Application web de Gestion des Projets

Guide et documentation

**Dévelopée par : El Hosni Yasin & Guedira Nouha 2017/2018**

**Entreprise concernée : Connect To All**

Table des matières

**Introduction1**

Tapez le titre du chapitre (niveau 2)2

Tapez le titre du chapitre (niveau 3)3

**Tapez le titre du chapitre (niveau 1)4**

Tapez le titre du chapitre (niveau 2)5

Tapez le titre du chapitre (niveau 3)6

# Introduction :

Cette application web à été conçus et développée pour but principale de permettre la gestion des projets et taches au sein d’une entreprise ou société, en effet la plateforme offre une Gestion des employés représentant les utilisateurs de l’application, avec une possibilité de suivis des taches de chaque employé ainsi que les projets de chaque chef de projet ,avec des interface graphiques bien accueillantes en vous donnant l’envie de parcourir l’application d’avantage et sans ennuie.

Page d’Installation / Admin authentification

L’ors de votre premier accès à l’application par l’url : *www……….com* , une page d’inscription d’Admin seras mise à votre disposition .

Figure 1 : Admine authentification

Une fois inscris, l’admin à accès à presque toute les actions de l’application.

* **Notes sur le code :**

Views :

**Layout\Structure :**

Se Situe dans le fichier (views/layouts/structure.blade.php),représente le Template principale que tous les autre vues vont hériter (étendre).

* JS :

<https://code.jquery.com/jquery-3.3.1.min.js>

<https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.3/umd/popper.min.js>

<https://stackpath.bootstrapcdn.com/bootstrap/4.1.2/js/bootstrap.min.js>

* Liens CSS :

[*https://stackpath.bootstrapcdn.com/bootstrap/4.1.2/css/bootstrap.min.css*](https://stackpath.bootstrapcdn.com/bootstrap/4.1.2/css/bootstrap.min.css)

[*https://cdn.datatables.net/1.10.19/css/jquery.dataTables.css*](https://cdn.datatables.net/1.10.19/css/jquery.dataTables.css)

[*https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css*](https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css)

*<link href="{{ asset('/css/calendar.css') }}" rel="stylesheet">*

Controllers :

* **HomeController :**

L’ors de l’accès à l’application , la fonction **index()** qui se trouve dans app/Http/Controllers/HomeController.php seras exécuté .

**Fonction Index () :**

Cette fonction vas premièrement vérifier si la base de données des utilisateur est vide en appelant la fonction **CheckFirstRegistration(),** dans le cas vrais , une redirection vers la route 'admin.register.index' seras effectuée.

Dans le cas contraire, la vue 'index' est appelée tout en envoyant les variable suivantes :

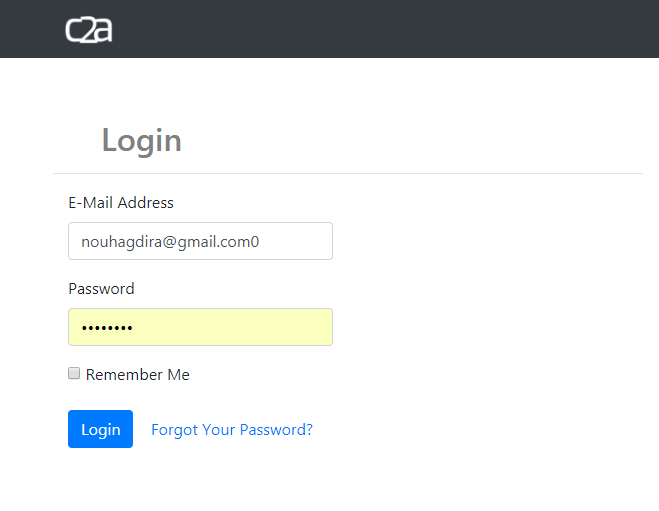
$LastFewProjects : les 3 derniers projects triés par état en-cours.

$LastFewTasks : les 3 dernières Taches triés par état en-cours.



Page de Login / authentification

Une fois la base de donnée des utilisateurs est remplie, l’accès à l’application seras automatiquement initié par une page d’authentification .

Figure 2: Page Login

Cette page vas étendre le layout ***structure.blade.php*** (voir Layout Structure) donc hérite ses scripts et liens css.

L’utilisateur ayant oublié son mot de passe pourras choisir l’option *(Forgot Yout Password ?)* et ensuite seras mené à insérer une adresse email valide, pour recevoire le lien qui le mèneras ensuite à la page de réinitialisation du mot de passe . (action en cours de modification ..)

Page d’accueil

L’utilisateur authentifié auras accès à la page d’acceuil ,celle-ci contient une barre de menu en haut , un calendrier à gauche et Les Derniers Projets et Taches (en-cours) au milieux.

