

курс

development

12 недель

iOS: разработка приложений с 0

r_d

iOS

программа курса

20 занятий

1

Swift: начало

2

ООП: основы

3

Создание iOS-
прило-
жения в Xcode

4

Создание
интерфейса iOS-
приложения

5

Динамические
интерфейсы, часть 1

6

Динамические
интерфейсы, часть 2

iOS

программа курса

20 занятий

7

Динамические
интерфейсы, часть 3

8

Навигация в
приложении, часть 1

9

Навигация в
приложении, часть 2

10

Анимации в iOS

11

Работа с памятью
в iOS

12

Многозадачность
в iOS, часть 1

iOS

программа курса

20 занятий

13

Многозадачность в
iOS, часть 2

14

Дебаг
iOS-
приложения

15

Тестирование

16

Хранение данных
в приложении

17

Работа с сетью
в приложении

18

Сборка
приложения

iOS

программа курса

20 занятий

19

Современные
архитектуры для
iOS приложений

20

Защита
курсовых
проектов

Работа с сетью в приложении

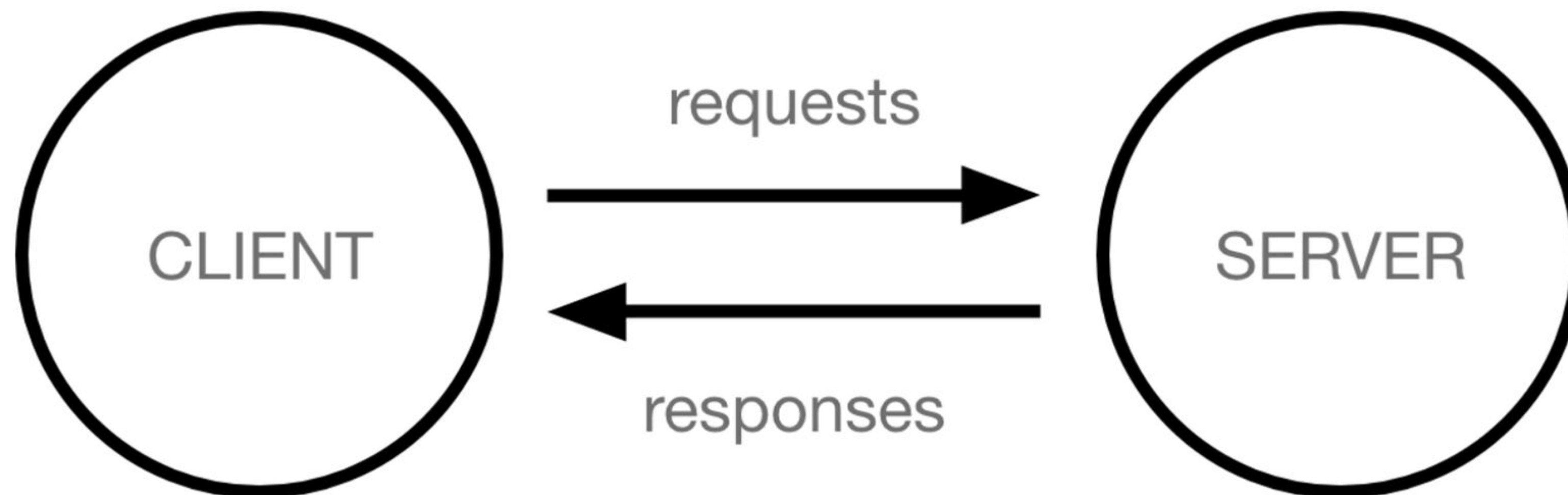
- Что такое интернет-запрос?
- Как делать интернет-запросы вообще?
- Как делать интернет-запросы в iOS приложении?

Что такое интернет-запрос

- Теория о том, как работают интернет-запросы, не особо нужна для того, чтобы написать простой код на iOS.
- Но если мы хотим ~~стать разработчиками~~ устроиться на работу, то должны знать теорию.
- А иначе как пройти собеседование? 🙄

Что такое интернет-запрос

Все компьютеры, имеющие доступ к сети Интернет, делятся на 2 типа — **клиенты** (clients) и **серверы** (servers). Они общаются друг с другом с помощью **запросов** (requests) и **ответов** (responses) на эти запросы.



Что такое интернет-запрос

- **Клиенты** — это самые обычные устройства, имеющие интернет-соединение. Например, ваш компьютер или планшет, через который вы сейчас смотрите эту лекцию.
- **Серверы** — это такие же компьютеры, как и в случае с клиентом. Но единственным отличием является то, что такие компьютеры “хранят” у себя локально веб страницы / сайты / приложения.
- К этой информации у клиентов есть доступ. С помощью конкретного IP адреса любой “клиент” может попасть на “сервер” и получить эти данные. Клиент скачивает себе локально эту веб-страницу, веб-сайт либо веб-приложение и воспроизводит пользователю весь контент на экране.

