

情報システムと Web 技術 # 5 — ユーザエクスペリエンス/Web アプリケーションのまとめ

久野 靖*

2011.11.1

1 はじめに

本科目「情報システムと Web 技術」も今回で最終回となりました。もし可能ならいくから時間を取って、それぞれの人の「作品サイト」について開発を進めてもらったり、途中経過の報告をしてもらおうと思っていますので最終回の内容は少なめにしています。また、内容自体もやわらか目のものとして「ユーザエクスペリエンス」を取り上げました。

なお、成績については最初にお話した通り、毎回の出席・授業参加と最終レポートでつけます。最終レポートは次のようにお願いします。

- 最終レポートの提出期限は「11 月末日まで」とします。
- レポート内容は # 1～# 5 の中で出ている任意の課題 1 つを選び、その課題をやった結果を報告するものです。
- レポートは紙でも電子的提出 (PDF) でも構いません。紙の場合は 3F のメール室の久野のボックスに入れてメールで提出連絡を。
- レポートには次の内容が含まれること。
 1. 表紙 (題名: 「情報システムと Web 技術」レポート)、学籍番号、氏名、提出日付
 2. 方針 — 選択した課題の再掲と、それをどのようにやろうと思ったかの方針の説明
 3. 本体 — 上記の方針に基づく設計内容、作成したものの説明、実行結果など (長いソースコードは付録にしてください)。
 4. 考察・感想 — 考察にはやった結果何が分かったかを書く。感想は考察とは分けて書く。
 5. 付録 (あれば)。

ということでよろしく。

2 ユーザエクスペリエンス (UX)

2.1 4つのデザイン品質

デザインという言葉はわが国では「見た目をかっこよくするもの」という意味合いで使われることがまだまだ多いようです (しかし「設計」と訳すとまた全然違う意味になるのはどうしてでしょうか)。とりあえずここでは、ソフトウェアの中身から外見まで全てをふくめた設計を「デザイン」と呼ぶものとします。

このとき、デザインの善し悪しを定める品質の「ものさし」として次の 4 つの次元が考えられます。

*経営システム科学専攻

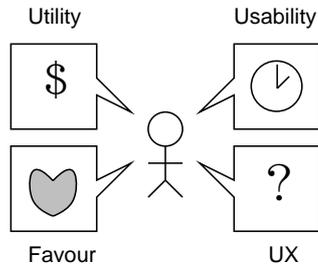


図 1: 4つのデザイン品質

- ユーティリティ (Utility) — 合目的性。ソフトウェアが提供する機能がどれくらい利用者にとって価値があるかという基準。
- ユーザビリティ (Usability) — 効用の分かりやすさ、利用方法の分かりやすさ、利用時の効率など、いわゆる「使いやすさ」
- ユーザエクスペリエンス (User Experience) — 利用者が持つ経験やそれに基づく愛着・思い出など
- (外観デザイン/意匠の) 好み (Favour) — 好ましさ/好き嫌い

このうちの、ユーティリティは「ソフトそのものの機能」として設計されるので、それは既に扱ったものと考えます(本当か?)。また、外観デザインはデザインの理論が色々あるけれど、これはいわゆるデザイナーさんをお願いするものとしてここでは除外します。

残りの2つですが、ユーザビリティについては「使いやすさ」として訳されることも多く、それを(学習、認知、効率など)さまざまな方向から評価する方法も確立していて、それなりに重要な要素として取り扱われています。

これに対して、ユーザエクスペリエンス (UX) は比較的新しい言葉であり、それが何を意味するかはあまり世の中には広く知られていませんでした。

しかし、iPhoneとかiPodのように革新的なインタフェースを持つ製品が現れ爆発的に売れるようになったため、それらは単なる「使いやすさ」だけではない何か、つまり「使った時の心地よさ」のようなものを持っている、ということは誰でも身近に感じるようになり、これによってユーザエクスペリエンスという言葉も市民権を得て来ている、というのが現状かと思います。

では、良いUXを提供するソフトはどのようにしたら設計できるのでしょうか? Steve Jobsみたいな天才なら何もしなくても考えただけでいいものが思い付くのかも知れませんが、我々凡人には何らかの「指針」が欲しいところです。

ユーザビリティであれば「速度」とか「タスクへの適合」とかいろいろな基準を設けて計測したりそれを改善したりできそうですが、UXの場合は何しろ「楽しさ」なのでそんな簡単に計測できそうにないです。ユーザをつれてきて試させればよって? 試すのはいいけど、それで「よくない」と言われた時にどうやって直せばいいか分からないですよ?

