

## Documentazione

# Sito web con pagina di registrazione e verifica dati

**Titolo del progetto:**

Sito web con pagina di registrazione e verifica dati

**Alunno/a:**

Filippo Finke

**Classe:**

Info I3AC

**Anno scolastico:**

2018/2019

**Docente responsabile:**

Francesco Mussi, Adriano Barchi, Luca Muggiasca, Elisa Nannini

1	Introduzione.....	3
1.1	Informazioni sul progetto .....	3
1.2	Abstract .....	3
1.3	Scopo .....	3
Analisi.....		3
1.4	Analisi del dominio .....	3
1.5	Analisi e specifica dei requisiti .....	3
1.6	Analisi dei costi.....	6
1.7	Pianificazione .....	6
1.8	Analisi dei mezzi.....	7
1.8.1	Software .....	7
1.8.2	Hardware.....	7
2	Progettazione .....	7
2.1	Design dell'architettura del sistema .....	7
2.2	Design dei dati e database.....	8
2.3	Design delle interfacce .....	9
2.4	Design procedurale .....	12
3	Implementazione .....	13
3.1	PreventBackHistory.....	13
3.2	Pagina di benvenuto .....	13
3.3	Pagina di registrazione.....	14
3.4	Pagina di controllo.....	15
3.5	Pagina finale.....	16
4	Test.....	17
4.1	Protocollo di test.....	17
4.2	Risultati test.....	21
4.3	Mancanze/limitazioni conosciute.....	21
5	Consuntivo.....	22
6	Conclusioni .....	22
6.1	Sviluppi futuri.....	22
6.2	Considerazioni personali.....	22
7	Bibliografia.....	23
7.1	Sitografia .....	23
8	Allegati.....	23

## 1 Introduzione

### 1.1 Informazioni sul progetto

Titolo del progetto: Sito web con pagina di registrazione e verifica dati

Alunno: Filippo Finke

Classe: I3AC

Docenti responsabili: Francesco Mussi, Adriano Barchi, Luca Muggiasca, Elisa Nannini

Data d'inizio: 07.09.2018

Data di fine: 09.11.2018

### 1.2 Abstract

*We see a lot of websites that request user's data and store it for personalized service. This web application allow users to compile a form with personal information and send these information to the owner of the application that store it and can use them for internal and commercial purposes, like survey, contest,...*

### 1.3 Scopo

Il prodotto deve poter ricavare delle informazioni inserite da un utente attraverso un sito web e poterle salvare all'interno di due file di formato csv. Le informazioni possono poi essere utilizzate dal gestore del sito web.

### Analisi

#### 1.4 Analisi del dominio

Questo prodotto è stato ideato con l'intento di avere un sito web per raccogliere in modo molto semplice di dati comuni di utenti. Il contesto di questo prodotto può essere vago, come per esempio invio di newsletter, statistica,... Attualmente il problema viene risolto da siti web molto avanzati (Come per esempio cognitiforms), esistono già alcuni prodotti simili. Gli utenti sono possibili acquirenti di prodotti che non per forza hanno un lavoro. Gli utenti non devono avere delle competenze avanzate per l'utilizzo del prodotto, servono solamente delle nozioni base che permettono all'utente di collegarsi al prodotto in rete. Per operare e eseguire cambiamenti al prodotto si devono avere delle conoscenze mediocri di vari linguaggi di programmazione che vengono utilizzati, fra qui: html, css, php.

#### 1.5 Analisi e specifica dei requisiti

Il committente ha bisogno di avere una pagina web tramite il quale ricavare dati specifici e poterli salvare all'interno di file csv in modo automatico. Il prodotto dovrà guidare l'utente al completamento di questi campi passando da una pagina nel quale sarà possibile compilare i campi richiesti dal committente nel quale verranno eseguite varie validazioni su di essi in seguito una pagina di conferma dei dati ed in fine ad una registrazione che salverà il tutto. L'interazione dell'utente deve avvenire attraverso un browser ovvero un programma che permette la navigazione su internet. Il prodotto deve avere un interfaccia semplice ed intuitiva in modo da non confondere l'utente. Il prodotto deve poter gestire il tutto senza un consumo elevato di risorse, inoltre non sono richieste delle sicurezze oltre alla validazione dei campi all'interno del dominio. Il software sarà ideato in modo da poter funzionare su Chrome, Firefox, Opera e Edge.

ID: REQ-1	
<b>Nome</b>	Gestione file csv
<b>Priorità</b>	1
<b>Versione</b>	1.0
<b>Note</b>	Si necessita di permessi per lettura e scrittura.
<b>Sotto requisiti</b>	
<b>001</b>	I file devono essere salvati all'interno della cartella Registrazioni.
<b>002</b>	Scrittura dei dati in un file con "Registrazione_tutte.csv".
<b>003</b>	Scrittura dei dati per giorno corrente in un file chiamato "Registrazione_aaaa-mm-gg.csv".

ID: REQ-2	
<b>Nome</b>	Classi per la validazione dei dati lato server
<b>Priorità</b>	1
<b>Versione</b>	1.0
<b>Note</b>	Si necessita di un account amministratore per modificare i servizi. / Dipende dal requisito REQ-1.
<b>Sotto requisiti</b>	
<b>001</b>	Dovrà essere possibile controllare i campi.
<b>002</b>	Dovrà essere possibile controllare dati obbligatori.

ID: REQ-3	
<b>Nome</b>	Validazione campi lato client
<b>Priorità</b>	1
<b>Versione</b>	1.0
<b>Note</b>	Dipende dal requisito REQ-5.
<b>Sotto requisiti</b>	
<b>001</b>	Dovrà essere possibile controllare i campi prima dell'invio dei dati.
<b>002</b>	Si dovrà avvisare l'utente per la mancanza di campi obbligatori.
<b>003</b>	Dovrà essere possibile controllare i campi obbligatori.

ID: REQ-4	
<b>Nome</b>	Pagina di benvenuto
<b>Priorità</b>	3
<b>Versione</b>	1.0
<b>Note</b>	Pagina principale del sito con la presenza di un tasto “Registrati” che porta al form di registrazione. Dipende dal requisito REQ-0.

ID: REQ-5	
<b>Nome</b>	Interfaccia di registrazione
<b>Priorità</b>	2
<b>Versione</b>	1.0
<b>Note</b>	Dipende dal requisito REQ-1, REQ-2, REQ-3.
<b>Sotto requisiti</b>	
<b>001</b>	Possibilità di resettare i dati immessi nell’interfaccia con un tasto “Cancella” .
<b>002</b>	Quando i dati vengono inviati si viene reindirizzati ad una pagina nel quale confermare i dati.

ID: REQ-6	
<b>Nome</b>	Pagina di controllo
<b>Priorità</b>	1
<b>Versione</b>	1.0
<b>Note</b>	Pagina che permette all’utente di verificare i dati immessi. Dipende dal requisito REQ-5.
<b>Sotto requisiti</b>	
<b>001</b>	Possibilità di correggere i dati inseriti con un tasto “Correggi”.
<b>002</b>	Possibilità di eseguire la registrazione dell’utente con un tasto “Registrazione”

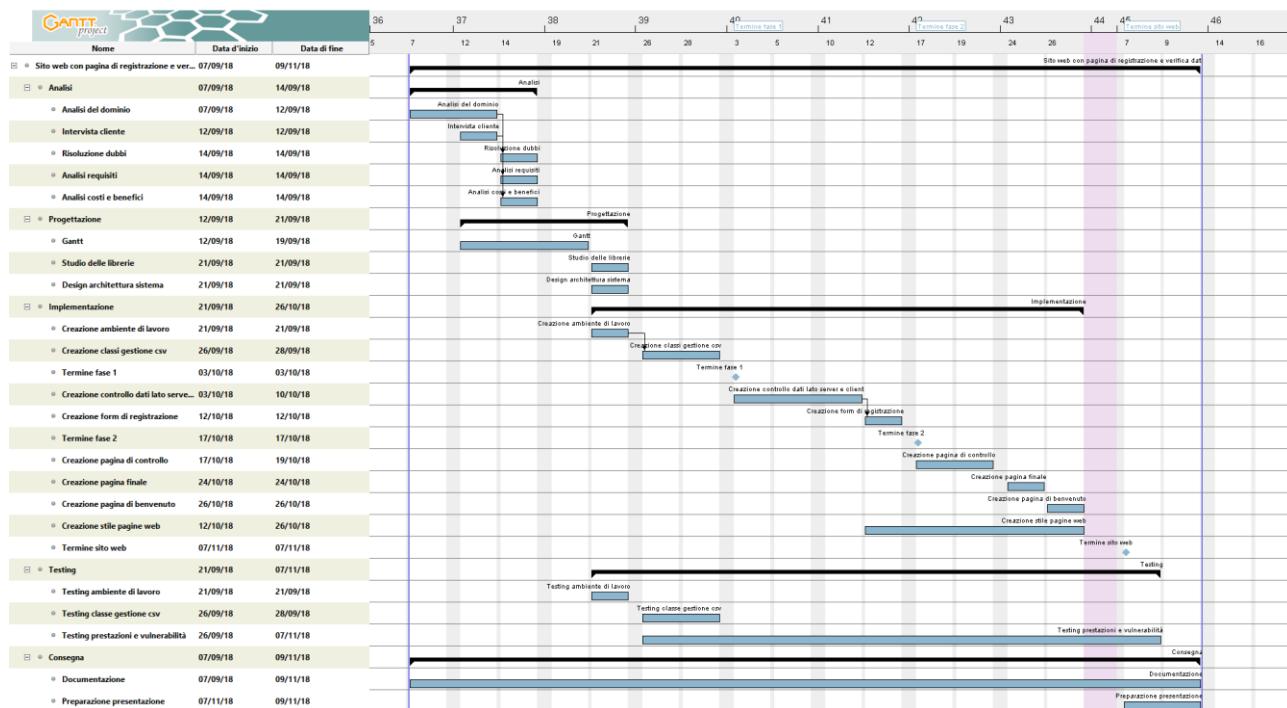
ID: REQ-7	
Nome	Pagina finale con visualizzazione dati
Priorità	1
Versione	1.0
Note	Pagina che visualizza i dati immessi dall'utente. Dipende dal requisito REQ-6.
<b>Sotto requisiti</b>	
001	I dati devono venire disposti come del form di registrazione.
002	I dati devono essere letti dal file csv.

## 1.6 Analisi dei costi

Categoria	Costo	Ore totali	Costo Totale
Mano d'opera	62 CHF/h	38h	2356CHF

## 1.7 Pianificazione

La fase che richiede maggior tempo per lo sviluppo del progetto è l'implementazione, nel quale ci sono le parti fondamentali per lo sviluppo di esso. Inoltre come punto fondamentale del progetto è segnata la fine del sito web con software lato server e client funzionanti.



## 1.8 Analisi dei mezzi

### 1.8.1 Software

Oracle VM VirtualBox 5.2.4 r119785 (Qt5.6.2)  
 SublimeText 3.0  
 Ubuntu Server 18.04.1 LTS  
 Apache 2.4  
 PHP 7.0  
 GanttProject 2.8.9  
 Microsoft Word 2016  
 Draw.io  
 Notify.js  
 jQuery-3.3.1.js  
 MaterializeCSS 1.00

### 1.8.2 Hardware

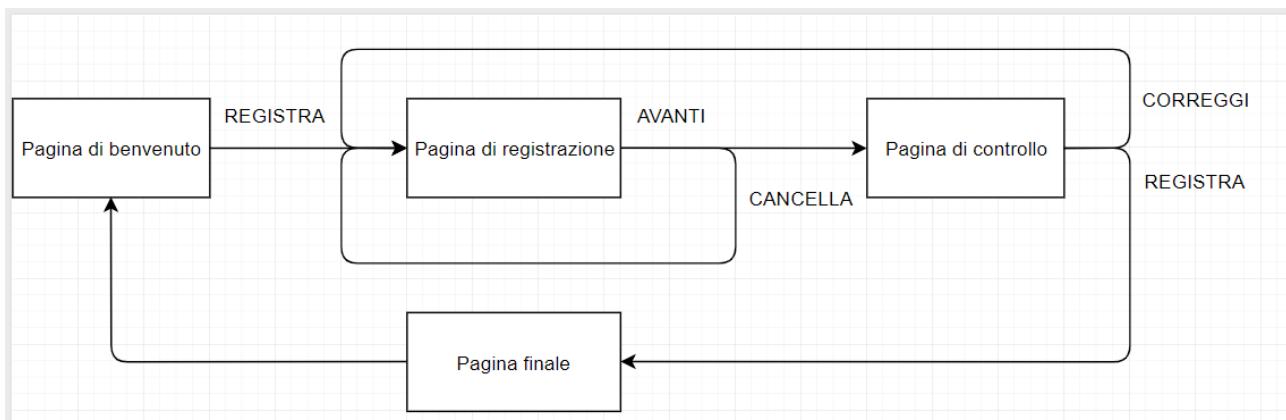
Computer portatile (Utilizzato per lo sviluppo del progetto)

- ASUS GL502VSK
- 16GB Ram
- i7-7700HQ
- GTX 1070 GDDR5
- Windows 10 Home

## 2 Progettazione

---

### 2.1 Design dell'architettura del sistema



Quando l'utente accede all'applicativo dal WEB si ritroverà alla pagina di benvenuto dal quale può decidere se procedere a registrarsi. In caso volesse registrarsi verrà reindirizzato ad una pagina di registrazione nel quale l'utente può inserire dati specifici, una volta completato il form può decidere se cancellare il form che rimanderà l'utente alla pagina di registrazione vuota, oppure procede. In caso l'utente abbia scelto di procedere verrà mandato ad una pagina di controllo nel quale può decidere di correggere i dati immessi, quindi tornare al form di registrazione oppure di confermare la registrazione. In caso di registrazione l'utente verrà rimandato ad una pagina finale nel quale potrà vedere tutti gli utenti registrati nel suo stesso giorno.

## 2.2 Design dei dati e database

I dati degli utenti devono essere salvati all'interno dei file con la seguente gerarchia:

Registrazioni

- └── Registrazione\_aaaa-mm-gg.csv
- └── Registrazioni\_tutte.csv

I file verranno salvati utilizzando il seguente formato:

`date;name;lastname;birthdate;gender;street;civicnumber;nap;city;telephone;email;hobby;occupation`

Il campo date viene assegnato in modo automatico dall'applicativo. Corrisponde alla data e ora di registrazione dell'utente.

Campo	Obbligatorio	Limite	Descrizione
Nome	Si	50 caratteri	Può contenere solamente lettere dell'alfabeto, lettere accentate e i caratteri “.” o “-“ una sola volta all'interno della stringa. Non può essere nullo.
Cognome	Si	50 caratteri	Può contenere solamente lettere dell'alfabeto, lettere accentate e i caratteri “.” o “-“ una sola volta all'interno della stringa. Non può essere nullo.
Data di nascita	Si		Deve essere una data nel formato: aaaa-mm-gg ES: 2018-10-13 La data non può essere nel futuro. Non può essere nullo.
Via	Si	50 caratteri	Può contenere solamente lettere dell'alfabeto, lettere accentate e i caratteri “.” o “-“ una sola volta all'interno della stringa. Non può essere nullo.
Numero civico	Si	4 caratteri	Può contenere numeri e opzionalmente una sola lettera dell'alfabeto. ES: 12C Non può essere nullo.
Città	Si	50 caratteri	Può contenere solamente lettere dell'alfabeto, lettere accentate e i caratteri “.” o “-“ una sola volta all'interno della stringa. Non può essere nullo.
Nap	Si	5 caratteri	Può contenere solamente numeri. Ha un minimo di 4 caratteri fino ad un massimo di 5.
Telefono	Si	13 caratteri	Può contenere solamente numeri e il carattere “+”. Deve avere un minimo di 10 caratteri numerici fino ad un massimo di 13.
Email	Si		Non può contenere caratteri speciali e deve avere al suo interno una “@” e un “.”. Deve rispettare il formato email. Non può essere nullo.
Genere	Si	1 carattere	Può contenere solamente “M” o “F”. Non può essere nullo.
Hobby	No	500 caratteri	Ha un limite di 500 caratteri e può essere nullo.
Professione	No	500 caratteri	Ha un limite di 500 caratteri e può essere nullo.

