

УДК 004.42

**ЦЕЛЕСООБРАЗНОСТЬ ИСПОЛЬЗОВАНИЯ ИГРОВЫХ СЕРВИСОВ,
ПРЕДОСТАВЛЯЕМЫХ APPLE ДЛЯ РАЗРАБОТКИ ПРИЛОЖЕНИЙ****А.И. ВАЛЬКОВИЧ***(Представлено: канд. техн. наук И.Б. БУРАЧЕНОК)*

Осуществлен сравнительный анализ сервисов, предоставляемых компанией Apple для разработки приложений для платформы iOS и собственных решений. Также автор делает вывод о целесообразности и выгоде от использования внутренних сервисов платформ в процессе разработки небольших приложений.

Современное приложение состоит из большого количества подсистем. Разработчику требуется сохранять данные в облаке, объединять пользователей в общие комнаты, присылать уведомления и выполнять множество других действий, которые являются второстепенными для приложения, но без них продукт окажется скучными не будет выделяться функционалом среди конкурентов. На этапе проектирования возникает вполне очевидный и важный вопрос – реализовывать весь функционал самому или использовать готовое решение? Безусловно, однозначного ответа на этот вопрос нет и не будет, все зависит от ситуации. Однако нужно понимать, что практически для любой задачи можно найти подходящее готовое решение, которое пусть и не полностью решит проблему, но избавит разработчика от необходимости разбираться во всех тонкостях и позволить сэкономить самый главный и ценный ресурс – время.

Компания Apple известна, в первую очередь, как компания, которая продает доступ к своей экосистеме. Большую часть стоимости ее устройств составляют сервисы и программное обеспечение, доступ к которым предоставляется как пользователям, так и разработчикам контента. В первую очередь это касается сервисов и функционала, который доступен вместе с подпиской на статус разработчика.

Для решения проблемы хранения пользовательских данных в облаке, можно выбрать одно из следующих направлений: развернуть собственную инфраструктуру или воспользоваться сервисом iCloud.

Разработка собственной инфраструктуры потребует поиска и покупки копии лицензионного ПО, предназначенного для хранения данных в БД, а также аренды или установки собственных серверов, которые необходимо расположить в разных частях света, и настроить репликацию между ними. Задача не из простых, тем более что требуемый продукт может требовать достаточно простой функционал и от БД нужно будет только сохранять и возвращать данные.

В случае если будет решено использовать iCloud, то разработчику не нужно заниматься развертыванием серверов, постоянным мониторингом системы и оплатой серверного оборудования – всеми этими задачами будет заниматься Apple. iCloud универсальная система, которая поддерживает все устройства, работающие на операционных системах iOS и macOS. Схема поддерживаемых сервисом iCloud устройств представлена на рисунке 1.

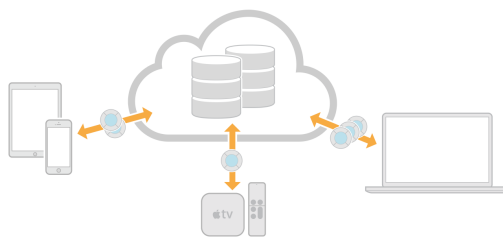


Рисунок 1. – Схема поддерживаемых сервисом iCloud устройств

Однако не стоит рассчитывать, что бесплатное решение от Apple будет во всем лучше собственной разработки. CloudKit имеет ряд жестких ограничений, которые могут быть критичны для многих приложений. Среди ограничений:

- 100 Мб дискового пространства;
- 2 Гб трафика;
- 40 запросов в секунду [1].

Данные ограничения с легкостью снимаются платной подпиской, но даже этого достаточно для большинства приложений, что делает данную технологию весьма полезной при ограниченном бюджете.

Практически любая современная компьютерная игра построена на взаимодействии между пользователями. Благодаря программистам из Apple, у iOS разработчиков есть возможность быстро и легко

оживить свою игру, добавив в нее элементы социальности и позволив сравнивать свои результаты с другими игроками или даже играть против них в реальном времени. Проанализируем три такие технологии.

Для реализации функционала таблицы рекордов игрока при помощи GameCenter от Apple, требуется создать таблицу в панели разработчика и одной строчкой кода отправлять результаты на сервер. Еще одной строчкой кода можно открыть таблицу или получить данные для отображения в собственном интерфейсе. Таблицы рекордов это один из самых мощных и действенных элементов социальности и разработка Apple значительно облегчает работу с большим количеством данных [2]. Пример таблицы рекордов представлен на рисунке 2.

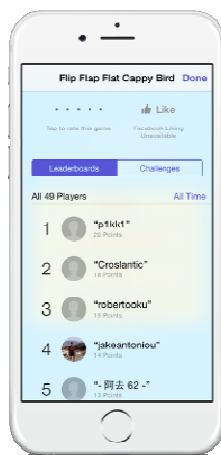


Рисунок 2. – Пример таблицы рекордов

Механизм достижений игроков, аналогично таблице рекордов, достаточно реализован Apple и доступен в панели управления игрой iTunesConnect. Разработчику необходимо создать достижение и вызвать событие его получения из игры в нужный момент времени. Сколько бы это заняло времени разрабатывая это с нуля, сложно представить, но Apple это сделали так, что практически все самые крупные разработчики мобильных игр отказываются от реализации собственного функционала в сторону встроенного сервиса достижений.

Одним из самых важных и сложных подсистем GameKit, является система игры «Игрок против игрока». Apple бесплатно предоставляет мощный функционал, на который многие компании тратят десятки тысяч долларов. При помощи данного сервиса, игроки могут играть против друг друга с разных устройств находясь в разных частях света, при этом имея минимальную задержку на передачу данных. Это один из самых спорных функционалов и многие предпочитают разворачивать свое собственное серверное решение, так как оно должно быть максимально гибким для разработки сложных и безопасных систем.

Проанализировав встроенные сервисы Apple, можно сделать **вывод**, что игровые сервисы позволяют реализовать базовые потребности разработчиков в необходимости внедрения функционала для социализации игрового процесса. Разработчики без труда могут реализовать игру против другого игрока, достижения и таблицы рекордов, а также позволяют хранить данные на удаленных серверах. Данный функционал предоставляется в рамках подписки разработчика и не требует дополнительных финансовых вложений, однако обладает не высокой гибкостью и не богатым функционалом. Собственные решения позволяют достичь большего уровня гибкости системы, однако их разработка и развертывание обходится в десятки раз дороже, чем встроенные сервисы Apple.

Таким образом, принято решение в дальнейшем использовать встроенные сервисы Apple в небольших и не сложных проектах, так как они позволяют значительно снизить стоимость разработки благодаря высокой скорости разработки.

ЛИТЕРАТУРА

1. CloudKit – AppleDeveloperPortal [Электронный ресурс]. – Режим доступа: <https://developer.apple.com/icloud/cloudkit/>. – Дата доступа: 24.09.2017.
2. Leaderboard – UnionAssets [Электронный ресурс]. – Режим доступа: <https://unionassets.com/iosnative/leaderboards-45/>. – Дата доступа: 25.09.2017.