

Modul 04

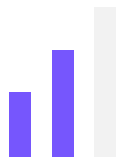
App Entwicklung für iOS

Nanodegree "iOS Developer/in"

Modulübersicht

Die Reise, ein iOS-Entwickler zu werden, beginnt in dem Moment, in dem eine Person die Idee für eine großartige App hat. Dieses Nanodegree-Programm bereitet die Lernenden darauf vor, ihre erste iOS-App zu veröffentlichen, egal ob sie bereits programmieren können oder gerade erst anfangen. Während die Lernenden die Swift-Programmiersprache meistern und eine Sammlung von Apps erstellen, um ihre Fähigkeiten zu präsentieren, profitieren sie von detaillierten Code-Reviews, wertvollen Karrieretipps und dem Coaching professioneller iOS-Entwickler.

Zielgruppe
Fortgeschrittene



Zeitaufwand
4 Monate bei 10h/Woche*



Software
XCode



Hardware
Mac PC



* Die Dauer dieses Programms ist eine Schätzung der Gesamtstunden, die der durchschnittliche Student benötigt, um alle erforderlichen Kursarbeiten abzuschließen, einschließlich Vorlesungs- und Projektzeit. Wenn Sie etwa 10-15 Stunden pro Woche mit dem Programm arbeiten, sollten Sie es innerhalb der angegebenen Zeit abschließen. Die tatsächlichen Stunden können variieren.

Swift Grundlagen



Die Lernenden werden eine Reihe von Kodierungsübungen absolvieren, um ihr Verständnis von Swift zu testen. Es wird Übungen für Variablen, Strings, if (else-if und else)-Anweisungen und Funktionen geben.

Projekt

Bundesliga Simulator

Bauen Sie eine einfache Konsolen-Anwendung mit Swift, um die Programmiersprache besser kennen zu lernen. Diese Anwendung gibt Ihnen aus einer Zufälligen Anzahl an Teams eine Simulierte Saison aus, und bestimmt am Ende den Gewinner. Anhand dieses Beispiels werden zum einen Ihre Programmierkenntnisse aufgefrischt und Sie lernen die Besonderheiten der Programmiersprache Swift kennen, die zum Entwickeln von iOS Anwendungen benötigt wird.

Thema 01

Variablen und Datentypen

- Deklarieren Sie Variablen und Konstanten mit grundlegenden Swift-Typen wie Bool, Int, Double und Float.
- Greifen Sie auf Werte von Variablen und Konstanten zu und ändern Sie diese.
- Beheben Sie Compiler-Probleme im Zusammenhang mit der falschen Verwendung von Variablen und Konstanten.
- Verwenden Sie Escape-Zeichen und String-Interpolation, um Variablen- und Konstantenwerte innerhalb von Strings zu formatieren.

Thema 02

Operations und Expressions

- Berechnen Sie neue Werte unter Verwendung vorhandener Variablen und Konstanten.
- Verwenden Sie Vergleichsoperatoren, um die Gleichheit zwischen zwei Werten zu bestimmen.
- Verwenden Sie boolesche Operatoren, um Ausdrücke aufzubauen, die Wahrheitswerte verwenden.

Thema 03

Verzweigungen

- Schreiben Sie boolesche Ausdrücke, um Entscheidungslogik darzustellen.
- Kombinieren Sie boolesche Ausdrücke mit logischen Operatoren.
- Verwenden Sie boolesche Ausdrücke zusammen mit if, else-if und else-Anweisungen, um den Ablauf der Codeausführung zu steuern.
- Verwenden Sie switch-Anweisungen, um Code basierend auf mehreren Werten einer einzigen Variablen auszuführen.
- Verwenden Sie for-, while- und do-while-Schleifen, um den Ablauf der Codeausführung zu steuern.

Thema 04

Funktionen

- • Bestehenden Code in wiederverwendbare Funktionen einpacken.
- • Funktionen korrekt definieren und aufrufen.
- • Funktionen mit Parametern und Rückgabetypen spezifizieren.
- • Unterscheidung zwischen Werten, die im Geltungsbereich (in-scope) und außerhalb des Geltungsbereichs (out-of-scope) liegen.
- • Lokale und externe Parameter korrekt verwenden.
- • Parametertypen und Rückgabetypen identifizieren.

