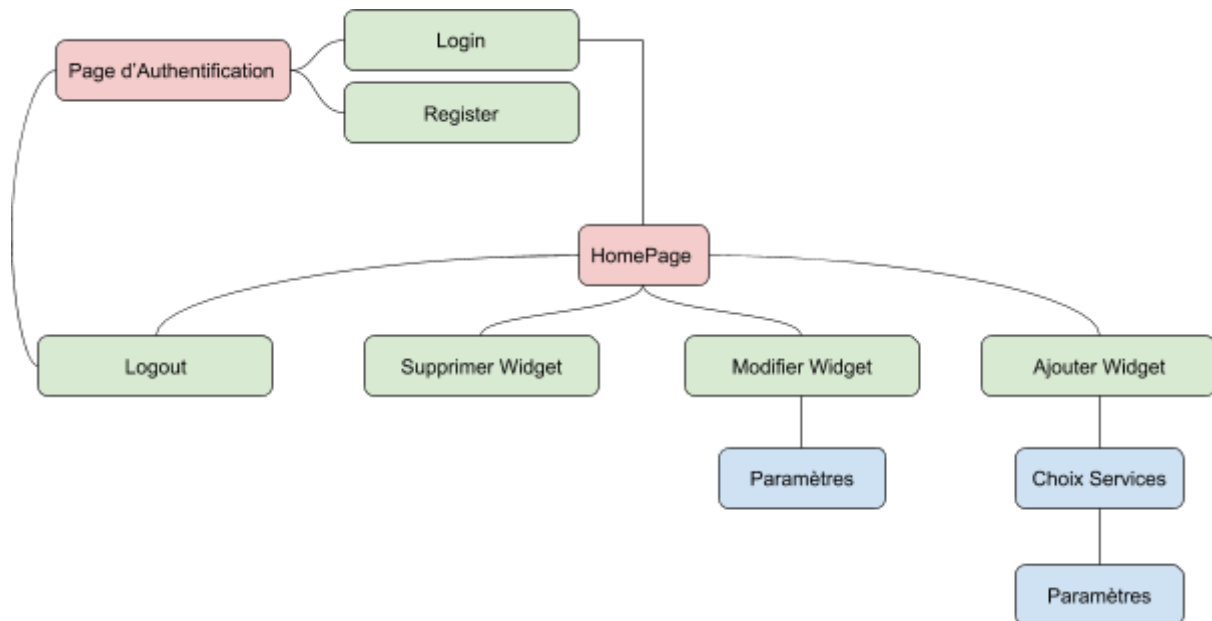


# Documentation: DashBoard



## Page d'authentification et Page de création de compte

Login	Vérification que l'utilisateur existe dans la base de donnée
Register	Vérification que l'utilisateur remplisse tous les champs
Message d'Erreur	Affichage de message d'erreur en fonction de l'erreur associé
Stylesheet	Fichier css utilisés pour la page de login et register : login.css et register.css
Source Page	Fichiers sources utilisés pour cette page : login.ejs et register.ejs

## HomePage

Logout	Redirection vers la page d'authentification
Dashboard	Redirection vers le Homepage
Ajouter un widget	Ouverture d'un formulaire pour entrer différent paramètres
Supprimer un widget	Supprime le widget de la base de donnée par le id et redirection vers le homepage
Modifier un widget	Modifie le widget de la base de donnée par le id et redirection vers le homepage
Source Page	Fichier sources pour cette page : index.ejs
Stylesheet	Fichier css pour cette page : index.css et widget.css

### Ajout de widget

On boucle sur notre tableau de fonction, le tableau de fonction contient les appels aux APIs. Un input sera disponible pour mettre le paramètre et on l'envoie à une route /addWidgets.

### Suppression de widget

Pour chaque widget, nous avons un ID, donc en cliquant sur le bouton "Supprimer", cela va envoyer dans une route /deleteWidgets l'ID de l'instance du widget et ensuite le supprimer.

### Modification de widget

Un modal serait créer pour chaque widget, qui nous enverra son ID à une route /modifyWidgets, on pourra donc chercher son ID et modifier les paramètres du widgets.