

携帯Webアプリケーション開発の基本と  
Java/OSSフレームワーク「mobylet」の紹介

竹内 真(株式会社レイハウオリ)

2009/06/13 Seasar Conference 2009 Spring

## ▶ 竹内真 (Takeuchi Shin)

### ■ 簡単な経歴

- 富士ソフト(株) → フリー → (株)レイハウオリ設立
- Webアプリケーション開発畑で育つ

### ■ 現在

- 色々なベンチャー企業と共同でのWebコンテンツ制作
- (株)リクルートへの技術支援
  - ◆ 共通化基盤チームのアドバイザー
  - ◆ 主にモバイル/SEO/フレームワーク構築などを担当

### ■ 未来

- mobyletを作りまくる & 使いまくる

## ▶ 携帯Webアプリケーション開発の基本

### ■ 携帯Webの歴史と今

- 3大キャリアの歩んだ歴史
- 激震の2009年度と今後

### ■ PCと携帯の違い

- 携帯だからできること
- 携帯だからできないこと/困ること

### ■ Javaと携帯

- 恐怖の「絵文字」

## ▶ Java/OSSフレームワーク「mobylet」の紹介

### ■ mobylet ?

- mobyletの紹介
- mobylet(ver 0.1)にできること

### ■ デモ

- フレームワークを一切使わない場合
- S2(SAStruts)と組み合わせた場合の設定について

### ■ ドキュメント & サポート

- ドキュメントの紹介
- サポート & 開発体制

# まずは基本

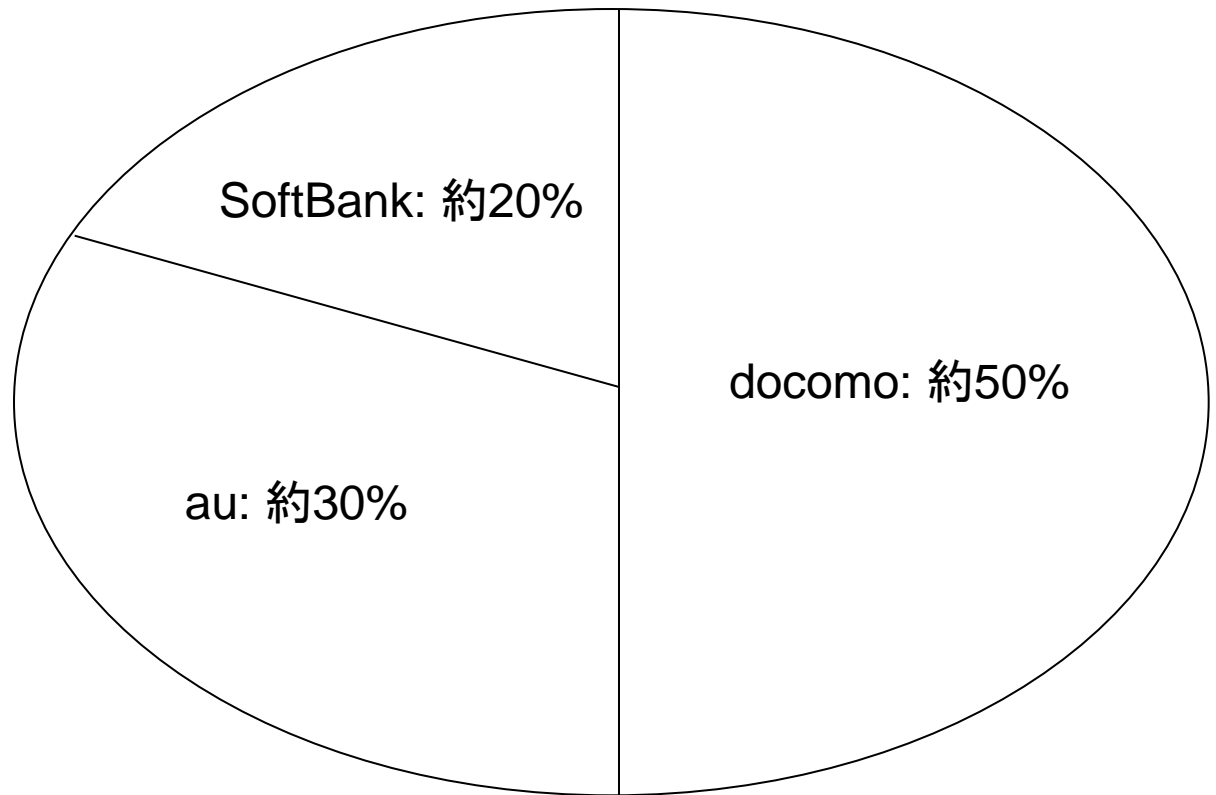
※知ってる人はつまらない話ですいません。。。

## ▶ 3大キャリアの歩んだ歴史(1/4)

### ■ 3大キャリアとは？

- docomo
- au
- SoftBank

現在のシェア(概算)



## ▶ 3大キャリアの歩んだ歴史(2/4)

### ■ docomoについて

- 1999年に「iモード」が登場
  - ◆ cHTMLというHTMLの廉価版のような仕様で記述する
  - ◆ uidやキャリア課金など「公式サイト」有利な設計
- 2005年頃からXHTML+i-CSSが利用可能に
  - ◆ この頃、cHTMLでもtableタグが使用可能になるなど、携帯サイトでリッチな構造のデザインが可能になってくる。
- 2009年「iモードブラウザ2.0」が発表
  - ◆ 携帯Webに革命が起こる。。。詳しくは後述。

## ▶ 3大キャリアの歩んだ歴史(3/4)

### ■ auについて

- 1999年に「ezweb」が登場
  - ◆ iモードに遅れること9ヶ月で登場
  - ◆ ezwebサーバという独特なGWサーバが存在する
