

**Univerzita Hradec Králové**  
**Fakulta informatiky a managementu**  
**Katedra informatiky a kvantitativních metod**

**Měření kvality mobilního připojení k internetu**  
Diplomová práce

Autor: Bc. Jan Čučiak  
Studijní obor: Aplikovaná informatika

Vedoucí práce: Ing. Pavel Kříž, Ph.D.

Hradec Králové

Duben 2020

Prohlášení:

Prohlašuji, že jsem diplomovou práci zpracoval samostatně a s použitím uvedené literatury.

V Hradci Králové dne 30.4.2020

Bc. Jan Čučiak

Poděkování:

Děkuji vedoucímu diplomové práce Ing. Pavlu Křížovi, Ph.D. za metodické vedení práce, cenné rady a možnost častých konzultací. Dále děkuji Ing. Janu Budinovi za technickou výpomoc a cenné rady.



## **Anotace**

**Klíčová slova** – mobilní internetové připojení, rychlost stahování, rychlost nahrávání, download, upload, 2G, 3G, 4G, GPRS, EDGE, UMTS, HSPA, HSPA+, LTE, LTE Advanced, měření internetu, rychlost internetu

V teoretické části diplomové práce jsou popsány aktuálně využívané mobilní datové technologie a parametry mobilního internetového připojení, které určují jeho kvalitu. V teoretické části byla dále provedena rešerše již existujících mobilních aplikací pro měření parametrů (mobilního) internetového připojení. Praktická část diplomové práce se zabývá návrhem a vývojem mobilní aplikace pro mobilní operační systém Android, která umožňuje měřit parametry (mobilního) internetového připojení – rychlost stahování, rychlost nahrávání a odezva. V závěru diplomové práce byla vytvořená mobilní aplikace využita pro měření parametrů mobilního internetového připojení v několika vybraných lokalitách. Byla provedena analýza naměřených hodnot a zhodnocení kvality mobilního internetového připojení ve vybraných lokalitách.

## **Annotation**

**Keywords** – mobile internet connection, upload, download, 2G, 3G, 4G, GPRS, EDGE, UMTS, HSPA, HSPA+, LTE, LTE Advanced, internet measurement, internet speed

**Title:** Measuring the quality of mobile internet connection

The theoretical part of the diploma thesis describes the currently used mobile data technologies and parameters of mobile internet connection, which determine its quality. In the theoretical part, a search of existing mobile applications for measuring the parameters of the (mobile) Internet connection was conducted. The practical part of the thesis deals with the analyze and development of a mobile application for the mobile operating system Android, which allows to measure the parameters of the (mobile) Internet connection - download speed, upload speed and ping. At the end of the diploma thesis, the created mobile application was used to measure the parameters of the mobile Internet connection in several selected locations. An analysis of measured values and evaluation of the quality of mobile internet connection in selected localities was performed.

# Obsah

1	Úvod.....	1
2	Cíl práce.....	3
3	Přehled mobilních technologií .....	5
3.1	Úvod do problematiky.....	5
3.2	GPRS .....	6
3.3	EDGE .....	7
3.4	UMTS, HSDPA, HSUPA.....	7
3.5	LTE.....	8
3.6	LTE Advanced.....	9
3.7	Sítě páté generace .....	10
3.8	Obecné využití mobilního internetu.....	11
3.8.1	FUP – Fair User Policy.....	11
3.9	Parametry internetového připojení.....	12
3.9.1	Rychlost stahování .....	12
3.9.2	Rychlost nahrávání .....	12
3.9.3	Odezva.....	13
3.9.4	Jitter .....	13
4	Rešerše existujících řešení.....	14
4.1	Speedtest.....	14
4.2	SPEEDCHECK Internet Speed test .....	16
4.3	SpeedSmart .....	17
4.4	RYCHLOST.CZ .....	18
4.5	DSL.CZ.....	21
4.6	FAST Speed Test.....	22
4.7	Shrnutí.....	24

5	Analýza vlastního řešení.....	26
5.1	Android .....	26
5.1.1	Jazykové mutace .....	27
5.2	Obecný návrh mobilní aplikace .....	28
5.3	Měření parametrů internetového připojení .....	29
5.3.1	Rychlost nahrávání a rychlost stahování .....	30
5.3.2	Měření odezvy.....	30
6	Návrh a implementace .....	32
6.1	Použité technologie.....	32
6.1.1	Programovací jazyk Java.....	32
6.1.2	Vývojové prostředí Android Studio.....	32
6.1.3	Knihovna JSch .....	33
6.1.4	Oracle DB .....	33
6.1.5	Oracle SQL Developer .....	33
6.1.6	JDBC .....	33
6.1.7	Maps SDK for Android .....	34
6.2	Oracle DB.....	34
6.2.1	Struktura .....	34
6.2.2	Práce s databází – třída SQLHelper .....	35
6.3	Využití SFTP serveru .....	38
6.4	Využití GPS.....	39
6.5	Použití Google Maps API .....	40
6.6	Algoritmy pro měření rychlosti nahrávání a rychlosti stahování.....	42
6.7	Algoritmus na měření odezvy internetového připojení .....	47
7	Měření a výsledky .....	50
7.1	Vybrané lokality pro měření.....	50

7.2	Metodika měření .....	51
7.2.1	Mobilní telefon Xiaomi Redmi 4A LTE .....	51
7.2.2	Počet provedených měření.....	51
7.2.3	Metodika měření.....	51
7.3	Analýza provedených měření.....	52
7.3.1	Souhrnný pohled na provedená měření .....	53
7.3.2	Zhodnocení mobilního internetu dle lokality – Holice.....	54
7.3.3	Zhodnocení mobilního internetu dle lokality – Pardubice.....	56
7.3.4	Zhodnocení mobilního internetu dle lokality – Hradec Králové.....	58
7.3.5	Zhodnocení mobilního internetu dle lokality – Říčky v Orlických horách 60	
7.4	Rychlost mobilního internetu v době epidemie a před.....	61
8	Závěry a doporučení .....	64
9	Seznam použitých zdrojů .....	65
10	Přílohy.....	68
10.1	Obsah přiloženého CD.....	68
10.2	Oskanované zadání diplomové práce .....	69
10.3	Ukázka mobilní aplikace .....	70

## Seznam obrázků

Obrázek 1 - Vývoj technologií v průběhu let [1] .....	5
Obrázek 2 - Technologie spadající do třetí generace [1] .....	8
Obrázek 3 – Pokrytí LTE Advanced, Vodafone 2019 [8] .....	10
Obrázek 4 – Ukázka příkazu ping, příkazový řádek, OS Windows. ....	13
Obrázek 5 - Ukázka aplikace Speedtest v prostředí iOS (Výsledek testu) .....	15
Obrázek 6 – Ukázka aplikace SPEEDCHECK v prostředí iOS. ....	17



Obrázek 7 – Ukázka zobrazení výsledku měření, aplikace pro iOS.....	18
Obrázek 8 – Ukázka zobrazení výsledků v prostředí iOS a webového prohlížeče Safari.....	20
Obrázek 9 - Ukázka zobrazení výsledků v prostředí iOS a webového prohlížeče Safari.....	22
Obrázek 10 – Ukázka aplikace FAST Speed Test v prostředí webového prohlížeče Firefox (verze 71).....	23
Obrázek 11 – Ukázka nastavení.....	23
Obrázek 12 – Procentuální podíl jednotlivých verzí.....	26
Obrázek 13 – Ukázka editoru překladů v prostředí Android Studio.....	27
Obrázek 14 – Základní schéma mobilní aplikace.....	28
Obrázek 15 – Zjednodušený návrh mobilní aplikace.....	29
Obrázek 16 – Struktura tabulky MEASUREMENTS.....	35
Obrázek 17 – Diagram průběhu testu.....	43
Obrázek 18 – Graf zobrazující naměřené hodnoty v lokalitě č. 1, Holice, Náměstí TGM.....	54
Obrázek 19 – Graf zobrazující naměřené hodnoty v lokalitě č. 2, Holice, Tyršova.....	55
Obrázek 20 - Graf zobrazující naměřené hodnoty v lokalitě č. 3, Pardubice, letiště terminál.....	56
Obrázek 21 - Graf zobrazující naměřené hodnoty v lokalitě č. 4, Pardubice, hlavní nádraží.....	57
Obrázek 22 - Graf zobrazující naměřené hodnoty v lokalitě č. 5, Pardubice, Pernštýnské náměstí.....	57
Obrázek 23 - Graf zobrazující naměřené hodnoty v lokalitě č. 6, Hradec Králové, Velké náměstí.....	58
Obrázek 24 - Graf zobrazující naměřené hodnoty v lokalitě č. 7, Říčky v Orlických horách.....	60

Obrázek 25 – Graf naměřených hodnot pro lokalitu č. 1.....	62
Obrázek 26 - Graf naměřených hodnot pro lokalitu č. 2.....	62
Obrázek 27 - Graf naměřených hodnot pro lokalitu č. 3.....	63

## Seznam tabulek

Tabulka 1 – Generace mobilních datových technologií .....	6
Tabulka 2 - Maximální teoretické rychlosti sítí třetí generace .....	8
Tabulka 3 - Pásma používána pro LTE v České republice.....	9
Tabulka 4 – Shrnutí možnosti testování parametrů. ....	24
Tabulka 5 – Shrnutí dodatečných možností aplikací. ....	25
Tabulka 6 – Vybrané lokality pro měření parametrů mobilního internetového připojení.....	50
Tabulka 7 – Přehled testovaných technologií a velikostí použitých testovacích souborů.....	53
Tabulka 8 – Průměrné naměřené hodnoty ze všech testů v rámci jednotlivých mobilních datových technologií.....	53

## Ukázky kódu

Ukázka kódu 1 – Získání všech výsledků.....	36
Ukázka kódu 2 - Zápis objektu Result do databáze. ....	37
Ukázka kódu 3 - Metoda writeData třídy SQLHelper.....	37
Ukázka kódu 4 - Metoda download.....	38
Ukázka kódu 5 - Použití třídy GPSHelper .....	39
Ukázka kódu 6 - Získání GPS souřadnic ze třídy GPSHelper.....	39
Ukázka kódu 7 - Metoda onMapReady třídy MapsActivity .....	40
Ukázka kódu 8 - Dotaz na získání měření z databáze.....	41

Ukázka kódu 9 - Třída DownloadFileTask.....	45
Ukázka kódu 10 - Využití třídy DownloadFileTask.....	46
Ukázka kódu 11 - Třída PingTask.....	47
Ukázka kódu 12 - Použití třídy PingTask .....	48

# 1 Úvod

Chytré mobilní telefony v současné době využívá každodenně stále větší množství lidí po celém světě. Mobilní telefony využívající moderní operační systémy (nejčastěji pak Android a iOS) nabízí široké spektrum využití, zejména pak skrze mobilní aplikace. Právě mobilní aplikace se staly trendem současné doby. K dispozici je stále větší a větší množství mobilních aplikací. Velké množství mobilních aplikací, které každodenně používáme využívá v průběhu svého běhu internetové připojení.

S rostoucím využitím chytrých mobilních telefonů roste touha po tzv. mobilitě. Mobilita znamená, že internetové připojení by mělo být dostupné a stabilní nezávisle na geografické lokalitě, kde se mobilní telefon nachází. Být mobilní a zároveň být neustále připojen k internetu nám umožňují mobilní datové sítě (technologie). Právě využití mobilních datových sítí a množství dat přenesených skrze tyto sítě je každoročně na vzestupu.

Dalo by se říci, že proti mobilitě se postavili čeští mobilní operátoři (coby poskytovatelé mobilního internetového připojení) zavedením tzv. FUP (angl. Fair User Policy). Ve stručnosti se jedná o skupinu zavedených pravidel, pomocí které se mobilní operátor snaží omezit množství dat, které může uživatel (zařízení) přenést za danou časovou jednotku. Příkladem může být, že mobilní operátor nabídne 4 GB dat na 30 dní za cenu 399,-, což je po přepočtení na dny pouhých 133 MB na 1 den. Po vyčerpání dat je připojení zcela zastaveno a pro jeho obnovení je nutné dokoupit si další balíček. FUP je mobilními operátory používáno i přesto, že v současné době se stále více diskutují a propagují sítě čtvrté generace. Do popředí se již také na úrovni mobilních operátorů začínají dostávat informace o sítích páté generace.

Aktuálně se v praxi setkáváme se sítěmi druhé generace (GPRS, EDGE), sítěmi třetí generace (UMTS, HSPA, HSPA+, LTE) a sítěmi čtvrté generace (LTE Advanced). Mezi těmito jednotlivými mobilními datovými technologiemi jsou relativně velké rychlostní rozdíly.

Pokud tedy chceme využívat mobilní internetové připojení, tak vždy záleží na dané mobilní datové technologii, která je dostupná v lokalitě, kde se uživatel (mobilní zařízení) nachází. Hlavním problémem je, že pokrytí jednotlivými mobilními datovými technologiemi se liší v závislosti na konkrétním mobilním operátorovi. Pro ilustraci problému uvažujme jednu konkrétní lokalitu (např. ulici ve městě). První operátor tuto lokalitu může pokrývat např. LTE, druhý pak HSPA+ a třetí technologií EDGE.

## 2 Cíl práce

V teoretické části diplomové práce je cílem teoretický popis a analýza všech současně využívaných mobilních datových technologií. Každá technologie bude stručně popsána a bude vypracováno souhrnné zhodnocení všech vybraných mobilních datových technologií.

Popsány budou rovněž jednotlivé parametry mobilního internetového připojení, kterými se práce dále zabývá – rychlost nahrávání, rychlost stahování a odezva.

Dále bude vypracována rešerše již existujících řešení – mobilních aplikací pro operační systém Android, které umožňují měření parametrů internetového připojení. Budou popsány možnosti vybraných mobilních aplikací a v závěru bude vypracováno souhrnné srovnání těchto mobilních aplikací.

V praktické části diplomové práce je cílem navrhnout mobilní aplikaci, která bude měřit tři základní parametry mobilního internetového – rychlost stahování, rychlost nahrávání a odezvu, přičemž algoritmy budou optimalizovány pro mobilní datové technologie. Algoritmus rovněž detekuje aktuálně využívanou mobilní datovou technologii a detekuje polohu zařízení. Tyto hodnoty budou uloženy do databáze pro pozdější zobrazení.

Cíle mobilní aplikace jsou následující.

- Navržení algoritmu pro měření parametrů mobilního internetového připojení – rychlost nahrávání, rychlosti stahování a odezvy.
- Možnost zvolení testování pouze jednoho parametru.
- Možnost zvolení konkrétní velikosti testovacího souboru.
- Detekce a optimalizace pro mobilní internetové připojení a jednotlivé mobilní datové technologie.
- Detekce možného zpomalení v rámci FUP.
- Uložení jednotlivých měření do databáze k pozdějšímu zpracování.
- Spolu s výsledky bude uložena také pozice, kde bylo měření provedeno (bude-li uživatelem povoleno).

Na závěr bude použita výsledná mobilní aplikace k vyhodnocení kvality mobilního internetového připojení. Ve vybraných lokalitách na území Královehradeckého a Pardubického kraje bude provedeno měření mobilního internetového připojení. Pro měření budou použity sítě druhé, třetí a čtvrté generace (budou-li v lokalitě dostupné). V každé vybrané lokalitě pak budou testováni dva vybraní mobilní operátoři (O2 a T-Mobile).

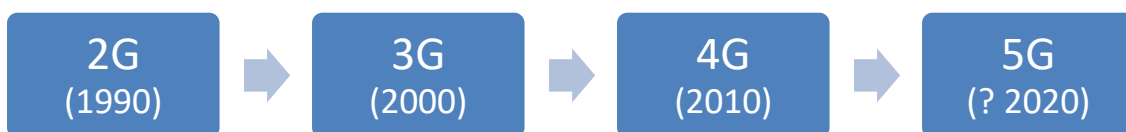
## 3 Přehled mobilních technologií

### 3.1 Úvod do problematiky

V současné době existuje několik typů mobilních datových technologií, které se využívají k připojení mobilních zařízení k internetu. Historie se začala psát okolo roku 1990, kdy byla uvedena technologie GPRS, označována také jako síť 2G. Aktuálně se setkáváme s dělením mobilních datových technologií do tzv. generací (2G, 3G, 4G, 5G). Toto dělení však není zcela jednoznačné, jelikož v rámci jedné generace existuje více mobilních datových technologií, mezi kterými jsou znatelné rozdíly. [\[1\]](#) [\[3\]](#)

Nejnovější technologií, který se aktuálně využívá je LTE Advanced. Tato technologie však ještě na území České republiky není příliš rozšířená.

Pomalu, ale jistě již také začínají do popředí široké veřejnosti vstupovat páté generace. První zemí, která spustila síť páté generace se stala Jižní Korea [\[17\]](#).



**Obrázek 1 - Vývoj technologií v průběhu let [\[1\]](#)**

Hlavním posláním mobilních technologií je zaručení „mobility“ [\[16\]](#). To znamená, že mobilní datové připojení by mělo být k dispozici všude tam, kde je potřeba (tj. být nezávislé na lokalitě, kde se mobilní internetové připojení využívá). Dále by mělo být mobilní internetové připojení dostatečně rychlé a stabilní.

Následující tabulka nabízí shrnující pohled na jednotlivé mobilní datové technologie. U každé mobilní datové technologie je uveden rok, kdy byla technologie uvedena do provozu, generace, do které lze mobilní datovou technologii zařadit a teoretickou rychlost, které technologie dosahovala.



**Tabulka 1 – Generace mobilních datových technologií**

<b>Technologie</b>	<b>Generace</b>	<b>Rok zavedení</b>	<b>Rychlost stahování <sup>1</sup></b>	<b>Rychlost nahrávání <sup>2</sup></b>
GPRS	2G	1997	80 kbps	40 kbps
EDGE	„2,5G“	2004	218 kbps	134 kbps
UMTS	3G	2001	2 Mbps	2 Mbps
HSDPA, HSUPA	„3,5G“	2006	14,4 Mbps	5,76 Mbps
HSDPA+, HSUPA+	„3,75G“	2008	42 Mbps	7,2 Mbps
LTE	„3,9G“	2009	172,8 Mbps	57,6 Mbps
LTE-Advanced	4G	2010	1 Gbps	100 Mbps

Zdroj: [\[15\]](#)

### **3.2 GPRS**

Mobilní datová technologie GPRS (General Packet Radio Service) byla uvedena v devadesátých letech minulého století. Jedná se o nejstarší mobilní datovou technologii, která umožňovala přenos dat (připojení k internetu). Běžná rychlost přenosu dat se pohybuje pouze okolo 80 kbps pro download, respektive 40 kbps pro upload, což v dnešní době není dostatečné pro normální využití internetu. Tato technologie spadá do tzv. generace 2,5G. Mobilní zařízení podporující technologii GPRS se dělí do 3 tříd: A, B, C. [\[5\]](#)

- Třída A umožňuje využít hlasové a datové služby současně.
- Třída B umožňuje v jeden okamžik využít buď hlasové nebo datové služby.
- Třída C umožňuje použití pouze datového připojení. Využití hlasových služeb není podporováno.

Po technické stránce představuje technologie GPRS nadstavbu nad technologií GSM. Technologie GSM je již využívána pouze pro hlasové služby. Pokrytí technologií GPRS je na všech místech, kde je k dispozici technologie GSM. [\[1\]](#) [\[4\]](#)

---

<sup>1</sup> Maximální teoretická rychlost stahování. V běžné praxi nedosažitelná.

<sup>2</sup> Maximální teoretická rychlost nahrávání. V běžné praxi nedosažitelná.

### **3.3 EDGE**

Mobilní datová technologie EDGE (Enhanced Data rates for GSM Evolution) byla uvedena v roce 2004. Po technické stránce se jedná o nadstavbu technologie GPRS. Mobilní datová technologie EDGE je často označována za tzv. 2,75. generaci. [1]

Uživatelsky běžná rychlost přenosu dat dosahuje 160 kbps pro download, respektive 80 kbps pro upload, což je v porovnání s technologií GPRS přibližně dvojnásobná rychlost přenosu dat. Tato rychlost dostačuje pouze na základní prohlížení internetu (např. prohlížení jízdních řádů), popřípadě na zasílání textových zpráv skrze mobilní aplikace. Avšak pokrytí touto technologií dosahuje 99% území České republiky. [2]

### **3.4 UMTS, HSDPA, HSUPA**

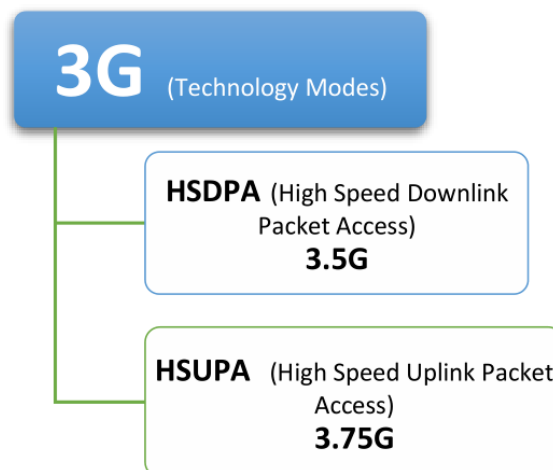
UMTS (Universal Mobile Telecommunication System) je mobilní technologie, která byla navržena pro využití hlasových a datových služeb současně. Mobilní datová technologie UMTS je označována za první technologii spadající do tzv. třetí generace. [1]

Třetí generace mobilních datových technologií je založena na širokopásmové bezdrátové síti a zahrnuje mezinárodní specifikace „Mobile Telecommunications 2000“ (IMT-2000). Podle mezinárodní specifikace IMT-2000 musí síť třetí generace dosahovat datové rychlosti nejméně 200 kbps, přičemž běžný datová rychlost se pohybuje okolo 2 Mbps. Tato rychlost je již dostatečná na běžné využívání připojení k internetu. [4][5]

Technologie UMTS využívá pro přenos dat jednu z následujících technologií.

