

За наявності високорівневих мов маніпулювання даними типу реляційної алгебри і SQL в настільних автоматизованих систем управління підтримувалися низькорівневі мови маніпулювання даними на рівні окремих рядків таблиць. У настільних автоматизованих систем управління були відсутні засоби підтримки посилальної і структурної цілісності бази даних. Ці функції повинні були виконувати додатки, проте незначність засобів розробки додатків іноді не дозволяла це зробити, і в цьому випадку ці функції повинні були виконуватися користувачем, вимагаючи від нього додаткового контролю при введенні і зміні інформації, що зберігається в базах даних.

Таким чином, використання баз даних у перших автоматизованих системах управління мало примітивний характер, проте послугувало значним стимулом для подальшого розвитку. Початкові етапи комп'ютеризації ділових процесів підприємства характеризувалися інформаційною підтримкою окремих бізнес-операцій. Збір інформації був основоположною завданням системи автоматизації, в той час як аналіз отриманих даних проводився вручну. Зовсім не так давно в розпорядженні підприємств були тільки системи оперативного та бухгалтерського обліку, які були покликані вирішувати локальні завдання, і не були повнофункціональними системами управління підприємством.

#### **Список використаних джерел:**

1. Жарська І. О. Інформаційні системи й технології в управлінні / І. О. Жарська // Маркетинг і менеджмент інновацій. – 2013. – № 1. – С. 100-111.
2. Берко А. Ю. Організація баз даних: навч. посібник / А. Ю. Берко, О. М. Верес, В. В. Пасічник. – Львів: Магнолія, 2008. – 456 с.
3. Курташова І. В. Автоматизовані системи управління / І. В. Курташова // Актуальні проблеми економіки. – 2013. – № 3. – С. 231-237.

**Сурай О.В.**

*студент,*

*Національний технічний університет України  
«Київський політехнічний інститут»*

### **СРАВНЕНИЕ КРОССПЛАТФОРМЕННЫХ ФРЕЙМВОРКОВ ДЛЯ РАЗРАБОТКИ МОБИЛЬНЫХ ПРИЛОЖЕНИЙ**

Фреймворк (от англ. framework – каркас, структура) – это программное обеспечение, облегчающее разработку и объединение разных компонентов большого программного проекта. В свою очередь кроссплатформенное программное обеспечение (фреймворк) подразумевает ПО, которое работает более чем на одной аппаратной платформе или операционной системе.

Сегодня разработка мобильных приложений в практически всех компаниях по производству софта подразумевает использование именно кроссплатформенных фреймворков, так как пользователи ожидают приложения

под различные ОС. Именно поэтому разработчики теперь пишут свои продукты не отдельно для iOS, Android или Windows, а одно, но для всех ОС сразу.

Многие всемирно известные компании-разработчики ПО давно уже имеют в линейке своей продукции собственный фреймворк для разработки мобильных приложений, как например у IBM – IBM Worklight, а у Adobe – Adobe PhoneGap. Но все эти фреймворки имеют свои отличия, слабые и сильные стороны, одни подходят для решения отдельных специфических задач лучше, чем другие. Поэму ниже представлена сравнительной характеристика существующих кроссплатформенных фреймворков.

#### Xamarin.

Основан на open-source реализации платформы.NET – Mono. Данная реализация включает в себя собственный компилятор C#, среду выполнения, а также основные.NET библиотеки. Предназначен для кроссплатформенной разработки мобильных приложений под iOS, Android и Windows Phone с использованием языка C#. Среда разработки – Xamarin Studio, которая также является кроссплатформенной IDE и работает как на MAC OS X, так и на Windows; имеет практически весь функционал Visual Studio. Также есть возможность использовать саму Visual Studio в чистом виде, но только при наличии business-лицензии. Фреймворк имеет компиляторы для iOS и Android, а также Xamarin.iOS и Xamarin.Android – библиотек классов для C#, которые предоставляют разработчику доступ к iOS SDK и Android SDK соответственно.

Фреймворк имеет отдельные компиляторы для iOS и Android, так как с точки зрения исполнения приложений между этими ОС существует отличие, а именно – способ их предварительной компиляции. Но при этом Xamarin имеет преимущество в том, что код компилируется сразу в нативный (родной) код, поэтому поведение, вид и производительность такая же, как и у родных для ОС приложений.

Имеет в наличии инструмент Xamarin.Forms, который позволяет создавать UI из набора визуальных элементов, описываемых на языке разметки XAML, который отображается в визуальные элементы соответствующей операционной системы. Для проверки работоспособности созданных приложений используется Xamarin Test Cloud – сервис автоматизированного тестирования на сотнях виртуальных мобильных устройствах.

#### Adobe PhoneGap.

Данный фреймворк является open-source дистрибутивом Apache Cordova, спонсируется самим Apache и Adobe. Позволяет просто создавать приложения при помощи web-технологий, таких как HTML, CSS и JavaScript.

Приложение работает как обычная web-страница внутри WebView, именно поэтому все и строится на основе HTML, CSS и JS. Но при этом PhoneGap API дает возможность использовать все устройства в приложении – камера, звук, GPS, файловая система, контакты и др. Приложения, созданные с помощью этого фреймворка, используют web представление для передачи контента.

