

## 2. 本書の目的、想定する読者

本書は、健康・医療行政を担う者、医療機関の担当者等が、疾患登録管理システム構築の一例として参照し、それぞれの立案のヒントになることを意図して編集された。

本書の作成にあたり、以下の手順で要件を抽出し、取りまとめることとした(図1)。

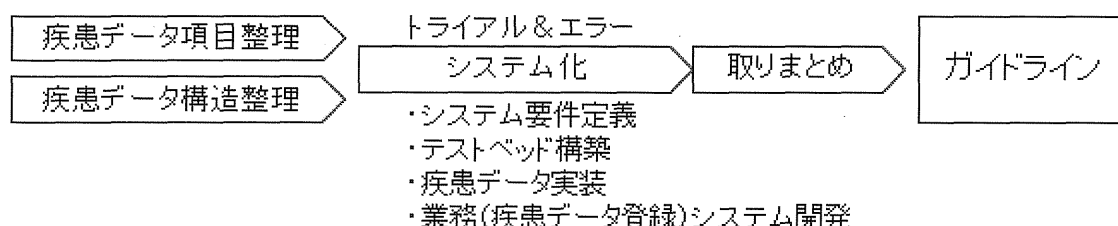


図1 作成手順

### ① 疾患データ項目整理

疾患データを登録するに当たり、疾患を構成する項目を網羅する。また、それぞれの項目ごとのオプションとして、入力必須の有無、凡例、単位、数値の幅などを定義する。疾患によって、必須となる項目や採用する数値の幅は異なる場合がある。

### ② 疾患データ構造整理

疾患データを構成するモジュールと、モジュールで選択するデータ項目を定義する。この時、各データ項目のオプションを選択する。

### ③ システム化

疾患データ構造の取り扱いや、ユーザインターフェースの共通化、データベースとのやり取りなど、システム面の柔軟性や拡張性を担保しながら、操作面のわかりやすさを実現するシステムの構築方法を検討する。

### ④ とりまとめ

上記のプロセスで得られた情報を、ノウハウや TIPS として整理する。

### ⑤ ガイドライン作成

文書として体裁を整え、表現等の整合をとり、読みやすさや使いやすさの観点で推敲して最終版を作成する。

本書の中に出てくる製品名は、法律によってそれぞれの製造者の著作権が保護されている。また、本書の著作権は国立保健医療科学院に帰属し、出典なしに引用したり、一部または全部を複製したりすることはできない。

