

Bogdan Kwiatkowski [et al.]

Organizacyjne aplikacje mobilne jako źródło informacji

Edukacja - Technika - Informatyka 5/2, 348-353

2014

Artykuł został opracowany do udostępnienia w internecie przez Muzeum Historii Polski w ramach prac podejmowanych na rzecz zapewnienia otwartego, powszechnego i trwałego dostępu do polskiego dorobku naukowego i kulturalnego. Artykuł jest umieszczony w kolekcji cyfrowej bazhum.muzhp.pl, gromadzącej zawartość polskich czasopism humanistycznych i społecznych.

Tekst jest udostępniony do wykorzystania w ramach dozwolonego użytku.

**Bogdan KWIATKOWSKI, Robert PEKALA, Bogusław TWARÓG,
Zbigniew GOMÓŁKA, Mateusz MICHNOWICZ**
Uniwersytet Rzeszowski, Polska

Organizacyjne aplikacje mobilne jako źródło informacji

Wprowadzenie

W ciągu kilku ostatnich lat zaobserwowano znaczący wzrost liczby urządzeń mobilnych. Obecnie urządzenia te zastępują papierowe kalendarze, notesy czy książki. Szybki dostęp do danych, informacji jest niekiedy konieczny, pozwala efektywnie wykorzystać czas, co w przypadku studentów jest bardzo istotne. Taką potrzebę zaobserwowano przy wyszukiwaniu informacji związanych z tokiem studiów na Wydziale Matematyczno-Przyrodniczym Uniwersytetu Rzeszowskiego. Rozwiązaniem problemu utrudnionego dostępu do danych z poziomu smartphona bądź tabletu było przygotowanie dedykowanego programu na urządzenie mobilne, w którym zostały zebrane niezbędne informacje, ważne zarówno z punktu widzenia studenta, jak i pracownika naukowego Uniwersytetu Rzeszowskiego. W tym celu stworzono aplikację pazUR, czyli „Podręczny Almanach Z Uniwersytetu Rzeszowskiego”.

1. Interfejs aplikacji „pazUR”

Budowa interfejsu aplikacji została poprzedzona sondażem wykonanym wśród studentów dotyczącym ich oczekiwań związanych z tworzoną aplikacją na urządzenia mobilne. Obecnie, w kwestii projektowania interfejsu, zaleca się stosowanie odpowiednich wzorców. Wyróżnia się kilka podstawowych wzorców [Neil, *Mobile Design...*], takich jak siatka, lista wypunktowana czy zakładki. Pośród tych wzorców wybrano układ z wykorzystaniem zakładek, ponieważ przyszły użytkownik aplikacji od razu będzie miał dostęp do pożądaných treści – nie będzie musiał szukać konkretnych funkcjonalności w gąszczu ikon, a dodatkowo, dzięki takiemu rozwiązaniu, programista-deweloper dostaje więcej przestrzeni, którą może wykorzystać na umieszczenie poszczególnych funkcjonalności. Minusem takiego rozwiązania jest ograniczona liczba zakładek, jaka może być wprowadzona do aplikacji – każdy kolejny przycisk zmniejsza obszar, w który użytkownik powinien kliknąć, jeżeli chce się przenieść do dalszej funkcjonalności. Przy projektowaniu aplikacji pod urządzenia z systemem operacyjnym Android należało dodatkowo trzymać się odpowiednich wzorców projektowania interfejsu zalecanych przez Google [<http://developer.android.com/design/patterns/index.html>]. W swojej dokumentacji – Android sugeruje używanie

tak zwanego action bar – jest to swoisty nagłówek aplikacji znajdujący się w jej górnej części, który pozwala ją zidentyfikować użytkownikowi (poprzez logotyp oraz nazwę aplikacji) oraz wspomaga jej nawigację poprzez umieszczenie dodatkowych odnośników. Mając na uwadze wskazówki zamieszczone w dokumentacji, zaprojektowano wstępny wygląd interfejsu aplikacji. Wprowadzony został nagłówek, główna zawartość aplikacji oraz menu z zakładkami.

2. Wybór technologii tworzenia aplikacji

Większość obecnie powstających aplikacji mobilnych jest oparta w głównej mierze na języku programowania obiektowego, jakim jest Java. Jednak w przypadku tworzenia aplikacji pazUR wykorzystano takie języki, jak HTML, CSS oraz JavaScript. Może się wydawać, że te trzy technologie mają niewiele wspólnego z tworzeniem aplikacji mobilnych, jednak dzięki dodatkowym, zewnętrznym bibliotekom mogą stać się równie potężnym narzędziem co sam język Java. Co więcej, wykorzystując JavaScript, można łatwiej i prościej tworzyć skrypty, które zostaną wykorzystane w aplikacji.

