

Übungsblatt 3

Ziele

- iOS Human Interface Guidelines kennenlernen
- Verstehen fremden Programmcodes
- Verstehen von Applikationsarchitekturen
- Die Location API kennenlernen

Aufgabe 1

Die „iOS Human Interface Guidelines“ [1] geben grundlegende Empfehlungen zur Entwicklung von iOS-Anwendungen und legen verbindliche Regeln für die Verwendung der Benutzungsschnittstellen-Elemente fest.

- Lesen Sie die Kapitel 2 und 3 der „iOS Human Interface Guidelines“.
- Schauen Sie sich exemplarisch die Details der Beschreibung eines Benutzungsschnittstellen-Elementes in Kapitel 7 an.
- (für Master-Studierende) Lesen Sie in Kapitel 5 die Seiten 47 bis 65.

[1] <http://developer.apple.com/library/ios/documentation/userexperience/conceptual/mobilehig/MobileHIG.pdf>

Aufgabe 2

Auf der Webseite der Vorlesung findet sich eine an iOS5 angepasste Version der „LocateMe“ [2] Beispielanwendung. Laden Sie die iOS5-Version von der Webseite herunter und öffnen Sie sie in Xcode 4.2. Stellen Sie das „Active Scheme“ um auf „iPhone 5.0 Simulator“. Der iOS5-Simulator erlaubt die Eingabe beliebiger GPS-Koordinaten (im Simulator Debug | Location | Custom Location...).

- Analysieren Sie zunächst das Storyboard. Vergleichen Sie die Struktur (besonders des Tab Bar Controllers) mit der Struktur Ihrer Tab-Bar-Anwendung aus der letzten Übung. Worin bestehen die Unterschiede und was wird durch die Struktur in „LocateMe“ zusätzlich ermöglicht?
- Begründen Sie, warum es nicht notwendig ist, dass die Start-Buttons in den Szenen „Get Location“ und „Track Location“ nicht mit IBAction-Methoden verbunden sind. Erklären Sie grob die Schritte, die das Programm nach Drücken des Start-Buttons in der „Get Location“ Szene ausführt.
- Machen Sie sich mit der Funktionsweise des „Picker Controls“ in der Szene „Setup View Controller“ vertraut und erklären Sie die grundlegende Funktionsweise (Woher kommen die Daten, die im Picker zu sehen sind? Was bewirkt die Interaktion des Benutzers?). Sie müssen nicht alle Funktionen des „Picker Controls“ erklären, sondern nur die, die in der Controller-Klasse (SetupViewController.{h,m}) verwendet werden. Wozu wird das Dictionary „setupInfo“ in der Klasse SetupViewController

verwendet? Wie werden die Daten vom SetupViewController zurück zum Get- bzw. TrackLocationViewController transferiert? Hinweis: [obj respondsToSelector:@selector(method)] überprüft, ob das Objekt obj die Methode method implementiert.

- d) Wozu dient die Klasse LocationDetailViewController? Wieso taucht sie nicht im Storyboard auf? Wann wird die Klasse instanziiert? Wie viele Instanzen der Klasse gibt es zur Laufzeit maximal?
- e) „CLLocation (Strings)“ ist ein Beispiel für eine Objective-C „Category“. Damit lässt sich die Funktionalität existierender Klassen durch Hinzufügen neuer Methoden erweitern (in diesem Fall CLLocation), ohne dass man (wie bei Vererbung) die existierenden Methoden der Klasse ändert. Categories können nur Methoden, aber keine Klassen- oder Instanzvariablen hinzufügen. Beschreiben Sie die Funktionalität der Category „CLLocation (Strings)“ und die Bedeutung der Localized Strings.
- f) Machen Sie sich mit der Verwendung des location managers in GetLocationViewController.{h,m} vertraut. Wie lange bleibt der location manager aktiv? Hinweis: [self performSelector:@selector(method) withObject:argument afterDelay:delay] führt nach „delay“ Sekunden die Methode „method“ von Objekt „self“ mit Argument „argument“ aus.

[1] <http://developer.apple.com/library/ios/#samplecode/LocateMe/Introduction/Intro.html>

Abgabe

- Geben Sie die Lösung bis zum 16.11.2011, 16:00 Uhr in UniWorX ab.
- Es ist nicht erlaubt die Aufgaben gemeinsam zu bearbeiten. Bitte beachten Sie dazu auch die Hinweise zu Plagiaten.
<http://www.medien.ifi.lmu.de/lehre/Plagiate-Ifi.pdf>