## 2.3 Design delle interfacce

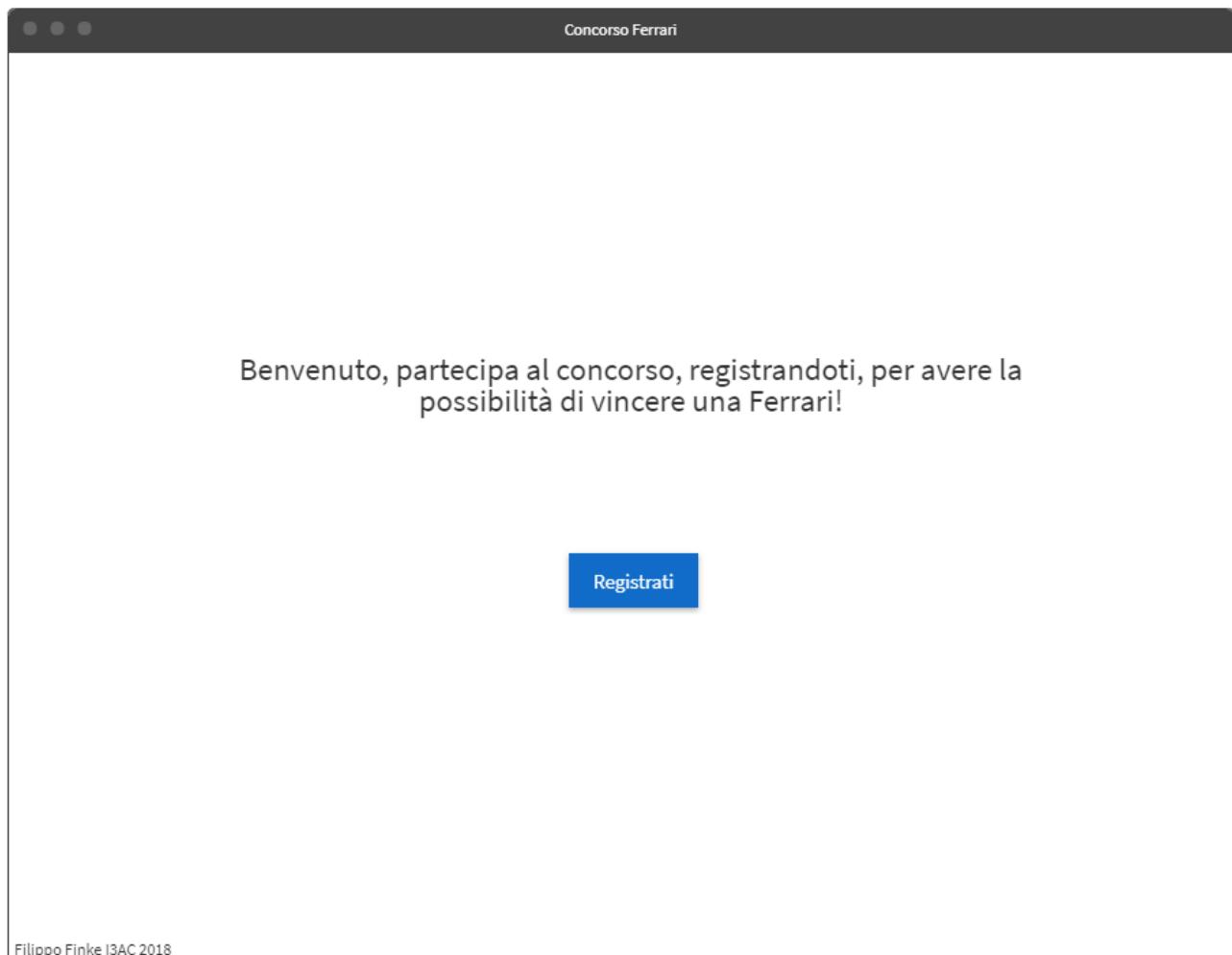


Figura 1 Pagina di benvenuto

Concorso Ferrari

Completa il modulo per partecipare al concorso!

Nome*	Cognome*
 Filippo	Finke
Data di nascita*	Genere*
 12 May 2016 	<input checked="" type="radio"/> Maschio <input type="radio"/> Femmina
Via*	Numero civico*
 Via Monte	12C
NAP*	Città*
 6828	Balerna
 Numero di telefono*	
 Indirizzo email*	
 Hobby	
 Professione	

I campi con \* sono obbligatori.

 Cancella  Avanti

Filippo Finke I3AC 2018

Figura 2 Pagina di registrazione

Concorso Ferrari

Verifica i tuoi dati

Nome	Cognome	Data di nascita	Via	Numero civico	Città	NAP	Numero di telef...	Email	Genere	Hobby	Professione
Filippo	Finke	22-12-2001	Via Monte	12C	Balerna	6828	+41091999999	filippo@finke.c...	Maschio	Programmazione	Apprendista

[Correggi](#) [Registra](#)

Filippo Finke I3AC 2018

Figura 3 Pagina di controllo

Concorso Ferrari

Registrazioni giornaliere

Nome	Cognome	Data di nascita	Via	Numero civico	Città	NAP	Numero di telef...	Email	Genere	Hobby	Professione
Filippo	Finke	22-12-2001	Via Monte	12C	Balerna	6828	+41091999999	filippo@finke.c...	Maschio	Programmazione	Apprendista
Filippo	Finke	22-12-2001	Via Monte	12C	Balerna	6828	+41091999999	filippo@finke.c...	Maschio	Programmazione	Apprendista

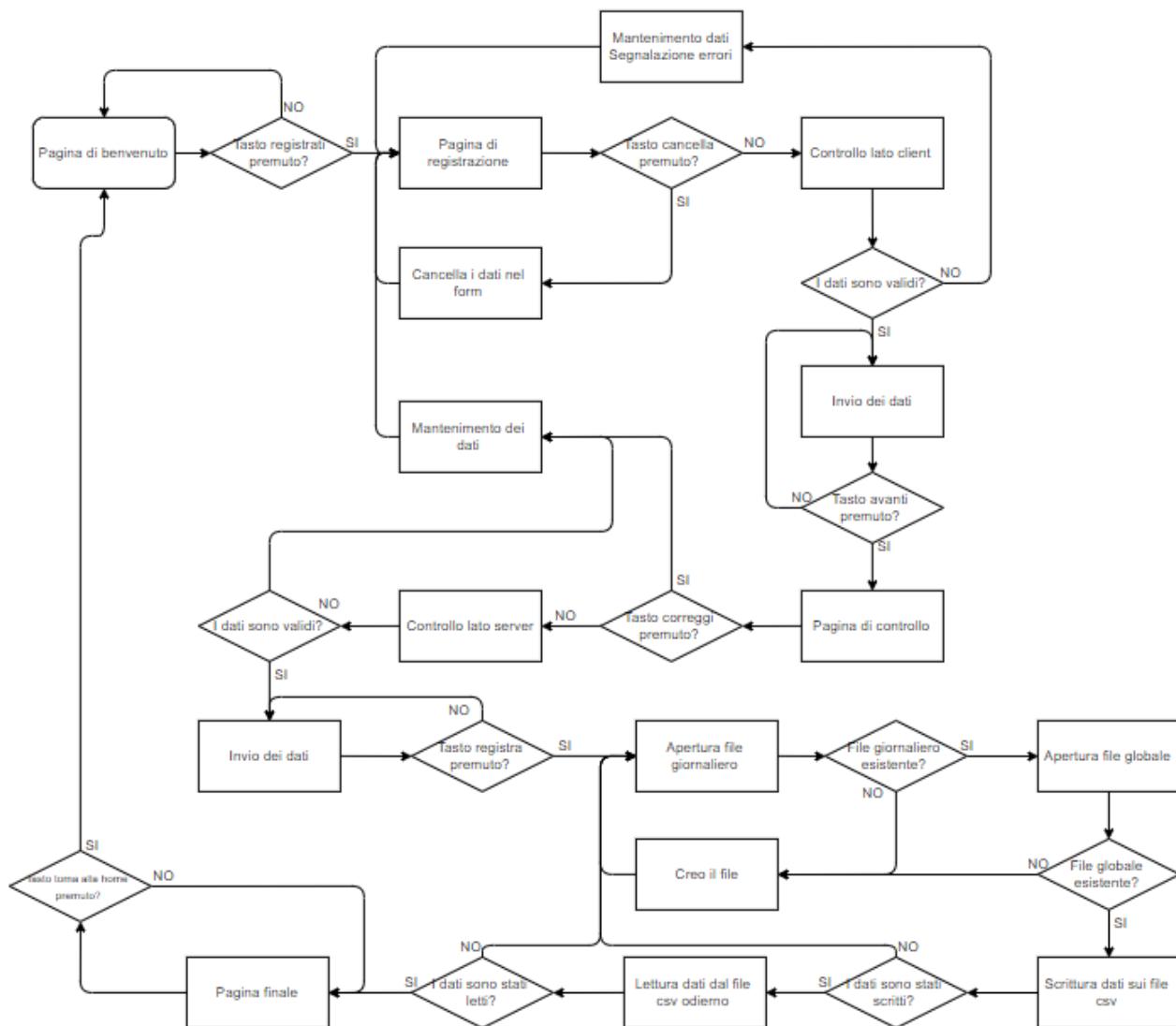
[Torna alla home](#)

Filippo Finke I3AC 2018

Figura 4 Pagina finale

## Sito web con pagina di registrazione e verifica dati

### 2.4 Design procedurale



CsvManager
- path: string
- delimiter: char
+ __construct(string path, string delimiter):
+ writeLine(array line):bool
+ readAll():array

Validator
- delimiter: char
- disabledCharsAtStart: string[]
+ containChar(string, char): bool
+ startsWith(string, char): bool
+ basic(string): bool
+ gender(string): bool
+ telephone(string): bool
+ nap(string): bool
+ birthDate(string, string): bool
+ general(string): bool
+ civicnumber(string): bool
+ email(string): bool
+ textArea(string): bool

### 3 Implementazione

#### 3.1 PreventBackHistory

All'interno delle pagine registrazione.php, controllo.php e resoconto.php utilizzo un file javascript esterno chiamato preventBackHistory.js che permette di bloccare l'utente di poter tornare alla pagina precedente e rieseguire delle richieste già complete in modo autonomo dal browser. All'interno di questo file si trova il seguente codice:

```
history.pushState(null, null, location.href);
window.onpopstate = function () {
    history.go(1);
};
```

Questo blocco di codice imposta a null la cronologia della pagina in modo da non permettere all'utente di tornare indietro.

