



ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

FAKULTA ELEKTROTECHNICKÁ

KATEDRA MĚŘENÍ

Diplomová práce

System pro sběr dat při subjektivním testování

Bc. Adam Hamr

Vedoucí práce: prof. Ing. Jan Holub, Ph.D.

25. května 2018

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Hamr** Jméno: **Adam** Osobní číslo: **347918**
Fakulta/ústav: **Fakulta elektrotechnická**
Zadávací katedra/ústav: **Katedra měření**
Studijní program: **Kybernetika a robotika**
Studijní obor: **Senzory a přístrojová technika**

II. ÚDAJE K DIPLOMOVÉ PRÁCI

Název diplomové práce:

Systém pro sběr dat při subjektivním testování

Název diplomové práce anglicky:

Data Acquisition System for Subjective Testing

Pokyny pro vypracování:

Navrhněte a realizujte systém pro sběr dat při subjektivním testování kvalitativních parametrů zvukových vzorků. Použitá platforma: zařízení s dotykovým displejem s omezeným výpočetním výkonem (tablet, smartphone) s OS Android. Vytvořený program musí umožňovat:

- načtení dat (soubory *.wav, 48kSa/s, 16bit, mono) z nadřazeného systému a volbu metodiky testu;
- generování náhodného pořadí při přehrávání;
- přehrání vzorků včetně vhodného časování a sběru dat;
- přenos dat do nadřazeného systému.

Pro komunikaci zvolte vhodnou bezdrátovou cestu (Bluetooth nebo Wifi). Nadřazený systém může být buď opět zařízení s OS Android, nebo PC s OS MS Windows. Funkčnost vytvořeného zařízení demonstřujete provedením subjektivního testu malého rozsahu a vyhodnocením dat.

Seznam doporučené literatury:

ITU-T P.800, ITU-T P.807, ITU-T P.835

Jméno a pracoviště vedoucí(ho) diplomové práce:

prof. Ing. Jan Holub, Ph.D., katedra měření FEL

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) diplomové práce:

Datum zadání diplomové práce: **18.02.2018**

Termín odevzdání diplomové práce: **25.05.2018**

Platnost zadání diplomové práce:

do konce letního semestru 2018/2019

prof. Ing. Jan Holub, Ph.D.
podpis vedoucí(ho) práce

podpis vedoucí(ho) ústavu/katedry

prof. Ing. Pavel Ripka, CSc.
podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Diplomant bere na vědomí, že je povinen vypracovat diplomovou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v diplomové práci.

Datum převzetí zadání

Podpis studenta

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval(a) samostatně a že jsem uvedl(a) veškeré použité informační zdroje v souladu s Metodickým pokynem o etické přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů. V souladu s ust. § 46 odst. 6 tohoto zákona tímto uděluji nevýhradní oprávnění (licenci) k užití této mojí práce, a to včetně všech počítačových programů, jež jsou její součástí či přílohou, a veškeré jejich dokumentace (dále souhrnně jen „Dílo“), a to všem osobám, které si přejí Dílo užít. Tyto osoby jsou oprávněny Dílo užít jakýmkoli způsobem, který nesnižuje hodnotu Díla, a za jakýmkoli účelem (včetně užití k výdělečným účelům). Toto oprávnění je časově, teritoriálně i množstevně neomezené. Každá osoba, která využije výše uvedenou licenci, se však zavazuje udělit ke každému dílu, které vznikne (byť jen zčásti) na základě Díla, úpravou Díla, spojením Díla s jiným dílem, zařazením Díla do díla souborného či zpracováním Díla (včetně překladu), licenci alespoň ve výše uvedeném rozsahu a zároveň zpřístupnit zdrojový kód takového díla alespoň srovnatelným způsobem a ve srovnatelném rozsahu, jako je zpřístupněn zdrojový kód Díla.

V Praze dne 25. května 2018

.....

České vysoké učení technické v Praze

Fakulta elektrotechnická

© 2018 Adam Hamr. Všechna práva vyhrazena.

Tato práce vznikla jako školní dílo na Českém vysokém učení technickém v Praze, Fakultě elektrotechnické. Práce je chráněna právními předpisy a mezinárodními úmluvami o právu autorském a právech souvisejících s právem autorským. K jejímu užití, s výjimkou bezúplatných zákonných licencí a nad rámec oprávnění uvedených v Prohlášení na předchozí straně, je nezbytný souhlas autora.

Odkaz na tuto práci

Hamr, Adam. *Systém pro sběr dat při subjektivním testování*. Diplomová práce. Praha: České vysoké učení technické v Praze, Fakulta elektrotechnická, 2018.

Abstrakt

Tato diplomová práce je zaměřena na implementaci mobilní aplikace Wireless Subjective Tester, která umožňuje provádět poslechové subjektivní testy. Je popsán návrh a implementace systému. Tento systém pomocí Wi-Fi přenáší data mezi administrátorem testu a subjekty, asynchronně přehrává zvukové soubory subjektům, získává výsledky testování od subjektů a předává je administrátorovi. Mobilní aplikace je implementována v prostředí Android Studio. Stručně jsou přiblíženy tři metodiky subjektivního testování definované doporučeními organizace ITU-T. Práce se nevěnuje vedení subjektivních testů ani analýze získaných subjektivních dat.

Klíčová slova subjektivní, testování, android, studio, Wi-Fi, sběr, dat, mobil, aplikace

Abstract

This master's thesis is aimed at implementation of mobile application Wireless Subjective Testing which enables the making of subjective listening opinion tests. The design and the implementation of the system are described. This system with the help of Wi-Fi transmits data between the test administrator and the subjects, asynchronously plays audio files to the subjects, gains the

results of testing from the subjects and delivers them to the administrator. The mobile application is implemented in the environment of Android Studio. Three methods of subjective testing are briefly described which are defined by the recommendation of the ITU-T organisation. The thesis is not aimed at leading subjective tests or the analysis of the subjectively obtained data.

Keywords subjective, testing, android, studio, Wi-Fi, data, acquisition, mobile, application

Obsah

Úvod	1
1 Analýza	3
1.1 Testovací metody podle doporučení ITU-T	3
1.2 Identifikace požadavků	5
1.3 Možnosti řešení	6
2 Návrh	9
2.1 Návrh funkcionality	9
2.2 Konfigurace testu	10
2.3 Návrh obrazovek	11
2.4 Návrh datové komunikace	12
3 Implementace	15
3.1 Zvolené řešení	15
3.2 Struktura aplikace	19
4 Výsledek implementace	23
4.1 Uživatelské rozhraní	23
Závěr	27
Literatura	29
A Seznam použitých zkratk	31
B Obsah příloženého CD	33

Seznam obrázků

2.1	Odeslání zadání od administrátora k subjektovi	13
2.2	Odeslání řešení od subjektu k administrátorovi	14
2.3	Rozlišení mezi příchozím archivem a textem	14
3.1	Diagram životního cyklu MediaPlayeru [1]	18
3.2	WSTActivity.java	20
3.3	P2pConnectFragment.java	21
3.4	P8XXFragment.java	21
4.1	Test podle P.800 ACR	23
4.2	P.800 DCR – hlasování	24
4.3	Testy podle P.807	24
4.4	Nastavení administrátorské konzole	25

Seznam tabulek

1.1	Porovnání technologií Wi-Fi a Bluetooth	6
-----	---------------------------------------------------	---

Úvod

V současné době je stále aktuální vydatný rozvoj hlasových systémů, ať už se jedná o přenosové soustavy, záznamová a archivační zařízení nebo povelovací automaty. Tento rozvoj vyžaduje masivní aktivitu v oblasti kódování a komprimace zvukových dat a při potlačování šumu.

