**Diario di lavoro -15.02.2017**

|  |  |
| --- | --- |
| Luogo | Trevano |
| Data | 14.02.2017 |

|  |
| --- |
| Lavori svolti |
| Riguardando la query di ieri per le assegnazioni dei progetti in corso:  $assignments = Assignment::*leftJoin*('users', 'assignments.user\_id', '=', 'users.id')  ->leftJoin('projects', 'assignments.project\_id', '=', 'projects.id')  ->leftJoin('project\_states', 'projects.state\_id', '=', 'project\_states.id')  ->where('project\_states.state','In corso')  ->get();  mi sono accorto che non venivano visualizzati in modo corretti i dati, l’errore è che l’allievo risulta essere il docente.  Ho cambiato la query nel seguente modo:  Assignment::*join*('users','assignments.user\_id', '=', 'users.id')  ->join('projects','assignments.project\_id','=', 'projects.id')  ->join('project\_states','projects.state\_id','=', 'project\_states.id')  ->where('project\_states.state','In corso')  ->select('users.name AS nome','users.surname AS cognome','projects.name AS progetto')  ->get();  E nel file *resources/views/Assignment/index.blade.php*, ovvero a view delle assegnazioni, posso richiamare i campi nel metodo *select* della query:  <td>{{ $assignment->nome . ' ' . $assignment->cognome}}</td>  <td>{{ $assignment->progetto}}</td>  Invece per quanto riguarda i docenti legati al progetto ho dovuto fare una nuova query poichè non si possono fare in una sola query due join alla stessa tabella. La query è la seguente:  @php($teachers = App\Project::*join*('users','projects.user\_id', '=', 'users.id') ->where('projects.name',$assignment->progetto) ->select('users.name AS nome','users.surname AS cognome') ->get())  Viene utilizzatoil nome del progetto, che è univoco, per andare a recupare il nome del docente responsabile che viene visualizzato tramite un *foreach*:  @foreach($teachers as $teacher)  <td>{{ $teacher->nome . ' ' . $teacher->cognome }}</td> @endforeach  Al momento la gestione delle assegnazioni in corso appare così:    Mentre lo storico delle assegnazioni:    La differenza tra la query delle assegnazioni e quelle concluse è nel where:  *In corso:*  ->where('project\_states.state','In corso')  *Storico*  ->where('project\_states.state','Concluso') |

|  |
| --- |
| Problemi riscontrati e soluzioni adottate |
| Oggi ho provato a mettere nel metodo che controlla i dati inseriti, nel controller, il parametro *unique*:  'name' => 'required|max:255|unique  In questo modo non potevo creare altri oggetti, in questo caso progetti, con lo stesso nome però non potevo neanche salvare un progetto poichè esiste già un record con quel nome univoco.  Guardando il seguente post su <http://stackoverflow.com/questions/23587833/laravel-validation-unique-on-update> ho trovato la seguente soluzione:  'name' => 'required|max:255|unique:projects,name,'.$id,  Dove viene specificato dopo *unique* il nome della tabella e il nome del campo. Mentre *$id* è l’id del record scelto. |

|  |
| --- |
| Punto della situazione rispetto alla pianificazione |
| In linea con la pianificazione |

|  |
| --- |
| Programma di massima per la prossima giornata di lavoro |
| Studio di laravel/implementazione |