

**SLOVENSKÁ TECHNICKÁ UNIVERZITA V
BRATISLAVE**

Fakulta informatiky a informačných technológií

**VIZUALIZÁCIA
ZAZNAMENANÉHO POHLADU**

Bakalárska práca

Martin Cíváň

Vedúci práce: Ing. Jakub Šimko, PhD.

Marec 2019

Anotácia

Bakalárska práca sa zaoberá upravením existujúceho nástroja na vizualizáciu zaznamenaného pohľadu, tak aby bol použiteľný s existujúcou infraštruktúrou na FIIT.

Vyhlásenie

Čestne vyhlasujem, že som túto prácu vypracoval samostatne, na základe konzultácií a s použitím uvedenej literatúry.

V Bratislave, 8.12.2018

Martin Cíváň

Zadanie

ZADANIE BAKALÁRSKEHO PROJEKTU

Meno študenta: **Civáň Martin**
Študijný odbor: Informatika
Študijný program: Informatika
Názov projektu: **Vizualizácia zaznamenaného pohľadu**

Zadanie:
Sledovanie pohľadu (eye-tracking) je veľmi perspektívnou oblasťou, využívanou nielen na výskum ale aj praktický vývoj softvéru v rámci štúdií použiteľnosti. Vizualizácia dát sledovania pohľadu, napríklad postupností fixácií v tzv. cestách pohľadu (scanpaths), patrí medzi nevyhnutné techniky v tejto oblasti. Nástroje umožňujúce túto vizualizáciu existujú, no nevyhovujú špecifickým podmienkam infraštruktúry laboratórií UXI.

Analyzujte existujúce techniky a nástroje vizualizácie záznamov o pohľade. Vytvorte vizualizačný nástroj pre zobrazovanie záznamov sledovania pohľadu. Nástroj bude vedieť zobrazovať cesty pohľadu nad podkladovými stimulmi ako sú statické obrázky, videozáznamy obrazoviek či webových stránok. Nástroj navrhnete tak, aby bola jednoduchá jeho integrácia s existujúcou infraštruktúrou v laboratóriách UXI. Nástroj by mal byť orientovaný všeobecne (nemal by fungovať len pre špecifický typ experimentov). Vlastnosti nástroja vhodne overte a vaše riešenie zdokumentujte.

Práca musí obsahovať:

- Anotáciu v slovenskom a anglickom jazyku
- Analýzu problému
- Opis riešenia
- Zhodnotenie
- Technickú dokumentáciu
- Zoznam použitej literatúry
- Elektronické médium obsahujúce vytvorený produkt spolu s dokumentáciou

Miesto vypracovania: Ústav informatiky, informačných systémov a softvérového inžinierstva, FIIT STU, Bratislava

Vedúci projektu: Ing. Jakub Šimko, PhD.

Termín odovzdania práce v zimnom semestri : 11. 12. 2018

Termín odovzdania práce v letnom semestri : 7. 5. 2019

**SLOVENSKÁ TECHNICKÁ UNIVERZITA
V BRATISLAVE**
Fakulta Informatiky a Informačných technológií
Iľkovičova 2, 842 16 Bratislava 4
1

Bratislava 17. 9. 2018

prof. Ing. Pavol Návrat, PhD.
riaditeľ ÚISI

Obsah

1. Úvod	9
2. Analýza	10
2.1. Vizualizácia zaznamenaného pohľadu	10
2.2. Existujúce riešenia	10
2.3. UXI infraštruktúra	12
2.4. Ogama na zobrazovanie údajov z UXI Infraštruktúry	12
2.4.1. Základné vlastnosti	12
2.4.2. Úložisko údajov	12
2.4.3. Databázový model	13
2.4.4. Zhodnotenie databázového modelu	15
2.5. Import dát	15
2.5.1. Súradnice pohľadu a myši	15
2.5.2. Označenie záznamu	16
2.5.3. Načítavanie nahrávky obrazovky	16
3. Import z UXI Infraštruktúry	17
3.1. Špecifikácia	17
3.2. Návrh	17
3.3. Trieda importu z UXI Infraštruktúry	17
3.4. Pridané grafické rozhranie importu z UXI Infraštruktúry	18
3.5. Scenár importu	18
3.6. Overenie riešenia	19
Literatúra	20
Zoznam	21
A. Plán pokračovania práce	22