Je obvyklé, že práce s daty bývá snadno měřitelná, pokud se jedná například o statistiky. Ale otázka kvality zvukových dat je natolik komplexní, že nejde změřit jako jedna nebo několik málo veličin měřícím přístrojem. Není možné vynechat subjektivní hodnocení při poslechu záznamu.

Řešení pro hledání tohoto hodnocení nabízejí metody subjektivního testování, při kterém několik desítek respondentů tzv. subjektů poslouchá zvuky ze sluchátek a pomocí hlasovacího zařízení je hodnotí.

Cíl

Tato práce si klade za cíl vyvinout mobilní aplikaci Wireless Subjective Tester, která umožňuje poslechové subjektivní testování podle konkrétních doporučení vydaných Telekomunikačním sektorem Mezinárodní telegrafní unie (ITU-T). Jedná se o doporučení ITU-T P.800 [2], ITU-T P.807 [3] a ITU-T P.835 [4].

Aplikaci má umožnit zvuky přehrávat a odečítat odpovědi od subjektů. Dále má odpovědi předat do centrálního bodu administrátorovi testu a v neposlední řadě se aplikace bude starat o šíření zadání k jednotlivým subjektům. Zadání požaduje bezdrátové řešení v prostředí operačního systému Android.

Současná řešení

V současné době neexistuje řešení v podobě mobilní aplikace, které by podporovalo subjektivní testování přímo podle doporučení ITU-T P.800, ITU-T P.807 nebo ITU-T P.835. Nástroje pro testy podle těchto doporučení zatím

ÚVOD

mají tradici spíše v systémech založených na PC. V rámci výzkumu Fakulty elektrotechnické ČVUT testy probíhají pomocí aplikace, která běží v prostředí LabView za použití hardwarových hlasovacích zařízení. Dále existují mobilní aplikace pro nějaké subjektivní testování, ale testují spíše hodnocení zvuku podle jiných doporučení nebo hodnocení videí, případně zdraví smyslových orgánů [5].

Mobilní telefon je však výhodná platforma pro subjektivní testování, což bude rozebráno dále.

Analýza

Předpokládá se, každý subjekt bude mít mobilní telefon se sluchátky, dotykovým displejem a operačním systémem Android. Aplikace poběží na každém telefonu samostatně, tedy nebude mít centrální server, který by si vyměňoval data s webovými prohlížeči subjektů.

Další součást produktu bude provozován na mobilním telefonu nebo tabletu administrátora testu. Odtud se bude rozesílat zadání testu subjektům a budou se zde sbírat odpovědi subjektů na testové otázky.

Aby bylo jasné, co je předmětem těchto otázek, budou následně rozebrány jednotlivé metody.

1.1 Testovací metody podle doporučení ITU-T

Tato práce se podle zadání má zaměřit na následující metodologie. Každá z nich je orientována na hodnocení kvality konkrétní součásti obecného zvukového řetězce:

P.800 Metody subjektivního určení kvality přenosu

P.807 Subjektivní metodologie posouzení srozumitelnosti řeči

P.835 Subjektivní metodologie pro ohodnocení řečových komunikačních systémů využívajících algoritmy potlačení šumu

Všechny tyto metodologie definují způsob (neboli rytmiku), podle kterého subjekt střídavě poslouchá zvukové vzorky a hlasuje. Každá metoda zvlášť říká, kolik vzorků se přehraje, než subjekt začne hlasovat, a z jakého pohledu subjekt hodnotí. Je dána doporučená celková doba jednoho testu. Neříká se však, kolik celkem musí být ohodnoceno vzorků, než je daný případ takzvaně otestovaný.

1.1.1 P.800 – Metody subjektivního určení kvality přenosu

Zaměřují se na přenosovou soustavu. Doporučení [2] zahrnuje tři metody poslechového určení kvality. Obsahuje také další způsoby nesouvisející s poslechovými testy. První dvě jsou součástí této práce, poslední z nich na základě ústní dohody není potřeba implementovat.

1. ACR – Absolute category rating
2. DCR – Degradation category rating
3. CCR – Comparison category rating

1.1.1.1 Absolute category rating

Rytmika – jeden vzorek → hlasování

Hlasování – stupnice od 1 do 5 mapuje možnosti od nejhorší po nejlepší kvalitu vzorku

1.1.1.2 Degradation category rating

Rytmika – vzorek v původní kvalitě → vzorek degradovaný → hlasování

Hlasování – stupnice od 1 do 5 mapuje možnosti od nejsilnější po nejslabší degradaci vzorku

1.1.2 P.807 – Metodologie posouzení srozumitelnosti řeči

Doporučení [3] nabízí dvě metody pro stanovení rozlišitelnosti toho, co bylo v nahrávce řečeno. Po vyslechnutí vzorku subjekt dostane na výběr několik slov a následně rozhoduje, které z nich slyšel.

1.1.2.1 Diagnostic rhyme test

Rytmika – jeden vzorek → hlasování → potvrzení volby

Hlasování – volba ze dvou variant

1.1.2.2 Modified rhyme test

Rytmika – jeden vzorek → hlasování → potvrzení volby

Hlasování – volba ze šesti variant

1.1.3 P.835 – Metodologie posouzení algoritmů potlačení šumu

Doporučení [4] popisuje, jak zachytit názor subjektů na kvalitu signálu, okolního šumu a celkový dojem

- Rytmika**
1. vzorek1 → hlasování o kvalitě signálu (S-MOS = signal mean opinion score)
 2. vzorek1 → hlasování o kvalitě šumu (N-MOS = noise mean opinion score)
 3. vzorek1 → hlasování o kvalitě celkové (G-MOS = global mean opinion score)
 4. vzorek2 → hlasování N-MOS
 5. vzorek2 → hlasování S-MOS
 6. vzorek2 → hlasování G-MOS
 7. ...

Hlasování – stupnice od 1 do 5 mapuje možnosti od nejhorší po nejlepší kvalitu vzorku

1.2 Identifikace požadavků

1. Bezdrátové odeslání zvukových vzorků a souboru s konfigurací testu ze zařízení administrátora všem subjektům
2. Subjekt nesmí být povinen aktivně připojovat zařízení do sítě nebo provádět složitou sekvenci inicializačních kroků
3. Subjekt potvrdí spuštění testu a od té doby nesmí být přerušen v testování ze strany administrátora nebo selháním aplikace
4. Přehrávání vzorků v náhodném pořadí
5. Tlačítka pro hlasování co největší, aby plně zabírala plochu displeje
6. Dokud se přehrává vzorek nesmí být umožněno hlasovat
7. Po hlasování systém po určité době (například 500 ms) automaticky přehraje další vzorek
8. Bezdrátové odeslání výsledků hodnocení po skončení testu

Technologie	Wi-Fi	Bluetooth
Změřený dosah	60 m	10 m
Deklarovaný maximální dosah	200 m [8]	30 m [9]
Deklarovaná přenosová rychlost	250 Mbps [8]	–

Tabulka 1.1: Porovnání technologií Wi-Fi a Bluetooth

1.3 Možnosti řešení

1.3.1 Výhody mobilního řešení

1. kvalitní interpretace zvuku
2. bezdrátové připojení k administrátorovi
3. hlasování na dotykovém displeji, aby byl zajištěn tichý provoz bez zvuků tlačítek

Je požadována co nejplošší frekvenční charakteristika přehrávače a minimálně 70 dB odstup signálu od šumu. U většiny dražších telefonů jsou tyto parametry splněny. To se týká jak současných [6] tak i starších [7] telefonů¹.

Subjekti pravděpodobně budou mít sluchátka drátová, ale podstatné je, že nebudou připojeni kabelem k nějakému nadřazenému systému. Testování tak může být přesunuto do libovolného prostředí (které stále splňuje podmínky pro testování) a během hodnocení mohou pohodlně provádět další činnost (jezdit na rotopedu, jíst apod.) stanovenou testem.

