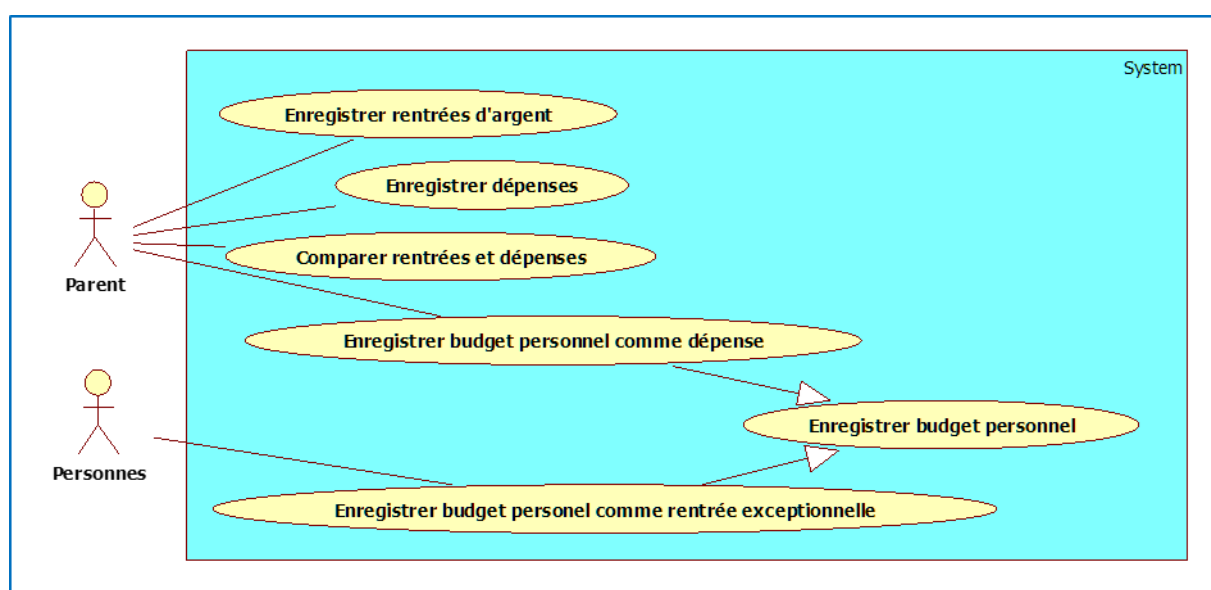


# Réalisation du projet en node.js : Budget familial

Le projet devra être rendu par mail sous la forme d'un fichier compressé contenant tout le projet Node.js, y compris le rapport. Le nom de votre fichier devra être nommé par les 2 noms de famille de votre binôme.

On part de l'étude du projet de budget familial faite en UML.

On a obtenu le diagramme de contexte suivant :



Puis cela a permis de définir les cas d'utilisation suivant :

Pour la saisie des dépenses :

Déroulement	Fonctions identifiées
<b>Le système</b> affiche la page de saisie des dépenses avec des listes de sélection pré-remplies pour <ul style="list-style-type: none"> <li>la personne</li> <li>le poste de dépense</li> </ul>	<code>getListPersonne()</code> déjà identifié <code>getListPoste()</code>
<b>L'utilisateur</b> saisit le montant de la dépense et sélectionne <ul style="list-style-type: none"> <li>la personne, si besoin</li> <li>le poste, si celui-ci est présent dans la liste sinon l'utilisateur peut saisir un nouveau poste</li> </ul>	
<b>L'utilisateur</b> valide la page de saisie des dépenses	
<b>Le système</b> enregistre <ul style="list-style-type: none"> <li>un nouveau poste de dépense, si nécessaire</li> <li>la dépense</li> </ul>	<code>newPoste()</code> <code>newDépense()</code>