Zoznam obrázkov

2.1. Trasa pohľadu	11
2.2. Mapa pozornosti	11
3.1. Formulár importu	18

Zoznam tabuliek

3.1. Porovnanie spotreby pamäte pri použití rôznych JSON parserov na testovacom vstupe.	17
--	----

Kapitola 1

Úvod

Pri zobrazovaní zaznamenaného pohľadu sa používajú rôzne zobrazovacie techniky. Okrem rôznorodých čiarových, prípadne stĺpcových grafov, ktoré môžu zobrazovať rôzne číselné alebo štatistické metriky je na základné nahliadnutie do dát vhodné použiť zobrazovanie pohľadu priamo na stimule.

Väčšina nástrojov na vytvorenie stimulov experimentu a samotný záznam ponúka aj samotné vizualizačné možnosti. Na FIIT však máme laboratórium, v ktorom sa môže konať experiment až pre 20 účastníkov naraz. Na takúto úlohu nie sú bežné nástroje optimálne a preto bola vyvinutá UXI Infraštruktúra.¹ Tá slúži na roz distribuovanie zadania experimentu do počítačov účastníkov, na samotné zobrazenie stimulov a zaznamenanie meraných údajov. Po experimente namerané údaje zhromažďí na jedno miesto. Vizualizačný nástroj však nemá a zaznamenané údaje nie je možné nijako jednoducho vizualizovať.

V rámci tejto práce sme preto navrhli a doplnili možnosť importu údajov z UXI Infraštruktúry do existujúceho programu Ogama. Ogama je softvér s otvoreným zdrojovým kódom, ktorý slúži na návrh experimentu, záznam a vizualizáciu. Po rýchлом a jednoduchom nainportovaní údajov je teda možné použiť vizualizačný modul Ogamy na základné nahliadnutie do získaných údajov.

¹ <https://github.com/uxifiit/>

Kapitola 2

Analýza

2.1. Vizualizácia zaznamenaného pohľadu

Základnou používanou technikou na zobrazovanie zaznamenaného pohľadu priamo na stimule patrí zobrazovanie trasy pohľadu na dynamickom alebo statickom stimule. Ide o zobrazenie postupnosti fixácií pohľadu ktoré sú pospájané sakádami. Je z nej dobre vidieť presnú trasu pohľadu. Ukážka na Obrázok 2.1, "Trasa pohľadu"

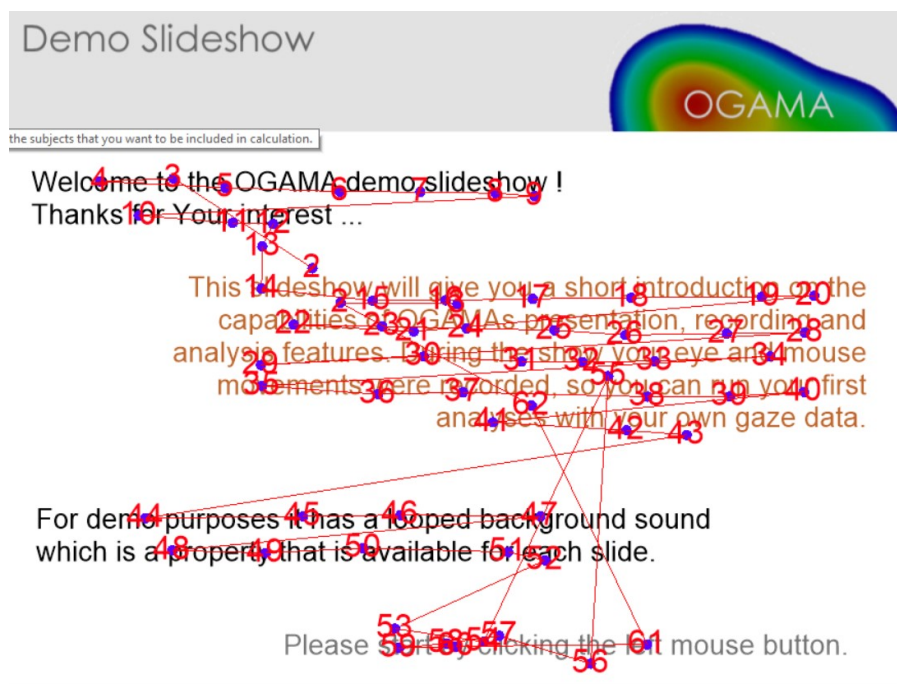
Pri viacerých účastníkoch experimentu je možné tiež tieto údaje agregovať do mapy pozornosti (angl. attention map). Z tejto mapy je zrejmé, v ktorých miestach bol pohľad najviac sústredený. Ukážka na Obrázok 2.2, "Mapa pozornosti" [1]