Aplikacja oparta na technologiach internetowych (webowych) ma więcej zalet aniżeli aplikacja napisana w języku Java. Przykładowo, aktualizacja aplikacji o nowe dane lub zmianę istniejących może odbywać się poprzez zmianę pliku po stronie serwera – użytkownik potrzebuje tylko dostępu do Internetu w celu ich aktualizacji. W przypadku aplikacji wykorzystującej język Java wiązałyby się to z ponownym kompilowaniem całej aplikacji oraz zmuszeniem użytkownika do kolejnej instalacji.

Niestety, samo wykorzystanie wspomnianych trzech języków nie udostępni takich funkcjonalności urządzenia jak przechwytywanie obrazu z kamery czy możliwości wykorzystania akcelerometru. Co więcej, aby móc korzystać z takiej aplikacji, konieczne jest połączenie z Internetem, ponieważ sama aplikacja znajduje się na zewnętrznym serwerze, a nie w pamięci telefonu. Istnieje jednak sposób, aby taka aplikacja (która jest nazywana aplikacją webową, właśnie ze względu na duże wykorzystanie języków internetowych) stała się aplikacją natywną, czyli wykorzystującą wszystkie możliwości urządzenia mobilnego. W tym celu został stworzony szkielet aplikacji (z ang. *framework*) o nazwie PhoneGap.

Framework PhoneGap został pierwotnie stworzony przez firmę Nitobi (obecnie Adobe Systems) w 2009 r. Oprogramowanie to jest łącznikiem pomiędzy aplikacjami internetowymi a możliwościami, jakie dają nam urządzenia mobilne [Stark, Jepson 2012]. PhoneGap zawiera szkielety projektów aplikacji natywnych dla poszczególnych platform mobilnych, przy czym tworzone projekty są aplikacjami webowymi z podwyższonym poziomem uprawnień. Oznacza to, że dzięki wykorzystaniu tego narzędzia z poziomu JavaScriptu istnieje możliwość otrzymania dostępu do takich funkcji telefonu, jak akcelerometr, kamera, głośnik czy inne. Aplikacja utworzona z pomocą HTML, CSS i JavaScript jest w pewien sposób „opakowywana” w natywną aplikację – ale i nie

tylko. Framework w dużej mierze bazuje na platformie programistycznej Apache Cordova [<http://cordova.apache.org>], jednakże dodatkowo PhoneGap daje możliwość kompilowania aplikacji na inne platformy mobilne, takie jak iOS czy Bada w chmurze, czyli na serwerach firmy Adobe.

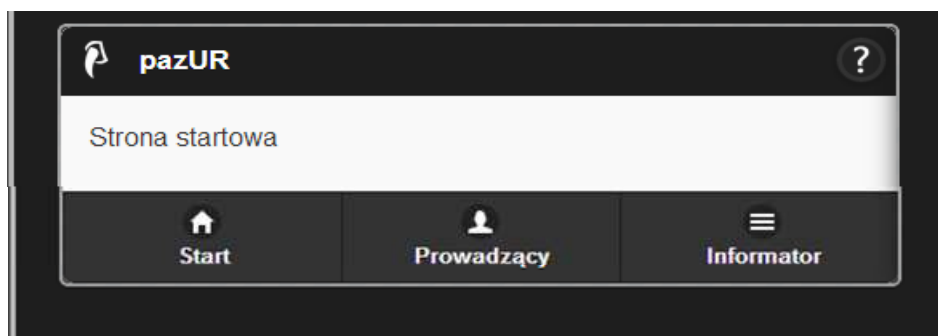
3. Tworzenie aplikacji

Bazując na wygenerowanym, podstawowym projekcie, przystąpiono do tworzenia właściwego kodu aplikacji pazUR. Najważniejszymi plikami w wygenerowanym projekcie są pliki znajdujące się w katalogu /www oraz w katalogach jemu podrzędnych. Główny szkielet aplikacji znajdował się w pliku index.html. W celu usprawnienia pracy przy pisaniu aplikacji został wykorzystany wcześniej wspomniany framework jQuery Mobile. Aby można było z niego korzystać w aplikacji, należało go pobrać go ze strony <http://jquerymobile.com/download> – na dzień 20 grudnia 2012 r. najnowszą wersją była wersja 1.4.0. Do działania niezbędne były biblioteki JavaScript w wersji zminimalizowanej, czyli takiej, której wielkość jest znacząco ograniczona – w tym przypadku wybrano plik jquery.mobile-1.4.0.min.js. Do poprawnego wyświetlania elementów niezbędny był podstawowy arkusz styli, również w wersji zminimalizowanej, ale z zaimplementowanym domyślnym, podstawowym wyglądem – w tym celu wybrano plik jquery.mobile-1.4.0.min.css.

