

Slovenská technická univerzita v Bratislave
FAKULTA INFORMATIKY A INFORMAČNÝCH TECHNOLOGIÍ

FIIT-5208-5639

Bc. Peter Demčák

Metodika vyhodnocovania hier na základe implicitnej spätnej väzby

Diplomová práca

Študijný program: Informačné systémy

Študijný odbor: 9.2.6 Informačné systémy

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

Vedúci projektu: Ing. Jakub Šimko PhD.

Máj 2015

Anotácia

Slovenská technická univerzita v Bratislave

FAKULTA INFORMATIKY A INFORMAČNÝCH TECHNOLOGIÍ

Študijný program: Informačné systémy

Autor: Bc. Peter Demčák

Bakalárska práca: Metodika vyhodnocovania hier na základe implicitnej spätnej väzby

Vedúci bakalárskej práce: Ing. Jakub Šimko PhD.

Máj 2015

Každá hra musí byť ľahko naučiteľná, inak hrozí že si jej hráči rýchlo nájdu alternatívny spôsob zábavy. Je preto vo veľkom záujme herných dizajnérov zabezpečiť zlepšovanie naučiteľnosti vytváraných hier počas vývoja, aby aj komplexné hry mohli priniesť finálnym hráčom čo najvyššiu možnú mieru spokojnosti.

Hry sú veľmi špecifickým typom aplikácií, pretože ich cieľom je vytvorenie autentického herného zážitku. Toto je príčina, pre ktorú sa návrh hier líši od návrhu konvenčných aplikácií a vyhodnocovanie ich naučiteľnosti tiež vyžaduje individuálny výskum.

Sledovanie pohľadu je relatívne nová technológia schopná lokalizovať pozíciu ľudského pohľadu na obrazovke. Použitie tejto technológie sa ukazuje ako sľubné pre vyhodnocovanie používateľského zážitku.

V tejto práci predstavujeme metódu založenú na implicitnej spätnej väzbe získanej zo zariadenia na sledovanie pohľadu, ktorá slúži na vyhodnocovanie iniciálnej naučiteľnosti hier vo vzťahu s vrodenu a vynárajúcou hernou zložitou. Túto metódu ďalej aplikujeme na viaceré existujúce hry a o dosiahnutých výsledkoch ďalej diskutujeme.

Annotation

Slovak University of Technology Bratislava

FACULTY OF INFORMATICS AND INFORMATION TECHNOLOGIES

Degree Course: Information Systems

Author: Peter Demčák

Bachelor thesis: Methodics of Game Evaluation Based on Implicit Feedback

Supervisor: Ing. Jakub Šimko PhD.

May 2015

All games need to be easily learnable. Otherwise, their players will soon start looking for alternative means of entertainment. It is in the great interest of game designers to improve the learnability of their games while the game is still in development, so that even the complex games can be fully enjoyed by their endpoint players.

Games are a very specific type of application, because their objective is to create an authentic game experience. This is why the design of games differs from conventional applications, and why the evaluation of their learnability requires individual research.

Eye-tracking is a relatively new technology capable of locating the position of human gaze on the screen. The use of this technology shows great promise for evaluation of user experience.

In this work, we propose a method based on implicit feedback from an eye-tracking device, which is used for evaluation of the initial learnability of games in relation to the game's innate and emergent complexity. We then apply this method to several existing games, and discuss the achieved results.

Obsah

1 Úvod	1
2 Herná naučiteľnosť, a prečo na nej záleží	3
2.1 Čo tvorí hru?	4
2.2 Charakteristiky hier a herného zážitku	5
2.3 Zložitosť a naučiteľnosť v kontexte hier	6
2.3.1 Herná zložitosť	6
2.3.2 Naučiteľnosť	7
2.3.3 Herná naučiteľnosť aj jej vzťah k hernej zložitosti	9
2.4 Vyhodnocovanie vlastností hier	10
2.4.1 Playtesting	11
2.4.2 Expertná metóda	11
2.4.3 Dotazník o ponorení	12
2.5 Zhrnutie	12
3 Implicitná spätná väzba a jej využitie	15
3.1 Sledovanie pohľadu	16
3.2 Metriky sledovania pohľadu	17
3.3 Analýza príbuzných experimentov založených na sledovaní pohľadu	18
3.3.1 Eivazi - 8-hlavoľam	18
3.3.2 Bartels - AMBR	18
3.3.3 Sundstedt - Labyrint	19
3.3.4 Pretorius - Network Management	20
3.3.5 Kickmeier-Rust - 80Days	20
3.4 Zhrnutie	21
4 LCM metóda	23
4.1 Zaradenie do vývojového procesu	23
4.2 Výzvy a obmedzenia	23
4.3 Prípady naučenia	24
4.4 Algoritmus porovnávania prípadov naučenia s nameranými údajmi	26
4.5 Spojenie výsledkov vyhodnocovania od viacerých používateľov	29
4.6 Výber účastníkov pre meranie naučiteľnosti	30
5 Implementácia nástroja na automatizované vyhodnocovanie naučiteľnosti	33
5.1 Nástroj Geddit	33

5.2 Tobii Studio	34
5.3 Vstupné údaje metódy	35
5.4 Spracovanie vstupných údajov	38
5.5 Výpočet metrik podobnosti	38
6 Overenie LCM metódy.....	41
6.1 Experiment.....	41
6.2 Vyhodnotenie výsledkov metódy	46
6.3 Porovnanie s explicitnou spätnou väzbou a diskusia.....	51
7 Zhodnotenie.....	55
Bibliografia.....	57

Prílohy

Príloha A – Technická dokumentácia

Príloha B – Príspevok na IIT.SRC 2015

Príloha C – Plagát na IIT.SRC 2015

Príloha D – Protokol experimentu

Príloha E – Obsah elektronického média

1 Úvod

Pre všetky aplikácie používané ľuďmi je dôležitá ich použiteľnosť. Pokiaľ aplikácia dokáže vykonávať všetku potrebnú funkcionálnosť, ale nikto ju nedokáže reálne použiť, táto aplikácia nemá prakticky žiaden reálny význam. Špeciálnu rolu má v použiteľnosti naučiteľnosť. Teda miera, do ktorej sa dokážu noví používatelia naučiť aplikáciu používať. Zvlášť akútnu výzvu predstavuje naučiteľnosť pre dnes bežný komerčný softvér, pri ktorom hrozí, že keď sa s ním používateľ nedokáže rýchlo naučiť pracovať, používateľ to jednoducho vzdá a nájde si iný softvér, s možno aj horšou funkcionálnosťou, ale lepšou naučiteľnosťou.

Typickým príkladom aplikácií pre ktoré je naučiteľnosť kritická sú hry. Pokiaľ hráč nedokáže hru dostatočne rýchlo pochopiť, stráca sa v jej herných prvkoch, a ešte k tomu sa v hre začnú z jeho pohľadu diať udalosti, ktorým nerozumie - zvlášť ak kvôli tomu začne prehrávať - hráč si veľmi rýchlo začne hľadať alternatívnu formu zábavy.

Trh z herným softvérom je v súčasnej dobe masívny, no zároveň preukazuje vysokú mieru podpory pre novú inovatívnu tvorbu. Nové hry však pre formovanie inovatívnych herných zážitkov a pre následne úspešné presadenie sa na trhu potrebujú poznať a odvracať riziká pre svoj herný návrh. Inak sa ľahko môže stať, že hra so zaujímavou myšlienkou zlyhá len na ľahko vyvrátiteľných chybách pri jej realizácii a testovaní.

Hry sú v súčasnosti sľubnou a popritom ešte nie úplne preskúmanou témou pre výskum. Štúdium hier sa okrem poznatkov z informačných technológií opiera o vedomosti z mnohých vedných aj rôznych profesijných oblastí. Samostatnou témou pre výskum je aj sila hier pre motiváciu menej zábavných aktivít, ktorú tiež nie je vhodné podceňovať.

Je prirodzené, že všetky spoločnosti - či jednotlivci - zaoberajúce sa dizajnom hier disponujú prostriedkami na overenie, či je v schopnostiach nových hráčov naučiť sa ich hru hrať. Všetky existujúce prístupy majú však v súčasnosti svoje nedostatky. Menovite tieto nedostatky vychádzajú z ich podstaty, že sú buď založené na rôznych heuristikách pre hry ktoré sú vyhodnocované expertmi, alebo na explicitnej spätnej väzbe od hráčov. Heuristiky vyžadujú bohaté skúsenosti expertov na ich efektívne uplatnenie, a aj tak nemusia odhaliť všetky problémy. Problémom explicitnej spätnej väzby je, že hráči často nevedia presne definovať svoje prežívanie v priebehu hry, a už vonkoncom nevedia s presnosťou na sekundy určiť, kedy boli v hre zmätení, a presne ukázať ktorý prvok hry u nich vyvolal, že ich hra nakoniec prestala baviť.

Alternatívnym prístupom, ktorý je opačný proti explicitnej spätnej väzbe, je spätná väzba implicitná. Pri implicitnej spätnej väzbe sú údaje o správaní hráča zbierané v priebehu interakcie s produktom bez používateľovho vedomia. Na konci testovacieho procesu môžu vlastníci produktu aplikovaním interpretačných metód extrahovať z týchto údajov informácie o vnútornom prežívaní používateľa. Tie je už ďalej možné premeniť na poznatky o tom, akým spôsobom sa dá daný produkt - alebo v našom prípade hra - zlepšiť.

Jednou z moderných technológií, ktoré ukazujú vysoký prísľub pre zber údajov o implicitnej spätnej väzbe, je takzvané sledovanie pohľadu. Tieto relatívne nové technológie - umožňujúce

určiť pozíciu bodu, na ktorý sa používateľ pri používaní aplikácie pozerá - sa stávajú sa čoraz dostupnejšími aj pre bežné použitie. So sledovaním pohľadu a vhodnými podpornými nástrojmi môžeme na video záberoch sledovať, ktoré súčasti rozhrania používateľa najviac zaujali, ktoré si nevšimol vôbec a podobne. Nás však zaujímajú aj ďalšie možnosti využitia údajov, ktoré sledovanie pohľadu v surovej forme poskytuje, ako aj návrh metód na ich ďalšiu automatickú interpretáciu. Návrh jednej takejto metódy - konkrétne určenej na vyhodnocovanie naučiteľnosti hier na základe údajov zo sledovania pohľadu a iných ukazovateľov implicitnej spätnej väzby - je výzva, ktorou sa zaoberáme v tejto práci.

Naším cieľom je návrh metódy, na základe ktorej je možné vyhodnotiť úroveň iniciálnej naučiteľnosti hier. Túto metódu potom experimentálne overujeme na súbore rôznych reálnych hier, a ďalej diskutujeme o jej výsledkoch, a navrhujeme spôsoby jej ďalšieho zlepšenia

Na začiatku práce sa venujeme analýze problémovej oblasti. V nasledujúcej kapitole najprv analyzujeme problematiku hier. Definujeme v nej, čo robí hry odlišnými od bežných typov aplikácií. Definujeme si hernú zložitost', bližšie si popisujeme význam naučiteľnosti v kontexte softvéru, a následne analyzujeme aplikáciu tohto konceptu do domény hier. Posledná časť kapitoly je venovaná overovaniu hier - existujúcim metodológiam a technikám - aby sme získali prehľad o reálnom veľkom obraze, do ktorého plánujeme našu metódu začleniť. Analýza z tejto kapitoly kladie teoretický základ pre celkové fungovanie našej metódy.

V tretej kapitole sa venujeme podrobnej analýze implicitnej spätnej väzby s upretím sa na technológie sledovania pohľadu - ako tieto technológie fungujú, čo znamenajú jednotlivé pojmy zaužívané v tejto oblasti, aké metriky založené na sledovaní pohľadu môžeme ďalej použiť, a aký je ich význam. V závere kapitoly prikladáme súhrn príkladov z niekoľkých relevantných experimentov, ktoré boli vykonané inými výskumníkmi v oblasti sledovania pohľadu.

Štvrtá kapitola je venovaná nami navrhovanej metóde dolovania prípadov naučenia (ďalej LCM metóda). Okrem definície samotnej metódy a popisu jej fungovanie v tejto kapitole rozvíjame aj spôsoby jej uplatnenia v procese vývoja hier.

Piata kapitola je venovaná implementácií nástroja Geddit, ktorý sme implementovali aby sme mohli automatizovane aplikovať našu metódu a vykonať jej experimentálne overenie. Táto kapitola zbežne popisuje niektoré dôležité implementačné súčasti našej metódy a definuje pre svojich používateľov formu dát, a používateľské rozhranie, ktoré aplikácia predpisuje.

