Going Beyond SAP ITS Mobile Apps to a Responsive Design Mobile Apps

JK (JayaKumar Pedapudi) – Principal Consultant – NTT DATA, Inc.
Introduction.
- Learning Points.
- What is Responsive Design and its Role?
- Design Trends & Responsive Design Layouts.

Return on Investment (ROI).

SAP ITS Mobile Technology
- ITS Mobile Intro and Pre-requisites.
- Advantages, Disadvantages and Standard Services
- ITS Mobile - Template Generation Process.
- ITS Mobile Customization.

Responsive Design Techniques and Tools.

Applying Responsive Design to SAP ITS Mobile App.

Challenges & Best Practices.

Wrap Up & Q&A.
World’s sixth largest IT Service provider & systems integrator

Founded 1967 in Japan as the technology arm of the NTT Group

$15B in annual revenues

60,000 professionals operating in 35 countries

Serving 80 of the Global Top 100 organizations

Executing aggressive globalization strategy

Global delivery provides high quality around-the-clock services for 2,500 clients in all global regions with scalable services
Learning Points

- Learn about responsive design and how it can be applied to ITS Mobile Apps.

- Consistent user experience can be achieved using the responsive mobile design with cost effective way.

- Learn Tools, Frameworks, Technologies, Best practices, Challenges and Solutions for designing responsive mobile applications that can lower the maintenance cost.
Tablets and smartphones become our default choice for connecting to the internet.

There are several ways to approach a mobile web presence, such as a separate mobile site, or a mobile app. The most recent and increasingly popular approach is responsive web design.

Responsive web design is a relatively new approach to mobile web design. Responsive web design is all about keeping it simple.

Responsive Design is a methodology and tool to approach any web app design or mobile design.
The idea is to create one website to fit literally all screen sizes, be it a desktop or laptop screen, game console, e-reader, tablet or smartphone in landscape or portrait mode.

Responsive websites are designed with a flexible grid and uses media queries to determine the screen size for every individual site visit and rescale the content accordingly.

Responsive Web Designs offer the user an optimal viewing experience, no matter the device.

There are many web frameworks in the market that uses this Responsive Design techniques.
Designing the mobile apps with responsive design technique, which adopts itself based on different device form factor is a most cost effective way.

- One Codebase for all devices.
- It’s a low maintenance, single application supporting multiple devices.
- Brand consistency across all the screen layouts and different devices gives more user friendly look and feel experience.
- Responsive web design is highly user friendly.
ITS Mobile – Introduction

- ITSmobile is an application using the Internet Transaction Server (ITS) and it uses the ITS template technique.

- ITSmobile is delivered with a template generator that allows you to generate the templates of simple screens directly.
  
  A template generator allows you to easily generate templates of simple application screens for mobile devices of any screen form factor.

- You can then change these templates further to meet your requirements.

- You can also create a completely new template generator, or Create a new template generator based on the one delivered.
ITS Mobile – Pre-requisites

- To be able to generate templates, you require a functioning SAP Web AS - ABAP application.

- The screen layout of the application must be designed for mobile devices and their limited visualization functions.
  - The HTML template generator uses the maintenance size of the screens (number of rows / columns) to build an HTML.
  - The application should use only screen elements that are supported by ITSmobile.

- On the SAP Web AS ABAP, you require the following Service/Patch status:
  - SAP Web AS ABAP 620: Basis Support Package 64
  - SAP Web AS ABAP 640: Kernel Patch 161, Basis Support Package 20

- You also require an ITS standalone with the following patch status: ITS 6.20, Patch Level 28

Follow @ASUG365 and #ASUG2013 on Twitter
Advantages

- SAP ITS helps visualize ABAP applications on mobile devices easily.
- The application can be developed and tested entirely in ABAP.
- We can use SAP GUI for Windows to perform ABAP debugging.
- Generating templates ensures a fast initial visualization of the application in HTML.
- Consistent branding can be maintained across all the templates.
Disadvantages

- SAP ITSmobile is not supported on previous Support Packages than specified.
- Requires some basic understanding of SAP ITS application development process and the process of generating the HTML templates from ITS Dynpros.
- To develop more complex screen layouts & user interaction, you need to have working knowledge on HTML, CSS and JavaScript Technologies.
- If you want to develop a brand new template generator, then you’ll need to understand how the standard template generators are built.
ITS Mobile – Standard Services

There are two standard services delivered by SAP for generating simple screen templates.

