Fondation Mérieux

LabBook

LabBook v3.6

Fiche patient et formulaires d'historique

Personnaliser les formulaires avec des fichiers TOML

V3.0

Octobre 2026

Table des matières

1. À qui s'adresse ce guide 3
2. Règles communes 4
3. La fiche patient 5
3.1 Section [description] : les éléments 5
3.2 Section [layout] : la disposition 6
3.3 Exemple pas à pas : ajouter « Établissement d'origine » 7
4. Le formulaire d'historique patient 8
4.1 Trois sections obligatoires 8
4.2 Règle de nommage des blocs 8
4.3 Le formulaire livré avec LabBook 3.6.28 8
4.4 Exemple pas à pas : ajouter un bloc « Traitements en cours » 9
4.5 Exemple minimal complet 10
5. Où l'utilisateur voit-il ces formulaires ? 11
6. Vérifier et dépanner 12
Annexe — Historique du document 13

1. À qui s'adresse ce guide

Public	Administrateur LabBook ou référent technique, à l'aise avec un éditeur de texte
Accès	**Structure › Configuration de formulaires** (guide D1, § 8.7)
Fichiers	`form_patient_<langue>.toml` (fiche patient) et `form_patient_hist_<langue>.toml` (historique)

LabBook décrit deux formulaires dans des fichiers texte au format **TOML** :

- la **fiche patient**, remplie à la création du patient : identité, coordonnées ;
- le **formulaire d'historique**, rempli à chaque demande : antécédents, symptômes, constantes, traitements… Il est **horodaté** et conservé à chaque saisie.

On peut ainsi **ajouter des champs propres au laboratoire**, masquer des champs inutiles ou réorganiser l'écran, sans toucher au code de LabBook.

> **Attention — méthode de travail**
>
> 1. **Téléchargez** toujours le fichier en place et gardez-en une copie.
> 2. Modifiez une copie, vérifiez la syntaxe (§ 6), testez avec **Aperçu**.
> 3. Ne déposez le fichier qu'ensuite.
>
> Un fichier mal formé peut empêcher l'affichage du formulaire.

2. Règles communes

Règle	Détail
Encodage	**UTF-8**
Nom du fichier	`form_patient_<LANG>.toml` ou `form_patient_hist_<LANG>.toml`
Codes langue	`fr`, `uk`, `us`, `es`, `ar`, `km`, `lo`, `mg`, `pt`
Langue par défaut	Si le fichier de la langue choisie n'existe pas, LabBook utilise le fichier `fr`
Fichier de test	Tout fichier qui commence par `form_patient_` (ou `form_patient_hist_`) et finit par `.toml` peut être déposé et testé avec **Aperçu**
Indentation	Les tabulations et l'alignement comptent : gardez la présentation du fichier d'origine
Plusieurs langues	Décrivez **les mêmes champs** dans chaque langue. Sinon, l'information saisie dépend de la langue de l'utilisateur
Emplacement sur le serveur	`storage/resource/form/patient/`

3. La fiche patient

Le fichier contient **deux sections obligatoires** :

- `[description]` : la liste des éléments ;
- `[layout]` : leur disposition à l'écran.

3.1 Section `[description]` : les éléments

Chaque élément commence par la ligne `[[description.form_element]]`.

Éléments prédéfinis

Les éléments prédéfinis sont des champs connus de LabBook : ils alimentent les colonnes de la fiche patient en base. On les déclare avec `labbook_ref` et `input_type`.

`labbook_ref`	Libellé	`input_type`	Remarque
`pat_ano`	Patient anonyme	`radio`	Oui / Non
`pat_code_lab`	Code patient interne au laboratoire	`text`	
`pat_code`	Code unique LabBook	—	Affichage seul, généré automatiquement ; pas d'`input_type`
`pat_name`	Nom	`text`	
`pat_midname`	Deuxième nom	`text`	
`pat_maiden`	Nom de jeune fille	`text`	
`pat_firstname`	Prénom	`text`	
`pat_sex`	Sexe	`radio`	Masculin / Féminin / Inconnu
`pat_birth`	Date de naissance	`date`	
`pat_birth_approx`	Date de naissance approximative	`radio`	Oui / Non
`pat_age`	Âge	`text` / `number`	
`pat_age_unit`	Unité de l'âge	`select`	Jours / Semaines / Mois / Années
`pat_nationality`	Nationalité	`select`	Liste fournie par LabBook
`pat_resident`	Résident	`radio`	Oui / Non
`pat_blood_group`	Groupe sanguin	`select`	A / B / AB / O
`pat_blood_rhesus`	Rhésus	`select`	+ / –
`pat_address`	Adresse	`textarea`	
`pat_phone1`, `pat_phone2`	Téléphones	`text`	

labbook_ref	Libellé	input_type	Remarque
pat_email	Email	text	Utilisé pour l'envoi des résultats (guide C4)
pat_profession	Profession	text	
search_zipcity	Recherche code postal / ville	—	Fonctionne si pat_zipcode et/ou pat_city sont présents (guide D1, § 8.6)
pat_pbox	Boîte postale	text	
pat_district	Quartier / secteur	text	
pat_zipcode	Code postal	text	
pat_city	Ville	text	