#### 3.2 Pagina di benvenuto

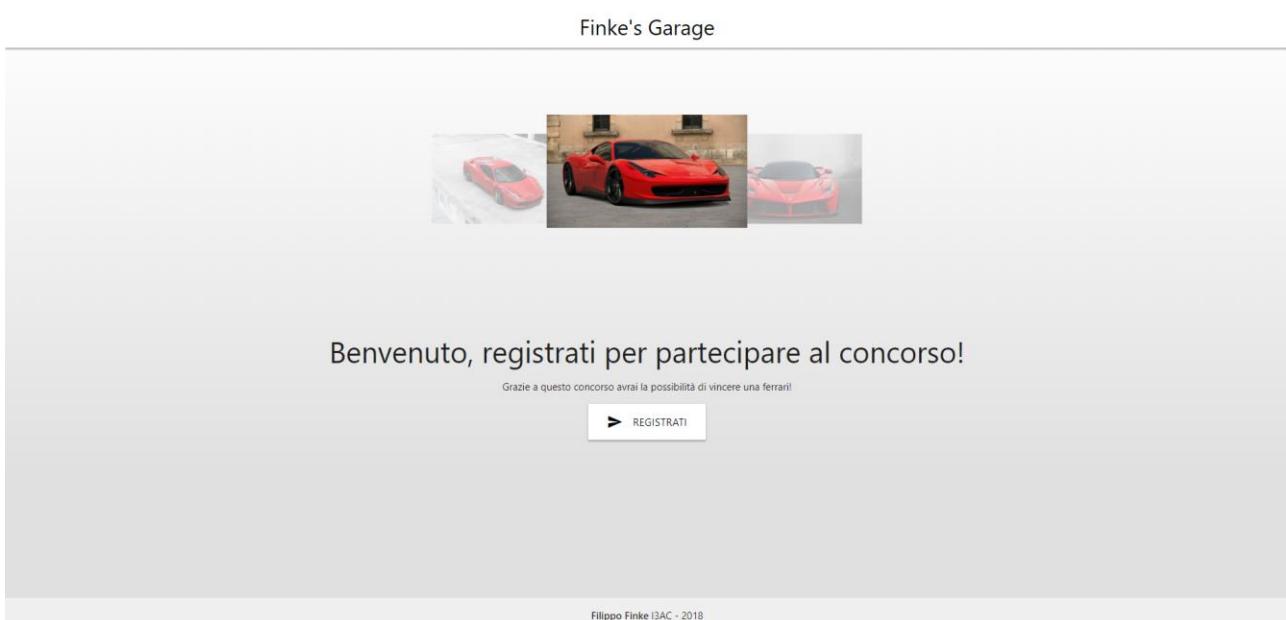


Figura 5 Pagina di benvenuto

Questa è la pagina di benvenuto dell'applicativo web. All'interno di questa pagina vi è uno slider gestito dalla libreria Materialize e viene creato ed applicato utilizzando il seguente blocco di codice:

```
$('.carousel').carousel({
  duration: 250
});
var inst = M.Carousel.getInstance($('.carousel'));
setInterval(function() {
  inst.next();
}, 2000);
```

In questo modo viene creato uno slider di immagini all'oggetto con la classe carousel che scorre ogni 2 secondi in modo automatico.

### 3.3 Pagina di registrazione

Figura 6 Pagina di registrazione

Questo è il formulario che l'utente deve riempire per procedere nell'utilizzo dell'applicazione.

All'interno di questa pagina utilizzo principalmente due classi, Validator.js e Notify.js.

All'interno della classe Validator ho creato tutti i metodi utilizzati per la validazione di ogni singolo campo, questo è un esempio di un validatore:

```
basic(string) {
    if (string.trim() == "" || string.trim() != string) {
        return false;
    }
    if (this.containsChar(string, this.delimiter)) {
        return false
    } else {
        this.disabledCharsAtStart.forEach(function(char) {
            if (string.startsWith(char)) {
                return false;
            }
        });
    }
    return true;
}
```

Questo metodo permette di controllare che una stringa non sia nulla, che non contenga il delimitatore del file csv e che non contenga dei caratteri specifici come inizio di stringa ("@", "+", "=", "-") in modo da prevenire csvInjection.

Per eseguire il controllo una volta premuto il tasto "Avanti" viene richiamato il metodo submitCheck che esegue i vari metodi di validazione sui campi sugli input.

Esempio di validazione del campo nome:

```
var name = validator.general($('#name').val());
if (!name) {
    $('#name').css("border-color", "red");
    filled = false;
    $.notify("Inserisci un nome valido!", "error");
}
```

In caso il validator del nome ritorni falso il bordo del campo viene colorato di rosso e mostrata una notifica a schermo come si vede nella figura numero 6.

### 3.4 Pagina di controllo

Finke's Garage

Nome	Filippo
Cognome	Finke
Data di nascita	2001-12-22
Genere	M
Via	Monte Generoso
Numero civico	12C
NAP	6828
Città	Balerna
Telefono	410916788827
Email	filippo.finke@samtrevano.ch
Hobby	Videogames
Professione	Informatico

Filippo Finke I3AC - 2018

Figura 7 Pagina di controllo

Questa è la pagina di controllo, all'interno di questa pagina ritengo fondamentale il seguente blocco di codice:

```
<form id="hiddenForm" action="registrazione.php" method="post">
    <?php
    foreach ($_POST as $field => $value) {
        echo '<input type="hidden" name="' . htmlspecialchars($field) . '"'
value="'
        . htmlspecialchars($value) . '">';
    }
    ?>
</form>
```

Questo blocco di codice permette di generare un form invisibile, in modo dinamico, con i campi che sono stati passati dalla pagina di registrazione. In questo modo posso trasferire i dati dalla pagina di controllo a quella di registrazione in caso di correzione dei dati oppure passarli alla pagina finale nel quale eseguirne la scrittura.. Questa tecnica la utilizzo anche in altre pagine.

### 3.5 Pagina finale

Finke's Garage												
Data di registrazione	Nome	Cognome	Data di nascita	Genere	Via	Numero civico	NAP	Città	Telefono	Email	Hobby	Professione
2018-10-26 15:55:53	Filippo	Finke	2001-12-22	M	Testing	12A	6900	Lugano	410910000000	test@gmail.com		
<a href="#">TORNA ALLA PAGINA DI BENVENUTO</a>												

Filippo Finke i3AC - 2018

Figura 8 Pagina finale

Questa è la pagina finale nel quale vengono stampate tutte le registrazioni del giorno corrente. All'interno di questa classe utilizzo principalmente la classe CsvManager per gestire i file csv e Validator che è la traduzione della classe creata in JavaScript tradotta in PHP per la validazione dei dati lato server.

All'interno di questa pagina utilizzo questo codice per prevenire la scrittura in caso di ricaricamento della pagina e di redirect alla home del sito in caso di accesso senza una richiesta di tipo post.

```
$filled = false;
if (isset($_SESSION["filled"])) {
    session_destroy();
    $filled = true;
}
else if ($_SERVER["REQUEST_METHOD"] != "POST") {
    header("Location: index.html");
    exit();
}
```

Mentre con questo blocco di codice scrivo i dati, che ricavo grazie alla richiesta POST, all'interno del file giornaliero e file globale ed una volta scritti ricarico la pagina ed assegno un flag nella sessione dell'utente. Ricarico la pagina in modo da eliminare la richiesta POST salvata in automatico dal browser. In questo modo non ho problemi di doppia scrittura dei dati.

```
$globalCsv = new CsvManager(PATH."/".$globalFile, ";");
$dailyCsv = new CsvManager(PATH."/".$dailyFile, ";");
if ($globalCsv -> writeLine($dataArray) && $dailyCsv -> writeLine($dataArray))
{
    $_SESSION["filled"] = true;
    header("Location: resoconto.php");
    exit();
} else {
    $errorMessage = "C'è stato un errore nel salvataggio dei dati,  
riprova!<br>Redirezionamento in 3 secondi.";
}
```

Inoltre i dati vengono comunque validati lato server per prevenire modifiche lato client.

Per eseguire la validazione di un dato utilizzo la classe Validator.

Esempio di validazione del campo nome:

```

if (!$_v->general($name))
    $errors[] = array("name", "Inserisci un nome valido!");
    
```

In questo caso chiamo il metodo di validazione generale utilizzato per Nome, Cognome, Via e Città attraverso l'oggetto \$v ed in caso il risultato sia falso aggiungo un messaggio di errore da stampare all'utente.