* **Notes sur le code :**

**Le calendrier :**

-Le code html du calendrier se trouve dans :

(views/Calendars/mycalendar.blade.php) , ce fichier utilise l’objet $month de classe (app\Calendar\Month )

-Son css définit dans le template :

*<link href="{{ asset('/css/calendar.css') }}" rel="stylesheet">*

-La classe Month définit les éléments suivant :

* **Variables globales :**

$days : Tout les jour de la semaine

$months : Tout les mois de l’année par nom

$month et $year : par défaut =null : Le mois compris entre 1 et 12 et l’année en entier passées dans l’url de la page .

* **Fonctions de la classe Month :**

**f.GetStartingDay() :** retourne le premier jour du mois courant sous format ( Y-M-01 )

**f.ToString() :** retourne le mois en toute lettre (ex: Janvier 2018)

**f.GetWeeks() :** retourne le nombre de semaine de le mois

**f.WithinMonth($date ) :** vérifie si le Jour ($date) est inclus dans le mois courant !

**f. Inweekend ($date) :** vérifie si le jour est en samedi ou dimanche

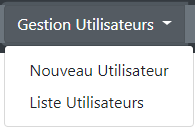
**f. Isfreeday($date)** : vérifie si le jour est une fête nationale estretourne le titre de la fête (pour l’année 2018 – 2019 - 2020)

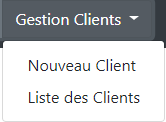
**f.Iscurrentdate($date) :** vérifie si la $date est le jour courant

**f.nextMonth() :** renvois le mois suivant sous format (Objet de Classe Month ).

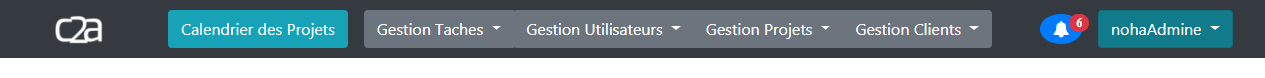
**f.previousMonth() :**renvois le mois précédent .

**Interface Admin et Gérant**







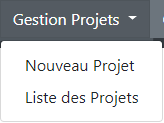


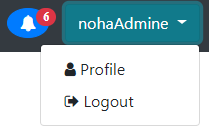
***Notifications :***

Derniers Projets crées

Ou modifiés







NomAdmine

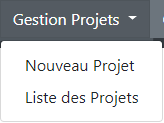
En accédant à un projet l’Admin et le Gérant auront le droit de modifier et clôturer un projet.

NB : Une fois le Projet clôturé , aucune modification n’est permise concernant le projet , ses taches et les membres concernés.

**Interface Chef de Projet**







Liste des Projets de ce Chef de Projet

Lister les taches de cet utilisateur

-Le Chef de Projet à accès aux Projets , ils peu les modifier,mais ne peu pas cloturer un Projet .

-Le Chef de Projet à accès aux Taches , ils peu les modifier,ajouter ou supprimer des membres ,et peu valider la tache .

Views :

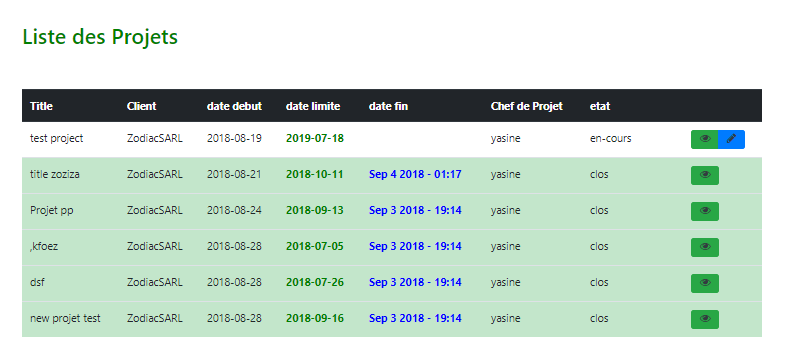
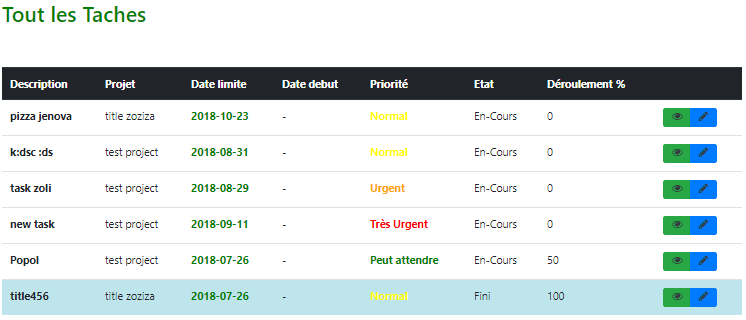
* **Projects\index :**

Figure 3: Liste des Projets

* **Tasks\index :**



Explain ..