Приложение можно скомпилировать под любую мобильную платформу, включая такие, как Tizen, Bada, Firefox OS. Для этого применяется облачный

сервис PhoneGap Build, с помощью которого все можно сделать практически в пару кликов.

Разработка приложений с помощью PhoneGap позволяет внедрить их в нативные – в результате получаем гибридные приложения. Но в этом и кроется минус использования фреймворков на основе web-технологий – приложения не являются нативными, то есть родными для ОС, а поэтому работают более медленно и не всегда стабильно, а также имеют функциональные ограничения. Именно поэтому подобные фреймворки не стоит использовать для достаточно серьезных проектов.

PhoneGap является идеальным вариантом для начинающих разработчиков мобильных приложений, так как порог вхождения довольно низок – достаточно знать лишь основные web-технологии, и это открытый и бесплатный продукт.

IBM MobileFirst (Worklight).

Данный фреймворк, как и предыдущий Adobe PhoneGap, создан на основе Apache Cordova, но компанией IBM. Предоставляет возможность создавать web-приложения, а также гибридные (с возможностью пересылать сообщения и данные между нативными и web модулями) как и предыдущий фреймворк.

MobileFirst Studio, IDE фреймворка, позволяет мобильным разработчикам в полной мере использовать функциональные возможности HTML5, а также дальнейшего расширения этих возможностей с помощью служебных программ и механизмов. Среди них – шифрование локально хранимых данных, при чем безопасность обеспечивается на каждом уровне – пользователь, устройство, данные, приложение; аутентификация в автономном режиме; сочетание HTML5 и родного кодирования; возможность интеграции с другими фреймворками, как например с PhoneGap, Sencha Touch, jQuery. Также имеется поддержка USSD-сервисов и сервис аналитики приложения.

Appcelerator Titanium.

Фреймворк позволяет создавать приложения, которые выглядят и ведут себя как нативные, но сами по себе написаны с использованием JavaScript (частью фреймворка является Titanium SDK, который поддерживает разработку на языке JavaScript). В данном случае js-код в режиме runtime транслируется в нативные view.

Titanium – отличная разработка, в которой можно найти все, что нужно для создания гибридных мобильных приложений. Для работы с фреймворком необходимо установить Titanium studio, а Titanium SDK содержит несколько API платформ и Cloud сервис, что работает как хранилище. Он идет с независимой платформой API, которая упрощает доступ к мобильному устройству.

UI можно создавать отдельно для каждой платформы с использованием фреймворка Alloy (интегрированный MVC фреймворк, который использует XML и CSS-подобный синтакс). Само создание отдельных UI для каждой операционной системы усложняет разработку и заметно снижает объем переисполняемого кода, но вся бизнес-логика, модель и само ядро приложения все же остаются одинаковыми для всех платформ.

Фреймворк имеет магазин плагинов и компонентов, из которых меньшая часть – бесплатные. Имеется все, что необходимо для создания приложений: аналитика, реклама, облачные хранилища, социальные сети и так далее. Аналитическая платформа обеспечивает мониторинг данных о приложении в режиме реального времени, мониторинг производительности, логов, поломок, и самого процесса создания приложения. Тесты полностью автоматизированы.

Еще одна особенность фреймворка – встроенные коннекторы к самым популярным enterprise-платформам (Salesforce, SAP, Oracle, Microsoft Dynamics и SharePoint) и к самым популярным приложениям (LinkedIn, PayPal, DropBox, Facebook, Twitter). Существует также возможность создавать свои собственные коннекторы к любым сервисам.

#### Kony Platform.

Предназначен для создания web, гибридных и нативных приложений, написанных с единым кодом, для смартфонов, планшетов и десктопных устройств. Поддерживается полный цикл создания приложения – дизайн, разработка, тестирование, развертывание и управление кроссплатформенным приложением.

Фреймворк обеспечивает возможность показывать и делиться прототипами и уже готовыми приложениями между дизайнерами и разработчиками, при чем с возможностью комментировать и обсуждать проблемные места на макетах. Превью приложений в реальном времени; удобная реализация для обновления приложений, а также для управления настройками и версиями. Также есть возможность отсылать сообщения и уведомления пользователям (push services). Имеется система отчетов и аналитики, логи поломок. Сервисы синхронизации позволяют подключаться к любым сторонним enterprise-сервисам.

Существует еще ряд кроссплатформенных фреймворков для создания мобильных приложений, который работают по тем же схемам – либо создание практически нативных приложений, либо создание гибридных с помощью web технологий. Выше представлены наиболее популярные среди разработчиков фреймворки. При чем достаточно большую популярность в последнее время набирают те, которые предполагают использования именно web технологий, так как HTML 5 развивается очень быстро и минимальное знание его уже дает возможность войти в сферу разработки мобильных приложений.

#### **Список использованных источников:**

1. Scott Olson, John Hunter, Ben Horgen, Kenny Goers. Professional Cross-Platform Mobile Development in C# // Indiana, 2012. – 360 с.
2. Shah Rukh Humayoun, Stefan Ehrhart and Achim Ebert. Developing Mobile Apps Using Cross-Platform Frameworks: A Case Study // Germany, 2013. – 371 с.