Introducere

Punerea problemei

Analiza stadiului actual în domeniul problemei

2.1 Sistemul de localizare a mijloacelor de transport

2.2 Studiu asupra soluțiilor existente

Bazele teoretice. Tehnologi folosite

Android OS

3.1. Android Studio

3.2. Google Maps Android API

4. Soluția propusă. Implementare

4.1 Flow Software. Organigrama sistemului.

4.2 Implementare.

4.2.1

5. Utilizarea aplicației

6. Realizare. Punere in functie si Testarea Aplicatiei.

7. Concluzii, contribuții și direcții de continuare a dezvoltării

8. Bibliografie

1.Introducere

Transportul public reprezintă una din principalele modalități de deplasare în interiorul unei localități, punând la dispoziție publicului larg posibilitatea de a ajunge dintr-un loc în altul, fără a avea un mijloc de transport personal chiar și între distanțele mari din mediul urban. În același timp folosirea transportului public poate fi un factor important pentru reducerea cheltuielilor personale daca este folosit în schimbul autovehicolului. Alte beneficii ale transportului public ar fi: decongestionarea traficului, reducerea emisiilor de carbon, crearea de locuri de muncă și multe altele.

1.2.Punerea problemei

Observăm că transportul public nu ar fi benefic doar pentru noi din punct de vedere financiar cât și mediului înconjurător, totuși în utilizarea lui intervin două probleme esențiale:

* cunoașterea traseelor și stațiilor mijloacelor de transport;
* cunoașterea momentului când mijlocul de transport ajunge într-o anumită stație.

Pentru rezolvarea primei probleme și anume informarea populației cu privire la rutele mijloacelor de transport în comun, cea mai frecventă metodă folosită este aceea de a afișa în majoritatea stațiilor și în interiorul mijloacelor de transport, o hartă ce prezintă toate traseele și statiile din acel oraș.

A doua problema este rezolvată în mare parte ca și prima, fiind afisat un orar prestabilit cu intervalele de circulație sau in cel mai bun caz un afișaj cu informații în timp real despre momentul cand va ajunge mijlocul de transport în acea stație. Totuși chiar daca ar părea că nevoia de informare a populației ar fi îndeplinită, o întrebare ar fi: cât de repede se actualizează hărțile din stații din tot orașul la o modificare de traseu a unei linii? Daca ar fi o modificare temporară de traseu cât de accesibilă va fi aceasta informație? Sau cum putem folosi transportul în comun daca nu știm unde sunt stațiile sau care ar fi cea mai apropiata stație de noi?

Observăm că doar aceste căi de informare aduc după sine îngreunarea folosirii transportului în comun prin incertitudinea unor informații și folosirea ineficientă a transportului prin asteptarea nedorită în stație. Pentru a intra în profunzimea problemei trebuie sa ne raportăm la un caz real și anume orașul Timișoara.

Orașul Timișoara dispune de tot ceea ce a fost descris mai sus cât și de mijloace suplimentare de informare. Site-ul Regiei Autonome de Transport Timișoara (RATT) conține o harta a transportului cu autobuze: urban, expres și metropolitan fără trasee deviate temporar, existând la momentul de față trasee neactualizate, mai conține o hartă a transportului electric tot fara trasee derivate temporar și nu în ultimul rând conține informații cu privire la timpul în care va ajunge un mijloc de transport într-o stație. Pe lânga aceasta mai există o aplicație mobilă android care ne oferă un acces mai ușor la informațiile afișate pe pagina oficială R.A.T.T. și anume prea datele cu privire la momentul cand ajunge un mijloc de transport în stație si le afițează sub forma unei liste.

Totuși pentru folosirea cea mai eficientă a transportului public în Timișoara ar fi nevoie de o sursa de încredere de informare, cu date actualizate în mod continu, să poată fi accesată din orice locație ne-am afla și în care să găsim atât traseul actual al unei linii cât și timpii de sosire a mijloacelor de transport în comun pentru fiecare stație.