2.2. Existujúce riešenia

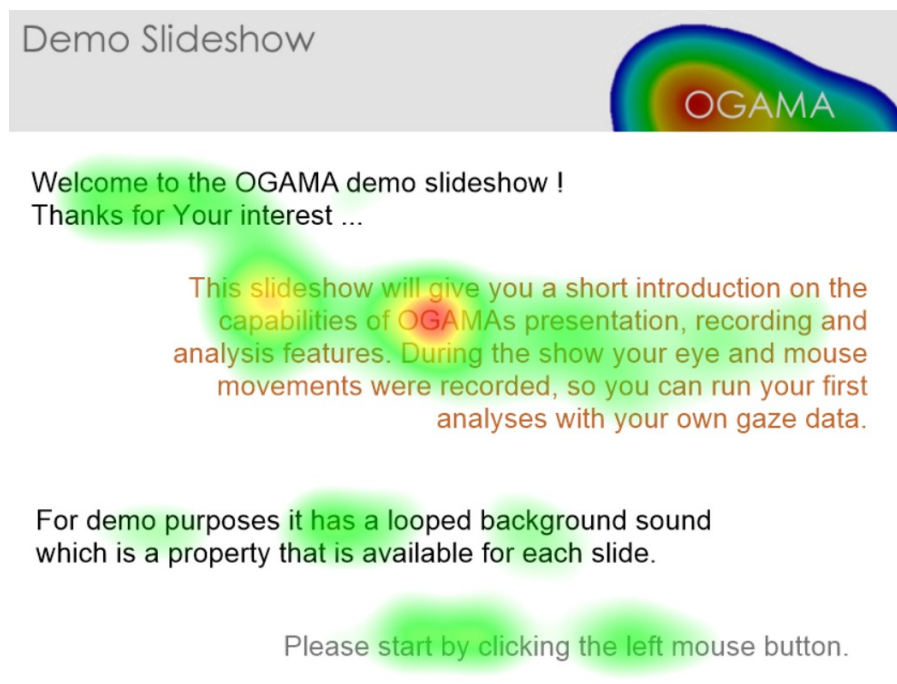
Key-trackerom od spoločnosti Tobii, ktoré sú momentálne na fakulte je na vytvorenie stimulov experimentu, záznam a vizualizáciu záznamu dodávaný softvér Tobii Studio. Ide o proprietárny softvér určený pre operačné systémy z rodiny Windows, ktorý podporuje najpoužívanejšie vyššie uvedené zobrazovacie techniky. Vie však zobrazovať len ním zaznamenané údaje bez nejakej možnosti importu už existujúcich údajov z iného zdroja.

Z riešení s otvoreným zdrojovým kódom stojí za zmienku Ogama.¹ Skladá sa z viacerých modulov: Dizajnovací modul je určený na vytvorenie stimulov a zostavenie experimentu. Podporuje rôzne scenáre, napríklad snímkovú prezentáciu, Flash video alebo len čistý záznam obrazovky. Záznamový modul umožňuje zobrazovanie stimulov, zaznamenávanie pohľadu, myši, prípadne obrazovky naraz jedného účastníka experimentu a následné vypočítanie a zápis údajov do databázy. Zaznamenané údaje sa dajú zobrazíť pomocou zobrazovacích modulov. Tieto moduly podporujú zobrazovanie postupnosti fixácií pohľadu a myši na stimuloch. Pri statických podkladoch vedú zobraziť aj tepelnú mapu. Taktiež je možné výsledok exportovať do videa v kontajneri AVI. V databázovom module sa dajú zobrazíť, prípadne aj upraviť hodnoty zaznamenaných údajov a taktiež je možný import údajov z CSV súboru.

