

管理栄養士国家試験対策 Web サイト構築に関する研究

上 田 秀 樹
木 村 雅 浩
村 上 ゆ き

I. はじめに

本学では、平成14年度から管理栄養士養成施設として、厚生労働省から認可を受けて、食物栄養学科では、その教育・養成のカリキュラムに沿った教育を行っている。殊に、平成17年度からは管理栄養士国家試験の内容が、新しい管理栄養士養成カリキュラムに沿った内容となり、従来に比べて、レベルの高い国家試験になると予想されている。また、今後、管理栄養士養成施設は増加することが予想されており、管理栄養士養成施設の外部評価として、国家試験の合格率が重要視されることが考えられる。以上の観点から、食物栄養学科では、国家試験対策に特化した科目を設け、合格率の向上を図っている。このような状況から、管理栄養士国家試験対策の一環として、各学生の学力や理解力に応じた学習システムの構築が必要であると考えられる。しかし、現在公開されている栄養関連システムの中で、国家試験対策の学習システムとして利用し得るものは見当たらないのが現状である。そこで、本研究では、本学の学内ネットワーク機器を活用した管理栄養士国家試験対策の学習システムを構築することを目的とする。

II. 研究方法

本研究では、過去に実施された国家試験問題について、それぞれの設問毎に年度、科目、難易度などの属性を基にデータベース化を行い、Web システムを構築する。

Web システム構築に当たっては、その動作要件について基本ソフトウェア (OS: Operating System) に依存せず、汎用性の高い IE (Internet Explorer) をクライアント側のフロントエンドソフトウェアとした。また、開発環境は Microsoft が提唱する .NET テクノロジーを前提とした。

Windows プラットフォームで動作するアプリケーションは開発言語の違いはあるが、ネイティブコード (マシン語) によりコード化されていたが、.NET では、アプリケーションの実行ファイルはマネージコード (Intermediate Language) である中間言語によって記述され、実行段階で .NET Framework の共通言語ランタイムによってネイティブ化される。このことにより、Windows 系のプラットフォームから開放され、アプリケーションの可搬性はよくなる。

さらに、Microsoft.NET では、図 1 のとおり構成する要素を 5 分野としており、それらの要素が本研究における主要な要素として機能することが、システム構築において重要であると考えられる。

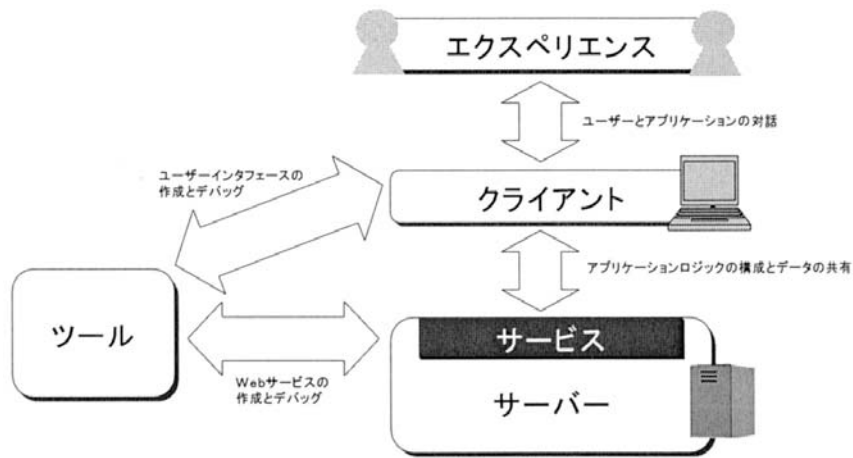


図1 .NET テクノロジーのアプリケーション開発

① エクスペリエンス

ユーザーがWebクライアントと対話する分野。Microsoft.NETでは対話することによって得られるものを重要視している。すなわち、アプリケーションソフトウェアを操作することは「ユーザーが望むことを実現したい」ための操作であって、ユーザーとアプリケーションソフトウェアの接点であるユーザーインターフェースは自ずと操作性に優れたものでなければならない。

本研究では、ユーザーが国家試験問題に集中できる操作環境を最優先課題とした。

② クライアント（コンピューター端末機）

ユーザーからすると「望むこと」が「いつでも」・「どこでも」叶えられれば、すぐれたコンピュータシステムになる。従来は、OSの違い、データ形式の違いが障害となっていた。

当面は、構築したWebシステムは学内ネットワークでのサービスとしているが、将来的には、学生が学外から本Webシステムにアクセスすることを想定しているため、クライアントによる動作環境の違いを吸収できるシステム構築が必須である。

③ サービス

ユーザーがクライアントを介して、「望むこと」を要求し、その要求を実現するのがサービスである。Webシステムの場合、サービスとはWebサービスであり、Webサーバーの機能と一体となってユーザーに提供される。Webサービスはサーバー及びそのほかのWebサービスと通信することで、アプリケーションロジックを構成する。

④ サーバー

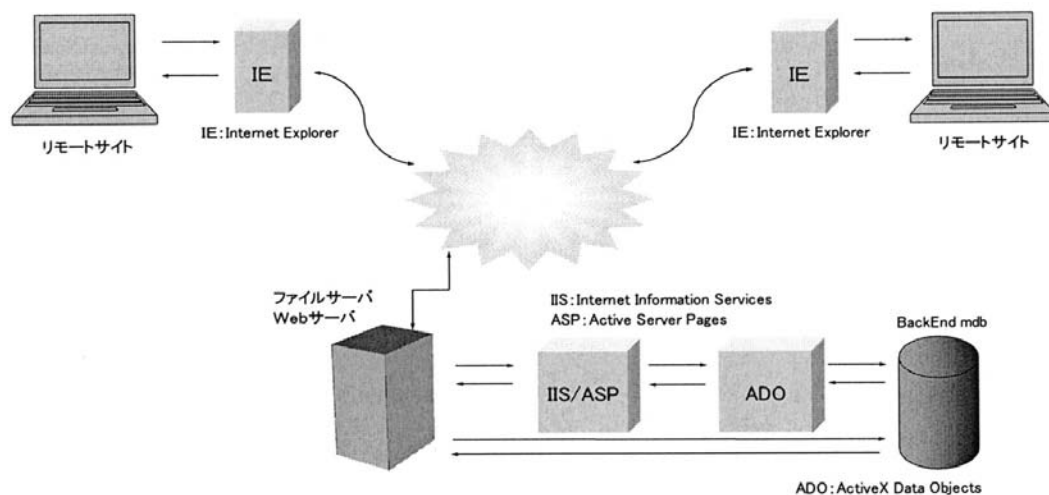
Web・ファイルサーバーのOS上で動作するアプリケーション。また、SQL Serverなどのミドルウェアを含む。本研究では、構築したデータベースはSQL Serverではなく、Microsoft Access 2003のJetエンジンをフロントエンドとした。