### 3. 疾患登録管理システム全体構造

#### 3.1. テストベッドサンプル開発の関係

下図に、希少・難治性疾患の登録システム、テストベッド、及びテストベッドでの動作サンプル開発の関係を示す。

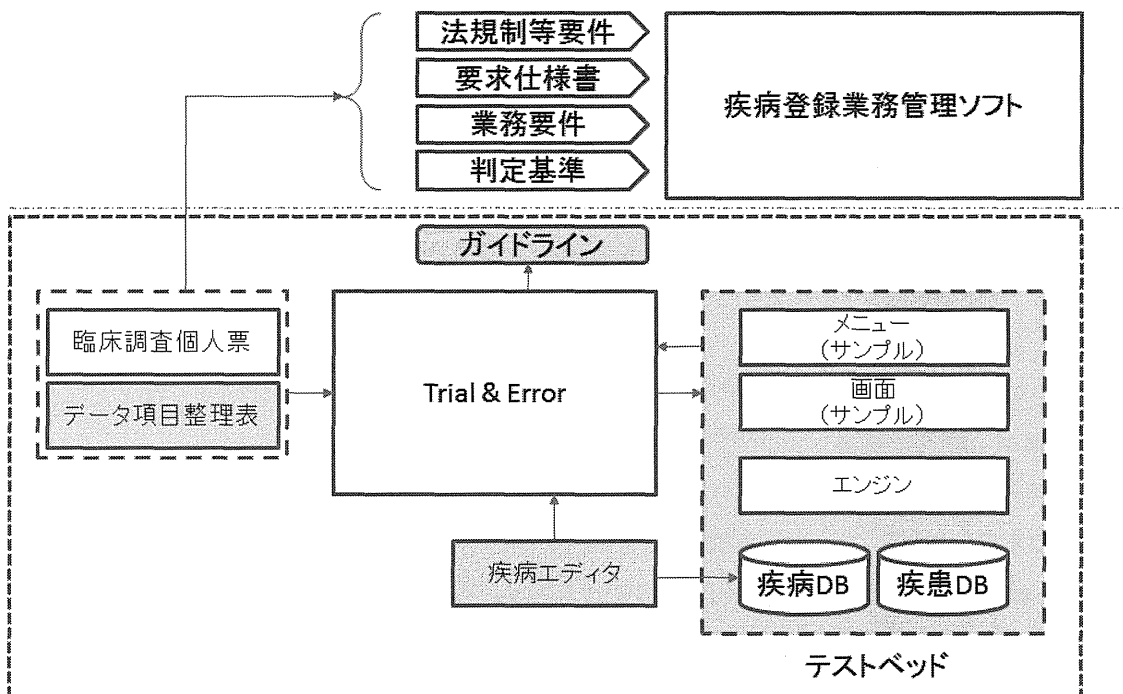


図 1: テストベッドサンプル開発の関係

### 3.2. 開発までの流れ

下図に、開発に必要な項目の全体構成と開発順序を示す。

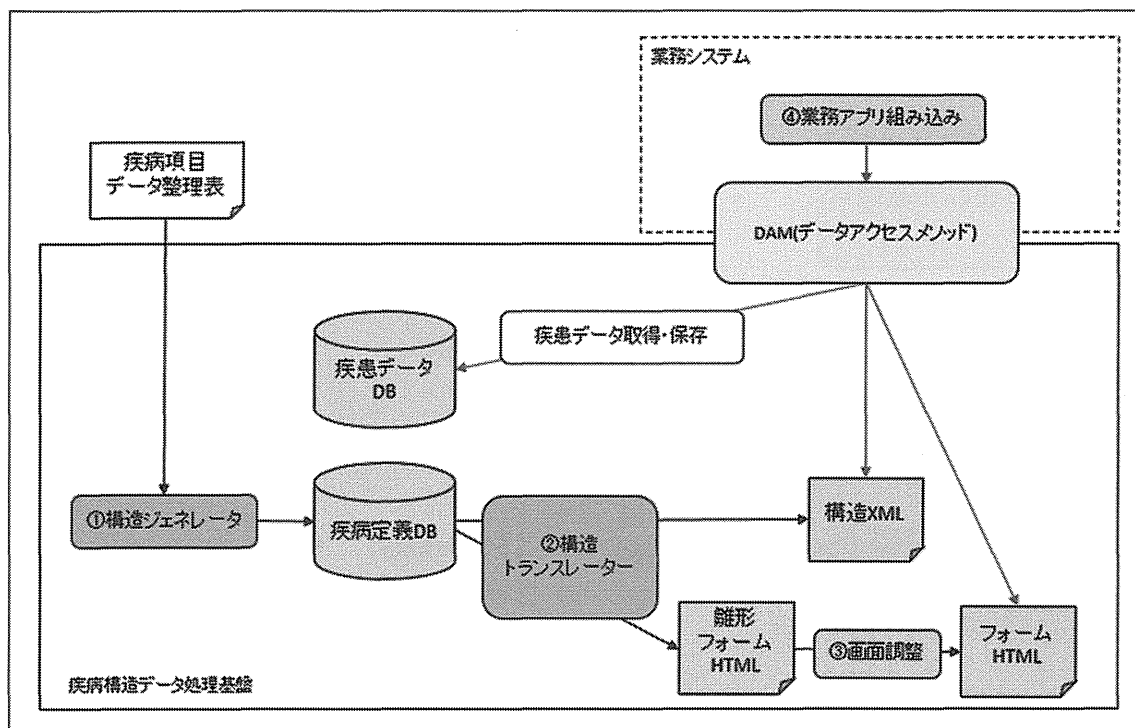


図 2:テストベッドサンプル開発の関係

#### 開発の順序

- ④ 業務アプリ組み込み:業務アプリに DAM(データアクセスメソッド)を利用して疾患データ取得・保存処理を組み込む
- ③ 画面調整:入力フォーム HTML のレイアウトと表示ロジックを調整する
- ② 構造トランスレータ:構造トランスレータを使用して構造 XML と雛形となる入力フォーム HTML を生成
- ① 構造ジェネレータ:構造ジェネレータを使用して疾病定義 DB を構築

### 3.3. 業務アプリ開発の流れ

各業務アプリケーションは、疾病の情報を登録する機能と、業務ごとの固有の機能に分けることができる。疾病の情報の登録に関しては、同様の入力を業務ごとに実装するのは非効率である。そのため、疾病の登録部に関しては共通化を行い、入力フォームを統一する。これらの入力フォームは、サンプルベッドでの成果を元に以下の手順で開発を行うものとする。

