**Waldo Low-level Design**

**1. Descriere sumară a funcţionalităţilor**

Aplicaţia detectează magazine de bicicletă din aproprierea utilizatorului. Pentru aceasta, trebuie să aibă datele pornite şi GPS-ul. Magazinele sunt afişate sub forma unei liste. Dacă se dă click pe un magazin, se deschide harta cu poziţia lui şi a utilizatorului. Dacă doreşte, poate primi comenzi vocale de navigaţie până la magazin prin intermediul Google Maps.

Din aplicaţie se pot şi adăuga magazine de bicicletă. Pe moment, datele introduse de utilizator vor fi trimise către un server, iar adăugarea efectivă se va face printr-un script care interacţionează cu interfaţa Google.

Cazurile de utilizare, precum şi scheme ale ecranelor se găsesc pe Dropbox.

**2. Clase şi metode**

**a) SplashScreen.java**

Primul ecran care se afişează utilizatorului este logoul Waldo. Clasa aferentă este SplashScreen.java, aflată în pachetul waldo.bike.waldo. Logica este următoarea: la deschiderea aplicaţiei se verifică să fie pornite datele şi activat GPS-ul. Verificarea efectivă se face prin apelul metodelor *checkGpsEnabled()*, respectiv *checkInternetConnection()* din Utilities.DeviceConnection.java. Acest lucru se întâmplă în metoda nativă sistemului onResume(), apelată la pornirea activităţii. Dacă oricare din cele două condiţii nu este îndeplinită, se afişează un mesaj de avertizare prin intermediul metodelor *showInternetDisabledAlertToUser()* şi *showGPSDisabledAlertToUser()*. Mesajele afişează şi un buton la a cărui apăsarea se deschide ecranul corespunzător pentru activare datelor sau GPS-ului.

Dacă ambele sunt activate, se intră direct în activitatea principală (MainActivity.java). Acest lucru se face prin apelul metodei startActivity(Context c) din onResume() sau onReceive() din NetworkChangeReceiver. Metoda onReceive() din această clasă are rolul de "a asculta" reţeaua - dacă se schimbă starea (se activează Internetul), mesajul de avertizare aferent este închis şi se porneşte ecranul principal, verificându-se în prealabil dacă şi GPS-ul este pornit (booleana isGPSEnabled). Această metodă este necesară, deoarece datele mobile nu se conectează instant. La întoarcerea în aplicaţie se apelează din nou onResume() care verifică starea datelor, dar ele sunt în curs de conectare. În momentul în care se conectează, metoda onReceive() din NetworkChangeReceiver este apelată.

Pentru aceasta, am definit în AndroidManifest.xml un receptor:

<receiver android:name="waldo.bike.waldo.SplashScreen$NetworkChangeReceiver">

<intent-filter>

<action android:name="android.net.conn.CONNECTIVITY\_CHANGE"/>

</intent-filter>

</receiver>

**Atenţie!** Dacă va fi vreodată nevoie să se schimbe activitatea în care se verifică starea datelor mobile, trebuie schimbat şi numele receptorului pentru a indica acea clasă în care "se ascultă" (android:name="waldo.bike.waldo.SplashScreen$NetworkChangeReceiver").

2. Utilities.DeviceConnection

Conţine metode ce verifică accesul la reţea ori locaţie. Numele metodelor sunt intuitive şi oferă o idee legat de ce fac.

3. Utilities.Constants

Toate constantele folosite în aplicaţie. Nu include mesajele de eroare/informare, care vor fi/sunt trecute în values/strings.xml