1.3.2 Volba principu bezdrátové komunikace

Nabízí se možnosti propojení účastníků pomocí Wi-Fi a přes Bluetooth. Typické vzdálenosti a přenosové rychlosti u Wi-Fi [8] jsou uváděny vyšší než u Bluetooth [9]. Změřené hodnoty by však u obou technologií byly menší. Pro účel této práce byl změřen dosah Wi-Fi mezi dvěma mobilními telefony propojenými protokolem Wi-Fi Direct [8].

Při pohledu na čísla v tabulce 1.1 je rozhodující vzdálenost, na kterou zařízení mohou spolehlivě komunikovat. Byla tedy zvolena technologie Wi-Fi, která dosáhne na několikanásobně delší vzdálenosti než Bluetooth.

1.3.3 Volba Wi-Fi standardu

Pomocí Wi-Fi se dá komunikovat buď po připojení do společné sítě poskytované centrálním přístupovým bodem. Nebo se mohou zařízení sami propojit mezi sebou podle principu Peer-to-peer.

¹Tyto zdroje uvádí změřené vlastnosti přehrávače v telefonech. Informace však neobsahují přesnost (nejistotu) měření a mohou se použít pouze jako orientační.

V prostředí Android již od verze 4.0 existuje možnost propojovat zařízení mezi sebou pomocí Peer-to-peer [10] a komunikace přes přístupový bod byla přidána ve verzi pozdější. Byl zvolen princip Peer-to-peer, protože je déle používaný, a proto by měl obsahovat méně chyb a být předmětem většího množství dokumentace. Více o této technologii je ve 3.1.2.3.

1.3.4 Vývojové prostředí

Z osobních důvodů bylo zvoleno vývojové prostředí Android Studio. Důvodem je především větší zkušenost.

Návrh

2.1 Návrh funkcionality

2.1.1 Činnost subjektů

Nastavení testu Subjekt, kterému se dostane telefon do ruky, otevře aplikaci a počká, až mu bude nastaven test. Nastavení testu je v tomto případě příjem zvukových vzorků a konfiguračního souboru testu v archivu .ZIP. Ten se po přijetí rozbalí a uvede se test do provozu. Podle požadavků nebude zatěžován inicializací aplikace, ta se tedy provede automaticky.

Spuštění testu Jakmile je test nastaven a subjekt stiskne „Spustit test“, telefon se odpojí od administrátorova zařízení a proběhne test. Po ukončení testu se aplikace vrátí na obrazovku, která začne vyhledávat k administrátorovo zařízení. Pokud je k dispozici soubor ve formátu .CSV s výsledky testu, který ještě nebyl odeslán, aplikace soubor předá administrátorovi.

Potvrzení hlasování Podle přání administrátora bude v konfiguraci testu nastaveno potvrzování každého hlasování (například nějakým tlačítkem „Enter“). Podle doporučení P.800 ani P.835 potvrzování není žádoucí, avšak P.807 ho vyžaduje. Proto se také různí názory na to, jestli se tato funkcionality má nebo nemá používat. Pro implementaci je tedy nejvhodnější nechat na rozhodnutí administrátora, jestli bude potvrzování zapnuto nebo ne.

Pokud bude zapnuto, bude potřeba, aby uživatel při každém hlasování stiskl tlačítko dvakrát. Pokud nějaké tlačítko bude stisknuto jednou, nestane se nic. Tato funkcionality se zdá být vhodnější, než překážející tlačítko Enter.

2.1.2 Činnost administrátora

Příprava konfigurace Administrátor nejprve připraví konfiguraci a archiv se soubory. Stiskne tedy tlačítko nastavení a v menu volí:

2. NÁVRH

- identifikátor testu
- podadresář s připravenými soubory vzorků
- testovací metodu
- vybere stupnici hodnocení
- položku „doubleClickNeeded“, která když je nenulová, je zapnuto potvrzování volby
- případně identifikátor dalšího testu, pokud se testy mají za sebe řetězit
- údaj v minutách, kolik času se subjektům doporučuje si vzít oddech před zahájením dalšího testu

Poté stiskne vyhledat všechna zařízení.

Připojení k subjektům Jakmile je připravena konfigurace, administrátor tlačítkem vyhledá zařízení subjektů. Dále se k jednomu nebo všem najednou připojí a tlačítkem „Send file“ rozešle archiv s testem. Zařízení subjektů se postupně odpojí. Po vyřešení testu začnou být zařízení subjektů k dispozici a administrátor se k nim může připojit, aby obdržel výsledky testu.

Konfigurace testu se ukládá a sdílí způsobem v následující sekci.

2.2 Konfigurace testu

Konfigurace může být uložena třemi různými způsoby:

- nastavení aplikace administrátora – zapamatuje se do příštího startu (případně restartu) aplikace nebo telefonu
- serializovaná konfigurace do formátu JSON (více o tomto formátu v 3.1.2.1) na cestě od administrátora k subjektům
- dočasné nastavení aplikace subjektu ve formě Java objektu

Struktura konfigurace v JSON je následující:

```
{  
  "testId": "ACRtestNr.1",  
  "testSubdirectory": "ACRtest",  
  "method": "P800Acr",  
  "options": [  
    "Bad intelligibility",  
    "Poor intelligibility",  
    "Average intelligibility",  
    "Good intelligibility",  
    "Excellent intelligibility"  
  ]  
}
```

```

],
"samples": [
  {
    "sampleName": "1",
    "sampleFileName": "1.wav"
  },
  {
    "sampleName": "2",
    "sampleFileName": "2.wav"
  },
  {
    "sampleName": "3",
    "sampleFileName": "3.wav"
  }
],
"doubleClickNeeded": false,
"nextTestId": "ACRtestNr.1",
"pauseBeforeNext": 10
}

```

Položka `testId` by teoreticky mohla být shodná s „`testSubdirectory`“. Avšak pokud by subjekt nebo vystřídáný subjekt se stejným zařízením chtěl test opakovat se stejným zadáním, musí se rozlišit, výsledky různých subjektů. Dále je v textu vidět zvolená metoda a odpovídající možnosti pro odpovědi².

Vzorky se hledají v souborovém systému zařízení pod názvem v poli „`samples`“ v podadresáři „`testSubdirectory`“. Dále se v tomto umístění nachází textový dokument s touto konfigurací ve formátu JSON a také se sem ukládají výsledky testů ve formátu .CSV.

2.3 Návrh obrazovek

Obrazovky pro subjekty jsou pouze dvě, aby ovládání bylo co nejjednodušší. Administrátor má o jednu obrazovku více:

- Subjekt
 1. Komunikace s administrátorem
 2. Obrazovka s testem
- Administrátor
 1. Správa účastníků
 2. Nastavení
 3. Vyzkoušení testu

²Pro metody podle P.807 jsou pro každý vzorek možnosti pro odpovědi jiné a jsou tedy zahrnuty v poli vzorků

2.4 Návrh datové komunikace

Aby výměna dat mohla probíhat podle 2.1, je potřeba zavést několik opatření, aby komunikace mohla probíhat v mezích volby standardu v 1.3.3.

2.4.1 Odesílání zadání testu

Na obr. 2.1 je schéma rozesílání testových zadání.

Před každým spojením musí oba účastníci komunikující peer-to-peer vyhledat zařízení. Aplikace subjektů toto bude provádět automaticky po spuštění, administrátor bude vyhledávat ručně. Po té se administrátor pošle žádost o připojení k účastníkovi na linkové vrstvě ISO/OSI modelu (více o komunikaci peer-to-peer v 3.1.2.3).