V šiestej kapitole popisujeme experiment ktorý sme vykonali na overenie LCM metódy. Vyhodnocujeme v nej výsledky, ktoré sme našou metódou dosiahli, ďalej o nich diskutujeme a porovnáваме ich s výsledkami bežne zaužívaných techník na vyhodnocovanie hier

2 Herná naučiteľnosť, a prečo na nej záleží

Hry sú pojmom, ktorý nemá svoj pôvod v softvérovom inžinierstve, a dokonca ho ešte historicky ďaleko predchádza. My sa v tejto práci budeme venovať počítačovým hrám, videohram, hernému softvéru; jednoducho hrám ktoré sú prakticky a fyzicky - ak sa to dá pri softvéri tak povedať - realizované ako softvér, a pre jednoduchosť ich budeme nazývať pojmom hry.

Hry je ťažké definovať, hoci podvedome majú všetci ľudia na prvý pohľad predstavu, čo hra je, a čo hra nie je - aspoň kým s touto otázkou nie sú priamo skonfrontovaní. Rôzne definície sa sústreďujú na hry ako interakciu, hry ako výzvu, niečo čo sa dá vyhrať alebo prehrať, niečo čo robím z vlastnej vôle a pre dosiahnutie cieľov, ktoré majú význam len v kontexte hry, bez nejakých ďalších vonkajších motivátorov. Hra bola veľmi elegantne zadefinovaná Schellom nasledovne [1]:

"Hra je aktivita riešiaci problém s hravým prístupom"

V tejto definícii je hneď na prvý pohľad kľúčový práve pojem "hravý prístup". Ako je vôbec možné, že riešenie problémov - niečo čo samo o sebe nesie negatívnu konotáciu - sa za určitých podmienok môže stať pre ľudský mozog príjemným alebo dokonca želaným zážitkom? Ako je možné tento prístup docieľiť aj pre produktívnu činnosť? Toto je jedna z hlavných príčin, prečo sú hry také fascinujúce pre výskum.

Ako jedinečné médium, ktoré umožňuje vzájomnú interakciu medzi sebou a človekom, sú hry predmetom štúdie mnohých vedných disciplín – psychológie, pedagogiky, literatúry, kognitívnej vedy [2] - a s príchodom informačných technológií a hier sprostredkovaných skrze počítače sa hry v posledných dekádach stávajú doménou vied ako je informatika alebo umelá inteligencia. Informačné technológie poskytli hrám vo všeobecnosti prístup k prostriedkom, ktoré otvorili nové možnosti pre dizajn a tvorbu úplne nových typov hier.

Moderný herný softvér prešiel od svojich skromných začiatkov v hrách ako Pong, Pacman alebo Rogue ďalekú cestu. Herný priemysel sa po spamätaní zo severoamerického pádu v roku 1983 [3] preukázal ako plnohodnotná a trvalá súčasť zábavného priemyslu, schopný konkurovať filmom s titulmi ako Grand Theft Auto V¹, ktoré lámu všetky ziskové rekordy v celej histórii zábavného priemyslu ako takého. Profil hráča počítačových hier už dávno nie je len titulom obmedzenej a spoločensky utiahnutej skupiny ľudí, nakoľko hráčmi sa už dnes dá nazvať veľká časť vlastníkov mobilných telefónov, a hry orientované na široké demografické skupiny poskytujú interaktívnu zábavu celému obyvateľstvu, ktoré je čoraz viac priateľské k tomuto novému a rozširujúcemu sa zábavnému médiu.

Okrem zábavy sú však hry zaujímavé aj z hľadiska ich aplikácie pre motiváciu produktívnej činnosti, nakoľko hráči sú schopní kvôli endogénnej hodnote ktorú hra vytvára do hry investovať vysoké množstvo energie, ktorú by inak neboli ochotní vynaložiť, alebo by ju boli ochotní vynaložiť len za vysokú, napríklad finančnú odmenu. Tieto tzv. seriózne hry sú preto

¹ <http://www.forbes.com/sites/erikkain/2013/09/20/grand-theft-auto-v-crosses-1b-in-sales-biggest-entertainment-launch-in-history/>

veľmi silným nástrojom, ktorý dokáže umožniť dosiahnutie rôznych aj ťažko dosiahnuteľných cieľov za spokojnosti všetkých zúčastnených, a ešte dosiahnuť aj lepšie výsledky [4].

Príkladom seriózných hier sú napríklad edukačné hry, ktoré sa podľa výskumov ukazujú ako efektívna pomôcka na sprostredkovanie vzdelávania [5], alebo tzv. hry s účelom, ktoré zabezpečujú spracovanie informácií vyžadujúce ľudskú inteligenciu (napr. získavanie metadát [6] alebo rozoznávanie textu v obrázkoch).

Vráťme sa však teraz od našej motivácie späť k predmetu našej práce. Naším cieľom je predstaviť metódu, ktorá herným vývojárom pomôže určiť, či hráči ich hru zvládajú, alebo nie. Pre návrh tejto metódy sme si najprv museli položiť otázku: Čo to vlastne znamená, že hráč hru zvláda? Na jej zodpovedanie sa obraciame na teoretické znalosti z oblasti hier. Aké sú úrovne interakcie hráča s hrou? Aký je súvis medzi hrami a naučiteľnosťou z oblasti softvéru? Ako súvisí naučiteľnosť v hrách s náročnosťou hry? A napokon: ako dokážeme tieto poznatky aplikovať na softvérové riešenie vyhodnocovania hernej naučiteľnosti? Analýza problémovej oblasti v nasledujúcich sekciách ďalej rozvíja práve tieto základné myšlienky.

2.1 Čo tvorí hru?

Pokiaľ chceme vyhodnocovať, či sa hráč naučil hrať hru, musíme si najskôr vedieť určiť, čo „naučenie sa hry“ vôbec znamená. Znamená to, že hráč vie, kde na obrazovke sa nachádzajú prvky herného rozhrania? Znamená to, že vedel splniť isté ciele ktoré sme mu my v rámci hry stanovili? Alebo záleží iba na fakte, či si hráč hru užil, a teda zjavne na vyššej abstraktnej rovine pochopil všetko to, čo potreboval? Je zjavné, že na to, aby sme sa v návrhu našej metódy pohli ďalej, potrebujeme si určiť pojmy na označenie javov v hrách, ktoré sa týkajú interakcie hráča s hrou.

Kvôli rázne subjektívnemu alebo niekedy až umeleckému charakteru hier je náročnou úlohou hry v plnej miere presne formalizovať, stanoviť ich stavebné komponenty, a na ich základe hry ďalej analyzovať. V oblasti návrhu a výskumu hier však prirodzene existuje už niekoľko alternatívnych modelov hier. O jeden z nich sa budeme v tejto práci aj opierať, pre jeho výraznú pochopiteľnosť a flexibilitu.

MDA rámec popísaný Hunicke et al. [7] delí hry na tri úrovne:

- **mechanika**
- **dynamika**
- **estetika**

Mechanika predstavuje všetky akcie, správania a ovládacie prvky, ktoré hráč v rámci hry môže používať. Príklad: *kliknutím na tlačidlo sa začne stavať dom*

Dynamika je úroveň medzi mechanikou a estetikou, ktorá slúži na formáciu herných zážitkov, a je zložitejšou časťou herného návrhu na ktorú má herný dizajnér dosah. Príklad: *stavaním budov vzrastá v meste populácia, ktorú treba uživiť*

Estetika je úroveň abstraktná, ktorá je definovaná na úrovni zážitkov, ktoré hráč počas hrania hry prežíva. Hunicke sa vyhýba pojmom ako zábava alebo *hranie* (en. *Game Play*, pojem nemá

jednoznačný slovenský preklad, ale znamená princípy interakcie hráča s hrou) tým, že definuje osem typov estetík, ktoré môže hráč počas hry zažívať, alebo ich od môže hry očakávať: cítenie (uspokojenie zmyslov), fantázia, rozprávanie, výzva, spoločenstvo (sociálny rámec), objavovanie, sebavyjadrenie a poddanie sa (trávenie času). Príklad: *výzva spôsobená potrebou uživiť populáciu*

2.2 Charakteristiky hier a herného zážitku

Jednou zo základných charakteristík hier, ktoré odlišujú herný zážitok od zážitku z používania odlišných typov softvérových aplikácií, je ich schopnosť "vtiahnuť" do seba ľudí. Herné ponorenie (en. *immersion*) je vlastnosť ktorú sa mnoho hier snaží s rôznym stupňom úspechu docieľiť. Hráči, herne orientované média aj herní dizajnéri pojem ponorenia bežne používajú pri popisovaní a recenzovaní hier, a v širokom zmysle mu každý v dostatočnej miere chápe, avšak je ťažké presne určiť jeho definíciu.

Jennett et al. [8] zhŕňajú viacero psychologických štúdií pre vysvetlenie fenoménu ponorenia, spolu s ďalšími pojmami, ktorými je možné vysvetliť fenomény, ktoré hráči pri hraní hier prežívajú.

Ponorenie do hry je možné definovať ako stupeň, v ktorom sú hráči do hry zaangažovaní. Existujú tri stupne ponorenia, pričom pri každom stupni je ponorenie vyššie.

1. **zapojenie** (en. *engagement*)
2. **zaujatie** (en. *engrossment*)
3. **totálne ponorenie** (en. *total immersion*)

Hráč v prvom stupni ponorenia pod názvom **zapojenie** je hrou v dostatočnej miere zaujatý, a je ochotný hre venovať svoj ďalší čas a energiu. Dosiahnutie stupňa zapojenia je však silno závislé na hráčových individuálnych herných preferenciách, a tiež na zvládnutí ovládacích prvkov hry. Bez zvládnutia ovládania hráč nedokáže dosiahnuť žiaden stupeň ponorenia, pretože sa musí neustále sústrediť na ovládacie prvky. Ovládanie hry by malo byť preto maximálne robustné, silné a neviditeľné, pretože pri jeho zlyhaní sa stratí akákoľvek nahromadená úroveň ponorenia, čo môže byť pre hráčov predtým dosahujúcich vyššie úrovne zvlášť frustrujúce.

Ďalej môže hra u hráčov svojou celkovou kvalitnou konštrukciou docieľiť stupeň **zaujatia**. Toto je stupeň ponorenia do hry, v ktorom hráč vníma hru už nielen ako hru, ale ako hru ktorú považuje za "dobrú". Ďalej tu narastá stupeň zaangažovanosti, hráč je na hre silno emocionálne zúčastnený, venuje jej všetku svoju pozornosť a úplne sa stráca bariéra medzi hráčom a hrou, čo sa vyznačuje tým, že hráč ovláda herné rozhranie a nemusí veľmi premýšľať pri jeho používaní.

Najvyšším stupňom ponorenia je **totálne ponorenie**, pri ktorom hráč dostáva do stavu v ktorom stráca pojem o svojom okolí, a pri narušení môže byť dezorientovaný.

Extrémnym prípadom pri dosiahnutí totálneho ponorenia je, keď pri niektorých hrách dôjde k tomu, že hráči zažívajú stav **sprítomnenia** (en. *presence*), čo je stav, pri ktorom majú hráči psychologický pocit fyzického bytia vo virtuálnom prostredí hry. Pravdepodobnosť výskytu stavu presencie vzrastá, pokiaľ používateľ vníma reakcie hry na svoje správanie ako prirodzené

a zákonité. Rozdiel medzi sprítomnením a totálnym ponorením je ten, že sprítomnenie je mentálny stav, totálne ponorenie je zážitok existujúci v čase.

Sprítomnenie nie je možné dosiahnuť v každej hre. Napríklad v hre Tetris je ťažko možné preniesť sa do jej virtuálneho sveta. Pre hry takéhoto typu je však tiež možné dosiahnutím stupňa totálneho ponorenia docieľiť iný extrémny duševný stav, pri ktorom je hráč natoľko zaangažovaný do hernej aktivity, že mu na ničom inom nezáleží. Tento stav sa zvykne nazývať **prúd** (en. *flow*) [9]. Pojem prúdu nepochádza natívne z domény hier, keďže sa bežne zvykne spájať so stavom extrémneho pohrúženia do riešenia nejakého problému vo všeobecnosti.

Pre lepšie umiestnenie konceptu prúdu do hier Sweetser a Wyeth [10] navrhujú pojem **GameFlow**. *GameFlow* je zložený z viacerých elementov, z ktorých jedným je ponorenie, definované konfliktne s predchádzajúcou definíciou ako efektívna strata pojmu o sebe a o čase, pričom niektoré charakteristiky ponorenia ako sústredenie na plnenie úlohy a kontrola nad hrou sú uvedené ako samostatné elementy. Okrem ponorenia *GameFlow* obsahuje aj odlišné prvky ako napríklad hráčske schopnosti, jasné ciele, alebo sociálna interakcia.

Hry a herný zážitok, ktorý u používateľov hraním vyvolávajú, sa odlišujú od zážitku, ktorý poskytujú ostatné typy bežného softvéru. Cieľom bežného softvéru je totiž vo vysokej miere efektívnosť, jednoduchosť a minimálna chybovosť, zatiaľ čo cieľom hier je v prvom rade radosť používateľa [11]. Hry už zo svojej podstaty ako zábavné médium navodzujú u hráčov úplne odlišné správanie než odlišné typy aplikácií, a preto vyžadujú ďalšie samostatné štúdium.