  - ◆ 基本的にはHDMLという特殊な言語で制作する
- 2001年頃からXHTMLが使用可能になる
- 2008年5月31日にHDML端末のサービスが終了
  - ◆ つまりXHTML一本になる



## ▶ 3大キャリアの歩んだ歴史(4/4)

### ■ SoftBankについて

- 1998年に「スカイウェブ」が登場
  - ◆ 「iモード」よりも登場が早いものの、いわゆるWWWではなく閉鎖的なコンテンツサービス網
- 1999年に「J-スカイ」が登場
  - ◆ ここからインターネット接続が可能となった
- 2003年、J-PhoneからVodafoneへ
- 2006年、VodafoneからSoftBankへ
- 2010年3月、2.5G以下の端末のサービス終了へ

## ▶ 激震の2009年度と今後(1/2)

### ■ 2009年度に起こる(起こった)こと

- iモードブラウザ2.0端末が発売となった
  - ◆ JavaScriptが使用可能(携帯ブラウザ初)
  - ◆ リファラが取れるようになった(docomo初)
  - ◆ QVGAとVGAの切り替えが可能(携帯ブラウザ初)
- 2010年3月末でSoftBank 2.5G端末のサービスが終了
  - ◆ まるで別キャリアのような2.5G以前と3Gのサポートが不要
  - ◆ 5バイト絵文字という史上最高にやっかいな文字が無くなる

2009年度は新しい仕様が生まれ、古い仕様が消える過渡期。

## ▶ 激震の2009年度と今後(2/2)

### ■ 今後について

- 他のキャリアもdocomoの新仕様を取り込む可能性が有る
  - ◆ 今までの歴史上、docomoの仕様は携帯業界全体の仕様になる
- Ajax等を使ったリッチなインターフェースが可能になる
- PCと同様にXSSの危険性が発生する
- ブラウザの解像度の切り替えを使ったUIも登場する
- Jsessionidはクッキーを使うことがスタンダードになる
  - ◆ 今まではそもそもdocomoでクッキーが使えなかった
  - ◆ auやSoftBankも端末でクッキーを保持できるものが増えてきた

## ▶ 携帯だからできること(1/5)

- 公式サイトに申請することができる
  - キャリア毎に「公式サイト」に申請することができる
    - ◆ 公式サイトに登録されると。。。。
      - キャリアの公式メニュー(iメニューなど)でサイトを辿れる
      - キャリアの検索ページで上部に表示される
      - キャリア課金システムが利用できる
      - 「iコンシェル」などキャリア毎の付帯システムが利用できる
- 公式サイトでない携帯サイトを「勝手サイト」と呼ぶ
  - 結構オフィシャルな呼ばれ方になってきた気がします。。。