Exemple — rendre le sexe obligatoire :

```
[[description.form_element]]
labbook_ref = "pat_sex"
input_type = "radio"
attr_required = true
```

Éléments personnalisés

On les déclare avec un `id` unique et un `label`, puis :

- `type`, pour un élément d'**affichage** : `h1` à `h6` (titres) ou `span` (texte) ;
- **ou** `input_type`, pour un champ de **saisie** : `text`, `textarea`, `number`, `select`, `radio`, `date`, `datetime-local`.

Attribut	Usage
`options = [{ value = "…", label = "…" }, …]`	**Obligatoire** pour `select` et `radio`
`attr_required = true`	Champ obligatoire : un astérisque est ajouté au libellé
`attr_value = "…"`	Valeur par défaut
`attr_rows`, `attr_cols`	Taille d'un `textarea`
`attr_min`, `attr_max`, `attr_step`	Bornes et pas d'un `number`

3.2 Section `[layout]` : la disposition

Chaque ligne d'écran commence par `[[layout.rows]]`, avec une liste de classes (`class`) et d'éléments (`elements`). Une ligne peut contenir des colonnes, qui contiennent elles-mêmes des éléments.

Classe	Effet
`row`	Une ligne

Classe	Effet
`col`, `col-md-6`, `col-md-12`	Une colonne ; `col-md-6` prend la moitié de la largeur sur un écran large
`mt-1` à `mt-3`	Marge en haut (`mb` en bas, `ms` début, `me` fin, `mx` horizontale, `my` verticale)
`panel-heading`	Ligne de titre, soulignée
`panel-title`	Titre en gras et vert
`flex-row`, `flex-md-row`	Aligne les éléments horizontalement (`md` : seulement si l'écran fait au moins 768 px)
`d-md-flex`	Transforme le conteneur en *flexbox*

Ces classes viennent de **Bootstrap 5** : https://getbootstrap.com/docs/5.0/layout/breakpoints/

3.3 Exemple pas à pas : ajouter « Établissement d'origine »

Besoin : le laboratoire reçoit des patients envoyés par trois centres de santé et veut le noter sur la fiche.

Étape 1 — Déclarer le champ dans `[description]`, par exemple après la profession :

```
[[description.form_element]]
id = "pat_origin"
label = "Établissement d'origine"
input_type = "select"
options = [
    { value = "", label = "" },
    { value = "CS_PIKINE", label = "Centre de santé de Pikine" },
    { value = "CS_THIAROYE", label = "Centre de santé de Thiaroye" },
    { value = "AUTRE", label = "Autre" }
]
```

Étape 2 — Le placer dans `[layout]`, par exemple dans la colonne des coordonnées :

```
{ element = "pat_origin", class = "flex-md-row" }
```

Étape 3 — Tester : enregistrez le fichier sous `form_patient_fr.toml`, déposez-le, puis cliquez sur **Aperçu**. Corrigez si besoin, puis déposez la version définitive `form_patient_fr.toml`.

Étape 4 — Utiliser la donnée sur le compte rendu (facultatif) : dans le modèle de document, la valeur est disponible sous `o.pat.pat_origin`, à encadrer par un test d'existence (guide D3).

4. Le formulaire d'historique patient

Il permet de saisir, à chaque demande, des informations **datées** qui s'accumulent : antécédents, symptômes, constantes… Chaque bloc affiche un **tableau d'historique** des saisies précédentes.

4.1 Trois sections obligatoires

Le fichier doit contenir **exactement** ces trois sections de premier niveau, et aucune autre :

Section	Rôle
`[description]`	Tous les champs, avec la même syntaxe que pour la fiche patient (§ 3.1)
`[history]`	Les **blocs** d'historique et les colonnes de leur tableau
`[layout]`	La disposition, y compris l'emplacement de chaque tableau d'historique

4.2 Règle de nommage des blocs

Chaque bloc a un identifiant (`id`). **Tous les champs du bloc commencent par cet identifiant suivi de « _ ».**

Bloc (`id`)	Champs
`sympt`	`sympt_name`, `sympt_date`
`const`	`const_temp`, `const_weight`, `const_bp`, `const_glucose`, `const_date`
`treat`	`treat_drug`, `treat_dose`, `treat_date`

Dans `[history]`, `columns` liste les champs affichés dans le tableau. Les autres champs du bloc restent saisissables et enregistrés, mais ne sont pas affichés dans le tableau.