*Services used as a basis for ITS mobile template generation.*

- **ITSMOBILE** - Standard Templates for runtime support
  (All generated application services obtain their JavaScript (file mobile.js) and their CSS stylesheets (file mobile.css) from this service.)

- **ITSGENMOBILE** - Generation Templates for Templates (Style "Mobile Devices")
- **ITSGENMOBILE4** - Generation Templates for Templates (Style "Mobile Devices 4")
- **ITSGENMOBILEXV** - Generation Templates for Language Support

*Services used for testing ITSmobile functions.*

- **ITSMOBILE00**
- **ITSMOBILE01**

**Caution:** ***Never attempt to modify any of these standard services***
ITS Mobile – Template generation process
ITS Mobile – Template generation process (Cont.)
ITS Mobile – Template generation process (Cont.)
ITS Mobile – Template generation process (Cont.)
ITS Mobile Customization

- No “Out of the box” solution, but a very adaptable and flexible framework.

- You can manually adjust individual templates in transaction SE80.

  “Remember to republish all the changed templates.”

- For a Responsive design templates, we will adapt or rewrite the template generator as needed based on the web development technology you’ve selected.

  - The choice of web development technology can be
    - As simple as plain HTML5, CSS3 and Java Scripts based or
    - It can be one of the finest web development frameworks available in the market to build more complex mobile applications which leverages the device specific resources depending on your requirement.
Word of Caution

- Never change the SAP delivered standard ITSmobile Services like ITSGENMOBILE or ITSMOBILE directly, since the generation itself and all generated services depends on these standard services.
- If you change these services, it is no longer possible to regenerate these delivered standard services.
- If you want to make changes, copy the templates to another theme and make the changes in that theme.
- Never use the existing application services/screens for generating or publishing unless you really making changes to that particular screen template.
- When generating make sure you choose the right template generator.
It is not a single piece of technology, but a set of techniques and ideas used to design and build the App.

http://d.alistapart.com/responsive-web-design/ex/ex-site-flexible.html

- Flexible Grid System
  - Fluid Grid.
  - Flexible Media.
  - CSS3 Media Queries
    - http://mediaqueri.es/
    - http://www.time.com/time/

- HTML5 & CSS3

- SAP ITS Technology
Fluid Grid

Instead of designing a layout based on rigid pixels or arbitrary percentage values, a fluid grid is more carefully designed in terms of proportions.

In order to calculate the proportions for each page element, you must divide the target element by its context.

\[
\text{Target / Context} = \text{Result}
\]

\[
300\text{px} / 960\text{px} = 31.25\%
\]
• CSS3 Media Queries
  • CSS3 media queries currently support many modern browsers.
  • These basically allow you to gather data about the site visitor and use it, to conditionally apply CSS styles.
  • If we wanted to apply some styling to mobile phones, our media query might look something like:

```css
@media screen and (max-width: 480px) {
  .content { float: left; }
  .social_icons {display: none; }

  // and so on...
}
```

• 320px
• 480px
• 600px
• 768px
• 900px
• 1200px
Responsive Design is a basically three step process.

Step #1 – Meta Tag

- Include this meta tag in the <head> of HTML page.
  
