ASSIGNMENT 9 - Automating Recovery After Failure

Graded as Pass / Fail, Individual Work
Due: Week 12, Submit during lab class

Objectives

1. Learn to automatically recover failures on behalf of the user when possible.

Extension Tasks

Task 1 - Let's Try That Again

When a user is downloading a file and loses their network connection in the middle of the transfer, they expect to be able to resume their transfer. Better yet, the application can automatically resume the transfer when they are reconnected!

- Create an application that downloads a content file from an HTTP server when the user presses a button. The file should be in excess of 10 MB (the exact size is not important for this task). Use either NSURLSession or a third-party networking library such as AFNetworking, ASIHTTPRequest, MKNetworkKit or RESTKit (these are third-party libraries, not Apple APIs) to download the file asynchronously, and use a progress view to display the download's completion percentage.
- When the application is closed in the middle of a transfer (e.g. the home button is pressed), or when network connectivity disappears

(such as switching off the Wi-Fi on your development computer), save the state of your transfer so that you can retry later, keeping the part of the file downloaded so far so that when you retry you don't have to download that part again (not all APIs support this so you may have to choose one that does or implement the capability yourself).

- If the application has an unfinished download left over, it should periodically poll the network to determine if the network is reachable. If so, it should attempt to resume the transfer.

Task 2 - Experience Report on Failure Recovery

Write a report (1500 words) about your design and experience completing the task. Specifically, rationalise your design from the perspective of the user: why is this capability necessary/important in a mobile data-driven application? Your report must contain the following sections:

1. Introduction (What are the key design principles that you will discuss in this report?)
2. Optimisation (What problems does your design solve?)
3. Drawbacks (Were there any problems with your design or implementation? How were the problems corrected?)
4. References (The external sources you used in preparing your report.)
5. Appendix (Commented code snippets as needed to support the report.)

Note: Each code snippet counts for 200 words and must have a caption.

Useful References

AFNetworking Home Page:

http://afnetworking.com

ASIHTTPRequest Home Page:

http://allseeing-i.com/ASIHTTPRequest/

MKNetworkKit Home Page:

http://blog.mugunthkumar.com/products/ios-framework-introducing-mknetworkkit

RESTKit Home Page:

http://restkit.org

Apple's Reachability Sample Project:

http://developer.apple.com/library/ios/#samplecode/Reachability/Introduction/Intro.html

Core/Extension Tasks

All tasks in this assignment are “extension”.

Submission

You are required to submit a printed report that adheres to the following:

- The header or footer must contain your name, student ID, and unit code.
- The document must have a title.
- Evidence that shows you completed each task must be presented in a separate section.
- The document should NOT have a table of contexts or cover page.

Demonstration

You may be asked to demonstrate your assignment in the lab. You should be able to do this and explain your code when asked in the lab session.