Dans `[layout]`, l'élément `{ element = "<id du bloc>" }` place le **tableau d'historique** du bloc.

4.3 Le formulaire livré avec LabBook 3.6.28

Le fichier par défaut contient trois blocs :

Bloc	Champs
Antécédents médicaux (`antecedents`)	Description, Date *
Symptômes (`sympt`)	Symptôme * (liste : difficultés à respirer, écoulement nasal, fièvre, yeux rouges…), Date d'apparition *
Constantes (`const`)	Température (°C), Poids (kg), Tension artérielle, Taux de sucre (g/L), Date des constantes *

4.4 Exemple pas à pas : ajouter un bloc « Traitements en cours »

1. Les champs, dans `[description]` :

```
[[description.form_element]]
id    = "label_treat"
label = "Traitements en cours"
type  = "h3"

[[description.form_element]]
id         = "treat_drug"
label      = "Médicament"
input_type = "text"
attr_required = true

[[description.form_element]]
id         = "treat_dose"
label      = "Posologie"
input_type = "text"

[[description.form_element]]
id         = "treat_date"
label      = "Date de début"
input_type = "date"
```

2. Le bloc, dans `[history]` :

```
[[history.block]]
id      = "treat"
label   = "Historique des traitements"
columns = ["treat_drug", "treat_dose", "treat_date"]
```

3. La disposition, dans `[layout]` : un titre, le formulaire, puis le tableau.

```
[[layout.rows]]
class    = "panel-heading row mt-4"
elements = [ { element = "label_treat", class = "panel-title" } ]

[[layout.rows]]
class    = "row mt-2"
elements = [
    { class = "col-md-12", elements = [
        { element = "treat_drug" },
        { element = "treat_dose" },
        { element = "treat_date" }
    ] }
]

[[layout.rows]]
class    = "row mt-2"
elements = [ { element = "treat", class = "col-md-12" } ]
```

4.5 Exemple minimal complet

```
[description]
[[description.form_element]]
id = "label_med"
label = "Antécédents médicaux"
type = "h3"

[[description.form_element]]
id = "med_datetime"
label = "Date et heure"
input_type = "datetime-local"

[[description.form_element]]
id = "med_comment"
label = "Commentaire"
input_type = "textarea"
attr_rows = "3"
attr_cols = "40"

[layout]
[[layout.rows]]
class = "panel-heading row"
elements = [ { element = "label_med", class = "panel-title" } ]

[[layout.rows]]
class = "row mt-2"
elements = [
  { class = "col-md-6", elements = [
      { element = "med_datetime", class = "flex-md-row" },
      { element = "med_comment", class = "flex-md-row" }
  ] },
  { element = "med", class = "col-md-6" }
]

[history]
[[history.block]]
id = "med"
columns = ["med_datetime", "med_comment"]
```

Ici, le formulaire est à gauche et le tableau d'historique à droite (deux colonnes `col-md-6`).

5. Où l'utilisateur voit-il ces formulaires ?

Formulaire	Écran	Rôles
Fiche patient	**Administratif › Nouvelle demande externe / hospitalisé › Nouveau patient**, ou **Modifier le dossier patient**	Secrétaires, techniciens, préleveurs, biologiste
Historique	Même parcours, bouton **Modifier l'historique patient**	Mêmes rôles

6. Vérifier et dépanner

Problème	Cause fréquente	Solution
Le formulaire ne s'affiche plus	Erreur de syntaxe TOML : guillemet, crochet, virgule	Passez le fichier dans un validateur (ex. https://www.toml-lint.com/) ; redéposez la copie d'origine
Un champ n'apparaît pas	Élément déclaré dans `[description]` mais absent de `[layout]`	Ajoutez-le dans une ligne de `[layout]`
Une liste est vide	`options` manquant pour `select` / `radio`	Ajoutez `options = [...]`
Le tableau d'historique est vide ou incomplet	Champs qui ne commencent pas par `<id du bloc>_` ; colonne absente de `columns`	Renommez les champs ; complétez `columns`
Erreur « section non autorisée » (historique)	Une section autre que `description`, `history` ou `layout`	Supprimez-la
Le formulaire change selon la langue	Fichiers de langues différents	Alignez le contenu des fichiers `fr`, `uk`, `pt`…

Annexe — Historique du document

Version	Date	LabBook	Modifications
3.0	10/2026	3.6.28	Refonte : règles communes, tableau complet des éléments prédéfinis (dont `pat_email`), deux exemples pas à pas, description du formulaire par défaut 3.6.28, dépannage
2	01/2026	3.6	Version précédente