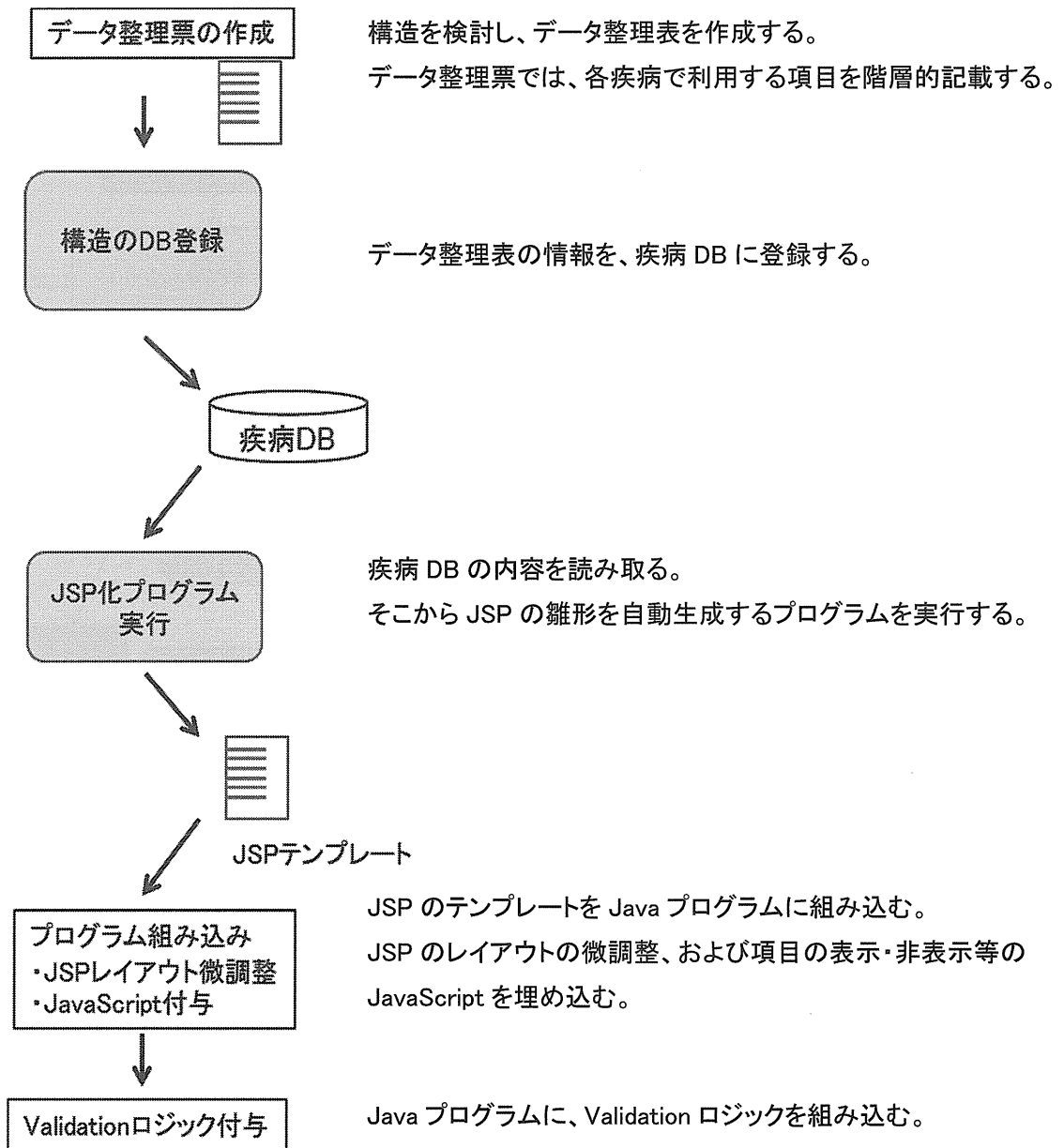


図 3:テストベッドサンプル開発の関係

フォーム等の登録には、API を用意しその API を呼び出しすることで、表示する。

HTML や業務に特化した部分に関しては、自由にレイアウトしてもよいものとする。統一入力フォームは、iframe 等で内部に表示するようにする。(図 4)

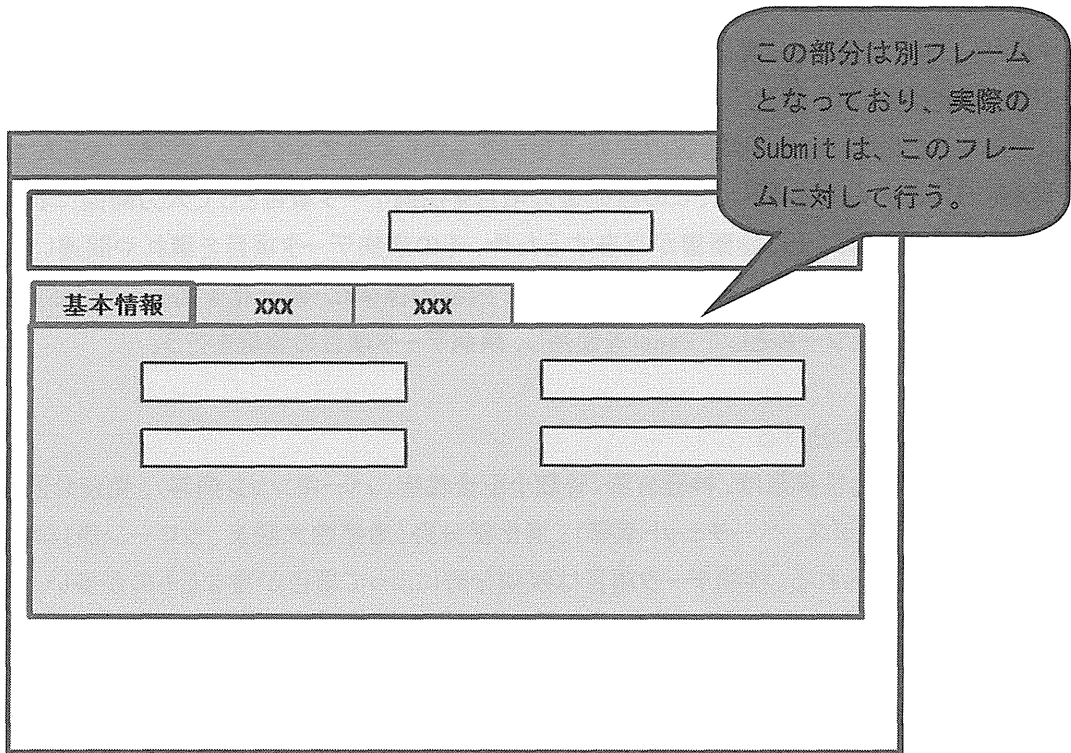


図 4: 業務アプリと疾患入力フォームの関係

## 4. データ整理表の作成

### 4.1. Excelによるデータ整理

希少・難治性疾患には医療費助成における公的書類である診断書(臨床調査個人票)が存在しており、各疾病によって診断書の記載項目があらかじめ決まっている。登録システムで医師が患者のデータを入力する際は、この診断書の項目を疾病データ項目として入力画面に表示させる。数百にわたる疾病ごとに診断書が存在するため、その疾病データ項目を漏れや間違いのないよう入力画面化していくには、項目を整理統合し、内容の標準化・統一化を図ることが必要である。この用途に適したツールとしてExcelを使用し、疾病データ項目の整理を行う。

#### (1) 疾病分野に分ける

希少・難治性疾患は、神経分野(脊髄小脳変性症、パーキンソン病等)、免疫分野(全身性エリテマトーデス、ベーチェット病等)、消化器分野(潰瘍性大腸炎、クローン病)などの疾病分野に分けられる。疾病データ項目は疾病分野によって項目が重なるものが多い。そこで疾病分野ごとにExcelのデータ整理表を作成し、共通項目の洗い出しを可能にする。

#### (2) 診断書から項目とその属性を拾い上げる

診断書に記載されている各項目について、「①、見出し」「②、①に属する質問項目」「③、②の補足質問」といった階層構造を踏まえながらデータ整理表上に記載する。項目名のほか、その属性(数値であればその単位、選択項目であれば選択肢)も記載する。