- **Technologie FDD** (Frequency Division Duplex) umožňuje hlasových a datových služeb současně.
- **Technologie TDD** (Time Division Duplex) nabízí podporu pouze pro datové služby, hlasové služby nejsou podporovány. Oproti technologii FDD nabízí vyšší rychlost přenosu dat.

Do sítě třetí generace spadají ještě další 2 mobilní datové technologie, které nabízí v porovnání s technologií UMTS několikanásobně rychlejší přenos dat.



**Obrázek 2 - Technologie spadající do třetí generace [1]**

Mobilní datové technologie HSDPA (a vylepšená varianta HSDPA+) a HSUPA (a vylepšená varianta HSUPA+) jsou vylepšením technologie UMTS. Technologie HSDPA (HSDPA+) je určena pro stahování (download) dat a technologie HSUPA (HSUPA+) je určena pro nahrávání (upload) dat. Hlavním cílem těchto technologií bylo zvýšit rychlost přenosu dat tak, aby byla dostatečná pro sledování, popřípadě sdílení multimediálních souborů. Sítě třetí generace nabízí oproti sítím druhé generace několikanásobně lepší odezvu (ping) připojení. [3]

Maximální teoretické rychlosti sítí třetí generace uvádí následující tabulka.

**Tabulka 2 - Maximální teoretické rychlosti sítí třetí generace**

	Download	Upload
UMTS	2 Mbps	2 Mbps
HSDPA	14,4 Mbps	-
HSDPA+	42 Mbps	-
HSUPA	-	5,76 Mbps
HSUPA+	-	7,2 Mbps

Zdroj: [15]

### 3.5 LTE

Mobilní datová technologie LTE (Long Term Evolution) byla uvedena do provozu v roce 2010. Tato technologie ještě spadá do sítí tzv. třetí generace, byť čeští operátoři ji mnohdy mylně označují za síť čtvrté generace.

U technologie LTE existuje pouze technologie pro datové služby, speciální technologie pro hlasové služby není k dispozici. Hlasové služby se proto využívají skrze datové spojení. O toto se stará VoLTE (Voice over LTE). Zařízení, které nepodporuje technologii VoLTE musí pro hlasové služby využít síť nižší generace (2G nebo 3G). [\[2\]](#) [\[5\]](#)

Maximální teoretická rychlost přenosu dat je 172,8 Mbps pro download, respektive 57,6 Mbps pro upload. O tuto přenosovou rychlost se dělí všechna zařízení, které jsou v danou chvíli současně připojena ke zdroji.

Mobilní datová technologie LTE je v České republice a na Slovensku šířena v různých frekvenčních pásmech následovně: [\[6\]](#)

- Česká republika: Pásma 1, 3, 7, 8 a 20
- Slovensko: Pásma 3 a 20

**Tabulka 3 - Pásma používaná pro LTE v České republice**

Pásmo	Frekvence	Komentář
1	2100 MHz	
3	1800 MHz	Pásmo určené primárně pro města s vysokými datovými nároky (Praha, Brno, Ostrava apod.)
7	2600 MHz	Toto pásmo se obvykle nepoužívá samostatně, ale doplňuje pásmo 1, 3 a 20 pro LTE Advanced.
8	900 MHz	Pásmo je primárně využíváno pro 2G. Mobilní operátor toto pásmo využívá pro pokrytí LTE mimo velká města.
20	800 MHz	Pásmo, které pokrývá většinu území České republiky. Bylo přiděleno pro LTE při uvolnění frekvencí spojeném s přechodem na digitální televizní vysílání.

**Zdroj:** [\[6\]](#) [\[7\]](#)

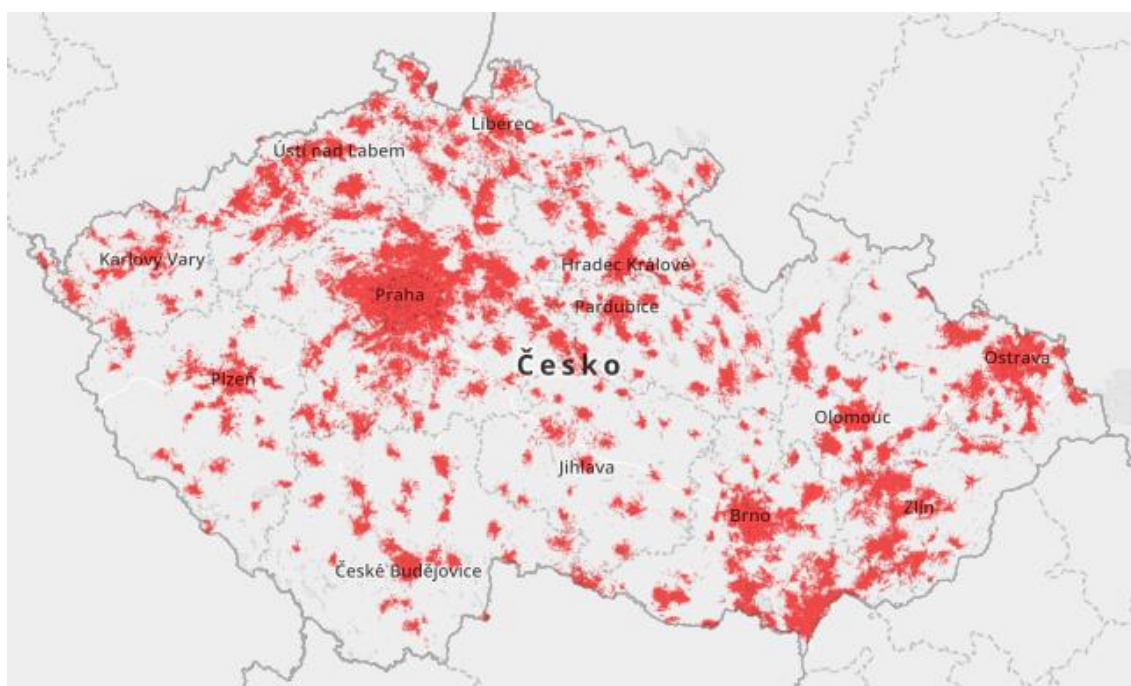
### **3.6 LTE Advanced**

Mobilní datová technologie LTE Advanced je technologií, která se již řadí do čtvrté generace. Podle Mezinárodní telekomunikační unie (ITU) lze za síť čtvrté generace považovat takovou síť, která umožní download 100 Mbps za pohybu a 1000 Mbps

v klidu. V současné době se jedná o nejnovější a nejrychlejší mobilní datovou technologii, která je v české republice běžně nabízena. [1]

Oproti mobilní technologii LTE nabízí větší přenosové rychlosti. Maximální přenosová rychlost LTE Advanced by se dle Českých operátorů měla pohybovat okolo 400 Mbps pro download, respektive 100 Mbps pro upload. [5]

Přestože technologie LTE Advanced není žádnou čerstvou novinkou, tak mapu pokrytí pouze pro technologie LTE Advanced nabízí pouze jeden z Českých operátorů – Vodafone.



Obrázek 3 – Pokrytí LTE Advanced, Vodafone 2019 [8]

### 3.7 Síť páté generace

Sítě páté generace nabídnou rychlejší a stabilnější mobilní internetové připojení, které by mělo fungovat společně se sítěmi třetí a čtvrté generace. Hlavním cílem sítě páté generace má být přibližně desetinásobné zvýšení rychlosti přenosu dat a snížení odezvy na jednotky milisekund v porovnání se sítěmi čtvrté generace. [1] [2]

Maximální teoretická rychlost sítě páté generace by měla dosahovat hodnoty 10 gigabitů za sekundu, což je v porovnání se sítěmi čtvrté generace 100x vyšší hodnota. Síť páté generace rovněž slibují zlepšení odezvy. Maximální odezva by se měla pohybovat někde kolem 4 milisekund. [18]

Dalším důvodem zavádění sítí páté generace je tzv. Internet věcí (IoT). V průběhu následujících desetiletí by mělo být k internetu připojené obrovské množství věcí, které mezi sebou budou vzájemně komunikovat. „Společnost Ericsson očekává, že do roku 2023 se celosvětově dosáhne na 3,5 miliard připojených zařízení k internetu věcí“. [\[4\]](#) [\[9\]](#)

Masivnější nasazení sítí páté generace se pak očekává v průběhu roku 2021–2022.

### **3.8 Obecné využití mobilního internetu**

Mobilní internet můžeme v dnešní době využívat na většinu běžných věcí, podobně, jako využíváme klasická připojení k internetu. Výjimku tvoří pouze sítě druhé generace (GPRS, EDGE), které svou rychlostí a odezvou používání internetu značně omezují. Obecně lze říci, že sítě druhé generace jsou v současné době vhodné pouze na „nouzové“ prohlížení internetu, psaní textových zpráv skrze mobilní aplikace, vyhledávání spojení v jízdnicích řádech, popřípadě také k M2M (machine to machine) komunikaci mezi dvěma zařízeními. Oproti tomu sítě třetí, a zejména pak čtvrté generace nabízí rychlost dostatečnou pro sledování videí, zasílání emailů s přílohami apod.

#### **3.8.1 FUP – Fair User Policy**

Hlavní brzdou, se kterou se můžeme setkat velice často u Českých operátorů je tzv. FUP – Fair User Policy. Jedná se o skupinu pravidel, pomocí kterých se snaží mobilní operátoři (poskytovatelé mobilního internetového připojení) omezit množství dat, které může uživatel přenést za jednotku času. Po vyčerpání specifického množství dat se připojení značně zpomalí, nebo dokonce zastaví.

V minulosti jsme se s FUP mohli setkat také u běžného připojení k internetu v domácnostech (například prostřednictvím ADSL, VDSL apod.). V současné době se s FUP již pouze setkáme zejména u mobilního internetového připojení.

Příkladem může být, že mobilní operátor nabídne 4 GB dat na 30 dní za cenu 399,- (za 1 měsíc)<sup>3</sup>, což je po přepočtení na dny pouhých 133 MB na 1 den. Po vyčerpání dat je připojení zcela zastaveno a pro jeho obnovení je nutné dokoupit si další balíček.

### **3.9 Parametry internetového připojení**

U internetového připojení můžeme pozorovat a měřit několik parametrů – rychlost stahování rychlost nahrávání, odezvu a jitter. Tyto jednotlivé parametry nám mohou udávat kvalitu internetového připojení a jeho možné teoretické využití.

#### **3.9.1 Rychlost stahování**

Rychlost stahování – anglicky download udává, za jak velký časový úsek stáhneme soubor o určité velikosti ze vzdáleného umístění (nejčastěji (S)FTP server, webový server apod.). Nejčastěji se udává v megabitech za jednotku sekundy (zkráceně Mbps).

Například chceme stáhnout z FTP serveru soubor o velikosti 30 MB. Takovýto soubor budeme stahovat po dobu 12 sekund. Výsledná průměrná rychlost stahování bude 2,5 MB (megabajtů) za sekundu neboli 20 Mbps.

#### **3.9.2 Rychlost nahrávání**

Rychlost nahrávání – anglicky upload udává, za jak velký časový úsek odešleme (nahrajeme) soubor o určité velikosti na vzdálené umístění. Nejčastěji se udává v megabitech za jednotku sekundy, podobně jako v případě rychlosti stahování.

Například chceme odeslat soubor na webový server o velikosti 10 MB. Odeslání souboru trvá 15 sekund. Výsledná průměrná rychlost nahrávání bude 0,67 MB za sekundu neboli 5,3 Mbps.

„Většina připojení je koncipována tak, aby dosahovala větší rychlosti stahování v porovnání s rychlostí nahrávání.“ [\[14\]](#)

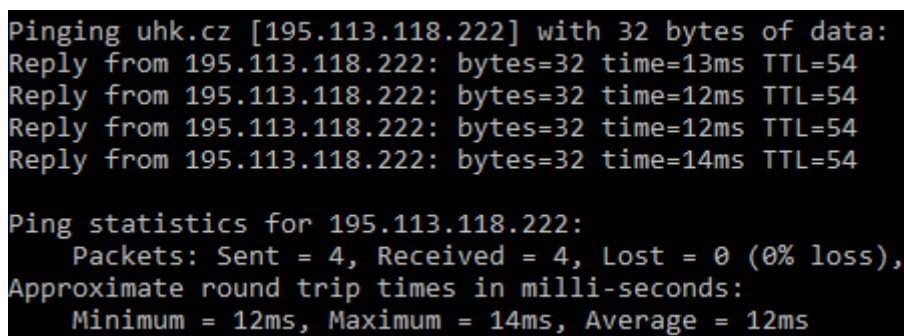
---

<sup>3</sup> Datový balíček nabízený v prosinci 2019 na stránkách společnosti T-Mobile (<https://www.t-mobile.cz/internet-do-mobilu>)

### 3.9.3 Odezva

Odezva (někdy označována také jako tzv. latence) – anglicky ping je parametr, který udává, za jaký časový úsek obdržíme odpověď od vzdáleného umístění (např. server) od doby odeslání požadavku. Zjednodušeně – odezva udává čas od odeslání požadavku do přijetí odpovědi na odeslaný požadavek. Odezva se nejčastěji udává v milisekundách (ms).

Například, chceme zjistit odezvu počítače vůči webovému serveru uhk.cz. Použijeme systémový příkaz „ping“ v prostředí příkazového řádku operačního systému Windows – „ping uhk.cz“.



```
Pinging uhk.cz [195.113.118.222] with 32 bytes of data:
Reply from 195.113.118.222: bytes=32 time=13ms TTL=54
Reply from 195.113.118.222: bytes=32 time=12ms TTL=54
Reply from 195.113.118.222: bytes=32 time=12ms TTL=54
Reply from 195.113.118.222: bytes=32 time=14ms TTL=54

Ping statistics for 195.113.118.222:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 12ms, Maximum = 14ms, Average = 12ms
```

Obrázek 4 – Ukázka příkazu ping, příkazový řádek, OS Windows.

Z obrázku výše vyplývá, že na webový server uhk.cz byl čtyřikrát za sebou odeslán požadavek na odpověď. Nejkratší doba mezi odesláním požadavku a obdržetím odpovědi byla 12 ms, nejdelší pak 14 ms. V průměru je tedy doba odezvy 12,75 ms.

### 3.9.4 Jitter

Jitter je parametr, který udává kolísání odezvy připojení. Jinak řečeno jedná se o změnu odezvy internetového připojení od průměrné odezvy v průběhu času.

Z obrázku č. 4, který je uveden výše vyplývá, že průměrná doba odezvy je 12,75 ms. Nejlepší odezva, která byla naměřena je 12 ms, nejhorší pak 14 ms. Nejhorší odchylka od průměrné doby odezvy je tedy 1,25 ms (14 ms – 12,75 ms) = jitter. V případě, že je jitter internetového připojení v rámci jednotek milisekund (do 15 ms) můžeme říci, že takové připojení je stabilní.



## 4 Rešerše existujících řešení

V současné době existují spousty aplikací na měření parametrů (mobilního) internetové připojení. Některé aplikace jsou pouze webové, jiné aplikace jsou k dispozici také pro různé platformy, zejména pak mobilní platformy.

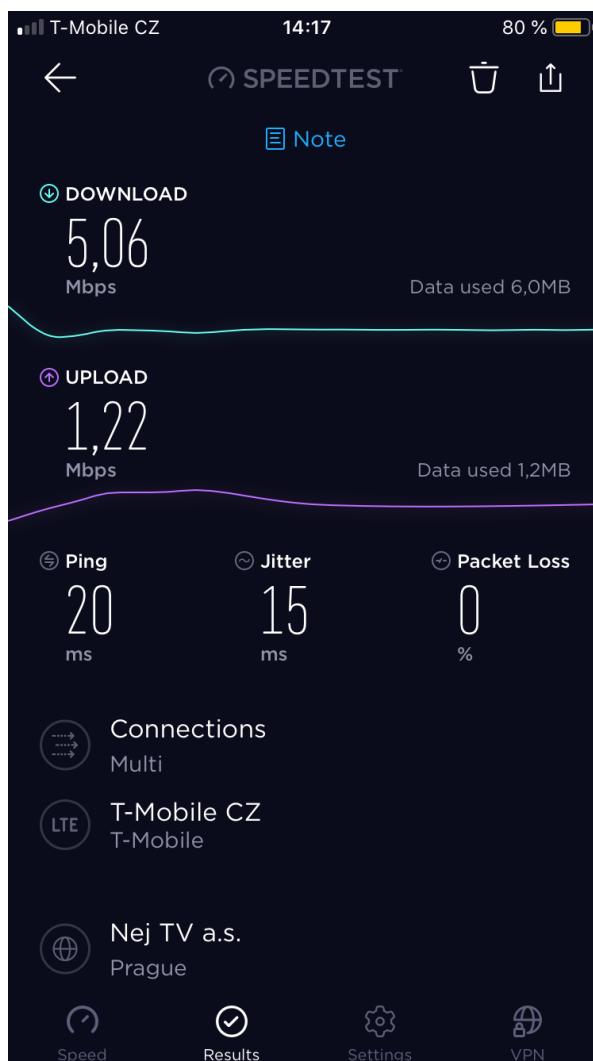
### 4.1 Speedtest

Následující kapitola vychází z oficiálních stránek a dokumentace aplikace Speedtest [\[10\]](#).

Mezi světově nejznámější a nejpoužívanější aplikace pro měření parametrů internetového připojení je Speedtest, která byla vytvořena a je spravována společností Ookla. Aplikace je k dispozici pro všechny dominantní platformy – iOS, Android, macOS, Windows, Google Chrome a Apple TV. Dále je možné využít také webovou verzi aplikace, která je k dispozici na adrese [www.speedtest.net](http://www.speedtest.net). Aplikace je k dispozici pro soukromé využití zdarma.

Aplikace měří rychlost stahování, rychlost nahrávání, odezvu, jitter a ztrátu paketů. V aplikaci bohužel chybí možnost zvolit částečný test (tj. zvolit například pouze měření rychlosti stahování, popřípadě rychlost nahrávání apod.). Aplikace vždy měří všechny možné parametry, což v případě použití mobilního připojení může mít za následek zbytečnou spotřebu mobilních dat. V aplikaci je možné zvolit server, vůči kterému se budou parametry měřit. Ve výchozím nastavení se aplikace pokouší nalézt nejbližší možný server pro měření.

Aplikace automaticky ukládá výsledky testů, které byly provedeny. Ve výsledcích testů je možné zpětně dohledat naměřené parametry – rychlost stahování, rychlost nahrávání, ping, jitter a ztrátu paketů. Dále je uložen server, vůči kterému se měřilo, mobilní datová technologie použita v testu (nebo Wi-Fi, popřípadě LAN), mobilní operátor a přibližná poloha, kde bylo měření prováděno.



**Obrázek 5 - Ukázka aplikace Speedtest v prostředí iOS (Výsledek testu)**

Testování je prováděno převážně skrze TCP. Pro zachování maximální kompatibility je k dispozici také http fallback. Jak aplikace Speedtest měření jednotlivých parametrů provádí je uvedeno v oficiální dokumentaci aplikace. [\[11\]](#)

Test doby odezvy a jitteru se provádí měřením času, který server potřebuje, aby odpověděl na požadavek ze strany klienta. Klient odešle zprávu na server, který po přijetí zprávy pošle zpět odpověď. Toto se opakuje několikrát, přičemž konečný výsledek je měřen v milisekundách a určuje jej nejnižší naměřená hodnota.

Test rychlosti stahování probíhá tak, že klient naváže více připojení na server skrze port 8080. Klient požádá server, aby odeslal klientovi počáteční testovací data. Klient si vypočítá rychlost přenosu počátečních dat, na základě, které si klient zvolí požadovanou velikost testovacích souborů (které bude žádat po serveru) pro

maximální využití síťového připojení v samotném testu. Klient si rovněž v průběhu testu může vyžádat další připojení k serveru a další testovací soubory, což by mělo vést k přesnějšímu výsledku testu. Test je ukončen, je-li vyčerpán čas určený pro test.