## ▶ 携帯だからできること(2/5)

- アクセスしたユーザを特定することができる
  - docomo
    - ◆ 「公式サイト」の場合は「uid」という契約に一意的なidが取得可能
    - ◆ 「iモードID」という契約に一意的なidも取得可能(勝手サイトもOK)
  - au
    - ◆ 「サブスクライバナンバー」という契約者一意的なidが取得可能
  - SoftBank
    - ◆ 「uid」(x-jphone-uid)という契約に一意的なidが取得可能
- 端末を特定することができる
  - シリアルナンバーを取得可能(auは不可能)

## ▶ 携帯だからできること(3/5)

### ■ 位置情報を取得することができる

- GPS搭載端末はGPSを使用した緯度/経度情報を取得することが可能
- GPS非搭載端末でも、基地局の位置から簡易的に割り出した緯度/経度情報を取得することが可能
  - ◆ 昔は簡易測地の精度も荒く、あまり使えなかったが、最近は基地局3点から測地する方式を取っており、かなり精度が高くなった。
  - ◆ とはいえ、首都圏などの基地局が密集しているエリアのことであって、地方や3点から測地が難しい海岸沿いなど、場所によってはどうしても荒い値になってしまう。

## ▶ 携帯だからできること(4/5)

- 携帯からのアクセスを制限することができる
  - IPアドレスで制限することができる
    - ◆ 携帯ブラウザからのアクセスは、必ず「キャリアゲートウェイ」を通ってくるため、各キャリアゲートウェイに振られているIPで制限することで、携帯(もしくは特定のキャリア)のみにアクセスを限定することが可能。
  - UserAgentで制限することができる
    - ◆ 携帯ブラウザはそれぞれPCのブラウザとは異なるUserAgentを持っているため、UserAgentヘッダを判定して制限することが可能
    - ◆ しかし、Firefoxやその他のツールを利用してUserAgentは偽装が可能なので、セキュリティ上の制限として使用することは避けたい

## ▶ 携帯だからできること(5/5)

### ■ 絵文字が使える

- 画像を使用しなくてもサイトを飾れる

- ◆ 絵文字1文字は全角1文字相当の2バイトしか使用しないため、軽量にサイトを飾ることができる

- コミュニケーションI/Fとして有用

- ◆ 静的コンテンツやメール、その他、絵文字を使うことで、文字だけのインターフェースに様々な表現を追加することが可能となる
- ◆ つまり、携帯サイトである以上、絵文字は有効活用したい資産のひとつであると言える



## ▶ 携帯だからできないこと/困ること(1/6)

- JavaScriptが使えない(使えなかった。。。)
  - 「iモードブラウザ2.0」搭載端末以外は全て使えない
    - ◆ このためトップページのonloadでリダイレクトして、携帯サイトとPCサイトを振り分けるような処理も割りと存在した
    - ◆ しかし、iモードブラウザ2.0の登場で、この裏技トリックもあまり有効では無くなって来るだろう。
  - 非同期通信も不可能だった
    - ◆ Flash対応端末の一部でFlashから非同期通信を行うパターン以外での非同期通信/制御は不可能だった。
  - XSSも起こらなかった
    - ◆ 今後、特に既存のサイトにはXSSの被害が懸念される。。。。

## ▶ 携帯だからできないこと/困ること(2/6)

### ■ キャッシュサイズ制限

- 端末毎に1ページの最大容量が決まっている
  - ◆ 昔だと5K、最近だと500Kといった、キャッシュサイズ制限がある
  - ◆ ボリューム(文字/画像)の多いページは複数ページに分割するなどの必要がある
- サイト毎に端末制限を掛ける必要もある
  - ◆ 例えば画像を多く必要とするサイトなどは100K以上のキャッシュサイズを持つ端末に限定するなどの制限を掛ける必要もある
- キャッシュサイズを超えた場合
  - ◆ 一部画像が表示されないパターンや、そもそも画面が表示できないケースなど、キャリアや端末によって挙動は様々。