Subjekt potvrdí žádost. Sice má provádět co nejméně činností během inicializace, takže by bylo nejlepší, kdyby se nemusel starat o propojení s administrátorem. Ale z bezpečnostních důvodů nelze komunikaci nastavit tak, aby se k zařízení subjektů mohl připojit kdokoliv a nahrát do něj cokoli. Proto při komunikaci peer-to-peer dochází k potvrzování připojení formou malého dialogu na straně žádaného o připojení. Po několika prvních spojení si operační systém zapamatuje, že s tímto účastníkem uživatel komunikuje a později potvrzení proběhne automaticky bez účasti uživatele.

Následně subjekt odešle krátký textový řetězec s názvem zařízení³ ve formátu JSON:

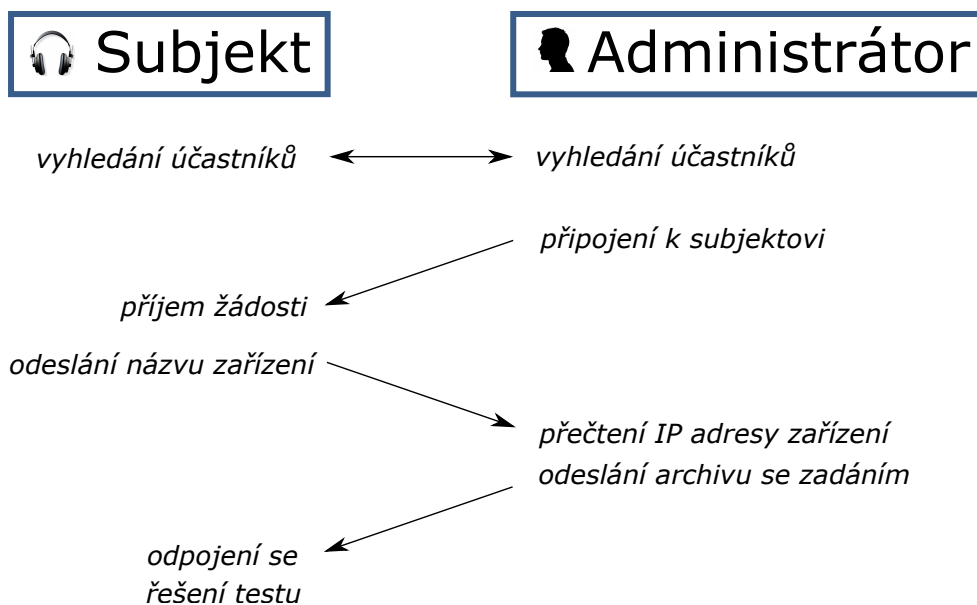
```
{  
  "deviceName": "Subjekt 01"  
}
```

Nyní si administrátor přečte ze socketu IP adresu účastníka a uloží si ho do seznamu účastníků na řádek se jménem, které právě obdržel. Tím může na tuto IP adresu odeslat archiv se zadáním testu. Subjekt si test uloží a spustí a může se odpojit od administrátora.

2.4.2 Odesílání výsledků testu

Jakmile je test vyřešen, můžou se podobným způsobem jako v předchozím odstavci spojit administrátor a subjekt. Tentokrát komunikace proběhne podle obr. 2.2. Tentokrát je uvedeno schéma, kdy si zařízení subjektů již pamatuje administrátorovu MAC adresu a nemusí tedy potvrzovat připojení administrátora ručně.

³Administrátor jako iniciátor peer-to-peer spojení zároveň všem účastníkům sděluje svou vlastní IP adresu a má roli Group owner. Připojení účastníci tedy mohou administrátora kontaktovat pomocí IP adresy jako klienti serveru. Dokud však administrátor nepřijme žádná data od ostatních, nezná jejich IP adresy.



Obrázek 2.1: Odeslání zadání od administrátora k subjektovi

Protože během testu bylo spojení přerušeno, mohlo dojít ke změně IP adresy subjektu a administrátor si ji aktualizuje. Děje se tak také proto, že subjekt zprávu na adresu Group ownera odesílá po každém navázání spojení. Přenese se soubor s výsledky a spojení se ukončí.

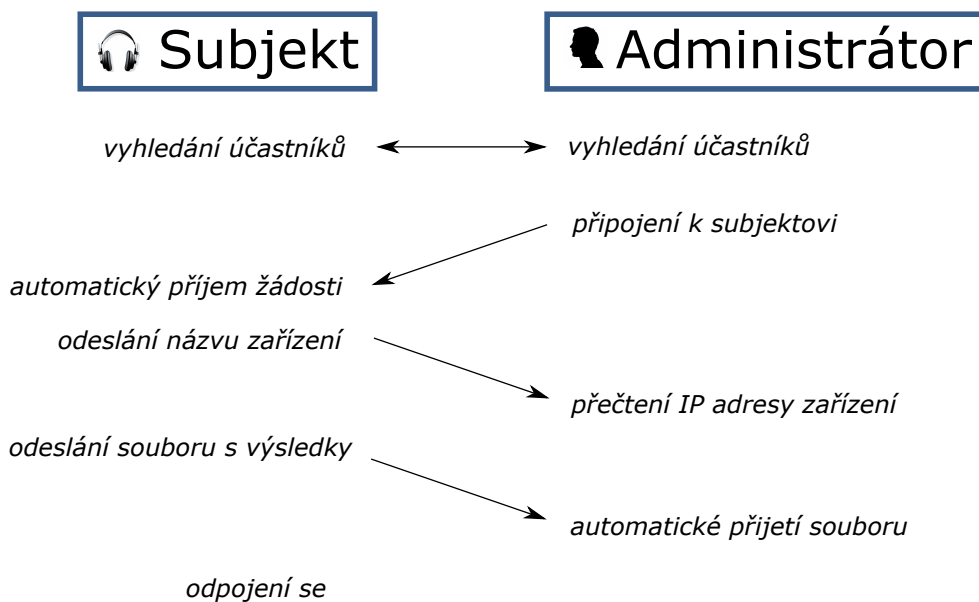
2.4.3 Přenos dat mezi subjektem a administrátorem

Je vidět, že se stejnou cestou jednou posílají soubory a jindy pouze text. Zatímco soubory je potřeba hned ukládat do souborového systému zařízení a později zpracovávat, na textové zprávy je nutné reagovat ihned.

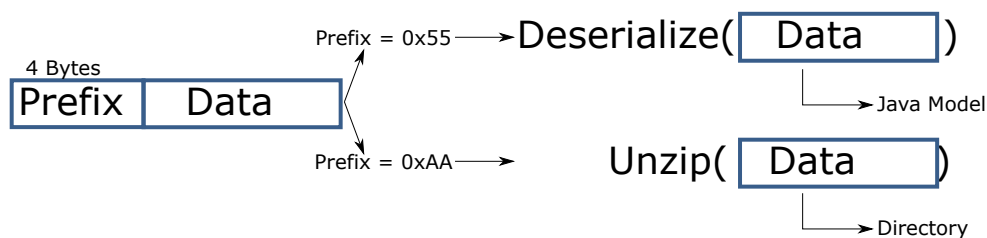
Pro rozlišení mezi příchozím souborem a příchozí zprávou se podle obr. 2.3 vkládá nezaměnitelný prefix do odesílaných dat. Z příchozích dat se naopak odmazává a po jeho přečtení se určí, jak s datovým proudem naložit.

Nabízí se další způsob, jak oba datové formáty rozlišit. Například komunikace zpráv na jiném portu než přenos souborů. V tomto případě by však zařízení musela čekat na dvou portech na příchozí data a vzhledem k malé četnosti komunikace by tento způsob znamenal víc zátěže než užítku.

Naopak použitý způsob s prefixy by se nehodil pro častou výměnu dat, protože při každém přenosu je vynaložena práce navíc.