Sama biblioteka jQuery Mobile mogłaby działać bez podstawy, jaką jest biblioteka jQuery. Główną bibliotekę pobrano ze strony <http://jquery.com/download>, jednak należało zwrócić uwagę, że nie mogła to być najnowsza wersja. Biblioteka jQuery Mobile jest tworzona w oparciu o poprzednie, stabilniejsze wydania głównej biblioteki jQuery, przez co nie jest kompatybilna z najnowszą wersją tejże. W przypadku wersji 1.4.0 jQuery Mobile bazowa biblioteka jQuery powinna być w wersji 1.9.1 – taką też pobrano, oczywiście również w wersji zminimalizowanej w celu jak największej optymalizacji aplikacji (plik o nazwie jquery-1.9.1.min.js).

Pobrane skrypty (jquery.mobile-1.4.0.min.js oraz jquery-1.9.1.min.js) umieszczono w katalogu /js stworzonej aplikacji, natomiast arkusz styli jquery.mobile-1.4.0.min.css w folderze /css. Bazując na tak przygotowanym kodzie, stworzono drugi plik HTML z taką samą strukturą, jednakże z inną nazwą pliku – help.html. W tym pliku zostanie umieszczona krótka instrukcja, jak powinno obsługiwać się aplikację. Po wstępnym przygotowaniu plików index.html oraz help.html podjęto przygotowania pliku index.css, który odpowiada za wygląd aplikacji. W związku z tym, że w aplikacji wykorzystano zewnętrzną bibliotekę, która stworzyła odpowiedni wygląd interfejsu, zawartość tego pliku usunięto – jednakże sam plik zachowano, gdyż był on potrzebny do stylowania elementów, które zostały umieszczone w pliku index.html. Gdy powyższe pliki zostały już odpowiednio przygotowane, zaczęto tworzyć kolejne strony aplikacji, począwszy od pliku index.html. Zgodnie z zaprojektowanym interfejsem po aplikacji

użytkownik będzie poruszał się za pomocą kart-zakładek. Dzięki zastosowaniu frameworka jQuery Mobile stworzenie takiego interfejsu opierało się na stworzeniu bloków <div> o odpowiednich atrybutach i identyfikatorach. Rys. 1 przedstawia finalny wygląd nagłówka i menu zakładowego aplikacji.

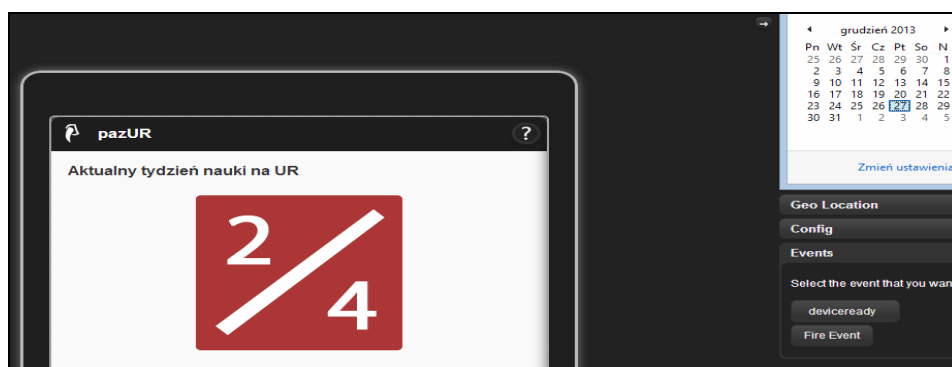


Rys. 1. Nagłówek i menu zakładowe aplikacji pazur

Źródło: opracowanie własne.

4. Skrypt obliczający bieżący tydzień zajęć na uczelni

Organizacja roku akademickiego na Uniwersytecie Rzeszowskim przewiduje podział tygodni nauki na tygodnie nieparzyste (oznaczane na planach zajęć jako 1/3) oraz na tygodnie parzyste (oznaczane jako 2/4). Parzystość bądź nieparzystość określa się na podstawie numeru tygodnia w roku. Przykładowo, tydzień zawierający dni 16–22 grudnia 2013 r. będzie tygodniem nieparzystym, gdyż jest to 51 tydzień roku. Na podstawie tych informacji została stworzona funkcja prezentująca aktualny tydzień nauki na Uniwersytecie Rzeszowskim w oparciu o język JavaScript (rys. 2).