Anche all'interno di questa pagina utilizzo la tecnica di creare un form invisibile per rimandare i dati dell'utente alla pagina di registrazione in caso di errore di validazione. Con il blocco seguente di codice controllo se sono stati generati degli errori e creo un form invisibile contenete le informazioni riguardanti tutti i campi inseriti dall'utente e aggiungo i messaggi di errore che verranno interpretati dalla pagina di registrazione sotto forma di notifica. Inoltre stampo un messaggio di errore nella pagina e attraverso JavaScript eseguo il submit del form dopo 3 secondi.

```

<?php
if(isset($errors) && count($errors) > 0 || !$filled) {
?>
<form id="hiddenForm" action="registrazione.php" method="post">
    <?php
    foreach ($_POST as $a => $b) {
        echo '<input type="hidden" name="' . htmlspecialchars($a) . '" value="' .
        htmlspecialchars($b) . '>';
    }
    foreach ($errors as $error) {
        echo '<input type="hidden" name="errors[]" value="' .
        htmlspecialchars($error[1]) . '>';
    }
?>
</form>
<script>
    $('#container').html("<h5 class='red-text'><?php echo $errorMessage;
?></h5>");
    setTimeout(function() {document.getElementById('hiddenForm').submit()}, 3000);
</script>
<?php
}
?>
    
```

## 4 Test

### 4.1 Protocollo di test

<b>Test Case:</b>	TC-001	<b>Nome:</b>	Test campi semplici
<b>Riferimento:</b>	REQ-02/03		
<b>Descrizione:</b>	Validazione dei campi semplici (Nome, Cognome, Via e Città) di tipo stringa. La stringa deve avere una lunghezza massima di cinquanta caratteri. Non può essere nulla e può contenere caratteri alfabetici, spazi, un . o un – seguiti e preceduto da dei caratteri.		
<b>Prerequisiti:</b>	La pagina registrazione.php deve essere attiva.		
<b>Procedura:</b>	<ol style="list-style-type: none"> <li>Inserire tutti i campi con almeno un carattere.</li> <li>All'interno del campo Nome, Cognome, Via e Città inserire dei dati non validi. Es: 1239123, Filippo1231, Filippo!@1</li> <li>Premere il tasto “Avanti”</li> </ol>		
<b>Risultati attesi:</b>	La pagina di registrazione deve ritornare un messaggio di errore che segnala i campi inseriti.		

<b>Test Case:</b>	TC-002	<b>Nome:</b>	Test campi di tipo text area
<b>Riferimento:</b>	REQ-02/03		
<b>Descrizione:</b>	Validazione dei campi di tipo text area (Hobby e Professione) di tipo stringa. La stringa deve avere una lunghezza massima di 500 caratteri. Può essere nulla e può contenere qualsiasi carattere al di fuori di testo tra i caratteri "<" e ">".		
<b>Prerequisiti:</b>	La pagina registrazione.php deve essere attiva.		
<b>Procedura:</b>	<ol style="list-style-type: none"> <li>1. Inserire tutti i campi con almeno un carattere</li> <li>2. All'interno del campo Hobby e Professione inserire dei dati non validi. Es: &lt;TestString&gt;, &lt;script src=&gt;, &lt;/777s&gt;</li> <li>3. Premere il tasto "Avanti"</li> </ol>		
<b>Risultati attesi:</b>	La pagina di registrazione deve ritornare un messaggio di errore che segnala i campi inseriti.		

<b>Test Case:</b>	TC-003	<b>Nome:</b>	Test campo email
<b>Riferimento:</b>	REQ-02/03		
<b>Descrizione:</b>	Validazione del campo email di tipo stringa. La stringa non può essere nulla, non deve contenere caratteri speciali e deve contenere almeno una @ e un .		
<b>Prerequisiti:</b>	La pagina registrazione.php deve essere attiva.		
<b>Procedura:</b>	<ol style="list-style-type: none"> <li>1. Inserire tutti i campi con almeno un carattere</li> <li>2. All'interno del campo Email inserire dei dati non validi. Es: 123123w@ @test.com, 123123, &lt;as@gmail.com</li> <li>3. Premere il tasto "Avanti"</li> </ol>		
<b>Risultati attesi:</b>	La pagina di registrazione deve ritornare un messaggio di errore che segnala il campo inserito.		

<b>Test Case:</b>	TC-004	<b>Nome:</b>	Test campo telefono
<b>Riferimento:</b>	REQ-02/03		
<b>Descrizione:</b>	Validazione del campo telefono di tipo stringa. La stringa non può essere nulla, non deve contenere caratteri speciali. Il formato del numero di telefono può essere con o senza prefisso e può contenere solamente numeri, "+" e spazi. Ci devono essere un minimo di 10 fino ad un massimo di 13 cifre.		
<b>Prerequisiti:</b>	La pagina registrazione.php deve essere attiva.		
<b>Procedura:</b>	<ol style="list-style-type: none"> <li>1. Inserire tutti i campi con almeno un carattere</li> <li>2. All'interno del campo telefono inserire dei dati non validi. Es: Numero123123, +411234567886868688, 123uijejwjdij923</li> <li>3. Premere il tasto "Avanti"</li> </ol>		
<b>Risultati attesi:</b>	La pagina di registrazione deve ritornare un messaggio di errore che segnala il campo inserito.		

<b>Test Case:</b>	TC-005	<b>Nome:</b>	Test campo NAP
<b>Riferimento:</b>	REQ-02/03		
<b>Descrizione:</b>	Validazione del campo NAP di tipo numerico. Ha un limite di 5 caratteri e può contenere solamente numeri. Minimo 4 caratteri.		
<b>Prerequisiti:</b>	La pagina registrazione.php deve essere attiva.		
<b>Procedura:</b>	1. Inserire tutti i campi con almeno un carattere 2. All'interno del campo NAP inserire dei dati non validi. Es: 123456, 1234, campo vuoto 3. Premere il tasto "Avanti"		
<b>Risultati attesi:</b>	La pagina di registrazione deve ritornare un messaggio di errore che segnala il campo inserito.		

<b>Test Case:</b>	TC-006	<b>Nome:</b>	Test campo numero civico
<b>Riferimento:</b>	REQ-02/03		
<b>Descrizione:</b>	Validazione del campo numero civico di tipo stringa. Non può essere nullo ha un massimo di 4 caratteri. Può contenere numeri ed un solo carattere dell'alfabeto. Il carattere alfabetico può essere scritto solamente dopo i caratteri numerici ed è opzionale.		
<b>Prerequisiti:</b>	La pagina registrazione.php deve essere attiva.		
<b>Procedura:</b>	1. Inserire tutti i campi con almeno un carattere 2. All'interno del campo Numero civico inserire dei dati non validi. Es: A123, 123AA, *123 3. Premere il tasto "Avanti"		
<b>Risultati attesi:</b>	La pagina di registrazione deve ritornare un messaggio di errore che segnala il campo inserito.		

<b>Test Case:</b>	TC-007	<b>Nome:</b>	Test campo data di nascita
<b>Riferimento:</b>	REQ-02/03		
<b>Descrizione:</b>	Validazione del campo data di nascita di tipo data. Non può essere nullo. La data ha il formato "aaaa-mm-gg". La data non può essere in futuro.		
<b>Prerequisiti:</b>	La pagina registrazione.php deve essere attiva.		
<b>Procedura:</b>	1. Inserire tutti i campi con almeno un carattere 2. All'interno del campo data premere con il mouse e selezionare una data nel passato. 3. Premere il tasto "Avanti"		
<b>Risultati attesi:</b>	La pagina di registrazione deve mostrare un popup dal quale è possibile selezionare una data e sottolineare il campo di verde.		

<b>Test Case:</b>	TC-008	<b>Nome:</b>	Test campo genere
<b>Riferimento:</b>	REQ-02/03		
<b>Descrizione:</b>	Validazione del campo genere di tipo stringa. Non può essere nullo e può essere solamente "F" oppure "M"		
<b>Prerequisiti:</b>	La pagina registrazione.php deve essere attiva.		
<b>Procedura:</b>	1. Inserire tutti i campi con almeno un carattere 2. Lasciare vuoto il campo che richiede il genere. 3. Premere il tasto "Avanti"		
<b>Risultati attesi:</b>	La pagina di registrazione deve mostrare un errore che segnala il campo da completare.		

<b>Test Case:</b>	TC-009	<b>Nome:</b>	Test classe di gestione file csv
<b>Riferimento:</b>	REQ-01		
<b>Descrizione:</b>	Controllare che la classe di gestione del file csv crei i file se essi sono inesistenti.		
<b>Prerequisiti:</b>	La pagina registrazione.php deve essere attiva.		
<b>Procedura:</b>	1. Eliminare la cartella Registrazioni 2. Recarsi nella pagina di registrazione e compilare il formulario. 3. Premere il tasto "Avanti", premere il tasto "Registra"		
<b>Risultati attesi:</b>	I file csv mancanti devono essere stati creati in modo automatico.		

<b>Test Case:</b>	TC-010	<b>Nome:</b>	Test pagina di registrazione
<b>Riferimento:</b>	REQ-05		
<b>Descrizione:</b>	Controllare che il comando di eseguire un reset dei dati del form sia correttamente funzionante. Il reset deve cancellare tutti i dati negli input, i contatori ed eventuali bordi colorati.		
<b>Prerequisiti:</b>	La pagina registrazione.php deve essere attiva.		
<b>Procedura:</b>	1. Recarsi nella pagina di registrazione e compilare il formulario. 2. Premere il tasto "Cancella", premere il tasto "Ok" nel popup.		
<b>Risultati attesi:</b>	Il formulario deve venire resettato completamente. I valori devono essere vuoti, i contatori dei caratteri impostati a zero e il colore dei bordi a grigio.		