¹ <http://www.ogama.net/> <https://github.com/avosskuehler/ogama/>



Obrázok 2.1. Zobrazenie trasy pohľadu (angl. scanpath) pomocou nástroja Ogama



Obrázok 2.2. Zobrazenie mapy pozornosti (angl. attention map) pomocou nástroja Ogama

2.3. UXI infraštruktúra

FIIT je vybavená laboratóriom s názvom UXClass pre skupinové eye-trackingové experimenty. Pre tieto účely bol vytvorený nástroj na riadenie takýchto experimentov s názvom UXI Infraštruktúra, ktorá sa skladá zo serverovej časti UXR na riadenie experimentu a klientskej časti UXC na záznam údajov od jednotlivých zúčastnených. Ide o aplikáciu založenú na technológii .NET, napísanú v jazyku C#, teda je určená pre operačné systémy rodiny Microsoft Windows. Zaznamenané údaje sú zhromaždené a uložené v JSON súboroch a záznam obrazovky, prípadne aj kamery, je uložený v kontajneri MP4. Aktuálne však neexistuje nástroj na plnohodnotné a pohodlné zobrazenie týchto údajov.

2.4. Ogama na zobrazovanie údajov z UXI Infraštruktúry

2.4.1. Základné vlastnosti

Ogama má základné požadované vizualizačné funkcie a má otvorený zdrojový kód preto je ju vhodné použiť ako základ riešenia. Ide o Windows Forms aplikáciu napísanú v jazyku C# s použitím rámca .NET verzie 4. Na zobrazovanie videa používa knižnicu DirectShow, na uskladnenie údajov slúži SQLite databáza. Vďaka rámcu .NET spolu s Windows Forms a DirectShow je momentálne možné túto aplikáciu natívne používať výhradne na operačných systémoch Microsoft Windows. To je síce nevýhoda ale nie je to zhoršenie oproti aktuálne používanému TobiiStudios, ktoré natívne podporuje tiež len operačný systém Windows.

2.4.2. Úložisko údajov

SQLite prináša so sebou výhody aj nevýhody - na používanie síce nie je potrebný špeciálny databázový server ale pri väčších meraniach to môže spôsobiť zvýšené hardvérové nároky na počítač. Z git repozitára je zrejmé, že v minulosti Ogama používala Microsoft SQL Server. Pre účely fakulty by centralizovaný databázový SQL server bol vhodnejší nielen kvôli nižším nárokom na počítač počas analyzovania ale aj kvôli lepším možnostiam zdieľania údajov medzi projektami. V prípade, keď sú údaje uložené na kvalitnom serveri s RAID-om tak sa používateľ nemusí priamo starať o ich zálohovanie. Uvažovali sme nad zmenou na multiplatformový server MySQL prípadne PostgreSQL ale zatiaľ sme sa rozhodli úložisko dať ponechať na SQLite z viacerých dôvodov: Zatiaľ nie je jasné, či sa Ogama uchyťí v praxi a teda sme si v rámci tejto práce určili ako prioritu základnú kompatibilitu, ktorou je možnosť jednoduchého importu dát a tiež nie je jasné, kto a ako by sa o prípadný server staral. SQLite momentálne nepredstavuje prekážku vo funkčnosti a keď sa naše riešenie osvedčí aj v praxi je možné ho kedykoľvek pomerne jednoducho upraviť aby pracovalo s iným SQL úložiskom.

