

# iOS-разработчик расширенный

## 13 месяцев

продолжительность курса

1 НЕДЕЛЯ | 13–15 ЧАСОВ

### 00

#### Бесплатная часть

- Как здесь всё устроено
- Запускаем приложение
- Проект: Игра Pong

4 НЕДЕЛИ | 70 ЧАСОВ

### 01

#### Подготовительная часть. Основы языка Swift

Старт подготовительной части

Спринты 1-2

10 НЕДЕЛЬ | 150 ЧАСОВ

### 02

#### Вёрстка, сеть и хранение данных

Проект: Приложение-квиз по топ-250 фильмам IMDb

Спринты 3-7 Неделя каникул

Жёсткий дедлайн

180 ЧАСОВ | 12 НЕДЕЛЬ

### 03

#### Навигация, работа с таблицами, авторизация

Проект: Приложение с лентой изображений

Спринты 8-13 Неделя каникул

Жёсткий дедлайн

8 НЕДЕЛЬ | 120 ЧАСОВ

### 04

#### Коллекции, встроенные контроллеры, базы данных и основы многопоточности

Проект: Приложение-трекер для всего

Спринты 14-17 Неделя каникул

Жёсткий дедлайн

8 НЕДЕЛЬ | 122 ЧАСА

### 05

#### SwiftUI, Combine, структурированная многопоточность

Проект: многоэкранное приложение на SwiftUI и Combine

Спринты 18-21 Неделя каникул

Жёсткий дедлайн

4 НЕДЕЛИ | 42 ЧАСА

### 06

#### Проектный месяц

Проект: Многоэкранное приложение на SwiftUI и Combine

Спринты 22-23 Неделя каникул

Жёсткий дедлайн

5 НЕДЕЛЬ | 60 ЧАСОВ

### 07

#### Командный дипломный проект

Выпускной проект: NFT-маркетплейс

Спринты 24-26

Жёсткий дедлайн



Выпускной.  
Получение диплома  
или сертификата

# Бесплатная вводная часть



1 неделя | 13–15 часов

Прежде чем погрузиться в учёбу и начать практиковаться, вы узнаете:

- В чём особенности платформы iOS и экосистемы Apple;
- Чем занимаются iOS-разработчики и разработчицы;
- Из каких блоков состоит программа;
- Какие проекты вам предстоит выполнить;
- Как в целом проходит обучение в Практикуме;
- Кто поможет вам учиться;
- Как найти работу после обучения.

Сразу после онбординга начнётся бесплатная часть курса. На ней вы:

- Познакомитесь с языком Swift;
- Поработаете со своим первым приложением (для этого не нужно заново учить математику — но нужен компьютер от Apple с системой macOS Ventura или более поздней версии);
- Увидите результаты своих трудов и поймёте, интересно ли вам заниматься iOS-разработкой в дальнейшем.

Игра Pong

# Платное продолжение



744 часа | 13 месяцев  
(включая 5 недель  
каникул)

Платная часть состоит из этапов: подготовки, работы с функциональными слоями приложения и реальными задачами на трёх проектах, перехода к командной разработке, командного дипломного проекта.



4 часа  
опциональный блок

После покупки курса мы предложим вам «Бассейн». Это несколько дополнительных уроков, которые станут доступны вместе с первым спринтом. Благодаря им вы познакомитесь с основными понятиями программирования. Простые и увлекательные задачи на базовую логику и алгоритмику помогут лучше усваивать дальнейший материал, особенно если вы сталкиваетесь с программированием впервые в жизни.

### Важно

Можно не проходить «Бассейн», а сразу отправиться исследовать iOS-галактику. Чтобы определиться, нужен вам этот этап или нет, мы предложим вам ответить на несколько вопросов.

## Содержание

Уроки

1. Исходные булевы операторы
  2. Возможные комбинации операторов
  3. Оператор НЕ
  4. Оператор И
  5. Оператор ветвления ЕСЛИ... ТО
  6. Как получить информацию
  7. Алгоритм
  8. Функция
  9. Цикл
  10. Массив
-

4 недели | 70 часов

Последовательно познакомитесь с языком программирования Swift. Изучать его будем с помощью практических задач в Playground (специальный режим в среде разработки Xcode).

## Знакомство с iOS и основами Swift

# 01

4 недели | 70 часов

Изучите основные конструкции языка программирования Swift с помощью практических задач в Playground, познакомитесь со средой разработки и тем, как устроены мобильные приложения.

## Содержание модуля 1

### Спринт 1

#### Темы

1. Начинаем путешествие
2. Playground
3. Начало работы: Xcode
4. Переменные
5. Условный оператор if
6. Коллекции и циклы
7. Функции
8. Перечисления

### Спринт 2

#### Темы

1. Структуры данных
2. Опционалы
3. Классы
4. Расширения
5. Протоколы

33 недели | 450 часов  
(включая каникулы)

На этом этапе вы пройдёте через три модуля, в каждом из которых самостоятельно напишете iOS-приложение, познакомитесь с функциональными слоями и освоите процесс создания приложения с нуля.

## Основы вёрстки, сетевого взаимодействия и хранения данных

# 02

10 недель | 150 часов

Проект этого модуля — одностраничное приложение с квизами о фильмах из топ-250 рейтинга самых популярных фильмов IMDb. Пользователь приложения последовательно отвечает на вопросы о рейтинге фильма. По итогам каждого раунда игры показывается статистика о количестве правильных ответов и лучших результатах пользователя. Цель игры — правильно ответить на все 10 вопросов раунда. Для пользователей приложение — это прежде всего работа с интерфейсом, который вы для них и делаете. В рамках работы над приложением вы сверстаете экран квиза, освоите HTTP и REST для работы с API IMDb, научитесь хранить данные пользователя и напишете свои первые тесты. Задачи и проект в этом модуле будут сначала проходить через самостоятельную проверку по чек-листу, а затем отправятся к код-ревьюеру.

## Содержание модуля 2

Спринт 3

Темы

1. Git
2. Обзор проекта
3. Знакомство с интерфейс билдером
4. Связь сториборда и кода

Задачи по спринту

- Создадите свой первый Pull Request в Git
- Создадите тренировочный экран с изображением и кнопкой
- Сделаете тренировочное приложение-счётчик, на экране которого будет текстовый элемент и кнопка

# Содержание модуля 2

---

## Спринт 4

### Темы

1. Знакомство с Figma
2. Вёрстка по макету
3. Реализация логики по макету

### Задачи по спринту

- Попробуете самостоятельно взять кнопку из макета в Figma и реализовать её точную копию, учитывая шрифты, форму и цвета в сториборде
  - Создадите экран запуска и основной экран приложения
  - Создадите логику приложения в соответствии с функциональными и техническими требованиями
- 

## Спринт 5

### Темы

1. Память и замыкания
2. Ответственность
3. Хранение данных

### Задачи по спринту

- Разобьёте логику работы приложения на несколько частей
  - Вынесете в отдельную сущность работу по генерации вопросов для квиза
  - Напишете класс для ведения статистики, которая будет учитывать лучший счёт игры в квиз среди всех игр
  - Сохраните статистику в UserDefaults
- 

## Спринт 6

### Темы

1. Сеть. Клиент-серверное взаимодействие
2. Сериализация и десериализация
3. URLSession, URLRequest

### Задачи по спринту

- Добавьте работу с сетью — фильмы для квиза должны загружаться из открытого API IMDb, должны быть использованы фильмы из топ-250 и самых популярных фильмов IMDb
  - Обработаете пользовательский сценарий, чтобы приложение показывало ошибку, если данные из сети не были загружены
- 

## Спринт 7

### Темы

1. Unit-тесты и продукт
  2. UI-тесты
  3. Архитектура и MVP
  4. Основные принципы разработки
  5. Рефакторинг на MVP
- 

### Задачи по спринту

- Напишете unit-тесты
  - Напишете UI-тесты
  - Проведёте рефакторинг проекта на MVP
- 

Одностраничное приложение с квизами по топ-250 фильмов по версии IMDb

Неделя каникул

Жёсткий дедлайн

# Навигация, работа с таблицами, авторизация и основы многопоточности

03

12 недель | 180 часов

В этом модуле вы освоите переходы между экранами, элементы вёрстки со скроллом и улучшите свои навыки клиент-серверного взаимодействия. Проектом этого модуля будет многостраничное приложение, предназначенное для просмотра изображений через API Unsplash. Задачи и проект в этом модуле будут сначала проходить через самостоятельную проверку по чек-листу, а затем отправятся к код-ревьюеру.

## Содержание модуля 3

### Спринт 8

#### Темы

1. Онбординг. Методологии разработки
2. Инициализация второго проекта
3. Таблицы
4. Практика по работе с таблицей в проекте

#### Задачи по спринту

- Сверстаете Launch Screen приложения
- Сверстаете главный экран приложения с лентой изображений

### Спринт 9

#### Темы

1. Онбординг в спринт
2. Навигация
3. ScrollView
4. Debug
5. Вёрстка кодом

#### Задачи по спринту

- Сверстаете экран профиля пользователя
- Добавьте ScrollView
- На экране SingleImageViewController реализуете кнопку «Поделиться»

### Спринт 10

#### Темы

1. Онбординг
2. Работа с Unsplash API
3. Авторизация: Алгоритм OAuth
4. Авторизация: Web view
5. Технология KVO
6. Делаем сетевой запрос
7. Логика splash screen

#### Задача по спринту

Реализуете экран авторизации с логотипом приложения и кнопкой «Войти» с помощью OAuth2.0

# Содержание модуля 3

---

## Спринт 11

### Темы

1. Swift Package Manager
2. Работа в многопоточной среде
3. Делаем запросы для экрана Профиля
4. Kingfisher
5. Information Security
6. Вёрстка кодом. Продолжение

### Задачи по спринту

- Добавьте показ индикатора загрузки
- Устраните дублирование запроса авторизационного кода
- Напишите запросы на получение изображения аватарки пользователя из сети
- Сохраните данные в Keychain

### ★ Задача со звёздочкой

Самостоятельно сверстаете кодом элементы, которые сейчас сделаны в Storyboard (например, ячейку в таблице).

---

## Спринт 12

### Темы

1. Запрос картинок из сети
2. Добавляем функциональность лайков
3. Анимации
4. Core Animation

### Задачи по спринту

- Добавьте функциональность лайков
  - Сделаете анимацию загрузки на главном экране
- 

## Спринт 13

### Темы

1. Charles Toolchain
2. Unit-тесты. Теория
3. Unit-тесты. Практика
4. UI-тесты
5. Финал проекта

### Задачи по спринту

- Протестируете краевые случаи с помощью Charles Proxy
  - Проведёте рефакторинг WebViewViewController
  - Покроете проект Unit-тестами и UI-тестами
- 

Приложение с лентой изображений

Неделя каникул

Жёсткий дедлайн

8 недель | 120 часов

В этом модуле вам предстоит сделать приложение-трекер, помогающее развивать привычки и отслеживать цели. Вы уделите особое внимание проектированию приложения и его архитектуре, освоите сложную вёрстку — работу с коллекциями, ещё раз попрактикуетесь в работе с таблицами. На этот раз все данные будут храниться локально в базе данных, а работа с информацией будет происходить на фоновом потоке.

## Содержание модуля 4

### Спринт 14

#### Темы

1. Онбординг в модуль. Описание проекта Tracker
2. Коллекции
3. UICollectionViewLayout

#### Задача по спринту

Сверстаете главный экран приложения, экран создания трекера и экран создания расписания

### Спринт 15

#### Темы

1. Базы данных. SQL
2. Core Data
3. Context
4. FetchRequest
5. FetchedResultsController

#### Задача по спринту

Закончите вёрстку экрана создания трекера и имплементируете в проект Core Data

### Спринт 16

#### Темы

1. Онбординг в спринт
2. Архитектура MVVM
3. SwiftUI и Combine
4. UIPageViewController
5. Иерархии вью контроллеров

#### Задачи по спринту

- Реализуете экран выбора категории привычки на MVVM
- Создадите онбординг приложения с помощью PageViewController

# Содержание модуля 4

---

Спринт 17

Темы

1. Локализация
2. Скриншотные тесты
3. Темная тема
4. Аналитика
5. UISearchController
6. Код-ревью

Задача по спринту

Реализуете скриншот-тесты, добавите тёмную тему для главного экрана, подключите библиотеку AppMetrica, проведёте код-ревью

---

Приложение-трекер для всего

Неделя каникул

Жёсткий дедлайн

## SwiftUI, Combine, структурированная многопоточность

05

8 недель | 122 часа

Проект модуля — многоэкранное приложение на SwiftUI. Вы освоите вёрстку и навигацию, реализуете архитектуру MVVM в SwiftUI и Combine, поработаете со структурированной многопоточностью, углубите навык анализа кода: попрактикуетесь в поиске ненужных зависимостей и антипаттернов в коде, в рефакторинге. Задачи и проект модуля вы сначала проверите сами по чек-листу, а затем отправите код-ревьюеру.

# Содержание модуля 5

---

Спринт 18

Темы

1. Паттерны
2. Кодогенерация

Задача по спринту

Реализуете сетевой слой приложения, используя кодогенерацию

---

Спринт 19

Темы

1. Базовая верстка с использованием SwiftUI
2. Настройка навигации с использованием SwiftUI

Задача по спринту

Сверстаете экраны приложения с использованием SwiftUI, включающего в себя tabbar с 2 вкладками

---

## Содержание модуля 5

---

Спринт 20

Темы

1. Анимация в SwiftUI
2. SwiftUI + Combine

Задача по спринту

Завершите вёрстку экранов в проекте «Расписание путешествий», включая добавление экранов с карточкой перевозчика и настройками, а также реализацию коллекции Stories

---

Спринт 21

Темы

1. Реализуем MVVM на SwiftUI + Combine
2. Что такое структурированная многопоточность — знакомимся с новым

Задача по спринту

Реализуете MVVM на SwiftUI и Combine в приложении

---

Проект: многоэкранное приложение на SwiftUI и Combine

Неделя каникул

Жёсткий дедлайн

## Проектный месяц

# 06

4 недели | 42 часа

Это обязательная практика программы «iOS-разработчик расширенный»; она даёт выпускникам преимущество при трудоустройстве.

Вы выполните проект, основанный на реальной бизнес-задаче, получите опыт работы в команде и добавите кейс в резюме.

## Содержание модуля 6

---

Спринт 22

---

Спринт 23

---

Неделя каникул

Жёсткий дедлайн

5 недель | 60 часов

## Переходный этап

Подготовьтесь к командной разработке — познакомьтесь с жизненным циклом продукта и работой с AppStore, Agile, инструментами для командной работы и основными рабочими процессами помимо написания кода.

### Спринт 24

#### Темы

1. Жизненный цикл продукта
2. Workflow
3. Git для командной разработки

# Этап дипломного командного проекта

Разделитесь на команды по 3-5 человек и вместе сделаете eCommerce приложение — небольшой NFT-маркетплейс. Это приложение с каталогом NFT-картинок, которые можно просматривать, выбирать, класть в корзину, проходить чекаут и совершать покупку. У вас будет техническое задание от менеджера и дизайн — нужно будет декомпозировать всю работу, разбить на спринты и сделать продукт за 4 недели

## Спринт 25

### 1 неделя

Выберете архитектуру и способ вёрстки

### 2 неделя

Сдадите 1/3 часть эпика на ревью

## Спринт 26

### 3 неделя

Обработаете фидбек и поработаете с 2 частью эпика

### 4 неделя

- Сдадите весь эпик и поработаете с фидбеком.
- Поделитесь результатами работы с другими командами на демо.
- Проведёте ретроспективу, сделаете выводы по всему командному проекту.

NFT-маркетплейс

Жёсткий дедлайн

Выпускной и получение диплома