2.2 ペルソナ/シナリオ法

今回は、UXを指向した設計のツールに成り得るものとしてペルソナ(Persona)を紹介しましょう。元ネタは次の本です。

- アラン・クーパー著, 山形浩生訳, コンピュータはむずかしすぎて使えない!, 翔泳社, 2000.

この本の当該部分(ペルソナを説明している部分)はコピーを配布しますが、全体として面白い本ですのでなるべくなら購入して一読を薦めます。

で、ペルソナとは何かですが、「架空のキャラクタ」「仮想ユーザ」であり、さまざまなユーザタイプに対応するペルソナを用意して「それぞれの使用状況(シナリオ)にあてはめて検討」することで、ユーザの行動やシステムの設計の適否を断するのに使います。このために、ペルソナに対しては次のような情報を設定します。

- 名前 — ペルソナは具体的な人としてそのふるまいを想像することが必要なため、必ず名前を持たせます。
- 年齢、性別、体格、職業/学校/学年などの基本情報 — これらが無いと具体的なペルソナを思い浮かべられないので必要です。
- 目的、趣味、嗜好、個性 — このバリエーションによって「そのシステムに対して何を期待しどのように行動するか」が変わって来るので、それぞれのペルソナのふるまいが規定され、その側面から見たシステムの評価が行えます。

実際には、3番目の目的等のところで「どのようなペルソナ群」を想定するかで、UXの評価がかなり左右されます。つまりペルソナが物差しになるわけですから。このため、「このシステムに対してはどのようなペルソナが主要なものか」を検討して、(3体くらいの)主要ペルソナとして絞り込む、というのがとても重要な設計作業になります。ですから、ペルソナを設計するときはこの3番目の部分は「でっちあげる」のではなく一定の手法で調査などを通じて作り上げるとしています。

ペルソナ群が完成したら、ペルソナが持つさまざまな目標/目的に応じて、またシステムの使用されるさまざまな場面に応じて、ペルソナが何を考え、何を感じ、どうふるまうかを「予想」することで、システムがそのペルソナに対してどれくらい良いUXを提供できているのか/いないのかを評価し、また不十分なところがあればそれを改良する方向を検討します。

質問 1 なぜこのような方法が望ましいのでしょうか？ これまでやってきたように、「一般にユーザはどうふるまうか」を検討する、という方法ではなぜ駄目なのでしょう？ または「ユーザを連れて来て実際に試してもらったり意見を聞く」という方法ではなぜ駄目なのでしょう？

表 1: ペルソナその 1

名前	三宅勇史(ユウシ)。
基本情報	年齢 35、男性、既婚(子供男子 5 歳)、身長 170、体重 70、標準体型。
背景	大学では文学部英文学科だが、そんなまじめにやらず、テニスのサークルに熱中して青春を謳歌した。3 年になるときにちやっかり就活に切り替えて東証 2 部上場クラスの金融機関をゲット。
職業	金融機関の企画部門の中堅。仕事は現在かなり忙しいが、やりがいも感じている。企画なのでさまざまなアイデアを出す。上司に却下されることも多いが、たまにクリーンヒットが出ることもあり、そこそこにやっている。
生活	都内 3LDK マンション購入しローン抱える。奥さんはテニスサークル仲間で 1 歳下。金融機関で忙しいので家のことはあまりできず、いつも文句を言われている。しかも GSSM に入ってますます悪化し、申し訳なくも思う。テニスやランニングなど運動系が趣味なので休日などは家族で汗を流していたが、GSSM に入ってからはその時間が削られていてちょっとフラストレーション(奥さんは一層)。
目的	今の仕事はやりがいはあるが、先輩から教わったことと自己流のアイデアだけでこの先どうなのか不安がある。そこで GSSM に来れば体系的な知識が得られてこの先安泰だろうと思い入学した。
研究テーマ	日本の個人向け金融マーケティングの将来(S 研究室)
性格	ちやっかり屋で抜目なく明るく楽天的だが、上記のように先を読む要領の良さもあってそれで GSSM という「手」を打ったつもり。