⑤ ツール

本研究の開発ツールはMicrosoft VisualStudio.NETである。Microsoft.NETテクノロジーに

よる Web 対応アプリケーションはそのプラットフォームである .NET Framework ライブラリーを利用することになる。VisualStudio.NET は GUI ベースの Web アプリケーション開発環境として優れている。更に、近年、.NET Framework 対応の Web フォーム用コンポーネント及びデータアクセスコンポーネントがサードパーティから多く市販され、Web アプリケーションの開発効率の向上が期待できる。

以上の 5 分野に順ずる Web システムとして、図 2 のとおり概要を示す。



1. 機材

1) 機器

今回は、サーバーマシンに開発環境を構築した。ハードウェアの必須要件は .NET テクノロジーの開発ソフトウェアである Visual Studio.NET の必須要件に準じた。

表 1 にハードウェア構成を示す。また、本 Web システム開発に用いたソフトウェアを表 2 に示す。

表 1 ハードウェア構成

項 目	内 容	容 量
CPU	Xeon プロセッサ 2.80MHz	デュアル構成
主メモリー	デュアルチャンネル DDR2-SDRAM	1 Gbyte
外部メモリー	SATA-HDD	70GB
ディスクシステム	RAID システム（ミラーリング構成）	
ディスク装置	CD - ROM	
	FDD	1 台
ディスプレイ	ハイカラー表示	1 台
ポインティングデバイス	Mouse	

表2 ソフトウェア構成

製品名	エディション
日本語OS	Microsoft Windows 2000 Server Service Pack 4 適用
Visual Studio.NET	Enterprise Architect MSDN Deluxe _Edition
.NET Framework SDK	Version 1.1
IIS	Internet information Server 5.0 ホットフィックス KB891781 適用
Web ブラウザ	Microsoft Internet Explorer 6.02 Service Pack 1 適用
MDAC	Microsoft Data Access Components 2.6 and 2.7
NETADVANTAGE 2004	Version 1.0
TRUEWEBGRID	for ASP.NET
SPREAD	for ASP.NET
VBReport.NET	Version 5.0