表 1: データ整理表の例における各列の内容

NO.	名称	説明	
1	第1階層	見出し1 :大見出しにあたる。(各疾病分野で共通)	
2	項目の名称	第2階層	①. 見出し2:見出しにあたる。
		第3階層	②.①に属する各質問項目。
		第4階層	③.②の下層に属する質問項目。
3	データ型	各項目のタイプ: ① 列挙型: 選択肢をチェックするタイプ。 ② 数値型: 数値を入力するタイプ。 ③ 文字型: 文字を入力するタイプ。 ④ 2数値型: 年月を入力するタイプ(「□年□月」、「□年□ヶ月」等) ⑤ 3数値型: 年月日を入力するタイプ。(□年□月□日)	
4	単一選択/複数選択	データ型が列挙型の場合: 選択肢を単一選択(ラジオボタン)又は複数選択(チェックボックス)であるかを示す。	
5	単位	データ型が数値型の場合: 数値の後ろに表示される単位。	

6	選択肢	各選択肢。
13	疾病名	各疾病で必要となる項目に「○」印を記載。

### (3) 項目の共通化

診断書上では、疾病によって同一項目に対する表現が違う場合がある。共通化できるものは共通化し、項目の重複を避ける。「ヘモグロビン」と「Hb」は同一のものであり、共通化が可能である。(図 5)

見出し	項目名	単位	疾患A	疾患B	疾患C	疾患D	疾患E
血液検査			○	○	○	○	○
	赤血球数	$\times 10^4/\mu\text{L}$	○				
	白血球数	$/\mu\text{L}$	○		○		○
	ヘモグロビン	g/dL	○				○
	Hb	g/dL			○		
	ヘマトクリット	%	○				
	血小板数	$\times 10^4/\mu\text{L}$	○		○		○
	リンパ球	%			○		

図 5: 項目の重複の例

## 4.2. 構造を意識した疾病データ項目整理

Excel によるデータ整理は、ある分野の専門家がその専門の疾病の整理を行う際には適しているが、疾病分野をまたがって共通項目を整理するには不向きである。そのため、疾病全体を管理し、別の疾病分野を参考に新しい疾病の項目を整理する場合には課題が残る。また、Excel 上では、同一項目が違う階層に存在すると共通化が不可能であり、項目の統一化という面では不十分である。こうした課題点を改善し、疾病全体における統合管理に適した方法として、疾病データを構造として捉えデータ整理を行うことが求められる。

### (1) 全体構造の設計

文書には構造があるように、さまざまな資料にも構造がある。例えば、一般的な文学作品の本は、「表紙」「裏表紙」「奥付」「目次」「本文」「あとがき」「解説」などの構造によって成り立っている。ビジネス書や技術書の場合は、索引や出典・引用元などの情報がこれに加わる。

データを整理するためにはこのような構造化が必須要件であり、こうした構造定義のことを、文書データベースでは DTD、XML 文書では XML スキーマなどという。こうした全体構造と個々のデータ項目名などが共通認識となっていれば、表示やデータ処理を容易にコントロールすることが可能となる。

カバーすべき疾患情報のすべてを網羅し、特性を分析することが可能な構造を決定しなければならない。また、登録分野によっては ISO13606 アーキタイプのような要素を当初から組み込むことも想定すべきである。

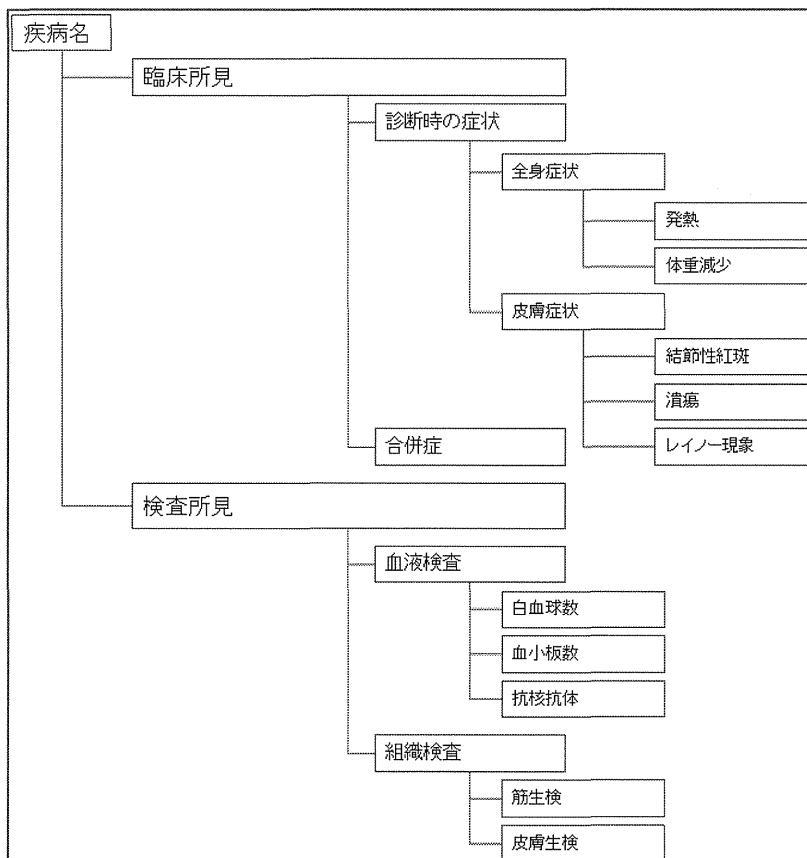


図 6: 疾病データ項目名における階層構造の例

第1階層	第2階層	第3階層	第4階層	データ型	単一選択 /複数選択	単位	選択肢1	選択肢2	選択肢3	選択肢4	疾患 A	疾患 B	疾患 C	疾患 D	疾患 E
臨床所見	診断時の症状	全身症状	発熱	見出し1 見出し2 見出し3 列挙型	単一選択		あり	なし			○	○	○	○	○
			体重減少	列挙型	単一選択		あり	なし			○		○	○	
		皮膚症状	結節性紅斑	見出し3 列挙型	単一選択		あり	なし			○	○	○		○
			潰瘍	列挙型	単一選択		あり	なし			○	○			○
			レイノー現象	列挙型	単一選択		あり	なし			○	○			
		合併症	合併症	見出し2 列挙型	複数選択		大動脈瘤	慢性呼吸不全	重症感染症	気道狭窄	○	○			○
検査所見	血液検査	白血球数	見出し1 見出し2	見出し1 見出し2		/μl					○	○	○	○	○
		血小板数	見出し2	見出し2		× 10 <sup>4</sup> /μl					○	○	○	○	○
		抗核抗体	見出し2	見出し2		倍					○	○	○	○	○
	組織検査	筋生検	見出し2 列挙型	見出し2 列挙型	単一選択		実施	未実施			○			○	
		皮膚生検	詳細 列挙型	文字型 列挙型	単一選択		実施	未実施			○	○		○	
			詳細 文字型	文字型							○				