## ▶ 携帯だからできないこと/困ること(3/6)

### ■ クッキー

#### ● クッキーを使えない端末がある

- ◆ au/SoftBankはクッキーが使える端末のシェアが増えてきた
- ◆ docomoは「iモードブラウザ2.0」対応端末のみ
- ◆ auの古い機種はキャリアゲートウェイでクッキーを保持する

#### ● JSESSIONIDの持ち方が悩ましい

- ◆ 以前はクッキーを使用せず、クエリにjsessionidパラメータを持つパターンが基本だったが、リファラによる情報漏えいの危険性や、docomoの「iモードブラウザ2.0」の登場により、これから飛躍的にクッキーを利用するパターンが増えると思われる。
- ◆ とはいえクッキーに対応していない端末はクエリで持つ必要がある

## ▶ 携帯だからできないこと/困ること(4/6)

### ■ ブラウザサイズ(画面解像度)

- 横幅120px程度のものから480px程度のものまで

- ◆ 例えば商品画像を表示するにも小さなブラウザでは小さな画像しか表示できないし、大きなブラウザでは小さくて見えないし、といった問題に直面する
- ◆ 端末のブラウザサイズ毎に商品画像の大きさも変えて表示することが望ましい
- ◆ とはいえ、大きな画像を使って<img>タグのwidth="80%"とかでリサイズするとキャッシュサイズオーバーとなるパターンもあるので要注意(実際に返す画像サイズを小さくする方が良い)

## ▶ 携帯だからできないこと/困ること(5/6)

### ■ SSL(証明書)

#### ● オレオレ証明書が使えない

- ◆ auの場合、オレオレ証明書では画面を表示することすらできないので要注意(その他のキャリアは警告が表示される)

#### ● 認証局の対応も端末毎にバラバラ

- ◆ 唯一Verisignだけは全端末OK
- ◆ その他の認証局のものは古い端末で対応していない場合があるので要注意

#### ● SSLサイトの場合の注意事項

- ◆ docomoで画像1枚単位の通信速度が極度に遅くなる
- ◆ auで何気ない改行が<br>と認識される場合がある

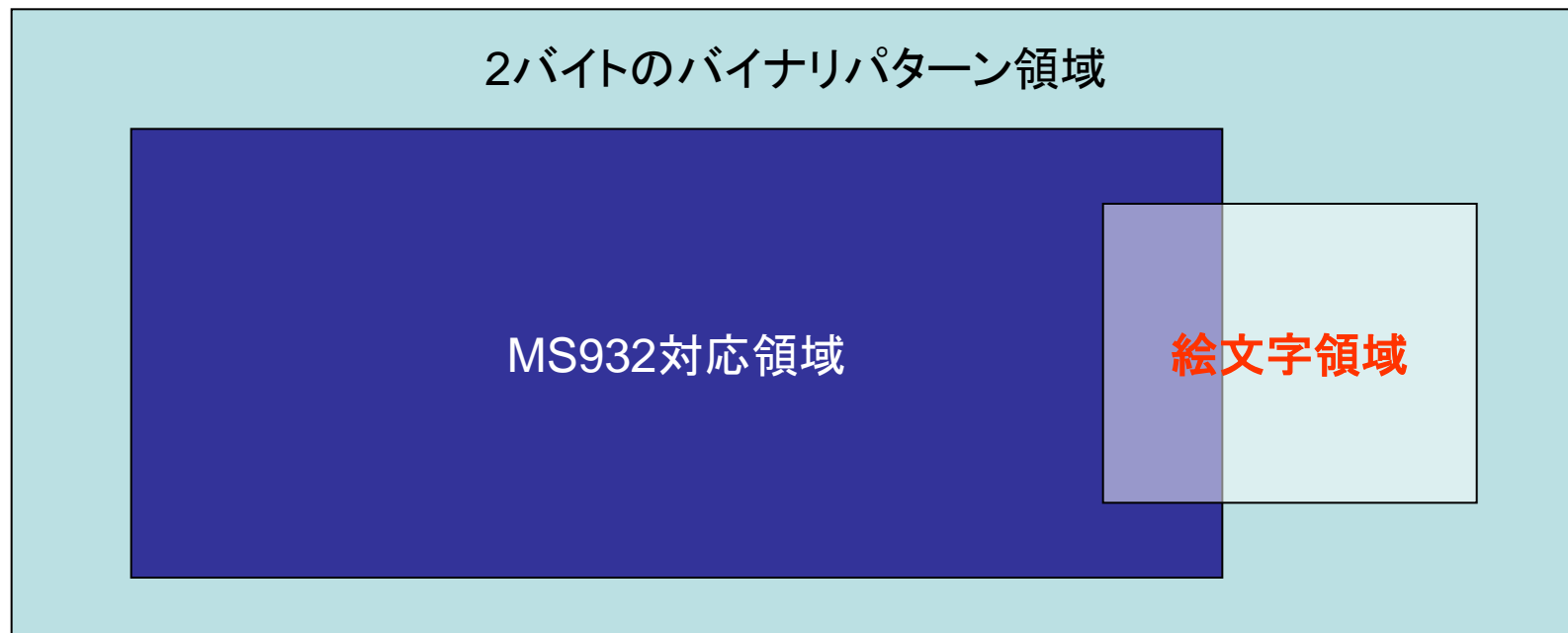
## ▶ 携帯だからできないこと/困ること(6/6)