Ponorenie ako charakteristika herného zážitku je silným ukazovateľom toho, nakoľko hráča hra zaujala. Ako sme tiež poukázali vymenovaním rôznych stupňov ponorenia, pre zvyšovanie úrovne ponorenia je dôležité odstránenie bariéry medzi hráčom a hrou, v čom zohráva kľúčovú rolu práve naučiteľnosť hry. V nasledujúcej sekcii sa venujeme už konkrétne pojmu naučiteľnosti v kontexte hier, čím sa viac približujeme k nášmu konceptu metodického vyhodnocovania hernej naučiteľnosti.

2.3 Zložitosť a naučiteľnosť v kontexte hier

2.3.1 Herná zložitosť

Jednou zo základných vlastností hier je ich zložitosť/jednoduchosť. Je však zjavné, že popísať hru ako zložitú alebo jednoduchú nestačí na opísanie jej kvality. Zložitá hra môže byť frustrujúco zložitá na pochopenie, ale takisto hlboko zložitá s tým, že zvládnutie jej zložitosti prináša do hry veľkú pestrosť možností. Na opačnej strane zložitostného spektra, hra môže byť príliš jednoduchá a rýchlo stratiť svoju zaujímavosť, alebo môže byť elegantne jednoduchá a pritom zábavná. Je zjavné, že pre lepšie pochopenie hernej zložitosti potrebujeme schopnosť hernú zložitosť bližšie klasifikovať.

Schell vo svojej knihe [1] definuje dva typy zložitosti:

- **vrodená zložitosť** (en. *innate complexity*)
- **vynárajúca zložitosť** (en. *emergent complexity*)

Vrodená zložitosť je daná pravidlami hry samotnými, a teda množstvom pravidiel, ktoré musí hráč zvládnuť aby mohol hru v plnej miere hrať. Príkladom sú všetky spôsoby pohybov figúrok v šachu. Keď sa odvoláme na Hunickovej MDA rámec [7] z úvodu kapitoly, môžeme tvrdiť, že táto zložitosť zodpovedá úrovni herných mechaník.

Vynárajúca zložitosť je zložitosť ktorá vzniká kombinovaním herných pravidiel v zložitých herných situáciách. Príkladom pre šach je množstvo rozložení figúriek na šachovnici alebo rôzne typy otvorení hry. V rámci MDA táto zložitosť korešponduje s úrovňou hernej dynamiky.

Hovoríme, že hra je elegantná vtedy, keď jej základné pravidlá dávajú hre malú vrodenu zložitosť, ale ich kombinovaním sa hráčom predstiera veľká vynárajúca zložitosť, ktorá robí hru pri každom hraní pestrou. Môžeme teda povedať, že elegantná hra je zložitá a jednoduchá zároveň.

2.3.2 Naučiteľnosť

Ako sme detailne rozoberali v kapitole 2.2, hry sa líšia od ostatných typov aplikácií jedinečnými charakteristikami, ktoré robia herný zážitok odlišným od zážitku z používania iných typov aplikácií. Pri návrhu hier sa preto herní dizajnéri riadia špecializovanými pravidlami.

Častým pomocným nástrojom pre navrhovanie hier sú heuristiky. Príkladom takýchto heuristík sú tie navrhnuté Desurvire et al. [12]. Tieto heuristiky nazvané ako heuristické vyhodnocovanie hrateľnosti (en. Heuristic Evaluation of Playability) sú zaradené do základných štyroch kategórií *hranie* (en. *Game Play*), *herný príbeh* (en. *Game Story*), *herná mechanika* (en. *Mechanics*) a *použiteľnosť* (en. *Usability*). Pokiaľ by sme vykonali podrobnú inšpekciu týchto heuristík, všimnime si, že už významom kľúčových pojmov nie úplne korešpondujú s definíciami z MDA rámca. Príklady: *Hranie, ktoré v MDA patrí do úrovne estetik, je osamostatnené od príbehu. Heuristiky z kategórie hrania súvisia v MDA skôr s úrovňami estetiky a dynamiky. Heuristiky hernej mechaniky súvisia v MDA s dynamikou aj mechanikou. Použiteľnosť korešponduje s úrovňou MDA mechanik, hoci v heuristikách predstavuje samostatnú kategóriu.*

Podstatnou informáciou z tejto situácie je fakt, že aj kategorizácia heuristík určených pre hry je nejednoznačná, ako a aj pojmy, ktoré môžu byť rôznymi expertmi chápané odlišne. Korhonen - ktorý vo svojom článku [11] porovnáva dvojicu súborov heuristík určených na vyhodnocovanie hier - tiež poukazuje na komplikácie s aplikáciou heuristík v experimente, kde tímy poverené s odhaľovaním problémov s hrateľnosťou pri odhalení rovnakých problémov ďalej často zaradili tieto problémy do odlišných heuristických kategórií. Je teda zjavné, že heuristický prístup vyhodnocovania hier potrebuje ďalšie zlepšovanie.

Na čo je však zaujímavé poukázať, je osamelá pozícia, ktorú má použiteľnosť - v tradičnom softvérovom zmysle - v týchto existujúcich heuristikách určených na vyhodnocovanie hier. Keďže už vieme, že použiteľnosť v hrách je úzko spojená s ponorením do hry, má zmysel začať uvažovať nad hernou použiteľnosťou v odlišnom svetle, než je len heuristická kategória ekvivalentná s tradičnou použiteľnosťou, ktorá sa musí v kontexte hier kombinovať s odlišnými hernými heuristikami, pretože sama o sebe sa na hry nešpecializuje.

Predpokladajme, že naučiteľnosť má pre hry aj iný význam, než len schopnosť používateľov naučiť sa pracovať s rozhraním. Aby sme mohli túto úvahu rozvíjať v ďalšej sekcii, analyzujeme čo znamenajú použiteľnosť a naučiteľnosť z hľadiska bežného softvéru:

Použiteľnosť bola definovaná Jakobom Nielsenom [13] v kontexte heuristického vyhodnocovania softvéru ako miera, v ktorej môže byť softvér použitý používateľmi na splnenie určitých cieľov. Použiteľnosť sa ďalej skladá z naučiteľnosti, výkonnosti, zapamätateľnosti, chýb a spokojnosti. My budeme ďalej špeciálnu pozornosť venovať hernej naučiteľnosti.

Podľa Grossman et al. štúdie softvérovej naučiteľnosti [14] má naučiteľnosť nie vždy presne stanovenú definíciu. V zdrojových textoch sa naučiteľnosť zvykne vyskytovať ako schopnosť dosiahnutia dobrej výkonnosti pri počítačových úlohách či v počítačom časovom intervale pri práci v softvérovom systéme, prípadne ako zlepšovanie výkonnosti v počítačom časovom intervale či počas celej histórie používania, alebo ako eventuálne dosiahnutie špecifickej či optimálnej výkonnosti.

Poznáme celkovo 7 typov metrík aplikovateľných pre meranie naučiteľnosti:

- **Úlohové metriky** - percento úspešných používateľov pri plnení úlohy, čas potrebný kým sa používatelia naučia vykonávať úlohu optimálne, pokles chýb
- **Príkazové metriky** - nárast počtu a zložitosti príkazov, počet príkazov známych používateľovi
- **Mentálne metriky** - znižovanie doby premýšľania
- **Subjektívne** - počet otázok používateľov ktoré sa týkajú naučiteľnosti, odpovede v dotazníku o naučiteľnosti
- **Dokumentačné** - množstvo pozretí do dokumentácie, trvanie čítania dokumentácie
- **Metriky použiteľnosti** - porovnávanie kvality používania, rozdiel medzi nováčikmi a expertmi
- **Pravidlové** - množstvo pravidiel ktoré popisujú systém

Pre vyhodnocovanie naučiteľnosti sa používajú rôzne formatívne(priebežne počas vývoja) alebo sumatívne(po ukončení vývoja) metodológie.

Formatívne metodológie

- Protokol premýšľania nahlas - počas tréningu, aj počas testovania
- Denník - Účastníci dokumentujú, čo sa naučili počas intervalu. Musia vedieť identifikovať problémy naučiteľnosti.
- Coaching - Podobné ako premýšľanie nahlas, iba sa nerozprávajú sami so sebou, ale môžu interagovať s coachom. Dobré pre vyhodnocovanie naučiteľnosti, lebo coacha sa používatelia skutočne zvyknú pýtať na veci, ktorým nechápu.

Sumatívne metodológie

- Protokol premýšľania nahlas s kvantitatívnym ohodnotením výkonnosti a času čítania dokumentácie

- Obmedzený čas na tréning s dokumentáciou po ktorom nasleduje zadanie úlohy. Výkonnosť vyhodnocujú sudcovia

Vieme teda o rôznych spôsoboch, ako sa naučiteľnosť meria. Čo ale presne predstavuje naučiteľnosť vo svojej podstate? Dokážeme si ju nejakým spôsobom rozdeliť? Grossman et al. takisto poskytujú odpoveď. Vo všeobecnosti poznáme dva typy naučiteľnosti:

1. **iniciálnu** (en. *initial*)
2. **rozšírenú** (en. *extended*)

Iniciálna naučiteľnosť určuje mieru, do ktorej je používateľ schopný zvládnuť v primeranom rozmere a krátkom čase používanie softvérového riešenia. Rozšírená naučiteľnosť znamená mieru, do ktorej je používateľ schopný optimalizovať svoju výkonnosť pri používaní softvéru. Do štádia kedy začne používateľ takto optimalizovať to môže trvať určitú dobu používania, niekedy aj niekoľko rokov.

Na začiatku tejto sekcie sme vzniesli pripomienku, že vzhľadom na charakter hier v ich kontexte možno naučiteľnosť nemusí predstavovať presne to isté, ako pri ostatných typoch softvéru. Poukázali na význam adresovania konceptu tradičnej použiteľnosti a naučiteľnosti v kontexte hier, a analyzovali sme význam a spôsoby vyhodnocovania tradičnej naučiteľnosti. V nasledujúcej sekcii sa budeme venovať nášmu záveru o v zt'ahu medzi naučiteľnosťou a hrami.

2.3.3 Herná naučiteľnosť aj jej vzťah k hernej zložitosti

Zadefinovali sme si naučiteľnosť a rozdelili sme ju na iniciálnu a rozšírenú. Pre hry má toto rozdelenie kľúčový význam, zvlášť keď si uvedomíme, že pre maximalizáciu herného zážitku by mal hráč zvládnuť všetky potrebné herné mechaniky čo najskôr, a nie až za niekoľko rokov. Keď hráč zvládne tieto mechaniky, je pripravený ponoriť do tej zaujímavej, vynárajúcej sa zložitosti hry.

V predchádzajúcej podkapitole sme si položili otázku, či má použiteľnosť, a teda v rozšírenom zmysle aj naučiteľnosť pri hrách presne ten istý význam, ako pri ostatných typoch aplikácií.

Pri bežnej aplikácií je cieľom iniciálnej naučiteľnosti, aby sa používateľ naučil vykonávať základné prípady použitia, a cieľom rozšírenej naučiteľnosti je zvládanie dodatočných vlastností aplikácie za účelom optimálnych výkonov pri práci.

Oproti tomu, v hre by mal byť hráč stále prekvapovaný novými situáciami (rôzne otvorenia v šachu, *boss* odolný voči nejakému typu útokov, náhle stúpnutie náročnosti v nebezpečnej oblasti a podobne). Náš záver pre problematiku iniciálnej a rozšírenej naučiteľnosti v hrách je preto nasledovný:

Iniciálna naučiteľnosť:

Iniciálna naučiteľnosť je prirodzene nutná už na začiatku hry, aby hráč zvládol základné ovládanie a herné mechaniky. Pokiaľ by sme na rôzne situácie v rámci hry, pri ktorých používateľ musí zvládnuť nové herné mechaniky pozerali ako na prípady použitia (v zmysle, že používateľ

interaguje s hrou za účelom dosiahnuť nejaký cieľ) potom sa iníciaľna naučiteľnosť vzťahuje aj na predstavovanie nových herných prvkov, aj neskôr v priebehu hry. Hry na učenie nových herných prvkov často aplikujú do istej miery mechaniku tutoriálov.

Iníciaľnu naučiteľnosť je možné považovať za uplatnenú aj situácií, kedy sa hráč neučí žiadnu novú hernú mechaniku, pracuje s mechanikami ktoré sa už naučil používať, ale ešte neoptimalizuje, naopak, učí sa zvládať základné herné dynamiky.

Príklad: Hráč už vie, že sa dá kliknúť na položky v inventári a že položky sa dajú kombinovať, ale ešte sa musí naučiť, že skombinovaním kameňa a palice sa vytvorí sekera, ktorú potrebuje na sekanie dreva.