Obrázek 2.2: Odeslání řešení od subjektu k administrátorovi



Obrázek 2.3: Rozlišení mezi přichozím archivem a textem

Implementace

3.1 Zvolené řešení

3.1.1 Použité technologie a knihovny

3.1.2 Randomizace přehrávání vzorků

Je požadováno, aby se zvukové soubory přehrávaly subjektům v náhodném pořadí. Přitom musí být zachovány vazby mezi vzorky, které k sobě patří. Má-li se v určitém kroku přehrát vzorek a následně jeho degradovaná varianta, je potřeba zachovat toto pořadí.

Proto jsou vzorky reprezentovány pomocí objektů, které mohou obsahovat adresy několika zvukových souborů.

Reference na objekty vzorků se při načítání testu řadí do datového typu `ArrayList` v pořadí, v jakém byly uloženy. Pak stačí vygenerovat náhodné číslo od nuly do posledního indexu vzorku v `ArrayListu`, vzorek na tomto indexu přehrát a referenci odebrat z `ArrayListu`. Jakmile v seznamu není už žádný vzorek, testování skončí.

Pro generování náhodných čísel je potřeba nalézt dostupnou náhodnou veličinu a vyčíslit její hodnotu v potřebném rozsahu. V počítačovém systému se taková veličina hledá těžko, ale při interakci s uživatelem lépe. Dá se spolehnout na to, že uživatel hlasuje v čase naprosto nahodilém.

Může se tedy s pomocí operačního systému zjistit, kolik uběhlo milisekund od 1. 1. 1970 do okamžiku, ve kterém uživatel hlasuje. K tomu slouží funkce `System.currentTimeMillis()`. Z tohoto čísla se vypočte zbytek po dělení počtem prvků v seznamu a vyjde náhodné číslo v potřebném rozsahu.

3.1.2.1 JSON

JSON je zkratka JavaScript Object Notation. Podle [11] se jedná o formát dat, který slouží k předávání dat, je člověku snadno čitelný a stroji snadno zpracovatelný. Notace se dá využít ke konverzi dat z jakéhokoliv programovacího jazyka do textového formátu. Jakýkoliv objekt lze mapovat do formátu JSON.

JSON odpovídá standardu ECMA, který také definuje například notaci Javascriptu.

Narozdíl od formátu XML má JSON výhodu v tom, že obsahuje méně textu mimo vlastní data.

Knihovna GSON V tomto projektu se používá knihovna GSON, která automatizuje převod mezi formátem JSON a Java objekty. Pro tuto knihovnu je charakteristické, že umí pracovat s libovolnými objekty, i když už není znám jejich zdrojový kód, a s generickými objekty [12].

3.1.2.2 MediaPlayer

Zvuky se v aplikaci nepřehrávají blokujícím způsobem ve hlavním vlákně, ale asynchronně ve vlákně, které běží v pozadí. K tomu se využívá MediaPlayer [1]. Jde o třídu, která umožňuje přehrávat audio-video soubory a datové proudy.

MediaPlayer je řízen jako stavový automat, což znázorňuje obr. 3.1. V oválech jsou reprezentovány stavy, do kterých se instance objektu MediaPlayer může dostat, šipky představují operace, které řídí přehrávání.

Zpracování chyb Za zmínku stojí, že je definován chybový stav, do kterého se instance může dostat, když například nenalezne požadovaný zvukový soubor. Pokud vývojář instanci přiřadil `OnErrorListener` a nastane nějaký výjimečný stav, instance nevyhodí výjimku, ale zavolá metodu `onError` v `OnErrorListeneru`. Pokud se do instance výjimka propaguje z vnějšku, je zachycena a následně zavolána metoda `OnError`.

Tento princip je výhodný. Pro zacházení s MediaPlayerem se většinou vyčlení celá jedna metoda. Její obsah provádí IO operace a přitom nemůže dojít k vyhození výjimky. Není tak nutné metodu zneprůhledňovat tím, že se celý její obsah uzavře do bloku „try ... catch“.

Aplikace MediaPlayeru V diplomové práci MediaPlayer prochází stavy

1. Idle,
2. Initialized,
3. Prepared,

4. Started a nakonec

5. PlaybackCompleted.

Jsou definovány listenery

- OnErrorListener
- OnCompletionListener

Aby se subjektu zabránilo hlasovat během přehrávání, jsou deaktivována hlasovací tlačítka ještě před spuštěním MediaPlayeru. Pomocí OnCompletion-Listeneru se zajistí, že po vyslechnutí zvuku jsou tlačítka opět aktivována.

3.1.2.3 Wi-Fi Direct

Jedná se o standard umožňující bezdrátové připojení bez použití přístupového bodu. Původní název je Wi-Fi peer-to-peer nebo Wi-Fi P2P, který však v prostředí Android zůstal dodnes zachován [10].

Zařízení s certifikačním označením Wi-Fi Direct mohou sdílet soubory, synchronizovat se a zobrazovat obsah bez použití směrovače a bez nutnosti připojení k internetu. Výhodou tohoto principu je, že komunikace je chráněna technologií WPA2⁴. Nevýhodou je, že ve starých zařízeních (například s verzí OS Android menší než 5.0) může Wi-Fi Direct kolidovat s připojením ke směrovači. Používají stejný komunikační kanál pro připojení pomocí jak Wi-Fi Direct, tak k přístupovému bodu.