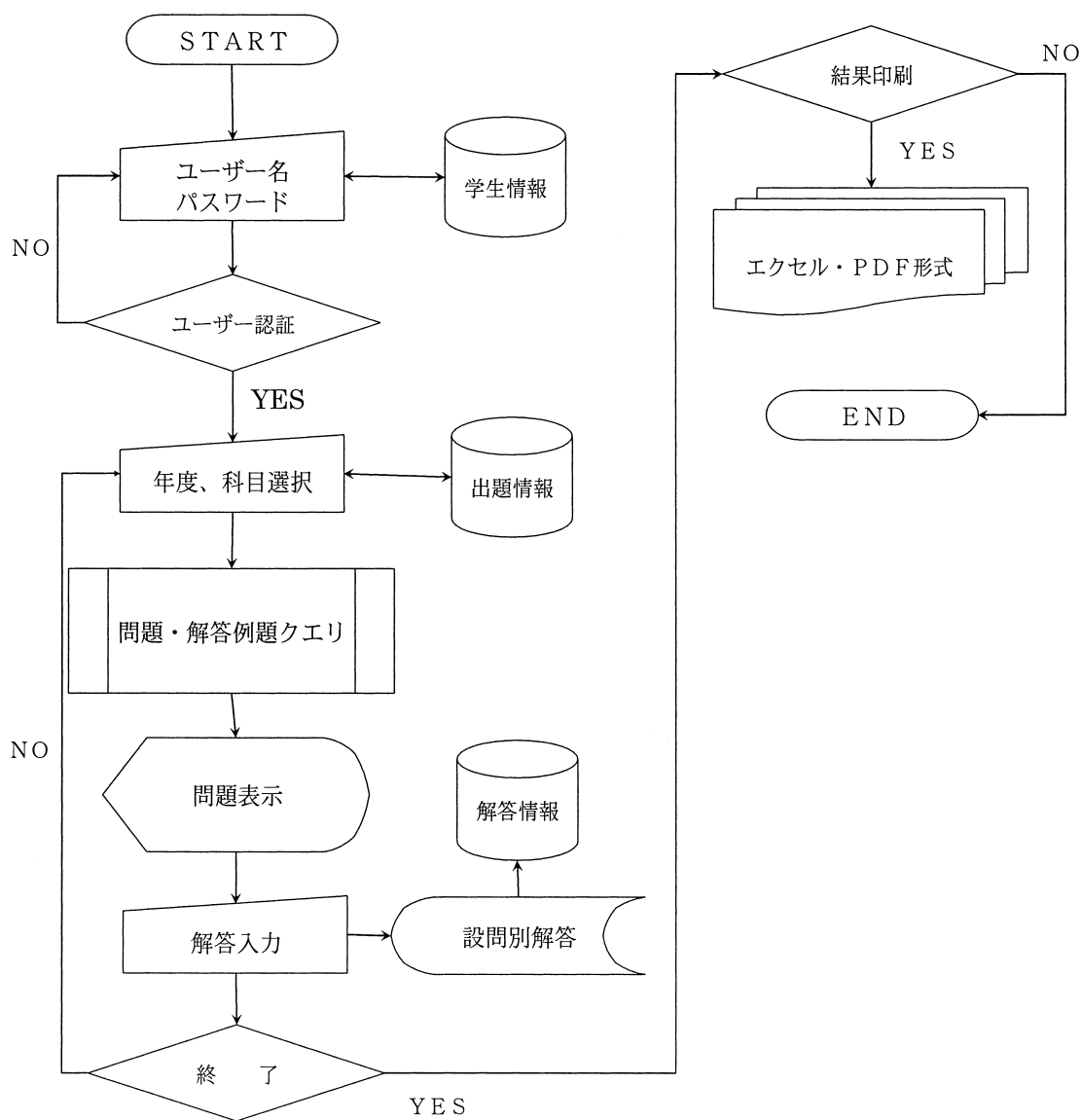


図3 Web システムの処理フローチャート

図 3 は Web システムの処理フローチャートである。データベースは主要なデータを学生情報・出題情報・解答情報の 3 要素に分類しデータベース設計を行った。学生個別の問題解答情報は個人情報となるので、本 Web システムにおけるセキュリティを高めるため、利用者のユーザー認証を必須とした。また、問題出題方法及び画面表示については、本研究における主要課題として検討した。また、結果出力は結果のプレビュー機能はもちろん多様な印刷形式に対応した。

III. 結 果

1. データベースの設計

データベースは Microsoft Access 2003 を用いて、テーブル、クエリを作成した。

1) テーブル項目

(1) 科目マスタ

科目 ID、科目名をフィールドに持つ。

(2) 年度マスタ

ID、NAME をフィールドに持つ。

このマスタは、過去の国家試験問題の識別子として、また、学内模擬試験の識別子として識別する属性を付加させる。

(3) 設問分類マスタ

問題 ID、大項目 ID、中項目 ID をフィールドに持つ。

このマスタにより、過去の試験問題を管理栄養士国家試験出題基準 1) に従って分類することができる。

問題 ID は、本データベースの中核的キーになっている。

キー構造は次のとおりである。

全 6 桁のうち、上位 3 桁は国家試験実施回数を保持しており、下位 3 桁はその実施年度の問題一連コードを保持している。なお、学内模擬試験は 3 桁で始まるコードを適用する。

(4) 大項目マスタ

科目 ID、大項目 ID、大項目名をフィールドに持つ。

このマスタにより、大項目と科目を対応づけることができた。

(5) 中項目マスタ

大項目 ID、中項目 ID、中項目名をフィールドに持つ。

このマスタにより、中項目と大項目を対応づけることができた。

(6) 学生マスタ

ID、NAME、PASSWORD をフィールドに持つ。

(7) 学生別解答マスタ

一連 CD、学生番号、回数、設問連番号、正誤をフィールドに持つ。

このマスタに、解答履歴を保持する。解答の積算を行い、成績評価の基本となるマスタ。

(8) 問題マスタ

設問 CD、科目 ID、設問例示番号、設問文章、回答肢数、解答タイプ、解答例示、解答をフィールドに持つ。

(9) 解答例マスタ

一連 CD、設問 CD、解答例示番号、解答例文、正誤をフィールドに持つ。

2) クエリの設定

テーブル間でのリレーションシップが必要な場合は、それぞれの ASP (Active Server Page) 内の PageLoad イベントハンドラで記述できるが、今回は Access 側で予め作成した。

(1) 設問表示クエリ

マスタテーブルのうち、問題マスタと解答例マスタのそれぞれのテーブルから、次のフィールドを選び、設問 CD フィールドを識別フィールドとしてクエリを作成した。

設問 CD、科目 ID、設問例示番号、設問文章、一連 CD、解答例示番号、解答例文、正誤

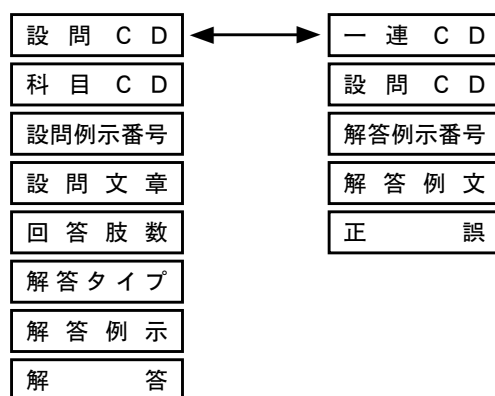


図 4 設問表示クエリ

2. サーバー及び開発環境の設定

1) Web サーバーの設定

Web サーバーの設定は Windows 2000 Server のコントロールパネルからアプリケーションの追加と削除から、Windows のコンポーネントの追加と削除を選択してインターネットインフォメーションサーバー (IIS) をインストールする。コンポーネントインストールの後、管理ツールにある IIS を起動する。既定の WEB サイトから仮想ディレクトリの新規作成を行い公開サイトの作成を行った。以上の公開サイト作成方法は後述する VisualStudio.NET による「新しいプロジェクト作成」からも実行可能である。

2) ファイルサーバーの設定

本研究では、データベースのフロントエンドとして Jet エンジンを使用するために、Web ページ上でデータベースアクセスをする場合には次の 2 つの方法がある。

a. VisualStudio.NET プロジェクトの Web.config 内でデータベース接続を記述する方法

※ 記述コード

```
<appSettings>
  <add key="conStringAccNw"
    value="PROVIDER=Microsoft.Jet.OLEDB.4.0;DATAsource=e:\¥koku01¥kokusidb.mdb" />
</appSettings>
```

b. ASP (Active Server Page) の PageLoad イベントでデータベース接続を記述する方法

※ 記述コード

```
If Session("fpdbpath")="" Then
  Dim dbpath As String
  dbpath = HttpContext.Current.Server.MapPath(Request.ApplicationPath + "/" )
  dbpath += "..¥koku02¥db¥kokusiDB.mdb"
  Session("fpdbpath")="Provider=Microsoft.Jet.OLEDB.4.0;Password=;User_
  ID=Admin;Data Source=" & dbpath & ""
End If
```