2.4.3. Databázový model

Databáza programu Ogama obsahuje tabuľku AOIs, v ktorej sú uložené oblasti záujmu (angl. AOI - areas of interest). Každý záznam obsahuje nasledovné údaje:

- jedinečného číslo záznamu ID
- číslo stimulu TrialID
- číslo prezentácie SlideNr
- názov tvaru ShapeName
- typ útvaru ShapeType (napr. Rectangle, Polyline - pre obdĺžnik resp, mnohouholník pod.).
- počet vrcholov ShapeNumPts
- zoznam súradníc vrcholov útvaru ShapePts
- názov skupiny útvarov ShapeGroup

Surové údaje z eye-trackera a záznamu myši sú uložené v samostatnej tabuľke pre každého účastníka zvlášť. Po konzultácii s pôvodným autorom vysvitlo, že to tak bolo z výkonnostných dôvodov, ktoré už dnes síce nemusia byť relevantné, no nebol nikdy zistený výslovný dôvod na vynaloženie úsilia na zmenu tejto architektúry. Tabuľka má vždy názov zložený z meno účastníka a príponu Rawdata. Má nasledovné štruktúru:

- jedinečné číslo záznamu ID
- meno účastníka SubjectName
- číslo stimulu TrialSequence
- čas v mikrosekundách od začiatku záznamu Time
- X-ový priemer zreničky PupilDiaX
- Y-ový priemer zreničky PupilDiaY
- X-ová súradnica pohľadu GazePosX
- Y-ová súradnica pohľadu GazePosY
- X-ová súradnica myši MousePosX
- Y-ová súradnica myši MousePosY
- číslo udalosti, ak nejaká nastala EventId

Tabuľky GazeFixations a GazeFixations obsahujú zoznam fixácií pohľadu resp. myši a majú nasledovné stĺpce:

- jedinečné číslo záznamu ID
- meno účastníka SubjectName
- číslo stimulu TrialID
- číslo stimulu TrialSequence
- poradie fixácie v rámci stimulu CountInTrial
- čas začiatku fixácie od začiatku stimulu StartTime
- dĺžka trvania fixácie Length
- X-ová súradnica fixácie PosX
- Y-ová súradnica fixácie PosY

Ďalšie parametre môžu byť umiestnené v tabuľke Params so stĺpcami ID, Param, Description.

Skupiny útvarov sú vymenované v tabuľke ShapeGroups so stĺpcami ID a ShapeGroup.

Jednotliví účastníci experimentu sú uložený v tabuľke Subjects, okrem čísla účastníka a jeho mena SubjectName sú tam aj informácie ako vek Age, pohlavie Sex, dominantná ruka Handedness, zaradenie účastníka do kategórie Category a prípadný komentár Comments.

Pre ďalšie vlastnosti účastníkov je určená tabuľka SubjectParameters, kde je okrem čísla záznamu ID a mena SubjectName samotný názov vlastnosti Param a jej hodnota ParamValue.

Udalosti záznamu sú v tabuľke TrialEvents v stĺpcoch:

- ID - jedinečné číslo záznamu
- SubjectName - meno účastníka
- TrialSequence - číslo stimulu
- EventId - číslo udalosti pre daného participanta
- EventTime - Čas udalosti
- EventType - Kategória udalosti (napr. Mouse pre udalosti týkajúce sa kliknutí myši)
- EventTask - Typ udalosti (napr. Up, Down pre stlačenie resp. uvoľnenie tlačidla)

- EventParam - Ďalšie doplnujúce informácie o udalosti (napr. súradnice kliknutia myši)

Zoznam záznamov stimulov je v tabuľke Trials. Okrem čísla záznamu (ID), mena účastníka (SubjectName) a zatiaľ mne neznámeho prázdneho stĺpca EliminateData sú tam nasledovné hodnoty:

- TrialID - číslo stimulu
- TrialName - názov stimulu
- TrialSequence - číslo stimulu v poradí • Category - kategória stimulu
- TrialStartTime - čas začiatku stimulu
- Duration - dĺžka trvania stimulu

2.4.4. Zhodnotenie databázového modelu

Databázový model by v budúcnosti mohol byť upravený aby názov merania resp. meno účastníka mohlo obsahovať všetky znaky. Na referencovanie merania a taktiež názvy tabuliek by malo slúžiť jeho číslo resp. ID. Taktiež mať údaje rozdelené do viacerých tabuliek s rovnakou štruktúrou je v celku netradičné a ak by sa preukázalo, že to nemá pozitívny vplyv na výkon, ich zlúčenie by mohlo zlepšiť celkovú kvalitu modelu a zdrojového kódu aplikácie.