Thema 05

Structures und Enums

- Mehrere Werte in Strukturen (structs) zusammenfassen.
- Instanzen von Strukturen erstellen.
- Funktionen (auch als Methoden bezeichnet) zu Strukturen hinzufügen.
- Eigenschaften von Strukturen zugreifen und Methoden aufrufen.
- Berechnete Eigenschaften definieren, die ihren Wert basierend auf anderen Werten berechnen.
- Aufzählungen (enums) definieren und Rohwerte für verschiedene Fälle zuweisen.
- Aufzählungen in Verbindung mit switch-Anweisungen verwenden.

Thema 06

Optionals

- Verstehen, wann ein Wert nil sein kann und wann ein optionaler Typ verwendet werden sollte.
- Variablen und Konstanten als explizite oder implizit ausgepackte Optionals deklarieren.
- Optionale Werte sowohl sicher als auch unsicher auspacken.
- Optional-Chaining und der Nil-Coalescing-Operator verwenden, um sicher auf optionale Werte zuzugreifen.

Thema 07

Strings

- Strings mithilfe ihrer integrierten Eigenschaften und Methoden definieren und bearbeiten.
- Häufige String-Operationen wie Verkettung (Concatenation) und das Finden von Teilstrings durchführen.
- Häufige String-Manipulationen wie das Hinzufügen, Entfernen und Ersetzen von Teilstrings durchführen.

Thema 08

Collections

- Ungeordnete Daten des gleichen Typs mit Arrays speichern.
- Auf Array-Inhalte zugreifen und diese ändern.
- Schlüssel-Wert-Paare mit Dictionaries speichern.
- Auf Dictionary-Inhalte zugreifen und diese ändern.
- Ungeordnete Daten des gleichen Typs mit Sets speichern.

Thema 09

Objektorientierte Programmierung

- Den Unterschied zwischen Wert- und Referenztypen verstehen und wie dies auf Strukturen und Klassen zutrifft.
- Eine Klasse erbt die Eigenschaften und Methoden einer anderen Klasse.
- Polymorphie verstehen: Wie ein Typ für einen anderen Typ eingesetzt werden kann und wie dies mit Vererbung zusammenhängt.
- Klassen schreiben, die demselben Protokoll entsprechen.
- Zusätzliche Funktionen zu Klassen mit Erweiterungen hinzufügen.

iOS App Entwicklung mit Swift



Mit Swift und Xcode, der Entwicklungsumgebung von Apple für die App-Entwicklung, wird ein App erstellt. Die Lernenden werden AutoLayout, UIButtons und UILabels verwenden, um eine Benutzeroberfläche zu erstellen und lernen, auf Berührungsereignisse in einer App mithilfe von ViewController und mehreren Ansichten zu reagieren. Sie werden auch lernen, wie man in einer Sprachaufnahme-App die Aufnahme und Wiedergabe von Audio einrichtet.

Projekt

Audio Mix App

Erstellen Sie eine iPhone-App, die Audio aufzeichnet und es mit verschiedenen Audiofiltern und -modi wiedergibt, einschließlich angepasster Geschwindigkeit und Tonhöhe, Echo und Hall.

Thema 01

Einführung in Xcode

- Navigieren Sie durch die wichtigsten Komponenten der Xcode-Entwicklungsumgebung, einschließlich des Navigators, des Debug-Bereichs und der Utilities.
- Erstellen Sie ein Xcode-Projekt für eine neue iOS-Anwendung.
- Erläutern Sie die Ziele und Architektur des Model View Controller (MVC)-Entwurfsmusters.

Thema 02

ViewController und mehrere Views

- Konfigurieren Sie den Anwendungsstatus an den geeigneten Anpassungspunkten im Lebenszyklus einer Ansicht.
- Erstellen und navigieren Sie Mehrfensteranwendungen mithilfe eines UINavigationController.
- Manipulieren Sie Benutzeroberflächenobjekte unter Verwendung von IBOutlets und IBActions.