Rychlost nahrávání je měřena podobně jako rychlost stahování. Klient vytvoří několik spojení na server skrze definovaný port a odešle na server počáteční testovací data. Klient si vypočítá rychlost přenosu počátečních dat, na základě, kterých si zvolí určitou velikost testovacích souborů, které bude posílat na server v průběhu testu. Klient opět odešle několik bloků (souborů) v průběhu testu. Test je ukončen, je-li vyčerpán čas určený pro test.

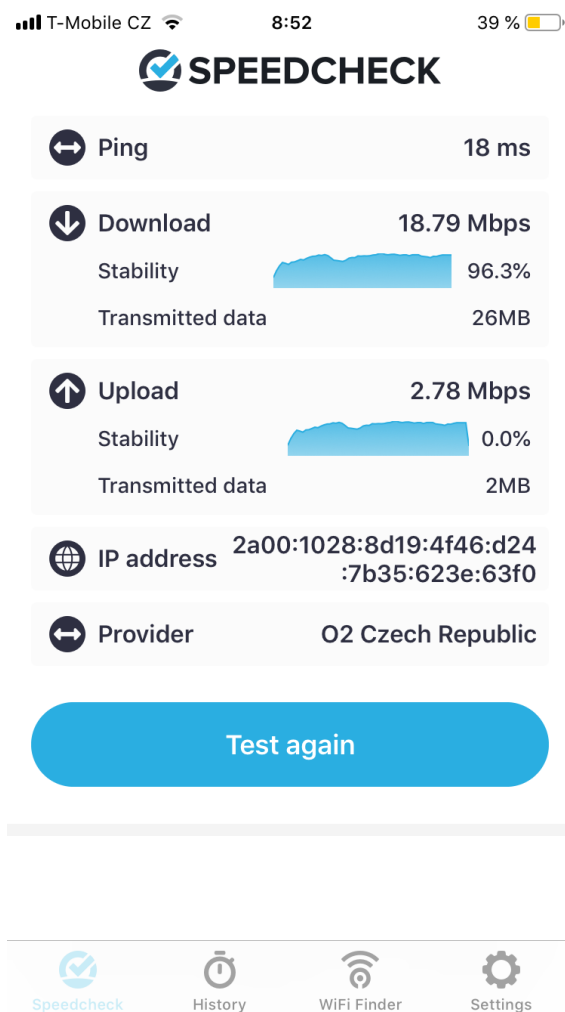
## **4.2 SPEEDCHECK Internet Speed test**

Další známou a často používanou aplikací pro měření parametrů internetového připojení je aplikace SPEEDCHECK Internet Speed test. Měření parametrů internetového připojení je možné provést z prostředí webového prohlížeče <sup>4</sup>, nebo pomocí mobilní aplikace, která je k dispozici pro Android a iOS.

Aplikace měří tři základní parametry internetového připojení – rychlost stahování, rychlost nahrávání a odezvu. V samotných výsledcích měření je zobrazeno množství dat, které bylo spotřebováno v průběhu testu rychlosti stahování a rychlosti nahrávání. Aplikace se také snaží měřit stabilitu rychlost stahování a rychlosti nahrávání. Tyto hodnoty mi však v rámci testů přišli zkreslené a nepřesné. V aplikaci chybí možnost výběru server, vůči kterému se bude internetové připojení měřit. Také chybí možnost zvolení jednotlivého testu – např. test rychlosti stahování, a je nutné tak vždy testovat všechny tři parametry. Aplikace umožňuje zobrazení historie jednotlivých testů, které byly na zařízení provedeny, a to jak v mobilních aplikacích, tak také na webu.

---

<sup>4</sup> Na adrese <https://www.speedcheck.org>.



Obrázek 6 – Ukázka aplikace SPEEDCHECK v prostředí iOS.

### 4.3 SpeedSmart

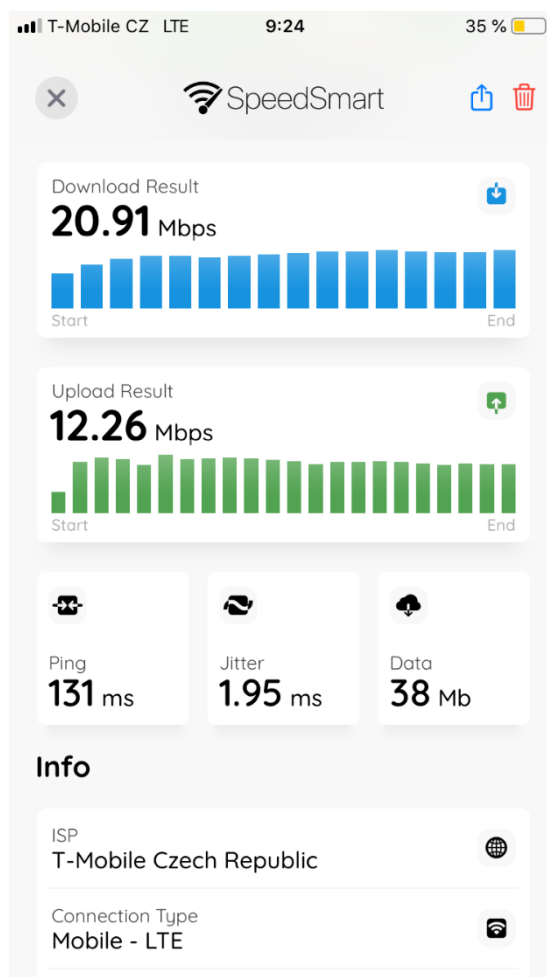
Mezi další populární nástroje pro měření parametrů internetového připojení se řadí aplikace SpeedSmart, která je dostupná pro mobilní platformy Android a iOS. Využit je ale také možné HTML5 test v prostředí webového prohlížeče <sup>5</sup>.

Aplikace SpeedSmart měří celkem čtyři parametry internetového připojení – rychlost stahování, rychlost nahrávání, odezvu a jitter. Testování všech čtyřech parametrů je vždy prováděno najednou – chybí možnost zvolit, který konkrétní parametr bychom chtěli testovat. V mobilních aplikacích, ale také přímo ve

---

<sup>5</sup> Na adrese <https://speedsmart.net>.

webovém prohlížeči je možnost vybrat si server, vůči kterému se budou testy provádět.



**Obrázek 7 – Ukázka zobrazení výsledku měření, aplikace pro iOS.**

Aplikace SpeedSmart automaticky ukládá výsledky testů, které byly v minulosti na zařízení provedeny. Ve výsledcích je rovněž zobrazeno množství dat, které bylo v průběhu testu spotřebováno. To je užitečné zejména pro mobilní internetová připojení. Metodika jednotlivých testů bohužel není zveřejněna.

#### **4.4 RYCHLOST.CZ**

Dalším nástrojem pro měření rychlosti internetu je RYCHLOST.CZ. K dispozici jsou mobilní aplikace pro mobilní platformy Android a iOS apod. Test je možné také

provést přímo na webové stránce <sup>6</sup> v prostředí webového prohlížeče. Pokud bychom se rozhodli provést měření přímo ve webovém prohlížeči mobilního zařízení, tak webová stránka je jednoduchá a dobře funguje také na mobilních zařízeních s relativně malým displejem. K dispozici je velké množství jazykových mutací.

Pomocí nástroje RYCHLOST.CZ lze (podobně jako v případě výše zmíněných mobilních aplikací) měřit všechny základní parametry internetového připojení – rychlost stahování, rychlost nahrávání, odezvu a jitter. K dispozici je rovněž také nástroj, určený pouze k měření rychlosti odezvy. Bohužel zde chybí možnost samotného testu pro rychlost stahování nebo rychlost nahrávání. Před spuštěním samotného testu je možné vybrat server, vůči kterému se budou jednotlivé testy provádět. K dispozici je výběr z 54 <sup>7</sup> serverů, z nichž naprostá většina se nachází na území Evropy. K dispozici je malé množství serverů nacházejících se na území USA, nebo např. Austrálie.

Na webové stránce RYCHLOST.CZ je možné si zobrazit historii měření rychlosti připojení, která je vztažena k IP adrese. Zde může docházet k zobrazení uživatelem nerealizovaných měření z důvodu, že IP adresa může být v závislosti na poskytovateli internetového připojení používána a sdílána více subjekty.

---

<sup>6</sup> Na adrese <https://rychlost.cz>.

<sup>7</sup> Stav k prosinci 2019.



**Obrázek 8 – Ukázka zobrazení výsledků v prostředí iOS a webového prohlížeče Safari.**

Na stránce nástroje RYCHLOST.CZ je uvedena stručná metodika fungování testu rychlosti stahování a rychlosti nahrávání. [\[12\]](#)

K dispozici jsou 3 verze měření rychlosti připojení.

- Flash verze.
- Stará javascriptová verze (2014)
- Nová javascriptová verze (HTML5)

Test rychlosti stahování probíhá tak, že je nejprve stažen testovací soubor o velikosti přibližně 100 kB. Pokud je tento testovací soubor stažen za více jak 5 sekund, je test rychlosti stahování ukončen. Pokud je testovací soubor stažen za kratší dobu než 5 sekund, je vypočtena nová velikost testovacího souboru, přičemž maximální velikost testovacího souboru je cca 5000 kB. V případě, že testovací soubor o velikosti cca 5000 kB je stažen za méně jak 5 sekund je již test považován za platný a není počítána nová velikost testovacího souboru.

Test rychlosti nahrávání probíhá obdobně jako test rychlosti stahování. K testu rychlosti nahrávání jsou použita data z testu rychlosti stahování. Maximální velikost souboru je v případě testu rychlosti nahrávání cca 1500 kB.

## **4.5 DSL.CZ**

Nástroj DSL.CZ podobně jako předchozí nástroj RYCHLOST.CZ měří rychlosti internetového připojení ve webovém prohlížeči<sup>8</sup>. K dispozici jsou však také mobilní aplikace pro Android a iOS. Samotná webová stránka, kde probíhá měření internetové připojení je jednoduchá a bez problému se zobrazuje také na displejích mobilních zařízení.

Samotný test měří základní parametry internetového připojení – rychlost stahování, rychlost nahrávání a odezvu, jitter chybí. Bohužel opět zde chybí možnost zvolit test pouze jednoho z parametrů (rychlost, stahování, rychlost nahrávání, popřípadě ping). Chybí možnost zvolení serveru, vůči kterému se bude měření provádět. V případě mobilních aplikací je možné zobrazit si historii provedených měření internetového připojení, na webu tato možnost zcela chybí.

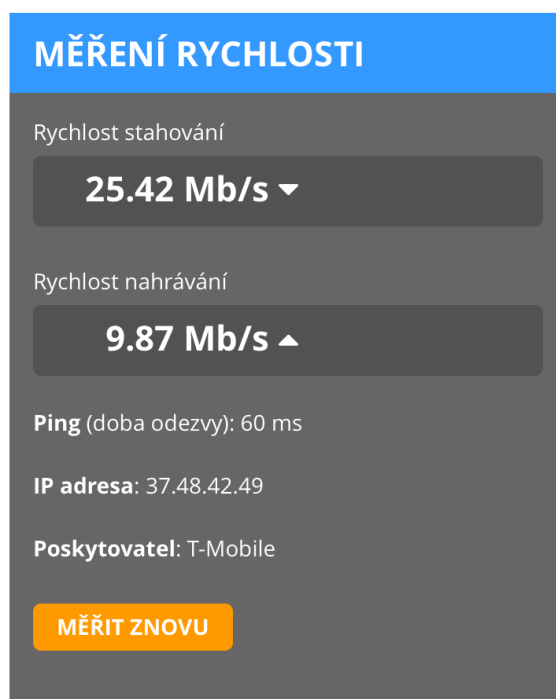
Metodika prováděných testů bohužel není zveřejněna.

---

<sup>8</sup> K dispozici na <http://www.dsl.cz/mereni-rychlosti>. [13]



## Výsledky měření rychlosti internetového připojení



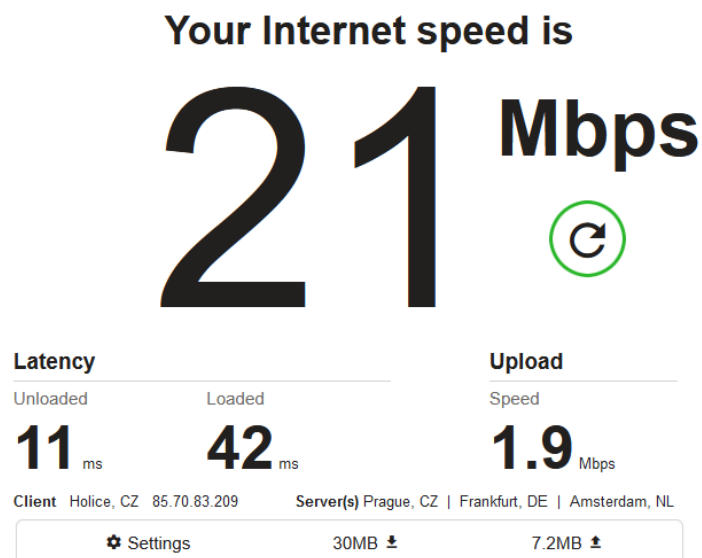
**Obrázek 9 - Ukázka zobrazení výsledků v prostředí iOS a webového prohlížeče Safari.**

### **4.6 FAST Speed Test**

Jako poslední jsem si zvolil aplikaci FAST Speed Test. Aplikace je k dispozici pro mobilní platformy Android a iOS, měření je však možné provádět také v prostředí webového prohlížeče <sup>9</sup>. Aplikace umožňuje měřit tři parametry internetového připojení – rychlost stahování, rychlost nahrávání a odezvu.

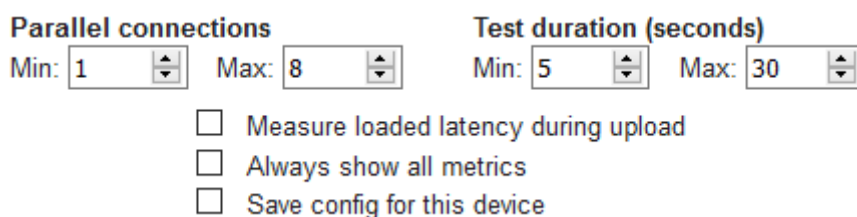
---

<sup>9</sup> Dostupný na adrese <https://fast.com>.



**Obrázek 10 – Ukázka aplikace FAST Speed Test v prostředí webového prohlížeče Firefox (verze 71).**

Jedná se o velice jednoduchou aplikaci. Po spuštění aplikace, respektive načtení webové stránky je automaticky spuštěn test parametrů internetového připojení. Vzhledem k jednoduchosti aplikace chybí možnost výběru serveru, vůči kterému se bude měření provádět. Chybí rovněž možnost zvolení parametru, který chceme testovat. Ve výsledcích je zobrazeno také množství dat, které bylo spotřebováno pro test stahování a test nahrávání a server, které byly použity.



**Obrázek 11 – Ukázka nastavení.**

Po dokončení samotného testu je možné otevřít a upravit nastavení samotných testů. Mezi nejvíce užitečné možnosti bych zařadil nastavení paralelních připojení k serveru a možnost nastavení časového rozmezí (doby trvání) samotného trvání testu. Nastavení je vždy možné uložit pro dané zařízení a při příštím spuštění testu bude automaticky použito.

## 4.7 Shrnutí

Všechny zde testované aplikace nabídli možnost testu tří základních parametrů mobilního internetového připojení – rychlosti stahování, rychlosti nahrávání a test odezvy. Polovina aplikací pak nabídla také test dalšího parametru jitteru. Z hlediska samotného testování nejvíce uživatelů zajímá rychlost stahování a odezva.

Tabulka č. 4 uvedená níže nabízí shrnující pohled na jednotlivé parametry internetového připojení, které aplikace testují.

**Tabulka 4 – Shrnutí možnosti testování parametrů.**

<i>Aplikace</i>	<i>Rychlost stahování</i>	<i>Rychlost nahrávání</i>	<i>Odezva</i>	<i>Jitter</i>
<b>Speedtest</b>	<b>ANO</b>	<b>ANO</b>	<b>ANO</b>	<b>ANO</b>
<b>SPEEDCHECK</b>	<b>ANO</b>	<b>ANO</b>	<b>ANO</b>	NE
<b>SpeedSmart</b>	<b>ANO</b>	<b>ANO</b>	<b>ANO</b>	<b>ANO</b>
<b>RYCHLOST.CZ</b>	<b>ANO</b>	<b>ANO</b>	<b>ANO</b>	<b>ANO</b>
<b>DSL.CZ</b>	<b>ANO</b>	<b>ANO</b>	<b>ANO</b>	NE
<b>FAST</b>	<b>ANO</b>	<b>ANO</b>	<b>ANO</b>	NE

Všechny testované nástroje disponují vlastní aplikací pro mobilní platformy Android a iOS. Právě tyto dvě platformy mají na trhu, zejména pak mezi mobilními telefony, největší podíl.

Polovina z testovaných aplikací umožní před samotným spuštěním měření vybrat server, vůči kterému se budou jednotlivé testy provádět. Většina aplikací rovněž zobrazí množství dat, které bylo v rámci měření spotřebováno. To může být užitečné především pro majitelé, kteří využívají mobilní internetové připojení. Všechny testované aplikace, s výjimkou aplikace FAST ukládají výsledky jednotlivých testů a umožní tak zobrazit historii již provedených měření.

Tabulka č. 5 uvedená na straně 23 nabízí shrnující pohled na dodatečné možnosti jednotlivých aplikací pro měření internetového připojení.

**Tabulka 5 - Shrnutí dodatečných možností aplikací.**

<i>Aplikace</i>	<i>Výběr serveru</i>	<i>Množství spotřebovaných dat</i>	<i>Historie provedených testů</i>
<b>Speedtest</b>	<b>ANO</b>	<b>ANO</b>	<b>ANO</b>
<b>SPEEDCHECK</b>	<b>NE</b>	<b>ANO</b>	<b>ANO</b>
<b>SpeedSmart</b>	<b>ANO</b>	<b>ANO</b>	<b>ANO</b>
<b>RYCHLOST.CZ</b>	<b>ANO</b>	<b>NE</b>	<b>ANO</b>
<b>DSL.CZ</b>	<b>NE</b>	<b>NE</b>	<b>ANO</b>
<b>FAST</b>	<b>NE</b>	<b>ANO</b>	<b>NE</b>

## 5 Analýza vlastního řešení

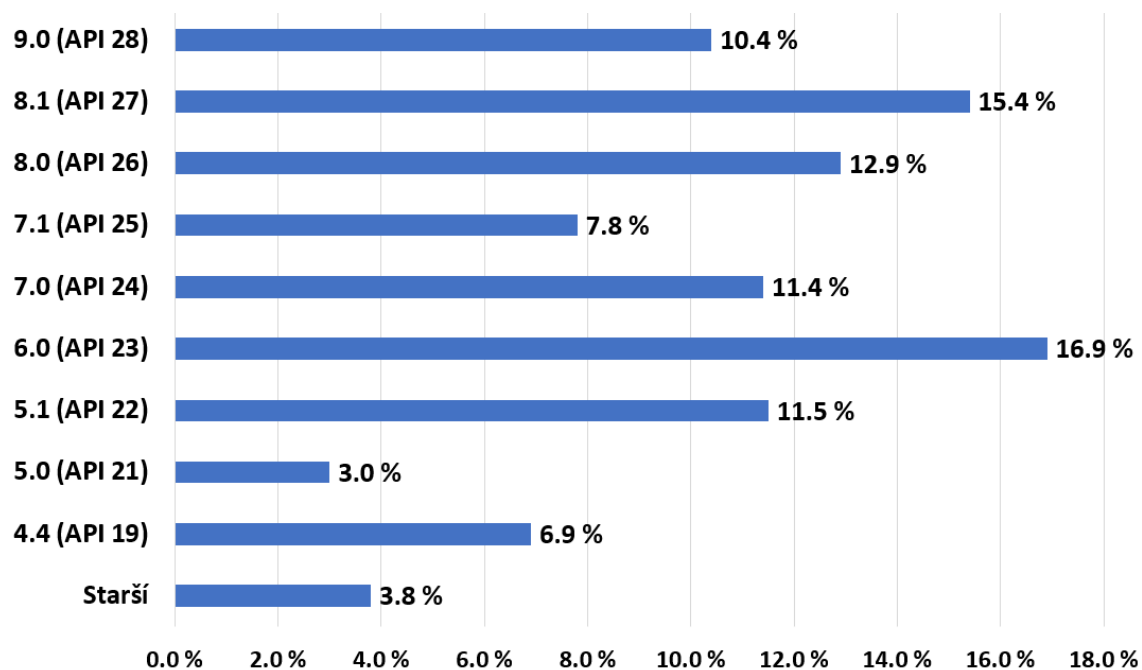
### 5.1 Android

Výslednou aplikaci cílím na operační systém (mobilní platformu) Android. Aktuálně se jedná o nejrozšířenější a nejvíce používaný mobilní operační systém mezi mobilními telefony a tablety. Dle zdroje [27] ovládal operační systém Android trh mobilních operačních systémů s podílem 74,13 procenta.