2.5. Import dát

Ide o upravený modul importu z CSV, ktorý bol pozmenený na čítanie z JSON súborov s automatickým priradením stĺpcov a potrebným predspracovaním údajov:

2.5.1. Súradnice pohľadu a myši

V Ogame sú súradnice pohľadu uložené ako 1 pár celých čísel X a Y, ktoré predstavujú pozíciu v pixloch na obrazovke. UX Infraštruktúra zaznamenáva normovanú pozíciu, čiže vo výstupe sú 2 čísla z intervalu 0; 1i pre každé oko. Transformácia spočíva vo vynásobení normovanej pozície nastaveným rozlíšením v projekte v Ogame. Výsledná súradnica vznikne agregovaním súradníc očí podľa nastavenia a to nasledovne: Buď je nastavený režim "priemerovania" kedy výsledné súradnice sú priemerom hodnoty z pravého a ľavého oka alebo je zvolené dominantné oko a je použitá súradnica len tohoto oka. V prípade, že v zázname je platná súradnica len jedného oka, v oboch prípadoch sa použije iba táto súradnica.

Ogame má časy jednotlivých záznamov uložené ako počet mikrosekúnd od začiatku nahrávania. UX Infraštruktúra ukladá ku každému záznamu časovú stopu aktuálneho času s mikrosekundami. Výsledný počet mikrosekúnd je vypočítaný odpočítaním časovej stopy začiatku nahrávania od časovej stopy záznamu.

Súradnice myši sú aj v Ogame aj v infraštruktúre v rovnakej forme (v pixloch), preto ich netreba transformovať.

2.5.2. Označenie záznamu

Ogama používa ako identifikátor merania meno účastníka ale keďže je tento názov použitý aj priamo v názve tabuľky v databáze tak nemôže obsahovať nepovolené znaky a začínať číslom. Ako názov teda bude použitý názov priečinku so súbormi, z ktorého budú odstránené nepovolené znaky. Ak boli odstránené všetky znaky, je používateľ upozornený a priečinok je preskočený.

2.5.3. Načítavanie nahrávky obrazovky

Načítavanie nahrávky obrazovky sa pri prehrávaní záznamu deje tak, že Ogame sa pozrie, či existuje .avi súbor s názvom participanta a číslom merania. Ak áno, tak ho použije ako podklad pod vizualizáciu záznamov. Na zobrazovanie nahrávky obrazovky sa v Ogame používa knižnica DirectShow s filtrom AVI Splitter , preto očakáva video v kontajneri AVI. Podporované kodeky videa závisia od aktuálne nainštalovaných kodekov na konkrétnej inštalácii Windowsu. Infraštruktúra bežne používa kontajner MP4 a kodek H264, konverzia do ostatných je pomerne jednoduchá pomocou knižnice FFMPEG.

Tá je použitá aj v UXI Infraštruktúre.

Kapitola 3

Import Z Uxi Infraštruktúry

3.1. Špecifikácia

Scenár importu by má pomocou pár jednoduchých krokov vedieť automaticky naimportovať zvolené údaje zaznamenané z Uxi Infraštruktúry. Mal by ho byť schopný vykonať aj úplný začiatok.

3.2. Návrh

Import z Uxi Infraštruktúry vychádza z možnosti importu z CSV súboru. Ten je vedený sprievodcom a teda aj import z Uxi Infraštruktúry má viacero krokov. Import z Uxi Infraštruktúry spúšťa pridané tlačidlo, ktoré je umiestnené vedľa tlačidla na import z CSV súboru.