Что такое интернет-запрос

Теперь представим в голове такую картинку — улица, по середине улицы с левой стороны находится ваш дом (“клиент”), справа — магазин с продуктами (“сервер”).

Давайте разберемся на этом примере с теми технологиями, которые являются компонентами в коммуникации клиента и сервера.



Что такое интернет-запрос

TCP/IP (Transmission Control Protocol and Internet Protocol)

Это 2 протокола (алгоритма) общения, которые определяют, как именно данные должны передаваться по интернету.

В нашем примере это будет какой-то транспорт, с помощью которого мы можем доехать до магазина, а выйдя из магазина с продуктами, загрузить их в машину / велосипед / etc и доехать к дому.

Что такое интернет-запрос

TCP/IP (Transmission Control Protocol and Internet Protocol)

Это 2 протокола (алгоритма) общения, которые определяют, как именно данные должны передаваться по интернету.

В нашем примере это будет какой-то транспорт, с помощью которого мы можем доехать до магазина, а выйдя из магазина с продуктами, загрузить их в машину / велосипед / etc и доехать к дому.

Что такое интернет-запрос

DNS (Domain Name Servers)

Это специальный адрес, с помощью которого клиент может найти конкретный сервер, на котором сохранён нужный нам веб-сайт (чтобы затем установить контакт и послать какое-то сообщение серверу).

В нашем примере DNS — физический адрес (улица, дом), по которому располагается наш магазин, где есть нужные нам продукты (контент веб-страницы).

Что такое интернет-запрос

HTTP — Hypertext Transfer Protocol.

Это протокол, который описывает формат общения (как должны выглядеть сообщения) между клиентом и сервером.

Из примера с дорогой HTTP — это язык, на котором мы делаем заказ в магазине. К примеру, русский 😊

Из всего вышеперечисленного нас, как iOS разработчиков, в первую очередь интересует что есть самое “прикладное” из этого набора. Поэтому далее мы немного подробнее поговорим именно про протокол HTTP.

Как делать интернет-запросы вообще

Когда мы говорим про прикладное программирование, например “написать iOS приложение”, в первую очередь мы подразумеваем, что наше приложение будет выступать в роли “клиента” (а не веб-сервера).

Далее, важно понимать, что такие вещи как TCP/IP, DNS, механизм установки интернет-соединения и прочие штуки из нашего примера уже реализованы в нашей операционной системе.

От нас, как от разработчиков, все что требуется — описать сам запрос (request) и получить ответ (response) от веб-сервера как реакцию на то, что мы ему послали.

И request и response описываются протоколом HTTP.

Как делать интернет-запросы вообще

Сперва, перед тем как отправить наше первое сообщение серверу, мы должны установить с ним соединение. Как только мы это сделали, мы должны сформировать наше сообщение, написанное в формате протокола HTTP.

Простейшее сообщение должно состоять из 2-х строк.

Как делать интернет-запросы вообще

В первой строке мы указываем метод URI, версию HTTP протокола и путь, по которому лежит наша web-страница.

На пример, первая строка может выглядеть вот так: GET / HTTP/1.1.

Как делать интернет-запросы вообще

Вторая строка должна состоять из как минимум одного **заголовка (header)**. Каждый заголовок описывается как key-value. Например заголовок “Host” (единственный, который является обязательным).

В итоге простейшее сообщение от клиента в формате HTTP протокола будет выглядеть таким образом.