図 7: 階層構造をデータ整理表に整理した例



## (2) データ項目バンクの作成

疾患を判断するための要素として、検査結果やさまざまな所見がある。こうした項目類は、それぞれが一意に識別されることが必要であり、疾患によって異なる名称を使用したり、同じ名称で異なる内容を使用しないようにする。そのため、すべてのデータ項目を洗い出し、バンクとして蓄積を行う。

## (3) 疾患名定義

個々の疾患について、通称名、国内正式名、英語名、ICD10 コードなど、システムで取り扱う名称を定義する必要がある。データベース上では、複数の名称を持っても支障はないが、ある検索キーとして任意の疾患名を選択した場合に、確実に一つの疾患を特定できなければならない。

## (4) データ名称定義

システムで取り扱うデータ項目の名称を定義する。疾患名同様に、複数の名称を持つことができるが、特に混同しないように配慮する必要がある。国際的な標準、または事実上の標準が存在する場合は、主なキーとして利用すべきである。

## (5) モジュール構成の設計

内容の塊(モジュール)としての構造は、併記したり、包含したりする関係になる。書籍を例にとれば、「目次」「本文」「あとがき」は個別の項目だが、「本文」の中には、「はじめに」「序章」「プロローグ」「第1章」「エピローグ」などの構造が含まれる。

データ構造で取り扱う場合は、包含関係がより複雑になり、同一項目が複数のモジュールに属している場合がある。他のモジュールに含まれることによって、値が変わる場合と、他のモジュールに含まれても値が変わらない場合とがあり、注意が必要である。例えば、書籍の「第1部第1章」の「第1章」と「第2部第1章」の「第1章」は、「その項目の中での最初の部分」という働きとしては同じだが、含まれる内容が異なる。この場合には、モジュールの階層ごとに取り扱えるようにする必要がある。