3.3. Trieda importu z Uxi Infraštruktúry

Import z Uxi infraštruktúry je riešený ako statická trieda do databázového modulu a funguje na podobnom princípe ako existujúci import z CSV súboru. Najprv sa vytvorí celý JSON objekt so záznamom pohľadu. Potom sa prechádzajú jednotlivé záznamy v objekte a robia sa potrebné transformácie. Transformovanými údajmi sa následne naplnia objekty databázových riadkov z tabuľky surových dát. Následne sa rovnakým spôsobom spracujú záznamy o myši. Potom sa všetky riadky zapíšu do databázy a pokračuje vykonávanie, ktoré je prebraté z importu z CSV súborov, ktoré zahŕňa napríklad výpočet oblastí záujmov.

V spotrebe pamäte pri použití rôznych JSON parserov sa pomerne výrazné rozdiely, úplné porovnanie je v tabuľke Tabuľka 3.1, "Porovnanie spotreby pamäte pri použití rôznych JSON parserov na testovacom vstupe."

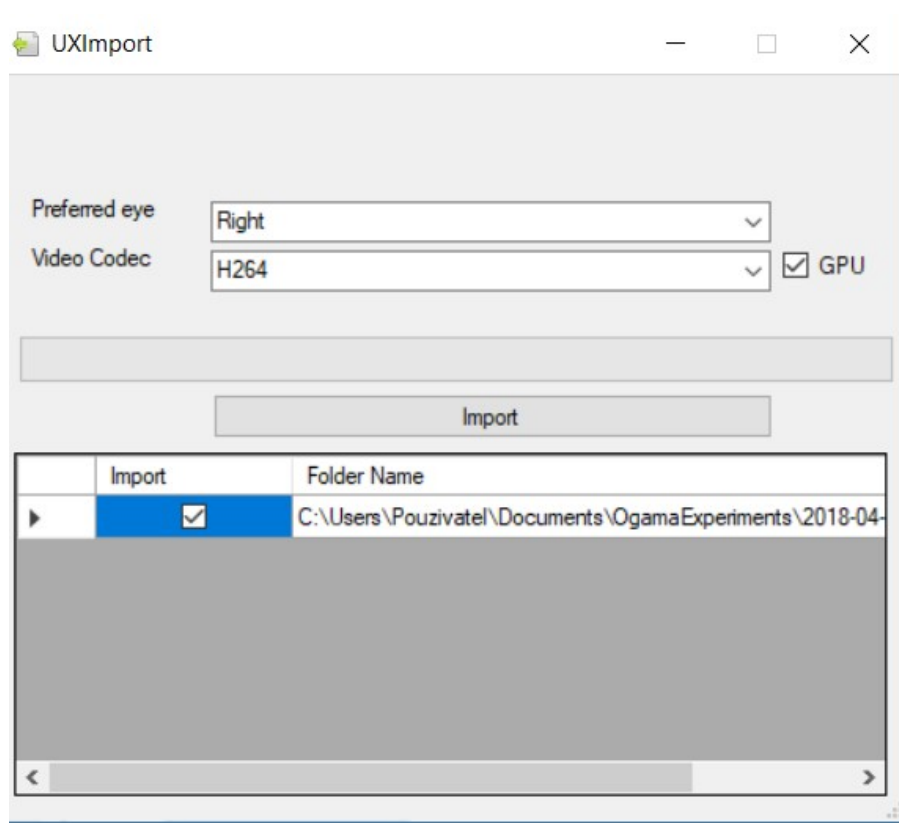
Tabuľka 3.1. Porovnanie spotreby pamäte pri použití rôznych JSON parserov na testovacom vstupe.

Parser	Spotreba pamäte
JavaScriptSerializer	~2GB
JSONSerializer	~3.5GB

Parser	Spotreba pamäte
JSONReader	~4GB

3.4. Pridané grafické rozhranie importu z UXI Infraštruktúry

Do okna databázového modulu bolo pridané tlačidlo UXImport, ktoré otvorí vyskakovacie okno prehliadača priečinkov. Po zvolení priečinku sa priečinok prehľadá a prehľadajú sa aj všetky priečinky priamo v ňom. Ak bola aspoň v jednom priečinku nájdená štruktúra súborov projektu UXI Infraštruktúry tak sa otvorí formulár s nastaveniami importu a zoznamom nájdených priečinkov, z ktorých je možný import vykonať. V opačnom prípade je používateľ o neúspechu informovaný vyskakovacím okienkom. Náhľad formuláru je na Obrázok 3.1, "Formulár importu".