<b>Test Case:</b>	TC-011	<b>Nome:</b>	Test pagina di controllo
<b>Riferimento:</b>	REQ-06		
<b>Descrizione:</b>	Controllare che il comando di correzione riporti alla pagina di registrazione con i campi già compilati in base al formulario precedenze.		
<b>Prerequisiti:</b>	La pagina registrazione.php e controllo.php deve essere attiva.		
<b>Procedura:</b>	1. Recarsi nella pagina di registrazione e compilare il formulario. 2. Premere il tasto "Avanti" 3. Premere il tasto "Correggi"		
<b>Risultati attesi:</b>	La pagina deve redirizionare l'utente alla pagina di registrazione.php con i campi già compilati in base al formulario precedente.		

#### 4.2 Risultati test

Codice test	Risultato	Note
TC-001	Passato	Nessuna
TC-002	Passato	Nessuna
TC-003	Passato	Nessuna
TC-004	Passato	Nessuna
TC-005	Passato	Nessuna
TC-006	Passato	Nessuna
TC-007	Passato	Nessuna
TC-008	Passato	Nessuna
TC-009	Passato	Nessuna
TC-010	Passato	Nessuna
TC-011	Passato	Nessuna

#### 4.3 Mancanze/limitazioni conosciute

L'applicativo non può essere visualizzato in modo corretto dal browser "Internet Explorer". "Internet Explorer" non supporta l'utilizzo di classi da parte di JavaScript quindi l'applicativo non può essere caricato.

**HTML1300:** È stata eseguita la navigazione.

`registrazione.php`

**SCRIPT1002:** Errore di sintassi

`validator.js (6,17)`

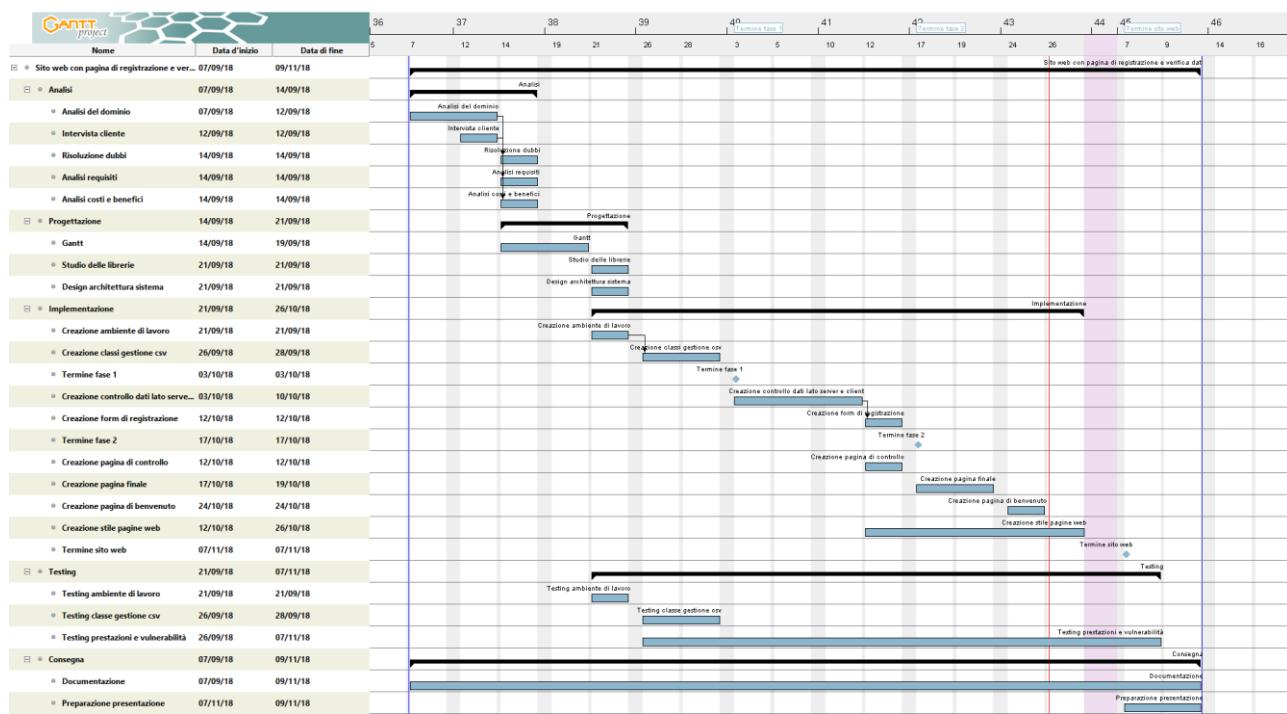
**SCRIPT5009:** 'Validator' non è definito

`registrazione.js (1,1)`

Inoltre non ritengo essenziale lo sviluppo di una versione personalizzata per questo browser poiché vecchio ed obsoleto.

**Sito web con pagina di registrazione e verifica dati**

## 5 Consuntivo



Nel gantt consuntivo si possono notare diversi cambiamenti:

- L'attività di sviluppo del gantt si è accorciata di una settimana.
- Lo sviluppo della pagina di controllo e di registrazione sono state unite nello stesso giorno.
- Lo sviluppo della pagina finale è stato prolungato di una settimana.
- Lo sviluppo della pagina di benvenuto è stato anticipato di una settimana.

## 6 Conclusioni

L'applicativo è stato sviluppato con lo scopo di apprendere la gestione di un progetto. Esistono molti tipi di applicativi che permettono personalizzazioni, modifiche e molto altro rispetto a ciò che ho sviluppato. Quindi ritengo che il mio progetto non sia stato sviluppato con lo scopo di essere utilizzato al di fuori del campo scolastico. L'applicativo è stato sviluppato interamente come richiesto dalla specifica.

### 6.1 Sviluppi futuri

Su questo prodotto possono essere sviluppati molti componenti aggiuntivi, come per esempio:

- Conferma d'iscrizione al modulo tramite posta elettronica.
- Termini e Condizioni
- Verifica telefonica
- Verifica del NAP tramite il campo Città.
- Maggiore sicurezza nel salvataggio dei dati.

...  
Alcune migliorie si potrebbero fare all'interno della gestione delle pagine web, in altre parole utilizzare delle sessioni. Gestire in maniera migliore i file csv in modo da supportare dati multi linea, ...

### 6.2 Considerazioni personali

Grazie a questo progetto ho imparato a sviluppare del codice orientato agli oggetti utilizzando il linguaggio JavaScript.

## 7 Bibliografia

---

### 7.1 Sitografia

- <https://www.ganttproject.biz/>, GanttProject: free desktop project management app, 14-09-2018.
- <https://www.draw.io/>, Flowchart Maker Online Diagram Software, 21-09-2018
- <https://www.ubuntu.com/download/server>, Download Ubuntu Server | Download | Ubuntu, 21-09-2018
- <http://php.net/manual/en/language.oop5.php>, PHP: Classes and Objects - Manual, 26-09-2018
- <https://www.phpliveregex.com/>, PHP Live Regex, 03-10-2018
- <https://payatu.com/csv-injection-basic-to-exploit/>, CSV injection: Basic to Exploit!!!! - payatu, 03-10-2018
- <https://materializecss.com/>, Documentation - Materialize, 05-10-2018
- <https://developer.mozilla.org/it/docs/Web/JavaScript/Reference/Classes>, Classes - JavaScript | MDN, 05-10-2018
- <https://materializecss.com/pickers.html>, Pickers - Materialize, 10-10-2018
- <https://stackoverflow.com/>, Stack Overflow, 10-10-2018
- <https://notifyjs.jpillora.com/>, Notify.js, 12-10-2018
- [https://rextester.com/l/php\\_online\\_compiler](https://rextester.com/l/php_online_compiler), compile php online, 17-10-2018
- [https://it.wikipedia.org/wiki/Cross-site\\_scripting](https://it.wikipedia.org/wiki/Cross-site_scripting), Cross-site scripting - Wikipedia, 19-10-2018

## 8 Allegati

---

- Diari di lavoro
- Quaderno dei compiti
- Prodotto
- Codice sorgente su GitHub <https://github.com/filippofinke/Progetto-1>

---

## 1 INFORMAZIONI GENERALI

Allievo/i	Nome:	Cognome:
		
<b>Luogo di lavoro</b>	Aula 417 (ex A-413) – Scuola d'Arti e Mestieri Trevano	
<b>Orientamento</b>	<input checked="" type="checkbox"/> 88602 Informatica aziendale	
<b>Docente responsabile</b>	Nome:	Cognome:
		
<b>Periodo</b>		
<b>Orario di lavoro</b>	Secondo orario scolastico 1° semestre	
<b>Numero di ore lezione</b>		
<b>Pianificazione (in H o %)</b>	Analisi: 10%	
	Implementazione: 50%	
	Test: 10%	
	Documentazione: 30%	

---

## 2 PROCEDURA

- L'allievo realizza il lavoro autonomamente o con il gruppo al quale è assegnato, sulla base del quaderno dei compiti ricevuto il 1 ° giorno.
- Il quaderno dei compiti è presentato, commentato e discusso con l'allievo. Con la sua firma, l'allievo accetta il lavoro proposto.
- L'allievo ha conoscenza della scheda di valutazione all'inizio del lavoro.
- L'allievo è responsabile dei suoi dati.
- In caso di problemi gravi, l'allievo avverte immediatamente il docente responsabile.
- L'allievo ha la possibilità di chiedere aiuto, ma deve menzionarlo nella documentazione.
- Alla fine del tempo a disposizione per la realizzazione del progetto, l'allievo deve inviare via email il progetto al docente responsabile. In parallelo, una copia cartacea della documentazione dovrà essere fornita sempre al docente responsabile. Quest'ultima deve essere in tutto identica alla versione elettronica.