### ■ その他

- メール: SPFレコードを設定する必要がある
- メール: キャリアのSMTPサーバは連続送信にシビア
- クローラ: モバイル専用のクローラが存在する
  - ◆ IP制限を掛ける場合はクローラのIPも開けないと、検索エンジンに一生インデックスされないので要注意
- WEB: `<input type="file">`は基本使えない
- WEB: リダイレクトは最大3回まで(2回まで推奨)
- WEB: 外部CSSはdocomoで読めない
- WEB: `secure.softbank.ne.jp`に泣く

## ▶ 恐怖の「絵文字」(1/7)

- 絵文字とは文字コード上の私的領域に存在する
  - つまり、例えば「SJIS」「MS932」とかで文字化けする



## ▶ 恐怖の「絵文字」(2/7)

- Javaは必ずUnicodeに変換する必要がある
  - 文字として扱う以上「char」として保持する必要があるが、入力文字(byte)→char変換する場合に文字化けする
  - 文字化けしないようにするには「絵文字」を正常に変換できる「文字コード」を用意する必要がある
  - しかもキャリア毎に「絵文字」の文字コードは異なるため、全キャリア分文字コードを作る必要がある
  - 文字コードを作るなんて、非常に情報も少ないし、面倒くさいし、難しいし。。。。ということで携帯サイトを作るモチベーションが著しく落ちるところは**ここだと思った**



▶ 恐怖の「絵文字」(3/7)

- なので。。。。

mobyletが<sup>レ</sup>作りました。

## ▶ 恐怖の「絵文字」(4/7)

### ■ キャリア毎の絵文字の変換

- それでもやっぱりキャリア毎に絵文字のコードが違うってことは、docomoの「太陽」をauで表示するときに、ちゃんとauの「太陽」の絵文字に変換してあげないといけない
- docomoが約250、auが約600、SoftBankが約400ある絵文字。全部のマッピングをするなんて気が遠くなる。。。
- ということで携帯サイトを作るモチベーションが著しく落ちるところは**ここだとも思った**

▶ 恐怖の「絵文字」(5/7)

- なので。。。。

mobyletが<sup>レ</sup>作りました。

## ▶ 恐怖の「絵文字」(6/7)

### ■ ひとつのViewで3キャリア出し分け

- 特に問題となっていた絵文字の出し分け部分をmobyletのFilter部分で吸収したため、1つのViewを作ることで、何も考えず、勝手に3キャリア出し分けができるようになりました
- このため、mobyletを使って携帯サイトを作る開発者は、

**面倒な絵文字を一切  
意識する必要がありません！**

▶ 恐怖の「絵文字」(7/7)

- Javaにとって恐怖だった絵文字は。。。

全く恐怖では  
なくなりました！

なにはともあれ

# 携帯WebApp フレームワーク

の「mobylet」です。

※それ以外の用途でのご利用は不可能(多分)です。

## ▶ mobyletの紹介

- 携帯Webアプリケーションを構築するためのフレームワーク
  - 3大キャリアに対応 (docomo/au/SoftBank)
    - ◆ 但し、SoftBankの2.5G以下の端末は除く(2010年3月サービス終了端末)
  - 上記以外のキャリアやPCでアクセスも可能なような設計
  - 依存ライブラリがほぼ0
  - 様々なフレームワークと融合できる(もちろんSeasar2とも)
  - 導入コストが小さい(簡単に導入できる)
  - 絵文字に完全対応
  - バリューエンジン社の無料端末プロファイルをデフォルトでサポート
  - 最小公倍数的な設計(あえてサービス依存するようなクラスも作る)

