

Java フレームワークを用いた工学研究科シラバスシステム構築

○原 祐一

名古屋大学 全学技術センター 工学系技術支援室 情報通信技術系

1. 概要

名古屋大学・工学研究科で利用しているシラバス・Web システムが 10 年以上経過したこと、サーバの老朽化など様々な不具合が発生するようになった。また、名大 ID との連携など、新たに追加したい要素も増え、現在のシステムでは対応することが困難となった。そこで、サーバリプレースを機にシラバス・Web システムを再開発することとなり、Java フレームワークを用いて再開発を行ったので、この手法を紹介する。

2. シラバスシステム再開発の背景

全学共通で利用できるシラバス・Web システムは存在するが、全学のシステムでは項目が足りないなど、全学共通のシラバスシステムだけでは対応できないという現状がある。そのため、工学研究科として、独自のシラバス・Web システムを持つことで不足部分を補っている。しかし、現在のシラバス・Web システムは外部の業者が開発したシステムであり、システムのカスタマイズが容易にできない。また、サーバの老朽化が進み、Web システムへのアクセスにとっても時間がかかるようになったことから、サーバリプレースを考える必要が出てきた。そこで、今のシステムの問題点の改善や追加要望を実現する形でサーバリプレースを業者に依頼見積もりを取ったところ、システムの再開発が必要であり、予想外の開発費用が必要であることが分かった。

工学研究科・教務課（以下、教務課と省略する）より、上記のような状況であるため、工学系技術支援室・情報通信技術系で、安価にシラバスシステムの再開発およびサーバリプレースを行うことができないかと相談があり、この依頼を受けることとした。

3. シラバスシステム・要望事項

教務課より上がった主な要望は以下の通りである。

- ・現在のシステムは表示に時間がかかるため、Web 画面表示を早くしてほしい
- ・管理者・教員画面へのログインは名大 ID を利用できるようにしてほしい
- ・項目の追加と各項目の入力文字数が 254 文字までなので、入力文字数を増やしてほしい
- ・印刷をする手順が複雑なので、もっと簡単にできるようにしてほしい
- ・サーバリプレース費用は 100 万円以下としてほしい

4. 現状システムの調査と対応方法の検討

現在のシステムを改良することで要望を実現できるか、それともシステムを再開発した方が良いかを判断するために現在のシステムを調査した。現在のシステムを調査したところ、データベースの正規化が不十分であるなど、システム内部でも問題点が多数でてきたことから、システムを改良するよりも再開発した方が良いと判断し、1 からシラバスシステムを開発することとし、下記の流れでシラバスシステム開発及びサーバリプレースを行うこととした。

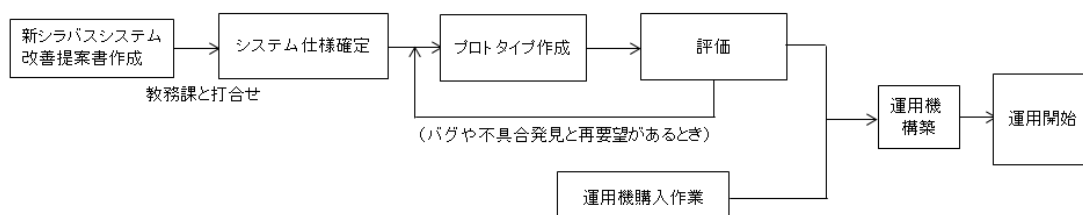


図 1 プロジェクトの流れ

5. シラバスシステム・システム構成

表 1 ユーザ別処理表

学生	<ul style="list-style-type: none"> シラバス情報を閲覧可能とする (非ログイン)
教員	<ul style="list-style-type: none"> 担当する講義情報を修正する (名大 ID でのログイン必須)
教務課	<ul style="list-style-type: none"> すべての講義情報の修正を可能とする。 新規講義の追加や新年度の切替等、シラバスシステムの運用を管理者画面より可能とする。 シラバス情報を PDF 出力可能とする。 (名大 ID でのログイン必須)