---

## 3 TITOLO

Sito web con pagina di registrazione e verifica dati

---

## 4 HARDWARE E SOFTWARE DISPONIBILE

1 PC della scuola + programmi concordati con i formatori

---

## 5 PREREQUISITI

---

## 6 DESCRIZIONE DEL PROGETTO

Si tratta di sviluppare un sito web che permetta la registrazione dei dati di un utente, la registrazione deve avvenire dopo la lettura di una pagina riassuntiva dei dati.

I dati devono essere tutti convalidati.

I dati saranno memorizzati in due file di testo in formato CSV

L'aspetto grafico del sito e la disposizione dei campi nel form, nella pagina riassuntiva e nella pagina di rilettura è a discrezione degli sviluppatori.

In particolare:

- All'entrata del sito ci dovrà essere una pagina con un benvenuto ed un "tasto" "Registrati" che porta al form di registrazione.
- Nel form di registrazione ci saranno dei campi obbligatori contrassegnati con un asterisco e dei campi opzionali
- Alla fine del form ci saranno due "Tasti", "Cancella" che azzererà tutti i campi e "Avanti" che porterà ad una pagina con una tabella riassuntiva per controllo dei dati.
- Alla fine della pagina di controllo ci saranno due tasti, "Correggi" che riporterà alla pagina di registrazione per permettere di modificare i dati e "Registra" che andrà a scrivere i nuovi dati nei files di memorizzazione
- Al momento della registrazione, ai dati verrà aggiunto un campo con la data e l'ora corrente della registrazione. (Data e ora lato server)
- Ci saranno due files, uno con tutte le registrazioni che si chiamerà "Registrazioni\_tutte.csv" e l'altro con le registrazioni del giorno corrente "Registrazione\_aaaa-mm-gg.csv"
- I due files CSV saranno scritti nella cartella "Registrazioni" sotto la cartella principale del sito.
- Ogni campo, dovrà essere validato prima della registrazione (a scelta, lato client o lato server).  
La validazione dei campi non può essere lasciata ai meccanismi del browser ma dovrà essere implementata.
- Alla fine della registrazione dovrà essere presentata una pagina con la medesima disposizione dei campi del formulario di immissione con i dati letti dal file "Registrazione\_aaaa-mm-gg.csv".
- Durante la compilazione del form, ci deve essere un meccanismo che non permetta di andare avanti se tutti i campi obbligatori non sono stati compilati o se una o più validazioni non sono andate a buon fine.

Campi necessari:

• Nome*	testo
• Cognome*	testo
• Data di nascita *	data
• No. Civico*	numerico, 3 cifre
• Città*	testo
• Nap*	numerico, 5 cifre
• No. Telefono*	testo (solo cifre, spazi e/o trattini)
• E-mail*	testo , (controllo formato e-mail <a href="mailto:testo@testo.testo">testo@testo.testo</a> )
• Genere (sesso)*	testo (F o M)
• Hobby	testo
• Professione	testo

I campi con l'asterisco sono obbligatori

---

## 7 RISULTATI FINALI

L'allievo è responsabile della consegna al docente responsabile di:

- Una pianificazione iniziale (entro la prima settimana)
- Una documentazione della pianificazione e progettazione
- Una documentazione dell'implementazione e test
- Un diario di lavoro giornaliero entro le 18:00
- *Alla fine, il sito funzionante con i files CSV contenenti almeno 20 registrazioni avvenute su almeno in tre giorni diversi.*

---

## 8 PUNTI TECNICI SPECIFICI VALUTATI

La griglia di valutazione definisce i criteri generali secondo cui il lavoro dell'allievo sarà valutato (documentazione, diario, rispetto dei standard, qualità, ...).

Inoltre, il lavoro sarà valutato sui seguenti 7 punti specifici (punti da A14 a A20):

1. 159, *Analisi del problema (programmazione)*
2. 124, *Ipotesi di test, casi di test*
3. 225, *Gestione delle versioni con un programma d'amministrazione*
4. 232, *Programmazione Web professionale*
5. 193, *Design del GUI*
6. 164, *Codifica: Gestione degli errori*
7. 123, Commenti del codice sorgente

---

## 9 FIRMA

**Allievo**

(luogo e data)

**Docente responsabile**

(luogo e data)

# Diario di lavoro

---

Luogo	SAM Trevano
Data	2018-11-09

## Lavori svolti

Ho terminato la presentazione e terminato la documentazione. Inoltre ho creato il file contenente tutti i vari pdf in un file unico. Alla fine della lezione ho consegnato il tutto.

## Problemi riscontrati e soluzioni adottate

---

---

## Punto della situazione rispetto alla pianificazione

---

---

---

## Programma di massima per la prossima giornata di lavoro

---

---

---

# Diario di lavoro

---

Luogo	SAM Trevano
Data	2018-11-07

**Lavori svolti**

Ho iniziato la stesura della presentazione e ho finito la documentazione.

**Problemi riscontrati e soluzioni adottate****Punto della situazione rispetto alla pianificazione****Programma di massima per la prossima giornata di lavoro**

# Diario di lavoro

Luogo	SAM Trevano
Data	2018-10-24

## Lavori svolti

Nella lezione di oggi ho finito di fare alcuni ritocchi al codice soprattutto nello stile.

Inoltre ho completato altri capitoli all'interno nella documentazione.

Sviluppo 2 ore

Documentazione 2 ore

## Problemi riscontrati e soluzioni adottate

(Empty form)

## Punto della situazione rispetto alla pianificazione

(Empty form)

## Programma di massima per la prossima giornata di lavoro

(Empty form)

# Diario di lavoro

Luogo	SAM Trevano
Data	2018-10-26

## Lavori svolti

Ho eseguito dei ritocchi alla documentazione nei primi 45 minuti.

Per i 30 minuti successivi ho implementato un metodo per prevenire automatismi del browser.

Durante il resto della lezione ho continuato la documentazione.

## Problemi riscontrati e soluzioni adottate

Ho riscontrato un problema con il quale permette all'utente di rieseguire delle richieste post salvate dal browser. Per ovviare a questo problema ho disabilitato la possibilità di tornare indietro di pagina.

Ho risolto cancellando la cronologia precedente della pagina con questo codice:

```
history.pushState(null, null, location.href);
window.onpopstate = function () {
    history.go(1);
};
```

Ho creato un file javascript da includere in tutte le pagine.

Il file è chiamato preventBackHistory.js

## Punto della situazione rispetto alla pianificazione


## Programma di massima per la prossima giornata di lavoro


# Diario di lavoro

Luogo	SAM Trevano
Data	2018-10-10

## Lavori svolti

Durante i primi 30 minuti di lezione ho creato la classe di validazione dei dati lato client chiamata Validator.js.

Nel resto delle ore ho implementato la classe da me creata nella pagina di registrazione.

## Problemi riscontrati e soluzioni adottate

Ho riscontrato dei problemi nella validazione delle email lato server, che ho risolto grazie a questo utente di stackoverflow.

<https://stackoverflow.com/questions/46155/how-to-validate-an-email-address-in-javascript>

```
var re = /^[^<>()[].,;:\s@"]+(\.[^<>()[].,;:\s@"]+)*|(\".+\")@\(([0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.\[0-9]{1,3}\]\)|(([a-zA-Z]-0-9)+\.)+[a-zA-Z]{2,})$/;
```

Ho utilizzato il suo regex per eseguire la verifica delle email valide.

Ho modificato il datepicker grazie alla documentazione di materialize:

<https://materializecss.com/pickers.html>

```
$('#birthdate').datepicker({  
    yearRange: 30  
});
```

Nel modo di avere un range degli anni maggiore.

## Punto della situazione rispetto alla pianificazione


## Programma di massima per la prossima giornata di lavoro

Pensare a dei testacase per le varie classi.