今回は、ASP の PageLoad イベントハンドラにコードを記述した。

3. プロジェクトの設定

IIS による Web サイトの設定ができていれば、VisualStudio.NET から「新しいプロジェクト作成」で簡単にプロジェクトの作成が行える。

1) Web フォームの作成

(1) ログインとユーザー認証

本システムは個人情報を取り扱うので、ユーザーを認証する必要がある。

本報では、ログインページに Web コントロールを追加して、PageLoad イベントハンドラにコードを記述し、XML ファイルをデータフォーマットとして利用した。

下記に、XML ファイル内のコードを記述する。

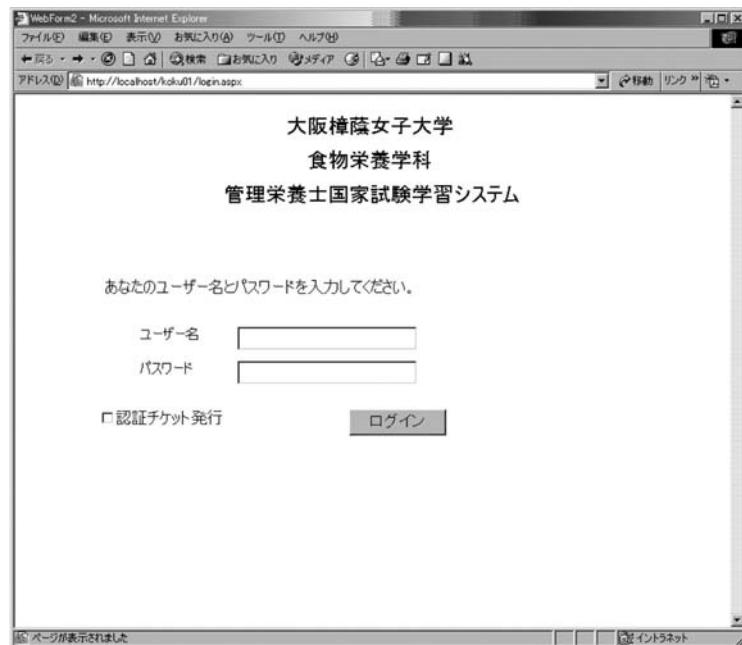


図5 ログイン画面

※ 記述コード

```
<?xml version="1.0" encoding="utf-8" ?>

<users>

<user name="〇〇" password="XX">

<user name="〇〇" password="XX">

</users>
```

なお、データの読み込みには DataSet オブジェクトを使って、ユーザー情報のデータベーステーブルとして扱うことが出来るようにした。

(2) 年度別管理栄養士試験問題の表示

本システムで最も重要な部分となる。ユーザーである学生がストレスなく、試験問題に取り組めるユーザーインターフェースを検討した。

まず、Web でデータベース接続を行う場合、前述の Web.config ファイル内で、データファイルを設定する方法以外には次の手法がある。

VisualStudio.NET の開発環境では標準で Web フォームコントロールに Repeater コントロールがあり、その HTML ソース内に DataSource として目的のデータベースを指定する。データの Web フォームへの表示はテンプレートタグを用いることで複雑なデータ表示が可能になる。

しかし、本研究で扱うデータリストは問題番号及び文章と解答例文章が階層化しており、一般的なテンプレートタグでは表示の制御には限界があった。図 6 では、その実行例を示している。国家試験問題文章の左端の＋マークもしくは、ブックのアイコンをクリックすると図 7 のとおり、解答例文章が表示される Web フォームを作成したが、国家試験の試験問題形式とは異なっていたため、本リスト表示形式は用いなかった。

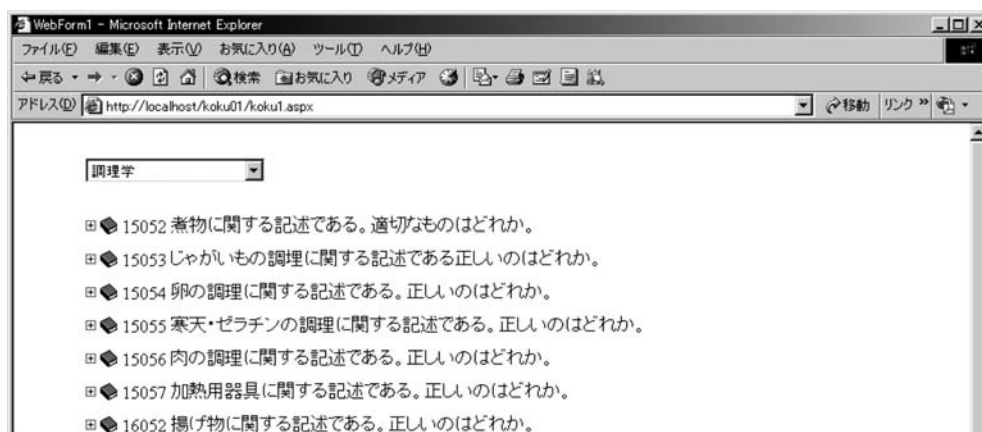


図 6 Repeater コントロールを使った問題リストの表示（問題文章のみ）

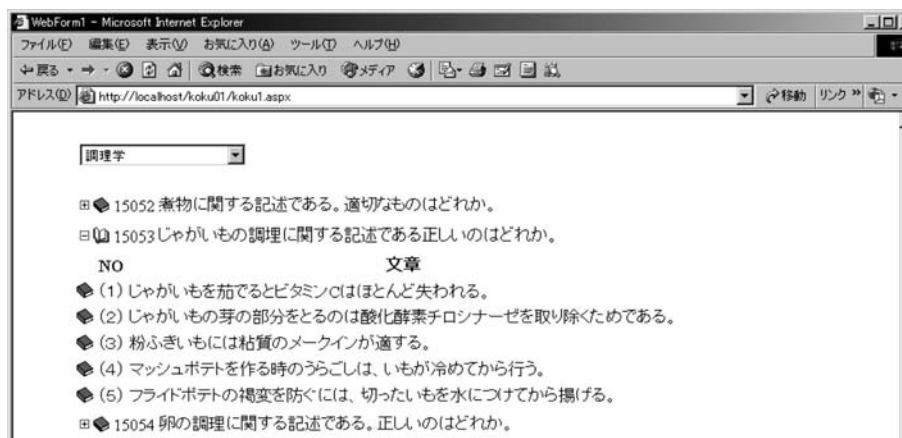


図7 Repeater コントロールを使った問題リストの表示（問題文章と解答例題文章）

本研究の重要な目的の1つである優れたユーザーインターフェースを作成するために、階層化されたデータリストを Web ページで容易に表示できる .NET Framework 対応の Web コンポーネントとして、SPREAD.NET（グレースシティ社製）を用いることにした。

