

Android мобилно приложение за персонален библиотечен каталог в облачна среда

Валентина Антонова, Пламен Вълчев

Android Mobile Application for Personal Library Catalog in the Cloud Environment: *This report presents the design and development of the Android mobile application for personal library catalog in the cloud environment. Maintenance of data using a cloud service type BaaS (Backend as a Service). For the purposes of the application is chosen platform Parse, which offers a non-relational database - NoSQL (Not only Structured Query Language). Implemented the following functionalities: unambiguous identification of the user through integration with the social network facebook; adding a book to the user library by - barcode scanner, form, search in Google Books, by a button when viewing a book in the library on the user's facebook friend; review / edit / delete book.*

Key words: *BaaS (Backend as a Service), NoSQL (Not only Structured Query Language), Parse*

ВЪВЕДЕНИЕ

Мобилните устройства са неразделна част от ежедневието. От устройства, използвани за телефонни обаждания и съобщения, те се превръщат в мултифункционални устройства. Подобрили мрежи, атрактивни тарифни планове, мобилни плащания, високо скоростен интернет и наличие на услуги като мобилни приложения влияят значително на растежа на продажбите на смартфони. Търсенето на мобилни приложения също нараства. В настоящия доклад се разглежда разработката на Android мобилно приложение за обслужване на персонален библиотечен каталог в облачна среда.

ПРОЕКТИРАНЕ И РЕАЛИЗИРАНЕ НА МОБИЛНОТО ПРИЛОЖЕНИЕ

Избор на Система за Управление на База от Данни

При реализирането на мобилните приложения се налага необходимостта от централизация на базата от данни за да се дава възможност за обмен на информация между потребители, използващи различни устройства или за да са достъпни данните на потребителя от различни устройства и приложения. За поддръжката на данните, в разработката се използва облачна услуга от тип BaaS (Backend as a Service). Повечето BaaS доставчици предлагат нерелационни бази от данни – NoSQL (Not only Structured Query Language). Основните предимства на NoSQL базите пред релационните са високата скалируемост и производителност. NoSQL базите от данни предоставят механизъм за съхранение и възстановяване на данни, който използва свободен съгласуван модел. Общоприетата класификация на нерелационните бази от данни се базира на модела на данните:

- Колона – Hbase, Cassandra.
- Документ – MongoDB, CouchDB.
- Ключ – стойност – Azure, RocksDB.
- Граф – Allegro, Bigdata.

Някои от най-популярните доставчици на BaaS услуги са: Parse; Appcelerator; StackMob; Kinvey; Firebase; Telerik Backend Services.

За целите на приложението е избрана платформа Parse. Платформата Parse е позната още преди придобиването ѝ от Facebook през 2013 година. Предимства:

- Осигурява широк набор от различни услуги в един пакет.
- BaaS платформите са проектирани като високо скалируеми – не се налага нужда от модификация на приложението при увеличение на потребителите.
- Лесна модификация на backend частта на приложението.
- Пести време и ресурси.

Платформата Parse поддържа: облачно-базирано хранилище (Cloud-Based Storage), позволяващо съхранение на nk2a изображения; управление на потребители; геолокации; социална интеграция; конфигурация; мониторинг на системата в реално време; изготвяне на различни отчети и други. Parse предлага голям брой от клиентски SDK-та: iOS, Android, JavaScript, OS X, Unity, PHP, .NET, Ruby, Python, ActionScript, .NET + Xamarin и други [1,3,4].

Проектиране на NoSQL базата от данни

Проектирането на релационни бази от данни се базира основно на нормализация на данните за да се намали или елиминира дублирането на данни в базата, докато основните техники прилагани при проектиране на NoSQL бази от данни целят да денормализират данните [2] [5]. Денормализацията е полезна:

- За намаляване на броя на заявките, като се групират необходимите данни за заявката на едно място.
- При сложни заявки, включващи съединения (joins).

В резултат от прилагане на денормализацията нараства обема на данните, което е недостатък.

Възможността за съхранение на данни без определена структура позволява формирането на класове със сложни вътрешни структури – вложени обекти с различни структури и други. Това води до:

- Намаляване на отношенията от тип един към много, чрез влагане на обекти и намаляване на необходимите съединения за извличане на данните.
- Замаскиране на „техническите“ разлики между обектите и моделиране на хетерогенни обекти, чрез използване на една колекция от документи или една таблица.