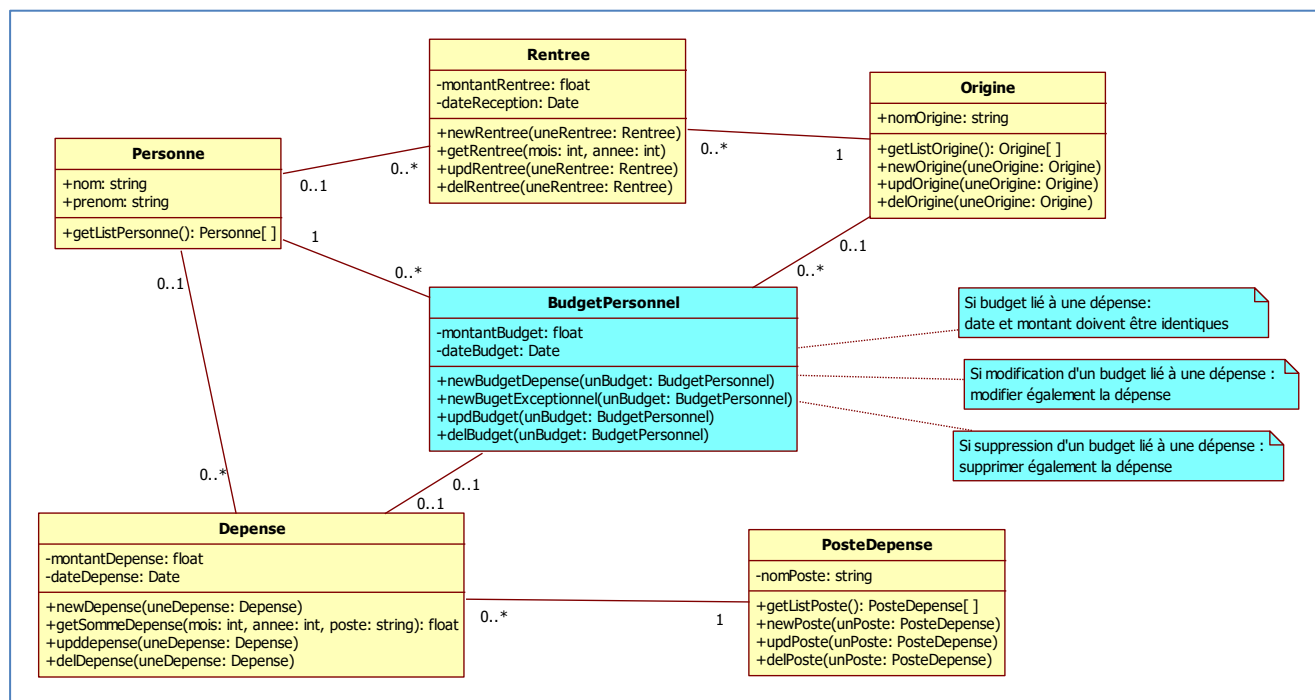
Pour la saisie des rentrées :

Déroulement	Fonctions identifiées
<p><b>Le système</b> affiche la page de saisie des rentrées avec des listes de sélection pré-remplies pour</p> <ul style="list-style-type: none"> <li>- la personne →</li> <li>- l'origine de la rentrée →</li> </ul>	<p>getListPersonne() getListOrigine()</p>
<p><b>L'utilisateur</b> saisit le montant de la rentrée et sélectionne</p> <ul style="list-style-type: none"> <li>- la personne, si besoin</li> <li>- l'origine, si celui-ci est présent dans la liste sinon l'utilisateur peut saisir une nouvelle origine</li> </ul>	
<p><b>L'utilisateur</b> valide la page de saisie des rentrées</p>	
<p><b>Le système</b> enregistre</p> <ul style="list-style-type: none"> <li>- une nouvelle origine de rentrées, si nécessaire →</li> <li>- la rentrée d'argent →</li> </ul>	<p>newOrigine() newRentrée()</p>

Pour la gestion du budget :

Déroulement	Fonctions identifiées
<p><b>Le système</b> affiche la page de saisie du budget avec des listes de sélection pré-remplies pour</p> <ul style="list-style-type: none"> <li>- la personne</li> <li>- l'origine de du budget</li> </ul>	<p>Utilisation des fonctions déjà identifiées : getListPersonne() et getListOrigine()</p>
<p><b>L'utilisateur</b> saisit le montant du budget et sélectionne</p> <ul style="list-style-type: none"> <li>- la personne</li> <li>- l'origine, si celui-ci est présent dans la liste sinon l'utilisateur peut saisir une nouvelle origine</li> </ul>	
<p><b>L'utilisateur</b> valide la page de saisie des rentrées</p>	
<p><b>Le système</b> enregistre</p> <ul style="list-style-type: none"> <li>- une nouvelle origine, si nécessaire →</li> <li>- le budget exceptionnel →</li> </ul>	<p>newOrigine() déjà identifiée newBudgetExceptionnel()</p>

Et enfin le diagramme de classe suivant :



Après avoir étudié la correction de cette étude UML, créer un projet au nom BudgetFamilial. Assurez-vous dès le départ qu'il respecte la structure de paquets suivante :

### Rappels :

- ✚ Pour chaque page créée, il y a des tests unitaires évaluant tous les messages en entrée et en sortie, valides et invalides.
- ✚ Les IHM seront testées et validées par des imprime-écran dans le rapport.
- ✚ Les noms des fonctions UML étant génériques par définition, vous les adapterez pour avoir des noms correspondant
- ✚ Pour chaque action (identification, ajout, modification et suppression) utiliser une **base de donnée**

## L'IHM : interface graphique

Cette partie vous laisse libre cours à votre imagination quant à l'organisation des fenêtres que vous devez créer.

Dans tous les cas, vous devez gérer au moins 3 fenêtres correspondant aux 3 cas d'utilisation : saisie des dépenses et des rentrées, et gestion de budget.

Il est autorisé, voir conseillé, de créer des fenêtres supplémentaires pour la gestion des origines et postes de dépense.

Le programme principal devra démarrer sur une page principale demandant un login à valider, puis ouvrant sur un menu permettant d'accéder aux différentes possibilités.

Votre rapport devra rendre compte des possibilités de navigation dans l'interface et des imprime-écran judicieux des fenêtres créées.

Toute autre amélioration que vous jugerez utile au projet.