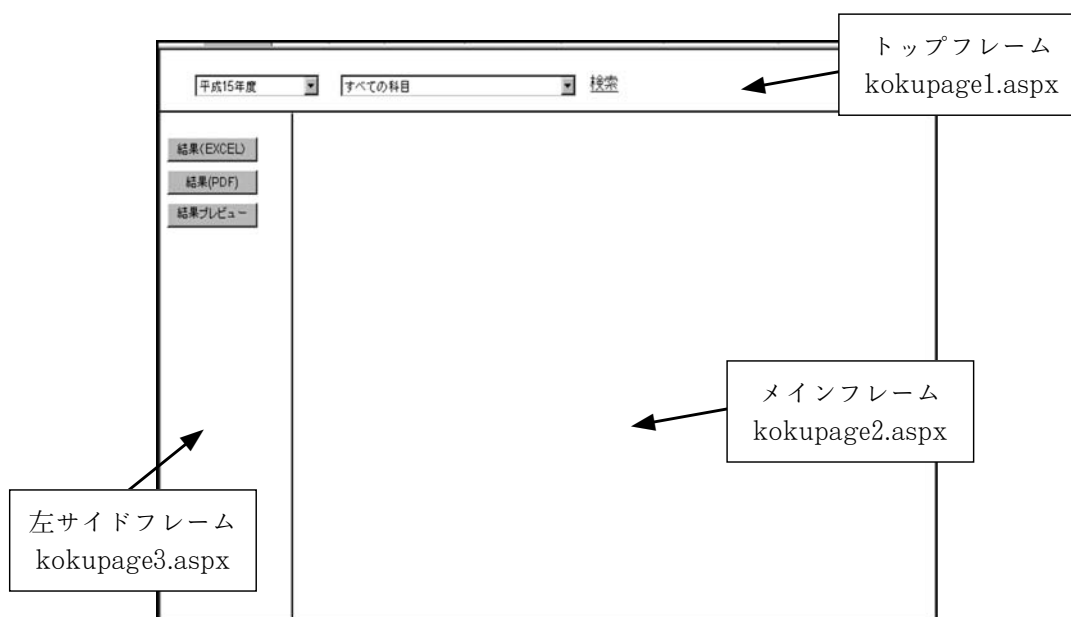


図8 問題表示 Web ページのフレーム構造

問題表示ページは図8のとおり HTML の3フレーム構造とした。

a. 年度及び科目による検索処理

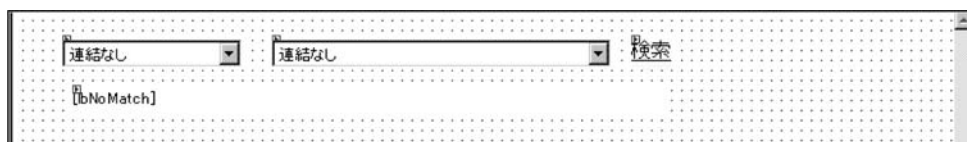


図9 3フレームのうちの TOP に配置される ASP ページ

図9はTOPフレームのASPページである。2つのリストボックスのうちの左側は過去の国家試験問題を年度で検索するためのキーワードを表示するものであり、年度マスタのデータをこのコンボボックスに item として読み込むように処理した。また、右側のコンボボックスは科目マスタのデータをコンボボックスに item として読み込むように処理した。これらの処理コードは本ページの PageLoad イベントハンドラで実行するようにした。以下に処理コードを記述する。

※ 記述コード

```
Private Sub Page_Load (ByVal sender As System.Object, ByVal e As System.EventArgs)
```

```
Handles MyBase.Load
```

```
(コード省略)
```

```
If Not IsPostBack Then
```

```
    nenCombo() ←※1
```

```
    groupCombo()←※2
```

```
End If
```

```
End Sub
```

※1 年度マスタからリストボックスにデータを読み込むサブルーチン

```
Sub nenCombo ()
```

```
    sqlstr = "SELECT id, nen from 年度マスタ "
```

```
    OleConnection1.ConnectionString = sqlConString
```

```
    OleConnection1.Open()
```

```
    Dim objCommand As New OleDb.OleDbCommand(sqlstr, OleConnection1)
```

```
    Dim objDataReader As OleDb.OleDbDataReader
```

```
    objDataReader = objCommand.ExecuteReader
```

```
    nenList.DataTextField = "nen"
```

```
    nenList.DataValueField = "id"
```

```
    nenList.DataSource = objDataReader
```

```
    nenList.DataBind()
```

```
    'nenList.SelectedIndex = nenList.Items.Count-1
```

```
    OleConnection1.Close()
```

```
End Sub
```

※2 科目マスタからリストボックスにデータを読み込むサブルーチンは省略

2つのコンボボックスの右側には、検索実行のラベルコントロールを配置した。このラベルコントロールには、HTML でメインフレームに kokupage2.aspx をリンク設定しているので、ラベルをクリックすることで、本ラベルにリンクされた kokupage2.aspx に表示ページが遷移する。

なお、kokupage1.aspx の年度コンボボックスから値を選択すると“NENDO” Session に値が保存される。同様に、科目コンボボックスからは“KA” Session に値が保存される。いずれ