Následne je na zodpovednosti herného dizajnéra určiť, ktoré herné dynamiky by sa mal hráč naučiť zvládnuť ešte v rámci iníciaľneho učenia, ktoré dynamiky stačí odložiť na rozšírené učenie pozdĺž celého trvania hry, a ktoré dynamiky možno ponechať na hráča; nech na ne príde sám. Iníciaľna naučiteľnosť sa tu jednoznačne javí ako tá najviac kritická pre hráčove správne uchopenie hry hneď od začiatku a má preto väčší význam, aby sme sa ňou ďalej zaoberali.

Rozšírená naučiteľnosť:

Pri opakovanom hraní môže postupne hráč spoznávať rôzne stratégie, ktorými môže reagovať na rôzne podmienky, v ktorých sa môže v hre ocitnúť. Niekedy môže hráč poznať viacero alternatívnych riešiteľských stratégií, a vybrať si tú optimálnu (ak taká existuje, optimálne riešenie môže závisieť na preferenciách hráča, dokonca je preferované aby bolo riešenie viac a boli dizajnérmi poriadne vybalansované so svojimi vlastnými výhodami a nevýhodami). Táto naučiteľnosť nastáva vtedy, keď hráč úplne ovláda všetky aktuálne dostupné herné mechaniky.

Ako sme už v priebehu kapitoly naznačili, vidíme veľké spojenie medzi naučiteľnosťou v hrách, a hernými mechanikami a dynamikami v rámci MDA. Hráč sa iníciaľnou naučiteľnosťou učí zvládať herné mechaniky a základné herné dynamiky, zatiaľ čo rozšírenou naučiteľnosťou hráč ďalej spoznáva herné dynamiky. V kapitole 2.3.1 sme opisovali, že vrodená zložitost' hier je založená na množstve herných mechaník, kým vynárajúca zložitost' je založená na pestrosti herných dynamík. Môžeme teda uzavrieť, že medzi zložitost'ou hier a ich naučiteľnosťou existuje určité prepojenie.

Iníciaľna naučiteľnosť v hrách slúži na ovládnutie vrodenej zložitosti hry a úvodu do vynárajúcej hernej zložitosti. Rozšírená naučiteľnosť v hrách slúži na zdokonaľovanie hráča vo vynárajúcej sa zložitosti hry.

2.4 Vyhodnocovanie vlastností hier

V súčasnosti sa vďaka dostupnosti služieb ako Steam², iTunes alebo Google Play, a tiež rastúcej podpore digitálneho predaja hier výrobcami herných konzol (Microsoft, Sony, Nintendo) výrazne otvára trh pre malé herné štúdiá, ktoré vyvíjajú tzv. *indie* hry (indie z anglického slova

² <http://store.steampowered.com/about/>

independent, tj. nezávislý). Príklady takýchto *indie* herných titulov sú hry ako Fez, The Stanley Parable, Don't Starve, Guacamelee! alebo Steamworld Dig.

Nielenže digitálny predaj uľahčuje efektívne obísť potrebu veľkého vydavateľa a okamžite spustiť masový predaj priamo na cieľový trh, ale rozvoj platforiem pre davové financovanie ako Kickstarter (kde v čase písania tejto práce hry tvoria kategóriu s najvyššou zarobenou finančnou hodnotou v dolároch³) zjednodušujú financovanie pre tých *indie* vývojárov, ktorí dokážu svojou tvorbou zaujať hráčsku verejnosť. *Indie* hry bývajú často pre trh inovujúce a prinášajú nové zaujímavé herné mechaniky, ktoré však vyžadujú vysokú mieru zdokonalenia, pokiaľ má byť hra po vypustení hrateľná, a umožniť *indie* štúdiu ďalší rozvoj. Preto má vyhodnocovanie hrateľnosti (vrátane naučiteľnosti) zvlášť vysoký význam aj pre *indie* herných vývojárov.

V tejto kapitole sa budeme venovať metódam, ktoré sú v praxi alebo výskumne aplikované na vyhodnocovanie hrateľnosti hier.

2.4.1 Playtesting

Štandardnou metodológiou pre vyhodnocovanie hier počas vývoja je tzv. *playtesting* [15], [1]. Cieľom *playtestingu* je zistiť, akým spôsobom sa hráči správajú v rámci hry, aké dávajú úlohám priority, ako vnímajú rôzne herné mechaniky, alebo ako zvládajú samotné ovládanie hry. Medzi typické príklady techník využívaných v rámci *playtestingu* patria premýšľanie nahlas (Používatelia nahlas vyjadrujú svoje myšlienky, a pozorovateľ im môže klásť dodatočné otázky na spresnenie zistených výsledkov testovania. Je tiež vhodné pri výkone protokolu vytvárať videozáznam, ktorý umožní dodatočné skúmanie používateľa bez straty informácií.), dotazníky, alebo interview po ukončení hrania. Tieto techniky sa v praxi často kombinujú.

Rozdiel medzi *playtestingom* a bežným testovaním použiteľnosti spočíva v zadaní úloh, ktoré používatelia na začiatku testovania dostanú. Zatiaľ čo pri testovaní použiteľnosti odlišných typov aplikácií dostanú používatelia zadanie s úlohami, ktoré majú v testovanom systéme/aplikácii vykonať, pri *playtestingu* hráči dostávajú iba jedinú úlohu: Hrať sa tak, akoby to robili prirodzene. Vďaka tomuto dokážu herní dizajnéri overiť dodatočné vlastnosti hry, ako napríklad či hráči chápu ciele hry, vedia ako tieto ciele dosiahnuť, a podobne [15]. Nevýhodou je, že takto vykonávaný test môže byť časovo náročnejší pre vytvorenie realistických podmienok, pri ktorých by hráč iniciálne prišiel do kontaktu s hernou mechanikou/dynamikou, ktorá je predmetom *playtestu*.

Výsledky *playtestingu* môžu herní dizajnéri následne využiť v kombinácii s niektorou z rôznych súborov heuristik na vyhodnocovanie hier pre spresnenie a štrukturalizáciu hrateľnosti vytváranej hry.

2.4.2 Expertná metóda

Alternatívne môžu byť expertmi z profesijnej oblasti tvorby hier ako užitočný nástroj pre okamžité vyhodnocovanie hrateľnosti hry využité heuristiky na vyhodnocovanie hier, teda expertnou metódou a bez potreby zabezpečovať veľké množstvo vhodných hráčov potrebných

³ <https://www.kickstarter.com/help/stats>

na vyhodnotenie cez *playtesting*. Uplatnenie heuristík vyžaduje iba podrobnú prehliadku hry expertom, ktorý na základe zoznamu heuristík a svojich vlastných skúseností s vývojom hier vyhodnotí vlastnosti danej hry.

Touto metódou však nemusia byť ani najlepšimi expertmi objavené všetky problémy hrateľnosti včas, zvlášť v prípade takých hier akými sú *MMORPG* (hry na hrdinov pre masívne veľa hráčov), kde neočakávané komplexné správanie veľkého množstva hráčov môže narušiť fungovanie akejkoľvek naplánovanej hernej mechaniky (napr. silnejší hráči bránia slabším hráčom v prístupe k istým zdrojom a tým neúmerne zvyšujú hodnotu daného zdroja).

Heuristiky hrateľnosti môžu byť počas vývoja hier užitočné na upriamanie pozornosti herných dizajnérov na vlastnosti, ktoré by mala kvalitná hra spĺňať. Avšak pre vyhodnocovanie hier či už cez čistou expertnú metódu alebo cez *playtesting* heuristiky hrateľnosti predstavujú niekoľko problémov, ktoré môžu oslabiť ich použiteľnosť [11].

Prvoradý problém heuristík predstavuje ich úroveň abstrakcie: príliš vysoká úroveň abstrakcie je ťažko pochopiteľná a dizajnéri ju nemusia vedieť aplikovať všade tam, kde je to potrebné; ak je naopak úroveň abstrakcie príliš nízka, vzťahuje sa výlučne len na určitú kategóriu hier a herných mechaník. Cílené sledovanie heuristík môže mať tiež za následok narušenie ponorenia do hry (viď. kapitola 2.2), ktoré je pritom jedným zo základných cieľov hry. Heuristiky neprinášajú zrozumiteľné pochopenie hry ako celku, a preto môžu sami o sebe identifikovať falošné problémy, alebo určité problémy neodhaliť vôbec [16]. Pre heuristické vyhodnocovanie je preto nutné, aby bol účastník skutočným expertom, dobre oboznámeným s významom heuristík a dokázal ich aplikovať v praxi, nakoľko udržiavať na pamäti rozsiahle zoznamy heuristík a aplikovať každú vo všetkých jej dopadoch v každom časovom okamihu nie je v ľudských schopnostiach.

2.4.3 Dotazník o ponorení

Ďalším aspektom herného zážitku, ktorý je vhodný na vyhodnocovanie, je úroveň ponorenia hráča do hry. Ponorenie do hry je náročné na vyhodnocovanie počas hrania hry, nakoľko samotné jeho vyhodnocovanie môže narušiť ponorenie.

Jennett et al. [8] pre tieto príčiny vytvorili záverečný dotazník o ponorení, ktorý je založený na takých konceptoch ako kognitívna absorpcia, prúd a sprítomnenie (viď. kapitola 2.2). Nedostatky tohto dotazníka sú podobné ako pri všetkých dotazníkoch: jeho časové oddelenie od skutočného prežívania používateľa v priebehu testovania, skreslenie subjektívnou retrospektívou, nemožnosť získania informácií o stave ponorenia v každom časovom okamihu a vplyv konkrétnych herných prvkov na tento stav.

2.5 Zhrnutie

V tejto kapitole sme vysvetlili MDA hierarchiu komponentov hier, na ktorú naša metóda ďalej do značnej miery nadväzuje. Tiež sme poukázali na fakt, že ponorenie, dôležitý aspekt herného zážitku je vysoko závislé na dobrej miere hernej naučiteľnosti. Herná naučiteľnosť sa delí na iniciálnu a rozšírenú, pričom iniciálna naučiteľnosť - ktorej sa v tejto práci venujeme primárne - zodpovedá učeniu sa herných mechaník, ale aj základných herných dynamík.

Ako je zjavné, v doméne vyhodnocovania hier existuje široký priestor na vylepšovanie. Pritom zlepšenie schopností herných vývojárov vyhodnocovať kvalitatívne vlastnosti herného zážitku hier, ktoré vytvárajú, a to priamo počas vývoja, môže mať vysoko kladný vplyv na kvalitu vyprodukovaných herných zážitkov. V nasledujúcej kapitole sa preto venujeme konceptu implicitnej spätnej väzby, ktorá je v doméne hier stále málo preskúmaná, a poukážeme na možnosti jej aplikácie na vyhodnocovanie hier. Konkrétne na vyhodnocovanie iniciálnej hernej naučiteľnosti, na ktorú sme v tejto kapitole poukázali ako na kľúčovú.

3 Implicitná spätná väzba a jej využitie

Pri vyhodnocovaní vnímania vlastností produktu používateľmi má centrálnu rolu akú techniku na zbieranie spätnej väzby používame. Vo všeobecnosti existujú dva základné prístupy k zbieraniu spätnej väzby: implicitná a explicitná spätná väzba. Rozdiel medzi týmito prístupmi spočíva v tom, že pri explicitnej spätnej väzbe používateľ vie o poskytnutí a obsahu spätnej väzby.

Jednou z najbežnejších metód na zber údajov od používateľa, je takzvaný protokol premýšľania nahlas, ktorý sme už bližšie opisovali v kontexte vyhodnocovania naučiteľnosti v sekcii 2.3.2 aj v kontexte *playtestingu* v sekcii 2.4.1. Premýšľanie nahlas je typickým príkladom zbierania explicitnej spätnej väzby, keďže používateľ nahlas rozpráva o svojom vnútornom vnímaní produktu či už v prítomnosti alebo neprítomnosti osoby riadiacej experiment.

Metóda premýšľania nahlas čelí však rôznym problémom, ktoré ju robia mene vhodnou pre účel jej využitia [17]:

- asynchrónnosť medzi výrokmi a vnemami/myšlienkami používateľa
- nepohodlnosť pre používateľa
- vyššia možnosť vydania falošne pozitívnej spätnej väzby pod sociálnym tlakom
- narušenie toku práce na úlohách

Posledný problém by sme mohli odstrániť presunutím zberu spätnej väzby na koniec experimentu vo forme interview alebo dotazníka, ale v takom prípade by sme stratili pojem o umiestnení používateľských vnemov v čase a ešte viac by rozšírila časová priepasť medzi používaním produktu a zberom dát.