希少・難治性疾患の疾病データ項目においては、データ整理表にて階層構造で記述した項目名がモジュールの単位となる。

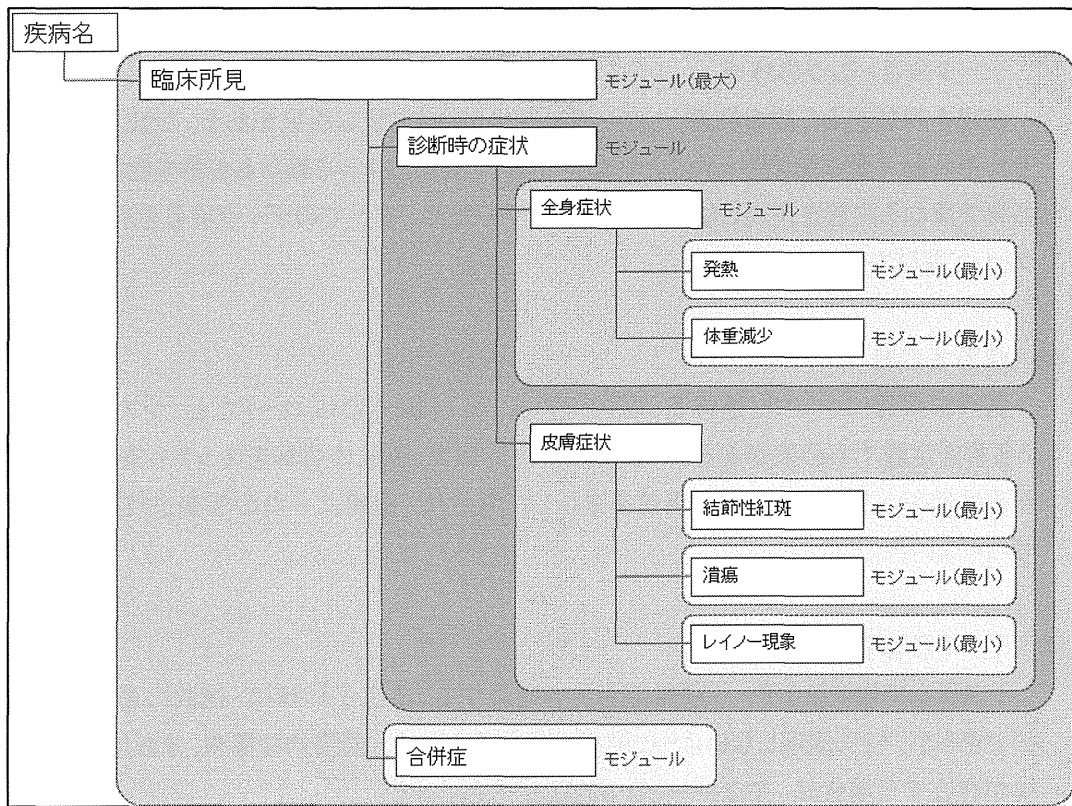


図 8: 疾病データ項目の階層構造をモジュール化

第1階層	第2階層	第3階層	第4階層	データ型	単一選択 /複数選択	単位	選択肢1	選択肢2	選択肢3	選択肢4	疾患 A	疾患 B	疾患 C	疾患 D	疾患 E
検査所見	画像検査	CT所見	見出し1	見出し2							○	○	○	○	○
			列挙型	単一選択	実施	未実施					○	○	○	○	○
	MRI所見	詳細	文字型	列挙型	単一選択	実施	未実施				○	○		○	○
		詳細	文字型	列挙型	単一選択	実施	未実施				○	○		○	○
血液検査	白血球数 クレアチニン 抗核抗体	詳細	見出し1	見出し2							○	○	○	○	○
			数値型	数値型	/μl						○	○	○	○	○
			数値型	数値型	mg/dL						○	○		○	○
			数値型	数値型	倍						○	○		○	○
重症度	重症度分類	○○分類	見出し1	見出し2							○	○	○	○	○
			列挙型	単一選択	I度	II度	III度				○			○	○
	重症度判定	クレアチニン 抗核抗体	見出し3	列挙型	単一選択	上昇	なし				○	○			○
			列挙型	単一選択	陽性	陰性					○	○			○

図 9: 同一項目が複数のモジュールに属しているデータ整理表の例

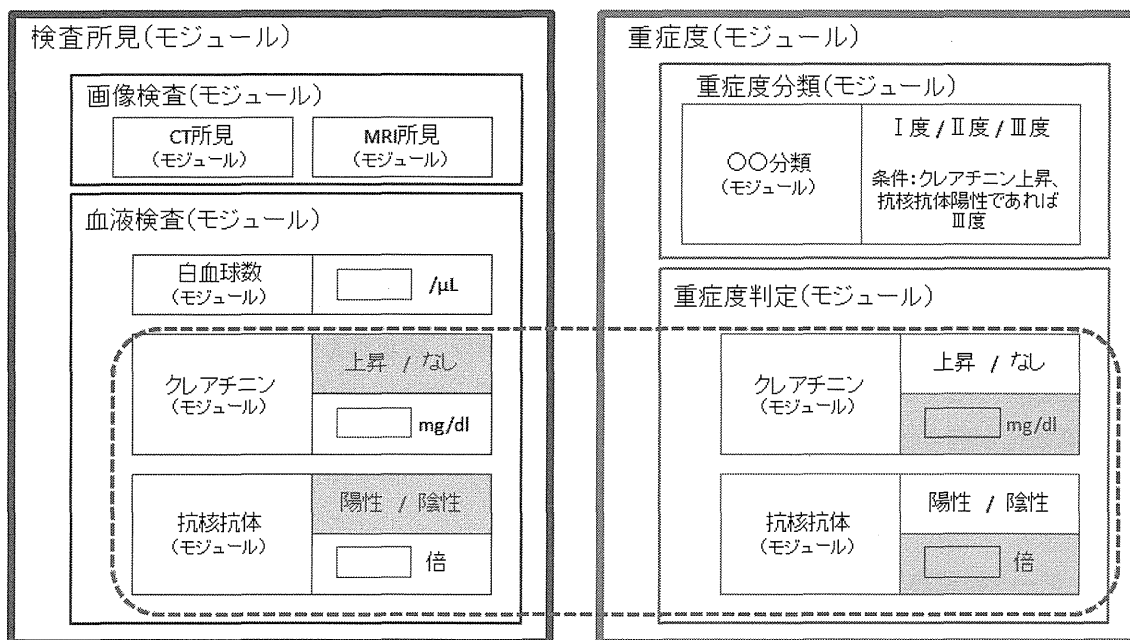


図 10: 同一項目が複数のモジュールに属している例

最大モジュール: 検査所見、重症度

2 番目に大きいモジュール: 画像検査、血液検査、重症度分類、重症度判定

最小モジュール: 白血球数、クレアチニン、抗核抗体

うち、クレアチニンは「上昇/なし」の選択肢および検査数値の属性を持つ。

また、抗核抗体は「陽性/陰性」の選択肢および検査数値の属性を持つ。

血液検査においては、クレアチニン・抗核抗体は検査数値の属性を使用する。

重症度判定においては、選択肢の属性を使用する。

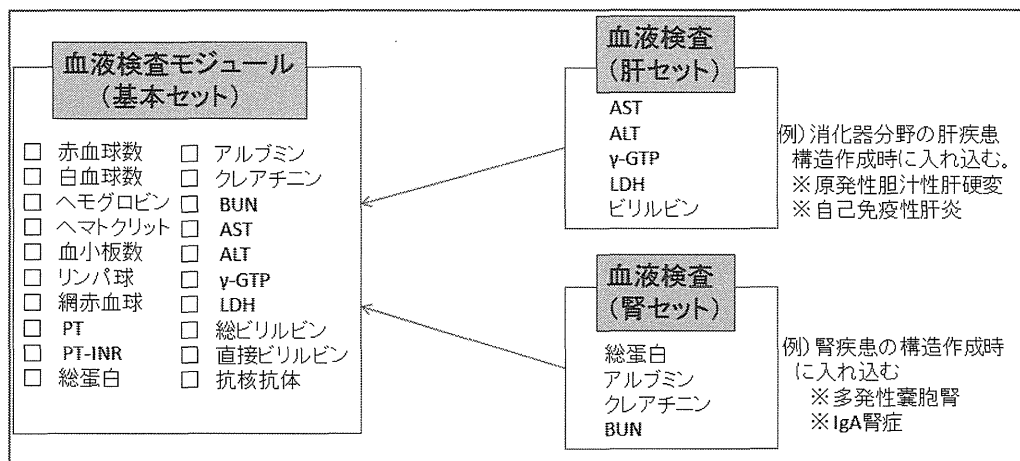


図 11: モジュールの組み合わせの例

血液検査は疾患によって、ある程度使用する代表的な項目が決まっている。

使用する項目をすべて選ばなくても代表的な項目を集めたセット(モジュール)を作成しておき、対象疾患の構造を組みあげることが必要である。

#### (6) 疾患ごとのデータ項目選択

疾患に関するパーツがそろったところで、疾患個々の構造を作成する。どのようなモジュールの組み合わせなのか、そのモジュールにはどのようなデータ項目を含むのかを選択する。データ項目整理表は、疾患ごとにこれらの構造や構成、含まれるデータ項目を整理した表である。モジュールの集積・組み合わせで整理されることにより、全疾患におけるデータ項目の標準化が図られる。

