name | Rappresenta il nome dell’impostazione. È un attributo di tipo stringa, ha un limite di 50 caratteri e non può essere nullo.  Esempio: website\_title |
| value | Rappresenta il valore dell’impostazione. È un attributo di tipo stringa, ha un limite di 255 caratteri e non può essere nullo.  Esempio: Gestione ritardi |

|  |  |
| --- | --- |
| **YEAR** | |
| Attributo | Descrizione |
| start\_first\_semester | Rappresenta la data di inizio del primo semestre. È un attributo di tipo DATE.  Esempio: 2019-05-10 |
| end\_first\_semester | Rappresenta la data di fine del primo semestre. È un attributo di tipo DATE.  Esempio: 2020-05-11 |
| start\_second\_semester | Rappresenta la data di inizio del primo semestre. È un attributo di tipo DATE.  Esempio: 2020-05-11 |
| end\_second\_semester | Rappresenta la data di fine del primo semestre. È un attributo di tipo DATE.  Esempio: 2020-09-12 |

DA\_REVISIONARE

### Schema logico

Questo è lo schema logico del database:

user(email, name, last\_name, password, permission)

token(email(FK), token, created\_at)

setting(name, value)

section(name)

student(email, name, last\_name, section(FK))

delay(id, email(FK), date, observations\*, recovered\*, justified)

year(start\_first\_semester, end\_first\_semester, start\_second\_semester, end\_second\_semester)

DA\_REVISIONARE

## Design delle interfacce

### Pagina di accesso

Questo è un mockup della pagina utilizzata per accedere all’applicativo web. Attraverso questa pagina è quindi possibile inserire username e password per poter accedere tramite il database locale.

DA\_REVISIONARE

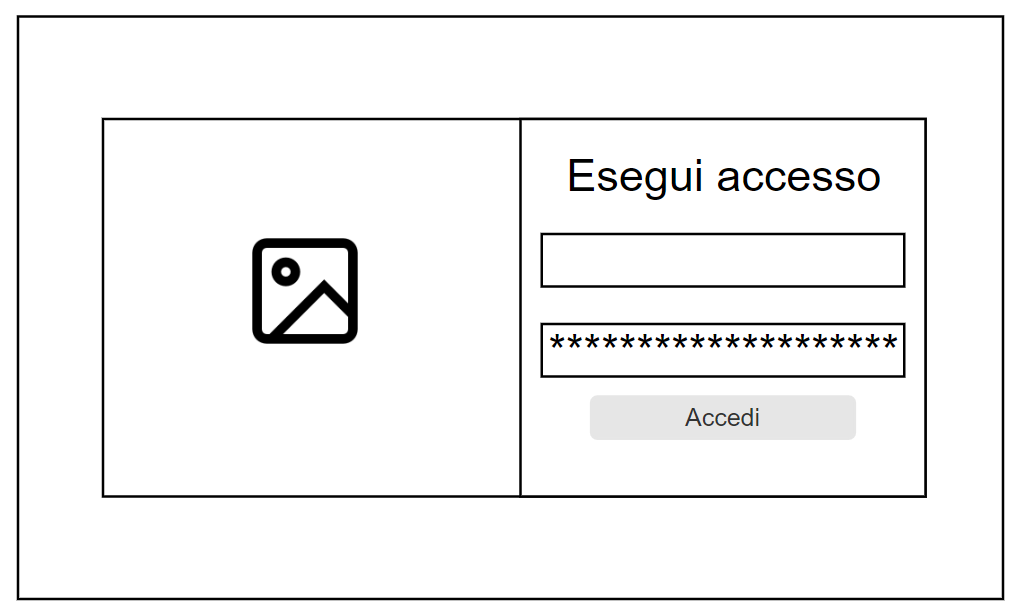


Figura Mockup pagina di accesso.

### Pagina cambio password

Questo è un mockup della pagina utilizzata per cambiare la password del proprio account utente all’interno dell’applicativo web.

DA\_REVISIONARE

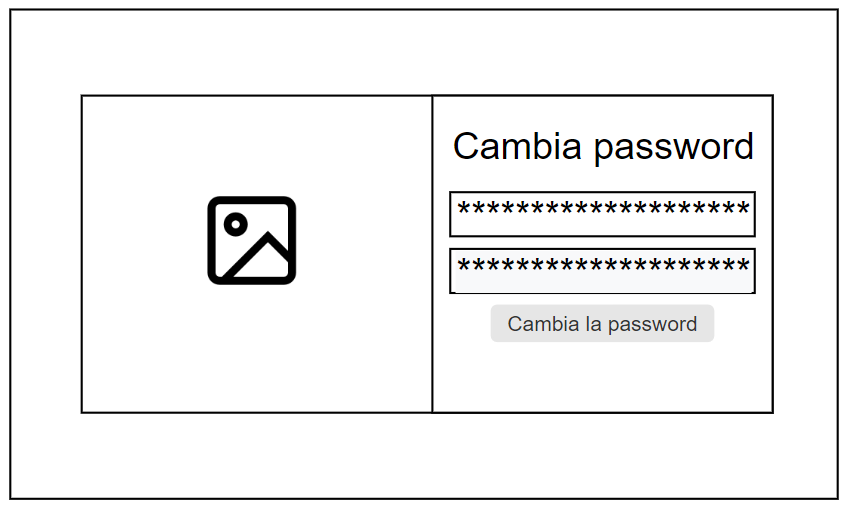


Figura Mockup pagina di cambio password.

