OBJECTIF GENERAL

L’objectif du projet consiste à piloter un switch/routeur sur base de la programmation. Le projet s’articule en deux parties, la création d’un menu de pilotage et la communication avec le switch/routeur. En nous basant sur des ressources sur internet, nous répondrions à la question de savoir, les différentes visions de l’articulation entre l’architecture MVC et 3tiers. Ensuite, expliquerons la manière dont notre organisation de code MVC s’articule avec le modèle en couche.

STRATEGIE DE RESOLUTION – METHODOLOGIE

L’architecture utilisé étant le modèle MVC, notre projet sera constitué de trois grands répertoires (Modèle, vue et contrôleur) ainsi qu’un fichier central donc toutes les requêtes(demande) passeront.

* Le fichier central appelé contrôleur principale (ou page de routage), réceptionnera la requête, l’analysera et la transmettra ou non au contrôleur correspondant.
* Le répertoire Modèle : contiendra tous les fichiers (initialement le fichier data.py) ayant très à la gestion de données (la sauvegarde et lecture).
* Le répertoire contrôleur : contiendra tous les fichiers(switchController.py) décrivant la logique de chaque fonctionnalité
* Le répertoire vu : contiendra tous les fichiers (menuVue, showConfigurationVue, updateConfigurationVue, notificationVue) représentant l’interface utilisateurs

Le projet étant réalisé avec le langage python, L’interface utilisateur sera réalisé à l’aide de la bibliothèque Tkinter et la communication avec le switch ou routeur sera fait a l’aide de la bibliothèque Netmiko

DIAGRAMME DE FLUX FONCTIONNEL

Diagram, timeline

Description automatically generated with medium confidence

INTERFACE DE PROGRAMMATION

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Nom de la fonction | Description | Entrée | Sortie |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |