

APLIKACJA MOBILNA „PRZYSTANEK”

Banaszkiewicz Karol, Dworak Mateusz, Łata Krzysztof, Repeć Grzegorz

Politechnika Lubelska, Wydział Elektrotechniki i Informatyki

Streszczenie. Artykuł opisuje budowę i działanie aplikacji mobilnej wykorzystującej wbudowany system GPS oraz Internet, która pełni rolę przenośnego rozkładu jazdy linii komunikacji miejskiej w Lublinie.

Słowa kluczowe: android, aplikacja, mobilna, przystanek, GPS, sqlite, internet

Mobile application „Przystanek”

Abstract. The article describes construction and performance of the mobile application using GPS and Internet connection. The application is supposed to substitute traditional bus schedule in Lublin.

Keywords: android, application, mobile, GPS, sqlite, internet

Wstęp

W dobie popularyzacji smartfonów i innych urządzeń mobilnych, trudno jest sobie wyobrazić kogoś, kto wychodząc z domu nie zabrałby ze sobą komunikatora mobilnego lub innego substytutu osobistego komputera. Każde, nawet najprostsze czynności, są już wykonywane z pośrednictwem smartfonów. Jako przykład niech posłuży zwykły kalkulator – czy statystyczny przechodzień (włączony studentów kierunków ścisłych) ma go przy sobie? Ależ oczywiście – jeśli tylko przyjmie się założenie, że aplikację "Kalkulator" zainstalowaną w telefonie można uznać za "posiadaną przy sobie". Gdyby jednak chodziło o fizyczne urządzenie, wyniki nie byłyby już tak optymistyczne.

Skoro telefon może zastąpić nie tylko notes i ołówek, ale także inne urządzenia elektroniczne, to dlaczego nie miałyby zastąpić wynalazku tak leciwego i niepewnego, jak rozkład jazdy autobusów? Aplikacja mobilna PRZYSTANEK wychodzi naprzeciw potrzebie przeniesienia tradycyjnego, papierowego rozkładu do smartfona.

1. Idea działania

Do poprawnego działania aplikacja wymaga jedynie urządzenia z systemem Android oraz modułem GPS (ang. Global Positioning System), zalecane jest również posiadanie połączenia z Internetem, w celu umożliwienia pobrania zawsze aktualnego rozkładu. Na podstawie położenia użytkownika określany jest najbliższy przystanek bądź przystanki, a po wybraniu jednego z nich lista dostępnych połączeń.



Fig. 1. Zrzut z aplikacji (wybór przystanku oraz linii)

Następnie ze strony Lubelskiego MPK (lub z wbudowanej bazy danych, w przypadku pracy w trybie offline rozkład dla wybranej linii.

Istnieje możliwość zwiększenia odległości, w jakiej wyszukiwane są przystanki, dzięki czemu użytkownik może sprawdzić rozkład jazdy autobusu, idąc na przystanek i jeśli zajdzie taka potrzeba, przyspieszyć kroku.



Fig. 2. Zrzut z aplikacji (wyświetlanie rozkładu)

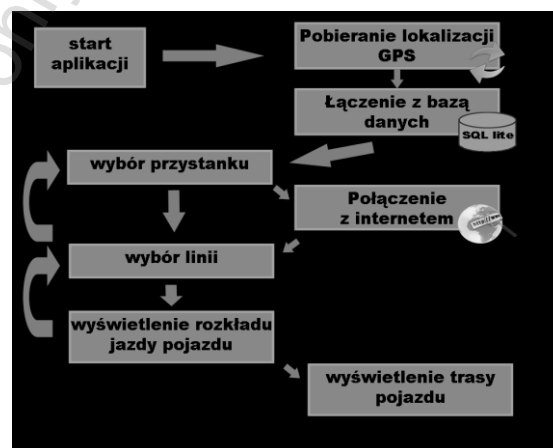


Fig. 3. Schemat działania aplikacji

2. A-GPS

A-GPS (ang. Assisted GPS) jest rodzajem systemu GPS wykorzystywanym w telefonii komórkowej. System ten wykorzystuje infrastrukturę operatorów GSM do skrócenia czasu niezbędnego na pierwsze znalezienie położenia. Aby móc z systemu korzystać, usługa musi być dostępna u operatora, w przeciwnym wypadku system GPS zadziała, jednak połączenie będzie trwało znacznie dłużej. W przypadku systemu A-GPS – czas połączenia jest naprawdę szybki, gdyż około 2-3 sekund (przy każdym kolejnym wyszukiwaniu – pierwsza inicjacja może potrwać nawet kilka minut) [1].

3. Android

Android jest darmową platformą dla telefonów komórkowych. Platforma Android zawiera zintegrowany zestaw oprogramowania dla urządzeń mobilnych. W skład Platformy

wchodzi system operacyjny, oprogramowanie pośrednie (ang. middleware), interfejs i aplikacje. Platforma Android nie jest związana z jakimikolwiek ograniczeniami praw autorskich – jest dostępna na licencji Apache License. Forma licencji, powszechny i darmowy dostęp do technologii jest gwarantem tego, że nie zabraknie na tą platformę aplikacji.

Cechy platformy Android:

- Application framework.
- Wirtualna maszyna Dalvik.
- Zintegrowana przeglądarka stron WWW.
- Zoptymalizowana grafika – biblioteki dla grafiki 2D, grafika 3D bazująca na OpenGL ES 1.0.
- SQLite – system zarządzania bazą danych.
- Wsparcie dla plików multimedialnych – mp3, mp4, JPG itp.
- Telefon GSM.
- Wsparcie dla technologii przesyłania danych – Bluetooth, EDGE, 3G i WiFi.
- Wsparcie dla urządzeń w telefonie takich jak kamera, GPS, kompas, akcelerometr.
- Bogate środowisko deweloperskie [2].

4. SQLite

Jest to system zarządzania bazą danych oraz biblioteka C implementująca taki system, obsługująca język SQL (ang. Structured Query Language). Została stworzona przez Richarda Hippa i jest dostępna na licencji public domain. Projekt został rozpoczęty w roku 2000. Biblioteka implementuje silnik SQL, dając możliwość używania bazy danych bez konieczności uruchamiania osobnego procesu RDBMS. W wielu zastosowaniach, a w szczególności w systemach wbudowanych, takie rozwiązanie jest najpraktyczniejsze. SQLite posiada również API do innych niż C języków programowania, a mianowicie: ActionScript, Perl, PHP, Ruby, C++, Delphi, Python, Java, Tcl, Visual Basic, platformy .NET i wielu innych; a także interfejs powłokowy. Baza może łączyć się przez ODBC. Zawartość bazy danych przetrzymywana jest w jednym pliku (do 2 TB). Baza SQLite jest utrzymywana na dysku przy użyciu B-drzew. Osobne drzewo jest używane dla każdej z tabel i każdego z indeksów. SQLite udostępnia transakcje ACID oraz implementuje większość standardu SQL 92. SQLite jest często wykorzystywany w większych aplikacjach oraz w systemach obsługi relacyjnych baz danych takich jak Kexi. Bazy danych zapisywane są jako pliki binarne. Ich bezpieczeństwo jest oparte na zabezpieczeniach oferowanych przez używany system plików. Istnieje też projekt oferujący szyfrowanie baz danych SQLite na bieżąco [3].

5. Realizacja

Naszym założeniem było stworzenie aplikacji przyjaznej użytkownikowi, miała być ona czytelna i prosta w obsłudze oraz zapewniać szybki dostęp do potrzebnych informacji. Założenie to udało się nam zrealizować, otrzymaliśmy prosty interfejs bez zbędnych dodatków obciążających działanie systemu na urządzeniu mobilnym, dzięki czemu możliwe jest działanie aplikacji w tle.

W przypadku pracy online pobierane są dane ze strony internetowej. Po zastosowaniu odpowiednich metod parsujących w kod HTML, możliwe jest uzyskanie niemal każdej informacji. Poniżej przykład metody pobierającej zawartość elementu objętego znacznikiem div o podanej w parametrze klasie :

```
List<TagNode> getDivByClass(String CSSClassname)
{
    List<TagNode> linkList =
        new ArrayList<TagNode> ();
    TagNode linkElements[] =
        rootNode.getElementsByName("div", true);
```

```
for (int i = 0;
    linkElements != null &&
        i < linkElements.length; i++)
{
    String classType =
        linkElements[i].getAttributeByName("class");

    if (classType != null &&
        classType.equals(CSSClassname))
    {
        linkList.add(linkElements[i]);
    }
}

return linkList;
}
```

Na urządzenie przesyłane są dane w postaci tekstowej co wpływa na szybkość uzyskania informacji oraz nie generuje wysokich kosztów za korzystanie z usługi Internet.

W przypadku pracy w trybie offline sprawa pobierania rozkładu jest bardzo prosta, ponieważ aplikacja zawiera wbudowaną bazę rozkładów jazdy (z możliwością aktualizacji), której obsługę umożliwiają podstawowe metody z biblioteki android.database.*.