Проектиране на приложението

Архитектурата на приложението се основава на популярния и широко разпространен архитектурен шаблон MVC (Model-View-Controller), където се разделя бизнес логиката от графичния интерфейс и данните:

- Модел – ядрото на приложението, определено от областта, за която се разработва. Обикновено това са данни от реалния свят, които са моделирани и подлежат на модификация и извличане.
- Изглед – тази част от MVC приложенията е отговорна за визуализацията на данните от модела.
- Контролер – тази част от MVC приложенията е отговорна за извличането на данни от модела, предварителна обработка на данните и зареждане на изгледа.

Основните предимства на MVC шаблона са:

- Моделът е независим от контролера и изгледа.
- Моделът може да бъде планиран и осъществен независимо от другите части на системата.
- За един и същи модел могат да бъдат осъществени различни изгледи.
- Контролерът и изгледът могат да бъдат променени, без да се налагат модификации в модела.

Функционалност на приложението

За да се инсталира приложението е необходимо устройство с камера и операционна система Android 5.0 (Lollipop)/по-нова версия. За вписване на потребител в приложението е необходимо той да има facebook профил, инсталирано приложение - facebook или facebook lite и постоянна интернет връзка.

Основният екран на приложението се състои от три части:

- в горния край на екрана се намира основната навигация;
- в средата на екрана се визуализира избрания екран;

- в долния край на екрана се намира слайдинг панел.

Навигацията се осъществява с помощта на четири бутона, осигуряващи достъп до различните екрани (фиг. 2).



Библиотека

Осигурява достъп до персоналната библиотека. Информацията за всяка книга включва: заглавие, подзаглавие, изображение на корица, автори, категории, брой страници, ISBN, описание. Всяка книга може да се добави към списък с любими книги, да се сподели, да се редактира или изтрие.



Любими

Осигурява достъп до списъка с любими книги.



Приятели

Осигурява достъп до библиотеките на facebook приятелите.

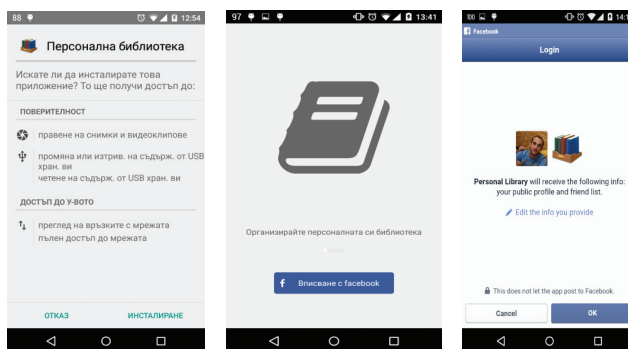


Търсене

Осигурява достъп до търсенето на книги в Google Books.

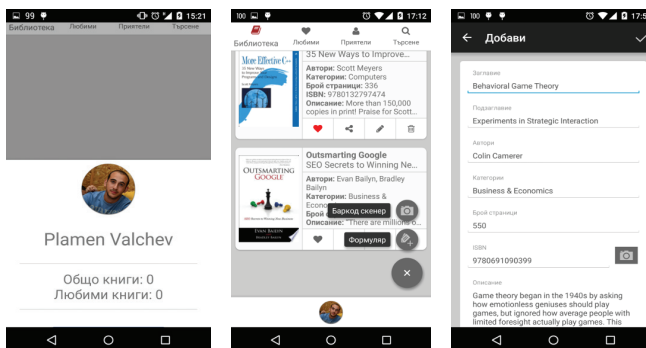
Реализирана е функционалност за:

- Еднозначна идентификация на потребител чрез интеграция със социалната мрежа на facebook (фиг. 1).



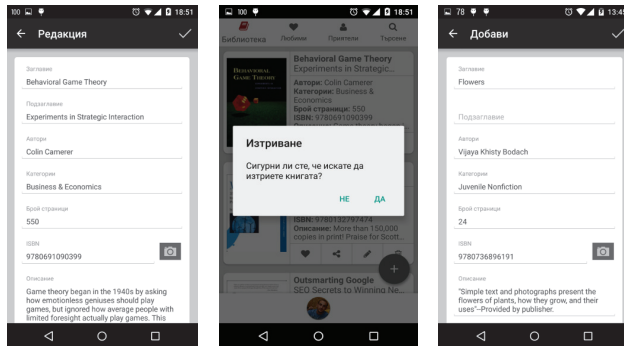
Фиг. 1. Инсталиране, вписване на потребител и достъп до facebook профил