Filippo Finke

306

Progetto 1


# Diario di lavoro

Luogo	SAM Trevano
Data	2018-10-12

## Lavori svolti

Implementato le pagine registrazione.php e controllo.php in tutte le 4 ore

## Problemi riscontrati e soluzioni adottate

Ho riscontrato un problema sul fatto di eseguire un reset dei campi all'interno di un form con dei valori predefiniti presi dall'array globale POST.

Ho risolto eseguendo un for tra tutti gli input del form e resettandoli.

```
function resetForm(event)
{
    event.preventDefault();
    var formelements = document.getElementById("reg-form").elements;
    for (i = 0; i < formelements.length; i++)
    {
        type = formelements[i].type.toLowerCase();
        switch (type)
        {
            case "text":
            case "textarea":
            case "tel":
            case "email":
            case "number":
                formelements[i].value = "";
                break;
            case "radio":
            case "checkbox":
                if (formelements[i].checked)
                {
                    formelements[i].checked = false;
                }
                break;
            default:
                break;
        }
    }
}
```

Ho deciso di adottare l'utilizzo di una libreria esterna per mostrare delle notifiche a schermo.

<https://notifyjs.jpillora.com/>

Punto della situazione rispetto alla pianificazione

Mi ritrovo avanti rispetto alla situazione da me pianificata.

Programma di massima per la prossima giornata di lavoro

Pensare a come sviluppare la pagina di riassunto finale.

# Diario di lavoro

Luogo	SAM Trevano
Data	2018-10-17

## Lavori svolti

Iniziata implementazione della pagina “resoconto.php” nel quale avviene la scrittura e la lettura dei dati dai file csv. In questa lezione ho impiegato 2 ore per creare questa pagina e mettere assieme le altre classi da me sviluppate.

## Problemi riscontrati e soluzioni adottate

Ho riscontrato dei problemi con la funzione fputcsv di php, ho risolto creando del codice che la rimpiazzasse

```
$csvline = "";
$x = 0;
foreach($line as $data)
{
    if($x != 0)
        $csvline .= $this->delimiter;
    $csvline .= $data;
    $x++;
}
```

## Punto della situazione rispetto alla pianificazione

Mi ritrovo in avanti di una azione all'interno del gantt.

## Programma di massima per la prossima giornata di lavoro

# Diario di lavoro

Luogo	SAM Trevano
Data	2018-10-19

## Lavori svolti

Nelle prime due ore ho terminato lo sviluppo della pagina di resoconto e risolto alcuni bug. Nel resto delle 2 ore ho corretto alcune cose all'interno della documentazione.

Ho risolto un problema nel diagramma UML delle classi.

## Problemi riscontrati e soluzioni adottate

Ho riscontrato un problema in un metodo da me creato nella classe Validator. Il metodo non doveva accettare stringhe con <> ma accettava comunque.

Ho risolto così:

```
if (string.length > 0 && string.length <= 500 && !/<(.*)>/ .test(string))
```

## Punto della situazione rispetto alla pianificazione


## Programma di massima per la prossima giornata di lavoro


# Diario di lavoro

Luogo	SAM Trevano
Data	2018-09-21

## Lavori svolti

Ore di teoria: 1.5 ore ca

Durante la teoria abbiamo discusso su come eseguire la progettazione partendo dai requisiti e dal mandato.

Nella parte pratica ho creato i diagrammi di flusso e ho completato il design di architettura. Ho completato i capitoli 2.1, 2.2 e 2.4 della documentazione.

Ho anche installato la macchina virtuale con ubuntu server, php e apache che utilizzerò per lo sviluppo.

## Problemi riscontrati e soluzioni adottate

Programma online per lo sviluppo di diagrammi.

Usato <http://www.draw.io>

## Punto della situazione rispetto alla pianificazione

Mi trovo giusto con la pianificazione. Installazione e test Virtual Machine fatti

## Programma di massima per la prossima giornata di lavoro

# Diario di lavoro

Luogo	SAM Trevano
Data	2018-09-26

## Lavori svolti

Dalle ore 13:15 alle ore 14:18 ho fatto il test sulla teoria di modulo 306.

Durante il tempo restante ho iniziato a sviluppare la classe per la gestione dei file csv come da gantt.

## Problemi riscontrati e soluzioni adottate

## Punto della situazione rispetto alla pianificazione

Virtual Machine e test effettuati.

Iniziato lo sviluppo della classe di gestione per file csv.

Mi trovo in linea con il gantt preventivo.

## Programma di massima per la prossima giornata di lavoro

Continuare lo sviluppo della classe per la gestione di file csv.

# Diario di lavoro

Luogo	SAM Trevano
Data	2018-09-2018

## Lavori svolti

Ho terminato i mockup e il diagramma di flusso utilizzando il sito <https://draw.io> e paint. Per creare il diagramma di flusso in modo corretto ho impiegato 3 ore di lezione. Inoltre ho riguardato il capitolo 2 della documentazione. Da ora posso iniziare l'implementazione.

## Problemi riscontrati e soluzioni adottate

Ho avuto problemi nella modifica di alcune immagini, ho risolto utilizzando paint.

## Punto della situazione rispetto alla pianificazione


## Programma di massima per la prossima giornata di lavoro


# Diario di lavoro

Luogo	SAM Trevano
Data	2018-10-03

## Lavori svolti

La prima ora della lezione ci è stata proposta una presentazione sulla fase di implementazione e test di un progetto. Nella seconda parte della documentazione abbiamo continuato con la documentazione e l'implementazione.

Creata classe CsvManager e Validator.

CsvManager permette la lettura e scrittura di dati su un file csv.

Validator permette di eseguire la validazione sui campi.

Modificato il diagramma di flusso e aggiunti i diagrammi UML delle due classi.

## Problemi riscontrati e soluzioni adottate


## Punto della situazione rispetto alla pianificazione


## Programma di massima per la prossima giornata di lavoro


# Diario di lavoro

Luogo	SAM Trevano
Data	2018-10-05

## Lavori svolti

Durante la lezione ho creato un test case poi confermato dal docente. Inoltre ho finito lo sviluppo della classe di validazione per i dati lato server ed iniziato la classe di validazione dati lato client. Per validazione lato server ho speso 1 ora mentre per il resto le rimanenti 3 ore nelle quali ho iniziato a sviluppare anche alcune interfacce grafiche usando <https://materializecss.com/>.

## Problemi riscontrati e soluzioni adottate

## Punto della situazione rispetto alla pianificazione

## Programma di massima per la prossima giornata di lavoro

Pensare alla struttura delle cartelle

# Diario di lavoro

Luogo	SAM Trevano
Data	2018-09-07

## Lavori svolti

Durante la lezione abbiamo analizzato il template della documentazione che verrà utilizzato per i progetti punto per punto con delle spiegazioni ed esempi. Inoltre abbiamo parlato di vari modelli di progetto attraverso una presentazione chiamata “Il ciclo di vita di un progetto”. Alla fine della lezione abbiamo parlato di come analizzare un mandato nei dettagli attraverso un'altra presentazione chiamata “L’analisi” e ci è stato dato un compito per la lezione successiva, ovvero di analizzare il progetto che ci è stato assegnato scrivendo delle domande da porre.

## Problemi riscontrati e soluzioni adottate

## Punto della situazione rispetto alla pianificazione

## Programma di massima per la prossima giornata di lavoro

Per la prossima giornata di lavoro dovrò analizzare il mandato in modo da capire totalmente la specifica.

# Diario di lavoro

Luogo	SAM Trevano
Data	2018-09-12

## Lavori svolti

Durante la lezione nella parte iniziale abbiamo posto le nostre domande riguardanti il progetto e i sori ne hanno preso nota in modo da risponderci entro la lezione stessa. Ci è stato dato del tempo per eseguire dell'analisi sul progetto. Alla fine della lezione ci sono state date le risposte alle varie domande.

## Problemi riscontrati e soluzioni adottate

## Punto della situazione rispetto alla pianificazione

## Programma di massima per la prossima giornata di lavoro

### Rivedere il quaderno del giorno

# Diario di lavoro

Luogo	SAM Trevano
Data	2018-09-14

## Lavori svolti

Durante la lezione ho completato l'analisi dei requisiti e controllata assieme al docente. Inoltre ho aggiunto delle informazioni nei capitoli 1.1 e 1.6 della documentazione. Ho anche installato un software per la gestione di progetti, chiamato GanttProject. Ho iniziato a creare uno schizzo del diagramma Gantt.

## Problemi riscontrati e soluzioni adottate

## Punto della situazione rispetto alla pianificazione

## Programma di massima per la prossima giornata di lavoro

Scegliere, installare e provare un software per la gestione di progetti (GANTT)

# Diario di lavoro

Luogo	SAM Trevano
Data	2018.09.19

## Lavori svolti

Durante la lezione abbiamo ricontrollato i requisiti scritti nella lezione precedente all'interno della documentazione. Ho inoltre continuato e terminato il progetto del gantt utilizzando il software “Gantt Project”.

Ho riempito il capitolo 1.7 della documentazione sul progetto.

## Problemi riscontrati e soluzioni adottate

## Punto della situazione rispetto alla pianificazione


## Programma di massima per la prossima giornata di lavoro
