

Студ. А. Н. Смолик

Науч. рук. ассист. И. А. Миронов
(кафедра информатики и веб-дизайна, БГТУ)

ОСОБЕННОСТИ РАЗРАБОТКИ ПРИЛОЖЕНИЯ ДЛЯ АДМИНИСТРИРОВАНИЯ БАЗЫ ДАННЫХ НА ЯЗЫКЕ ПРОГРАММИРОВАНИЯ SWIFT

Данная статья была выполнена с целью знакомства с новым языком программирования SWIFT и операционной системой Mac OS X, которые в связи с их широким использованием представляют, как практический, так и научный интерес.

В статье будут рассмотрены все этапы создания приложения для администрирования базы данных под операционную систему Mac OS X на примере сети ресторанов быстрого питания.

Для реализации данного проекта требовалось выполнить ряд ключевых задач:

- спроектировать клиент-серверную архитектуру приложения;
- спроектировать логическую структуру базы данных;
- разработать интерфейс программирования приложений (API) для доступа к базе данных;
- разработать приложение для администрирования базы данных под Mac OS X.

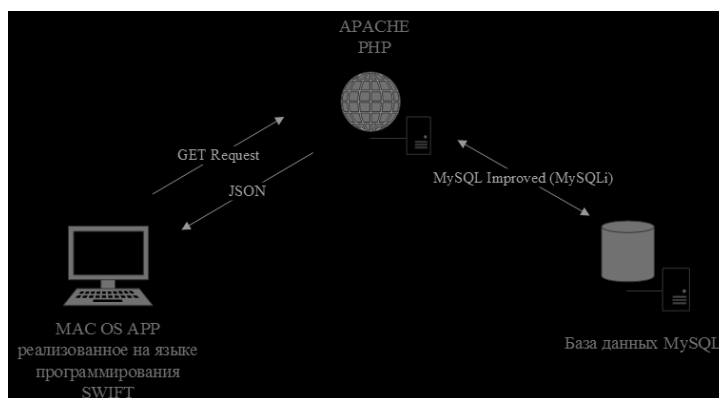


Рисунок 1 – Архитектура клиент-серверного приложения

На рис. 1 представлена архитектура клиент-серверного приложения. MAC OS X APP приложение отправляет GET-запросы на сервер, который возвращает данные в формате JSON — простом, основанном на подмножестве языка программирования JavaScript, способе сохранять и передавать структурированные данные.

Сервер взаимодействует с базой данных с помощью MySQLi — обновлённой версией драйвера PHP MySQL, который даёт различные

улучшения в работе с базами данных. MySQLi имеет 3 основные класса:

- `mysqli` – необходим для установки соединения с БД и будет полезен, если необходимо выполнить запросы так, как это делали в старом расширении MySQL;
- `mysqli_stmt` – необходим для использования новой возможности MySQLi, выполнять запросы по подготовленным выражениям;
- `mysqli_result` – объединяет функции для получения результатов запросов, сделанных с помощью `mysqli` или `mysqli_stmt`.

В качестве СУБД была выбрана MySQL – свободно распространяемая система, к тому же достаточно быстрая, надежная и, главное, простая в использовании, вполне подходящая для не слишком глобальных проектов. База данных состоит из четырех таблиц, в которых хранятся список ресторанов, список товаров, меню ресторанов, текущие заказы.

В качестве веб-сервера был выбран Apache, как наиболее устойчивый и популярный в Интернете. По разным оценкам его доля составляет почти 50% от общего числа используемых веб-серверов. API разработано на языке программирования PHP и включает в себя 14 основных функций, необходимых для администрирования БД.

Для разработки Desktop приложения использовалась среда разработки Xcode 7, которую Apple предоставляет сообществу для создания приложений под Mac и iOS.

В составе Xcode имеется удобный инструмент для проектирования пользовательского интерфейса – *Storyboard*. Работать с ним проще, чем с HTML и CSS, так как он не требует никаких знаний в программировании.