První verze operačního systému Android 1.0 byla poprvé představena v roce 2008. Nejnovější verze Android 10 byla představena v září 2019. [28]

Výslednou aplikaci pak cílím na Android 5.0 (API 21) a vyšší. Tato verze operačního systému android byla představena v první polovině roku 2014. V této verzi bylo přidáno nové API rozhraní pro práci se sítí – např. třída „ConnectivityManager“. [29]

Systémem Android 5.0 a novějším v současné době disponuje více než 90% zařízení. Z důvodu rychlého vývoje mobilního operačního systému Android již nemá smysl cílit na verze Android 4.4 a nižší.



Obrázek 12 – Procentuální podíl jednotlivých verzí

Graf zobrazený výše znázorňuje procentuální zastoupení jednotlivých verzí operačního systému Android k prosinci 2019. [30]

Základním stavebním kamenem aplikací pro operační systém Android jsou tzv. aktivity. Ke každé aktivitě vždy přísluší tzv. layout (grafické rozvržení dané aktivity). Aktivita obsahuje programový kód a samotnou programovou logiku (v tomto případě využít jazyk Java). Layout pak obsahuje popis jednotlivých grafických komponent v aktivitě a jejich vazby.

### 5.1.1 Jazykové mutace

Android Studio umožňuje velice jednoduše tvořit mobilní aplikace, které budou obsahovat několik jazykových mutací.

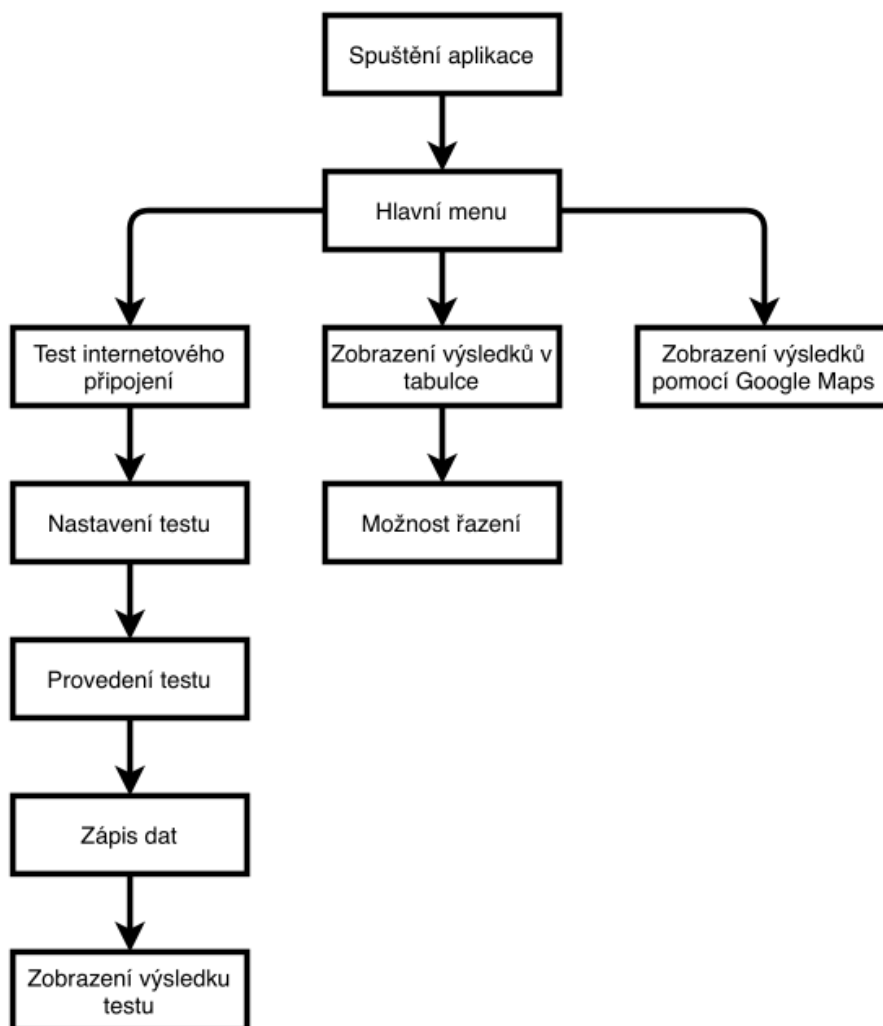
Obrázek níže demonstruje použití editoru překladů pro vytváření jazykových mutací aplikace. Mobilní aplikace vytvořena v rámci této diplomové práce obsahuje anglickou (výchozí) a českou jazykovou mutaci.

Key	Resource Fol...	Untra...	Default Value	Czech (cs)
app_name	app\src\mai	<input type="checkbox"/>	Diplom	Diplom
google_maps_key	app\src\deb	<input checked="" type="checkbox"/>	AlzaSyDSp5v9DfNh	
stringAuto	app\src\mai	<input type="checkbox"/>	Auto	Automaticky
stringComplete	app\src\mai	<input type="checkbox"/>	Complete	Kompletní
stringDownloadOnly	app\src\mai	<input type="checkbox"/>	Download only	Pouze rychlost stahování
stringManual	app\src\mai	<input type="checkbox"/>	Manual	Manuálně
stringPingOnly	app\src\mai	<input type="checkbox"/>	Ping only	Pouze rychlost odezvy

**Obrázek 13 - Ukázka editoru překladů v prostředí Android Studio**

Pod jedním klíčem je uloženo více překladů daného textu, přičemž operační systém Android automaticky vybere nejvhodnější překlad v závislosti na nastavení jazykových preferencí v systému. Jako výchozí hodnota klíče by měl být anglický překlad (vzhledem k potencionálně největšímu rozšíření).

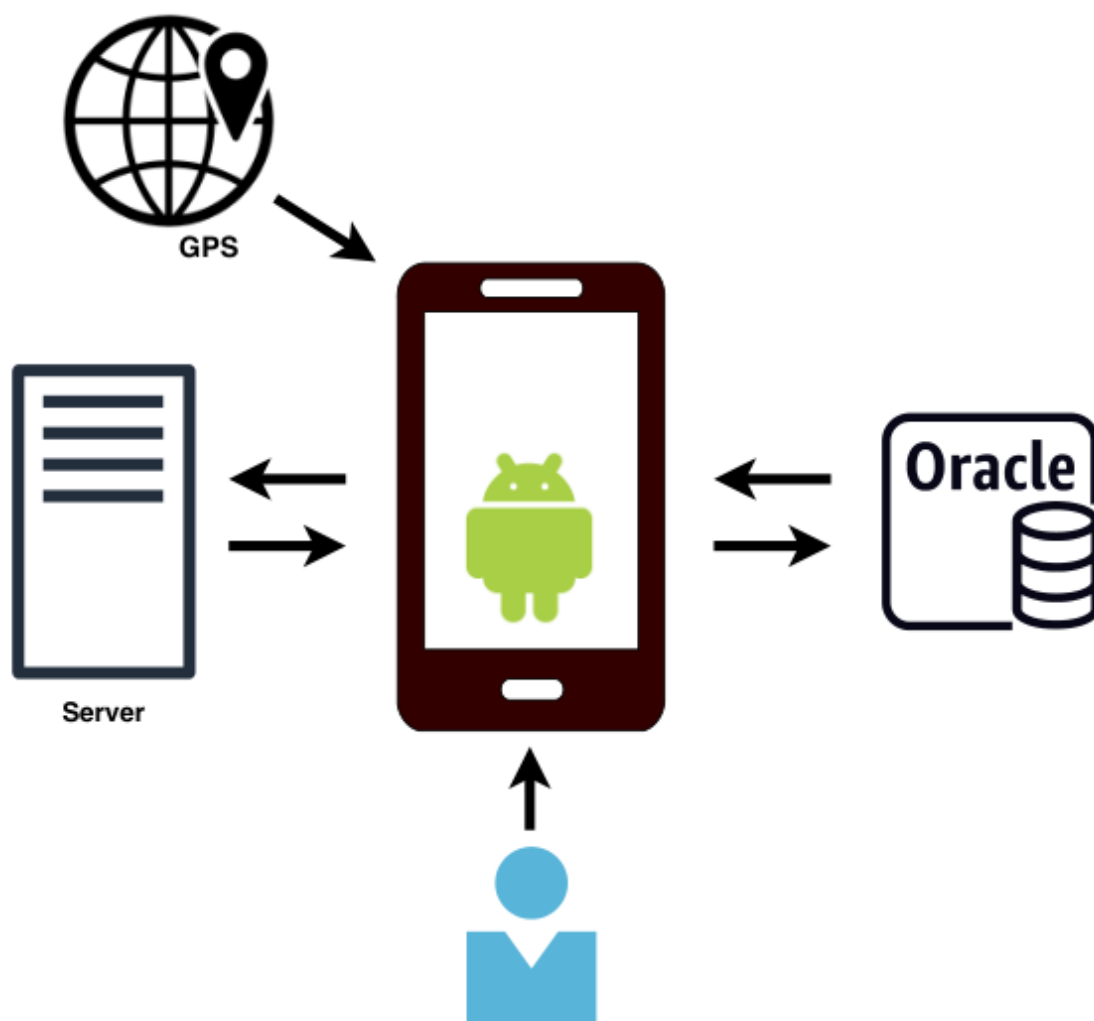
## 5.2 Obecný návrh mobilní aplikace



Obrázek 14 – Základní schéma mobilní aplikace

Diagram zobrazený výše znázorňuje základní schéma mobilní aplikace z pohledu uživatele. Po spuštění aplikace se zobrazí uživateli hlavní menu, ve kterém bude mít uživatel na výběr ze tří možností.

- Přejít na aktivitu, kde si uživatel bude moci zvolit nastavení samotného testu a test poté spustit.
- Přejít na aktivitu, kde budou v tabulkové podobě zobrazeny již provedené výsledky.
- Přejít na aktivitu, kde budou na mapě zobrazeny již provedené výsledky.



**Obrázek 15 – Zjednodušený návrh mobilní aplikace**

Na obrázku č. 15 je znázorněn zjednodušený návrh mobilní aplikace. Mobilní aplikace přistupuje na server, kam posílá požadavky na měření. V průběhu měření mohou být také uloženy GPS souřadnice. Naměřené hodnoty jsou ukládány co Oracle SQL databáze.

### **5.3 Měření parametrů internetového připojení**

Základní myšlenkou (vlastností) operačního systému Android je, že síťové operace mohou být časově náročné a musí být prováděny mimo hlavní vlákno, které zajišťuje běh samotné aplikace. Vzhledem k tomuto faktu bude muset být pro samotné síťové spojení, stahování testovacího souboru a nahrávání testovacího souboru vytvořeno vlákno vedlejší, které bude zpracovávat síťové požadavky.



### 5.3.1 Rychlost nahrávání a rychlost stahování

Pro algoritmus na měření rychlosti stahování a rychlosti nahrávání bude využito připojení na vzdálený SFTP server prostřednictvím knihovny JSCH. SFTP server je hostován v prostředí Univerzity Hradec Králové a jeho garantovaná rychlost je 500 Mbps (tj. jeho minimální rychlost), což je pro testovací účely zcela dostatečné, zejména pak po testování mobilních datových technologií. V průběhu měření rychlosti stahování, resp. rychlosti nahrávání bude využit SFTP server pro stahování, resp. nahrávání testovacího(ch) souboru(ů).

Před samotným spuštěním testu bude provedena detekce aktuálně používané mobilní datové technologie (popř. Wi-Fi), na základě, které bude zvolena vhodná velikost testovacího souboru vzhledem k používané technologii. Před spuštěním samotného testu, bude proveden tzv. „před test“, která má za cíl detekovat možné omezení mobilního internetového připojení prostřednictvím tzv. FUP ze strany mobilního operátora. Toto bude provedeno pomocí stažení testovacího souboru o velikosti 200 kB. V případě, že bude doba stahování, resp. nahrávání větší než specifikovaný časový okamžik, je s největší pravděpodobností aplikováno FUP (nebo je omezena rychlost mobilní datové technologie např. velmi špatným signálem) a nemá smysl pokračovat ve stahování, resp. nahrávání většího testovacího souboru, jelikož by tato operace trvala příliš dlouho. V případě manuálního zvolení velikosti testovacího souboru není detekce FUP prováděna.

Dále bude přes samotným přenosem (stahování, nahrávání) testovacího souboru zaznamenán systémový čas. Po skončení přenosu bude čas opět zaznamenán a bude vypočtena doba přenosu souboru – bude prováděno samostatně pro test rychlosti nahrávání a pro test rychlosti stahování. Čas připojení a odpojení od serveru se do tohoto času nezapočítává. Vzhledem k faktu, že známe velikost testovacího souboru a čas přenosu, jsme schopni jednoduše vypočítat rychlost přenosu.

### 5.3.2 Měření odezvy

První myšlenkou při vytváření algoritmu na bylo využít systémový příkaz ping. Tento příkaz, který je dobře známý z ostatních operačních systémů, jakými jsou

např. Windows, či Linux se však bohužel ve výsledku ukázal jako zcela nevhodný, a to hned z několika důvodů.

Jako první problém se ukázalo, že příkaz ping není podporován stejně ve všech distribucích operačního systému Android. V některých distribucích příkaz ping dokonce není vůbec k dispozici. Dalším problémem bylo, že výstup z příkazu ping v prostředí operačního systému Android se značně odlišuje v porovnáním např. s výstupem v prostředí operačního systému Windows. Z výstupu příkazu ping v operačním systému Android nelze v některých distribucích získat časy jednotlivých odpovědí ze strany serveru.

Z výše uvedených důvodů jsem se nakonec rozhodl pro měření odezvy internetového připojení využít třídu Socket. Algoritmus pak měří dobu, za kterou proběhne připojení k danému serveru.

## 6 Návrh a implementace

### 6.1 Použité technologie

#### 6.1.1 Programovací jazyk Java

Java je objektově orientovaný programovací jazyk. Programovací jazyk Java byl vyvinut společností Sun Microsystems na počátku devadesátých let, přičemž první verze Java 1.0 byla oficiálně představena v roce 1995, aktuálně poslední verzí je Java 12 (březen 2019). Aktuálně se jedná o jeden z nejvíce rozšířených a používaných programovacích jazyků [20].

Shankar v článku What is Java [19] dále uvádí, že mezi hlavní přednosti jazyka patří, že je multiplatformní (tj. nezáleží na prostředí, ve kterém je výsledný program spuštěný, je tedy snadno přenositelný). Jazyk Java rovněž disponuje automatickou alokací a dealokací paměti (angl. garbage collector). Mezi další výhody patří jednoduchost a univerzálnost jazyka.

Výsledná mobilní aplikace, která je vytvářena v rámci této diplomové práce je napsaná v programovacím jazyce Java a cílí na mobilní platformu Android.

#### 6.1.2 Vývojové prostředí Android Studio

Android Studio je oficiální vývojové prostředí (IDE) určené pro vývoj aplikací pro mobilní platformu Android. První verze byla oficiálně představena v květnu 2013. Android Studio je založené na vývojovém prostředí IntelliJ IDEA. Ve vývojovém prostředí Android Studio je možné výsledné aplikace spouštět buď ve virtuálním mobilním telefonu, nebo je možné aplikace testovat a spouštět na fyzickém zařízení.

Na oficiálních stránkách produktu [21] se uvádí, že mezi hlavní výhody vývojového prostředí Android Studio patří výkonný emulátor, předpřipravené šablony kódů, integrace verzovacího systému GitHub apod. Android Studio je univerzální vývojové prostředí, ve kterém je vytvářet aplikace na všechna zařízení, kde je k dispozici operační systém Android.

Pro vývoj mobilní aplikace byla použita verze Android Studio 3.5.

### 6.1.3 Knihovna JSch

JSch je knihovna napsaná čistě v jazyce Java a jedná se o implementaci SSH2. Knihovna umožňuje navázat připojení ke vzdálenému sshd serveru a využít tzv. „port forwarding“, přenos souborů atd. Knihovna je volně dostupná pod licencí BSD. [\[22\]](#)

V této práci je knihovna JSch využita pro připojení ke vzdálenému SFTP serveru a následném nahrávání a stahování souborů.

V rámci projektu byla použita knihovna JSch ve verzi 0.1.55.

### 6.1.4 Oracle DB

Oracle DB je SQL databázový systém vyvinutý společností Oracle. Databázový systém je multiplatformní, nabízí pokročilé možnosti spravování a zpracování dat, vysoký výkon a snadnou škálovatelnost. Pro samotnou práci s databází je využit jazyk SQL, popřípadě jazyk PL/SQL.

Oracle DB implementuje systém relační databáze. Relační databáze je databáze, která je založena na tabulkách, ve kterých jsou uložena samotná data. [\[23\]](#)

### 6.1.5 Oracle SQL Developer

Oracle SQL Developer je vývojové prostředí (IDE) pro práci s SQL primárně určeno pro práci v databázích Oracle. Jedná se o multiplatformní vývojové prostředí napsané v jazyce Java. Toto vývojové prostředí je společností Oracle poskytováno zdarma.

Ve vývojovém prostředí Oracle SQL Developer jsou posílány SQL příkazy do databáze. Pomocí těchto příkazů je možné zobrazit, vytvářet a upravovat databázové objekty. Oracle SQL Developer dále nabízí možnost vytvářet složitější funkce a procedury v programovacím jazyce PL/SQL, které lze provádět a volat přímo v databázích Oracle. [\[24\]](#)

### 6.1.6 JDBC

JDBC (Java Database Connectivity) je aplikační programovací rozhraní (API) pro programovací jazyk Java, které definuje, přístup klienta k databázi. JDBC je součástí

Javy SE. JDBC podporuje vytváření a spouštění příkazů, jakými jsou například: SELECT, CREATE, INSERT, UPDATE a DELETE.

Pro přístup ke konkrétnímu databázovému serveru je potřeba JDBC driver, který poskytuje tvůrce databázového serveru, v mém případě společnost Oracle – Oracle JDBC. [\[25\]](#)

V praktické části bude použit Oracle JDBC ve verzi 8, který umožní připojení k Oracle databázi až do verze 19c (19.3.). [\[26\]](#)

### **6.1.7 Maps SDK for Android**

Maps SDK for Android je API rozhraní, které automaticky zpracovává přístup k mapám Google v prostředí operačního systému Android. Vývojové prostředí Android Studio obsahuje přímo podporu pro použití Maps SDK v mobilních aplikacích.

S využitím Maps SDK for Android je v aplikacích na mapě možné zobrazit vlastní objekty, např.: ikony ukotvené na konkrétní pozici v mapě (tzv. markery), sady úseček, uzavřené obrazce (tzv. polygony) a další.

Maps SDK for Android je k dispozici zdarma, ale je nutné provést generaci API klíče. API klíč je jedinečný identifikátor v rámci projektu, který se používá k ověření požadavků spojených s projektem. [\[31\]](#)

## **6.2 Oracle DB**

### **6.2.1 Struktura**

Pro ukládání jednotlivých výsledků byla vybrána databáze Oracle. Vzhledem k faktu, že je potřebné ukládat pouze výsledky jednotlivých testů, databáze obsahuje pouze jedinou tabulku MEASUREMENTS.

Primárním klíčem v tabulce je atribut MEASUREMENT\_ID, který jednoznačně identifikuje právě jedno měření. Pro zajištění automatického generování jednoznačného identifikátoru je využita sekvence MEASUREMENT a každá další hodnota je generována pomocí MEASUREMENT.NEXTVAL. Atribut DEVICE\_ID identifikuje dané zařízení, na kterém bylo měření prováděno. Atributy LATITUDE a LONGITUDE určují polohu, kde bylo měření prováděno. Naměřené parametry

internetového připojení jsou uloženy v attributech DOWNLOAD, UPLOAD a PING. Další dva atributy, FILESIZE určuje velikost testovacího souboru, který byl použit a TYPE určuje typ mobilní datové technologie. Poslední atribut DATETIME uchovává datum a čas, kdy bylo měření provedeno.

MEASUREMENTS		
P *	MEASUREMENT_ID	NUMBER (*,0)
*	DEVICE_ID	VARCHAR2 (64 BYTE)
	LATITUDE	FLOAT (126)
	LONGITUDE	FLOAT (126)
	DOWNLOAD	FLOAT (126)
	UPLOAD	FLOAT (126)
	PING	FLOAT (126)
*	FILESIZE	NUMBER
*	TYPE	VARCHAR2 (20 BYTE)
*	DATETIME	TIMESTAMP
	MEASUREMENTS_PK (MEASUREMENT_ID)	
	MEASUREMENTS_PK (MEASUREMENT_ID)	

**Obrázek 16 – Struktura tabulky MEASUREMENTS**

Z obrázku výše vyplývá, že povinnými atributy jsou MEASUREMENT\_ID, DEVICE\_ID, FILESIZE, TYPE a DATETIME. Ostatní atributy povinné nejsou, a to z důvodu, že GPS lokace může být zakázána uživatelem, popřípadě nedostupná a současně je možné provádět testování pouze jednoho z parametrů internetového připojení (tj. zbývající dva parametry měřeny nebyly a v databázi budou nabývat hodnoty NULL).