Obrázok 3.1. Snímka formuláru nastavení importu

3.5. Scenár importu

Predpoklad: používateľ má v Ogame otvorený projekt, do ktorého chce údaje nainportovať a má otvorený databázový modul.

1. Používateľ klikne na tlačidlo UXImport

2. Aplikácia otvorí prehliadač priečinkov
3. Používateľ vyberie priečinkov, z ktorého chce naimportovať údaje
4. Aplikácia prehľadá priečinkov a zobrazí formulár nastavení importu spolu s výsledkom
5. Používateľ vyplní formulár a spustí import tlačidlom Import
6. Aplikácia vykoná import a po skončení informuje používateľa vyskakovacím okienkom
7. Používateľ uloží zmeny v databázovom module pomocou tlačidla s logom diskety
8. Prípád použitia končí.

3.6. Overenie riešenia

Riešenie som zatiaľ odskúšal na dvoch sadách údajov a ďalej sa ho chystám overiť tak, že oslovím pár ľudí, ktorý majú zaznamenané údaje UXI Infraštruktúrou a poprosím ich aby sa pokúsili ich údaje naimportovať a potom ich skúsili analyzovať. Okrem funkčnosti importu ma bude zaujímať ako sa s takouto úlohou budú vedieť vysporiadať úplne noví používatelia, ktorý Ogamu nikdy predtým nepoužili. Tiež aj ich názor na riešenie spolu s ich možnými návrhmi na zlepšenie.

Zhodnotenie Upravený existujúci nástroj vie poslúžiť na prvé nahliadnutie do zaznamenaných údajov a je možné získať lepší pohľad na zaznamenané údaje. Natívne funguje, podobne ako TobiiStudio, iba na Windowse avšak nové riešenie je zadarmo a umožňuje veľmi jednoducho pomocou niekoľkých kliknutí naimportovať údaje z UX Infraštruktúry. S importovanými údajmi je potom možné pracovať rovnako ako s údajmi, ktoré boli Ogamou priamo nahraté, napríklad je ich možné vizualizovať pomocou modulu na analýzu. Import videa zatiaľ nie je automatizovaný, to je potrebné skonvertovať do jedného z nainštalovaných kodekov v kontajneri AVI a nakopírovať do priečinka Thumbs v rámci projektu pod názvom vo formáte Meno Účastníka-0

Zatiaľ je ešte implementovaný iba import údajov o pohľade a myši ale plánujeme dorobiť aj import udalostí. Taktiež je v pláne analyzovať a prípadne aj implementovať export definície experimentu vo formáte UXI infraštruktúry.

Literatúra

- [1] Blascheck, T., Kurzhals, K., Raschke, M., Burch, M., Weiskopf, D., and Ertl, T. State-of-the-Art of Visualization for Eye Tracking Data. In EuroVis - STARs (2014), R. Borgo, R. Maciejewski, and I. Viola, Eds., The Eurographics Association.

Zoznam

D

Databázy

Microsoft SQL Server, 12

PostgreSQL, 12

SQLite, 12

K

Knižnice

.NET, 12

AVI Splitter, 16

DirectShow, 16

Windows Forms, 12

S

Softvér

Ogama, 10

UXI infraštruktúra, 12

V

Vizualizačné techniky

mapa pozornosti (angl. attention map),

10

trasa pohľadu, 10

Dodatok A

Plán Pokračovania Práce

1. Behom prvých týždňov mať dokončený automatizovaný import aj s videom a importom udalostí.
2. Následne otestovať riešenie s pár používateľmi
3. Analyzovať možnosti exportu definície stimulov experimentu pre UXI infraštruktúru aby bol využiteľý aj dizajnovací modul.
4. Podľa výsledku analýzy doimplementovať riešenie.
5. Pred koncom celkový produkt ešte raz otestovať s pár používateľmi
6. Dokončiť prácu.