Drugą wbudowaną bazą jest zbiór lokalizacji GPS wszystkich przystanków komunikacji miejskiej znajdujących się na terenie Lublina. Dzięki niej możemy sprawdzić na jakim przystanku się znajdujemy oraz wyświetlić dla niego rozkład jazdy pojazdów MPK. Dysponując odczytami z odbiornika GPS, czasami przydatna jest umiejętność obliczenia odległości pomiędzy dwoma punktami. Odczyt z odbiornika zawiera długość i szerokość geograficzną obliczenie odległości wymaga, więc specjalnego wzoru. Implementacja obliczania odległości pomiędzy dwoma punktami wygląda następująco :

```
double distance(Coordinate first, Coordinate
second)
{
    double a, c, d, R;
    double dLat, dLong;
    R = 6371;

    dLat = radians(second.x - first.x);
    dLong = radians(second.y - first.y);

    a = Math.sin(dLat / 2) * Math.sin(dLat / 2) +
        Math.cos(radians(first.x)) *
        Math.cos(radians(second.x)) *
        Math.sin(dLong / 2) * Math.sin(dLong / 2);

    c = 2 * Math.atan2(Math.sqrt(a), Math.sqrt(1 -
a));

    d = R * c;

    return d*1000;
}
```

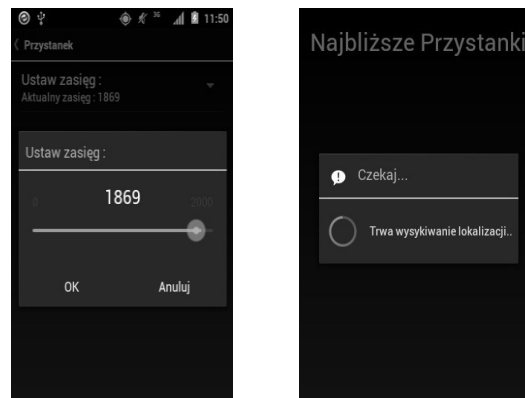


Fig. 4. Zrzut z aplikacji (ustawienia promienia skanowania)

Umożliwiamy użytkownikowi sprawdzenie rozkładu na przystanku, na którym aktualnie się nie znajduje. Ze względu na ich dość duże skupienie ograniczyliśmy możliwość ich skanowania do obszaru o promieniu nie większym niż 2 kilometry.

6. Analiza rynku

Nieprawdą byłoby stwierdzenie, że w dziedzinie rozkładów autobusowych nasza aplikacja jest jedyna i pozbawiona konkurencji. Nie my pierwsi, postanowiliśmy wprowadzić na rynek coś, co ułatwi ludziom korzystanie z komunikacji miejskiej. Aktualny rozkład jazdy jest przecież stale dostępny na stronie internetowej MPK, a na przystankach zaczęto umieszczać dodatkowo kody QR (ang. Quick Response), po zeskanowaniu których automatycznie zostaje się przeniesionym do odpowiedniej jej części. Istnieje także kilka komercyjnych aplikacji pozwalających na odczytanie rozkładu dla wybranego przystanku (najczęściej jest on pobierany z Internetu). Na tle w/w rozwiązań nasza aplikacja wyróżnia się przede wszystkim tym, że poza interesującą nas linią, niczego nie musimy wybierać – program sam określi, z którego przystanku zamierzamy jechać na podstawie danych GPS [4] [5]. Nie ma również konieczności uruchamiania przeglądarki internetowej i odnajdywania interesującego nas połączenia (liczba kliknięć zostaje zredukowana do absolutnego minimum), a ilość danych, jakie pobierane są z Internetu, jest nieporównywalnie mniejsza niż w przypadku skorzystania z witryny MPK, co często ma niemalże znaczenie dla użytkowników smart fonów.

Choć kody QR zapewniają podobną funkcjonalność i znacznie skracają czas dostępu do pożądaných danych, to jednak są one równie podatne na zniszczenia, co zwykle papierowe rozkłady umieszczone na każdym przystanku, a tych z kolei, jak nietrudno zauważyć, często brakuje, zwłaszcza na obrzeżach miasta, gdzie są rzadziej wymieniane.

Dodatkową wadą kodów QR i klasycznych rozkładów jest fakt, że aby je odczytać, musimy już znaleźć się na przystanku, a co za tym idzie, nie możemy z wyprzedzeniem stwierdzić, czy i jak bardzo powinniśmy się spieszyć, aby zdążyć na dany autobus.

Dzięki zastosowaniu GPS oraz pobieraniu danych zarówno z Internetu jak i z wbudowanej bazy, możliwe było zapewnienie większej funkcjonalności, a także zwiększenie wygody obsługi i niezawodności naszej aplikacji.