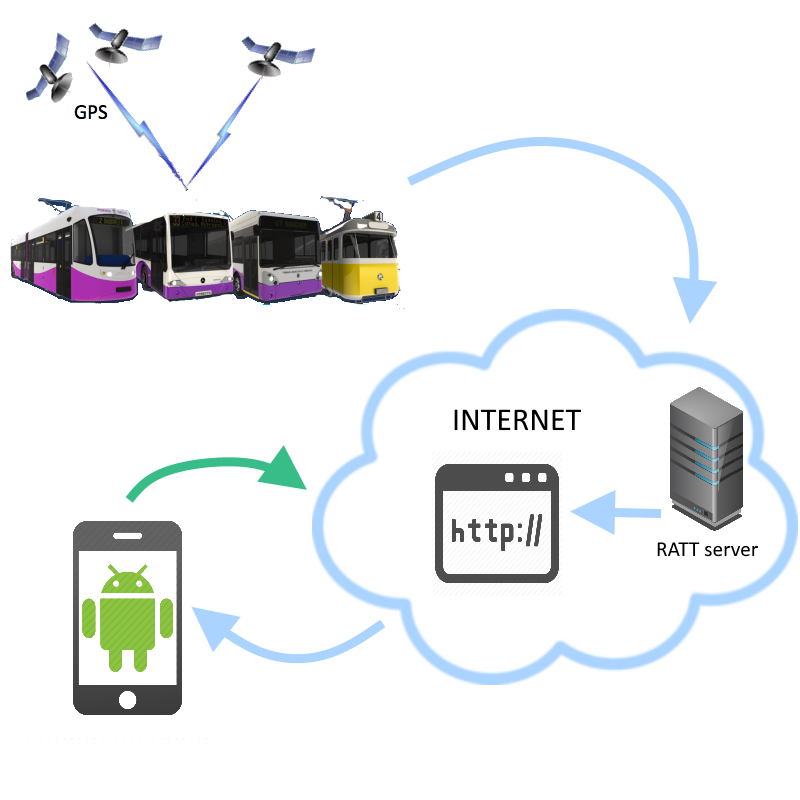
O soluție ar fi o aplicație mobilă ce ar avea acces întotdeauna la stațiile și traseele oficiale ale transportului în comun cu un mod interactiv de afisare pe o hartă dinamica a Timisoarei și la pentru fiecare stație timpul de sosire al fiecărui mijloc de transport.

