Diario di lavoro

|  |  |
| --- | --- |
| Luogo | Canobbio |
| Data | 09.03.2020 |

|  |
| --- |
| Lavori svolti |
| 08h20 – 08h50  Mi sono occupato di risolvere un warning di PHP il quale segnalava che alcuni headers venivano modificati dopo che il contenuto della pagina era già stato inviato all’utente. Il problema affliggeva il percorso riguardante la creazione dei PDF. La libreria utilizzata per creazione dei PDF si occupava già di mandare gli header adeguati, che però venivano rimandati anche dalla libreria php-rest la quale si occupa di gestire le risposte. Vedi sezione “Problemi riscontrati e soluzioni adottate” per la soluzione del problema.  08h50 – 09h10  Ho aggiunto il nome del file dei PDF il quale viene generato automaticamente con l’ultima data di aggiornamento del congedo. Inoltre ho aggiunto anche un bottone il quale permette di scaricare il PDF dei congedi direttamente dal browser.    Figura Schermata visualizzazione PDF.  09h10 – 09h40  Ho risolto un problema nell’aggiornamento dei congedi il quale non permetteva il salvataggio delle modifiche della tipologia di supplenza, il problema era dovuto alla riassegnazione di una variabile. Vedi sezione “Problemi riscontrati e soluzioni adottate” per la soluzione del problema.  09h40 – 09h50  Ho reso dinamica la seleziona del font nella stampa del nome e cognome dei supplenti all’interno della generazione del PDF, questo in modo che il testo sia sempre all’interno delle celle. Per fare questo ho utilizzato un ciclo il quale controlla la lunghezza della stringa da stampare decrementando la grandezza del font fino a quando resta all’interno della cella:  $fontSize = 7;  while ($this->GetStringWidth($substitute) >= $hoursWidth) {  $this->SetFont('Arial', '', $fontSize);  $fontSize--;  }  In questo modo il font è dinamico e il testo verrà sempre stampato all’interno della cella.    Figura Risultato font dinamico.  10h05 – 11h35  Mi sono occupato di commentare tutto il codice sorgente del progetto scritto fino ad ora.  13h15 – 14h45  Ho caricato l’ultima versione dell’applicativo su infomaniak, successivamente dopo svariati test di performance mi sono accorto che in alcuni casi il caricamento del sito è molto lento, questo dovuto anche all’hosting. Ho dunque scaricato tutte le dipendenze localmente e creato una versione compressa di tutte le risorse in modo da far scaricare meno dati agli utenti. Successivamente non essendo contento del tempo di caricamento di circa 6 secondi:    Figura 3 Analisi richieste, 6 secondi per caricare.  Ho aggiunto alla risposta di qualsiasi risorsa degli headers per permettere al browser di eseguire una copia locale dei file per un determinato tempo, in questo caso per un’ora. Il codice è il seguente:  header\_remove("Pragma");  return $response  ->withHeader("Cache-Control", "public, max-age=3600")  ->withHeader("Expires", gmdate('D, d M Y H:i:s', time()+3600).'GMT')  ->withHeader("Content-Type", $type)  ->withBody(file\_get\_contents($file))  ->withStatus(200);  Il codice in questione si occupa di aggiungere l’header Cache-Control per il protocollo HTTP versione 1.1 dicendo di eseguire una cache per un ora dei file. Viene aggiunto l’header Expires per supportare anche HTTP 1.0 e viene rimosso l’header Pragma per permettere la cache.    Figura 4 Analisi richieste, 255 ms per caricare.  La cache ha abbassato i tempi di caricamento da 5,48 secondi a 255 millisecondi sulla stessa pagina ovvero il 2149% più veloce. |

|  |
| --- |
| Problemi riscontrati e soluzioni adottate |
| Problema headers  La libreria di creazione dei PDF si occupa di inviare gli headers adeguati e di stampare il contenuto se il tipo di output è “I”, ho quindi aggiunto una condizione il quale permette di distinguere quale tipo di output si sta utilizzando e in base a quello aggiungere ulteriori headers oppure no. Ho dunque utilizzato il seguente codice per eseguire la distinzione:  public function getContent($type)  {  $output = $this->Output($type);  if ($type == "I") {  return true;  } else {  return $output;  }  }  Il quale poi viene gestito dal controller delle richieste di congedo. Il controllo è il seguente:  if (!ModelsRequests::generatePdfForId($id)) {  return $response->withStatus(404)->withText("Il congedo non è stato trovato!");  }  In questo modo non vengono inviati headers dopo che del contenuto è già stato spedito all’utente e quindi il problema è stato risolto.  Problema salvataggio modifiche congedi  Ho riscontrato un problema nel salvataggio delle modifiche dei congedi il quale era causato dalla riassegnazione di una variabile con dei valori errati. Per risolvero ho semplicemente istanziato un’altra variabile la quale si occupa di tenere lo stato originale della variabile che andrà modificata.  var type = "";  if(substitute.type != "") {  type = " (" + substitute.type + ")";  } |

|  |
| --- |
| Punto della situazione rispetto alla pianificazione |
| Al momento ho un leggero vantaggio rispetto alla pianifica. |

|  |
| --- |
| Programma di massima per la prossima giornata di lavoro |
| Aggiornare la documentazione. |