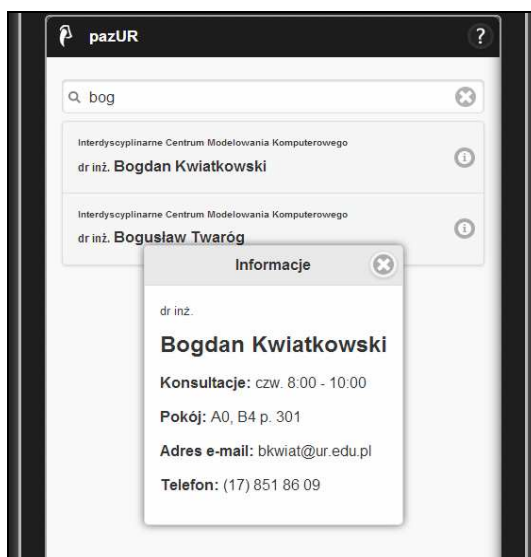


Rys. 2. Wyświetlenie parzystości tygodnia

Źródło: opracowanie własne.

5. Lista nauczycieli wraz z informacją o godzinach konsultacji

Kolejną funkcją, którą należało wprowadzić w aplikacji, to możliwość sprawdzenia godzin konsultacji nauczycieli akademickich prowadzących zajęcia wraz z danymi kontaktowymi do nauczyciela w postaci adresu e-mail wraz z numerem telefonu do jego gabinetu. Należy również wspomnieć, że tworzona lista będzie przechowywana na zewnętrznym serwerze, aby można ją było łatwo aktualizować – zamiast aktualizowania całej aplikacji, aktualizowany jest jeden plik na serwerze – przyspiesza to znacznie cały proces dostarczenia nowych danych (rys. 3).



Rys. 3. Informacje na temat wybranego nauczyciela

Źródło: opracowanie własne.

Podsumowanie

Prezentowana aplikacja oprócz pokazanych możliwości posiada jeszcze wiele innych, np.: zakładka z podstawowymi informacjami dotyczącymi władz Wydziału Matematyczno-Przyrodniczego, system automatycznych powiadomień studentów np. o godzinach rektorskich itp. Został również opracowany przystępny system pomocy w użytkowaniu niniejszej aplikacji. Dzięki takiej aplikacji studenci mogą w prosty i nowoczesny sposób uzyskać dostęp do interesujących ich danych za pomocą urządzenia mobilnego, takiego jak smartpho-ne czy tablet. Co więcej, przy wykorzystaniu programu pazUR znacząco poprawił się czas wyszukiwania konkretnych informacji, na przykład godzin konsultacji nauczycieli akademickich czy komunikatu dotyczącego bieżącego dnia tygodnia. Potwierdza to tezę, że w dobie tak ogromnej ekspansji urządzeń mobilnych ko-

rzystnie z takich pomocy jest bardzo sprzyjające w poprawnym oraz prawidłowym zorganizowaniu toku studiów. Aplikacja zaprezentowana w artykule ma charakter rozwojowy i dzięki technologii PhoneGap możliwe jest stworzenie aplikacji na inne platformy mobilne, takie jak Windows Phone czy Apple iOS, które powoli zwiększają swój udział w rynku urządzeń mobilnych. Warto również wspomnieć, że taka aplikacja, która jest dedykowana dla konkretnej uczelni wyższej, jest pewnym wyróżnikiem, który podkreśla jej rozwój oraz nowoczesność.

Literatura

<http://api.jquery.com>, jQuery API Documentation – dostęp: 22.12.13.

<http://cordova.apache.org/>, Apache Cordova – dostęp: 22.12.13.

<http://developer.android.com/design/patterns/index.html> – Patterns | Android Developers – dostęp: 25.12.13.

Neil T., *Mobile Design Pattern Gallery: UI Patterns for Mobile Applications*.

Stark J., Jepson B. (2012), *Android. Tworzenie aplikacji w oparciu o HTML, CSS i JavaScript*.

Streszczenie

W artykule przedstawiono aplikację mobilną przeznaczoną dla studentów oraz pracowników naukowych Uniwersytetu Rzeszowskiego. W celu zaprojektowania i stworzenia prezentowanej aplikacji wykorzystano takie technologie, jak HTML, CSS oraz JavaScript wraz z wykorzystaniem bibliotek PhoneGap, jQuery oraz jQuery Mobile.

Słowa kluczowe: Aplikacja mobilna, Android, technologie internetowe, PhoneGap.

Mobile application – Rzeszow University guide

Abstract

Aim of this article was to create mobile application dedicated to students and teachers of Rzeszow University. For this purpose was used technologies like HTML, CSS and JavaScript with additional libraries: PhoneGap, jQuery and jQuery Mobile.

Key words: mobile application, Android, Internet technology, PhoneGap.