Carnegie Mellon Silicon Valley Software Engineering Technical-Track 2013 Summer Practicum: Project Proposal

2013 Mobile App Performance Challenge: Native vs. HTML5 Hybrid Apps

Appception, Inc. (http://www.appception.com)
Carl Stehle (650.938.8046., carl@appception.com)
March 22, 2012

1. Title

2013 Mobile App Performance Challenge: Native vs. HTML5 Hybrid Apps

2. Abstract

While it is no mystery that mobile computing is one of the highest-growth areas in the software industry, a lesser known component of this is the emergence of so-called HTML5 hybrid apps, which industry analysts at Gartner estimate will comprise over 50% of the app market by 2016 (http://www.gartner.com/newsroom/id/2324917).

But in this rapidly changing industry, HTML5 hybrid apps must appear and perform competitively with native apps in order to gain market share.

The purpose of this project is to benchmark, analyze, and compare the HTML5 hybrid approach with native to assess competitiveness in visual appearance and performance, and to forecast future trends.

A secondary goal is to evaluate the impact that recent technologies (e.g. GPUs, software libraries, standards) have or will have on hybrid/native comparisons.

This project is multi-disciplinary; it involves both software technology and market research.

3. Background of Sponsoring Organization

Appception is a Silicon Valley start-up that builds tools to help developers create mobile apps. The Appception IDE is a cloud-based development system for creating and testing HTML5 multi-platform mobile apps using a web browser without installing any software.

Carl Stehle is the Founder of the company, and one of its principal developers. He has a long history of software development, specializing in networks and embedded systems, and entrepreneurship, spanning over 30 years.

4. Project Proposal Details

a. Background and problem context

HTML5 is the 'language of the web' and runs almost universally across desktop and laptop web browsers. Adoption across mobile devices (specifically within mobile apps), however, is lacking and mainly constrained by a widespread perception of sluggish performance (or responsiveness).

Mobile app developers have therefore favoured native languages such as Java, C/C++, Objective-C, and others in order to optimize user interface (UI) experiences. This results in duplication of app development effort and testing across mobile platforms, and inhibits porting of existing web applications onto mobile devices.

The performance issue is clouded by several factors: poor design practices and implementations of HTML5 apps, unoptimized HTML5 UI component libraries, behaviour of HTML5 on older devices without hardware graphics acceleration (GPUs), and inconsistent mobile platform support for HTML5.

Nearly all of the above issues can be overcome on recent generations of mobile devices with modern programming practices; however, in order to alter the currently poor perception of HTML5 on mobiles within the developer community, strong evidence of actual performance is needed, including convincing "A-B" comparisons with native apps.

b. Overview of proposed project

The project seeks to dispel myths of poor HTML5 performance on mobiles by two methods: first, by comprehensively benchmarking a set of HTML5 hybrid apps containing leading HTML5 UI libraries against comparable native apps; and second, by challenging developers to identify which of two apps with similar visuals and behaviour is native, and which is HTML5 hybrid.

c. High level description of project and scope

The project consists of two main components, and depending on available time, an optional follow-on task.

The first component is to identify and benchmark a set of common UI operations on HTML5 hybrid and native mobile apps. Since there are multiple competing UI frameworks for HTML5, several leading UI libraries will be used for the HTML5 apps. Standard vendor libraries will be used for native apps.

The second component is to create two nearly identical demonstration apps, one using HTML5 and the other written in native languages. These apps will be shown to developers in a survey designed to measure the ability of developers to correctly identify the HTML5 or native implementation.

An *optional* follow-on task would be to assess the impact of recent technologies on the hybrid/native debate, e.g. the famo.us UI which boasts native-level performance with standard HTML (no 'canvas' tag, no OpenGL); or WebGL (available on BlackBerry 10 devices).

d. List major project goals, sub-goals, and objectives

Benchmarking goals include specifying a canonical set of primitive and composite UI operations which can be objectively measured on both HTML5 hybrid and native apps, and performing the measurements on devices of different form factors (i.e. phones and tablets) and different operating environments (i.e. iOS and Android, at least) and across multiple OS release versions.

The UI operations must be chosen to cover the anticipated set of normal mobile user interactions, and place any notable variances into the appropriate context.

Use of multiple device form factors will show the effects, if any, of hardware designs on app performance; tablets, being larger, may be expected to have higher performance than phones.

The advances in mobile devices have not been uniform across vendors or operating environment releases. A representative set of units will be chosen for both vendor comparison and evolution of the hardware and software operating environments.

Goals for the demonstration apps are more subjective: a survey of developers will be undertaken to measure if one app implementation or another is considered superior, or if both are equivalent.

Measuring the impact of recent technologies will of course depend on available releases, but will seek to identify important future trends (a key skill for technology market researchers).

e. Technologies/skills expected and required
Team members are expected to have experience with HTML5 and CSS3. Some knowledge of basic graphics rendering, and a good understanding of algorithm design and performance tradeoffs is assumed. Knowledge of browser rendering internals (e.g. WebKit or Gecko) would be a plus.

App development experience (native or HTML5-based) is not required, but would also be useful.

f. Expected team size
Team size may range from 2-4 members. Less than 2 will likely not be able to accomplish the goals in the available time. More than 4 will require that the scope of the project be expanded; this can certainly be accommodated, but is not required.

g. Currently known obstacles
Mobile device GPUs are generally opaque; some assumptions about internal processing must either be experimentally verified or otherwise determined. Also, the set of tests to measure performance must be sufficiently robust so that no major use cases are missed.

Creating convincing demonstration apps is challenging for different reasons; perception is subjective and must be understood as such, and developers may be resistant to using HTML5 for various reasons; evidence of its superiority must be seen as overwhelming.

h. Currently known risks
The primary risks of the benchmarking are misidentifying important UI operations, incomplete or incorrect test definitions, or poor testing methodology or measurements.

The primary risks of the demonstration apps are missing certain subjective criteria which developers may consider important.

i. Team roles/responsibility
The team should appoint a 'team leader'. This is not necessarily the most senior developer, but should be someone who can coordinate the tasks and represent the interests of the entire team, particularly as they relate to demonstrating progress and meeting the specifications and schedule.

j. Expected use of deliverables at project completion
The deliverables are anticipated to be used by Appception for educational and marketing purposes, and for publishing and presentation by the team at conferences, 'meetups', or other suitable venues.

k. Preliminary project roadmap
The client will first communicate and discuss the issues in detail with the team. Then, the team will choose a leader and assign tasks to members.

Next, the basic outline of the benchmarking tasks will be defined. Some preliminary work may be done before finalizing task definitions.

Planning for demonstration apps may be done somewhat in parallel, but may change depending on benchmarking results.

Research in recent technologies will commence in the background and, based on progress, may be scheduled after demonstration apps are completed.

l. Criteria for measures of success

Success will be measured by the effectiveness of the benchmarking methodology, testing work, and demonstrations to produce comprehensive, consistent, and convincing results.

A high degree of confidence in forecasting future trends by the team will also be deemed as successful.

5. Issues and Constraints

NDA: N/A
Citizenship: N/A
IP: N/A
Locality: N/A