### 6.2.2 Práce s databází – třída SQLHelper

Pro samotnou práci z databází (načítání a zápis dat) je vytvořena pomocná třída SQLHelper. Tato třída využívá objekty z knihovny Oracle JDBC.

Třída SQLHelper obsahuje následující metody.

- Metoda connect, která vytváří spojení se samotnou databází a inicializuje objekt Connection. Pro připojení jsou využity údaje (server, port, přihlašovací jméno a heslo) z privátních proměnných, které byly inicializovány v konstruktoru třídy.
- Metoda commit, která ukončuje otevřenou databázovou transakci a uloží změny provedené v databázi.

- Metoda disconnect, která ukončuje spojení s databází.
- Metoda getData, která vrací datovou strukturu (objekt) ArrayList s vybranými výsledky z databáze. Parametrem metody je pak konkrétní SQL příkaz, který se má provést vyhledání požadovaných dat.
- Metoda writeData, která do databáze uloží objekt typu Result, který je předán v parametru metody.

Jelikož síťové operace mohou být náročné na čas, nelze je provádět přímo v hlavním vlákne. Z tohoto důvodu je ve třídách, které pracují s databází vytvořena pomocná vnitřní třída, která rozšiřuje třídu AsyncTask. V této třídě pak probíhá v druhém vlákne práce z databází.

```

ArrayList<Result> list;

@Override
protected ArrayList<Result> doInBackground(Void... voids) {
    SQLHelper sqlHelper = new SQLHelper(
        server: "xxx",
        username: "xxx",
        password: "xxx",
        port: "xxx");

    try {
        sqlHelper.connect();
        list = sqlHelper.getData( sql: "SELECT * FROM MEASUREMENTS");
        sqlHelper.disconnect();
    } catch (SQLException | ClassNotFoundException ex) {
        sqlHelper.disconnect();
        ex.printStackTrace();
    }

    return list;
}

```

### Ukázka kódu 1 – Získání všech výsledků

Ukázka kódu č. 1 demonstruje načítání dat z databáze. V tomto konkrétním případě je vytvořeno spojení s databází, z tabulky jsou vybrány všechny uložené výsledky (všechny řádky tabulky), které jsou následně uloženy do proměnné list. Po uložení všech hodnot je připojení s databází ukončeno a výsledek dotazu je v proměnné list vrácen z metody.

```

@Override
protected void doInBackground(Result... results) {
    SQLHelper sqlHelper = new SQLHelper(
        server: "xxx",
        username: "xxx",
        password: "xxx",
        port: "xxx");

    try {
        sqlHelper.connect();
        sqlHelper.writeData(results[0]);
        sqlHelper.disconnect();
    } catch (SQLException | ClassNotFoundException ex) {
        sqlHelper.disconnect();
        ex.printStackTrace();
    }

    return null;
}

```

### Ukázka kódu 2 - Zápis objektu Result do databáze.

Ukázka kódu č. 2 zobrazuje zápis objektu Result do Oracle databáze. Zápis probíhá v metodě doInBackground, kde se jako parametr předá objekt typu Result, který se má do databáze uložit. Běh této metody probíhá mimo hlavní vlákno.

Samotný zápis dat do databáze pak má na starosti metoda writeData ve třídě SQLHelper s využitím objektu PreparedStatement, jak znázorňuje ukázka kódu č. 3.

```

public void writeData(Result result) throws SQLException {
    PreparedStatement stmt=con.prepareStatement( sql: "INSERT INTO MEASUREMENTS VALUES" +
        "(MEASUREMENT.NEXTVAL, ?, ?, ?, ?, ?, ?, ?, ?, CURRENT_TIMESTAMP)");

    stmt.setString( parameterIndex: 1, result.getIdDevice());
    stmt.setFloat( parameterIndex: 2, result.getLatitude());
    stmt.setFloat( parameterIndex: 3, result.getLongitude());
    stmt.setFloat( parameterIndex: 4, result.getDownload());
    stmt.setFloat( parameterIndex: 5, result.getUpload());
    stmt.setFloat( parameterIndex: 6, result.getPing());
    stmt.setInt( parameterIndex: 7, result.getFileSize());
    stmt.setString( parameterIndex: 8, result.getType());

    stmt.executeUpdate();
    commit();
}

```

### Ukázka kódu 3 - Metoda writeData třídy SQLHelper.



### 6.3 Využití SFTP serveru

Samotné měření rychlosti (mobilního) internetového připojení je řešeno pomocí stahování, resp. nahrávání testovacího souboru na SFTP server. K tomu účelu je vytvořena třída SFTPClient, která má na starosti za pomoci knihovny JSch spojení s SFTP serverem a následný přenos souborů.

Třída SFTPClient obsahuje.

- Privátní proměnné host, user a password (typu String), port (typu Integer) a session (datového typu Session).
- Konstruktor třídy, do kterého je nutné jako parametr zadat adresu SFTP serveru, port SFTP serveru, přihlašovací jméno a heslo.
- Metoda connect, která slouží pro vytvoření připojení na SFTP server.
- Metoda download, která obsahuje dva parametry – cestu k souboru na SFTP serveru, který se má stáhnout a cílovou cestu, kam se má daný soubor stáhnout.
- Metoda upload, která obsahuje dva parametry – cestu k souboru, který se má na SFTP server nahrát a cílovou cestu na SFTP serveru, kam se má soubor uložit.
- Metoda disconnect, která ukončuje spojení s SFTP serverem.

```
public void download(String source, String destination) throws JSchException, SftpException {  
    Channel channel = session.openChannel("sftp");  
    channel.connect();  
    ChannelSftp sftpChannel = (ChannelSftp) channel;  
    sftpChannel.get(source, destination);  
    sftpChannel.exit();  
}
```

#### Ukázka kódu 4 - Metoda download.

Ukázka kódu č. 4 zobrazuje metodu download. V případě chyby v průběhu běhu této metody (při stahování souboru z SFTP serveru) je vyvolána výjimka JSchException, popřípadě SftpException. V případě vyvolání jedné z těchto výjimek, je výjimka zachycena a ošetřena na vyšší úrovni aplikace.

## 6.4 Využití GPS

Mobilní aplikace umožňuje detekovat a uložit souřadnice GPS, kde bylo měření parametrů internetového připojení prováděno. K tomu účelu je k dispozici jednoduchá třída GPSHelper, která implementuje rozhraní LocationListener.

Třída GPSHelper obsahuje:

- Privátní proměnné longitude a latitude datového typu float, které reprezentují jednotlivé GPS souřadnice.
- Privátní proměnnou GPSSet datového typu boolean, která indikuje, zda již byly zaznamenány souřadnice GPS.
- Přepsanou metodu onLocationChanged ve které probíhá ukládání GPS souřadnic do proměnných longitude a latitude.
- Metody getLongitude, getLatitude a isGPSSet, které vrací GPS souřadnice, resp. Logickou hodnotu, zda již GPS souřadnice byly zaznamenány.

Ukázka kódu č. 5 ukazuje zobrazuje inicializaci a nastavení po získání souřadnic GPS. Nejprve je vytvořena instance třídy GPSHelper, následně je vytvořena proměna locationManager datového typu LocationManager pomocí které se přistupuje k systémové službě poskytující podporu pro lokaci GPS. Na závěr je provedeno nastavení, jak často se má poloha získávat.

```
GPSHelper gps = new GPSHelper();
LocationManager locationManager =
    (LocationManager) getSystemService(Context.LOCATION_SERVICE);
locationManager.requestLocationUpdates(
    locationManager.GPS_PROVIDER, minTime: 1000, minDistance: 1, gps);
```

### Ukázka kódu 5 - Použití třídy GPSHelper

V průběhu běhu aplikace pak stačí zavolat metody, které vrací jednotlivé GPS souřadnice. V případě, že se nepodařilo získat GPS lokaci (např. umístění zařízení uvnitř budovy) vrací níže uvedené metody hodnotu 0.0f. Pro ověření, že byly GPS souřadnice získány je možné také využít metodu isGPSSet.

```
float tempLatitude = gps.getLatitude();
float tempLongitude = gps.getLongitude();
```

### Ukázka kódu 6 - Získání GPS souřadnic ze třídy GPSHelper

## 6.5 Použití Google Maps API

Zobrazit jednotlivé výsledky měření, za podmínky, že bylo v průběhu testu povoleno detekovat a uložit GPS lokaci, je možné také prostřednictvím Google Maps API. Pro tento účel je vytvořena třída (aktivita) MapsActivity. Pro každou aplikaci, ve které se využívá Google Maps API musí být vygenerován jedinečný tzv. Google Maps Key. Tento klíč je uložen v souboru google\_maps\_api.xml.

Veškeré vykreslování jednotlivých provedených měření probíhá v metodě onMapReady, kterou můžeme vidět níže – ukázka kódu č. 7.

```
@Override
public void onMapReady(GoogleMap googleMap) {

    mMap = googleMap;

    DB2 d = new DB2();

    try {
        ArrayList<Result> data = d.execute().get();

        for (int i = 0; i < data.size(); i++) {
            Result temp = data.get(i);

            LatLng marker = new LatLng(temp.getLatitude(), temp.getLongitude());

            Timestamp ts = temp.getDatetime();
            Date date = new Date();
            date.setTime(ts.getTime());
            String fDate = new SimpleDateFormat("dd.MM.yyyy HH:mm").format(date);

            String title = String.valueOf(temp.getDownload()) + " mbps / "
                + String.valueOf(temp.getUpload()) + " mbps";
            String snippet = "Ping: " + temp.getPing() + " ms, "
                + temp.getType() + ", " + fDate;

            mMap.addMarker(new MarkerOptions()
                .position(marker)
                .title(title)
                .snippet(snippet));

            if (i + 1 == data.size()) {
                mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(marker, 14.0f));
            }
        }
    } catch (ExecutionException | InterruptedException ex) {
        ex.printStackTrace();
    }
}
```

Ukázka kódu 7 - Metoda onMapReady třídy MapsActivity

Třída MapsActivity dále obsahuje privátní vnořenou třídu, jejímž úkolem je získat v samostatném vlákně všechny naměřené výsledky z databáze. Pomocí dotazu, který je uveden na obrázku č. 13 níže jsou získány všechny řádky z tabulky, které jsou seřazeny podle data, kdy bylo měření provedeno. Zároveň jsou vynechány ty měření, kde nebyly zaznamenány a uloženy GPS souřadnice.

```
SELECT * FROM MEASUREMENTS
WHERE LATITUDE != 0.0 AND LONGITUDE != 0.0
ORDER BY DATETIME;
```

### Ukázka kódu 8 - Dotaz na získání měření z databáze

Na počátku metody onMapReady je vytvořena instance třídy DB2. Pomocí metody d.execute().get() jsou získány z databáze měření, které jsou dále zpracovávány. V cyklu jsou postupně procházeny a zpracovány všechny získané měření. Na počátku cyklu je do proměnné temp uložen výsledek, který se bude zpracovávat. Do proměnné marker (datového typu LatLng) jsou uloženy GPS souřadnice, kde bylo měření prováděno. V dalších čtyřech řádcích kódu je pak zpracován a zformátován čas, kdy bylo měření provedeno.

Do proměnné title je uložena rychlost stahování a rychlost nahrávání. Do proměnné snippet je pak uložena rychlost odezvy, typ mobilní datové technologie a čas, kdy bylo měření provedeno.

Pomocí metody addMarker je pak na mapu přidán bod (marker), který je následně zobrazen na mapě. Do této metody jsou předány výše uložené proměnné marker, title a snippet.

V případě, že je zpracováváno poslední měření, je v závěru cyklu nastaveno zobrazení mapy na GPS souřadnice tohoto posledního měření. V závěru cyklu je rovněž nastaveno výchozí přiblížení mapy.

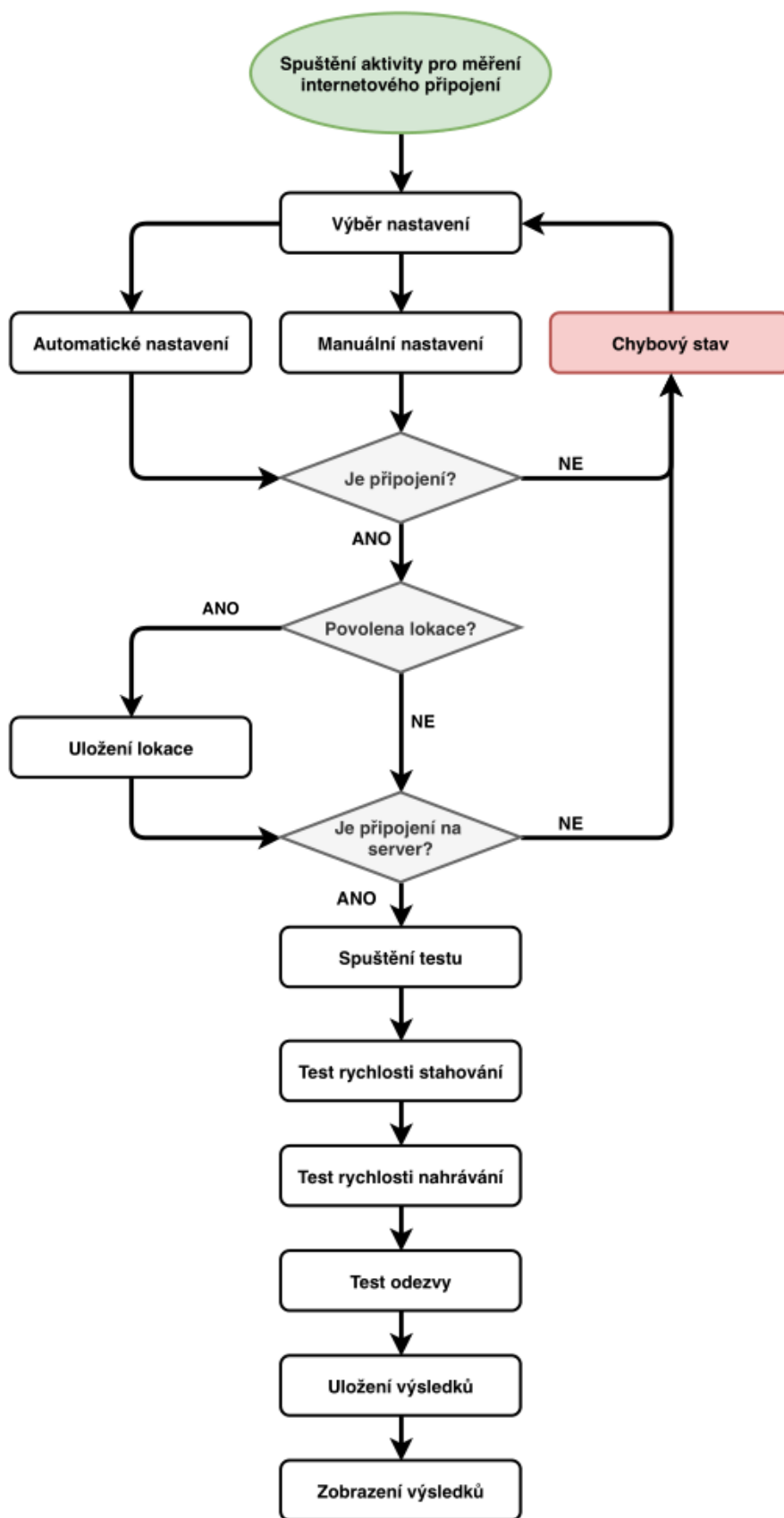
## **6.6 Algoritmy pro měření rychlosti nahrávání a rychlosti stahování**

V následující kapitole se již zaměříme přímo na implementaci algoritmů pro měření rychlosti nahrávání a rychlosti stahování.

Před spuštěním samotného testu měření parametrů internetového připojení je nutné zvolit nastavení. Ve výchozím stavu je nastavena automatická detekce parametrů pro test – tzn. algoritmus automaticky detekuje typ aktuálně využívané mobilní datové technologie a zvolí vhodné velikosti testovacích souborů. K tomuto účelu je napsána pomocná třída `NetInfo`, která využívá rozhraní `NetworkInfo` pomocí kterého je typ aktuálně použité mobilní datové technologie detekován. Je možné také automatické nastavení deaktivovat a zvolit vlastní velikost testovacího souboru. Dále je možné zvolit, zda se bude provádět kompletní test internetového připojení (rychlost stahování + rychlost nahrávání + odezva), nebo zda zvolíme jeden z parametrů. Po dokončení výběru nastavení je spuštěn test.

V první fázi samotného testu je vytvořeno připojení na SFTP server, odkud se budou testovací soubory stahovat, resp. nahrávat. K tomu účelu je napsána pomocná třída `SFTPClient`, která má spojení s SFTP serverem na starost. Třída má na starosti následující činnosti.

- Vytvoření spojení na SFTP server.
- Stahování souborů z SFTP serveru.
- Nahrávání souborů na SFTP server.
- Ukončení spojení.



Obrázek 17 - Diagram průběhu testu

Diagram uvedený na předchozí straně zobrazuje základní logiku algoritmu pro měření rychlosti stahování, rychlosti nahrávání a měření odezvy. V diagramu je zahrnut kompletní test parametrů internetového připojení.

Měření probíhá tak, že je z SFTP serveru stažen soubor, resp. nahrán soubor o určité velikosti, přičemž se měří čas přenosu souboru. Ke stažení souboru je k dispozici metoda `download`, která má dva parametry – cestu ke zdrojovému souboru a cestu k cílovému souboru (tj. kam má být soubor stažen). K nahrání souboru je k dispozici metoda `upload`, která má rovněž dva parametry – cestu ke zdrojovému souboru a cestu, kam má být soubor nahrán.

Před a po ukončení přenosu je uložen systémový čas pomocí metody `System.currentTimeMillis`. Metoda `System.currentTimeMillis` vrací aktuální hodnotu systémového času v milisekundách (datový typ `Long`). Z těchto dvou času je vypočtena doba přenosu, přičemž čas připojení a odpojení z SFTP serveru se do této doby nezapočítává – tzn. jedná se o čas, který byl třeba pouze k vlastnímu přenosu dat. Vzhledem k tomu, že známe velikost testovacího souboru a dobu, za kterou byl soubor přenesen je možné vypočítat rychlost stahování, resp. rychlost nahrávání pomocí následujícího vztahu:  $\text{rychlost} = \text{velikost souboru} / \text{čas přenosu souboru}$ .

V případě automatického nastavení je pak ze serveru stažen malý testovací soubor o velikosti 200 kB, který má za cíl detekovat možné zpomalení internetového připojení prostřednictvím FUP. Pokud stahování tohoto souboru trvá déle než 10 sekund, je vyhodnoceno potenciální možné zpomalení prostřednictvím FUP, jelikož rychlost stahování je menší než 20 kbps. V takovém případě, je test rychlosti stahování ukončen a větší testovací soubor již není vzhledem k malé rychlosti připojení stahován (především z časových důvodů).

Měření jednotlivých parametrů (rychlosti stahování, rychlosti nahrávání a odezvy) probíhá ve třídách `DownloadFileTask`, `UploadFileTask` a `PingTask`.

Třídy `DownloadFileTask` a `UploadFileTask` rozšiřují třídu `AsyncTask` a přepisují její metodu `doInBackground`.

```

public class DownloadFileTask extends AsyncTask<Void, Void, Void> {

    private SFTPClient sftpClient;
    private int fileSize;
    private long time;

    public DownloadFileTask(int fileSize) {
        this.fileSize = fileSize;
    }

    @Override
    protected Void doInBackground(Void... voids) {
        try {
            sftpClient = new SFTPClient( host: "XXX", port: 123,
                user: "XXX", password: "XXX");
            sftpClient.connect();

            System.out.println("Download: SFTP Connection is established.");
            long time1 = System.currentTimeMillis();
            sftpClient.download( source: "d/" + fileSize + ".dat",
                destination: "/data/data/com.example.diplom/cache/test.dat");
            long time2 = System.currentTimeMillis();
            time = time2 - time1;

            sftpClient.disconnect();
        } catch (JSchException | SftpException ex) {
            ex.printStackTrace();
            System.out.println("Download: SFTP Connection is not established.");
        }

        return null;
    }

    public long getTimeOfDownload() {
        return time;
    }

    public int getFileSize() {
        return fileSize;
    }
}

```

### Ukázka kódu 9 - Třída DownloadFileTask

Ukázka kódu č. 9 zobrazuje kód třídy DownloadFileTask. Tato třída má tři privátní proměnné – sftpClient typu SFTPClient, fileSize typu Integer a time typu Long. Třída dále obsahuje konstruktor, ve kterém je inicializována proměnná fileSize, do které je uložena hodnota, která odpovídá velikosti testovacího souboru v MB.