7. Perspektywy rozwoju

Teraz, gdy już otrzymaliśmy funkcjonalną wersję aplikacji, pojawiło się dla niej wiele potencjalnych ścieżek rozwoju, o których początkowo nie myśleliśmy. Posiadając gotowy algorytm parsujący rozkład jazdy ze strony MPK, możemy rozszerzyć obszar działania aplikacji o inne miasta na terenie naszego kraju w sposób niewidoczny dla użytkownika – wybór miasta, dla którego wyświetlany byłby rozkład, mógłby zostać dokonany na podstawie danych GPS. Oczywiście dodanie każdego miasta wiązałoby się z koniecznością żmudnego wprowadzania położenia poszczególnych przystanków do bazy danych oraz przeprowadzenia testów wymagających zwiedzenia znacznej części danego miasta.

Kolejnym możliwym sposobem na poszerzenie potencjalnego grona odbiorców jest przeniesienie aplikacji na inne platformy mobilne, przede wszystkim Windows Phone 7 firmy Microsoft oraz iOS firmy Apple. Rynek urządzeń mobilnych jest rozwija się bardzo dynamicznie, w związku z czym nierozsądne byłoby porzucenie innych platform tylko dlatego, że obecnie Android posiada jego większość.

Pod koniec 2010 roku miała miejsce premiera pierwszego telefonu z nowym systemem od Microsoftu – Windows Phone. Obecnie system ten jest już zainstalowany na kilkunastu modelach

smartfonów i szybko zdobywa popularność. Prognozy na przyszłość również są bardzo optymistyczne, a nadchodząca premiera Windows 8 z interfejsem Metro może dodatkowo przysporzyć mu fanów. To właśnie informacja o tej premierze, zaowocowała zawieszeniem prac nad wersją aplikacji PRZYSTANEK pod WP7.

Microsoft ogłosił porzucenie Silverlight (technologii stosowanej do tworzenia aplikacji użytkowych na telefony z ich systemem) i przejście na zupełnie nową platformę WinRT, która ma być wspólna dla komputerów osobistych i urządzeń mobilnych, w tym smart fonów.

Pomimo że iPhone w Polsce cieszy się dużo mniejszą popularnością, niż ma to miejsce np. w Stanach Zjednoczonych, użytkownicy iOS wciąż stanowią niemałą grupę docelową – jest to spowodowane głównie tym, że system ten jest zainstalowany na wszystkich urządzeniach Apple (z wyłączeniem komputerów osobistych), z których większość spełnia wymagania naszej aplikacji (nawet odtwarzacze muzyczne od 5-tej generacji wyposażone są w moduł GPS).

Planowane jest również dodanie automatycznego wyszukiwania linii autobusowych na podstawie obecnego położenia i podanej lokalizacji docelowej, a także, jeśli Lubelskie MPK wyrazi chęć współpracy, integrację z ich systemem umożliwiającym zakup biletu przez Internet [6].

8. Wnioski

W dniu dzisiejszym aplikacja działa jedynie w Lublinie, jednak jej rozbudowa na inne miasta jest zadaniem dość prostym. Wystarczy strona lokalnego przewoźnika oraz dodatkowe wpisy w bazie współrzędnych przystanków. Dodatkowo podczas testów w terenie potwierdziły się obawy odnośnie dokładności GPS używanego w telefonach.

Ograniczenia systemu powodują przekłamania w podawanej lokalizacji nawet do 15 metrów, dlatego ustawianie promienia w jakim szukamy przystanków na mniejszą odległość może niekorzystnie wpływać na działanie naszej aplikacji. W kwestii bezpieczeństwa aplikacja nie odbiega od najwyższych standardów. Wyznaczone za pomocą telefonu współrzędne używane są jedynie w obrębie aplikacji i nie są udostępnione innym programom na telefonie.

Tego typu zabezpieczenie zapewnia sam system, który separuje dane różnych aplikacji od siebie. Połączenie z Internetem z poziomu aplikacji jest tak samo bezpieczne jak wyświetlenie strony z rozkładem przez zwykłą przeglądarkę internetową. Dodatkowo aplikacja może działać przez długi czas nie korzystając z połączenia internetowego, wystarczy że co jakiś czas zsynchronizujemy bazę danych z serwerem.

Literatura

- [1] Wikipedia.pl
- [2] Android.com.pl
- [3] Developer.android.com
- [4] play.google.com
- [5] mmpk.info
- [6] mpay.pl

Karol Banaszekiewicz
e-mail: karol2034@gamil.com

Ukończył VIII LO im. Zofii Nałkowskiej w Lublinie. Aktualnie student III roku Wydziału Elektrotechniki i Informatyki na Politechnice Lubelskiej. Zainteresowania: programowanie Windows Phone 7.