#### 4.3. データ整理表に対する外部意見の集約

各疾病のデータ項目は、対象疾患に関わる学会や厚生労働省研究班等から寄せられる素案や意見をもとに作成される。学会や研究班の意見の取りまとめにあたっては、モジュールの組み合わせで項目が作られることを周知し、どのようなモジュールがあるのかを明示した上で意見を求める。

## 5. 外部機能とのインターフェース

### 5.1. 疾患診断ロジック設定

疾患診断ロジックとは、疾患の構成要素の各値について、特定の項目を特定の順番で評価することで、患者の状態がその疾患に含まれるかどうかを評価するための手順である。これは登録システムとは異なった機能(外部機能)だが、登録システムで管理する疾病データベース及び疾患データベースを使用するため、API を定義し、外部機能からのアクセスが可能なように設計する必要がある。

### 5.2. 重症度診断ロジック設定

重症度診断ロジックとは、疾患診断で使われた個々の数値を特定の順番で参照し、評価することで、患者の容体の重篤さを評価するための手順である。留意点は上記と同様である。

### 5.3. 国際的な判定基準の参照

判断基準、重症度の基準は、国によって異なることが多く、一律に判断することができない。ただし、国際的なハーモナイズや研究結果の相互比較等、個別の判定基準を適用することが必要なケースが考えられる。留意点は上記と同様である。

## 6. バリデーション

### 6.1. バリデーションの種類

バリデーションとは、疾患データのデータ項目に入力された入力値がその項目の定義に対して正しいものであるか、該当する疾病定義との関係性において有効な値であるかの評価を行う仕組みを指す。バリデーション処理は、疾患データの保存・登録時に行われ、入力値の不正、疾病定義との整合性の不具合が検出された場合には、データ入力者へその旨を通知し、修正を促す。バリデーションは、大きく分けて次の 2 系統の処理に分類される。ひとつは、疾病定義に関わらず、データ項目に対して定義される、共通のバリデーション処理、もうひとつは、疾患毎で結果の異なる、疾患別バリデーションフィルタである。

### 6.2. 共通のバリデーション処理

共通のバリデーション処理は、データ項目自体の定義に対して妥当性のチェックを行う。データ項目の定義には、各データ項目のデータ型に対して、以下の例のようなバリデーションの確認項目があげられる。

#### (1) 文字型

入力された文字列のサイズの上限值が最大の桁数としてチェックされる。なお、全角・半角の区別はされず、単純に入力された文字の数によって評価される。

#### (2) 数値型

数値入力時には、半角数字のみが入力値として許され、最小値、最大値、整数部の桁数と、あれば小数点以下の桁数をチェックする。

例えば、血液分野の末梢血検査の血小板数について、単位は  $10^4/\mu\text{L}$  とされ、その場合の入力値の範囲は、最小値から最大値が、0.1~99.9 の数値と定義されている。数値と小数点以外の、+（プラス） -（マイナス）記号を含む文字の入力、0.1 より少ない数値、99.9 を上回る数値の場合と、また 0.1~99.9 の範囲内であっても、整数値の 2 桁を越えた入力、小数点以下 2 桁以上の入力値の場合は、エラーとされ、その旨を入力者へ通知する。

#### (3) 列挙型

選択項目の選択値の範囲と異なる選択がないか、入力値 1~ の選択肢との比較、それと合わせて、単一選択か、複数選択のどちらかによって選択された項目の数をチェックする。

例えば、ある疾患データにおいて、受診状況の選択肢として、単一選択が許されている場合、入力値に「入院」、「入院と通院半々」、「通院」、「往診あり」、「入院なし」、「施設入所」、「その他」という選択肢のリストが提示されたとすると、その場合の入力値は、何も選択されていないか、もしくは、この中のどれかでなければならない。更にこの場合は、単一選択である為、ひとつの状態の指定のみが可能であり、複数の入力値を選択することは出来ない。以上に反する場合には、入力値はエラーとされ、その旨を入力者へ通知する。「入院」、「入院と通院

半々」、「通院」、「往診あり」、「入通院なし」、「施設入所」のどれか 1 つを選択できない場合は、「その他」を選択し、次の入力値である「その他の内容」として、その内容が入力されるものである。

#### (4) 2 数値型、3 数値型

入力された値は、「年月」もしくは「年月日」を西暦で表し、西暦 1900 年 1 月 1 日から、2099 年 12 月 31 日までの有効な年月日であるかをチェックする。

例えば、年度については、西暦 1900 年から、2099 年のある数値であるか、月については、1 月から 12 月のある月、日付については、1 日から、入力された年度のその月の末日までの日付が入力されたかをチェックする。それ以外の入力値は、エラーとされ、その旨を入力者へ通知する。

#### (5) 郵便番号、電話番号

郵便番号と電話番号については、例外的に郵便番号型、電話番号型として入力パターンのチェックを行い、入力値に不具合がある場合は、エラーの場合は、その旨を入力者へ通知する。

また、他のデータ項目との関連性を考えた上での、該当データ項目自体の入力の要否については、共通バリデーション処理の対象とはされない。例えば、受診状況で通院以外を選択していた場合の通院回数に入力値は必要とされないが、このような複数のデータ項目間での矛盾などの関係性のチェックについては、データ入力画面側での処理で、入力が抑制されるものとし、データベースサーバでの共通バリデーション処理においては、この評価は行わないものとする。