В *Storyboard* разработчик работает с экранами, которые связываете с помощью переходов. Затем проектируется интерфейс каждого экрана в отдельности с использованием стандартных элементов из библиотеки объектов (кнопки, текстовые поля, таблицы и др.).

На рис. 2 изображен фрагмент пользовательского интерфейса созданный при помощи *Storyboard*. В Xcode разработчикам на выбор предлагается два языка программирования – SWIFT или Objective-C. Для разработки приложения был выбран SWIFT, позиционируемый Apple как простой инструмент с выразительным синтаксисом, который благодаря своей интерактивности предполагает более быструю разработку приложений. Но несмотря на свою простоту, Swift работает достаточно быстро. Даже обычный алгоритм поиска в глубину в нем выполняется гораздо быстрее: до 2,6 раза быстрее чем в Objective-C и до 8,4 раза быстрее чем в Python 2.7.

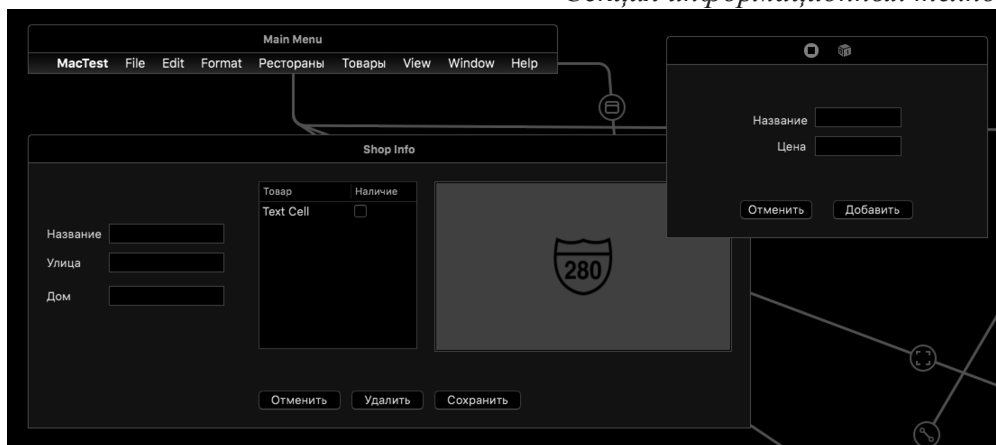


Рисунок 2 – Фрагмент пользовательского интерфейса на базе Storyboard

Для SWIFT разработчиков в Xcode имеется полезный инструмент – *Playground*. Написав строку кода, и результат появляется мгновенно. Если код запускается некоторое время, например, под действием цикла, то можно увидеть прогресс на временной шкале помощника. Временная шкала показывает переменные на графике, рисуя каждый шаг, когда создается представление. *Playground* идеален для проектирования новых алгоритмов, создания тестов, экспериментов с API.

В ходе разработки данного проекта был получен опыт в разработке приложений для операционной системы Mac OS X, развертывании клиент-серверных приложений, проектировании баз данных. Разработанный продукт позволяет клиентам формировать и оплачивать заказ из мобильного приложения, а затем получать его с помощью QR-кода. Администрирование сети ресторанов (управление заказами, редактирование меню и др.) осуществляется из Mac OS X приложения.

УДК 004.031+ 659.4.012.12

Студ. А. А. Сугако

Науч. рук. ассист. М. Ф. Кудлацкая
(кафедра информатики и веб-дизайна, БГТУ)

МОНИТОРИНГ УПОМИНАНИЯ БРЕНДОВ В СОЦИАЛЬНЫХ СЕТЯХ

Компьютерные социальные сети становятся все более популярными в современном информационном обществе. Посредством таких сетей осуществляется обмен огромным количеством информации. Анализ информации, существующей в социальных сетях, позволяет делать определенные выводы о процессах, протекающих в обществе, прогнозировать поведение его участников, моделировать социальное