2.Analiza stadiului actual în domeniul problemei

2.1 Sistemul de localizare a mijloacelor de transport

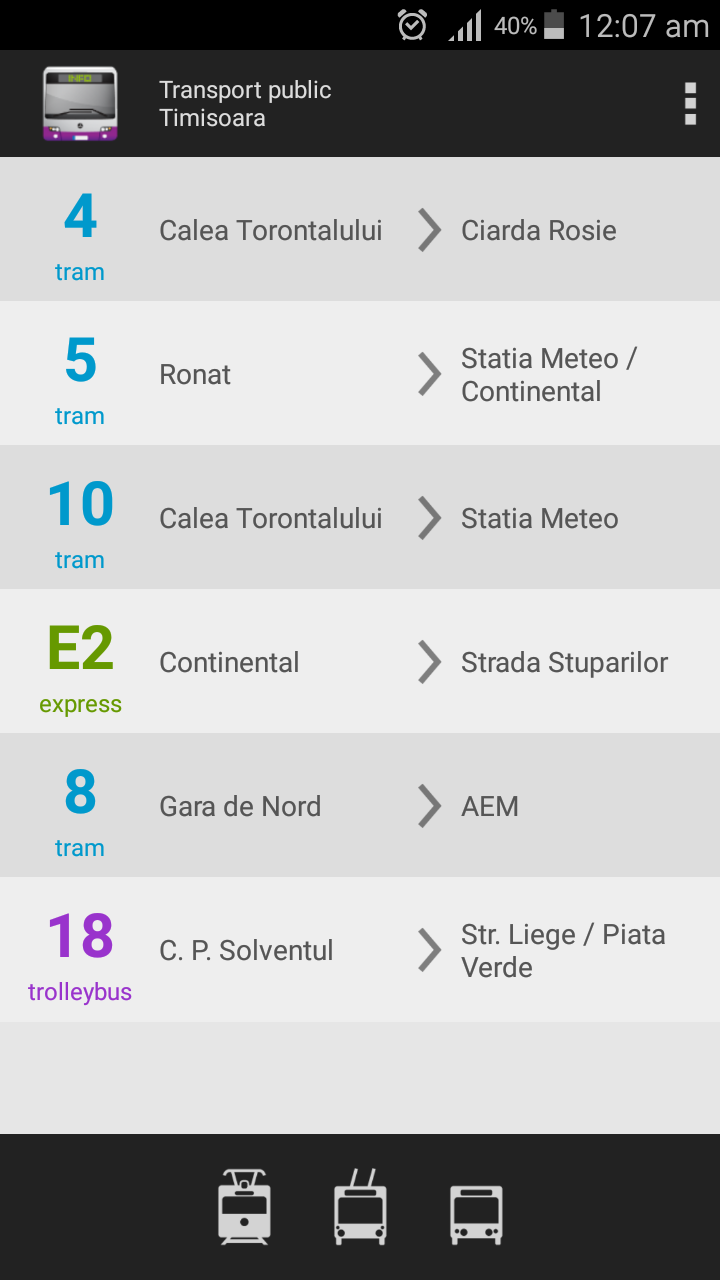
Pentru o bună gestionare a mijloacelor de transport, fiecare vehicul este echipat cu un dispozitiv de localizare GPS care este conectat la o rețea locala pentru centralizarea datelor. Odată ce informațiile privind localizarea mijloacelor de transport ajung la server, un algoritm de aproximarea timpilor va calcula pentru fiecare stație durata în care mijlocul de transport în comun va ajunge în respectiva stație. După terminarea calculelor, datele vor fi afișate sub forma de tabel în pagina web oficială a regiei de transport și pe panourile digitale plasate în stațiile din oraș.

O reprezentare grafică a sistemului de localizare se gasește în figura de mai jos.



2.2 Studiu asupra soluțiilor existente

La momentul actual există trei aplicații mobile pentru informarea cu privire la timpul de calatorie al mijloacelor de transport în comun, fiecare fiind dezvoltată pe cate una din principalele sisteme de operare mobile: iOS, Windows Phone și Android. În următoarele rânduri ne vom opri la aplicația folosită în Timișoara de cel mai mare numar de utilizatori, și anume Public Transport Timisoara pentru Android.



Aplicația dispune de o listă cu toate mijloacele de transport, iar la selectarea uneia dintre linii se va afișa o lista cu stațiile într-un anumit sens de deplasare cu timpii la care transportul ajunge în stație. Informațiile cu privire la timpul sosirii le extrage din calea:

<http://ratt.ro/txt/afis_msg.php?id_traseu=XXXX&id_statie=XXXX>

La momentul selectării unei anumite linii, aplicația va apela serverul RATT pentru fiecare stație cu parametrii trimiși: id-ul traseului selectat și id-ul stației. Stațiile și traseele disponibile sunt stocate în memoria telefonului printr-un fișier de tipul CSV creat manual cu ajutorul de mai multe persoane din mediul online, acesta putând fi actualizat doar prin actualizarea aplicației.

Prin folosirea îndelungată a acestei aplicații utilizatorii au observant că la orele de vârf, datorita cererilor la server pentru fiecare stație în parte și a mulțimii de utilizatori, serverul nu mai poate face față, iar informația devine inaccesibilă în momentele critice ale zilei.