この質問の答えは次のように要約されます。

表 2: ペルソナその 2

名前	阿部智子(トモコ)。
基本情報	年齢 29、女性、独身、身長 150、体重秘密、小柄で細身。
背景	高校までにプログラミングにはまり、大学の工学部情報工学科に進学してそのまま大手ソフトウェア製造企業に就職。
職業	ソフトウェア製造企業で開発者をずっとやっている。学部卒で就職したのもう 7 年、それなりに仕事もできるので開発のサブリーダーくらいまでやっている。開発はおおむねうまく行くが、時々顧客の仕様がすごく後で変更とか外注したモジュールとの結合が全然駄目とかで泥沼になることがあり、納得が行かない。GSSM に行けばこのような問題を克服する方法も見つかるのではと思い入学。
生活	都内 1R マンション賃貸生活。独身貴族だがそろそろ結婚はしたいけれど仕事が忙しく GSSM もさらに忙しいのであてがえない。休みの日はハーブとかフラワーとかの趣味で店や公園などめぐったりするが、仕事が火を吹いたり GSSM の課題が重なったりしたら週末は無くなる。
目的	とにかく真面目なので仕事が泥沼になるのが納得行かず、それを学問で何とかできるのではと思って GSSM に入学している。
研究テーマ	ソフトウェア開発の期限や品質を保証する方法 (N 研究室)
性格	真面目な性格でちょっと暗く大人しい。真剣にやれば報われるはずという信念があるが、ソフト業界は必ずしもそうでないのでそこに葛藤がある。

- 「一般的なユーザ」を対象とすると、システム開発者の都合でどうにでもねじ曲げてしまえるので、システム開発者の勝手に合わせたものができてしまう。また、さまざまなユーザに合わせるということは「あれもこれも」になって機能が多くなるばかりで、特定のユーザ層に絞った(そのユーザ層にとって心地よい)設計にならない。
- 「本当のユーザ」を連れて来るとすると、そのユーザが実際にシステムを使って見られるくらいシステムがちゃんとできていないとどうにもならない。「こんな風なシステムだったら」という話をユーザにしてもユーザには理解できない。必要があるたびにユーザに来てもらうことは大変すぎる。ユーザが自分の「気持ち」について正直に言うかどうか(あるいはそもそも自分の「気持ち」を正しく認識できるかどうか)は疑問。

これらのことから、一般的なユーザを対象とすることは適切でないし、本当のユーザに頼むことも適切でないわけです。

質問 2 想定したペルソナが「正しい」ことはどうやって検証できるのでしょうか? ペルソナが正しくない(実際のユーザを正しく反映していない)ことのリスクは問題ではないのでしょうか?

この質問の答えは次のように要約されます。

- ペルソナは「できるだけ具体的で、ふるまいが想像できることによって、先に挙げた『一般ユーザの罨』にはまらずにシステムの検討ができる」ことが最大の利点。だから、実際のユーザに非常に合っていないといけない、ということはない。正確さよりも厳密さを重視する(←非常に普通とは違った基準)。

2.3 例: GSSM 入学者のペルソナ

では、具体例として「GSSM 入学者」のペルソナを 3 人、久野がでっちあげたので、このペルソナにとって「GSSM」というシステムの UX はどうなのか、皆様に検討して頂きたいと思います。

演習 1 3 人のペルソナにとって、GSSM というシステムはどのような体験であり、どのように映っているでしょう。どういうところが嬉しく、どういうところが不満でしょう。ペルソナの設定に即して検討しなさい。

表 3: ペルソナその 3

名前	奈良武裕 (タケヒロ)。
基本情報	年齢 52、男性、既婚 (子供男子 22、女子 20)、身長 175、やせ型。
背景	団塊世代の 1 まわり下。大学で経営工学科ができたときに入り、それを活かして学部卒で小規模ながらゼネコンの資材部門に入社。
職業	ずっとゼネコンの資材部門で課長まで来たが、その上は難しそうだしそろそろ定年が見えて先を考えている。建設資材の流通は不合理なところも色々あるが、最適化理論などをうまく使えばずっとよくなるのではと思っていたので、それをネタに学位を取って大学に行きたい。修士からだが GSSM で博士まで進めば偉くなった知合いもいるので何とかならないかと。
生活	郊外 (上野から 45 分・バス 5 分) 1 戸建て。子供は 2 人とも大学でもうすぐ終わるので独立させてその先は奥さんと 2 人暮らしを予想。奥さんは趣味のフラワーアレンジメントで教室を開いたりして自由に行っている。自分は無趣味人間だったが会社でも教えたりするのは好きなので大学の先生になれば楽しそう。
目的	とにかく大学教員になるには博士は必要だからそのために GSSM を活用したい。修士からだと 5 年だがうまく短縮も…
研究テーマ	建設資材の流通システムの合理化可能性 (S 研究室)
性格	真面目で大人しく無趣味な仕事人間だが、教えている時は自分が偉くなったような気がして嬉しい。でもいぼるわけではなく、相手が教わったことが身について幸せなら自分も幸せ。

