CMT 314: MOBILE APPLICATION DEVELOPMENT

Lecture Five Notes

MOBILE APPLICATION MEDIUM TYPES

5.1 INTRODUCTION

The ***mobile application medium type*** is the *type of application framework or mobile technology that presents content or information to the user*. It is a technical approach regarding which type of medium to use; this decision is determined by the *impact* it will have on the user experience. The *technical capabilities* and *capacity* of the publisher also factor into which approach to take.

Figure 6-1 illustrates the spectrum of mobile media; it starts with the basic text-based experiences and moves on to the more immersive experiences.

Figure 6-1. Multiple mobile application medium types

5.1.1 SMS

The most basic mobile application you can create is an SMS application. Although it might seem odd to consider text messages applications, they are nonetheless a designed experience. Given the

ubiquity of devices that support SMS, these applications can be useful tools when integrated with other mobile application types.

Typically, the user sends a single keyword to a five-digit short code in order to return information or a link to premium content. For example, sending the keyword “freebie” to a hypothetical short code “12345” might return a text message with a coupon code that could be redeemed at a retail location, or it could include a link to a free ringtone.

SMS applications can be both “free,” meaning that there is no additional charge beyond the text message fees an operator charges, or “premium,” meaning that you are charged an additional fee in exchange for access to premium content.

The SMS-to-Billboard that BBC World News put up in Midtown Manhattan.

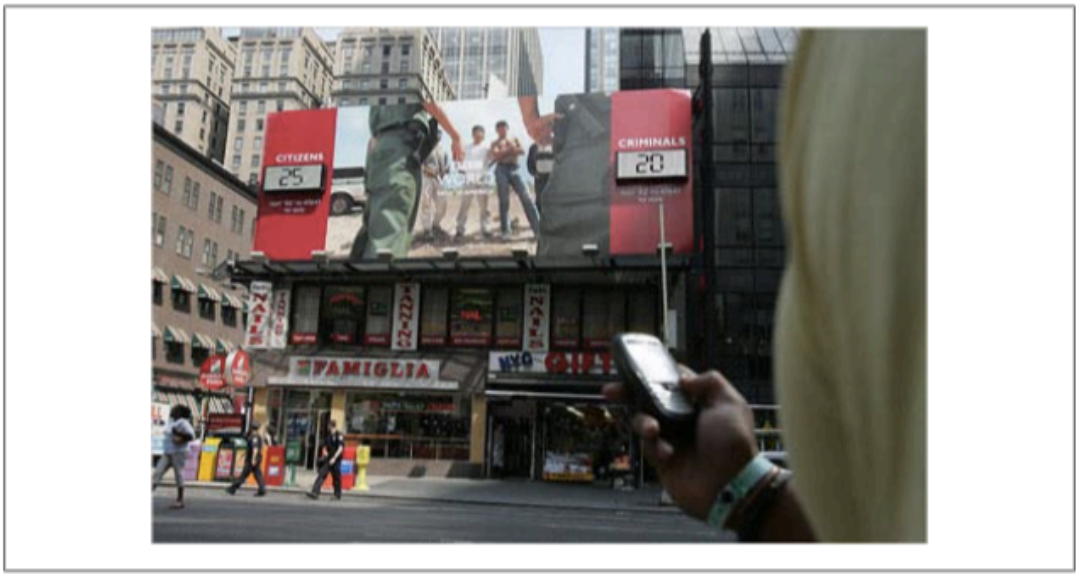

Figure 6-2. An SMS application to interact with a billboard in Manhattan

Pros

The pros of SMS applications include:

- They work on any mobile device nearly instantaneously.
- They’re useful for sending timely alerts to the user.
- They can be incorporated into any web or mobile application.
- They can be simple to set up and manage.

Cons

The cons of SMS applications include:

- They're limited to 160 characters.
- They provide a limited text-based experience.
- They can be very expensive.

5.1.2 Mobile Websites

As you might expect, a mobile website is a website designed specifically for mobile devices, not to be confused with viewing a site made for desktop browsers on a mobile browser. Mobile websites are characterized by their simple "drill-down" architecture, or the simple presentation of navigation links that take you to a page a level deeper as shown in Figure 6-3.

Figure 6-3. An example of a mobile website

Mobile websites often have a simple design and are typically informational in nature, offering few - if any - of the interactive elements you might expect from a desktop site. Mobile websites have made up the majority of what we consider the mobile web for the past decade, starting with the early WML-based sites (not much more than a list of links) and moving to today's websites, with a richer experience that more closely resembles the visual aesthetic users have come to expect with web content.

Though mobile websites are fairly easy to create, they fail to display consistently across multiple mobile browsers—a trait common to all mobile web mediums. The mobile web has been gradually increasing in usage over the years in most major markets, but the limited experience offered little incentive to the user. Many compare the mobile web to a 10-year-old version of the Web: slow, expensive to use, and not much to look at.

As better mobile browsers started being introduced to device platforms like the iPhone and Android, the quality of mobile websites began to improve dramatically, and with it, usage improved.

Pros

The pros of mobile websites are:

- They are easy to create, maintain, and publish.
- Nearly all mobile devices can view mobile websites.

Cons

The cons of mobile websites are:

- They can be difficult to support across multiple devices.
- They offer users a limited experience.
- Most mobile websites are simply desktop content reformatted for mobile devices.
- They can load pages slowly, due to network latency.

5.1.3 Mobile Widgets

Mobile web widgets are small web applications that can't run by themselves; they need to be executed on top of something else.

Opera Widgets, Nokia Web RunTime (WRT), Yahoo! Blueprint, and Adobe Flash Lite are all examples of widget platforms that work on a number of mobile handsets.

