Plan

Introduction

Contexte de travail

Cahier de charges

Méthodologie d’approche

* Module (définition)
* Eléments d’un module

1. Vues
2. Ressources
3. Les processus métier du module
4. La base de données

Implémentation

* Mise en place de l’environnement de travail
* Mise en place de l’environnement de développement
  + Les vues
  + Les ressources
  + Les processus métier
  + La base de données

Conclusion

METHODOLODIE D’APPROCHE

**Définition de Module**

* Les URIs REST pour accéder aux différentes ressources côté serveur
* Les éléments côté client (architecture mvc)
  + Vues, (le « V » de MVC)
  + Contrôleurs (le C de MVC)
  + services
* Les éléments côté serveur
  + Base de données, (le M de MVC)
  + Description des processus métier
  + Les opérations liées aux différentes ressources

IMPLEMENTATTION

**Mise en place de l’environnement de travail**

L’environnement de travail ici est celui dans lequel les utilisateurs de la plateforme pourront en fonction de leur rôle développé, déployé, ou seulement être simples utilisateurs d’une application concernant leurs organisations.

Il est composé :

1. D’une zone d’affichage des tâches en attentes, en gros de différentes sortes de notifications venant des modules dont l’utilisateur connecté a accès.
2. Une fois le module d’intérêt sélectionné, ce dernier s’installe dans l’environnement de travail, la sidebar de gauche subdivisé en trois parties affichant respectivement :
   * Un menu Dashboard qui a pour but de renvoyer sur le **Dashboard** du module encours d’exécution
   * Les différents menus du module encours d’exécution
   * L’ensemble des modules liés à celui en cours d’exécution
3. De l’environnement de travail proprement dit.

**Mise en place de l’environnement de développement**

En guise de test d’évaluation, l’environnement de développement représentant ici le module de génération des modules avenir, à lui-même été développé en tant que module de la plateforme. L’auto-génération d’un module aussi complexe que le générateur lui-même nous rassure de l’efficacité du générateur étant donné qu’il est destiné à générer des applications en entreprise.

Le générateur a été conçu à l’image des composants d’un module précités dans la définition du module comme suit :

**Les vues**

L’interface d’assemblage des vues d’un module est composée de 7 onglets :

1. **Treeview**

Le treeview affiche l’arborescence des composants html d’une vue.

Il permet donc d’avoir un aperçu global sur l’organisation d’une vue ainsi nous pourrons la manipuler facilement en quelques clics. Plus encore, l’arborescence est dotée de quatre fonctions réductrices du temps de développement à savoir :

* + **Up**

Elle permet de remonter un composant dans l’arborescence au sein de son parent .

* + **Down**

Elle fait le contraire de l’up.

* + **Copy**

Elle permet de copier une partie de l’arborescence et de la coller dans un parent, c’est une manière de réutiliser les composants déjà présents dans la vue.

* + **Delete**

Comme son nom l’indique, elle enlève définitivement un composant de la vue.

Le treeview est construit à partir de la palette des composants, les propriétés d’un composant pouvant être renseigner dans la partie « properties ».

1. **Fonctions**

Fonctions formelles ou bouts de code, c’est ici que sont éditées les fonctions qui seront les « Controller » du module.

1. **Actions**

Les actions d’un module sont celles qui s’affichent au niveau de la barre des actions en fonction du module et de la vue courantes. Elles assurent des fonctionnalités du module faisables à partir de la vue courante.

1. **Variables**
2. **configs**

C’est ici que la vue est configurée pour s’encastrer dans le module. Sera-t-elle accessible par le menu du module, quel est son url d’accès, et quel est rôle avoir pour y accéder son autant de questions à répondre pour finaliser la configuration d’une vue.

1. **Composants**

Les composants html utilisés pour construire une vue sont organisés ici en fonction de leurs catégories respectives. Ils sont complétés en fonction du besoin des utilisateurs du module de génération.

1. **Views**

La liste des vues du module en cours d’édition.