演習 2 皆様に課題としてお願いしている「共有日記サイト」のペルソナ群を作成してみなさい。主要ペルソナとしてどのようなユーザを考えればよいでしょうか。それぞれのペルソナはどのような特性を持っているのでしょうか。

2.4 ペルソナ/シナリオ法 (2)

ペルソナを作った後どうするかについては、クーパーの本にはほとんど書かれていません。これに対し、比較的新しい次の本は題名が題名だけあって、その先の部分まで書かれています。

- 棚橋弘季, ペルソナ作って、それからどうするの? — ユーザ中心デザインで作る Web サイト —, ソフトバンククリエイティブ, 2008.

これによると、ペルソナができれば各ペルソナごとにそれぞれの場面におけるシナリオを作り、次に複数のペルソナのシナリオを統合します。その統合したシナリオをサポートするように、システムを作る、ということになります。今回はとてもそこまでできないので (しかもこんなひとことで紹介したのは概要の概要にしか過ぎません)。

この本は非常に詳しくペルソナ/シナリオ法の思想背景や手順を解説していますが、読んだ時の面白さでいえばクーパーの本が絶対に面白いので、興味あればということ。

3 サーブレットによる Web アプリケーション製作 (まとめ編)

3.1 WebStates へのデータベースアクセス機能の追加

前回は WebStates フレームワークの導入までで手一杯でデータベースを使うところまで行かなかったので、今回ようやくデータベースが使えるように機能を追加しました。その概要をまとめます。

- 文字列変数 `sqlurl`, `sqluser`, `sqlpass` にそれぞれ JDBC URL、DB のユーザ名、DB のパスワードを入れておくと、初期化時に自動的にデータベース接続をおこなう。

- `String[][] select(String)` — SQL の `select` 文を発行して、結果を文字列の 2 次元配列として返す。
- `int update(String)` — SQL の `insert` 文、`delete` 文、`update` 文を発行したい場合には結果は帰らないのでこちらを使う。返される製数値は変更のあったタプル数。

DBMS とやりとりするというのは結局これだけなので、これだけを使って今回は掲示板を動かしてみます。

3.2 掲示板の状態遷移は？

次に、掲示板の状態遷移を考えてみましょう。ユーザを特定するために最初に `login` 画面があり、正しく `login` すると共有日記が見られる。そこまではいいですね。では、日記を書き込む時はどうでしょう？ 別画面に遷移して、そこで書き込む、というのは当たり前のように思えます。

しかし、ユーザに (誰かのペルソナに!) なったつもりで考えてみてください。せっかく今書き込まれている日記を見ているのに、画面が切り替わってそれらが消えてしまって、それらを「覚えた状態で」自分の書き込みをする、というのは嬉しくないと思いませんか? (図 2 上)

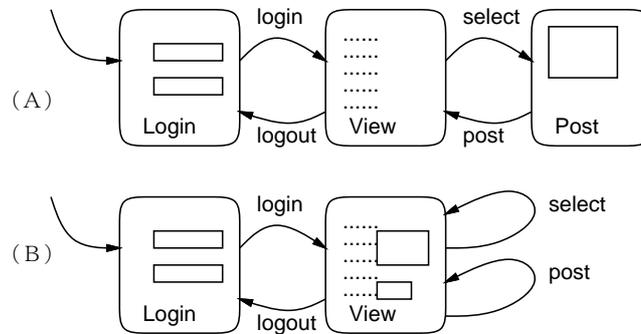


図 2: 共有日記の状態遷移案

そういうわけで、日記を見ている画面に書き込み一覧も一緒に表示され、そこで既にある書き込みを見ながら自分の書き込みを打ち込めるように考える (図??下)。

あともう 1 つ、「どの日の記事を」書き込むかについての指定方法の問題がある。たとえば、(A) の別画面方式で、そこに年月日を入れる部品があって選択することもできるけれど、それは嬉しくないですよ? せっかく日記の一覧が見えているのだから、「この日」と言える方がずっと嬉しいですね?

