PathCase for Tablet Computers

Steve Johnson

Case Western Reserve University

January 24, 2011

Abstract

PathCase on tablets is important because it will allow clinicians to easily explore possible diagnoses. It also provides a testbed for graph interfaces and algorithms which I'm sure I will invent and implement before the end of 2011.

1 Introduction and Background

What is PathCase? ¡Explanation of PathCase¿

2 Possible Approaches to Tablet Implemenation

2.1 iPad SDK

The most obvious path to implementing the PathCase pathway viewer on a tablet is to implement an iPad app using Objective-C and the iOS SDK. This method would provide fast execution, a native interface, App Store distribution, and the ability to render high-quality graphics.

The development time of this approach would be reasonable, though other approaches might have faster development time. It is not free to develop on the iPad ($100 per year),

but this cost is not prohibitive. The resulting application would only work on the iPad.

2.2   Android SDK

Another way would be to use the Android SDK and target Android-based tablets. This approach would have most of the same advantages as developing for the iPad, but Android tablets currently on the market have not been well received, and the author is not familiar with the SDK.

2.3   Javascript

The most portable way to implement the application would be to use Javascript APIs that have been standardized recently ("HTML5"). This approach would have a relatively rapid rate of development. The application would run on all Android and iOS devices, both in web browsers and in packagese made for the respective app stores.

However, there may be memory limit issues due to the harsh memory constraints of current tablets. Memory usage is easier to control using the native SDKs. Also, the interface would be less sophisticated due to the immature state of Javascript GUI libraries.

3   Literature Survey

4   Work To Date

Efforts so far have been put into the graph view and fixing various issues with the iPad development environment.
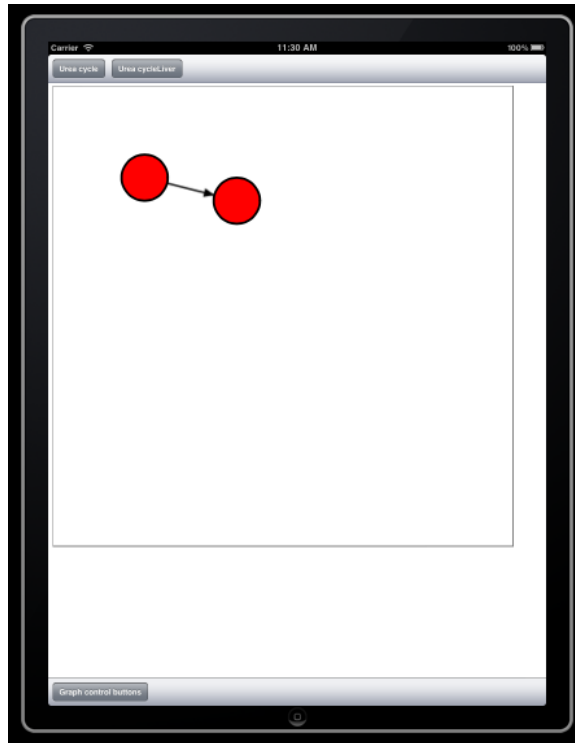
Figure 1: Basic interface

5　Results

6　Future Work

I am working with Xinjian to make a web API that can be used easily with iPad libraries.
In the meantime, I have a local server written in Python to serve sample JSON data sets.
This sample server is being used to test code for HTTP requests and JSON deserialization.

7　Conclusion