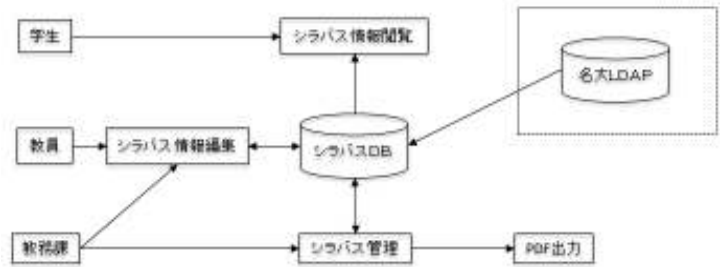


図 2 システム概念図

6. 新シラバスシステム・開発

表 2 開発情報

項目	ソフトウェア		
開発言語	Java	Spring フレームワーク	システムベース
		iText	PDF 出力
開発環境	Windows 7 + eclipse		

名大 ID 認証

図 3 のように名大 LDAP に問合せた結果情報とシラバスユーザ情報を用いて、ログイン認証を行う形をとった。

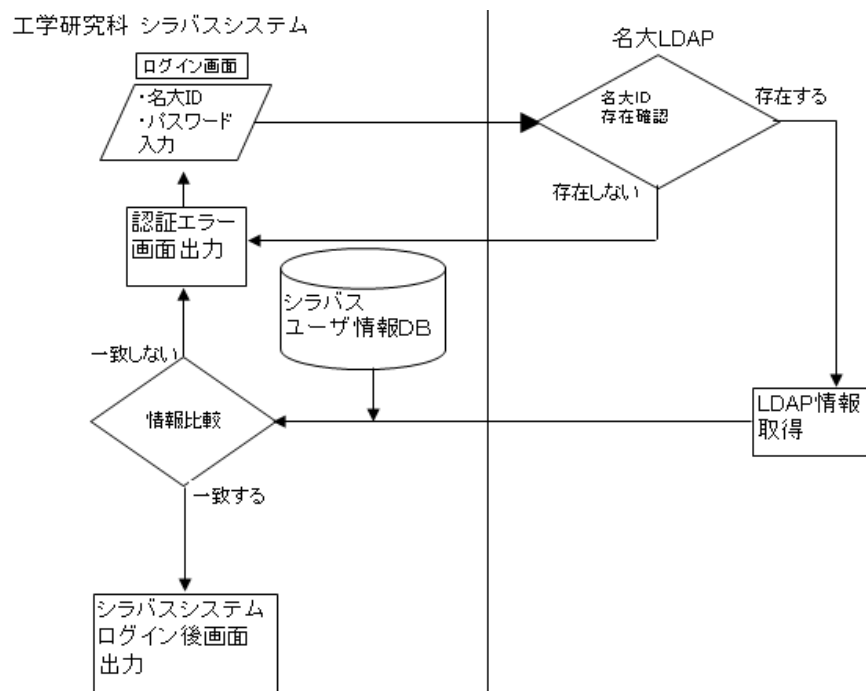


図 3 名大認証・処理フロー

Spring フレームワーク

今回、システムのベースとして、Web の遷移画面の引き継ぎがやりやすく、リクエストに対する統一的な処理を記述できるフロントコントロールパターンを利用できる Java フレームワーク「Spring」を利用して Web アプリケーション開発を行うこととした。