も、設問表示クエリの検索文字列のキーとして用いられるようにした。

b. 問題及び解答例文章と解答フォームの表示

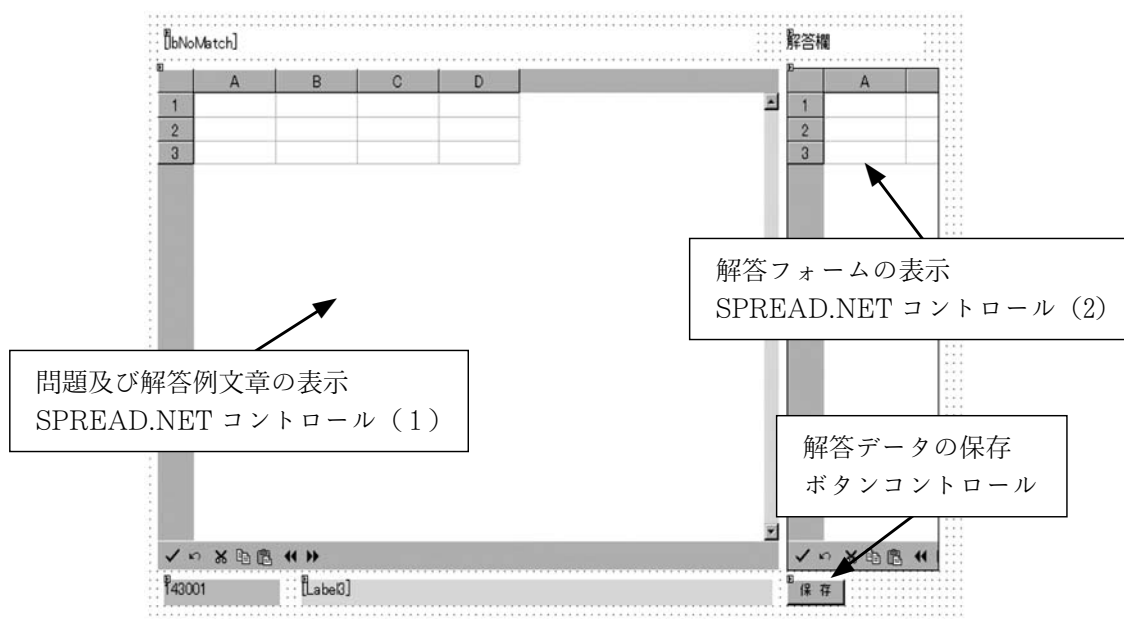


図 10 kokupage2.aspx のコントロール配置

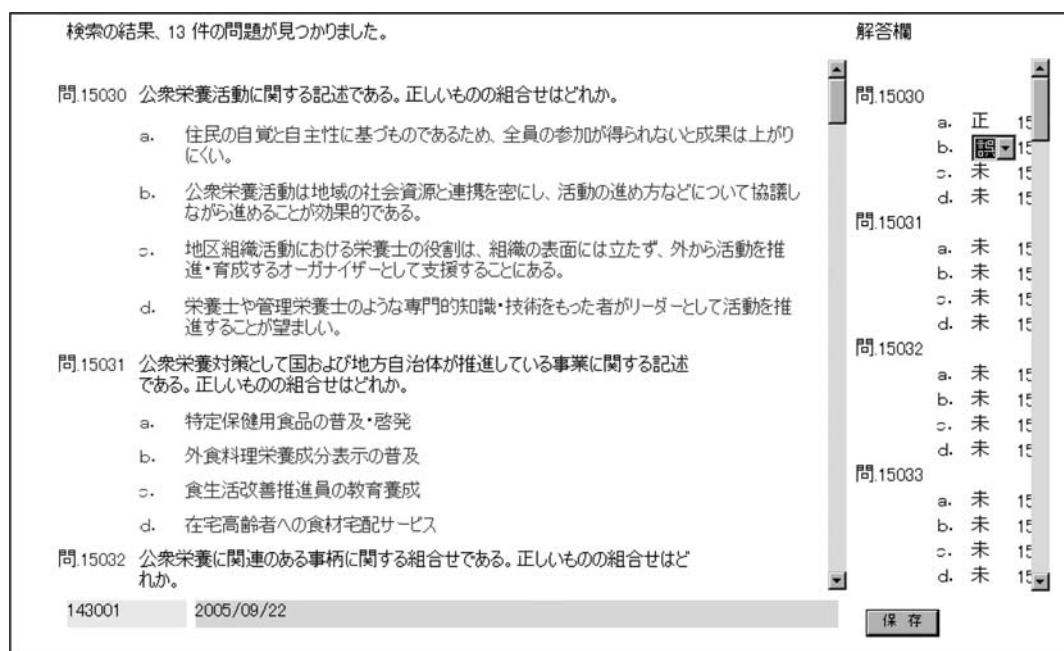


図 11 kokupage2.aspx のコントロール配置とデータ表示例

SPREAD.NET コントロールを使用することにより、図 6 及び 7 の問題文章及び解答例文章の表示に比べて、非常に見やすくなった。特に、問題文章や解答例文章の文字数によっては 1 行や 2～3 行の複数行の対応には、FpSpread1.Sheets (0) .SetRowHeight (row, XX) の XX の部分に整数を設定することで、セルの高さが設定できる。

また、セルの文字色については、FpSpread1.ActiveSheetView.SetStyleInfo (row, col - 1, style)

の style の値に、フォント・文字サイズ・背景色・前景色を設定する。以下に実際のコードを記述する。

※ SPREAD オブジェクトのプロパティ設定

FpSpread1.Sheets(0).SpanModel.Add(row, 1, 1, 6) ← 1 ～ 6 カラムのセルを結合する。

Select Case Len(mojisu) ←変数 mojisus の値によって、行の高さを変化させる。

Case 1 To 39

FpSpread1.Sheets(0).SetRowHeight(row, 35)

Case 40 To 79

FpSpread1.Sheets(0).SetRowHeight(row, 50)

Case 80 To 119

FpSpread1.Sheets(0).SetRowHeight(row, 65)

Case 120 To 149

FpSpread1.Sheets(0).SetRowHeight(row, 80)