Pros

The pros of mobile web widgets are:

- They are easy to create, using basic HTML, CSS, and JavaScript knowledge.
- They can be simple to deploy across multiple handsets.
- They offer an improved user experience and a richer design, tapping into device features and offline use.

Cons

The cons of mobile web widgets are:

- They typically require a compatible widget platform to be installed on the device.
- They cannot run in any mobile web browser.
- They require learning additional proprietary, non-web-standard techniques.

5.1.4 Mobile Web Applications

Mobile web applications are mobile applications that do not need to be installed or compiled on the target device. Using XHTML, CSS, and JavaScript, they are able to provide an application-like experience to the end user while running in any mobile web browser.

Web applications allow users to interact with content in real time, where a click or touch performs an action within the current view.

The Web 2.0 movement brought user-centered design principles to the desktop web, and those same principles were sorely needed in the mobile web space as well.

The challenge, as always, was device fragmentation. The mobile browsers were years behind the desktop browsers by then, making it nearly impossible for a mobile device to render a comparable experience.

Figure 6-5. The Facebook mobile web app

To make matters worse, the perceived market demand for mobile web applications was not seen as a priority with many operators and device makers. It was the classic chicken- or-the-egg scenario. What had to come first, market demand to drive browser innovation or optimized content to drive the market?

With the introduction of the first iPhone, we saw a cataclysmic change across the board. Developers quickly got on board, creating mobile web applications optimized mostly for the iPhone. The combination of a high-profile device with an incredibly powerful mobile web browser and a quickly increasing catalog of nicely optimized experiences created the perfect storm the community had been waiting for.

Usage of the mobile web exploded with not just users of the iPhone, but users of other handsets, too. Operators and device makers saw that consumers wanted not just the mobile web on their handsets, but the regular Web, too. There was a sudden strong unilateral move by all operators and devices makers to put better mobile web browsers in their phones that could leverage this new application medium.

Pros

The pros of mobile web applications are:

- They are easy to create, using basic HTML, CSS, and JavaScript knowledge.
- They are simple to deploy across multiple handsets.
- They offer a better user experience and a rich design, tapping into device features.
- Content is accessible on any mobile web browser.

Cons

The cons of mobile web applications are:

- The optimal experience might not be available on all handsets.
- They can be challenging (but not impossible) to support across multiple devices.
- They don't always support native application features, like offline mode, location lookup, filesystem access, camera, and so on.

5.1.5 Native Applications

The next mobile application medium is the oldest and the most common; it is referred to as native applications, which is actually a *misnomer* because a mobile web app or mobile web widget can target the native features of the device as well. These applications actually should be called "platform applications," as they have to be developed and compiled for each mobile platform.
These native or platform applications are built specifically for devices that run the platform in question.

In the smartphone space, the platform SDKs get much more specific. Although many smartphones are also powered by Java, an operating system layer and APIs added to allow developers to more easily offload complex tasks to the API instead of writing methods from scratch. In addition to Java, other smartphone programming languages include versions of C, C++, and Objective-C

Figure 6-6. A native application in the iPhone

Creating a platform application means deciding which devices to target, having a means of testing and certification, and a method to distribute the application to users. The vast majority of platform applications are certified, sold, and distributed either through an operator portal or an app store. It is possible to create an application and publish it for free on the Web, but it is rarely done.

Because platform applications sit on top of the platform layer, they can tap into the majority of the device features, working online or

offline, accessing the location and the filesystem - and if there's camera on the device, then you can probably do something with it as well. Hence the need for certification before the application is distributed, to ensure that no one distributes an application that steals a user's personal data or maliciously uses the device to spread viruses.

Pros

The pros of native applications include:

- They offer a best-in-class user experience, offering a rich design and tapping into device features and offline use.
- They are relatively simple to develop for a single platform.
- You can charge for applications.

Cons

The cons of native applications include:

- They cannot be easily ported to other mobile platforms.
- Developing, testing, and supporting multiple device platforms is incredibly costly.
- They require certification and distribution from a third party that you have no control over.
- They require you to share revenue with the one or more third parties.

5.1.6 Games

The final mobile medium is games, the most popular of all media available to mobile devices. Technically games are really just native applications that use the similar platform SDKs to create immersive experiences. We treat them differently from native applications for two reasons: they cannot be easily duplicated with web technologies, and porting them to multiple mobile platforms is a bit easier than typical platform-based applications.

Although you can do many things with a powerful mobile web browser, creating an immersive gaming experience is not one of them. The reason games are relatively easy to port ("relatively" being the key word), is that the bulk of the gaming experience is in the graphics and actually uses very little of the device APIs.

The game mechanics are the only thing that needs to adapt to the various platforms. Like in console gaming, there are a great number

of mobile game porting shops that can quickly take a game written in one language and port it to another.

Figure 6-7. An example game for the iPhone

Pros

The pros of game applications are:

- They provide a simple and easy way to create an immersive experience.
- They can be ported to multiple devices relatively easily.

Cons

The cons of game applications are:

- They can be costly to develop as an original game title.
- They cannot easily be ported to the mobile web.

5.2 Mobile Application Media Matrix

In summary, to aid in comparing and contrasting which of these mobile application media is best for your mobile product, we've placed them into a matrix.

Table 6-1. Mobile application media matrix

	Device support	Complexity	User experience	Language	Offline support	Device features
SMS	All	Simple	Limited	N/A	No	None
Mobile websites	All	Simple	Limited	HTML	No	None
Mobile web widgets	Some	Medium	Great	HTML	Limited	Limited
Mobile web applications	Some	Medium	Great	HTML, CSS, JavaScript	Limited	Limited
Native applications	All	Complex	Excellent	Various	Yes	Yes
Games	All	Complex	Excellent	Various	Yes	Yes