## ▶ mobylet (ver 0.1) にできること (1/5)

### ■ キャリアや端末の判定ができる

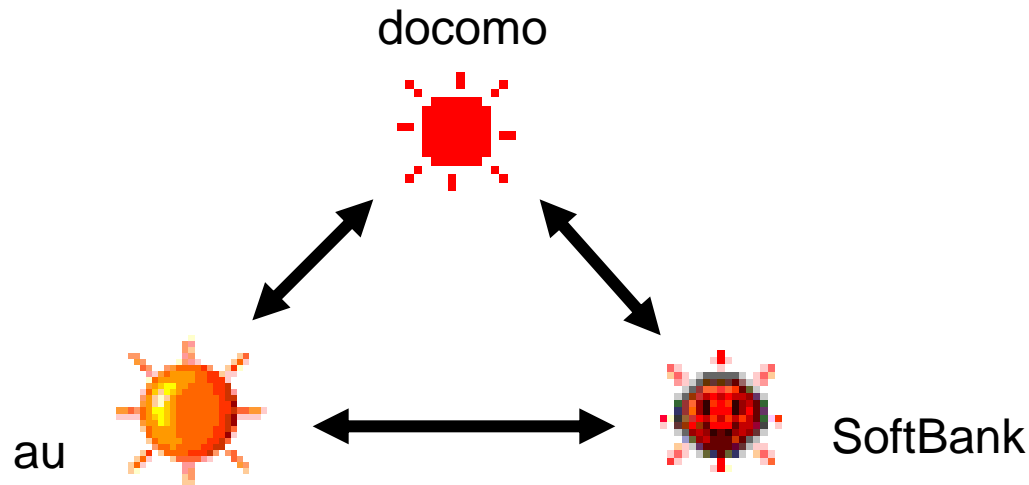
- アクセスしたUserAgentをキーにしてキャリアを判別します
  - ◆ Mobylet m = MobyletFactory.getInstance();
  - ◆ m.getCarrier();
- アクセスしたUserAgentをキーに端末を判定し、プロフィール情報とのマッチングを行い、デバイス情報を取得します。
  - ◆ m.getDevice();
- 他にも「Mobylet」クラスを基点に色々な情報が取り出せます
  - ◆ DIコンテナを使っている場合はMobyletクラスをDIできます



## ▶ mobylet (ver 0.1) にできること (2/5)

### ■ 絵文字関連

- 絵文字の表示が可能 (i絵文字、カスタムタグなどが利用可能)
- 絵文字の入力が可能 (フォーム入力)
- キャリア毎の絵文字の変換が可能



## ▶ mobylet (ver 0.1) にできること (3/5)

- バリュースタジオ社の端末情報をそのまま使える
  - 携帯の端末プロフィール情報は今まで有料のものしかありませんでしたが、バリュースタジオ社が無償公開を開始したため、mobyletはこれに乗っかり、デフォルトでバリュースタジオ社のプロフィール情報をサポートしています。
    - ◆ また、バリュースタジオ社とも直接交渉し、mobylet-core.jarにある時点（ちょっと古い）のプロフィール情報を同梱し、mobyletライブラリに内包する形での2次配布が可能となりました。
    - ◆ バリュースタジオ社のCEO北野さん、有難う御座います！
- I/Fを実装することにより他社のプロフィールも使える
  - XML形式のものなどにも対応することは可能です。