**3.Bazele teoretice. Tehnologi folosite**

**3.1 Android OS**

Android este un sistem de operare, având la bază ca nucleu Linux și este dezvoltat în mare parte pentru telefoanele mobile și tablete. Începând din anul 2013 Android este sistemul de operare cel mai raspândit pe dispozitivele mobile din întreaga lume.

Android s-a format în anul 2005, moment în care telefoanele mobile inteligente erau într-un moment de început din punct de vedere tehnologic, din acest motiv dezvoltarea telefoanelor inteligente la acel moment era o mină de aur.

Android, Inc a început ca o companie de software de sine stătătoare care nu o vândut nici un produs timp de doi ani. Cu o echipa mica de ingineri software ei plănuiau să construiască noua generație de software pentru telefoanele mobile inteligente. Concentrându-se pe ideea de a conecta utilizatorul cât mai mult cu internetul și de a crea un mediu propice pentru orice dezvoltator software, compania o avut un plan bine pus la bază, încât în anul 2005 cand softwere-ul a fost terminat, o mulțime de investitori erau interesați de noua generație a experienței mobile. În acel moment Google căuta o companie de telefoane mobile pentru a concura împotrivă la Microsoft și Blackberry. Fondatorii Google dorind ca mai multe telefoane mobile să dețină în mod implicit ca motorul lor de căutare, platforma deschisă Android le-a oferit exact ceea ce aveau ei nevoie.

Primul telefon Android a fost T-Mobile G1, un telefon cu o tastatură QWERTY și un ecran tactil de dimensiuni mari pentru vremea respectivă. Acesta a fost unul dintre produsele create sub considerentele Google și produse de HTC.



În momentul lansării telefonului era o perioadă în care publicul nu era obisnuit cu o interfață doar tactilă a telefonului, ci erau oameni care încă simțeau nevoia unei tastaturi cu butoane fizice, acest telefon îmbinăndu-le pe ambele.

De atunci Android a evoluat tot mai mult pastrăndu-și principiile cu care a început: de a deveni un produs deschis pentru modificări și accesibil gratuit tuturor, revoluționând cu totul gradul libertate și de conectivitate ce îl conferă utilizatorului unui telefon mobil. Android și-a extins aria de dispozitive pe care poate rula de la telefoane, tablete pana la ceasuri și dispozitive din interiorul unei mașini.

**3.2. Android Studio**

Android Studio este platforma oficială de dezvoltare a aplicațiilor mobile Android, construită având la baza mediul de dezvoltare IntelliJ IDE creat de JetBrains. Este distribuită de către Google sub licența Apache Licence 2.0 ce oferă utilizatorilor dreptul gratuit de a folosi platforma gratuit atât în scopuri personale cât și în cele comerciale.

Fiecare proiect din Android Studio conține unul sau mai multe module ce conține urmatoarele foldere:

* *manifests*: conține fișierul AndroidManifest.xml ce prezintă informații esențiale despre aplicație, informații care trebuie să fie cunoscute de către sistem înainte de a rula codul. În conținutul fișierului este precizat numele pachetului Java al aplicației, aici se declara ce fel de permisiuni are nevoie aplicația pentru accesul parților protejate a unor diferite funcții API sau legături între alte aplicații, este descrisă lista de componente din aplicație: activități, servicii și în ce fel de condiții se rulează fiecare și lista de librarii folosite;
* *Java*: conține fișiere de cod sursă Java;
* *res*: conține fișiere non-cod cum ar fi scheme XML (layouts), anumite texte predefinite si imagini bitmap.