だから、たとえ別画面であっても、日を選択するとその日の記事を書き込む画面に移る、というふうにしたいわけです。で、別画面でなく一緒にする場合は…たとえば、日記が縦に長くて、その最後に書き込み一覧がある、ということだと、結局書いている時はその日の日記が見られないので別画面と大してかわらない (スクロールすれば見れるけれど)。そこで、書きたい日を選ぶと同じ表示画面だけれど書き込み欄がその日のところに挿入された状態に切り替わる、というふうに考えてみました (図 3)。

3.3 多数のボタンのある例: Sample51

さて、先のような方式を作成するとして、伝統的なやり方 (そのつど HTML を生成する) では、特定の日を選択するとフォームが提出され、その日のところに書き込み欄がある状態のフォームが返されてくるわけです。つまり、すべての日にボタンのついた大きいフォームを作ることになります。そして、それぞれの日は互いに区別をつけられるようにする必要があります。このため、それぞれの日のボタンは次のように「日番号」を持つようにします。

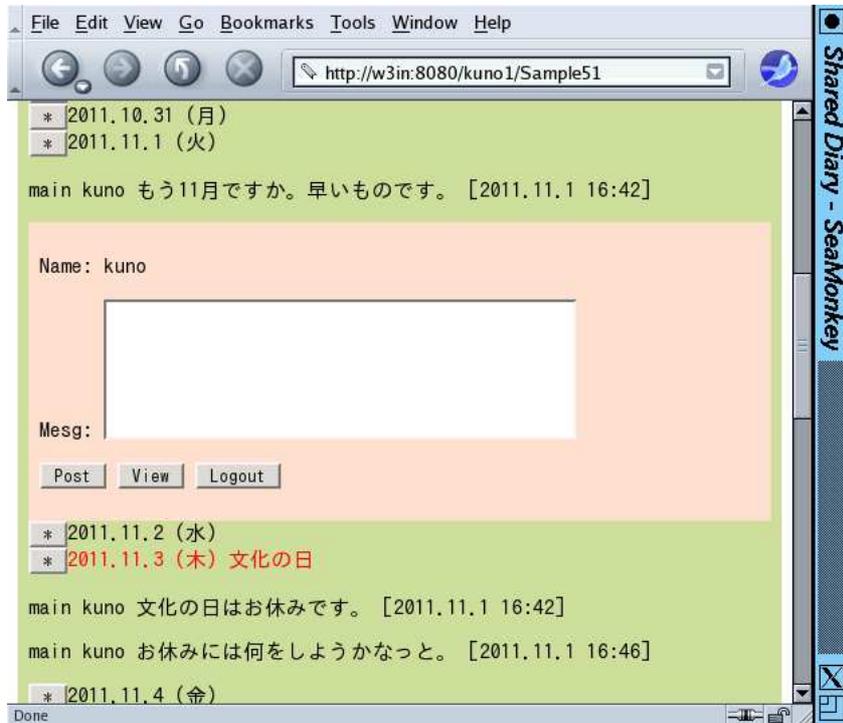


図 3: 書き込み欄が選択に応じて移動する案

```
<button name=cmd value=285>*</button>
```

いちおう、この版の全体を示しておきます。

```
import javax.servlet.*;
import javax.servlet.http.*;
import java.lang.reflect.*;
import java.io.*;
import java.util.*;
import java.sql.*;

public class Sample51 extends WebStates {
    // variables
    public String cmd, uname, pass, user, mesg, sday, eday, cday;

    // initailzation
    public Sample51() {
        sqlurl = "jdbc:postgresql://smb:5432/kuno";
        sqluser = "kuno";
        title = "Shared Diary";
        attrs = new String[]{"user", "sday", "eday", "cday"};
        params = new String[]{"cmd", "uname", "pass", "mesg"};
        states = new String[][]{
            {"Login", "cmd:login:doLogin"},
            {"View", "cmd:post:doPost", "cmd:logout:doLogout", "*:*:doSwitch"}};
    }
}
```

```

// showXXX() --- print page for state XXX.
public void showLogin() {
    putLogin();
}
public void showView() {
    String[][] d = select(String.format(
        "select dayno, year, month, date, day, holiday from diary_date where" +
        " dayno between %s and %s", sday, eday));
    String[][] t = select(String.format(
        "select dayno, txtno, catg, uname, text from diary_text where" +
        " dayno between %s and %s", sday, eday));
    int p = 0;
    pr.println("<div class=main><form method=post action='#'>");
    for(int i = 0; i < d.length; ++i) {
        String color = "black";
        if(d[i][4].equals("±")) { color = "blue"; }
        if(d[i][4].equals("日")) { color = "red"; }
        if(!d[i][5].equals("")) { color = "red"; }
        pr.printf("<div style='color:%s'>" +
            "<button name=cmd value=%s>*</button>%s.%s.%s (%s) %s</div>\n",
            color, d[i][0], d[i][1], d[i][2], d[i][3], d[i][4], d[i][5]);
        while(p < t.length && t[p][0].equals(d[i][0])) {
            pr.printf("<p>%s %s %s</p>\n", t[p][2], t[p][3], t[p][4]);
            p = p + 1;
        }
        if(d[i][0].equals(cday)) { putForm(); }
    }
    pr.println("</form></div>");
}

// doXXX() --- do some operation; return state or next operation.
public String doLogin() {
    String[][] r = select(String.format(
        "select pass from diary_user where uname = '%s'", uname));
