MOBILE DEVELOPMENT
NETWORKING: PART 1

Rudd Taylor
Founder, SALT

LEARNING OBJECTIVES

‣ Recognize the benefits of JSON and REST

‣ Identify the benefits of using NSURLSession

‣ Evaluate and parse data returned in JSON format

‣ Integrate cocoa pods into our projects and use SwiftyJSON for better parsing

REVIEW OF PERSISTENCE

INTRO TO NETWORKING

SOME TERMINOLOGY

‣ Request: The concept of what is sent to some remote server somewhere. Has a method, may include parameters

   ‣ E.g. Get http://google.com

      ‣ Get the contents of google.com

‣ Response: The response that the web server gives back. Includes a status code and, possibly, content

‣ HTTP: A very common protocol for making network requests

‣ JSON: A way we format data to make it machine readable

   ‣ e.g. {"name": "rudd"} is a simple blob of JSON

NETWORKING

WHAT CAN GO WRONG?

WHAT CAN GO WRONG?

‣ No network connection (client-side)

‣ Destination offline (server-side)

‣ The network can be slow

‣ The request can not be authorized

‣ Lost network connection mid-request/response

‣ Lots more…

HANDLE ERRORS

‣ If you make network connections in your app, it's a best practice to handle errors when possible, and communicate errors to the customer when not

BEST PRACTICES

‣ Users get charged for data: Use only what you must

    ‣ They do not share a filesystem

‣ Allow users to cancel lengthy downloads

‣ When something fails, either handle it or message the user

‣ Cache when needed (this is a little more advanced)

NETWORKING

IOS NETWORKING

NSURLSESSION

‣ A shared context in which our requests are made

  ‣ Configurable way to share state among connections

    ‣ e.g. "All my connections should use a cookie…"

  ‣ Can queue up several types of connections

    ‣ Download: For downloading files

    ‣ Upload: For uploading files

    ‣ Data: For small bits of data transfer, what we'll use today

NSURLSESSION SYNTAX

```
if let url = NSURL(string: "http://somewebsite.com/params") {
    let task = NSURLSession.sharedSession().dataTaskWithURL(url,
completionHandler: { (data, response, error) -> Void in
        println(response) ^ Closure starts here
        println(data)
        println(error)
        var stringOfData = NSString(data: data, encoding:
NSUTF8StringEncoding)
         println(stringOfData)
    })
    task.resume()
}
```

NETWORKING CODE-ALONG

GROUP ASSIGNMENT

‣ Add the display of networking errors to this project.

‣ Display an error message to the user, along with the error that gets returned from the call.

‣ Bonus: Display the HTML content of the request in a web view

NETWORKING

JSON

JSON

‣ JSON is a data-interchange format

  ‣ i.e. A way of formatting text so it's easily machine-readable and network-transferable

‣ Similar to the Plists we've been working with, it can store:

  ‣ Arrays

  ‣ Dictionaries

  ‣ Strings

  ‣ Numbers

  ‣ And any combination of the above

JSON

‣ Does not specify anything about formatting, like HTML does

JSON EXAMPLES

‣ {"name": "rudd"} // A 'dictionary'-esque data structure

‣ ["rudd", "travis", "sean"] // An array of strings

‣ {"response": "ok", content: {message: "hello world!"}} // A complex structure

‣ [1, 2, "rudd"] // Note that JSON does not specify type information

JSON CODE WALKTHROUGH

GROUP ASSIGNMENT

‣ Change the app to display recent Mashable headlines, using:

   ‣ http://mashable.com/stories.json

‣ Bonus 1: Display author in home screen table view

‣ Bonus 2: Print the titles along with the webview

‣ Bonus 3: Make larger table view cells that display the content of the article, in text format