

Diario di lavoro

Luogo	Canobbio
Data	02.03.2020

Lavori svolti
08h20 – 08h30 Ho aggiunto alle dipendenze la libreria FPDF la quale verrà utilizzata per la generazione dei PDF. Utilizzando composer ho dunque aggiunto la libreria:
<pre>"require": { "filippofinke/php-rest": "dev-master", "fpdf/fpdf": "dev-master" },</pre>
Successivamente ho testato il funzionamento della libreria provando a generare un semplice file pdf come descritto nella guida: http://www.fpdf.org/
08h30 – 09h50 Ho aggiornato nuovamente il calendario dell'applicativo aggiungendo le date delle lezioni serali le quali erano mancanti, essendo la libreria dinamica mi è bastato aggiungerlo all'array di date:
<pre>var hours = [{start:"08:20", end:"09:05", allow:true}, {start:"09:05", end:"09:50", allow:true}, {start:"10:05", end:"10:50", allow:true}, {start:"10:50", end:"11:35", allow:true}, {start:"11:35", end:"12:20", allow:true}, {start:"13:15", end:"14:00", allow:true}, {start:"14:00", end:"14:45", allow:true}, {start:"15:00", end:"15:45", allow:true}, {start:"15:45", end:"16:30", allow:true}, {start:"16:30", end:"17:15", allow:true}, {start:"16:00", end:"16:45", allow:true}, {start:"16:45", end:"17:30", allow:true}, {start:"17:30", end:"18:15", allow:true}, {start:"18:30", end:"19:15", allow:true}, {start:"19:15", end:"20:00", allow:true}, {start:"20:15", end:"21:00", allow:true}, {start:"21:00", end:"21:45", allow:true},];</pre>

Le date sono poi state mostrate all'interno del calendario. Il quale però non è responsive per dispositivi con piccoli schermi, quindi ho reso il tutto responsive attraverso l'utilizzo di alcune regole CSS. Per iniziare ho sovrascritto la grandezza delle colonne con una larghezza fissa:

```
.calendar .col-1 {
  display: block;
  width: 100px;
  flex: none;
  max-width: none;
  -webkit-box-flex: none;
}

.calendar .col-2 {
  display: block;
  width: 200px;
  flex: none;
  max-width: none;
  -webkit-box-flex: none;
}
```

Successivamente ho aggiunto un contenitore esterno al calendario il quale permette di fare scorrere orizzontalmente il contenuto del calendario:

```
.calendar-container {
  overflow: auto;
  padding-bottom: 20px;
}
```

In questo modo il risultato del calendario è stato il seguente:

<div>X</div> <div>Settimana</div>		08:20 09:05	09:05 09:50	10:05 10:50	10:50 11:35	11:35 12:20	13:15 14:00	14:00 14:45	15:00 15:45
gg/mm/aaaa	LUNEDÌ								
gg/mm/aaaa	MARTEDÌ								
gg/mm/aaaa	MERCOLEDÌ								
gg/mm/aaaa	GIOVEDÌ								
gg/mm/aaaa	VENERDÌ								
gg/mm/aaaa	SABATO								

Figura 1 Calendario su laptop.

<input type="button" value="X"/>	Settimana <input type="button" value="⬇"/>	08:20 09:05	09:05 09:50	10:05 10:50	10:50 11:35
<input type="text" value="gg/mm/aaaa"/>	LUNEDÌ				
<input type="text" value="gg/mm/aaaa"/>	MARTEDÌ				
<input type="text" value="gg/mm/aaaa"/>	MERCOLEDÌ				
<input type="text" value="gg/mm/aaaa"/>	GIOVEDÌ				
<input type="text" value="gg/mm/aaaa"/>	VENERDÌ				
<input type="text" value="gg/mm/aaaa"/>	SABATO				

Figura 2 Calendario su ipad.

<input type="button" value="X"/>	Settimana <input type="button" value="⬇"/>
<input type="text" value="gg/mm/aaaa"/>	LUNEDÌ
<input type="text" value="gg/mm/aaaa"/>	MARTEDÌ
<input type="text" value="gg/mm/aaaa"/>	MERCOLEDÌ
<input type="text" value="gg/mm/aaaa"/>	GIOVEDÌ
<input type="text" value="gg/mm/aaaa"/>	VENERDÌ
<input type="text" value="gg/mm/aaaa"/>	SABATO

Figura 3 Calendario su iPhone 6.

In tutte e tre le versioni è possibile scorrere il calendario orizzontalmente in modo che sia accessibile su qualsiasi dispositivo come per esempio telefoni. Successivamente sempre riguardante il calendario ho aggiunto dei controlli sulla selezione delle date, ora la selezione della data del giorno è limitata solamente al giorno corrente per ogni settimana. Questo grazie all'utilizzo di HTML5:

```

var date = document.createElement("input");
date.type = "date";
date.step = "7";
var min = new Date(monday.getTime() + 86400000 * i);
date.min = min.getFullYear() + "-" + zeroPad((min.getMonth() + 1), 2) + "-" +
zeroPad(min.getDate(), 2);
date.classList = "form-control col-7 float-left ml-3";
date.setAttribute("data-index", i);
date.addEventListener("change", (event) => {
    this.dates[event.target.dataset.index] = event.target.value;
});

```

Nella creazione dell'input data attraverso JavaScript ho aggiunto i controlli HTML5: step e min. Questi due controlli permettono di impostare di quanto andare avanti nel calendario e il valore minimo. In questo caso è permesso andare avanti di settimana in settimana con il minimo il lunedì corrente.