- Добавяне на книга в библиотечния каталог на потребителя чрез (фиг. 2):



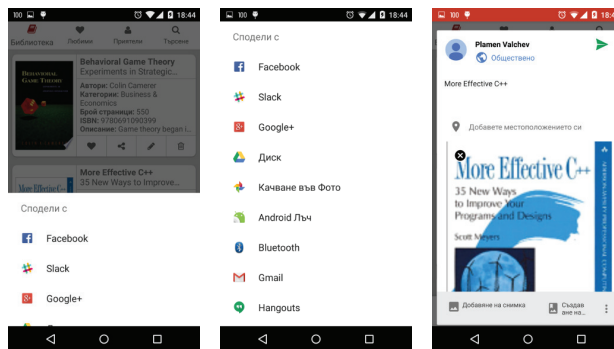
Фиг. 2. Основен екран, библиотека и формуляр за добавяне на книга

- Баркод скенер.
- Форма за въвеждане.
- Търсене в Google Books.
- Бутон при преглед на книга в каталога на facebook приятел на потребителя.
- Преглед/редакция/изтриване на книга (фиг.3).



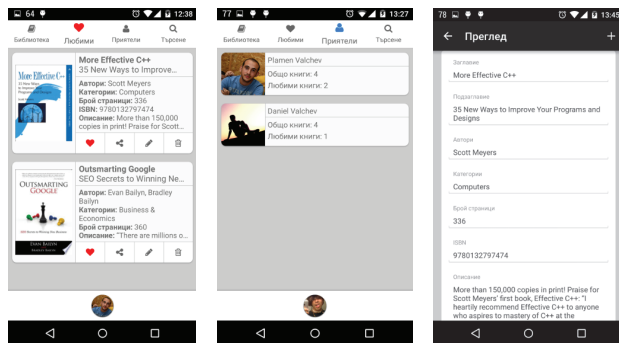
Фиг. 3. Преглед, редакция, изтриване и добавяне на книга

- Споделяне на данни за книга с приложения – facebook, Google+ и др. (фиг. 4).



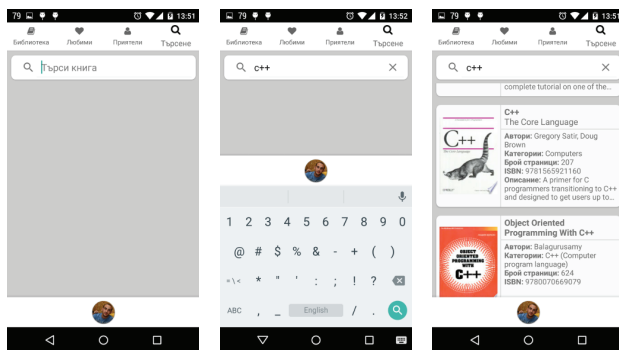
Фиг. 4. Споделяне на книга

- Преглед на библиотеките на facebook приятелите на потребителя.
- Поддръжка на списък с любими книги (фиг. 5).



Фиг. 5. Списък с любими книги, списък с приятели и преглед на книга

- Търсене в Google Books (фиг. 6).



Фиг. 6. Търсене в Google Books

ЗАКЛЮЧЕНИЕ

Нарастващото търсене на мобилни приложения налага необходимост от използване на иновативни подходи за разработка на софтуер. Предимства на BaaS услугите са съкратеното време за разработка и разходите, които се спестяват от закупуването и поддръжката на собствена инфраструктура.

Приложението може да бъде развито с добавяне на функционалности за:

- Съхранение и четене на електронни книги.
- Покупка на книги.
- Рейтинг.
- Импорт и експорт на информация.

ЛИТЕРАТУРА

- [1] <https://www.parse.com/docs/android/guide>
- [2] <https://highlyscalable.wordpress.com/2012/03/01/nosql-data-modeling-techniques>
- [3] <http://developer.android.com>
- [4] <https://developers.facebook.com/docs/android>
- [5] <https://www.kidscodex.com/database-design>

За контакти:

инж. Валентина Антонова, Катедра “Компютърни науки и технологии”,
Технически университет Варна, тел.: 052-383 638, e-mail: valyvarna@yahoo.com
инж. Пламен Вълчев, pvalchev@abv.bg

Докладът е рецензиран.