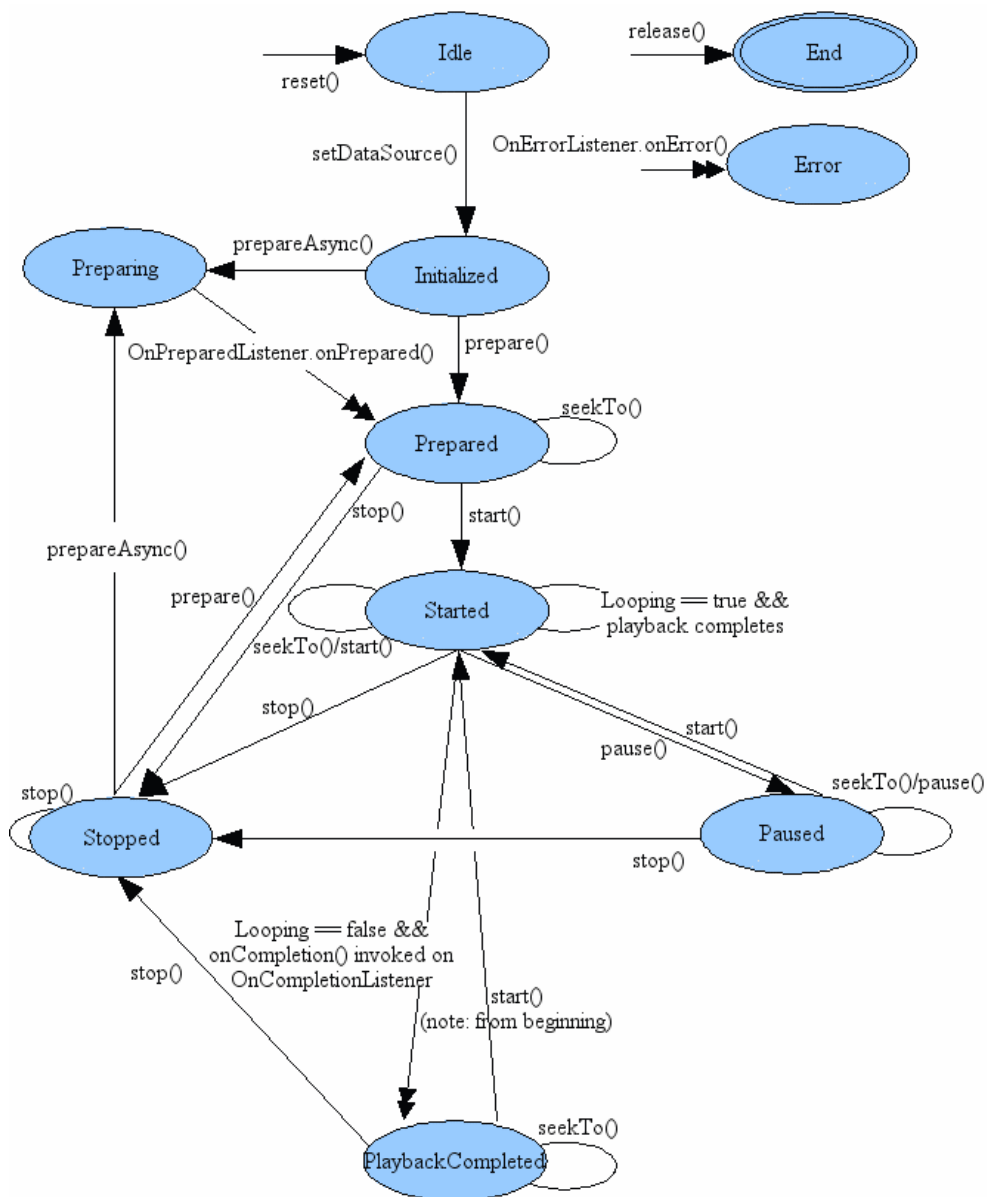
3.1.2.4 Wi-Fi Direct Demo

Ukázkový projekt vydaný společností Google, Inc., který demonstruje funkčnost Wi-Fi peer-to-peer v projektech běžících pod systémem Android. Slouží jako vzor, který ukazuje, jak je v Androidu podporována Wi-Fi komunikace mezi zařízeními. Projekt je určitým etalonem, který definuje konkrétní způsob, jak správně používat aplikační programovací rozhraní (API) a jak nejefektivněji implementovat peer-to-peer.

Projekt byl před časem smazán z webu Google a nyní je dostupný pouze v sekundárním zdroji [13]. Funguje tak, že umožňuje sestavit skupinu účastníků a jednosměrně posílat obrazové soubory. V diplomové práci z tohoto projektu vychází a částečně používá jeho backendovou část (vnitřní logiku). Avšak bylo nutné provést řadu úprav. V diplomové práci zbyly z původního projektu názvy souborů a několik fragmentů kódu.

⁴WPA2 je t.č. poslední zabezpečovací technologií pro Wi-Fi [8]

3. IMPLEMENTACE



Obrázek 3.1: Diagram životního cyklu MediaPlayeru [1]

3.2 Struktura aplikace

Z obr. 2.2 a obr. 2.1 je zřejmé, že subjekt i administrátor provádí zrcadlově stejné operace při přenosu dat. Hodí se tedy implementovat síťové operace jen jednou a implementaci používat na obou stranách. Proto byla vytvořena jen jedna aplikace, která je stejná pro subjekty i pro administrátora, ale má dva oddělené spouštěcí soubory. Případně je možné ve zdrojovém kódu zakomentovat výrobu spouštěcího souboru administrátora a po kompilaci a instalaci vznikne jen spouštěcí soubor pro subjekty.

Odlišnosti mezi komunikací subjektů a administrátora jsou řešeny dědičností, jak je uvedeno dále.

3.2.1 WSTActivity

Ústřední třídou nebo aktivitou v aplikaci je `WSTActivity.java`, která má na sobě navázány třídy zajišťující přenos dat po Wi-Fi a ze/do souborů. Je abstraktní a dědí od ní třídy `WSTHeadActivity.java` a `WSTSubjectActivity.java`, jak je vidět na obr. 3.2. První z nich používá administrátor a druhou subjekt. Tyto třídy jsou konkrétní a kreslí obrazovky za pomoci XML souborů s rozvržením (layoutem) ovládacích prvků. Také se podle těchto tříd jmenují spouštěcí soubory – `WSTHead` a `WSTSubject`.

`WSTHeadActivity.java` je frontend zajišťující správu účastníků a dědí vazby na třídy zajišťující IO operace. `WSTSubjectActivity.java` automatizuje komunikaci subjektů s administrátorem. Samočinně volá metody pro IO operace a také má vazby na třídy, které realizují testování.

3.2.2 P2pConnectFragment

`P2pConnectFragment.java` je objekt, který figuruje jako vlastnost (property) `WSTActivity.java`, zajišťuje peer-to-peer komunikaci, otevírá port pro asynchronní příjem dat⁵ a manipuluje s přijatými daty.

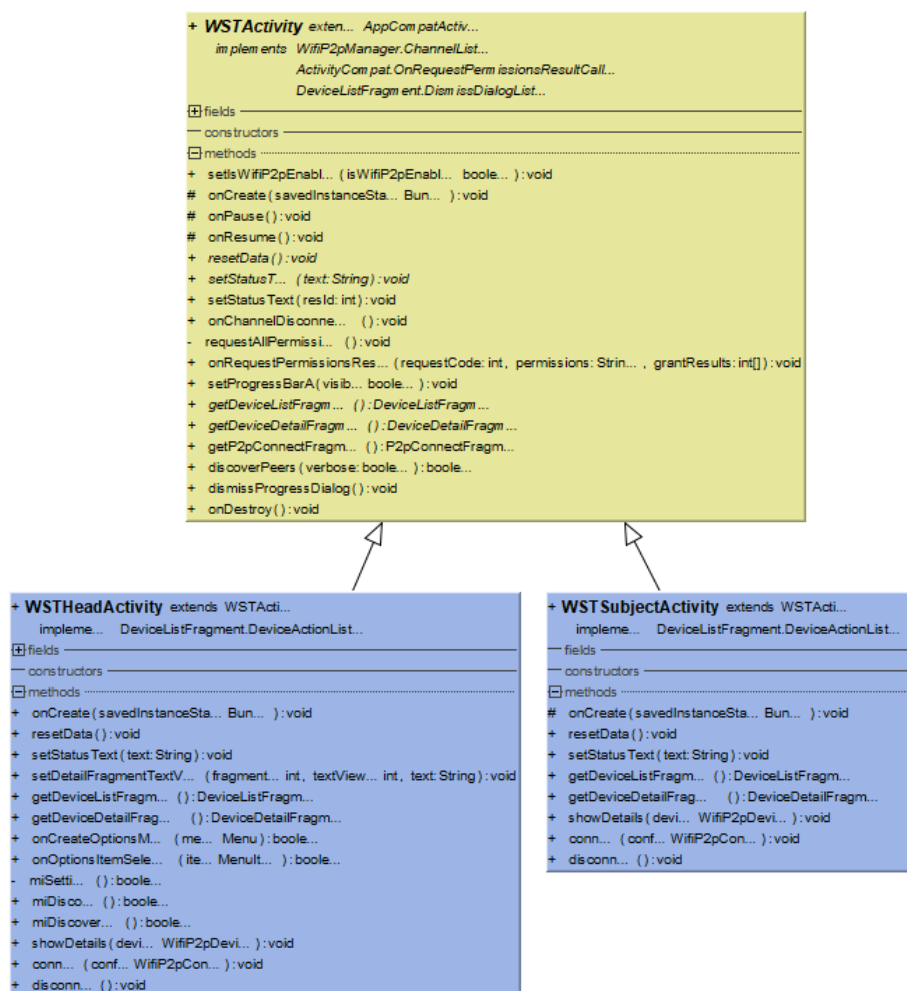
Poněkud nestandardně nevykresluje žádné rozvržení (layout), ale dědí od knihovny třídy `RxFragment.java` pro asynchronní operace. Jak je zřejmé z obr. 3.3 dědí od `P2pConnectFragment.java` ekvivalenty pro subjekty a pro administrátora. `P2pConnectHeadFragment.java` pro administrátora pomáhá se správou účastníků.