Pre potrebu odpovedať na nedostatky metód explicitnej spätnej väzby predstavuje sľubné riešenie použitie spätnej väzby implicitnej. Pretože implicitná spätná väzba nie je závislá na pozornosti účastníka testu, môže sa zbierať kedykoľvek v priebehu vykonávania testu a poskytovať informácie o používateľskom zážitku v každom časovom okamihu. Navyše, pokiaľ nie je metóda alebo technológia zberu implicitnej spätnej väzby príliš invazívna, dokážeme ju získať bez narušenia používateľského zážitku, čo je zvlášť užitočné pre špecifiká hier, ktoré sú tak silno závislé na hernom zážitku.

Typickými príkladmi techník na zber implicitnej spätnej väzby je napríklad zber údajov o klikaní myšou, vykonávaní atomických akcií v rozhraní, alebo meranie času do splnenia úlohy. Zvlášť sú tieto techniky populárne pri sledovaní činnosti používateľov vo webových rozhraniach. V hrách môže byť pre ich komplexitu zvlášť pri rozsiahlejších hrách o niečo náročnejšie identifikovať dôležité atomické akcie, avšak v akejkol'vek hre je možné zabezpečiť ich logovanie. Naša metóda do značnej miery stavia aj na tomto prístupe.

Jednou z výskumne zaujímavých a relatívne málo prebádaných techník zberu implicitnej spätnej väzby je technológia sledovania pohľadu používateľa (en. *gaze tracking*), ktorá vyžaduje špeciálne softvérové a hardvérové technológie. V nasledujúcich sekciiach vysvetlíjeme základné koncepty technológií sledovania pohľadu a metód s nimi spojenými, uvádzame užitočné príklady metrík, ktoré možno pri sledovaní pohľadu uplatniť a výber

príkladov experimentov, v ktorých výskumníci uplatnili sledovanie pohľadu na vyhodnocovanie používateľského zážitku (ako ukážku možností sledovania pohľadu, inšpiráciu a referenciu pri tvorbe našich vlastných experimentov na overenie našej metódy).

3.1 Sledovanie pohľadu

Sledovanie pohľadu je technológia, ktorá ako názov napovedá, umožňuje určenie polohy bodu na obrazovke, na ktorý sa používateľ v danej dobe pozerá. Väčšina súčasných zariadení na sledovanie pohľadu [18] využíva na svoje fungovanie odraz infračerveného lúča od používateľovej sietnice na porovnanie polohy sietnice a zrenice. Toto umožňuje dosiahnuť meranie polohy pohľadu s chybou presnosti približne jeden uhlový stupeň.

Technológie na sledovanie pohľadu majú uplatnenie v mnohých odborných oblastiach, ako je psychológia, marketing, zdravotníctvo [19], vzdelávanie [20], a prirodzene nachádza využitie aj pri vyhodnocovaní používateľského zážitku.

Jedná sa o relatívne novo rozvíjajúcu sa technológiu, ktorá sa v rozsahu posledných rokov stáva čoraz dostupnejšou, a existuje reálna možnosť, že zariadenia na sledovanie pohľadu sa stanú aj bežnou súčasťou kamier používaných v osobných počítačoch. Takisto sa kvalitatívne zlepšujú technické možnosti týchto zariadení (napríklad odbúravanie nárokov na vzdialenosť používateľa od obrazovky [21]). Pre pokročilý rozvoj tejto technológie, ako aj prísľub zaujímavých údajov na analýzu sa javí táto technológia ako výskumne zaujímavá a je vhodné sa ňou patrične zaoberať.

V súčasnosti sa výskumy založené na sledovaní pohľadu bežne opierajú o hypotézu [22], že smer pohľadu je okamžite časovo mapovaný na myšlienkové procesy. Pri riešení úlohy teda riešiteľ spracúva informácie na ktoré sa pozerá, a pozerá na ne až kým ich nespracuje. Hoci táto implikácia nemusí byť vždy všeobecne platná a nie všetky myšlienkové procesy sa musia odrážať v správaní pohľadu, sledovania pohľadu je stále vysoko cenným vstupom pri skúmaní používateľského zážitku. So sledovaním pohľadu sa dnes už nepracuje len pri výskume, ale aj pri praktickom vyhodnocovaní používateľského zážitku trhových produktov a služieb.

V oblasti sledovania pohľadu existuje niekoľko široko zaužívaných pojmov pre fenomény, ktoré sú spojené so správaním ľudského pohľadu [23]. Fixácia (en. *Fixation*) predstavuje sústredenie pohľadu na pomerne nemennú pozíciu v rámci sledovanej oblasti s trvaním približne 200-300 milisekúnd. Sakáda(en. *Saccade*) je rýchly pohyb smeru pohľadu medzi jednotlivými fixáciami, počas ktorého oko nevníma nič na prejdenej dráhe.

Oblasť záujmu, ďalej AOI (z anglického Area of Interest) je ohraničená - spravidla obdĺžniková - oblasť na obrazovke reprezentujúca prvok používateľského rozhrania. Je odporúčane definovať AOI ako čo najviac homogénne oblasti (napr. reklamy, navigácia, obsah), a v prípade, že neskôr získame potrebu analyzovať údaje na nižšej úrovni (napr. pre jednotlivé tlačidlá kontrolného panelu), údaje je možné agregovať dodatočne.

V oblasti sledovania pohľadu existuje niekoľko typov vizualizácií dát, ktoré výrazne uľahčujú ľudským výskumníkom prehľad v dátach o pohľade. Najznámejšími takýmito vizualizáciami sú skenovací dráha a tepelná mapa.

Skenovacia dráha (en. Scanpath) je dráha pohybov oka zvolených pozorovateľom, ktorá sa zobrazuje ako graf v ktorom uzly znázorňujú fixácie a hrany znázorňujú sakády. Čím je uzol väčší, tým dlhšie trvala fixácia. Nevýhodou pre využitie tejto vizualizácie osobitne pre hry je, že pre často meniace sa herné rozhranie pri statickej dráhe nevieme s presnosťou určiť ktorý prvok rozhrania sa pod jej uzlami nachádzal.

Tepelná mapa je vizualizácia, pri ktorá slúži na zvýraznenie oblastí na ktoré sa používateľ najviac pozeral. Body v ktorých sa nachádzali fixácie sú vyznačené teplými farbami, a vzdáľovaním od týchto bodov sa farba mení na studenšie farby až do priesvitnosti. Nevýhodou tejto vizualizácie osobitne pre hry je opäť fakt, že nie je úplne vhodná pre často meniace sa herné rozhranie, zvlášť pri hrách, kde je často celá plocha aspoň v istom čase využitá.

3.2 Metriky sledovania pohľadu

Údaje zo zariadenia na sledovanie pohľadu sú cenným zdrojom, ktorý je však nutné pre vyhodnocovanie používateľského zážitku určitým spôsobom interpretovať a kvantifikovať, aby bolo možné vykonávať porovnávanie medzi rôznymi súbormi dát.

Pre tieto účely nám môžu poslúžiť štandardné metriky sledovania pohľadu, ktoré sú rôznymi známymi spôsobmi využívané v už existujúcich výskumoch [18].

Čas zotrvania (en. *Dwell Time*) je metrika, ktorá definuje pre každú AOI (oblasť záujmu, obdĺžniková oblasť obrazovky) dĺžku času, ktorú používateľ zotrval pohľadom v jej oblasti. Do tohto času sa započítavajú fixácie aj sakády. Platí, že čas zotrvania pod 100ms nie je dostatočný pre plné spracovanie informácií z AOI, a čas nad 500ms je už viac-menej vždy možné považovať za dostatočný. Pokiaľ nevieme zabezpečiť rovnakú dĺžku trvania merania (čo je vhodné, pretože pri dlhšom vystavení stimulom je správanie používateľa prirodzene odlišné), je vhodné počítať relatívnu hodnotu času zotrvania v percentách.

Počet fixácií (en. *Number of Fixations*) je podobná metrika, ktorá počíta počet fixácií na AOI počas priebehu testovania. Vysoko koreluje s časom zotrvania, a preto sa zvyčajne ako smerodajná metrika uvádza iba čas zotrvania. Pokiaľ by sme chceli poukázať na rozdiel medzi časom zotrvania a počtom fixácií (tj. krátke alebo dlhé fixácie v danej AOI), môžeme využiť nasledujúcu odvodenú metriku:

Trvanie fixácie (en. *Fixation Duration*) ako priemerná dĺžka fixácií zvyčajne dosahuje hodnoty medzi 150 a 300ms. Platí, že čím viac je používateľ zaangažovaný do objektu v AOI, tým dlhšie je trvanie fixácií.

Poradie (en. *Sequence*) označuje poradie, v ktorom sa používatelia na danú AOI pozreli. Táto metrika hovorí o tom, aká je výraznosť danej AOI v kontexte zadanej úlohy ktorú používatelia majú začať riešiť.

Čas do prvej fixácie (en. *Time to First Fixation*) je metrika s podobným významom, pretože hovorí o priemernom čase, kedy sa používatelia prvý krát pozreli na danú AOI. Môže byť zvlášť zaujímavá, pokiaľ je v našom záujme docieľiť, aby si používateľ všimol a spracoval obsah AOI

v rámci určitého časového intervalu. Príklad: Doba celkového navštívenia stránky je 7s, oznámenie by si mal všimnúť do prvých 5s

Opakované návštevy (en. *Revisits*) číselne vyjadrujú počet opakovaných návštev AOI pohľadom po jej predchádzajúcom opustení. Vyjadruje tzv. lepkavosť (en. *stickiness*) prvku rozhrania, pre definovanie prvkov ku ktorým sa používatelia neustále vracajú.

Pomer zásahov (en. *Hit Ratio*) vyjadruje pomer používateľov ktorý pohľadom navštívili danú AOI na celkový počet používateľov. Táto metrika zreteľne vyjadruje viditeľnosť prvku v rozhraní.

3.3 Analýza príbuzných experimentov založených na sledovaní pohľadu

V nasledujúcej kapitole uvádzame analýzu niekoľkých experimentov vykonaných inými výskumníkmi, ktoré využívajú ako bázu pre pozorovanie používateľa technológie sledovania pohľadu. Naším cieľom bude poukázať na možnosti a obmedzenia metód sledovania pohľadu, a vytvoriť si základný prehľad o štruktúre experimentov ktoré používajú takéto metódy.

3.3.1 Eivazi - 8-hlavolam

Účelom experimentu [17] bolo overenie hypotézy, že údaje zo sledovania pohľadu je možné aplikovať s využitím strojového učenia na predvídanie ľudských mentálnych stavov počas riešenia problému. Mentálne stavy boli pri tréningu a overovaní získané na základe protokolu premýšľania nahlas, ktorý sprevádzal riešenie úloh 8-hlavolamu so sledovaním pohľadu.

Experiment so sledovaním pohľadu vykonali autori nad skupinou zloženou zo 14 študentov alebo výskumníkov. Údaje zo sledovania pohľadu dvoch účastníkov boli neskôr vylúčené zo vzorky dát z príčin ich slabej kvality.

Po vykonaní kalibrácie a zahraní sa jednej skúšobnej hry hráči v troch testovacích intervaloch vyriešili tri hlavolamy rovnakej náročnosti. Počas experimentu účastníci komentovali svoj proces riešenia hlavolamu a ich komentáre boli nahrávané.

Dvaja experimentátori po vykonaní experimentu klasifikovali komentáre do jednej zo šesť kategórií modifikovanej O'Harovej a Paynovej schémy popisujúcej kognitívne fázy riešenia problému, pričom títo experimentátori dosiahli relatívne vysokú presnosť prekrytia 86%.

Na klasifikáciu údajov zo sledovania pohľadu bola potom použitá SVM metóda počítačového učenia. Tá pri overení klasifikátora na inej ako tréningovej vzorke bola následne dosiahnutá presnosť 53%. Relatívne malú presnosť autori odôvodňujú výberom aplikovanej metódy počítačového učenia a navrhujú niekoľko zlepšení, vrátane zväčšenia objemu tréningovej vzorky.

3.3.2 Bartels - AMBR

Štúdia [24] s využitím sledovania pohľadu bola vykonaná nad činnosťou v AMBR (Agent-based Modeling and Behavior Representation Project) kde účastníci plnili zjednodušenú hernú úlohu riadenia letov, ktoré prilietajú a odlietajú z oblasti. Účelom bolo overenie existujúceho modelu správania sa používateľov pod záťažou v dvoch verziách používateľského rozhrania (textové a farebné) s odkazom na nedostatky predtým vykonaných experimentov, ktoré

sledovanie pohľadu nepoužívali (využitie metódy merania kognitívnej záťaže NASA TLX ktorá vyhodnocuje záťaž až po ukončení výkonu testovacieho scenára).

Všetci 16 účastníci boli skúsení hráči počítačových hier. Po vykonaní experimentu sa kvôli nízkej kvalite údajov v prípade dvoch účastníkov veľkosť testovacej vzorky znížila na 14.