### 6.3. 疾患別バリデーションフィルタ作成

疾患別のバリデーションフィルタとは、任意の疾病定義の入力可能な全データ項目のリストと、それらの内の必須となるデータ項目の情報からなる、疾患データ項目の最大範囲と最小範囲のフィルタである。例として、血液分野の疾病 A と疾病 B を挙げる。疾病 A では、検査所見のうち骨髄検査の情報が入力項目としてあるが、疾病 B ではその項目は入力項目とはなっていない。また、疾病 B では、疾患データ更新入力時には、患者の生活状況に関する情報が必須項目となるが、疾病 B の新規入力時および疾病 A の新規入力時、更新入力時のどちらにおいても、入力項目ではあるが、必須項目ではない。このように同じ分野の疾病においても項目の有無、あるいはその項目の必須の有無に差異がある。

また、上記の必須項目は、他の入力値との関係性を持たずに、無条件で必須となる項目であったが、他のデータ項目との関係性を持って入力の要否が決定される必須のデータ項目については、データ入力画面側での処理で、入力が抑制されるものとする。加えて、データベースサーバでの共通バリデーション処理においては、この評価は行わないものとする。例として、ある検査に対して実施／未実施が選択肢としてあり、実施を選択した場合に、その検査結果の情報が必須となることがある。

## 7. 画面作成

### 7.1. 画面制御データ作成

画面は、データベースやバリデーションを行うエンジンとの API を利用可能で、システムを効率的に動作させるとともに、変更が反映される柔軟な仕組みを持つことが要求される。

### 7.2. 疾病構造情報

データベースに登録された疾病構造情報は、画面作成に必要な疾病毎の入力可能な全データ項目のリストと、それら各データ項目が属するグループ(タブ、カテゴリ、サブカテゴリ、質問/タイトル)、入力値前テキスト、項目統合 ID、データ型、単位、入力範囲(最小値、最大値)、桁数、(列挙型の)入力値、必須項目であるかの情報を持つ。しかし、その他の画面制御に必要な表示ロジック対象と表示ロジック値、画面上で行われるべき自動計算の処理、画面上でのテーブル表示の構成などについての情報は、持ち合わせない。よって、それらの画面制御が必要な場合には、疾病毎に生成された疾患情報入力画面 HTML テンプレートにおいて、該当データ項目の入力欄に対して、この後で述べる修正を行う必要がある。

### 7.3. HTML テンプレート自動生成コマンド

HTML のテンプレートは、データベースサーバ内のコマンドを呼び出して、上記の疾病構造情報を元に、規定のフォルダ内に自動生成される。HTML テンプレート自動生成コマンドは、QM Console からデータベースサーバ内の規定のアカウントへログインして呼び出すことが出来る。

### 7.4. 表示ロジック

表示ロジック対象と表示ロジック値を踏まえた、表示ロジックの対照となるデータ項目には、他のデータ項目との関連性を考慮する必要がある、共通バリデーション処理の対象とはならないデータ項目がある。それら入力画面上での表示、非表示動作の制御が必要となるデータ項目は、HTML テンプレートへの修正が必要となる。HTML テンプレートへの修正は、AngularJS の仕様に基づき、HTML タグに、ng-if 属性を付加して、以下の例のように対応することが可能である。

### 7.5. 画面内計算値の反映

他のデータ項目の入力値を踏まえて、その計算結果を、任意データ項目へ反映させたい場合などの画面上で行われるべき自動計算の処理には、AngularJS の仕様に基づき、HTML タグに、ng-change 属性を付加して、以下の例の様に自動計算式を作成して対応することが可能である。

### 7.6. HTML テンプレートの保管場所

上記必要な修正を行った後、作成された HTML テンプレートは、データベースサーバ内の規定のフォルダ内に保管することによって、疾患データ入力画面のテンプレートとされる。



## 8. システム登録

### 8.1. 構造データの疾病データベースへの登録

希少・難治性疾患登録データベースは、マルチバリューデータベースの利用を想定している。マルチバリューデータベースは、インターネット型の検索サイトや通販サイトなどで利用されるデータベースであり、日々膨大なデータが蓄積されながら、同時にハードワークで利用されることを念頭に置いた、ビッグデータ型のデータベースである。データ項目整理表で整理された構造、構成について、マルチバリューデータベースで扱うことができるように変換し、疾患構造を管理できるようデータベース化する必要がある。

## 9. 運用試験

### 9.1. テストベッドの運用

希少・難治性疾患では、従来からデータ項目整理表を用いて疾患の管理を行ってきた。疾患データ登録テストベッドは、マルチバリューデータベースに加え、フィルタリングのエンジン、画面制御のAPIを実装した疾患登録システムの試験基盤である。新たな構造や構成は、このテストベッドによって互換性や動作状況を確認し、実運用システムへの影響を低減させながら実装させることができる。また、同様の考え方で開発された疾患登録事業、データ管理事業の用に供すことを想定している。

## 10. 疾患構造管理エディタの利用

本書の第 4 項の作業は、表形式ソフトを用いて手作業で作成することを念頭に置いているが、作業をシステム化することが可能である。SGML などの文書エディタに近いシステムを用いることで、構造や構成を定義し、データ項目などのパーツを設定することができる。また、文書内容を編集する機能を用いることで、疾患ごとの構造やモジュール構成を作成することができる。また、手作業で作成していたデータ項目整理表を出力することも可能となる。こうして作成された構造情報を疾病データベースに書き込むことで、登録内容の更新や変更に対応することが可能である。また、疾患データベースに蓄積された個々のデータについて、様々な要求にこたえて編集加工することが可能となる。

疾患登録管理構築ガイドライン 本編

2014年12月20日 初版 第1版発行

厚生労働科学研究費補助金 難治性疾患等克服研究事業（難治性疾患等政策研究事業  
（難治性疾患政策研究事業））「今後の難病対策のあり方に関する研究（研究代表者曾根  
智史）」研究班