Třída dále obsahuje přepsanou metodu `doInBackground`, ve které probíhá samotný přenos souboru a následné vyhodnocení rychlosti. Nejprve je inicializována proměnná `sftpClient` (datového typu `SFTPClient`). Následně je pomocí metody `connect` vytvořeno připojení k SFTP serveru. Poté je zaznamenán do proměnné `time1` systémový čas a je spuštěn přenos testovacího souboru, po kterém je do proměnné `time2` uložen systémový čas. Na závěr je vypočten čas samotného přenosu (rozdílem dvou uložených časů) a pomocí metody `disconnect` dochází k ukončení spojení s SFTP serverem.

Třída dále obsahuje metodu `getTimeOfDownload`, která vrací čas přenosu souboru v milisekundách a metodu `getFileSize`, která vrací velikost testovacího souboru v MB. Obě tyto metody jsou využity pro výpočet rychlosti stahování.

```
DownloadFileTask d = new DownloadFileTask(fileSizeMB);
d.execute().get();

double time = d.getTimeOfDownload() / 1000.0;
int fileSize = d.getFileSize();
double speed = (fileSize / time) * 8;

DecimalFormat df = new DecimalFormat( pattern: "#0.00");
double speedFormatted = Double.valueOf(df.format(speed));
```

### Ukázka kódu 10 - Využití třídy `DownloadFileTask`

Využití třídy `DownloadFileTask` pro měření rychlosti stahování demonstruje ukázka kódu č. 10. Nejprve je vytvořen objekt třídy `DownloadFileTask`, do jehož konstruktoru zadáme velikost testovacího souboru v MB. Poté je zavolána metoda `execute`, která zavolá metodu `doInBackground` (popsáno na předchozí straně). Pomocí metody `get` se čeká až bude metoda `doInBackground` vykonání (tj. až bude dokončen přenos testovacího souboru). Následně je vypočtena rychlost přenosu, která je pomocí třídy `DecimalFormat` zaokrouhlena na dvě desetinná místa.

Třída `UploadFileTask` a algoritmus měření rychlosti stahování je koncipován na stejné logice kódu. Jediným rozdílem je, že přenos souboru probíhá ze serveru do zařízení. Místo metody `getTimeOfDownload` (ve třídě `DownloadFileTask`) je k dispozici metoda `getTimeOfUpload` (ve třídě `UploadFileTask`).

V případě chyby v průběhu měření rychlosti stahování, respektive rychlosti nahrávání, vrací metoda `getTimeOfDownload`, resp. `getTimeOfUpload` hodnotu -1.

## 6.7 Algoritmus na měření odezvy internetového připojení

Pro měření odezvy internetového připojení je určena třída PingTask, která rozšiřuje třídu AsyncTask. Vzhledem k faktu, že měření odezvy je síťová operace, která může být časově náročná, je nutné ji provádět mimo hlavní vlákno.

```
public class PingTask extends AsyncTask<Void, Void, Void> {

    private String host;
    private int port;
    private int count;
    private long[] times;

    public PingTask(String host, int port, int count) {
        this.host = host;
        this.port = port;
        this.count = count;
        this.times = new long[count];
    }

    public long getPing() {
        long sum = 0;

        for (int i = 0; i < count ; i++) {
            sum = sum + times[i];
        }

        return sum / count;
    }

    @Override
    protected Void doInBackground(Void... voids) {
        try {
            SocketAddress sockaddr = new InetSocketAddress(host, port);
            Socket sock = new Socket();

            for (int i = 0; i < count; i++) {
                long time1 = System.currentTimeMillis();
                sock.connect(sockaddr, timeout: 5000);
                long time2 = System.currentTimeMillis();
                long time = time2 - time1;
                times[i] = time;
            }
        } catch (IOException e) {
            e.printStackTrace();
        }
        return null;
    }
}
```

Ukázka kódu 11 - Třída PingTask

Třída PingTask, která je zobrazena na předchozí straně – ukázka kódu č. 11, obsahuje celkem 4 privátní proměnné.

- Proměnná host datového typu String, která slouží k uložení adresy serveru, vůči kterému se bude odezva měřit.
- Proměnná port datového typu Integer, která slouží k uložení portu serveru.
- Proměnná count datového typu Integer, do které je uložen, kolikrát se má měření provést.
- Proměnná times datového typu Long[] (pole hodnot), do které jsou ukládány jednotlivé výsledky měření.

Třída dále obsahuje konstruktor, který inicializuje všechny privátní proměnné.

Metoda getPing se stará o výpočet průměrné doby odezvy internetového připojení, kterou vrací.

V metodě doInBackground probíhá samotné měření odezvy internetového připojení. Nejprve jsou vytvořeny proměnné sockaddr (objekt SocketAddress) a sock (objekt Socket). Poté následuje cyklus, který provede jednotlivá měření, přičemž celkový počet měření je uložen v proměnné count. Do proměnné time1 je uložen systémový čas. Pomocí metody connect je vytvořeno připojení k danému serveru s limitním časem 5 000 milisekund. V případě, že by připojení k serveru trvalo déle než 5 000 milisekund je vyvolána výjimka IOException. V případě, že by připojení trvalo déle, než zmiňovaných 5 000 milisekund je kvalita internetového připojení velice špatná a nemá smysl pokračovat v měření.

Po úspěšném připojení k serveru je do proměnné time2 uložen systémový čas. Poté je vypočtena doba odezvy internetového připojení rozdílem dvou uložených časů (time2 – time1) a tato hodnota je uložena do pole times.

```
PingTask pt = new PingTask(host, port, count);  
pt.execute().get();  
long ping = pt.getPing();
```

### Ukázka kódu 12 - Použití třídy PingTask

Ukázka kódu č. 12 ukazuje použití třídy PingTask pro měření odezvy internetového připojení. Nejprve je vytvořena proměnná typu PingTask, následně je zavolána

metoda `get`, která ve třídě volá metodu `doInBackground`. Na závěr je vrácena doba odezvy internetového připojení v milisekundách pomocí metody `getPing`. V případě, že při měření rychlosti odezvy nastala chyba (např. nedostupný server) vrací metoda `getPing` hodnotu `-1`.

## 7 Měření a výsledky

### 7.1 Vybrané lokality pro měření

Pro měření parametrů mobilního internetového připojení bylo vybráno celkem 8 lokalit na území Královehradeckého a Pardubického kraje. Kromě těchto lokalit jsem vybral také 3 lokality na území hlavní města Prahy, kde by mělo být využití mobilních datových technologií největší z důvodu potencionálně většího počtu připojených zařízení v porovnání s ostatními vybranými lokalitami.

V každé z vybraných lokalit bylo cílem provést měření alespoň dvou mobilních datových technologií pod dvěma mobilními operátory. V tomto měření jsem se rozhodl provést jednotlivé měření pod mobilním operátorem T-Mobile a Vodafone. Vždy byly vybrány alespoň dvě mobilní datové technologie, které jsou v dané lokalitě dostupné pro oba mobilní operátory.

Tabulka uvedená níže (a na další straně) zobrazuje vybrané lokality, kde byly mobilní datové technologie testovány. V tabulce je současně uvedeny GPS souřadnice, kde byla jednotlivá měření prováděna.

**Tabulka 6 – Vybrané lokality pro měření parametrů mobilního internetového připojení**

	<b>Lokalita</b>	<b>Měřené technologie</b>	<b>GPS souřadnice*</b>
<b>1</b>	<b>Holice</b> , náměstí TGM	EDGE, HSPA+, LTE	50.067416, 15.986847
<b>2</b>	<b>Holice</b> , Tyršova	EDGE, HSPA+, LTE	50.072952, 15.991339
<b>3</b>	<b>Pardubice</b> , letiště terminál	EDGE, LTE	50.0177208, 15.7253317
<b>4</b>	<b>Pardubice</b> , budova hlavní nádraží	EDGE, LTE	50.032410, 15.756090
<b>5</b>	<b>Pardubice</b> , Pernštýnské náměstí	EDGE, HSPA+, LTE	50.038578, 15.779091

6	Hradec Králové, Velké náměstí	EDGE, HSPA+, LTE	50.209605, 15.833577
7	Říčky v Orlických horách	EDGE, UMTS, LTE	50.226248, 16.460373

\* Přibližná GPS pozice, kde bylo měření prováděno.

## 7.2 Metodika měření

### 7.2.1 Mobilní telefon Xiaomi Redmi 4A LTE

Pro měření parametrů internetového připojení ve vybraných lokalitách jsem využil mobilní telefon Xiaomi Redmi 4A LTE. Tento mobilní telefon umožňuje využívat následující mobilní datové technologie: GPRS, EDGE, UMTS/CDMA, HSPA, HSPA+ a LTE. Mobilní telefon disponuje operačním systémem Android ve verzi 6.0.1.

### 7.2.2 Počet provedených měření

Vzhledem k omezení FUP, které mobilní operátoři aplikují bylo v každé vybrané lokalitě cílem provést 5 měření pro technologie které spadají do třetí a čtvrté generace, pro sítě druhé generace pak byly provedeny 3 měření pro každou vybranou lokalitu.

Při rozhodování, kolik provést měření jsem vycházel z předpokladu, že drtivá většina zařízení je připojena a využívá sítě třetí a čtvrté generace (vzhledem k dostupnosti ve vybraných lokalitách). U sítí třetí a čtvrté generace může vzhledem k vysokému počtu připojených zařízení a nerovnoměrnosti využití může docházet k relativně velkému kolísání jednotlivých měřených parametrů (zejména pak rychlost stahování, rychlost nahrávání). U sítí druhé generace, vzhledem k relativně malým rychlostem (řádově 100–200 kbps) nemá význam provádět velké množství měření.

### 7.2.3 Metodika měření

V každé z vybraných lokalit bylo cílem provést měření jednotlivých datových mobilních technologií s určitým odstupem mezi jednotlivými měření -tj. neprovést všechna měření jedné vybrané mobilní datové technologie bezprostředně za sebou. Pokud jsou měření pro vybranou datovou technologii prováděna s určitým

odstupem mezi jednotlivými měření, lze získat větší objektivnost měření – v intervalu např. 10 minut se může využití mobilní sítě (mobilní datové technologie) výrazně změnit.

Jednotlivé kroky č. 1 až č. 7 uvedené níže znázorňují použitou metodiku, pomocí které bylo měření mobilních datových technologií prováděno pro jednu danou lokalitu. S využitím této metodiky je zajištěno, že jednotlivá měření pro danou vybranou mobilní datovou technologii budou provedena s určitým rozestupem mezi jednotlivými měřeními. Takto by měla být zajištěna objektivnost měření, zejména pak s porovnáním metodiky, kdy by byla všechna měření pro danou mobilní datovou technologii provedena bezprostředně za sebou.

- 1) Vybrání a otestování první mobilní datové technologie (např. LTE) pod prvním mobilním operátorem.
- 2) Vybrání a otestování druhé mobilní (např. EDGE) datové technologie pod prvním mobilním operátorem.
- 3) Vybrání a otestování další mobilní datové technologie pod prvním mobilním operátorem.
- 4) Otestování první vybrané mobilní datové technologie (např. LTE) pod druhým mobilním operátorem – je-li v dané lokalitě k dispozici.
- 5) Otestování druhé mobilní datové technologie pod druhým mobilním operátorem – je-li v dané lokalitě k dispozici.
- 6) Otestování další mobilní datové technologie pod druhým mobilním operátorem – je-li v dané lokalitě k dispozici.
- 7) Skok na krok č. 1 a opakování kroků 1 až 6, dokud není proveden cílený počet měření jednotlivých mobilních datových technologií (3 měření pro síť druhé generace, 5 měření pro síť třetí a čtvrté generace).

### **7.3 Analýza provedených měření**

V celkem 7 lokalitách bylo v průběhu 37 dnů provedeno celkem 142 měření parametrů mobilního internetového připojení.

Tabulka uvedená níže zobrazuje, jaké testovací soubory byly použity pro měření parametrů mobilního internetového připojení v rámci jednotlivých datových

technologií. Pro test rychlosti stahování a rychlosti nahrávání byl použit soubor o totožné velikosti. Vycházel z předpokladu, že u mobilních datových technologií se rychlost stahování a rychlost nahrávání příliš neliší (např. jako tomu může být u VDSL – rychlost nahrávání 50 Mbps, rychlost stahování 5 Mbps).

<b>Mobilní datová technologie</b>	<b>Velikost testovacího souboru</b>
EDGE	0,2 MB / 1 MB
UMTS	2 MB
HSPA+	4 MB
LTE	8 MB

**Tabulka 7 – Přehled testovaných technologií a velikostí použitých testovacích souborů**

### 7.3.1 Souhrnný pohled na provedená měření

Tabulka uvedená níže zobrazuje průměrné hodnoty rychlosti stahování, rychlosti nahrávání a odezvy mobilního internetového připojení dle technologie. Tabulka má demonstrovat prakticky naměřené hodnoty mobilního internetového připojení. Můžeme zde vidět například, že mobilní datová technologie LTE průměrně dosahuje v praxi téměř dvojnásobných rychlosti v porovnání s mobilní datovou technologií HSPA+.

**Tabulka 8 – Průměrné naměřené hodnoty ze všech testů v rámci jednotlivých mobilních datových technologií**

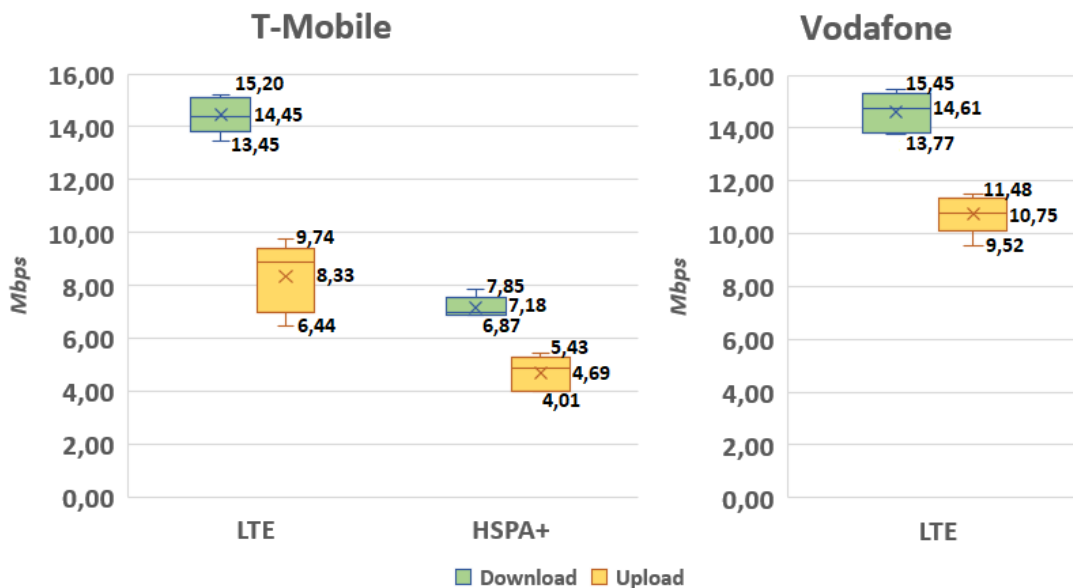
<b>Mobilní datová technologie</b>	<b>Rychlost stahování</b>	<b>Rychlost nahrávání</b>	<b>Odezva</b>
EDGE	0,14 Mbps	0,11 Mbps	256 ms
UMTS	1,97 Mbps	1,13 Mbps	52 ms
HSPA+	7,97 Mbps	5,21 Mbps	35 ms
LTE	14,45 Mbps	9,33 Mbps	27 ms



### 7.3.2 Zhodnocení mobilního internetu dle lokality – Holice

Holice je menší město v Pardubickém kraji, kde žije přibližně 8 tisíc obyvatel. V Holicích byly vybrány na měření parametrů mobilního internetového připojení dvě lokality: Náměstí TGM, jež je centrem města, a ulice Tyršova, kde bydlím. Vzdálenost mezi těmito dvěma lokalitami je přibližně 900 metrů.

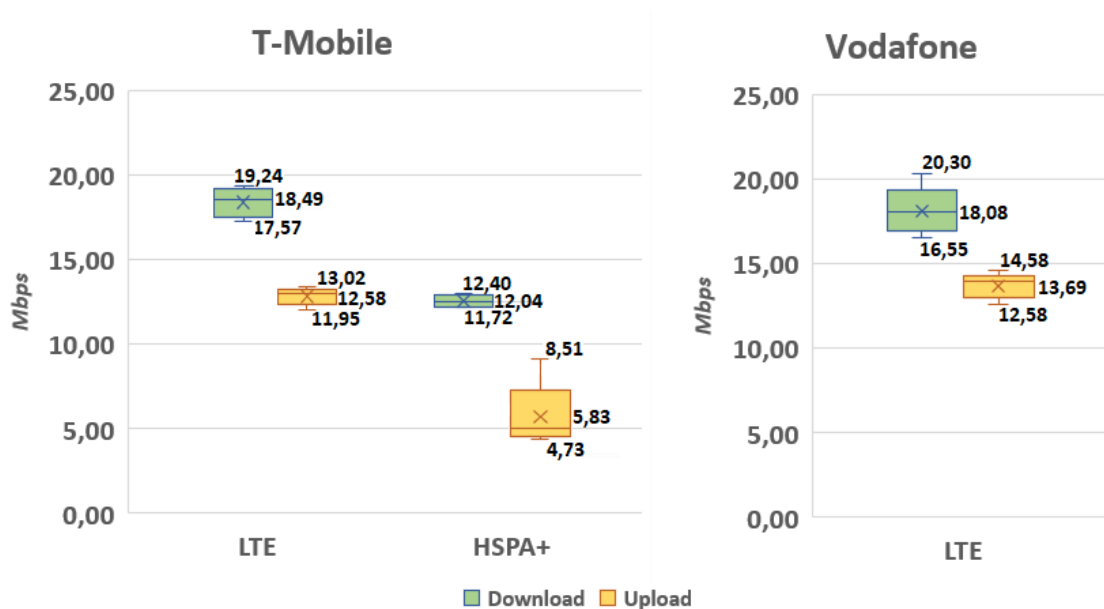
Graf uvedený níže zobrazuje naměřené hodnoty pro první lokalitu v Holicích – Náměstí TGM. Naměřené hodnoty rychlosti stahování se v případě mobilní datové technologie LTE mezi operátory téměř neliší. Naměřená rychlost stahování je v průměru 14,53 Mbps. Nejlepší a nejhorší naměřené hodnoty jsou pak vyznačeny v grafu. Naměřené rychlosti nahrávání jsou pak lepší v případě mobilního operátora Vodafone – průměrná rychlost stahování 10,75 Mbps, v porovnání s mobilním operátorem T-Mobile, který dosáhl průměrné rychlosti 8,33 Mbps. Mobilní datová technologie HSPA+ pak nabídla průměrnou rychlost stahování 7,18 Mbps a průměrnou rychlost nahrávání 4,69 Mbps.



**Obrázek 18 – Graf zobrazující naměřené hodnoty v lokalitě č. 1, Holice, Náměstí TGM**

Odezva internetového připojení byla pro datovou technologii LTE pro oba mobilní operátory shodná. Průměrná doba odezvy pro technologii LTE byla 29 milisekund. Průměrná doba odezvy pro mobilní datovou technologii HSPA+ (mobilní operátor T-Mobile) pak byla 33 milisekund. Průměrná doba odezvy pro mobilní datovou

technologie EDGE byla v případě mobilního operátora T-Mobile byla 221 milisekund a 194 v případě mobilního operátora Vodafone.