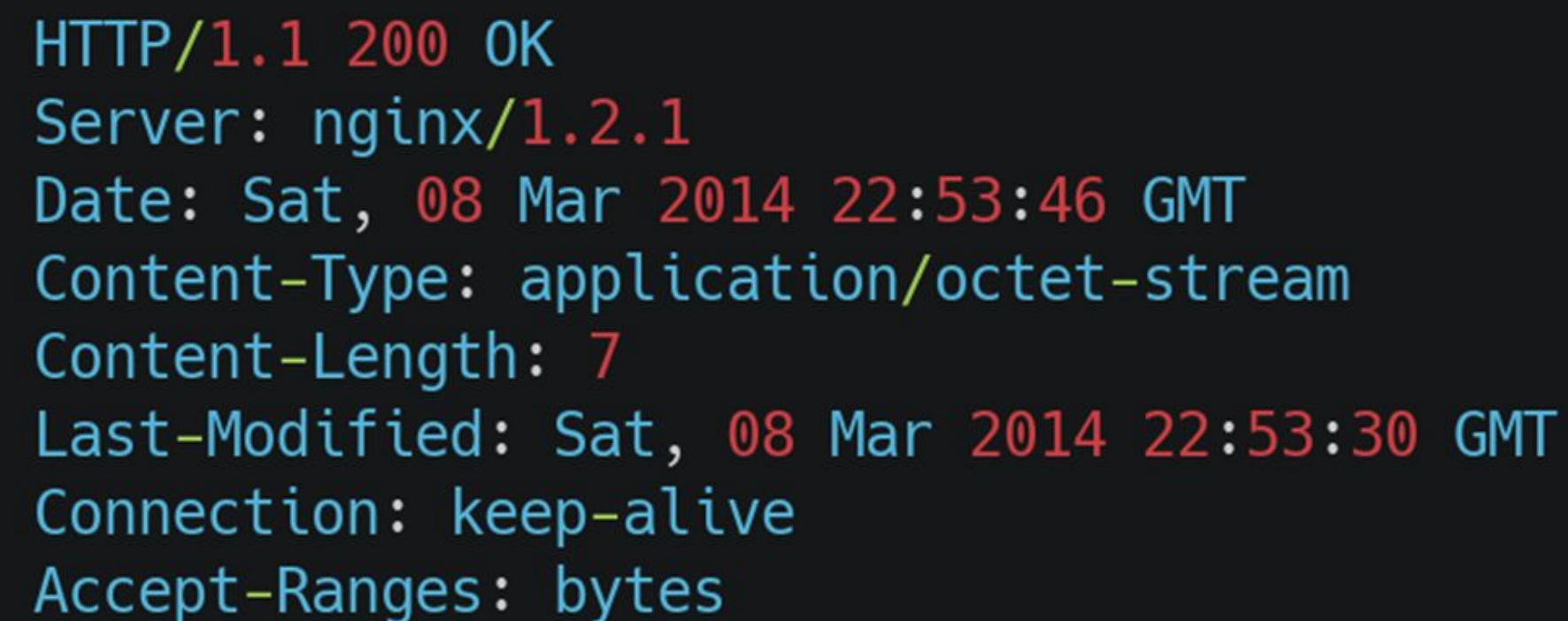
```
GET / HTTP/1.1  
Host: alizar.habrahabr.ru
```

Как делать интернет-запросы вообще

После того, как сервер получит такой request, он его обрабатает.

Допустим, что все хорошо и по пути “/” на запрос GET сервер нашел у себя такую страницу и должен нам вернуть какой-то response в ответ.

Формат ответа такой: HTTP/Версия Код состояния Пояснение.

A terminal window with a dark background and three colored window control buttons (red, yellow, green) in the top-left corner. It displays an HTTP response in a color-coded format: 'HTTP/1.1 200 OK' on the first line, followed by headers: 'Server: nginx/1.2.1', 'Date: Sat, 08 Mar 2014 22:53:46 GMT', 'Content-Type: application/octet-stream', 'Content-Length: 7', 'Last-Modified: Sat, 08 Mar 2014 22:53:30 GMT', 'Connection: keep-alive', and 'Accept-Ranges: bytes'.

```
HTTP/1.1 200 OK
Server: nginx/1.2.1
Date: Sat, 08 Mar 2014 22:53:46 GMT
Content-Type: application/octet-stream
Content-Length: 7
Last-Modified: Sat, 08 Mar 2014 22:53:30 GMT
Connection: keep-alive
Accept-Ranges: bytes
```

Wisdom

Как делать интернет-запросы вообще

В реальном мире для клиент-серверной коммуникации у нас существуют реальные клиенты (наши с вами компьютеры и планшеты) и реальные сервера (напр., google или amazon).

В разработке клиент-серверной коммуникации мы, как те, кто разрабатывает клиентскую часть, должны иметь инструменты, которые помогут нам как-то без участия реального сервера помочь создать симуляцию такого клиент-серверного общения.

Postman — одна из самых широкоизвестных и распространенных программ для эмуляции сервера и его API endpoints (пути, по которым клиент ходит, чтобы получить контент).