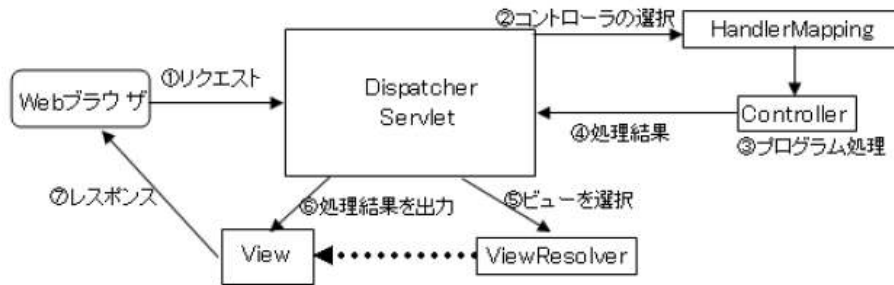


図 4 Spring 処理フロー

「AbstractController」クラスや「SimpleFormController」クラスを中心に使い、シラバスシステムを開発した。

「Controller」クラスを利用するには、Java ファイルの作成の他に Servlet を定義している「xml」ファイルにも定義を追加する必要がある。

```
<beans>
  <bean id="handlerMapping" class="org.springframework.web.servlet.handler.SimpleUrlHandlerMapping">
    <property name="mappings">
      <props>
        <prop key="/index.html">indexController</prop>
      </props>
    </property>
  </bean>
  <bean id="indexController" class="controller.IndexController">
    <property name="cacheSeconds"><value>0</value></property>
  </bean>
  <以下省略>
</beans>
```

図 5 Servlet 定義

```
public class SyllabusController extends AbstractController {
  public ModelAndView handleRequestInternal(HttpServletRequest request, HttpServletResponse response) throws Exception {
    ModelAndView modelAndView = new ModelAndView();
    (処理省略)
    modelAndView.setViewName("index"); //index.jsp ビュー選択
    return modelAndView;
  }
}
```

図 6 AbstractController

iText

旧シラバスシステムは、プリンタに印刷を行う場合、システム側にプリンタドライバを認識させる記述が必要であり、プリンタを変更するごとにプログラムを書き直す必要がある。また、サーバとプリンタを繋げる必要があり、サーバを置く場所を限定されてしまいます。そのため、プリンタ印刷は使い勝手が悪いというデメリットがあった。

そこで、Web から PDF 出力すれば、どのクライアント環境からでも印刷可能となり、サーバを手元に置いておく必要がなくなるなど、メリットがある。そこで、新シラバスシステムでは、講義情報を PDF 出力できるようにした。

```
Document doc = new Document(PageSize.A4); // 汎用の文書オブジェクトを生成
PdfWriter pw = PdfWriter.getInstance(doc, byteOut); // 出力先を指定し、文書を PDF として出力
doc.open();//出力開始
Font font = new Font(BaseFont.createFont("HeiseiKakuGo-W5","UniJIS-UCS2-H",BaseFont.NOT_EMBEDDED));// 日本語フォントの設定
doc.add(new Paragraph(15f,"シラバスシステム", font)); //出力
doc.close();// 出力終了

// ブラウザへの出力
response.setContentType("application/pdf");
response.setHeader("Content-Disposition","attachment; filename=syllabus.pdf");
response.setHeader("Cache-Control", "public");
response.setHeader("Pragma", "public");
response.setContentLength(byteOut.size());
OutputStream out = response.getOutputStream();
out.write(byteOut.toByteArray());
out.flush();
out.close();
```

図7 iText

7. まとめ

今回のシステム再開で最も改善した箇所は、教務課が操作する管理者機能である。ユーザ情報の更新、新規年度の追加、新規講義の追加、学生閲覧画面・表示／非表示設定など、シラバスシステムを操作する基本的なことは、教務課管理者画面より行うことができるように開発した。これにより、開発者がかかわることなく、メンテナンスフリーで運用することができる。しかし、システム開発に入る前に仕様をまとめたにもかかわらず、数回の追加要望で仕様変更があり、プログラミングし直す箇所がいくつかあった。仕様作成時にクライアントの要望をすべて引き出す技術が不足していると実感した。今後、プロジェクトマネージャの知識・技術も身につけていく必要があると感じている。

【参考文献】

村山雅彦 他, Spring による Web アプリケーションスーパーサンプル, ソフトバンク クリエイティブ (株)

以上