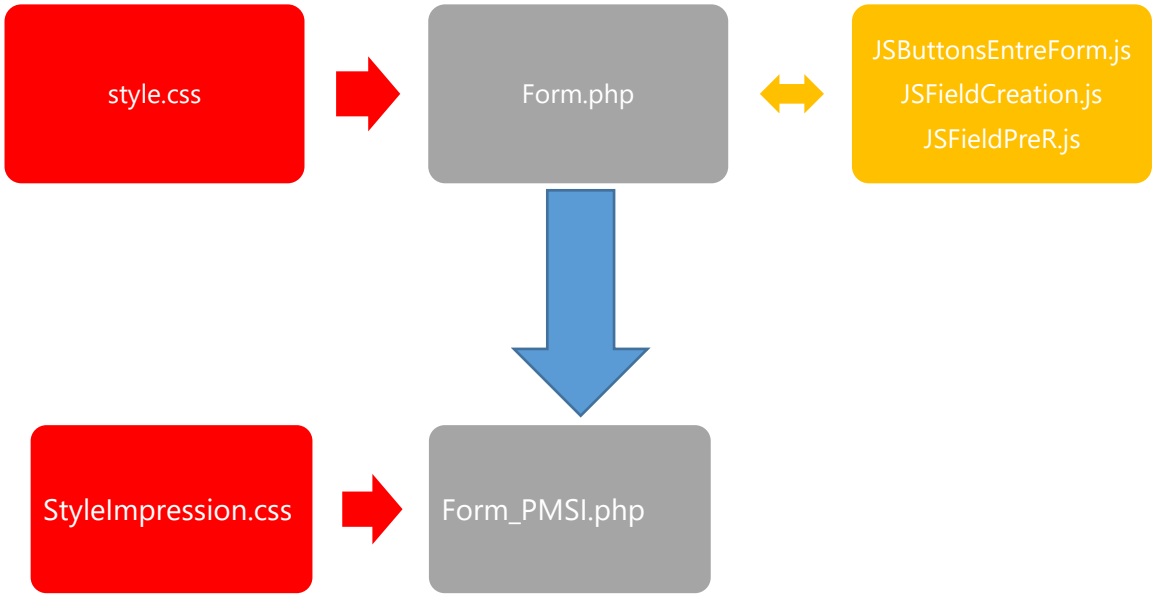
FormEntrée

Documentation technique

Ce document est une documentation technique détaillée de la webapplication « FormEntrée ». Il existe aussi une documentation utilisateur.

Table des matières

1. Organisation globale



L'utilisateur appelle la page Form.php. Cette page va appeler style.css pour la feuille de style, et différents fichiers .js .

Une fois le formulaire remplis, il peut cliquer sur « Générer PMSI » ce qui va transmettre le formulaire, et appeler Form_PMSI.php. Celui-ci utilise comme feuille de style StyleImpression.css.

2. Contenu des différents fichiers

1. Form.php

Fichier principal, qui va appeler comme modules à inclure :

style.css, JSFieldPreR.js, JSFieldCreation.js , JSButtonsEntreForm.js

Form.php va afficher les données ainsi :

- Logo/Titre de l'application
- Choix de localisation (PTG/PTH)
- Différents champs de recueil des données :
    - Histoire de la maladie

    - Examen clinique
    - Projet de soins
- Champs de production, ou les différents éléments pourront être copiés pour être collés dans le logiciel métier.
- Bouton pour valider le formulaire et l'envoyer à Form_PMSI.php .

La philosophie du programme est la suivante : L'utilisateur va remplir les différents champs, et à chaque sélection ou désélection de ces champs, une fonction en JavaScript sera appelée (myFunctionZoneFinale() pour la presque totalité des champs, sauf pour le choix du chirurgien).

Cette fonction va mettre en forme les données puis les faire afficher dans les Champs de production « A copier coller ». Ces champs sont donc remis à jour à chaque utilisation de myFunctionZoneFinale(). Ceci à pour intérêt d'éviter qu'une modification soit faite sans que les Champs de production soient modifiés.

Modification possible : Retirer l'appel à cette fonction à chaque itération, et d'utiliser un bouton final pour l'appeler …

2. style.css

Il s'agit de la feuille de style qui sera appelée / utilisée par Form.php. Elle contient de multiples classes, qui sont commentées dans la feuille de style. Elle permet entre autre l'animation des boutons « Copier ».

3. JSFieldPreR.js

Contient la seule fonction auto_fulfill().

Celle-ci est appelée par Form.php lors de l'utilisation du menu de selection initial PTH/PTG.

Sa fonction est de remplir certains champs (opération, etiologie), mais aussi de sélectionner quelles mesures d'amplitudes faire apparaître (extension ou abduction).