Celý experiment prebiehal v troch behoch rozdelených do troch dní. Prvý beh bol určený na tréning účastníkov v plnení úlohy riadenia lietadiel. Na začiatku experimentu účastníci vyplnili dotazník o svojich predchádzajúcich herných skúsenostiach, a bolo im vysvetlené použitie zariadenia na sledovanie pohľadu, po čom začali riešiť riadiacu úlohu.

V druhom a treťom behu účastníci vykonali celkovo 6 rôznych scenárov riadenia lietadiel, pričom druhý beh bol považovaný za tréning a tretí za hlavný test. Polovicu scenárov hráči vykonali s textovým a polovicu s farebným rozhraním. Polovica účastníkov pritom najprv vykonávala scenáre s textovým rozhraním a druhá polovica najprv s farebným. Po dokončení každého scenára hráči vyplnili TLX záťažové hodnotenie daného scenára.

Na základe údajov zo sledovania pohľadu vyšli najavo extrémne rozdiely medzi správaním pri dvoch typoch používateľských rozhraní, vyplývajúce z inherentných vlastností odlišných typov používateľského rozhrania, aj z odlišnosti používateľmi zvolených riešiteľských stratégií. Pri farebnom rozhraní používatelia venovali výrazne vyššiu pozornosť displeju s radarom a textovým správam venovali omnoho nižšiu pozornosť ako pri textovom rozhraní. Tento rozdiel medzi rozhraniami navyše rástol úmerne so zvyšujúcou sa kognitívnou záťažou, a pri farebnom rozhraní sa táto odlišná stratégia ukázala ako vysoko efektívna, na čo poukazuje zvýšená časová výkonnosť hráčov a znížené úrovne TLX pracovnej záťaže. Toto poukazuje na značný stupeň prepojenia medzi činnosťou pohľadu a kognitívnou činnosťou.

3.3.3 Sundstedt - Labyrint

Motiváciou tejto štúdie [25] bolo zlepšiť existujúce metódy na pridelenie zdrojov na vykresľovanie v rámci hernej grafiky. Metóda tzv. salienčných máp (en. *Saliency Maps*) zvažuje množstvo faktorov ako je pohyb, farby a kontrast, ale neberie do úvahy prítomnosť úlohy ktorú hráč aktuálne plní. Autori predpokladajú, že algoritmy predvídajúce správanie pohľadu by mohli existujúce metódy obohatiť a tým šetriť výpočtové zdroje, ktoré môžu byť využité inak, a teda potenciálne poskytnúť používateľom kvalitnejší herný zážitok. V princípe by boli šetrené zdroje grafickej karty tým, že by grafika bola menej optimalizovaná v tých častiach obrazovky, ktorým hráči aj tak venujú menšiu pozornosť.

Cieľom kvantitatívneho experimentu bolo preto poukázať, akým spôsobom úloha mení správanie pohľadu. Experiment vykonali na jednoduchom labyrinte, pričom ako hypotézu autori určili, že časti labyrintu (reprezentované ako AOI v kontexte sledovania pohľadu) majú rozdielne úrovne dôležitosti, a teda hráči by mali venovať vyššiu pozornosť ceste von, a menšiu pozornosť slepým koncom.

Experimentu sa zúčastnilo 40 účastníkov, pričom každým zo štyroch typov podmienok pri úlohe bolo vystavených 10 z nich (Podmienky: nemodifikované, s rozptyľujúcimi grafickými

efektmi, so zvukovými efektmi, iba video nahrávka hrania hry). Každé meranie trvalo 72 ± 12 s podľa dĺžky trvania hry.

Experiment potvrdil, že používatelia skutočne venujú viac pozornosti ceste von z labyrintu. Sledovanie pohľadu teda dokáže určiť triedy objektov na ktoré sa hráči sústredia, nad rozdiel od existujúcich prístupov salienčných máp.

Počet fixácií za sekundu bol pri všetkých podmienkach približne rovnaký, takže autori si urobili záver, že stratégia riešenia úlohy bola približne rovnaká - hráči priebežne hľadajú cestu von a aj navigujú, neriešia tieto úlohy sekvenčne.

Žiadne podstatné rozdiely v umiestnení fixácií medzi podmienkami, ktorým bol používateľ vystavený - vrátane tých podmienok, že iba sledoval video - podľa autorov svedčia o tom, že aj keď používateľ nedostane úlohu, keď je vystavený takému zjavnému problému ako je labyrint, úloha vzniká "v jeho/jej hlave".

3.3.4 Pretorius - Network Management

Pri správe sietí používatelia na to určených nástrojov narábajú s množstvom komplexných informácií, ktoré musia byť vhodným spôsobom sprostredkované a zložito vizualizované.

Aplikácia na ktorej bolo testovanie realizované bola vyvíjaná pod dohľadom dvoch expertov vykonávajúcich heuristické vyhodnocovanie na základe existujúcej komplexnej vyhodnocovacej metodológie. Obsahom článku bolo zavedenie technológií sledovania pohľadu do tejto metodológie.

Autori do metodológie začlenili viaceré metriky sledovania pohľadu ako počet fixácií a dĺžku fixácií, počet fixácií na AOI a dĺžka fixácií na AOI, čas do prvej fixácie a skenovaciu dráhu. Pilotný experiment bol vykonávaný so šiestimi účastníkmi, a trval približne 40 až 60 minút, pričom v strede experimentu dostali používatelia prestávku. Každý účastník vykonal dva typy scenárov. Medzi úlohami bola kontrolovaná presnosť údajov a v prípade potreby vykonali recalibráciu. Pre každú úlohu bola následne vykonaná samostatná analýza. Na konci experimentu všetci používatelia vyplnili dotazník k použiteľnosti.

Pozorovaním výsledkov zo sledovania pohľadu dospeli autori k niekoľkým záverom o rozhraní nástroja: používatelia venovali vyššiu pozornosť grafickým prvkom než textovým. Jedna vysoko kritická a preto aj často navštevovaná časť obrazovky bola príliš malá, a keď používatelia hľadali údaje, kým ich úspešne našli, prechádzali vysoké množstvo AOI. Sledovanie pohľadu sa preto ukázalo ako užitočný príspevok pre zaužívanú vyhodnocovaciu metodológiu.

3.3.5 Kickmeier-Rust - 80Days

Objektom experimentu [4] so sledovaním pohľadu bola edukačná hra 80Days. Táto detská hra učí hráčov svetovú geografiu tým, že hráč v priebehu hry pomáha mimozemšťanovi mapovať Zem.

Cieľom experimentu bolo zistiť závislosti medzi správaním pohľadu hráčov, a zlepšením, ktoré dosiahli medzi papierovým pre-testom a post-testom. Experimentu sa zúčastnilo celkovo 9 detí, ktorým bolo počas experimentu povolené hrať sa tak dlho, ako len chceli.

Na základe výsledkov testov a údajov zo sledovania pohľadu bolo pozorované, že deti ktoré sa naučili najviac venovali najviac pozornosti najviac sledovanej AOI, v ktorej boli obsiahnuté edukačné informácie. Deti s horšími výsledkami boli tiež charakteristické kratšími sakádami, teda deti s lepšími výsledkami boli zjavne pokojnejšie. Pozitívny efekt na zlepšenie mala aj dlhšia doba trvania hry.

3.4 Zhrnutie

V tejto kapitole sme vysvetlili podstatu implicitnej spätnej väzby, ako aj jej výhod voči zberu spätnej väzby na explicitnej báze – menovite sa jedná o jej menšie nároky na čas aj energiu pri samotnom zbere spätnej väzby, ako aj fakt, že implicitná spätná väzba inherentne nenarúša prirodzené správanie pozorovaného účastníka.

V analýze sledovania pohľadu – technike na zber implicitnej spätnej väzby, ktorá prináša istú mieru náhľadu do kognitívnych úkonov pozorovaného – sme poukázali, že možnosti jej využitia sú široké ako dokazujú rôzne – tiež na hry zamerané – experimenty. Sledovanie pohľadu nám poskytuje zdroj relevantných údajov aj pre aj pre vyhodnocovanie hernej naučiteľnosti, kde je jej pre jej vyhodnotenie dôležité poznať, či pozorovaná osoba spracovala informáciu, ktorú jej hra poskytuje, alebo učenie sa herných prvkov zlyhalo z iných príčin.

V nasledujúcej kapitole aplikujeme tieto zozbierané poznatky pri návrhu našej metódy, ktorej úlohou je využitie implicitnej spätnej väzby zo sledovania pohľadu pre vyhodnocovanie naučiteľnosti v rámci počítačových hier.

4 LCM metóda

V predchádzajúcich kapitolách sme si vysvetlili, že naučiteľnosť v hrách sa v niektorých aspektoch významne líši od bežnej naučiteľnosti, tak ako ju chápeme v kontexte iných softvérových aplikácií. Ďalej sme poukázali na význam jej vyhodnocovania v priebehu vývoja hier, a vysvetlili sme si, ako pre tieto účely funguje metodológia bežne známa ako *playtesting*. V poslednej kapitole sme si vysvetlili výhody implicitnej spätnej väzby, a tvrdili sme, že jej využitie – zvlášť v spojení s technológiami sledovania pohľadu – môže mať rázny prínos pre vyhodnocovanie naučiteľnosti v hrách počas *playtestingu*.

V nasledujúcej kapitole aplikujeme vyššie zhrnuté poznatky a venujeme sa popisu metódy dolovania prípadov naučenia (en. Learning Case Mining) – ďalej len „LCM metódy“ – našej navrhovanej metódy pre vyhodnocovanie naučiteľnosti hier. Cieľom LCM metódy je vyhodnotenie úspešnosti naučenia herných mechaník a základných herných dynamík, tak ako ich zadefinuje herný dizajnér. V rámci tejto metódy využijeme technológie na sledovanie pohľadu používateľa, ako aj iné ukazovatele implicitnej spätnej väzby.

4.1 Zaradenie do vývojového procesu

Vývoj hier zvyčajne možno popísať špirálovým modelom vývoja softvéru. V rámci tohto modelu je pre nás zaujímavá fáza vývoja a testovania, kedy je v každej iterácii vytvorený nový rozšírený prototyp či prototypy, zvyčajne obsahujúce nové, alebo pozmenené herné mechaniky a dynamiky.

Počas každej iterácie je dôležité overovanie takýchto prototypov. Herný prototyp je overovaný na rozličné vlastnosti: napríklad kvalitu, spokojnosť hráčov, náročnosť, výkonnosť, a aj naučiteľnosť. V rámci druhej kapitoly sme si uviedli *playtesting* ako metodológiu často používanú na tieto účely, na vyhodnocovanie hier na základe interakcie hráča s hrou.

Metódu LCM sme navrhovali tak, aby ju bolo možné začleniť do *playtestingu* a aby tam doplnila bežne používané techniky zberu spätnej väzby, ako napríklad pozorovanie hráčov, protokol premýšľania nahlas, interview a dotazníky. Každý *playtest* je zameraný na overovanie určitých herných mechaník alebo dynamík a metóda LCM tiež slúži na vyhodnocovanie iniciálnej naučiteľnosti herných mechaník alebo dynamík, ktoré sú počas tohto *playtestu* overované.

Alternatívne, keďže LCM je automatizovateľná metóda, je ju možné na údaje z *playtestu* aplikovať aj okamžite po ukončení hrania hry. Vedúci *playtestu* tím ešte pred sumatívnym zberom explicitnej spätnej väzby získa sumárny náhľad z LCM do priebehu hry, a otázky určená na upresnenie problémov naučiteľnosti môže sústrediť práve na tie časti hry, v ktorých mal hráč najväčšie problémy.

4.2 Výzvy a obmedzenia

Pred tým, ako popíšeme samotné jadro LCM metódy, analyzujeme otázky, na ktoré sme si počas návrhu metódy museli zodpovedať. Tieto otázky predstavujú prevažne výzvy, ktoré vychádzajú z podstaty hier samotných, ako aj z techník zberu spätnej väzby, na ktorých je LCM metóda založená. Ich riešenie sme navrhli tak, aby sme čo najpresnejšie zodpovedali účelom našej metódy.

Zvlášť v doméne počítačových hier predstavuje problém sledovania pohľadu špeciálnu výzvu [26], pretože počas herného procesu sa dynamicky mení vnútorný stav hry, čo spôsobuje zmeny herného rozhrania premietaného hráčovi na obrazovke. Toto môže pri stanovovaní AOI v rámci rozhrania predstavovať významný problém pre experimentátorov.

Ďalšou výzvou ktorej musíme čeliť, je výber typov hier, na ktoré je možné aplikovať našu metódu tak, aby boli údaje zo sledovania pohľadu pre jej žánre a vzhľad používateľského rozhrania vôbec aplikovateľné. Pri ďalšom návrhu našej metódy sme sa snažili o návrh, ktorý by bol čo najviac všeobecný.