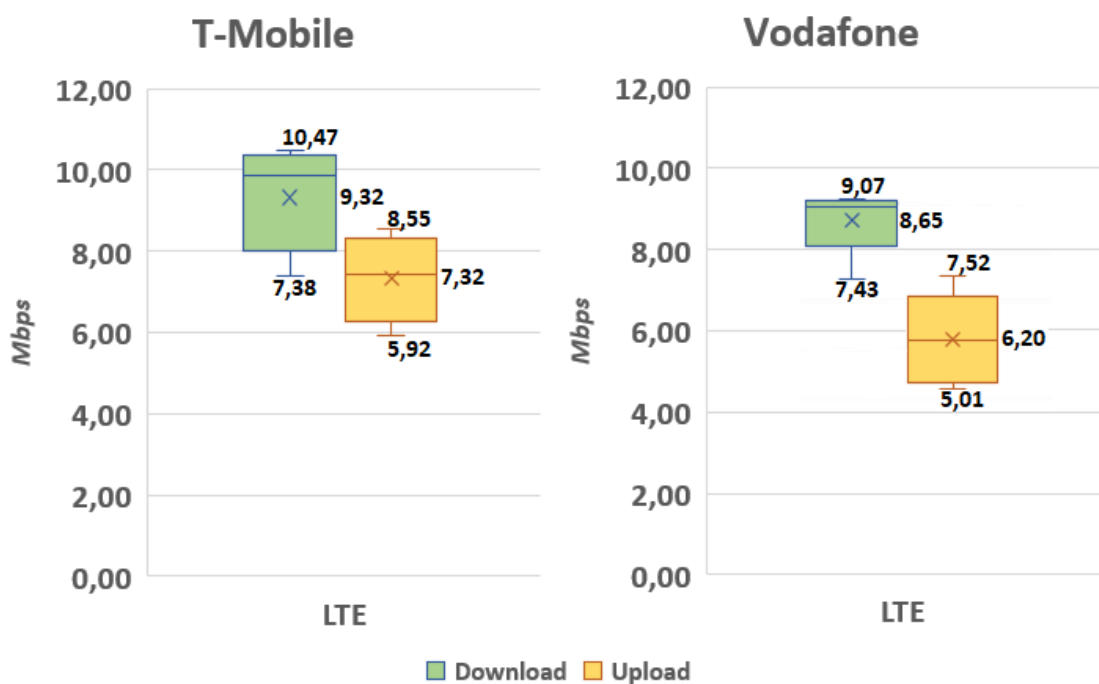
**Obrázek 19 – Graf zobrazující naměřené hodnoty v lokalitě č. 2, Holice, Tyršova**

Graf uvedený výše zobrazuje naměřené hodnoty v druhé lokalitě v Holicích – ulice Tyršova. Naměřená data jsou ve všech případech lepší v porovnání v první lokalitou Holice – Náměstí TGM. Průměrná rychlost stahování pro mobilní datovou technologii LTE byla v případě mobilního operátora O2 18,49 Mbps, v případě Vodafone pak 18,08 Mbps. Průměrná rychlost stahování pak byla v případě O2 12,58 Mbps a v případě Vodafone 13,69 Mbps. Opět zde platí, že naměřené hodnoty mezi jednotlivými operátory se příliš nelišily.

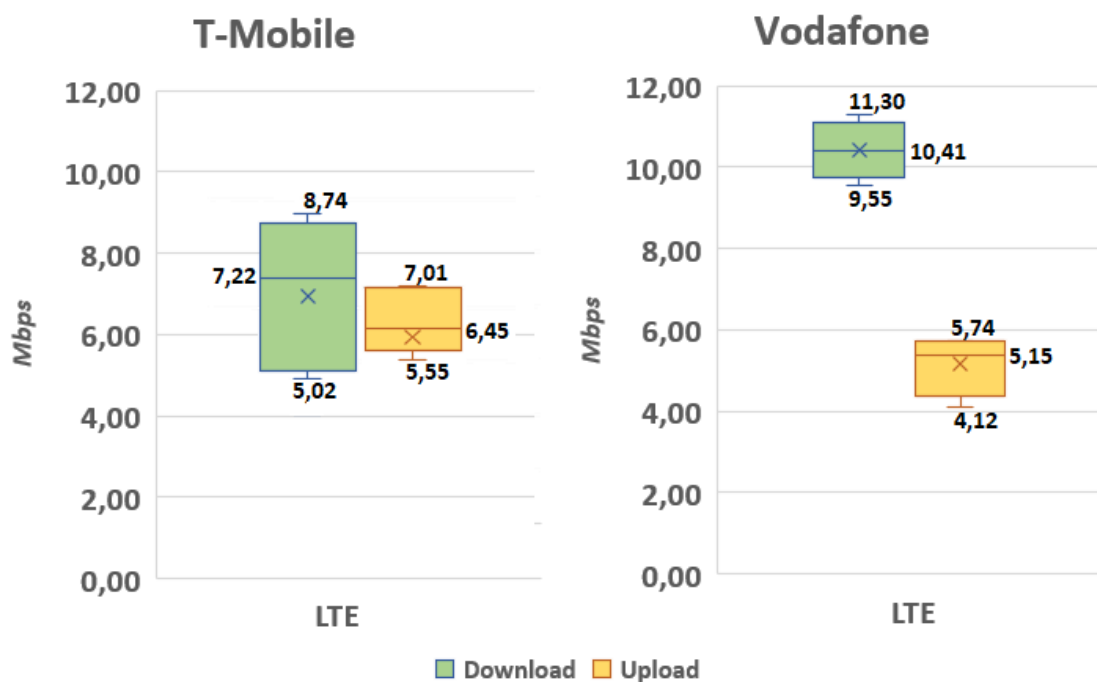
Mobilní datová technologie EDGE spadající do druhé generace pak nabídla průměrnou rychlost stahování 0,15 Mbps a průměrnou rychlost nahrávání 0,10 Mbps. Mezi operátory nebyl rozdíl v naměřených hodnotách. Vzhledem k nízké rychlosti této technologie nemá význam naměřené hodnoty dále vyhodnocovat.

### 7.3.3 Zhodnocení mobilního internetu dle lokality – Pardubice

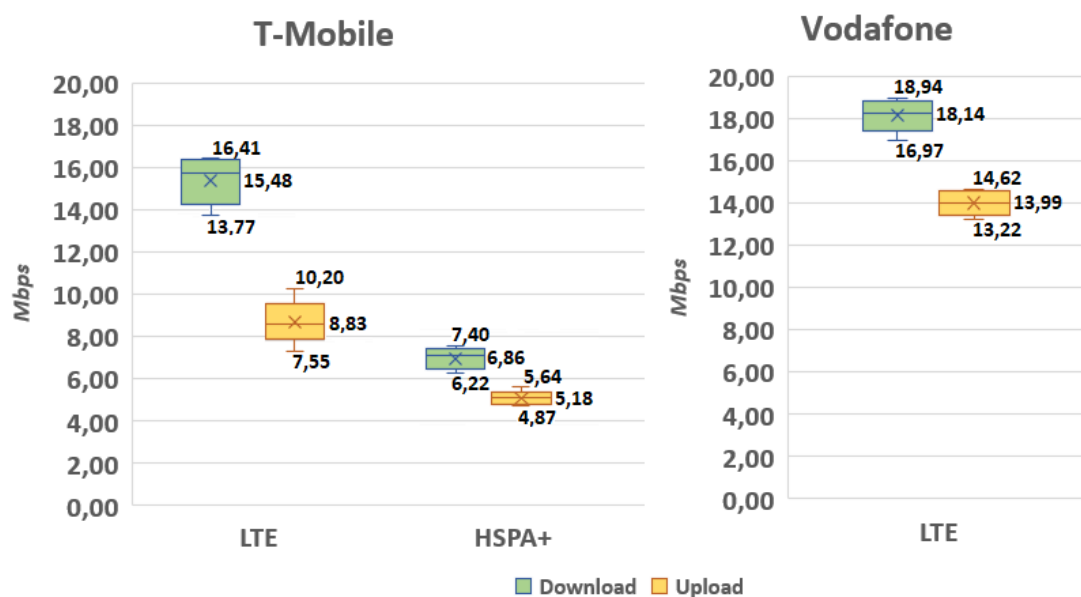
V krajském městě Pardubice byly vybrány celkem tři lokality pro měření parametrů mobilního internetového připojení – hlavní vlakové nádraží, Pernštýnské náměstí (centrum města) a nový letištní terminál na okraji města.



Obrázek 20 - Graf zobrazující naměřené hodnoty v lokalitě č. 3, Pardubice, letištní terminál



Obrázek 21 - Graf zobrazující naměřené hodnoty v lokalitě č. 4, Pardubice, hlavní nádraží



Obrázek 22 - Graf zobrazující naměřené hodnoty v lokalitě č. 5, Pardubice, Pernštýnské náměstí

Tři grafy uvedené výše a na předchozí straně zobrazují rozsahy naměřených hodnot, včetně minimální a maximální hodnot po jednotlivé měření mobilní datové technologie.

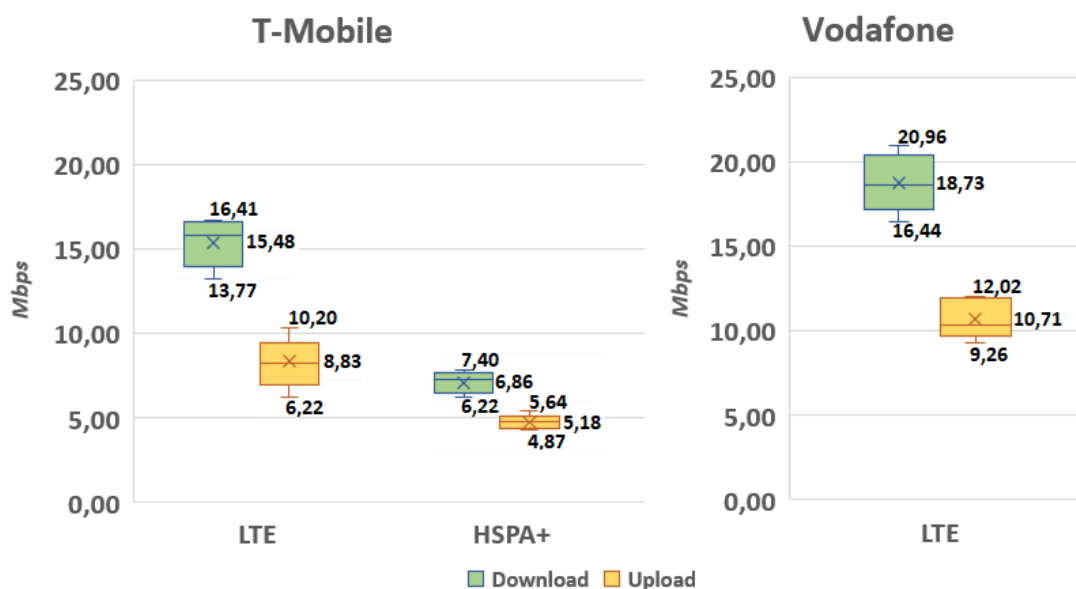
Mobilní datová technologie LTE dosáhla nejlepších výsledků v lokalitě Pernštýnské náměstí, kde byla naměřená průměrná rychlost stahování 15,48 Mbps, rychlost nahrávání 8,83 Mbps pro mobilního operátora T-Mobile, respektive 18,18 Mbps a 13,99 Mbps pro mobilního operátora Vodafone. Ve zbylých dvou lokalitách byly naměřeny rychlosti řádově o několik jednotek menší. Průměrná odezva mobilní datové technologie LTE se pak pohybovala v rozmezí 20–30 milisekund a nedocházelo zde k výkyvům naměřených hodnot.

Průměrná rychlost stahování pro mobilní datovou technologii HSPA+ (operátor T-Mobile) byla v centru města 6,86 Mbps a rychlost nahrávání pak 5,18. Průměrná odezva pak bylo 38 milisekund.

Rychlost mobilní datové technologie EDGE byla ve všech třech lokalitách obdobná, průměrná rychlost stahování byla 0,14 Mbps, průměrná rychlost nahrávání 0,11 Mbps a průměrná naměřená odezva 260 milisekund.

#### 7.3.4 Zhodnocení mobilního internetu dle lokality – Hradec Králové

V dalším krajském městě – Hradec Králové byly parametry mobilního internetového připojení měřeny v lokalitě Velké náměstí, jedná se o střed města.



Obrázek 23 - Graf zobrazující naměřené hodnoty v lokalitě č. 6, Hradec Králové, Velké náměstí

Mobilní operátor Vodafone v případě této lokality nabídl přibližně o 3 Mbps lepší výsledky v porovnání s mobilním operátorem T-Mobile. Průměrné rychlosti byly 15,48 Mbps pro rychlost stahování a 8,83 Mbps pro rychlost nahrávání v případě mobilního operátora T-Mobile, respektive 18,73 Mbps a 10,71 Mbps v případě mobilního operátora Vodafone. Průměrná doba odezvy pak byla 30 milisekund a naměřené hodnoty se mezi operátory výrazněji nelišily.

V rámci mobilního operátora T-Mobile byla testována také mobilní datová technologie HSPA+. Průměrné rychlosti této technologie byly 6,86 Mbps pro rychlost stahování a 5,18 Mbps pro rychlost nahrávání, Průměrná naměřená odezva pak byla 45 milisekund.

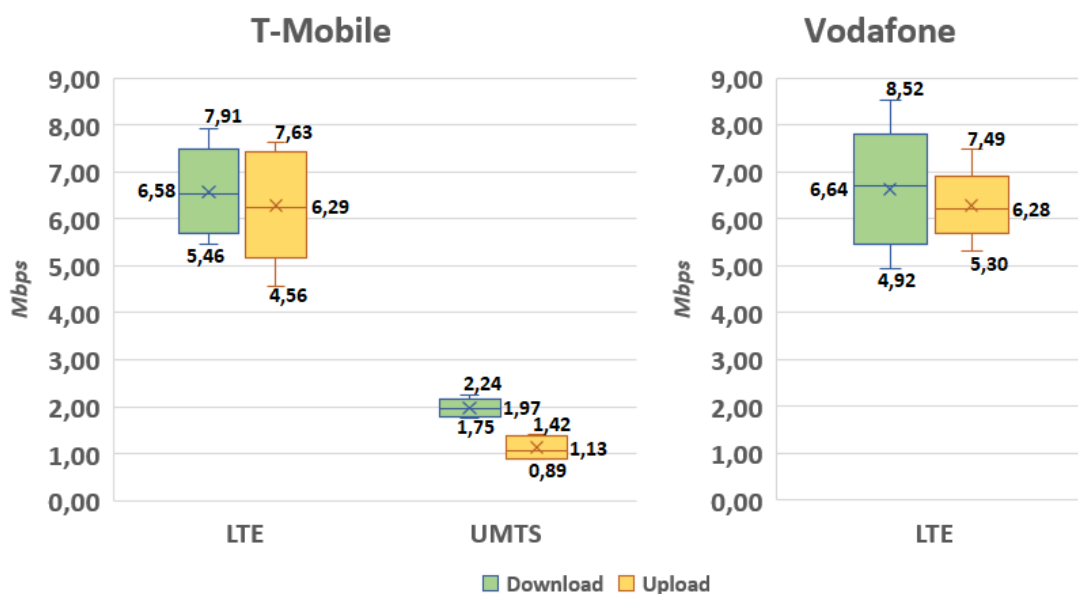
V případě mobilní technologie EDGE, podobně jako v předchozích případech, byla průměrná rychlost stahování a nahrávání 0,11 Mbps, průměrná rychlost odezvy pak byla 260 milisekund.

### 7.3.5 Zhodnocení mobilního internetu dle lokality – Říčky v Orlických horách

Jako poslední lokalita pro měření parametrů mobilního internetového připojení byla vybrána malá obec v Orlických horách – Říčky v Orlických horách. Graf uvedený níže zobrazuje naměřené hodnoty v této lokalitě.

Naměřené hodnoty mobilní datové technologie LTE byly v této lokalitě téměř shodné pro oba mobilní operátory. Průměrná rychlost stahování činila 6,61 Mbps, průměrná rychlost nahrávání pak 6,29 Mbps. Průměrná doba odezvy byla 45 milisekund, což je horší výsledek v porovnání s ostatními lokalitami, kde bylo měření mobilní datové technologie LTE prováděno.

Vzhledem k nedostupnosti mobilní datové technologie HSPA+, byla měřena pod mobilním operátorem T-Mobile mobilní datová technologie UMTS. Průměrná rychlost stahování této technologie byla 1,97 Mbps a průměrná rychlost nahrávání pak 1,13 Mbps. Průměrná odezva byla 52 milisekund.



Obrázek 24 - Graf zobrazující naměřené hodnoty v lokalitě č. 7, Říčky v Orlických horách

Průměrná rychlost stahování pro mobilní datovou technologii EDGE byla 0,15 Mbps, respektive 0,12 Mbps pro rychlost nahrávání. Průměrná odezva byla 378

milisekund, což je o více než 100 milisekund horší v porovnání s ostatními lokalitami, kde byla mobilní datová technologie EDGE měřena.

#### **7.4 Rychlost mobilního internetu v době epidemie a před**

Vzhledem k pandemii koronového viru, které propukla v březnu 2020, jsem nemohl provést všechna plánovaná měření v plánovaných lokalitách. Rozhodl jsem se proto ve vybraných lokalitách provést další měření v době pandemie koronového viru a tyto výsledky porovnat s běžnou situací před vypuknutím pandemie. Zaměřil jsem se přitom na porovnání mobilní datové technologie LTE.

V době pandemie koronového viru se čeští mobilní operátoři rozhodli za určitých podmínek poskytnout mobilní data zdarma (v některých případech zcela s neomezeným FUP). Domníval jsem se, že právě tato akce by mohla mít výraznější vliv na výslednou rychlost mobilního internetového připojení z důvodu vyššího možného využití z důvodu poskytnutých dat zdarma.

K tomuto testu byly vybrány následující lokality:

Lokalita 1 – Holic, náměstí TGM,

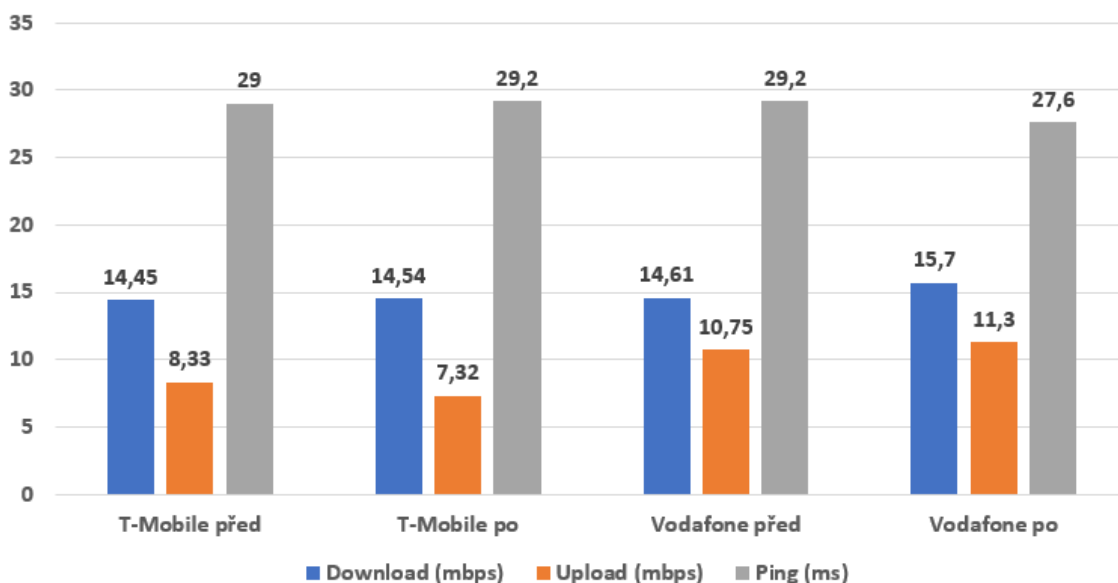
Lokalita 2 – Holic, Tyršova,

Lokalita 3 – Pardubice, Pernštýnské náměstí.

Graf uvedený na následující straně zobrazuje průměrné naměřené hodnoty v lokalitě č. 1 (Holic, náměstí TGM) pro rychlost stahování, rychlost nahrávání a odezvu v době před a po vypuknutí epidemie koronového viru. Z grafu vyplývá, že nově naměřené hodnoty se příliš neliší s hodnotami, které byly naměřeny v době před

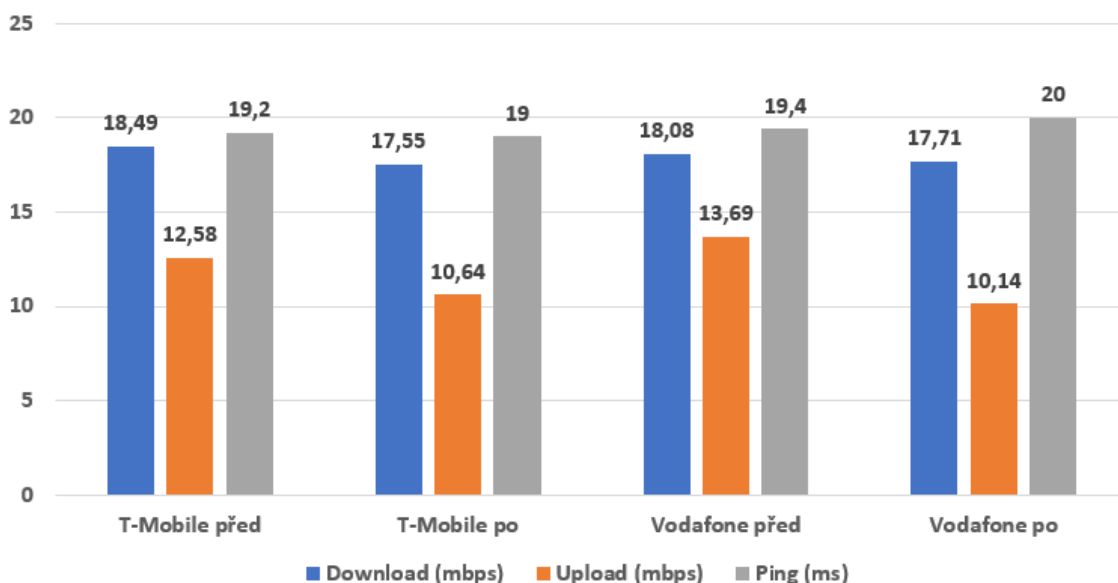


vypuknutí korona viru. V případě mobilního operátora Vodafone byly v době epidemie naměřeny dokonce mírně lepší hodnoty ve všech měřených parametrech.