Figure 4: Liste des Taches

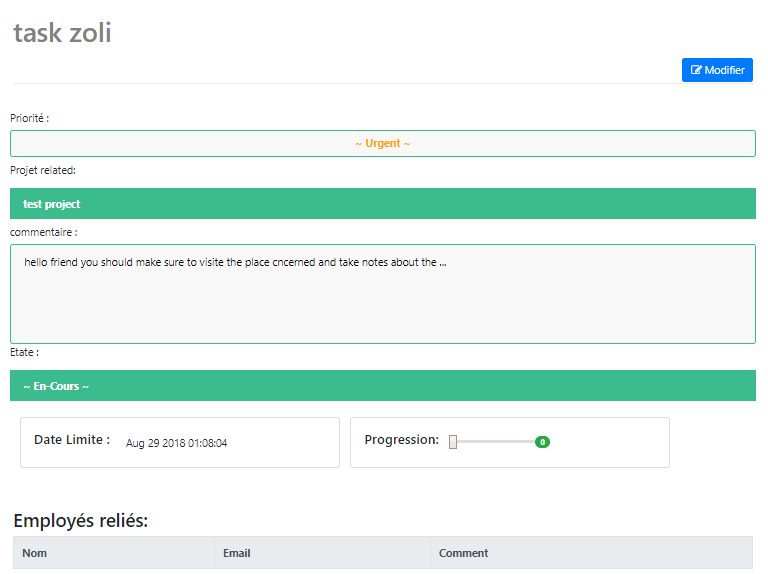
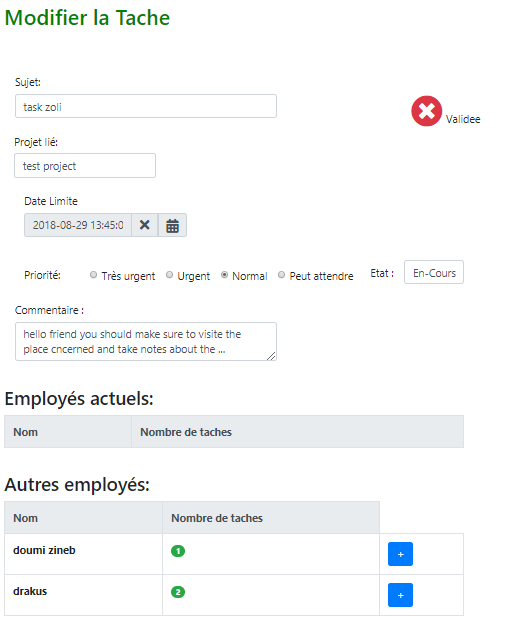
* **Tasks\show :**

Figure 5: Voir Tache

* **Tasks\edit :**

Explain..

Figure 6: Mofier une Tache



Explain the part of adding and removing employees..



**Interface Employé**

Nom

Afficher tout les Projets

Lister les taches attribués à cet employé

* **Notes sur le code :**

Controllers :

* **TaskController :**

**Fonction MyTasks($id):** cette fonction prend en paramètre l’id d’utilisateur courant (l’employé).

Retourne la vue ‘tasks.mesTaches’ avec les variables :

L’array $ts : stockant tout les taches de cet employé

$t->project\_title : retourne le titre du projet relié à chaque tache

**//** les 2 variables si dessous ne sont pas encors pris en charge ils serons traités dans une prochaine version de l’application **//**

**$s\_d :** la date du début de tache(en cours de traitement)

**$f\_d :** la date de fin de tache(en cours de traitement)

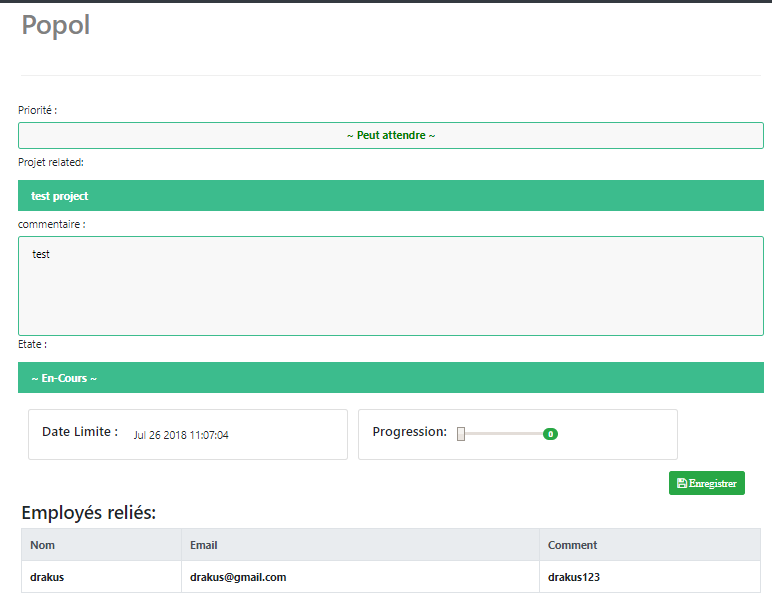
s

Views :

* **Task\show :**

En accédant à une de ses Taches, l’employé peu modifier le pourcentage de progression dans la tache tant que le projet relié à cette tache n’est pas clôturé.

Figure 3: Tache Employé



**Task Name**