3.2.3 P8XXFragment

`P8XXFragment.java` je abstraktní třída, od které dědí třídy nebo v tomto případě fragmenty pro provádění testů. Pro každou testovou metodu musí vzniknout nový dědic jako ukazuje obr. 3.4. Stálo za zamyslení, které testové metody by se daly sloučit do jedné třídy a jednotlivé metody rozlišit určitou podmínkou. Avšak tato varianta se nehodí, pokud může vzniknout požadavek

⁵Každý účastník, který přijímá data se tedy stává TCP serverem. To platí i pro subjekty

3. IMPLEMENTACE



Obrázek 3.2: WSTActivity.java

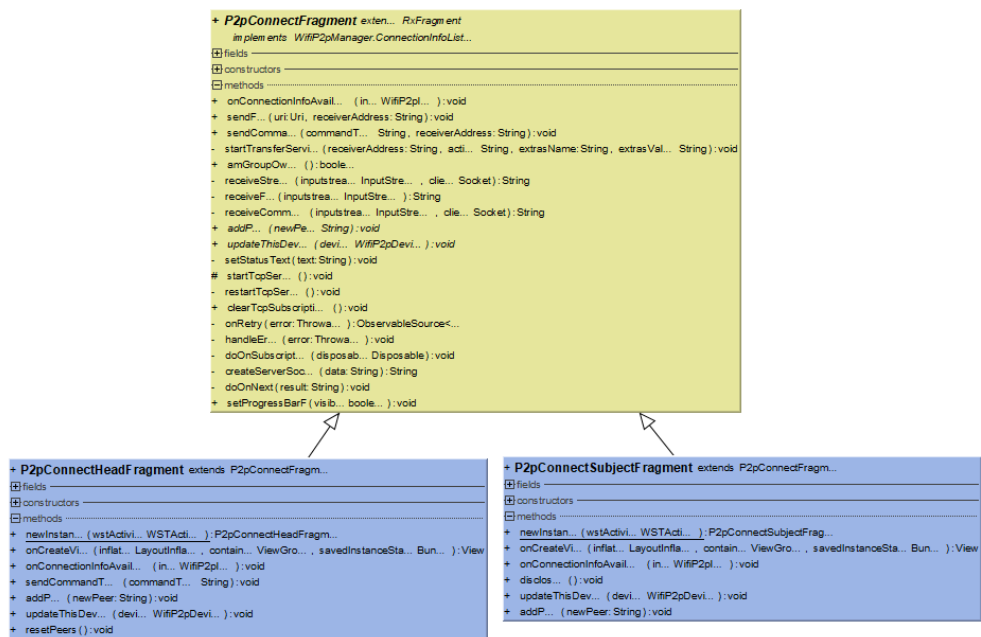
na změnu chování určité testové metody. Každá změna jediné testové metody by znamenala zajistit, aby další ze sloučených metod zůstala nezměněná a také by se musely testovat všechny dotčené metody.

Vzhledem k tomu, že třídy všech testových metod dědí od společné třídy, není velkou zátěží vyvíjet konkrétní implementace. To popisuje následující odstavec.

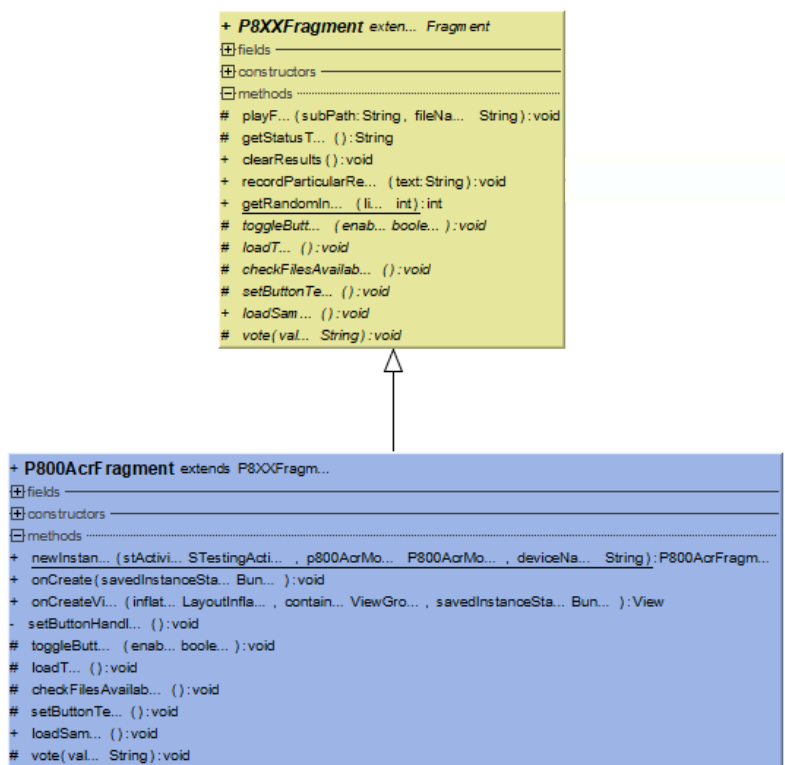
3.2.3.1 Přidání nové testové metody

Jak bylo zmíněno, je potřeba pro každou testovou metodu vytvořit samostatnou třídu typu .java, která dědí od P8XXFragment.java a také rozvržení (layout) podle vzoru ostatních rozvržení v podadresáři res-layout projektu. V dědící třídě je potřeba následně implementovat metody:

3.2. Struktura aplikace



Obrázek 3.3: P2pConnectFragment.java



Obrázek 3.4: P8XXFragment.java

3. IMPLEMENTACE

- newInstance
- onCreate
- onCreateView
- setButtonHandlers
- View.OnClickListener.onClick
- toggleButtons
- loadTest
- checkFilesAvailability
- setButtonTexts
- loadSample
- vote

Výsledek implementace

4.1 Uživatelské rozhraní

4.1.1 Test ACR podle doporučení P.800

Na obr. 4.1 je vidět ovládání pro test podle doporučení P.800 ACR.

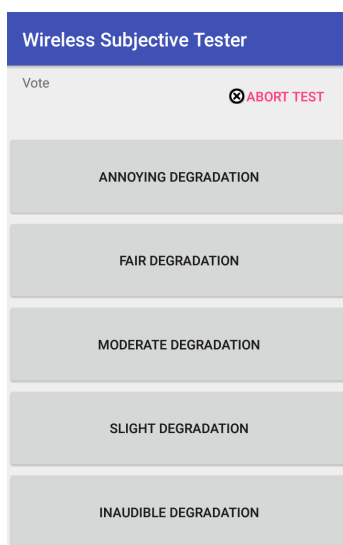
4.1.2 Test DCR podle doporučení P.800

Na obr. 4.2 je vidět ovládání pro test podle doporučení P.800 DCR.