    if(r.length != 1 || !r[0][0].equals(pass)) {
        putNote("invalid login."); return "Login";
    } else {
        Calendar c = Calendar.getInstance();
        r = select(String.format(
            "select dayno from diary_date where year=%s and month=%s and date=%s",
            c.get(Calendar.YEAR), c.get(Calendar.MONTH)+1, c.get(Calendar.DATE)));
        if(r.length != 1) { putNote("today cannot be found???"); return "Login"; }
        int i = Integer.parseInt(r[0][0]), j = Math.max(1, i-20), k = j + 60;
        setattr("cday", cday = ("" + i));
        setattr("sday", sday = ("" + j)); setattr("eday", eday = ("" + k));
        putNote("welcome."); setattr("user", user = uname); return "View";
    }
}

```

```

}
public String doLogout() {
    setattr("user", null); return "Login";
}
public String doSwitch() {
    try {
        int i = Integer.parseInt(cmd), j = Math.max(1, i-20), k = j + 60;
        setattr("cday", cday = (" " + i));
        setattr("sday", sday = (" " + j)); setattr("eday", eday = (" " + k));
    } catch(Exception ex) { };
    return "View";
}
public String doPost() {
    String[] [] r = select(String.format(
        "select txtno from diary_text where dayno=%s", cday));
    int i = 1;
    if(r.length > 0) {
        r = select(String.format(
            "select max(txtno)+1 from diary_text where dayno=%s", cday));
        i = Integer.parseInt(r[0][0]);
    }
    Calendar c = Calendar.getInstance();
    update(String.format(
        "insert into diary_text values(%s,%d,'main','%s','%s [%d.%d.%d %02d:%02d]')",
        cday, i, user, msg, c.get(Calendar.YEAR), c.get(Calendar.MONTH)+1,
        c.get(Calendar.DATE), c.get(Calendar.HOUR_OF_DAY), c.get(Calendar.MINUTE)));
    return "View";
}

// putXXX() --- print out specific HTML part
void putLogin() {
    pr.println("<div class=form><form method=post action='#'>");
    pr.println("<p>Username: <input name=uname type=text size=12></p>");
    pr.println("<p>Password: <input name=pass type=password size=12></p>");
    pr.println("<p><button type=submit name=cmd value=login>Login</button></p>");
    pr.println("</form></div>");
}
void putForm() {
    pr.println("<div class=form>");
    pr.printf("<p>Name: %s</p>\n", user);
    pr.println("<p>Mesg: <textarea name=msg cols=40 rows=5></textarea></p>");
    pr.println("<p><button name=cmd value=post>Post</button>");
    pr.println("<button name=cmd value=view>View</button>");
    pr.println("<button name=cmd value=logout>Logout</button></p>");
    pr.println("</div>");
}
}
}

```

3.4 DOM 操作によるフォームの調整: Sample52

先のバージョンは一応使えますが、どうもまだいまいちな気がします。ボタンが縦にずらっと並んでいる見た目も問題ですし、フォームが「挿入」されるので日記の並びがその箇所で中断されている感じがします。