4. JSFieldCreation.js

Contient deux fonctions : myFunctionZoneFinale() et myChirZone().

- myFunctionZoneFinale() :
  Va être appelée presque systématiquement après chaque remplissage de zone et changement (OnBlur). Réintègre à chaque fois toutes les informations remplies dans les champs utilisateurs, va conditionner ces informations, les formater et les faire apparaître dans les Champs de production de Form.php nommés (TexteSortie1, TexteSortie2, TexteSortie3, TexteSortie1b).
- myChirZone() : Va gérer la sélection du chirurgien. A été séparée pour éviter le bug de remise à 0 du champ « Consultation de contrôle radioclinique avec le Dr xxx le xxx »

Modification possible : De façon générale, on peut s'interroger sur la possibilité / nécessité de segmenter les fonctions pour éviter un nouveau calcul de tout le formatage après chaque modification de zone. Augmente la complexité pour le développeur, épargne quelques (négligeables ?) ressources machine.

5. JSButtonsEntreForm.js

Le script appelé par les différents boutons de « copier » : Commenté dans le fichier, l'idée est d'écouter le bouton, et au clic, va lancer la selection du champ désigné, et dans un test copier le contenu dans le presse papier. Si réussi, affiche l'animation, sinon lance un message console.

6. FormPMSI.php

Sera appelé à la validation du formulaire contenu sur Form.php.

Il contient :

- Les différentes variables contenant les POST_ précédents.
- L'affichage de différentes informations dont la date, celle du lundi de la semaine, le numéro de la semaine, l'UF du service.
- Le tableau du haut (va reprendre de façon stéréotypée la Finalité principale de prise en charge (préremplie Prise en charge rééducative), la manifestation etiologique principale et l'affection étiologique. Avec la zone de codage correspondant.
- Le tableau du milieu qui consistera en une répétition de lignes avec deux zones : A droite le Label , à gauche, le code CIM10. Ce tableau fait appel à une classe pour être construit cf ci-dessous.
- Le tableau du bas, qui lui aussi est construit à l'aide de la classe. On préremplis l'ECG systématiquement.

- Concernant la classe TableauGen :

    - Son objectif est de permettre de :
        - digérer des données (soit unitaires, soit listes)
        - créer un tableau
        - le remplir avec ces données
    - Elle se construit à l'aide de $nom_du_tableau new TableauGen(nombre_de_lignes) ;
    - Ses paramètres : Sont tous privés donc non accessibles de l'extérieur.
    - Ses méthodes (publiques) :
        - print_debut() : Créé les premières lignes nécessaires à la création et la mise en forme de ce tableau.
        - attr_var_labl_list() : Va récupérer la liste passée en argument, et pour chaque élément la mettre dans _my_array.
        - attr_var_labl_item() : Va intégrer dans _my_array la chaine de caractères passés en argument.
        - array_construct() : Récupère tout le tableau virtuel _my_array, avec les variables modifiées, et va imprimer les différentes lignes du tableau physique.
        - print_fin() : Pour les dernières lignes du tableau.

- debug_array() : Va afficher le tableau virtuel _my_array.

Modification possible : Utiliser un tableau CIM 10 pour récupérer les codes automatiquement, mais la précision des labels de cette classification peut rentre la tâche ardue.

7. StyleImpression.css

La feuille de style css appellée par FormPMSI.php.

3. Pour tout renseignement, mentions légales.

L'ensemble du code est disponible sur :

Pour tout renseignement : Alain Carrot datacar38@gmail.com

Ce site n'utilise pas de cookies, et n'enregistre aucune donnée personnelle.

La web application « FormEntrée » est un outil visant à simplifier la mise en page, et tout élément qui serait copié dans un dossier médical devra être relu par le praticien utilisateur pour en vérifier la validité.

L'auteur de la webapplication « FormEntrée » ne peut en aucun cas être tenu responsable de la pertinence, de l'exactitude, de l'intégrité ou de la qualité du contenu mis en ligne. La responsabilité de l'auteur de ce site ne peut être engagée en cas de dommages matériels ou intellectuels résultant de l'utilisation ou de la non-utilisation des informations contenues dans le site web, d'informations erronées ou incomplètes, dans la mesure où il ne peut pas être établi qu'il s'agit d'un acte délibéré ou d'une négligence de la part de l'auteur.
Toutes les informations contenues dans ce site sont libres et sans engagement. L'auteur de ce site se réserve expressément le droit de modifier, de compléter ou de supprimer tout ou partie du contenu du site sans préavis ainsi que d'en suspendre la publication à titre temporaire ou définitif.