  ```html
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  ```

- Override simple styles with complex style based on screen width
  
  ```html
  <link rel="stylesheet" media="screen and (min-width:801px)" href="complex.css" title="complex layout"/>
  ```

- Internet Explorer 8 or older doesn't support media query.
  
  ```html
  <!--[if lt IE 9]>
  <script src="http://css3-mediaqueries-js.googlecode.com/svn/trunk/css3-mediaqueries.js"></script>
  <![endif]-->
  ```
Step #2 – HTML Structure

```html
/* Sample Structure of the Layout */

<div id="pagewrap">
  <header id="header">
    <hgroup>
      <h1 id="site-logo">Demo</h1>
      <h2 id="site-description">Site Description</h2>
    </hgroup>
    <nav>
      <ul id="main-nav">
        <li><a href="#">Home</a></li>
      </ul>
    </nav>
    <form id="searchform">
      <input type="search"/>
    </form>
  </header>
  <div id="content">
    <article class="post">
      Blog post
    </article>
  </div>
  <aside id="sidebar">
    <section class="widget">
      Widget.
    </section>
  </aside>
  <footer id="footer">
    Footer
  </footer>
</div>
```

CSS to reset the HTML5 elements (article, aside, figure, header, footer, etc.) to block element.

```
article, aside, details, figcaption, figure, footer, header, hgroup, menu, nav, section {
  display: block;
}
```
CSS to reset the HTML5 elements (article, aside, figure, header, footer, etc.) to block element.

```
article, aside, details, figcaption, figure, footer, header, hgroup, menu, nav, section {
    display: block;
}
```
Step #3 – Media Queries

```html
<link href="media-queries.css" rel="stylesheet" type="text/css" />

These set of rules will be in effect if the viewport width is 980px or less.

```css
/* for 980px or less */

@media screen and (max-width: 980px) {
  #pagewrap {
    width: 94%;
  }
  #content {
    width: 65%;
  }
  #sidebar {
    width: 30%;
  }
}
```

For viewport 700px or less, specify the #content and #sidebar to auto width and remove the float so they will display as full width.

```css
/* for 700px or less */

@media screen and (max-width: 700px) {
  #content {
    width: auto;
    float: none;
  }
  #sidebar {
    width: auto;
    float: none;
  }
}
```
Step #3 – Media Queries (cont.)

For 480px or less (mobile screen), reset the #header height to auto, change the h1 font size to 24px and hide the #sidebar.

/* for 480px or less */
@media screen and (max-width: 480px) {
  #header {
    height: auto;
  }
  #h1 {
    font-size: 24px;
  }
  #sidebar {
    display: none;
  }
}

/* Flexible Images */
img {
  max-width: 100%;
  height: auto;
  width: auto\9; /* ie8 */
}

/* Flexible Embedded Videos */
.video embed, .video object, .video iframe {
  width: 100%;
  height: auto;
}
How to apply these design techniques?

- Manually adjust individual templates in transaction SE80.

- Adapt or rewrite the template generator as needed based on the web development technology you’ve selected.

“Developing custom template generator is most cost effective approach”. 
Building ITS Mobile custom template generator

- Table **W3GENSTYLES** - W3: Generation Styles for HTML Templates

**YMOBIHTML5**

**ZCL_ITS_GEN_HTML5_MOBILE**

*For Mobile Responsive Design Style (HTML5)*

<table>
<thead>
<tr>
<th>STYLE</th>
<th>ABAP_CLASS</th>
<th>INFO</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td></td>
<td>Dummy, Business HTML implemented within program</td>
</tr>
<tr>
<td>C</td>
<td></td>
<td>Dummy, classic implemented within program</td>
</tr>
<tr>
<td>MOBILE</td>
<td>CL_ITS_GENERATE_HTML_MOBILE</td>
<td>For mobile device</td>
</tr>
<tr>
<td>MOBILE4</td>
<td>CL_ITS_GENERATE_HTML_MOBILE4</td>
<td>For mobile devices (without html table)</td>
</tr>
<tr>
<td>MOBILEXV</td>
<td>CL_ITS_GENERATE_HTML_XV_MOBILE</td>
<td>Mobile Geräte mit Spracheingabe</td>
</tr>
<tr>
<td>W</td>
<td></td>
<td>Dummy, Business WebGUI implemented within program</td>
</tr>
<tr>
<td>YMOBIHTML5</td>
<td>ZCL_ITS_GEN_HTML5_MOBILE</td>
<td>For Mobile Responsive Design Style (HTML5)</td>
</tr>
</tbody>
</table>
Building ITS Mobile custom template generator