Figura 4 Esempio selezione mercoledì.

10h05 – 10h10

Ora viene mostrata la modalità di visualizzazione dei congedi all'interno della pagina di modifica. Questo perché se un utente fa parte della Direzione, anche accedendo dal contenitore della segreteria, potrà accettare i congedi direttamente. La modalità di visualizzazione dipende dal permesso attraverso la sessione:

```
<?php echo ($editing)?"Modifica come ".$_SESSION["permission"]:"Home"; ?>
```

E viene mostrato nel seguente modo:

Gestione congedi

[Dashboard](#) > Modifica come Direzione

Figura 5 Modalità di visualizzazione.

10h10 – 11h35

Ho trasferito la gestione delle ore e dei giorni del calendario nel file di configurazione dell'applicativo in modo che tutto sia dinamico. I dati delle ore e dei giorni vengono presi da PHP e stampati in modo che JavaScript li possa leggere, questo avviene nel seguente modo:

```
var labels = [
    <?php foreach(CALENDAR_LABELS as $label): ?>
        "<?php echo $label; ?>",
    <?php endforeach; ?>
];
var hours = [
    <?php foreach (CALENDAR_HOURS as $hour): ?>
        {start:"<?php echo $hour["start"]; ?>", end:"<?php echo $hour["end"];
?>", allow: <?php echo $hour["allow"]? 'true' : 'false'; ?>},
    <?php endforeach; ?>
];
```

13h15 – 14h45

Ho iniziato a creare una copia fedele del file PDF dei congedi, utilizzando la libreria FPDF importata nella mattinata. Il codice per la creazione del PDF è molto lungo, quindi non l'ho inserito nel diario. Il risultato del PDF è il seguente:

Visualizza PDF

Stampa

Chiudi

Figura 6 Parte del calendario.

Visualizza PDF

Stampa

Chiudi

Figura 7 Pagina con motivazioni.

Ho inoltre implementato il tasto stampa che permette di aprire la finestra di stampa del PDF in automatico. La funzione che richiama la finestra di stampa è la seguente:

```
function printPdf() {
    $('#iframe').get(0).contentWindow.print();
}
```

La funzione in questione si occupa di aprire la finestra di stampa predefinita del browser corrente. Esempio su chrome:

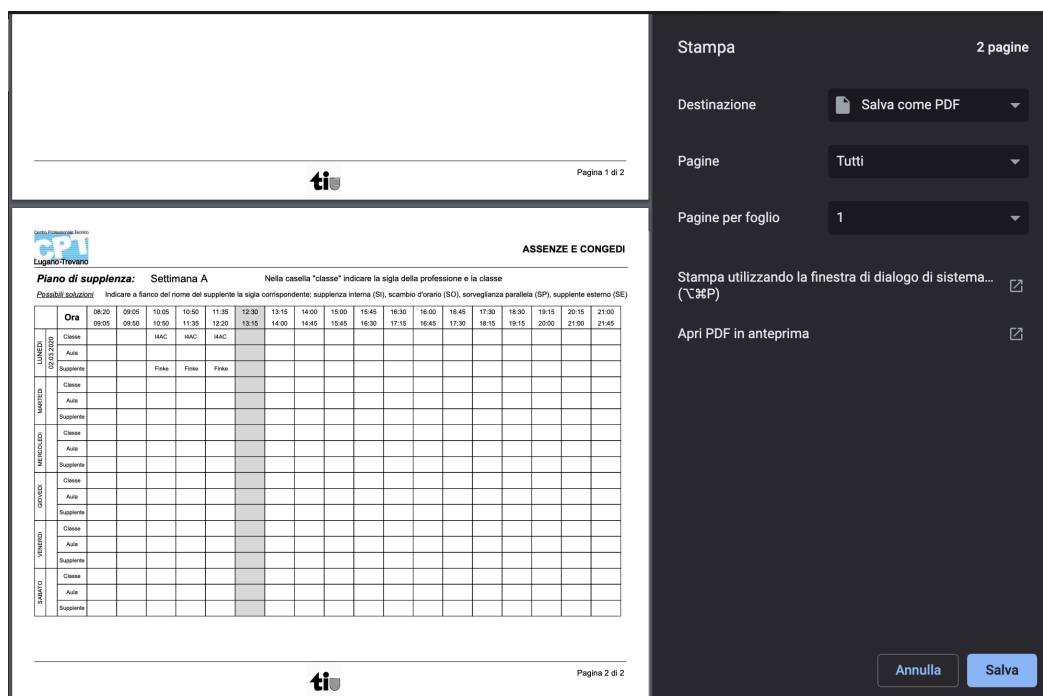


Figura 8 Finestra di stampa, chrome.

Problemi riscontrati e soluzioni adottate

Nessun problema riscontrato.

Punto della situazione rispetto alla pianificazione

Ho un leggero vantaggio rispetto alla pianifica, molto tempo è stato utilizzato per l'implementazione del PDF anche se non ancora finito.

Programma di massima per la prossima giornata di lavoro

Continuare la generazione di PDF.