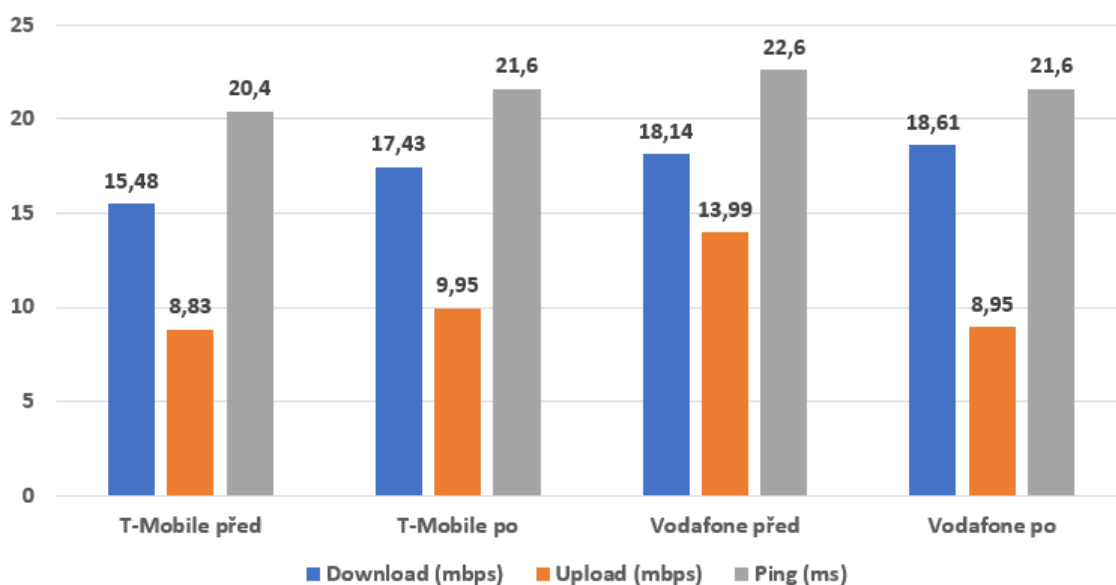
**Obrázek 25 - Graf naměřených hodnot pro lokalitu č. 1.**

Z grafu uvedeného níže vyplývá, že v lokalitě č. 2 (Holice, Tyršova) je již patrný mírný pokles v rychlosti stahování a rychlosti nahrávání v době před a po vypuknutí epidemie. Stále se však jedná pouze o mírný rozdíl řádově v jednotkách megabitů za sekundu, který není nikterak významný. Odezva internetového připojení je pak téměř na stejné úrovni.



**Obrázek 26 - Graf naměřených hodnot pro lokalitu č. 2.**

Graf zobrazený níže průměrné naměřené hodnoty pro krajské město Pardubice – lokalita č. 3. Z grafu vyplývá, že nově naměřené hodnoty v době epidemie jsou lepší v porovnání s naměřenými hodnotami před vypuknutím epidemie. Výjimku tvoří pouze rychlost stahování u mobilního operátora, která je v tomto případě výrazně nižší. Odezva internetového připojení je téměř na stejné úrovni.



**Obrázek 27 - Graf naměřených hodnot pro lokalitu č. 3.**

Výše provedená měření a grafy ukazují, že má domněnka byla mylná. Nově naměřené hodnoty v době epidemie se ve vybraných lokalitách příliš neliší s hodnotami, které byly naměřené v době před vypuknutím epidemie. Mobilní data zdarma, která poskytují mobilní operátoři tak nemají žádný zásadní vliv na rychlost mobilního internetového připojení.

## 8 Závěry a doporučení

V teoretické část diplomové byl proveden rozbor mobilních datových technologií, které se v současné době využívají na území České republiky. Dále byla provedena rešerše mobilních aplikací, které lze využít pro měření parametrů (mobilního) internetového připojení. Byly popsány rozdíly mezi jednotlivými aplikacemi.

Praktická část diplomové práce se zabývá analýzou a následně vytvoření mobilní aplikace po měření parametrů mobilního internetového připojení. Aplikace je primárně učena pro mobilní zařízení s operačním systémem Android. Mobilní aplikace umožňuje měřit tři základní parametry mobilního internetového připojení – rychlost stahování, rychlost nahrávání a odezvu. Výsledky jednotlivých měření jsou ukládány do databáze a mohou být později zobrazeny.

Na závěr této práce bylo provedeno testování kvality mobilního internetového připojení v několika lokalitách s využitím vytvořené aplikace. Výsledky byly následně vyhodnoceny a bylo provedeno porovnání a vyhodnocení kvality mobilního internetového připojení. Pro toto testování byli vybráni dva mobilní operátoři – T-Mobile a Vodafone.

V testování kvality mobilního internetového připojení se ukázalo, že ve všech měřených lokalitách byla dostupná mobilní datová technologie LTE. Rychlost tohoto připojení se v provedených měřeních ukázala jako dostatečná pro běžné využití internetu. Oproti tomu mobilní datová technologie EDGE se dle teoretických předpokladů ukázala jako nevhodná, až na několik výjimek, jakými jsou např. základní prohlížení internetu, zasílání textových zpráv apod.

Vzhledem k epidemii korona viru, kvůli které nebylo možné provést měření kvality internetového připojení ve všech plánovaných lokalitách, bylo ve třech vybraných lokalitách provedeno měření, které porovnálo rychlost mobilního internetu před a v průběhu epidemie. Předpokládal jsem, že v průběhu epidemie bude rychlost mobilního internetu nižší, z důvodu potencionálně vyššího využití. Toto tvrzení se však nepotvrdilo. V průběhu měření se ukázalo, že rychlost mobilního internetového připojení je na téměř totožné úrovni.

## 9 Seznam použitých zdrojů

- [1] UD DIN ARSHAD, Qazi Kamal, Ahsan Ullah KASHIF a Ijaz Mansoor QUERSHI. A Review on the Evolution of Cellular Technologies. *2019 16th International Bhurban Conference on Applied Sciences and Technology (IBCAST)* [online]. IEEE, 2019, 2019, 989-993 [cit. 2019-04-17]. DOI: 10.1109/IBCAST.2019.8667173. ISBN978-1-5386-7729-2. Dostupné z: <https://ieeexplore.ieee.org/document/8667173/>
- [2] Kalra, Bharti & Chauhan, D. (2014). A Comparative Study of Mobile Wireless Communication Network: 1G to 5G. *International Journal of Computer Science and Information Technology Research*. 2. 430-433. [cit. 2019-04-17]. Dostupné z: [https://www.researchgate.net/publication/318673817\\_A\\_Comparative\\_Study\\_of\\_Mobile\\_Wireless\\_Communication\\_Network\\_1G\\_to\\_5G](https://www.researchgate.net/publication/318673817_A_Comparative_Study_of_Mobile_Wireless_Communication_Network_1G_to_5G)
- [3] SINGH, Amritpal. A Review of Different Generations of Mobile Technology. *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)* [online]. 2015, August 2015, 2015(Volume 4 8), 3404-3408 [cit. 2019-04-17]. Dostupné z: <http://ijarcet.org/wp-content/uploads/IJARCET-VOL-4-ISSUE-8-3404-3408.pdf>
- [4] RU, Cha a Rajeev GUPTA. A Comparative Study of Various Generations in Mobile Technology. *International Journal of Engineering Trends and Technology* [online]. 2015, 28(7), 328-332 [cit. 2019-04-17]. DOI: 10.14445/22315381/IJETT-V28P263. ISSN 22315381. Dostupné z: <http://www.ijettjournal.org/archive/ijett-v28p263>
- [5] Technologie mobilního internetu – od CSD po LTE Advanced (vědecké okénko). Mobilizujeme.cz [online]. 2012, 12. února 2012 [cit. 2018-03-10]. Dostupné z: <https://mobilizujeme.cz/clanky/technologie-mobilniho-internetu-od-csd-po-lte-advanced-vedecke-okenko>
- [6] RYŠÁNEK, F. *Pásma LTE/UMTS/EDGE/GSM používaná v České republice* [online]. [cit. 2019-04-23]. Dostupné z: <http://www.fccps.cz/pasma-lteumtsedegsgsm-pouzivana-v-ceske-republice-1379>
- [7] BÍLÝ, Vladimír. *Frekvenční přiděl na pásmech GSM, DCS, UMTS a LTE v České republice* [online]. 2016 [cit. 2019-04-23]. Dostupné z: <https://www.gsmweb.cz/clanky/freq2.htm>

- [8] Mapa pokrytí - Vodafone.cz. Vodafone [online]. 2019 [cit. 2019-04-23]. Dostupné z: <https://www.vodafone.cz/mapa-pokryti/>
- [9] Co je to 5G internet? Vše, co potřebujete vědět o síti nové generace. *Svět Androida* [online]. 13. 12. 2018 [cit. 2019-04-23]. Dostupné z: <https://www.svetandroida.cz/5g-internet-site-rychlost/>
- [10] Speedtest Knowledge Base. *Speedtest Knowledge Base* [online]. 2019, 2019 [cit. 2019-12-20]. Dostupné z: <https://www.speedtest.net/about/knowledge>
- [11] OVERTURF, Carin. How does Speedtest measure my network speeds? SPEEDTEST [online]. 2019 [cit. 2019-12-11]. Dostupné z: [https://help.speedtest.net/hc/en-us/articles/360038679354-How-does-Speedtest-measure-my-network-speeds-?fbclid=IwAR0opNHLicKsVVKWGWcrplzzLj7yz90vSJEb0VC4LtO\\_HbyZve1cRUttK0c](https://help.speedtest.net/hc/en-us/articles/360038679354-How-does-Speedtest-measure-my-network-speeds-?fbclid=IwAR0opNHLicKsVVKWGWcrplzzLj7yz90vSJEb0VC4LtO_HbyZve1cRUttK0c)
- [12] RYCHLOST.CZ Metodika testování. RYCHLOST.CZ [online]. 2019, 2019 [cit. 2019-12-20]. Dostupné z: <https://rychlost.cz/informace/princip/>
- [13] DSL.CZ Měření rychlosti připojení k internetu. *DSL.CZ* [online]. [cit. 2019-12-20]. Dostupné z: <http://www.dsl.cz/mereni-rychlosti>
- [14] OVERTURF, Carin. *What do Ping, Jitter, Download, Upload, and packet loss mean and why are they important?* [online]. November 2019 [cit. 2019-12-20]. Dostupné z: <https://help.speedtest.net/hc/en-us/articles/360039161553-What-do-Ping-Jitter-Download-Upload-and-packet-loss-mean-and-why-are-they-important-?>
- [15] Technologie mobilního internetu – od CSD po LTE Advanced (vědecké okénko). Mobilizujeme.cz [online]. 2012, 12. února 2012 [cit. 2018-03-10]. Dostupné z: <https://mobilizujeme.cz/clanky/technologie-mobilniho-internetu-od-csd-po-lte-advanced-vedecke-okenko>
- [16] ZELENKA, Josef a Jiří KYSELA. Informační a komunikační technologie v cestovním ruchu. Vyd. 4., přeprac. a dopl. Hradec Králové: Gaudeamus, 2013. ISBN 978-80-7435-242-3.
- [17] Jižní Korea jako první země spustila 5G síť, je až stokrát rychlejší. *ČT24* [online]. 2019-04-06 [cit. 2019-20-12]. Dostupné z: <https://ct24.ceskatelevize.cz/svet/2780102-jizni-korea-jako-prvni-zeme-spustila-5g-sit-je-az-stokrat-rychlejsi>
- [18] HOFFMAN, Chris. *What Is 5G, and How Fast Will It Be?* [online]. March 15, 2019 [cit. 2019-12-20]. Dostupné z: <https://www.howtogeek.com/340002/what-is-5g-and-how-fast-will-it-be/>

- [19] SHANKAR, Ramya. *What is Java?* [online]. 27 Jan 2020 [cit. 2020-01-31]. Dostupné z: <https://hackr.io/blog/what-is-java>
- [20] TIOBE - The Software Quality Company. *TIOBE Index* [online]. TIOBE software BV [cit. 2020-01-31]. Dostupné z: <https://www.tiobe.com/tiobe-index/>
- [21] Google. *Meet Android Studio* [online]. 2020 [cit. 2020-01-31]. Dostupné z: <https://developer.android.com/studio/intro>
- [22] JCRAFT. JSch - Java Secure Channel. *JCraft: Code the Craft, Craft the Code* [online]. 2016 [cit. 2020-02-10]. Dostupné z: <http://www.jcraft.com/jsch/>
- [23] *Database Concepts: Introduction to Oracle Database* [online]. [cit. 2020-02-10]. Dostupné z: <https://docs.oracle.com/database/121/CNCPT/intro.htm#CNCPT001>
- [24] Oracle SQL Developer. *Oracle* [online]. [cit. 2020-02-14]. Dostupné z: <https://www.oracle.com/database/technologies/appdev/sql-developer.html>
- [25] JDBC. Oracle FAQ [online]. [cit. 2020-02-14]. Dostupné z: <https://www.orafaq.com/wiki/JDBC>
- [26] ORACLE. JDBC and UCP Downloads page. *Oracle FAQ* [online]. [cit. 2020-02-14]. Dostupné z: <https://www.oracle.com/database/technologies/appdev/jdbc-downloads.html>
- [27] Mobile operating systems' market share worldwide from January 2012 to December 2019. *Statista* [online]. 2020, January 2020 [cit. 2020-02-14]. Dostupné z: <https://www.statista.com/statistics/272698/global-market-share-held-by-mobile-operating-systems-since-2009/>
- [28] RAPHAEL, JR. Android versions: A living history from 1.0 to 10. *Computer World US* [online]. Sep 13, 2019 [cit. 2020-01-05]. Dostupné z: <https://www.computerworld.com/article/3235946/android-versions-a-living-history-from-1-0-to-today.html>
- [29] GOOGLE. Android 5.0 APIs [online]. [cit. 2020-01-05]. Dostupné z: <https://developer.android.com/about/versions/android-5.0.html>
- [30] *Distribution dashboard* [online]. 2019-12-27 [cit. 2020-02-14]. Dostupné z: <https://developer.android.com/about/dashboards>
- [31] GOOGLE. *Maps SDK for Android: Overview* [online]. January 2020 [cit. 2020-02-16]. Dostupné z: <https://developers.google.com/maps/documentation/android-sdk/intro>

## 10 Přílohy

### 10.1 Obsah přiloženého CD

- Archiv **Diplom.zip**  
Tento archiv obsahuje soubory projektu, který byl vytvořen ve vývojovém prostředí Android Studio.
- Soubor **db.sql**  
Skript, který vytvoří v projektu použitou databázovou strukturu.
- Soubor **Data.xlsx**  
Soubor, který obsahuje všechny měření, která byla provedena v rámci testování a vyhodnocení kvality mobilního internetového připojení.
- Soubor **Text.pdf**  
Soubor, který obsahuje text diplomové práce ve formátu PDF.

## 10.2 Oskenované zadání diplomové práce

UNIVERZITA HRADEC KRÁLOVÉ  
Fakulta informatiky a managementu  
Akademický rok: 2019/2020

Studijní program: Aplikovaná informatika  
Forma studia: Kombinované  
Obor/kombinace: Aplikovaná informatika (ai2-k)

### Podklad pro zadání DIPLOMOVÉ práce studenta

Jméno a příjmení: Bc. Jan Čučiak  
Osobní číslo: I1800314  
Adresa: Tyršova 764, Holice, 53401 Holice v Čechách, Česká republika  
Téma práce: Měření kvality mobilního připojení k internetu  
Téma práce anglicky:  
Vedoucí práce: Ing. Pavel Kříž, Ph.D.  
Katedra informatiky a kvantitativních metod

#### Zásady pro vypracování:

Cíl: Analýza současného stavu mobilního Internetu v České republice (technologie, pokrytí). Navrhnout a implementovat mobilní aplikaci pro měření kvality internetového připojení (rychlost uploadu, downloadu, latence). Jako protistranu měření bude možné vybrat různé servery. Následně provést měření a výsledky vyhodnotit. Výsledky budou mj. zakresleny do mapy.

#### Osnova:

1. Úvod
2. Cíl
3. Přehled mobilních technologií
4. Rešerše existujících řešení
5. Analýza vlastního řešení
6. Návrh a implementace
7. Měření a výsledky
8. Závěr

#### Seznam doporučené literatury:

[https://www.researchgate.net/publication/48183158\\_Optimization\\_and\\_Performance\\_Analysis\\_of\\_High\\_Speed\\_Mobile\\_Access\\_Networks](https://www.researchgate.net/publication/48183158_Optimization_and_Performance_Analysis_of_High_Speed_Mobile_Access_Networks)  
<http://wndw.net/book.html>  
<https://www.wiley.com/en-us/Cellular+Technologies+for+Emerging+Markets%3A+2G%2C+3G+and+Beyond-p-9780470779477>  
<https://www.elsevier.com/books/cellular-internet-of-things/liberg/978-0-12-812458-1>  
<https://www.oreilly.com/library/view/advanced-wireless-networks/9781119096856/>

Podpis studenta:



Datum: 17. 9. 2019

Podpis vedoucího práce:

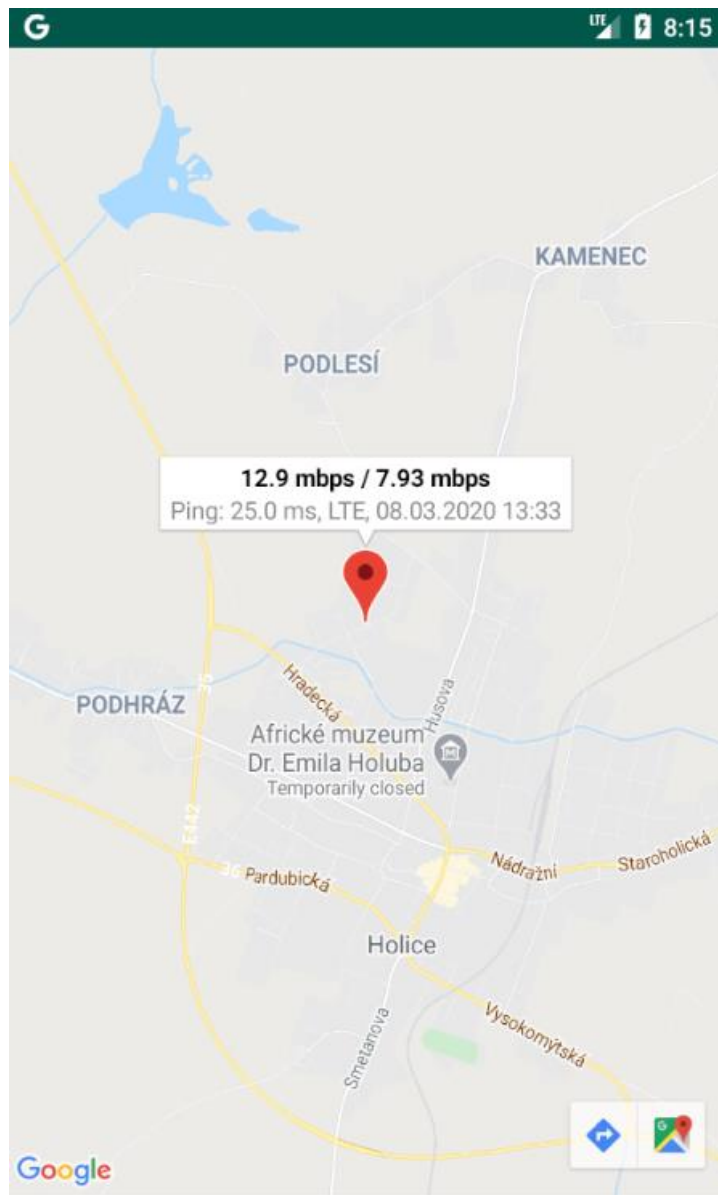


Datum: 17. 9. 2019



### 10.3 Ukázka mobilní aplikace

Ukázka zobrazení výsledku měření na mapě.



Ukázka možnosti nastavení testu.

**Diplom**

## Speed test

Test type:  
Complete

Test settings:  
Manual

Download - Size of test file  
8 MB

Upload - Size of test file  
2 MB

Save GPS?  
Yes

Custom server:  
www.google.com

**RUN TEST**

Ukázka zobrazení výsledků provedeného testu internetového připojení.