そこで、単純な HTML から 1 つ進歩して、JavaScript を使って「クリックした位置にフォームが移動してくるとともに、その日の日記を書くモードになる」ようにしてみました (図 4)。

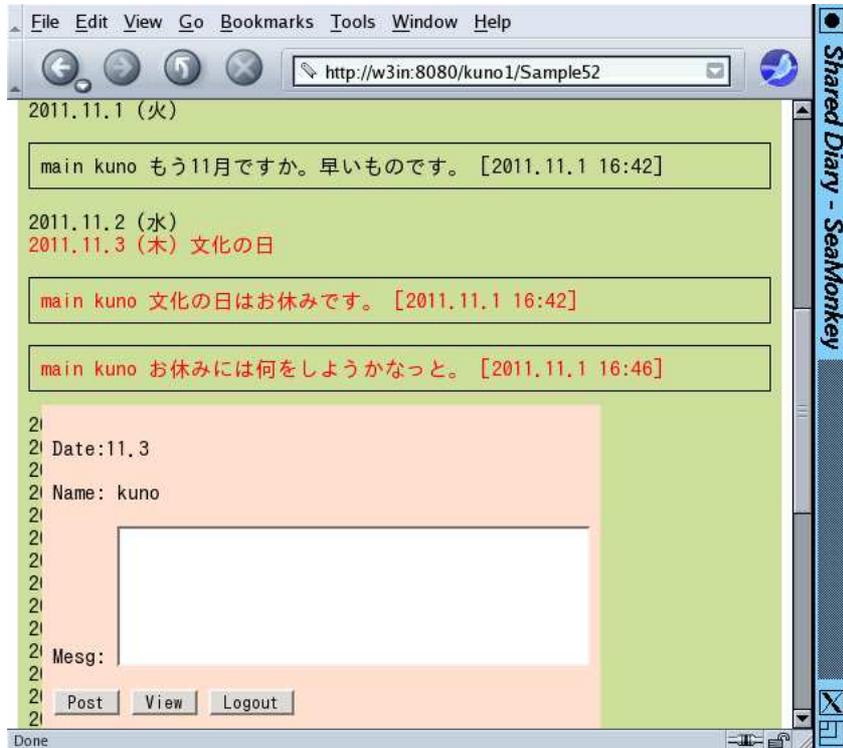


図 4: 書き込み欄がクリックで移動する案

```
import javax.servlet.*;
import javax.servlet.http.*;
import java.lang.reflect.*;
import java.io.*;
import java.util.*;
import java.sql.*;

public class Sample52 extends WebStates {
    // variables
    public String cmd, uname, pass, user, mesg, sday, eday, cday, wday;

    // initialization
    public Sample52() {
        sqlurl = "jdbc:postgresql://smb:5432/kuno";
        sqluser = "kuno";
        title = "Shared Diary";
        attrs = new String[]{"user", "sday", "eday", "cday"};
        params = new String[]{"cmd", "uname", "pass", "mesg", "wday"};
    }
}
```

```

states = new String[][]{
    {"Login", "cmd:login:doLogin"},
    {"View", "cmd:post:doPost", "cmd:logout:doLogout", "*:*:doSwitch"}};
}

// showXXX() --- print page for state XXX.
public void showLogin() {
    putLogin();
}

public void showView() {
    String[][] d = select(String.format(
        "select dayno, year, month, date, day, holiday from diary_date where" +
        " dayno between %s and %s", sday, eday));
    String[][] t = select(String.format(
        "select dayno, txtno, catg, uname, text from diary_text where" +
        " dayno between %s and %s", sday, eday));
    int p = 0;
    pr.println("<div class=main>");
    for(int i = 0; i < d.length; ++i) {
        String color = "black";
        if(d[i][4].equals("±")) { color = "blue"; }
        if(d[i][4].equals("日")) { color = "red"; }
        if(!d[i][5].equals("")) { color = "red"; }
        pr.printf("<div onclick='sel(%s,this,%s,%s)' style='color:%s'>" +
            "%s.%s.%s (%s) %s<br>\n",
            d[i][0], d[i][2], d[i][3], color,
            d[i][1], d[i][2], d[i][3], d[i][4], d[i][5]);
        while(p < t.length && t[p][0].equals(d[i][0])) {
            pr.printf("<p style='padding: 1ex; border: solid black 1px'>%s %s %s</p>\n",
                t[p][2], t[p][3], t[p][4]);
            p = p + 1;
        }
        pr.println("</div>");
    }
    putForm();
    pr.println("</div>");
}

// doXXX() --- do some operation; return state or next operation.
public String doLogin() {
    String[][] r = select(String.format(
        "select pass from diary_user where uname = '%s'", uname));
    if(r.length != 1 || !r[0][0].equals(pass)) {
        putNote("invalid login."); return "Login";
    } else {
        Calendar c = Calendar.getInstance();
        r = select(String.format(