## ▶ mobylet (ver 0.1) にできること (4/5)

- ブラウザサイズ (横幅等) に応じたサイズの画像を表示できる
  - 端末毎に異なるサイズの画像を出し分けることができます。
    - ◆ `<m:img src="/img/demo.jpg" magniWidth="0.8" />`
    - ◆ 上記のように書くとdemo.jpgがブラウザ幅の0.8倍の大きさにFitして表示されるようになります。
    - ◆ この処理はFilter以下で制御しているため、APサーバ上のディレクトリに配置した画像のみ、対象とすることが可能です。
  - 但しAPサーバも結構負荷が掛かる (動的に処理するため)
  - 特にCPU/メモリが潤沢なシステム若しくはアクセスの軽いシステムで使うことをオススメします。
  - 画像変換部分だけに特化したアプリケーションを立てて、ASP形式にするのも良いかもしれません
  - JavaのImageAPIを使って制御しています。

## ▶ mobylet (ver 0.1) にできること (5/5)

### ■ GPSとGoogleMapのI/Fが標準搭載

- index.jsp

- ◆ `<m:gps kickBackUrl="gps.jsp" />`

- gps.jsp

- ◆ `<m:googlemap key="取得したキーを入れます" />`

- これで現在位置の地図を表示するシステムが完成です

### ■ GoogleMap Mobileを使う場合はキーが必要です

- GoogleMap Mobileのページからキーを発行して貰ってくださいね。

## ▶ フレームワークを一切使わない場合 (1/3)

- SVNにexampleがあるので使ってみてください。
  - mobylet-exampleというプロジェクトです
    - ◆ ちなみにこのプロジェクトは↓の設定がほぼ終了しています
      - 端末プロファイルだけは最新化した方が良くも
- 実際に設定すると次のような5ステップの設定が必要です。
  - mobylet-core / mobylet-taglibs (任意)をDLしてWebappに配備
    - ◆ maven2を使用している方はpom.xmlにdependencyを追加でOK
  - mobylet-charsetをDLして任意のディレクトリに配備
  - バリューエンジン社から最新の端末情報プロファイルをDLして配備
    - ◆ <http://valueengine.jp/index.html>
  - web.xmlにMobyletFilterを適用するよう設定を追加
  - JVMパラメータにcharsetのjarを読み込む設定を追加

- ▶ フレームワークを一切使わない場合 (2/3)
  - ではとりあえずみなさんの携帯で見てください
    - 画面ではFire Mobile Simulatorを使用します
  - URLは以下に書いておきますが。。。

<http://leihauoli.org/mobylet-example/>

- ▶ フレームワークを一切使わない場合 (3/3)
  - 面倒なのでQRでどうぞ。



## ▶ S2と組み合わせた設定について

- S2専用のmobylet-s2extensionsがあります。
- フレームワークを使わない設定に加えて次の設定を追加
  - mobylet-s2extensionsをDLしてWebappに配備
    - ◆ maven2を使用している方はpom.xmlにdependencyを追加でOK
  - web.xmlにS2MobyletFilterを適用するよう設定を追加
  - app.diconにmobylet.diconをinclude
- 一応SAStruts版のデモも作りました。
  - mobylet-example-sastrutsというプロジェクトでSVNにあります。



## ▶ ドキュメントの紹介

- 今のところ以下のサイトしかありません。。。
  - 携帯では見れません。。。
    - ◆ これから充実させて行きたい！

<http://mobylet.sandbox.seasar.org/>

## ▶ サポート & 開発体制

- 竹内 (stakeuchi) を中心に開発 & 品質向上しまくります。
  - 他に弊社の山口 (yamaguchi)、八柳 (mickey8) も参加しています。
- 質問や疑問はじゃんじゃんseasarのusersMLに下さい。
- MLはこっぱずかしいので個人的に...という方もOK
  - [s.takeuchi@leihauoli.com](mailto:s.takeuchi@leihauoli.com) へどうぞ！
- Blogも今日立ち上げてみました。
  - <http://blog.livedoor.jp/mobylet/>
    - ◆ ライブドアでmobyletで検索すると出るかな？

## ▶ mobyletの今後の夢

- mobylet-mailを作りたい。
- GAE (Google App Engine) に対応したい。
- 各種フレームワークのextensionを作りたい。
- ドキュメントを充実させたい
- チュートリアルを作りたい
- なんなら携帯ナレヅ集だって作りたい
- 他にも沢山、沢山。。。

夢は沢山あるけれど、皆様に使って貰える日を夢見て  
日々精進して参ります！

M ありがとうございました

ご清聴有難う御座いました！