### Pagina di recupero password

Questo è un mockup della pagina utilizzata per richiedere una e-mail di recupero password.

DA\_REVISIONARE



Figura Mockup pagina di recupero password.

### Pagina di aggiunta ritardi

Questo è un mockup della pagina utilizzata per aggiungere ritardi agli studenti.

DA\_REVISIONARE

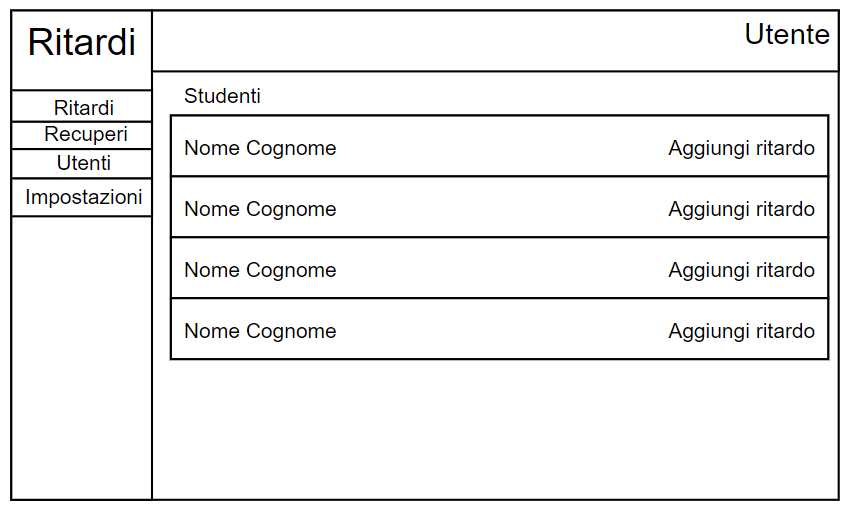


Figura Mockup pagina di aggiunta ritardi.

### Pagina di aggiunta recuperi

Questo è un mockup della pagina utilizzata per aggiungere recuperi di ritardi agli studenti.

DA\_REVISIONARE

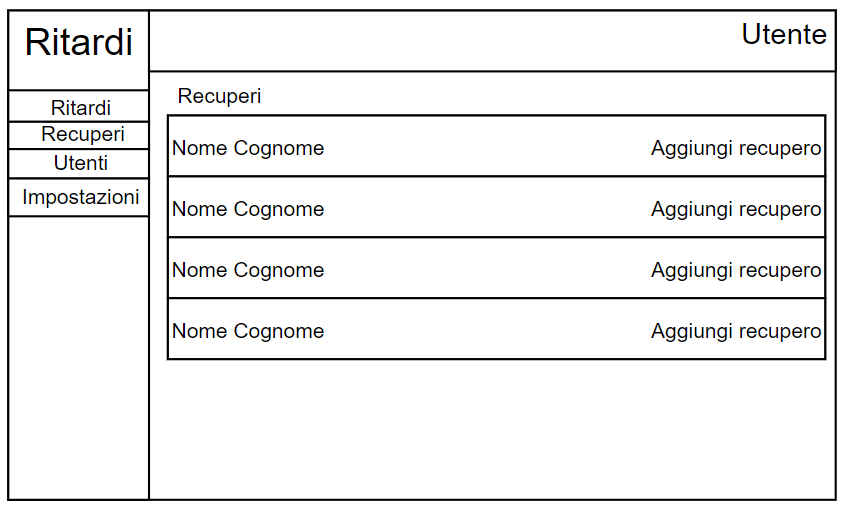


Figura Mockup pagina di aggiunta recuperi.

### Pagina di gestione utenti

Questo è un mockup della pagina utilizzata per gestire gli utenti all’interno dell’applicativo web.