Dodatočnou výzvou je samotná dostupnosť úprav samotných hier. V prípade, že by bola metóda aplikovaná na hry, ktorých autori majú motiváciu pre potreby testovania naučiteľnosti vykonať v hre úpravy, boli by sme schopní zaznamenávať si pozície prvkov používateľského rozhrania, ktoré tvoria AOI v dátach zo sledovania pohľadu, a to paralelne so sledovaním pohľadu samotným.

Jediným, minimálnym, a viac-menej len implementačným problémom je v prípade zberu údajov priamo z hry časová synchronizácia týchto údajov s údajmi sledovania pohľadu.

V prípade, že by sme neboli schopní zabezpečiť zaznamenávanie údajov o pozícií prvkov priamo z hry, naša metóda by bola použiteľná iba na hry s viac statickými prvkami rozhrania.

Pri ďalšom návrhu našej metódy sme sa rozhodli venovať sa hrám, pri ktorých sme schopní zaznamenávať si dodatočné údaje o ich používaní. Nakoľko je naša metóda určená pre herných návrhárov, ktorí sú schopní vykonávať úpravy vo svojich hrách, tento krok má pri návrhu našej metódy svoje opodstatnenie. Údaje z hier môžu vďaka tomuto predpokladu významne obohatiť naše možnosti na vyhodnocovanie hier - s náhľadom do vnútornej logiky hry - a naša metóda umožní vyhodnocovanie hier aj s dynamickejšim herným rozhraním.

Pri ďalšom návrhu našej metódy sme sa teda zamerali na vytvorenie metódy na overovanie naučiteľnosti hier vo všeobecnom rámci, bez zamerania na konkrétne hry alebo konkrétne herné žánre.

4.3 Prípady naučenia

Naša metóda slúži na vyhodnotenie iniciálnej naučiteľnosti herných mechaník a základných herných dynamík, o ktorých naučenie sa zaujíma herný návrhár. Herný návrhár si pri plánovaní *playtestu* (vyhodnocovanie hry na základe prirodzeného herného zážitku hráčov) určí prípady naučenia, ktoré sú zadefinované ako interakcie používateľa s hrou, pri ktorých sa hráč naučí konkrétnu hernú mechaniku alebo dynamiku. Prípady naučenia by mali zahŕňať:

1. Definíciu učenej hernej mechaniky/dynamiky
2. Stimuly z rozhrania, ktorých cieľom je naučiť používateľa hernú mechaniku/dynamiku
3. Stimuly z rozhrania súvisiace s hernou mechanikou/dynamikou
4. Použitie ovládacích prvkov súvisiacich s mechanikou/dynamikou

Prípady naučenia sú na najnižšej úrovni zložené z udalostí, ktoré musia počas hrania hry prebehnúť pre ich úspešné naučenie sa. Opisy prípadov naučenia môžu obsahovať nasledovné typy udalostí, ktoré možno priamo zaznamenávať:

- Fixácia na AOI(oblasť záujmu, obdĺžniková oblasť na obrazovke)
- Interakcia kurzora s prvkom herného rozhrania
- Stlačenie tlačidla na vstupnom zariadení (myš, klávesnica, herný ovládač)
- Vlastná logická udalosť (udalosť závislá na kontexte hry)

Prvé tri typy udalostí je možné určiť na základe údajov zo sledovania pohľadu v kombinácií s údajmi z logu pozícií prvkov herného rozhrania. Pre doplnenie našej metódy o udalosti na vyššej logickej úrovni súvisiace s vnútornou logikou hry (napríklad *hráč trafil cieľ*), naša metóda umožňuje herným vývojárom vkladať svojich vlastných udalostí zaznamenaných v počas behu hry.

Základné typy udalostí sa môžu usporiadať do sekvencií, a tak vytvoriť jednoduché zápisy prípadov naučenia ako obyčajné postupnosti úkonov, ktoré po sebe s neznámym časovým odstupom nasledujú. Takéto postupnosti by bolo potom relatívne jednoduché porovnávať so sekvenčnými údajmi o správaní používateľa. My však uprednostňujeme formu zápisu prípadov naučenia s o niečo vyššou vyjadrovacou silou. Na tieto účely sme si zvolili orientované grafy.

Orientované grafy nám umožňujú popísať správanie, ktoré by sa malo u hráča preukazovať pri učení sa herných prvkov na vyššej úrovni. Finálne správanie je síce v konečnom rámci sekvencia udalostí, avšak potenciálne sa hráč môže pri hraní hry správať rôznymi spôsobmi, či už podľa toho ako sa dynamicky rozvíja samotná hra (Príklad: *V strategickej hre môžu byť výsledky konfliktov dvoch armád závislé do istej miery na umelej inteligencii jednotiek, alebo jednoducho na šťastí*), ako sa hráč v rámci svojich možností sám rozhodne (Príklad: *Akú stratégiu si hráč v strategickej hre zvolí na ochranu svojho mesta*), alebo ako sa hráč správa podvedome (Príklad: *Hráč si pokojne stavia obytné centrum svojho mesta, a ani si nevšimol, že ho napadla nepriateľská armáda*).

Prípady naučenia by tiež mali byť schopné túto realitu istým spôsobom zachytiť. Naša forma zápisu prípadov naučenia preto ďalej podporuje nasledovné operácie:

- Sekvencia
- Rozhodovanie
- Paralelizmus
- Cyklus

Jednotlivé operácie (okrem sekvencie, ktorá je základom vzájomného spájania udalostí) si vysvetlíjeme nasledovne: (Uvádzame aj príklad zo žánru stratégií v reálnom čase)

Rozhodovanie nám pomáha modelovať situácie, kedy sa prípad naučenia môže rozvíjať do viacerých alternatívnych smerov, pričom každý z nich je správny a rovnako vedie k splneniu cieľov celkového prípadu naučenia. Príklad: *Hráč si uvedomí, že nemá zdroje na postavenie*

budovy bud' prečítaním potrebných nákladov, alebo kliknutím na tlačidlo postaví, a všimnutím si varovania o nedostatku zdrojov.

Paralelizmus v prípadoch naučenia slúži na opis viacerých sekvencií udalostí, ktoré sa môžu počas hernej interakcie vyskytovať súčasne. Táto charakteristika vyplýva zo zložitosti hier, kde hráč často spracováva rôzne podnety takmer naraz, a podľa typu hry sa snaží na ne patričným spôsobom reagovať. Príklad: *Hráč posielal robotníkov ťažiť zdroje a zároveň tieto zdroje používa na budovanie armády.*

Cyklus môžeme použiť pri zápise prípadu naučenia, pokiaľ by mal používateľ opakovať určitú činnosť. Cykly ohraničené podmienkou (napr. kým hráč neurobí toto, tak...) môžeme vyjadriť jednoducho aj vložením očakávanej podmienky ako udalosti do sekvencie. Pri porovnávaní prípadu naučenia s nameranými údajmi o ňom budeme vedieť, ak nebola splnená podmienka. Zadefinovanie cyklov explicitne ohraničených podmienkou by však nepotrebné skomplikovalo popis prípadov použitia. V našom kontexte teda cyklom myslíme jednoduchý cyklus s predurčeným počtom opakovaní – teda obyčajne zjednodušenie zápisu k-násobného opakovania cyklickej sekvencie. Príklad: *Hráč postaví tri farmy*

4.4 Algoritmus porovnávania prípadov naučenia s nameranými údajmi

Ako sme opísali v predchádzajúcich kapitolách, naša metóda na vyhodnocovanie hernej naučiteľnosti je založená na porovnávaní údajov zozbieraných pri hraní hry s prípadmi naučenia – modelmi popisujúcimi interakcie používateľa s hrou, pri ktorej sa má hráč naučiť nejakú hernú mechaniku alebo dynamiku.

Z pohľadu herných návrhárov je našim záujmom vedieť určiť, či hráč splnil prípad naučenia bez ťažkostí, alebo naopak, či nemal problém s pochopením istých herných prvkov. Medzi dvoma hráčmi chceme byť schopní určiť, ktorý z nich pochopil cieľ prípadu naučenia ľahšie. A v prípade, že niekto hernú mechaniku nepochopil vôbec, je v našom záujme tiež identifikovať koreň problému naučiteľnosti v našej hre.

Keďže ale údaje z používania sú charakterizované ako postupnosť udalostí v čase, a prípady naučenia sme definovali ako orientované grafy využívajúce operácie ako rozhodovanie, paralelizmus, alebo cyklus, naša metóda musí zahŕňať tiež metódu na kvantifikáciu stupňa podobnosti medzi nameranými údajmi a týmito našimi preddefinovanými modelmi.

Dolovanie biznis procesov (en. *business process mining*, ďalej BPM) je disciplína, ktorá je na prvý pohľad podobná k riešeniu nášho problému. Jej účelom je totiž umožniť pochopenie procesov na základe faktov a zlepšovanie existujúcich procesov na základe takto získaných poznatkov [27]. BPM sa odlišuje a špecializuje od obyčajného dolovania v dátach svojim zameraním na procesy. BPM nezahŕňa len odhaľovanie procesov v údajoch, ale tiež hľadanie odchýlok medzi procesmi a ich realizáciou, predpovedanie oneskorení, poskytovanie rád pri rozhodovaní, hľadanie návrhov na zlepšenie biznis procesov, a podobne.

Oblasťou, ktorá je pre nás zvlášť zaujímavá v rámci BMP je preverovanie zhody (en. *conformance checking*). Ako jej názov napovedá, preverovanie zhody sa zoberá priradovaním položiek z logu udalostí k položkám procesného modelu, a porovnávaním medzi nimi.

Preverovanie zhody môže mať pre biznis dva rôzne ciele. Tým prvým je zisťovanie, či procesný model nie je odlišný od reality – ak by bol nesprávny, je možné, že je potrebné ho zmeniť. Druhým cieľom je zistiť, či sa aktivity biznisu nelíšia nesprávnym spôsobom od procesného modelu – ak áno, je potrebné vykonať akcie pre ich nápravu a dosiahnutie lepšej zhody s modelom.

Pokiaľ si zameníme procesné modely za prípady naučenia, uvedomíme si, že preverovanie zhody v BPM má veľa spoločného s hlavným konceptom našej metódy vyhodnocovania naučiteľnosti. Kým BPM preverovanie zhody umožňuje identifikovať nenaplnené biznis procesy a vykonať nápravné akcie, cieľom LCM(naša metóda) je identifikovať problémy s naučením a na základe nameraných výsledkov pomôcť navrhnuť nápravné akcie pre zlepšenie naučiteľnosti.

Algoritmus používaný pri preverovaní zhody na vyhodnocovanie naplnenia modelu reálnymi údajmi sa nazýva *Token Replay*. Token Replay je založený na procesných modeloch vo forme tzv. *workflow sietí* čo sú Petriho siete s určeným počiatočným a koncovým uzlom, pričom všetky ostatné uzly sa nachádzajú na ceste medzi týmito dvoma uzlami.

Pri každom štádiu si algoritmus Token Replay zaznamenáva štyri počítadlá tokenov: vyprodukované tokeny, spotrebované tokeny, chýbajúce tokeny a zostávajúce tokeny. Algoritmus postupne prechádza workflow sieťou podľa postupnosti akcií ktoré boli vykonané. Pri každom prechode sa počítadlo vyprodukovaných tokenov zvyšuje o počet vstupov do prechodu a počítadlo spotrebovaných tokenov o počet výstupov. V prípade, že pri akcií nie je možný žiaden prechod, alebo sa prechod zasekne kvôli tomu, že potrebná akcia chýba vo vstupnej postupnosti, inkrementuje sa počítadlo chýbajúcich tokenov a miesto v ktorom chýbal token je označené. Na konci behu algoritmu sa hodnota počítadla zostávajúcich tokenov stanoví ako počet miest, ktoré boli takto označené v jeho priebehu.

Počítaním tokenov na konkrétnych miestach workflow siete dokážeme následne identifikovať problematické miesta - tam kde chýbajú správne akcie, alebo sa nachádzajú akcie navyše. Workflow siete tiež umožňujú paralelizmus a rozhodovanie, podobne ako je v našom návrhu riešenia.

Nedostatkom tohto algoritmu pre naše použitie je však fakt, že operuje nad akciami usporiadanými v tzv. prípadoch (en. *case*), čo sú krátke inštancie biznis procesov, zatiaľ čo nami zozbierané údaje sú reprezentované ako kontinuálny prúd udalostí z priebehu testovania hry, ktoré sú prirodzene radovo dlhšie, než aj tá najdlhšia možná cesta cez prípad naučenia.

Token Replay je teda algoritmus prispôsobený na konkrétne potreby dolovania biznis procesov, avšak pre naše potreby nie je ideálny.