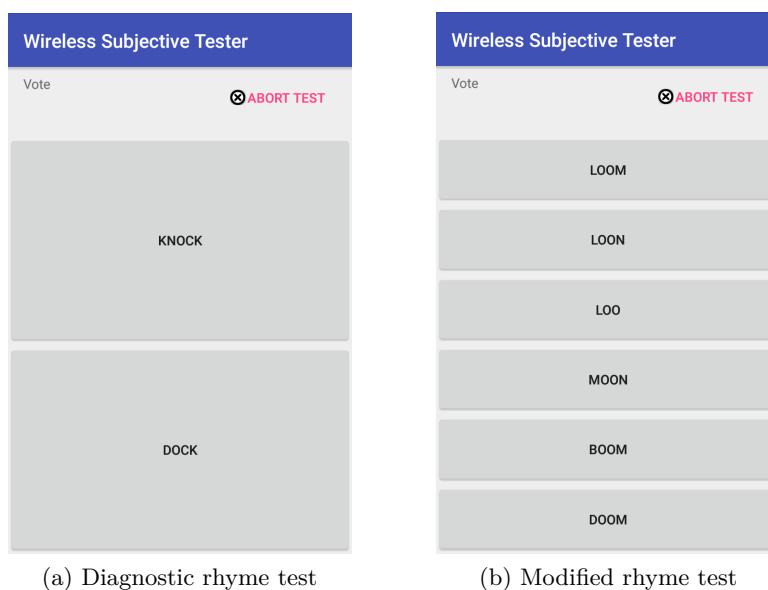


Obrázek 4.1: Test podle P.800 ACR

4. VÝSLEDEK IMPLEMENTACE



Obrázek 4.2: P.800 DCR – hlasování



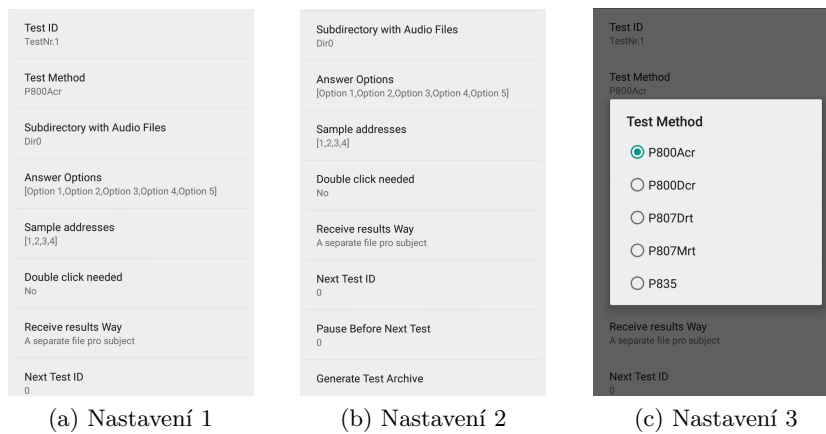
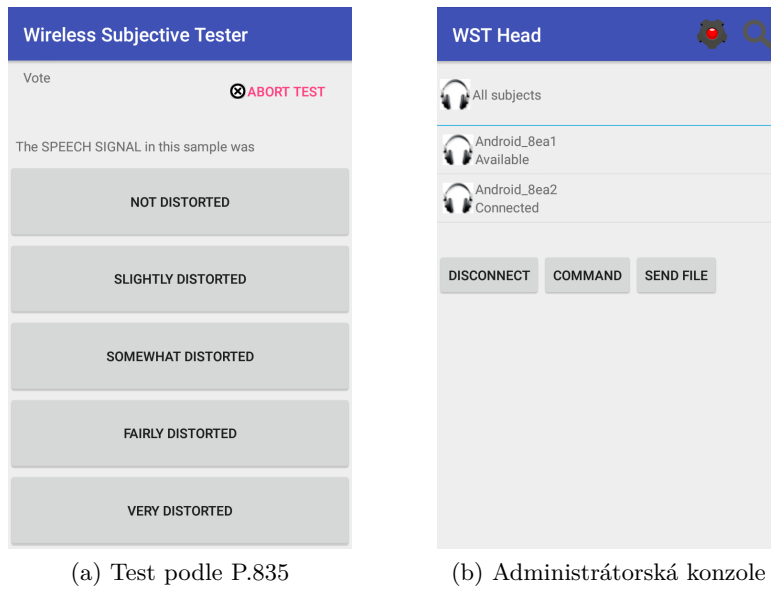
Obrázek 4.3: Testy podle P.807

4.1.3 Testy podle doporučení P.807

Na obr. 4.3 je vidět ovládání pro test podle doporučení P.807 DRT a MRT.

4.1.4 Test podle doporučení P.835

Na obr. 4.4a je vidět ovládání pro test podle doporučení P.835.



Obrázek 4.4: Nastavení administrátorské konzole

4.1.5 Činnost administrátora

Na obr. 4.4b je správa účastníků na zařízení administrátora.

4.1.6 Administrátorská nastavení

Na obr. 4.4 je vidět nastavení, které odpovídá 2.2.

Závěr

Výsledek práce

V rámci práce byla provedena implementace mobilní aplikace, která poskytuje platformu pro subjektivní testování. Ukázalo se, že testování pomocí aplikace je spolehlivé. Nebylo dosud dokončeno přenášení dat mezi subjektem a nadřazeným systémem. Předpokládá se dokončení funkcionality před obhajobou práce.

Zhodnocení použité technologie peer-to-peer

Technologie se během parciálních testů jevila jako velmi užitečná. Jakmile je navázáno spojení, je přenos dat spolehlivý. Nevýhoda je navazování spojení. Někdy se může stát, že spojení z aplikace nejde navázat, dokud se nerestartuje Wi-Fi v telefonu.

Zhodnocení randomizace vzorků

Generování náhodných čísel se ukázalo být velmi spolehlivé, skutečně byly generovány sekvence indexů vzorků náhodné a nikoliv pseudonáhodné.

Literatura

- [1] MediaPlayer. [cit. 2018-5-23]. Dostupné z: <https://developer.android.com/reference/android/media/MediaPlayer>
- [2] International Telecommunication Union: *Recommendation ITU-T P.800 [online]*. srpen 1996, [cit. 2018-5-23]. Dostupné z: <https://www.itu.int/rec/T-REC-P.800-199608-I/en>
- [3] International Telecommunication Union: *Recommendation ITU-T P.807 [online]*. únor 2016, [cit. 2018-5-23]. Dostupné z: <https://www.itu.int/rec/T-REC-P.807/en>
- [4] International Telecommunication Union: *Recommendation ITU-T P.835 [online]*. listopad 2003, [cit. 2018-5-23]. Dostupné z: <https://www.itu.int/rec/T-REC-P.835/en>
- [5] Aplikace pro poslechový test. [cit. 2018-5-23]. Dostupné z: <https://play.google.com/store/apps/details?id=mobile.eaudiologia>
- [6] Which phone has the best Audio? [cit. 2018-5-23]. Dostupné z: <https://www.androidauthority.com/best-of-android-2017-audio-820937>
- [7] Samsung Galaxy S6 review. [cit. 2018-5-23]. Dostupné z: https://www.gsmarena.com/samsung_galaxy_s6-review-1227p7.php
- [8] Wi-Fi Direct. [cit. 2018-5-23]. Dostupné z: <https://www.wi-fi.org/discover-wi-fi/wi-fi-direct>
- [9] Bluetooth Range. [cit. 2018-5-23]. Dostupné z: <http://www.bluaiir.pl/bluetooth-range>
- [10] Wi-Fi peer-to-peer overview. [cit. 2018-5-23]. Dostupné z: <https://developer.android.com/guide/topics/connectivity/wifip2p>
- [11] Introducing JSON. [cit. 2018-5-23]. Dostupné z: <http://www.json.org/>

LITERATURA

- [12] knihovna – GSON. [cit. 2018-5-23]. Dostupné z: <https://github.com/google/gson>
- [13] Wi-Fi Direct Demo. [cit. 2018-5-23]. Dostupné z: <https://github.com/Miserlou/Android-SDK-Samples/tree/master/WiFiDirectDemo>

Seznam použitých zkratek

ITU International Telegraph Union

ITU-T ITU Telecommunication Standardization Sector

ACR Absolute category rating

DCR Degradation category rating

DRT Diagnostic rhyme test

MRT Modified rhyme test

MOS Mean opinion score

SNR Signal-to-noise ratio

GUI Graphical user interface

XML Extensible markup language

JSON JavaScript object notation

Obsah přiloženého CD

	readme.txt	stručný popis obsahu CD
	apk	adresář s instalačním balíkem ve formátu APK
	src	
	impl	zdrojové kódy implementace
	text	text práce
	thesis.pdf	text práce ve formátu PDF