Thema 03

Auto Layout und Buttons

- Verwenden Sie Storyboards, das visuelle Bearbeitungswerkzeug von Xcode, um Benutzeroberflächenobjekte zu positionieren, zu skalieren und zu konfigurieren.
- Verknüpfen Sie Benutzeroberflächenobjekte in einem Storyboard mit ihren entsprechenden Controllern mithilfe von IBOutlets.
- Definieren Sie Rückruffunktionen, sogenannte IBActions, die als Reaktion auf Benutzerinteraktionen aufgerufen werden.
- Erstellen Sie AutoLayout-Beschränkungen, um sicherzustellen, dass UI-Elemente unabhängig von der Gerätegröße und -abmessungen korrekt dimensioniert und positioniert werden.

Thema 04

Delegation & Audio-Aufnahmen

- Schreiben Sie Protokolle, um Funktionen auszudrücken, die von Swift-Klassen übernommen werden können.
- Verwenden Sie Protokolle, um die Verantwortlichkeiten einer bestimmten Aufgabe oder Aufgabengruppe einem anderen Objekt zu übertragen.
- Erstellen Sie eine Verbindung zu einem AVAudioRecorder, um Audio mit dem Mikrofon eines iOS-Geräts aufzunehmen und zu speichern.
- Verwenden Sie Segues, um zwischen Ansichten in einer Anwendung zu wechseln.

Thema 05

Wiederholung und Effekte

- Erstellen und konfigurieren Sie StackViews, die Layoutbeschränkungen für ihre Unteransichten enthalten und automatisch konfigurieren.
- Wiedergabe von Audio mit Objekten, die in der AVFoundation-Bibliothek definiert sind.
- Anwenden von Audio-Wiedergabeeffekten mithilfe von Audio-Nodes, die über eine benutzerdefinierte Schnittstelle zugänglich gemacht werden.

UIKit Grundlagen



Entwickeln Sie eine App mit UIKit, dem Front-End-Framework von Apple zur Entwicklung schneller und leistungsstarker Benutzeroberflächen. Erfahren Sie mehr über das Delegationsmuster, um Verbindungen zwischen dem Modell, der Ansicht und dem Controller der App herzustellen, und fügen Sie Tabellenansichten und Navigation zu einer App hinzu.

Projekt

Reisetagebuch App

Erstellen Sie eine App, mit der ein Benutzer ein Bild aufnehmen und oben und unten Text hinzufügen kann, um einen Eintrag in ihrem Reisetagebuch zu erstellen. Der Benutzer kann das Foto auf Facebook und Twitter teilen sowie per SMS oder E-Mail versenden. Die beschrifteten Bilder werden in einer Tab-Ansicht mit zwei Tabs angezeigt: einer TableView und einer collectionView.

Thema 01

Outlets und IBActions

- Verstehen Sie, wie Sie Outlets und Actions sowohl programmatisch als auch grafisch über Storyboard verbinden können.
- Verwenden Sie grundlegende UIKit-Klassen wie UIButton, UILabel und UISwitch.
- Üben Sie das Debuggen von Problemen mit IBOutlet und IBActions.

Thema 02

Views und Segues

- Erfahren Sie, wie Apple zwischen modaler Präsentation und Navigation unterscheidet.
- Lernen Sie, wie man Views modal präsentiert.
- Verwenden Sie leistungsstarke UIKit-Klassen wie UIImagePickerController, UIAlertController und UINavigationController.

Thema 03

Das Delegate Pattern

- Lernen Sie, wie Delegates wichtige Verbindungen zwischen Modell, Ansicht und Controller herstellen.
- Implementieren Sie UIKit-Komponenten, die das Delegate-Muster nutzen, wie UITextField und UITextFieldDelegate.
- Zeigen Sie Ihr Verständnis, indem Sie eine Reihe von Challenge-Apps erstellen.

Thema 04

Table Views

- Lernen Sie die wichtigsten Methoden von UITableViewDelegate und UITableViewDataSource.
- Erforschen Sie den Code für mehrere Apps mit Tabellen und implementieren Sie dann Ihre eigene UITableView.
- Üben Sie das Manipulieren von Tabellenzellen.

Thema 05

Navigation

- Lernen Sie, wie iOS Navigation-Stacks verwendet, um mehrere Ansichten in einer App zu verwalten.
- Erstellen Sie die Navigation, die es einem Benutzer ermöglicht, auf einer Tabelle eine Zeile anzutippen und die Details eines Elements anzuzeigen.
- Lernen Sie Navigationsklassen wie UINavigationController und UIBarButtonItem kennen.

Datenspeicherung



Erfahren Sie mehr über einfache Persistenz, das iOS-Dateisystem und die "Sandbox". Richten Sie die benötigten Klassen ein, um Core Data zum Erstellen, Speichern und Löschen von Modellobjekten einzurichten. Aktivieren Sie die reaktive Aktualisierung der Benutzeroberfläche, sobald das Modell geändert wird. Lernen Sie, Benutzerdaten sicher zwischen Versionen zu migrieren.

Projekt

Dein erste eigene iOS App

Erstelle eine App, die Bilder von Flickr herunterlädt und speichert. Die App ermöglicht es Benutzern, auf einer Karte Pins abzulegen, als ob es Stationen einer Tour wären. Benutzer können dann Bilder für den Standort herunterladen und sowohl die Bilder als auch die Verknüpfung der Bilder mit dem Pin dauerhaft speichern. Anschließend wirst du alles gelernte in deiner eigenen ersten iOS App in die Tat umsetzen und unter Beweis stellen, was du in diesem Modul gelernt hast.

Thema 01

Einfache Speicherung

- Lernen Sie einfache Persistenz kennen und wie Sie kleine Datenstücke speichern können.
- Erfahren Sie, wie Sie Benutzereinstellungen mithilfe von `NSUserDefaults` festlegen können.
- Üben Sie das Festlegen einfacher Präferenzen in einer bestehenden App.

Thema 02

iOS Filesystem

- Lernen Sie das iOS-Dateisystem und die "Sandbox" kennen.
- Sehen Sie, wie Sie auf diese Dateien mit `NSFileManager` zugreifen können.
- Verwenden Sie den Dateimanager, um eine Datei zu speichern und zu lesen.

Thema 03

Core Data

- Machen Sie sich mit Core Data vertraut, dem Framework von Apple zur Verwaltung der Daten-Ebene.
- Erforschen Sie, was eine Daten-Ebene ist.
- Konvertieren Sie eine nicht-Core Data-Notiz-App, um ein Core Data-Modell zu haben.

Thema 04

Core Data Stack

- Richten Sie die Klassen ein, die wir benötigen, um Core Data zum Laufen zu bringen.
- Verwenden Sie den Stack, um die Erstellung und Löschung von Modellobjekten zu verwalten.
- Persistieren Sie Änderungen, damit Daten erhalten bleiben, wenn die App oder das Gerät neu gestartet wird.

Thema 05

Core Data effizienter gestalten

- Aktualisieren Sie die Benutzeroberfläche reaktiv, wenn sich das Modell ändert.
- Richten Sie einen `NSFetchedResultsController` ein, um Datenänderungen zu beobachten und die UI zu benachrichtigen.
- Passen Sie eine Tabelle an, um mit einem fetched results controller als Datenquelle zu arbeiten.

Unser bewährter Ansatz

Für den Aufbau berufsrelevanter digitaler Kompetenzen



Erfahrene Entwickler/innen

Wir begleiten dich auf dem Weg zum App Entwickler

- Personalisiertes Projektfeedback mit Codeüberprüfung durch unser geschultes Personal.
- Jeden Monat 2x 60 Minuten Meeting mit unseren Tutor/innen.



Marvin Gatermann

iOS & Android Developer



Till Hemmerich

iOS & Android Developer



Gordon Lucas

iOS & Android Developer

Vielen Dank.

www.syntax-institut.de/selbstzahler/