- Class - ZCL_ITS_GEN_HTML5_MOBILE

CL_ITS_GENERATE_HTML_MOBILE4 as a Superclass
Building ITS Mobile custom template generator

- Class - **ZCL_ITC_GEN_HTML5_MOBILE** - Methods

![Class Builder: Change Class ZCL_ITC_GEN_HTML5_MOBILE](image_url)

Follow @ASUG365 and #ASUG2013 on Twitter
Building ITS Mobile custom template generator

**Method** - DESCRIPTION_TEXT

```
METHOD IF_ITS_GENERATE_TEMPLATE-DESCRIPTION_TEXT.

* CALL METHOD SUPER->IF_ITS_GENERATE_TEMPLATE-DESCRIPTION_TEXT.
  
  + EXPORTING
  *  ) = PI_STYLE
  + RECEIVING
  * PE_DESCRIPTION = PE_DESCRIPTION
  +

  CASE pi_style:
  WHEN 'MOBILE1'.
    pe_description = 'Mobile Devices (Without HTML Tables) (001).
  WHEN 'MOBILE1_IF'.
    pe_description = 'Mobile Devices (4), Older Internet Expl.' (003).
  WHEN 'MOBIHTML'.
    pe_description = 'For Mobile Responsive Design Style (HTML5) (004).
  WHEN OTHERS.
    pe_description = 'Unknown style'(002).
  END_CASE.
  
ENDMETHOD.
```
Building ITS Mobile custom template generator

- **Method - CONSTRUCTOR**

```java
METHOD constructor.

DATA:
  l_theme TYPE _t_theme_for_templates.

super->constructor( ).

_html_width_factor_containers = '1.00'.
_html_width_factor_elements = '0.82'.

_loop_line_name = 'LOOP_INDEX'.

l_theme-service = 'YITSMGENHTML5'. "For Mobile Responsive Design Style (HTML5)

l_theme-theme = '99'.

INSERT l_theme INTO _themes_for_templates INDEX 1.

ENDMETHOD.
```
Building ITS Mobile custom template generator

- Internet Service – YITSMGENHTML5

- Copy from standard Internet Service – ITSGENMOBILE
<!DOCTYPE html>
<html>
<head>
<title>Demo</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="http://code.jquery.com/mobile/1.2.1/jquery.mobile-1.2.1.min.css" />
<script src="http://code.jquery.com/jquery-1.8.3.min.js"></script>
<script src="http://code.jquery.com/mobile/1.2.1/jquery.mobile-1.2.1.min.js"></script>
</head>
<body>
<div data-role="page">
  <div data-role="header">
    <h1>ASUG Annual Conference</h1>
  </div><!-- /header -->

  <div data-role="content">
    <p>Hello world</p>
  </div><!-- /content -->
</div><!-- /page -->
</body>
</html>
<!DOCTYPE html>
<html class="MobileHtml">
    if (~itsmobileNameSpace != "")
        ~current_service = ~itsmobileNameSpace & ~service;
    elseif (strsub(~sources, 0, 1) == "/")
        ~current_service = strsub(~sources, 0, strchr(~sources, "/", 1)+1) & ~service;
    else
        ~current_service = ~service;
    end;