Case 150 To 300

FpSpread1.Sheets(0).SetRowHeight(row, 95)

End Select

style = New StyleInfo

style.Font.Size = FontUnit.Parse(12)

style.Font.Name = "MS P ゴシック "

'style.BackColor = Color.Blue

style.ForeColor = Color.Blue

FpSpread1.ActiveSheetView.SetStyleInfo(row, col - 1, style) ←セルの文字属性の書式設定。

※ 記述コード (kokupage2.aspx の Pageload イベント内)

If Session("fpdbpath") = "" Then

Dim dbpath As String

' サーバの物理パスを取得します。

dbpath = HttpContext.Current.Server.MapPath(Request.ApplicationPath + "/")

' データベースへのファイルパスを設定します。

dbpath += "..¥koku02¥db¥kokusiDB.mdb"

Session("fpdbpath")="Provider=Microsoft.Jet.OLEDB.4.0;

Password=;User ID=Admin;Data Source=" & dbpath & ""

End If

' データベースの接続文字列をセッション値から読み込みます。

```
sqlConString = Session("fpdbpath")
```

```
Label3.Text = Calendar1.TodaysDate
```

' データを取得し、シートを作成します。

```
GetData()            ←問題文章及び解答例文章の表示サブルーチン
```

```
PutTouanData()   ←解答フォームの表示サブルーチン
```

※ 記述コード (GetData サブルーチンの一部、その他は省略)

```
sqlstr = "SELECT * FROM 設問表示クエリ "
```

```
sqlcnt = "SELECT COUNT(*) AS Expr1 FROM 設問表示クエリ "
```

```
nenid = Session("NENDO")
```

```
groupid = Session("KA")
```

```
If groupid <= 1 Then
```

```
    sqlstr += " where 年度 = " & nenid    (sqlcnt も同値)
```

```
Else
```

```
    sqlstr += " where 年度 = " & nenid & " and 科目 ID = " & groupid(sqlcnt も同値)
```

```
End If
```

```
OleConnection3.ConnectionString = sqlConString
```

```
Try
```

```
    OleConnection3.Open()
```

```
    Catch ex As OleDb.OleDbException
```

```
        lbNoMatch.Font.Bold = True
```

```
        lbNoMatch.ForeColor = Color.Red
```

```
        lbNoMatch.Text = " エラー：データベースが指定された位置に見つかりません。
```

```
        kokupage.aspx.vb(Page_Load イベント) のデータベースの場所を修正してください。"
```

```
        FpSpread1.Visible = False
```

```
    Exit Sub
```

```
End Try
```

解答フォームは問題及び解答例文章の処理とほぼ同様であるが、解答の選択肢を予めコンボボックスコントロールに値設定したところが異なる。

※ 記述コード (解答フォームのセルをコンボボックスコントロールに設定)

```
Sub MakeComboCell(ByVal row As Integer)
```

```
    Dim combo As New myComboEditor
```

```
    FpSpread2.ActiveSheetView.Cells(row, 2).Editor = combo
```

```
FpSpread2.ActiveSheetView.SetValue(row, 2, combo.Items(0))
End Sub
```

c. 解答データの取得と保存

解答フォームにあるコンボボックス値（未・正・誤）のデータは SPREAD コントロールのセル値として、次の記述コードによって取得した。

※ 記述コード（コンボボックスの値を取得）

```
curskotae = Left(FpSpread2.Sheets(0).GetText(crow, 2), 1) ←変数にセル値を取得
skotae(execount) = Left(FpSpread2.Sheets(0).GetText(crow, 2), 1)←問題番号をセル値から取得
Select Case (curskotae)
    Case "未" ←セル値の文字列によって、解答を整数に変換
        ikotae(execount) = 0
    Case "正"
        ikotae(execount) = 1
    Case "誤"
        ikotae(execount) = 2
End Select
```

また、解答データは ACCESS の次の S Q L 文によって、学生別解答テーブルに保存した。

※ 記述コード（コンボボックスの値を取得）

```
Dim sb2 As New StringBuilder("")
sb2.Append(" UPDATE 学生別解答テーブル ")
sb2.Append(" SET 学籍番号 =@ 学籍番号 , 回数 =@ 回数 , 設問連番号 =@ 設問連番号 , 正誤 =@ 正誤 ")
```

(3) 解答集計と結果出力

学生別解答テーブルに保存された解答データは図 8 の左サイドフレームの kokupage3.aspx に図 12 のとおり、配置した 3 つボタンコントロールによって、結果出力を行った。

1) エクセルファイル及び P D F ファイルへの結果出力

結果の様式は資料の結果出力例 1 のとおりに設計した。本結果様式は、図 12 のそれぞれのボタンに割り当てた。エクセル及び P D F はファイルとして任意に保存先を指定することが出来る。

また、結果プレビューは、印刷プレビューとして機能し、プレビューの画面にも保存ボタンを配置し、エクセル及び P D F ファイルのいずれかに保存することが出来るようにした。



図 12 結果出力ボタン

これらの機能は、表2のVB-REPORT.NETコントロールの機能を利用した。このコントロールの優れた点は、出力帳票を設計する時に、エクセルのワークシートを使ってレイアウトを設計することが出来る点である。