DA\_REVISIONARE

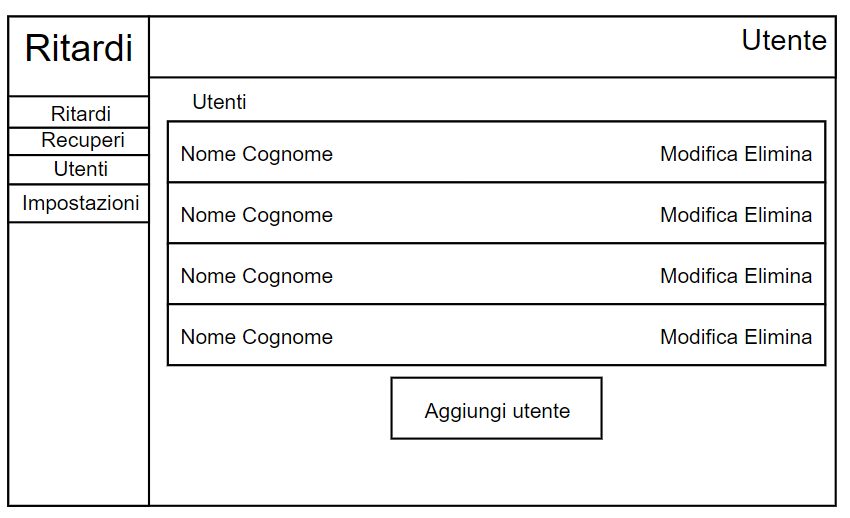


Figura Mockup pagina di gestione utenti.

### Pagina impostazioni

Questo è un mockup della pagina utilizzata per cambiare le impostazioni dell’applicativo web.

DA\_REVISIONARE

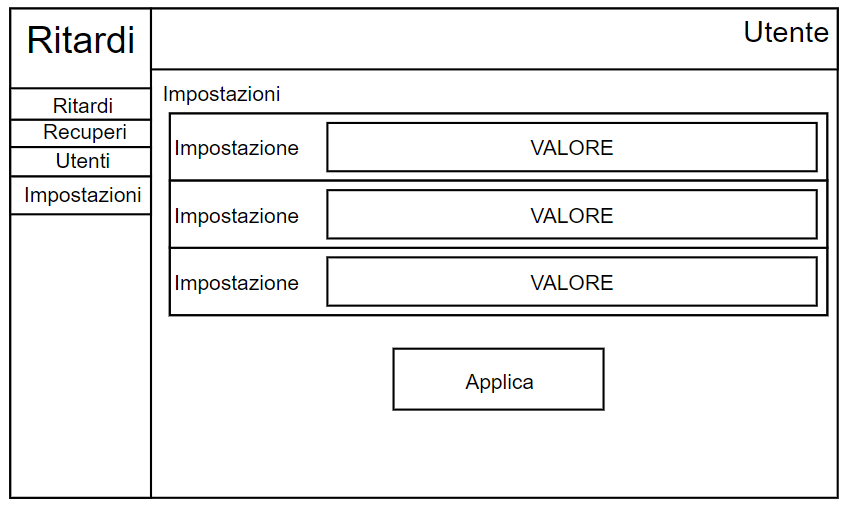


Figura Mockup pagina impostazioni.

# Implementazione

DA\_FARE

# Test

## Protocollo di test

DA\_FARE

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-001  REQ-012 | **Nome:** | Import a card with KIC, KID and KIK keys, but not shown with the GUI |
| **Descrizione:** | Import a card with KIC, KID and KIK keys with no obfuscation, but not shown with the GUI | | |
| **Prerequisiti:** | Store on local PC: Profile\_1.2.001.xml (appendix n\_n) and Cards\_1.2.001.txt (appendix n\_n).  PIN (OTA\_VIEW\_PIN\_PUK\_KEY) and ADM (OTA\_VIEW\_ADM\_KEY) user right not set. | | |
| **Procedura:** | 1. Go to “Cards manager” menu,  in main page click “Import Profiles” link, Select the “1.2.001.xml” file, Import the Profile 2. Go to “Cards manager” menu,  in main page click “Import Cards” link, Select the “1.2.001.txt” file, Delete the cards,  Select the “1.2.001.txt” file, Import the cards 3. Research the “41795924770” Card, Click the imsi card link Check the card details 4. Execute the SQL: SELECT imsi, dir, keyset, cntr, rawtohex(kickey), rawtohex(kidkey), rawtohex(kikkey), rawtohex(chv), rawtohex(dap)FROM otacardkey a where imsi='340041795924770' ORDER BY keyset; | | |
| **Risultati attesi:** | Keys visible in the DB (OtaCardKey) but not visible in the GUI (Card details) | | |

## Risultati test

DA\_FARE

## Mancanze/limitazioni conosciute

DA\_FARE

# Consuntivo

DA\_FARE

# Conclusioni

DA\_FARE

## Sviluppi futuri

DA\_FARE

## Considerazioni personali

DA\_FARE

# Sitografia

DA\_FARE

# Allegati

DA\_FARE

DA\_REVISIONARE = Completato ma da controllare.

IN\_CORSO = Documentato parzialmente.

DA\_FARE = Da documentare.