</html>
Applying Responsive Design to SAP ITS Mobile App.
Applying Responsive Design to SAP ITS Mobile App.

```
if (-itsmobileJsInclude != "") {  <!-- customer include specified in gui settings -->
    <script type="text/javascript" language="javascript" src="mimeType(-service=-current_service)">
    else {
        <script type="text/javascript" language="javascript" src="mimeType(-service=itsmobile_service)"
    }
end;
if (-itsmobileCustomJsInclude != "") {
    <script type="text/javascript" language="javascript" src="mimeType(-service=-current_service)"
    end;

    <script type="text/javascript" language="JavaScript">
        var itsmobile_eos = "\wgateurl(-okcode="/nex")";
    </script>

    <script language="JavaScript1.2">
        include(-service="ITSMOBILE", -theme="99", -language="", -name="_javascript.html")

        function selectItem(itemno)
        {  setValue('elements[\dg_selidx[1]\']', itemno);
            submitForm('SEL');
        }
    </script>

    <script src="http://code.jquery.com/jquery-1.8.3.min.js"></script>
    <script src="http://code.jquery.com/mobile/1.2.1/jquery.mobile-1.2.1.min.js"></script>

    <!-- START script to support SAP dynamic search -->
    <script type="text/javascript" src="sapsearch.js"></script>
    <!-- END script to support SAP dynamic search -->
    <script type="text/javascript" src="accordianscript.js"></script>
```
Applying Responsive Design to SAP ITS Mobile App.

```html
<!-- main screen begin -->
<div width="100%">
    if (~itsmobileCuaInclude != "") <!-- customers cua area -->
        include(~service=~current_service, ~language="", ~theme="",
    elseif (~itsmobileNoCuaInclude != "1" && ~itsmobileNoCuaInclude
        include(~service="ITSMOBILE", ~language="", ~theme="99",
    end;
</div>
<div data-role="page" id="home">
</div>
</div>
<!-- main screen end -->
Applying Responsive Design to SAP ITS Mobile App.
Applying Responsive Design to SAP ITS Mobile App.
Applying Responsive Design to SAP ITS Mobile App.

**Existing ITS Mobile App**

- Warehouse Menu
  - Put-Away
  - Stock Transfer
  - Inventory Lookup
  - Inventory Count
  - My Profile
  - Logoff

**Responsive Designed ITS Mobile App**

- Warehouse Menu
  - Put-Away
  - Stock Transfer
  - Inventory Lookup
  - Inventory Count
  - My Profile
  - Logoff
Applying Responsive Design to SAP ITS Mobile App.

Existing ITS Mobile App

Responsive Designed ITS Mobile App
Applying Responsive Design to SAP ITS Mobile App.

Existing ITS Mobile App

Responsive Designed ITS Mobile App

Follow @ASUG365 and #ASUG2013 on Twitter
Generating screen for every small changes was tedious and lot of effort. After lot of research, we found a transaction that help do this job easily.

Transaction Code: SIAC_REGENERATE_TEMP. (Use with caution)

During the development, we faced issues publishing the new changes to the Internet Service each time the layout was changed. It was always picking the cached version of the template instead of the newly published version.

We developed a custom program to reset the MIMES Objects.
Best Practices

- Design the transaction screens to fit the mobile device screen size by following design standards.
- Keep the screen layout as simple as possible for flexibility.
- Always maintain a separate resource files for CSS, JavaScript, etc.
- Use percentage (%) value to make the containers fluid.
- `css3-mediaqueries.js` is required to enable media queries for browsers that don't support media queries.
- Use CSS to override the layout structure based on the viewport width.
- Use `max-width:100%` and `height:auto` to make the images flexible.
- Use `width:100%` and `height:auto` to make the embedded videos flexible.
- Use `-webkit-text-size-adjust:none` to disable text size adjust on iPhone.
- Responsive Design, Different tools, frameworks and technologies used for this design technique.

- Building custom template generator for high usability, lower maintenance cost and brand consistency.

- Adopting these techniques to ITS Mobile App for richer user experience with high user friendliness, consistent branding, and cost effective.

- Challenges, Solutions and Best Practices.
Presented By: JK ( JayaKumar Pedapudi )

Email: JayaKumar.Pedapudi@nttdata.com

Further Info: Pat Gray – Director, Marketing

Email: Pat.Gray@nttdata.com

Visit us in Booth: #448
THANK YOU FOR PARTICIPATING

Please provide feedback on this session by completing a short survey via the event mobile application.

SESSION CODE: 0614

For ongoing education on this area of focus, visit www.ASUG.com
Follow @ASUG365 and ASUG CEO Bridgette Chambers @BChambersASUG on Twitter to keep up to date with everything at ASUG.

Follow the ASUGNews team of Tom Wailgum: @twailgum and Courtney Bjorlin: @cbjorlin for all things SAP.