次に、VB-REPORT.NET コントロールを使って、エクセルファイルに結果を出力するコードを記述する。

※ 記述コード（ボタンコントロールの Button_Click イベントハンドラに記述）

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
```

```
Handles Button1.Click
```

```
Dim strRptMode As String ' 出力方法
```

```
Dim strURL As String ' URL
```

```
strRptMode = "1" ←エクセルファイルに結果を出力する。
```

```
strURL = "WebReportOut.aspx?&TVL=" + HttpUtility.UrlEncode("XXX") + _
```

```
"&RNO=" + HttpUtility.UrlEncode(strRptMode)
```

```
Response.Clear()
```

```
Response.Redirect(strURL, True)
```

```
Response.End()
```

```
End Sub
```

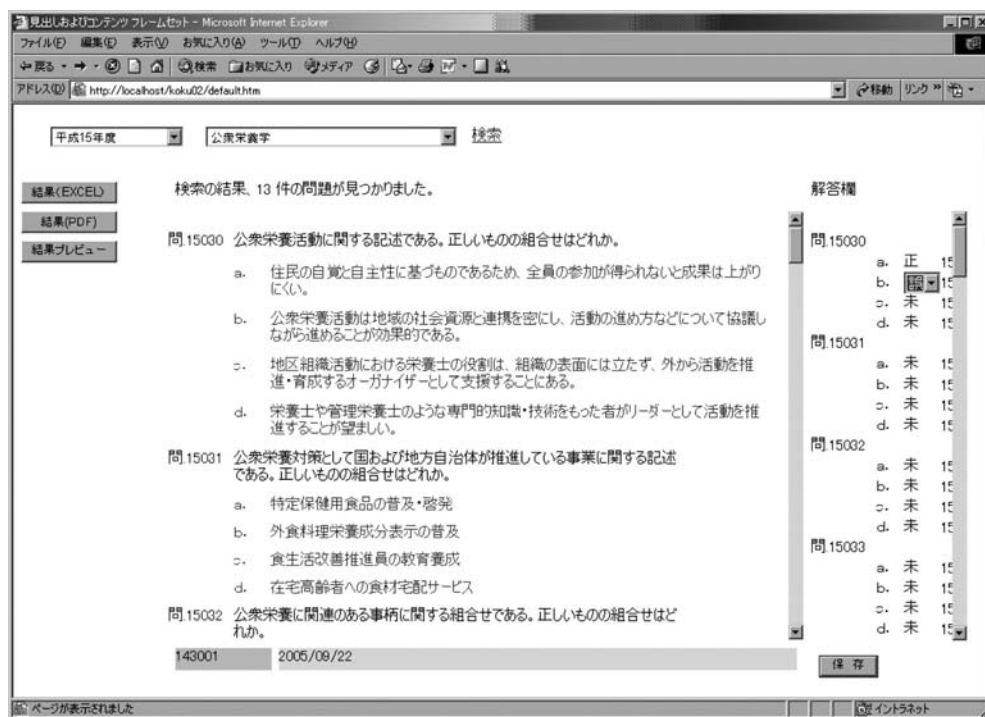


図 13 問題検索と問題文章及び解答例文章の表示と解答フォームの結合画面

図 13 は図 8 のフレーム構造の Web ページに実際のデータを表示したものである。問題番号や問題文章及び解答例文章に色設定をしたことで、見やすい画面構成になった。また、問題文章及び解答例文章の画面と解答フォーム画面はそれぞれ独立したことで、ユーザーによる解答のチェック性が向上したと考える。実際、国家試験においても、問題用紙と解答マークシートは独立しており、試験環境を Web 上でも再現できたと考える。

4. まとめ

本報では、管理栄養士国家試験対策の一環で、学生の学習を支援するツールとして学内ネットワークを利用した Web システム構築に関して、データベースの設計とユーザーインターフェースを検討した。現在の Web システム開発環境に求められるハードウェアや利用ソフトウェアについてはネットワークの規模や利用状況などによって選択肢は多岐に及ぶと考える。

国家試験の過去の問題や模擬試験問題をデータベース化することにおいてはほぼその基本形が定まったと考える。また、ユーザーインターフェースや結果出力についても一定の水準に達していると考える。

今後は、以下の事項について検討を要する。

- (1) 利用状況によるサーバーの CPU やメモリー等ハードウェアのパフォーマンスの評価
- (2) ユーザーインターフェースの評価
- (3) ASP ページ内のコードのデバッグ
- (4) 結果出力様式の検討

以上、小数のモニターからの評価を基に本システムの最適化を図ってゆきたい。

参考資料

- 1) 葛西秋雄 (2003)、Access2000/2003 による Web サーバ構築技法、技術評論社
- 2) 葛西秋雄 (2003)、Visual Studio.NET2003 VS.NET+ACCESS で始めるデータベース連動 web フォーム、ラトルズ
- 3) 有限会社ガリバー (2003)、VisualBasic.NET+ASP.NET で始める Web アプリケーション開発、ソフトバンクパブリッシング株式会社
- 4) 赤間信幸 (2004)、.NET エンタープライズ Web アプリケーション開発技術大全 Vol.3、ASP.NET 応用編、日経 B P ソフトプレス
- 5) Gary Cornell 著、アスキー書籍編集部訳 (1998)、VBScript セルフマスターブック、アスキー出版局
- 6) 生形洋一・堀田健也 (2003)、ASP.NET プログラミング本格入門、技術評論社
- 7) 宮坂雅輝 (2005)、SQL ハンドブック第 2 版、ソフトバンクパブリッシング株式会社

大阪樟蔭女子大学 管理栄養士国家試験自己学習システムによる学習結果

2005/9/30

学籍番号: XXXXXX

氏名: XXXXXXXXXX

次のとおり、学習結果をお知らせいたします。

科 目	出題数	正答数	%	マーク	評 価	キーワード
すべての科目	150	97	64.7	△		
栄養指導論	14	8	57.1	×	学習が不十分です。	
臨床栄養学	15	10	66.7	△	もう少しの学習が必要です。	
公衆栄養学	13	8	61.5	△	もう少しの学習が必要です。	
給食管理	15	11	73.3	○	継続した学習が必要です。	
公衆衛生学	13	8	61.5	△	もう少しの学習が必要です。	
食品衛生学	5	3	60.0	△	もう少しの学習が必要です。	
食品加工学	5	4	80.0	◎	出来なかった項目の学習が必要です。	
解剖生理学	10	8	80.0	◎	出来なかった項目の学習が必要です。	
病理学	5	2	40.0	×	学習が不十分です。	
生化学	12	10	83.3	◎	出来なかった項目の学習が必要です。	
栄養学	20	12	60.0	△	もう少しの学習が必要です。	
健康管理概論	5	3	60.0	△	もう少しの学習が必要です。	
食品学	18	10	55.6	×	学習が不十分です。	