Как делать интернет-запросы в iOS приложении

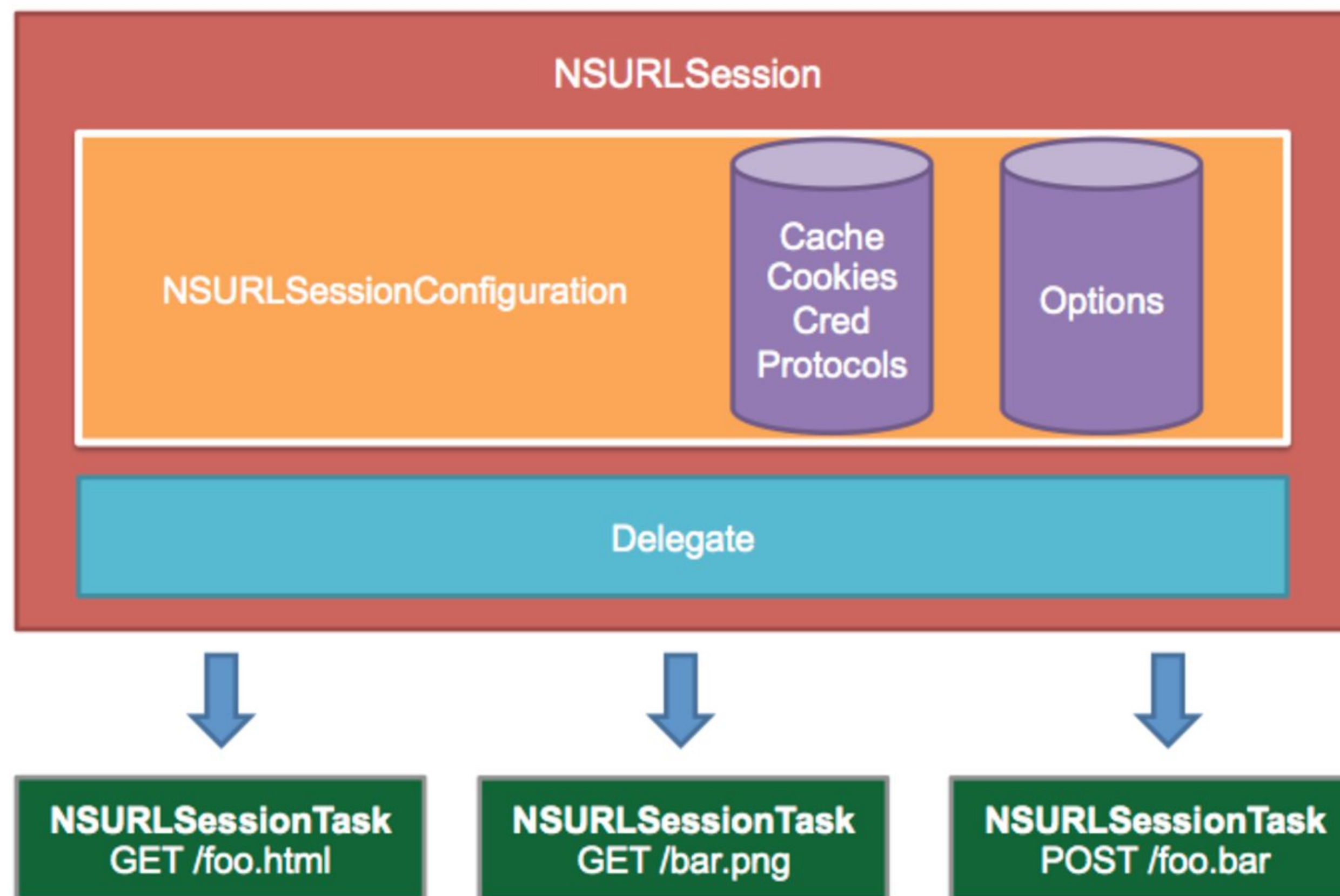
Начиная с iOS 7 в системе появился класс `NSURLSession`, объект которого управляет набором задач по передаче данных по сети.

Этот класс и его наследники занимаются обработкой запросов основанных на протоколе HTTP (отправка и приём).

Создать экземпляр класса `NSURLSession` можно либо используя статическое свойство `shared` либо через инициализатор, который принимает аргумент класса `NSURLSessionConfiguration`.

Как делать интернет-запросы в iOS приложении

Общая схема и иерархия классов, связанных с URLSession.



Как делать интернет-запросы в iOS приложении

Существует 3 готовых экземпляра класса `URLSessionConfiguration`:

1. `defaultSessionConfiguration` — общий набор настроек по-умолчанию, общий глобальный кеш, учетные записи, хранилище cookies.
2. `ephemeralSessionConfiguration` — тоже, что и `default`, но данные сохраняются в памяти (можно сделать “частную” сессию).
3. `backgroundSessionConfiguration` — позволяет выполнять задачи, пока приложение в фоне.

Как делать интернет-запросы в iOS приложении

Существует 3 экземпляра класса `NSURLSession` для выполнения запросов:

1. `NSURLSessionDataTask` — для простых GET запросов
2. `NSURLSessionUploadTask` — для загрузки данных с клиента на веб-сервер, используя POST / PUT запросы
3. `NSURLSessionDownloadTask` — для загрузки контента с веб-сервера на диск

Demo Time 🎉🎉🎉