```

```

        "select dayno from diary_date where year=%s and month=%s and date=%s",
        c.get(Calendar.YEAR), c.get(Calendar.MONTH)+1, c.get(Calendar.DATE));
    if(r.length != 1) { putNote("today cannot be found???"); return "Login"; }
    int i = Integer.parseInt(r[0][0]), j = Math.max(1, i-20), k = j + 60;
    setattr("cday", cday = ("" + i));
    setattr("sday", sday = ("" + j)); setattr("eday", eday = ("" + k));
    putNote("welcome."); setattr("user", user = uname); return "View";
    }
}
public String doLogout() {
    setattr("user", null); return "Login";
}
public String doSwitch() {
    if(!wday.equals("")) {
        setattr("cday", cday = wday);
        int i = Integer.parseInt(cday), j = Math.max(1, i-20), k = j + 60;
        setattr("sday", sday = ("" + j)); setattr("eday", eday = ("" + k));
    }
    return "View";
}
public String doPost() {
    setattr("cday", cday = wday);
    String[][] r = select(String.format(
        "select txtno from diary_text where dayno=%s", cday));
    int i = 1;
    if(r.length > 0) {
        r = select(String.format(
            "select max(txtno)+1 from diary_text where dayno=%s", cday));
        i = Integer.parseInt(r[0][0]);
    }
    Calendar c = Calendar.getInstance();
    update(String.format(
        "insert into diary_text values(%s,%d,'main','%s','%s [%d.%d.%d %02d:%02d] ')",
        cday, i, user, msg, c.get(Calendar.YEAR), c.get(Calendar.MONTH)+1,
        c.get(Calendar.DATE), c.get(Calendar.HOUR_OF_DAY), c.get(Calendar.MINUTE)));
    return "View";
}

// putXXX() --- print out specific HTML part
void putLogin() {
    pr.println("<div class=form><form method=post action='#?'>");
    pr.println("<p>Username: <input name=uname type=text size=12></p>");
    pr.println("<p>Password: <input name=pass type=password size=12></p>");
    pr.println("<p><button type=submit name=cmd value=login>Login</button></p>");
    pr.println("</form></div>");
}
void putForm() {

```

```

pr.println("<div id=f0 class=form dstyle='padding:2ex'>");
pr.println("<form method=post action='#'>");
pr.println("<input type=hidden id=i0 name=wday value=1>");
pr.println("<p id=p0>Date: 2011.1.1</p>");
pr.printf("<p>Name: %s</p>\n", user);
pr.println("<p>Mesg: <textarea name=mesg cols=40 rows=5></textarea></p>");
pr.println("<p><button name=cmd value=post>Post</button>");
pr.println("<button name=cmd value=view>View</button>");
pr.println("<button name=cmd value=logout>Logout</button></p>");
pr.println("</form></div>");
}
void putScript() {
String[] a = {
    "function sel(d,e,mm,dd) {",
    "    var form = document.getElementById('f0');",
    "    form.style.position = 'absolute';",
    "    form.style.left = e.offsetLeft + 10;",
    "    form.style.top = e.offsetTop + e.offsetHeight + 10;",
    "    document.getElementById('i0').value = d;",
    "    document.getElementById('p0').innerHTML = 'Date:' + mm + '.' + dd;",
    "}",
};
pr.println("<script type='text/javascript'>");
for(String s: a) { pr.println(s); }
pr.println("</script>");
}
}

```

演習 3 これらのうち好きな版をもとに改良や機能追加を行ってみなさい。

4 まとめ

この講義では、業務分析やデータベース設計、システム設計などの際に使われる代表的な技法を紹介するとともに、ちょっとだけ「体験」という部分と、データベースに基づく Web アプリケーションを JDBC+サーブレットにより構成してみるという部分を組み合わせて実施してみました。今年が最初の年なので色々ご不便をお掛けしたところもあると思いますが、何かのヒントになれば幸いです。おつかれ様でした。