Mediul Android Studio folosește Gradle ca și fundație pentru sistemul de compilare, doar cu mai multe caracteristici configurate special pentru Android. Sistemul de compilare rulează ca o componentă integrată în meniul Android Studio și permite personalizarea, configurarea si extinderea procesului de compilare, crearea de multiple fișiere APK pentru aplicație, cu diferite caracteristici folosind acelasi proiect și aceleași module, de exemplu crearea de versiuni separate ale aplicației: o versiune gratuită si una platită. Fișierele de compilare Android Studio se numesc build.gradle și sunt fișiere text ce folosesc sintaxa Groovy pentru configurarea compilării. La importarea unu proiect existent, Android Studio genereaza automat fișierele necesare pentru compilare.

Android Studio asista la scrierea codului încât dispune de un dispozitiv de analiză de performanță a codului pe masură ce este scris. În modul de depanare ne ofera informații la fiecare linie cu privire la: valorile variabilelor, obiectele referențiate, valorile returnate de către metode. Tot în ajutorul dezvoltatorului, în vederea unei depanari și monitorizări cât mai exacte a aplicației, Android Studio dispune de informații cu privire la memorie si folosirea procesorului, gasirea obiectelor dezalocate, pierderilor de memorie, optimizarea performanței grafica si analizarea traficului internet.

Pentru a rula aplicația avem două opțiuni: folosind Android Emulator sau prin conectarea prin USB a unui dispozitiv cu un sistem de operare Android. Dupa selectarea uneia dintre opțiuni, se va instala aplicația pe emulator sau dispozitivul conectat, urmând ca la a doua rulare sa nu mai fie instalat din nou ci doar actualizat cu noile modificări efectuate, aceasta reducând semnificativ timpul de lansare pe dispozitiv sau emulator.

3.3. Google Maps Android API

Google Maps Android API permite dezvoltatorilor să înglobeze harți Google în orice aplicație. Interfața manipulează în mod automat accesul la serverele Google Maps prin descarcărcarea datelor necesare, afișarea hărții și raspunderea la schimbarea sau apropierea unei anumite locații pentru furnizarea unor detalii suplimentare. Deasemenea permite adaugarea de diferite componente optice pentru îmbunatățirea experienției utilizatorului cum ar fi: adaugarea anumitor puncte semnificative pe hartă (eng. marker), desenarea a diferitor linii sau poligoane pe harta și alte straturi grafice suprapuse peste harta de bază oferită de interfață.

Pentru a accesa interfața, aplicația are nevoie de o cheie API pentru a i se permite accesul la serverele Google Maps. Acest cod este generat gratuit și nu limitează numarul de utilizatori al aplicației.

În mod implicit, fișierul XML ce definește locul unde va fi poziționată harta în aplicație, res/layout/activity\_maps.xml, va conține următorul cod:

<fragment xmlns:android=<http://schemas.android.com/apk/res/android>   
    xmlns:tools=<http://schemas.android.com/tools>   
    android:layout\_width="match\_parent"   
    android:layout\_height="match\_parent"   
    android:id="@+id/map"   
    tools:context=".MapsActivity"   
    android:name="com.google.android.gms.maps.SupportMapFragment" />

Fișierul Java, MapsActivity.java, va implementa interfața OnMapReadyCallback care necesită suprascrierea a doua metode: onCreate si onMapReady. Prima metodă va fi apelată la pronirea aplicației și va porni procesul asincron de descărcare a hărților, iar a doua metodă se va executa atunci când hărțile s-au descărcat cu succes și cuprinde configurațiile de inceput ale harții cum ar fi: poziția de început a harții definită prin latitudine si longitudine, valoarea de apropiere față de coordonatele stabilite, metode pentru implementarea evenimentelor de selectare ale unor elemente grafice de pe hartă.

4. Soluția propusă. Implementare

4.1 Flow Software. Organigrama sistemului.

4.2 Implementare.

4.2.1

5. Utilizarea aplicației

6. Realizare. Punere in functie si Testarea Aplicatiei.

7. Concluzii, contribuții și direcții de continuare a dezvoltării

8. Bibliografie

Referințe:

https://developer.android.com/studio/intro/index.html

https://developers.google.com/maps/documentation/android-api/