Náš algoritmus je založený na predpoklade, že každý prípad naučenia je - ako orientovaný diagram - možné prepísať na množinu všetkých možných ciest cez takýto diagram od začiatku po koniec. Medzi týmito cestami určíme tú, ktorá sa najviac podobá na sekvenciu reálnych herných udalostí, a v nej určíme miesta, v ktorých sa líši od modelu prípadu naučenia. Tieto miesta môžu podľa svojho charakteru svedčiť o problémoch s naučiteľnosťou. Tento prístup

môžeme zvoliť preto, že hráč by síce mal úspešne zvládnuť cieľ prípadu naučenia, ale hra by mu stále mala umožniť aj zlyhať. Ďalej sa preto môžeme zaujímať už iba o ten najlepší možný prípad výskytu cesty, ktorá sa približuje modelu prípadu naučenia.

Pokiaľ už uvažujeme nad našim problémom ako nad hľadaním podobnosti medzi reálnymi hernými udalosťami a jednotlivými cestami cez prípad naučenia - teda dvoma sekvenciami - môžeme na ich porovnávanie použiť existujúce metódy určené na porovnávanie reťazcov. Jednou z takýchto metód je tzv. Levenshteinova vzdialenosť.

Levenshteinova vzdialenosť je definovaná ako vzdialenosť dvoch reťazcov daná počtom operácií potrebných na to, aby sa tieto reťazce stali totožnými. Tieto operácie sú nasledovné:

- **Adícia** - Pridanie znaku
- **Delícia** - Odstránenie znaku
- **Substitúcia** - Náhrada znaku

Vzhľadom na naše potreby a charakter údajov s ktorými pracujeme však vieme, že medzi udalosťami sa môže vyskytnúť jednoduchá výmena poradia, ktorá však nie je to isté, ako keď sa v sekvenciách udalostí vyskytnú dve kompletne substitúcie. Preto budeme uvažovať upravenú, tzv. Damerau-Levenshteinovu vzdialenosť [28], pôvodne určenú na detekciu preklepov v texte. Tá totiž pridáva do Levenshteinovej vzdialenosti štvrtú, dodatočnú operáciu:

- **Transpozícia** - Premiestnenie susedných znakov

Pri bližšom nahliadnutí na tieto štyri operácie si uvedomíme, že nie je úplne možné a vhodné pre výpočet podobnosti našich reťazcov jednoducho spočítať všetky operácie do jedinej metriky vzdialenosti. Nie všetky operácie majú totiž rovnaký dopad na rozdiel medzi cestou cez prípad naučenia a nameranou sekvenciou udalostí.

Keďže sekvencia nameraných udalostí z hrania hry a zvlášť udalostí založených na pohľade je pomerne hustá, adícia udalostí redundantných z hľadiska prípadu naučenia sa v nameraných údajoch vyskytuje pomerne často. Jednotlivé kroky prípadu naučenia tiež môžu byť časovo distribuované nepravidelne a počas dlhšieho časového intervalu, pričom prípad naučenia môže obsahovať situácie, kedy v ňom hráč jednoducho nemôže ďalej pokračovať, kým sa v hre neocitne v novej situácii (Príklad: *V strategickej hre po spustení stavby budovy kasárni musí hráč čakať kým sa budova postaví. Pred nasledujúcou akciou tvorby vojakov bude preto zvýšené množstvo adícií, keďže aj v prípade, že hráč by počas celej stavby kasárni nič nerobil, minimálne jeho oči budú pravdepodobne skenovať prvky grafického rozhrania na obrazovke*). V týchto momentoch pritom adície nepredstavujú žiaden problém.

Operácia adície preto bude mať na vyhodnocovanie vzdialenosti len malý vplyv, hoci neprimerane vysoké množstvo pridaní na mieste, kde ich neočakávame tiež môže niesť informáciu o probléme s učením sa.

Vzhľadom na fakt, že namerané údaje sú husté, operácie delície a substitúcie tiež prakticky splývajú do jednej operácie, keďže je vysoko pravdepodobné, že ak v nameraných údajoch chýba očakávaná udalosť, na jej mieste sa bude nachádzať aspoň jedna odlišná udalosť, ktorá

nemusí mať s prípadom naučenia žiaden súvis. Oproti adícií sú však operácie delécie výrazne závažnejšie až katastrofálne na celkovú naučiteľnosť prípadu naučenia, keďže delécie v najlepšej ceste cez prípad naučenia svedčia o tom, že hráč nedosahuje cieľ prípadu naučenia (Príklad: *V strategickej hre hráč napriek očakávaniam dizajnérov nikdy nepostaví okolo svojho mesta ochranné hradby aby sa ochránil pred nepriateľmi*).

Význam transpozície susedných znakov je závislý na kontexte vymenených udalostí, keďže jeho význam je väčší ako pridanie znaku a svedčí na odlišnosti reálneho správania hráča od prípadu naučenia, ale je menej katastrofálny ako delécia znakov, ktorá hovorí o tom, že hráč istú udalosť nevykonal vôbec (Príklad: *V strategickej hre hráč vždy prečíta popis budovy, ktorú môže postaviť až potom, akú ju postaví, hoci sme očakávali, že využije možnosť informovať sa o budove, ktorú stavia vopred*).

Ako sme poukázali, jednotlivé operácie Damerau-Levensteinovej vzdialenosti majú polárne odlišný dopad na celkové vyhodnotenie podobnosti medzi cestou cez prípad naučenia a reálnymi nameranými údajmi. Celkovú vzájomnú vzdialenosť by sme teda mohli vypočítať ako vážený súčet počtov Damerau-Levenshtein operácií, avšak nakoľko nepoznáme váhy jednotlivých operácií (a jednoznačne ich ani nie je možné určiť, keďže každé vyhodnocovanie môže mať rôzny počet a hustotu herných udalostí) a každá operácia nesie odlišnú informáciu s vlastnou pridanou hodnotou, pri výpočte podobnosti počítame počty týchto operácií ako samostatné metriky.

Pri výbere najlepšieho výskytu prípadu naučenia vyberáme teda prioritne: najprv podľa počtu odstránených znakov, potom podľa počtu odlišne usporiadaných znakov a napokon podľa počtu pridaných znakov vo vnútri cesty.

4.5 Spojenie výsledkov vyhodnocovania od viacerých používateľov

V predchádzajúcej sekcii sme diskutovali metriky podobnosti, ktoré v našej metóde uplatňujeme na vyhodnocovanie prípadu naučenia nad nameranými údajmi z hrania hry. Tieto metriky nám poskytujú informáciu o tom, či hráč dosiahol cieľ prípadu naučenia a zvládol požadovanú hernú mechaniku/dynamiku.

Na základe poznatkov len o hernom zážitku jedného hráča nedokážeme s vysokou spoľahlivosťou určiť, či sa nesplnený alebo čiastočne nesplnený prípad naučenia rovná objektívnemu problému naučiteľnosti; Alebo naopak, či hráč na základe svojich predchádzajúcich skúseností neprekonal existujúci problém naučiteľnosti, ktorý by iných hráčov dezorientoval. Pre tieto príčiny v našej metóde uvažujeme tiež spojenie výsledkov viacčlennej množiny testujúcich hráčov.

Pre každého hráča máme k dispozícii jeho najbližší prechod prípadom naučenia, spolu s metrikami pridania akcií, odstránenia akcií, a premiestnenia susedných akcií, z ktorých každá nesie odlišnú informačnú hodnotu.

Adície medzi jednotlivými akciami prípadu naučenia nám v súčte pre jednotlivé prechody dávajú informáciu o počte aktivít, ktoré hráči vykonávali medzi akciami, no nie sú súčasťou prípadu naučenia. Preto pokiaľ by akcie mali nasledovať takmer bezprostredne po sebe, vysoké

hodnoty adícií hovoria o tom, že používatelia nevedeli, čo majú v danom bode urobiť. Operáciu delécie vkladáme do súčtu pre každú akciu, čím získame informáciu o akciách, ktoré hráči počas hrania nevykonávali vôbec. Za predpokladu, že sme prípad naučenia definovali správne, opakovaná delécia akcie predstavuje katastrofálny problém s naučiteľnosťou, pretože hráč sa zjavne vyhýbal aktivite, ktorú sme od neho očakávali.

Transpozície tiež spájame v súčte pre každý predpis akcie v prípade naučenia. Hodnoty nad nulou hovoria o tom, že hráči akciu zvyknú konať neskôr než sme čakali, nízke hodnoty zase že akcie konajú skôr, než sme čakali – dopady čoho je nutné interpretovať podľa konkrétneho prípadu. Príklad: Hráč sa môže pokúsiť postaviť budovu, a až potom si všimnúť, že na to nemá na to suroviny. Hráč by ale nemal najskôr stavať vojenské, a až potom ťažobné budovy (inak bude mať nedostatok surovín).

Spojenia metrík, na ktoré sme v tejto kapitole poukázali, je ďalej možné patričným spôsobom vizualizovať v rámci prípady naučenia – napríklad zvýraznením akcií a prechodov s varovnými hodnotami metrík. Buď farebne, alebo škálovaním grafických prvkov prípadu naučenia. Vizualizácia, ktorú sme zvolili aj pri overovaní metódy zvyrazňuje potenciálne problémy naučiteľnosti teplejšími farbami, zatiaľ čo menej problémové predpisy udalostí sú označené studenými farbami (viď. kapitola 6).

4.6 Výber účastníkov pre meranie naučiteľnosti

Tak ako v každej oblasti merania, kde je predmet merania značne subjektívny, aj pri meraní naučiteľnosti zohráva dôležitú rolu výber účastníkov merania. Správnym výberom účastníkov sa snažíme o pokrytie čo najširšej škály problémov naučiteľnosti, ktoré odlišní používatelia odhalia.

Nakoľko naša metóda je automatizovaná a poskytuje nám numerické metriky, jedná sa o kvantitatívnu metódu vyhodnocovania, do ktorej sme schopní začleniť toľko účastníkov, koľko uznáme za potrebné a koľko nám prostriedky na testovanie umožnia. Aj pri tejto kvantitatívnej metóde má však podstatný význam poznať rôzne kategórie hráčov, tieto kategórie v dostatočnom počte pokryť a pre každého hráča poznať, do ktorej kategórie patrí. Kategorizácia hráčov nám neskôr umožní zoskupovanie výsledkov testu podľa kategórií, a tak identifikovať problémy s naučiteľnosťou pre konkrétne skupiny hráčov.

Primárne rozdelenie hráčov do kategórií je podľa ich úrovne skúseností s hrami. Aj hráč, ktorý testovanú hru vidí po prvýkrát môže mať na základe hrania podobných hier, alebo dostatočných skúseností s hrami vo všeobecnosti väčšie predpoklady pochopiť niektoré herné prvky aj bez toho, aby ich ho hra v dostatočnej miere naučila. Takýto hráči sa pri bežnom rozdelení nazývajú *hardcore*, zatiaľ čo hráči s obmedzenými skúsenosťami s hrami sa označujú pojmom *casual*. Kategória hráčov, ktorí príležitostne hrajú hry, avšak nachádzajú sa niekde medzi týmito dvoma kategóriami sa zvyknú označovať pojmom *core*. Pokiaľ nie je naša hra vyslovene určená len pre jednu z týchto troch skupín a všetci účastníci neboli s týmto úmyslom správne vybraní zo správnej cieľovej skupiny, musíme pred vykonaním merania zistiť, do ktorej kategórie hráč patrí.

Hoci naša metóda pojednáva s iniciálnou naučiteľnosťou, nie je nutné, aby účastník testovania naučiteľnosti prišiel pri testovaní s hrou do kontaktu po prvý krát. Pojednáva sa totiž o iniciálnej naučiteľnosti na úrovni konkrétnych herných mechaník alebo dynamík, z ktorých hráč nemusí nutne zvládnuť všetky hneď od začiatku. Prvoradé je iba to, aby hráč pred meraním neprišiel do kontaktu s konkrétnou mechanikou/dynamikou.

5 Implementácia nástroja na automatizované vyhodnocovanie naučiteľnosti

V predchádzajúcej kapitole sme navrhli metódu dolovania prípadov naučenia (ďalej LCM metóda), ktorá umožňuje vyhodnocovanie naučiteľnosti hier, založenú na hľadaní podobnosti medzi modelom naučenia (zadefinovaným herným dizajnérom ako orientovaný diagram akcií, ktorého cieľom je hráča naučiť určitú hernú mechaniku alebo dynamiku) a reálnym správaním hráča počas playtestu (vyhodnocovanie hier na základe prirodzenej interakcie hráča s hrou).

V nasledujúcej kapitole diskutujeme základné implementačné princípy a komponenty nášho riešenia implementujúceho LCM metódu. Začíname pohľadom zhora nadol, od požiadaviek, ktoré sme mali na naše riešenie. Vysvetľujeme hlavné implementačné rozhodnutia a popisujeme formát vstupných údajov. Ďalej sa v skratke venujeme implementácií niektorých významných postupov obsiahnutých v našom riešení.

5.1 Nástroj Geddit

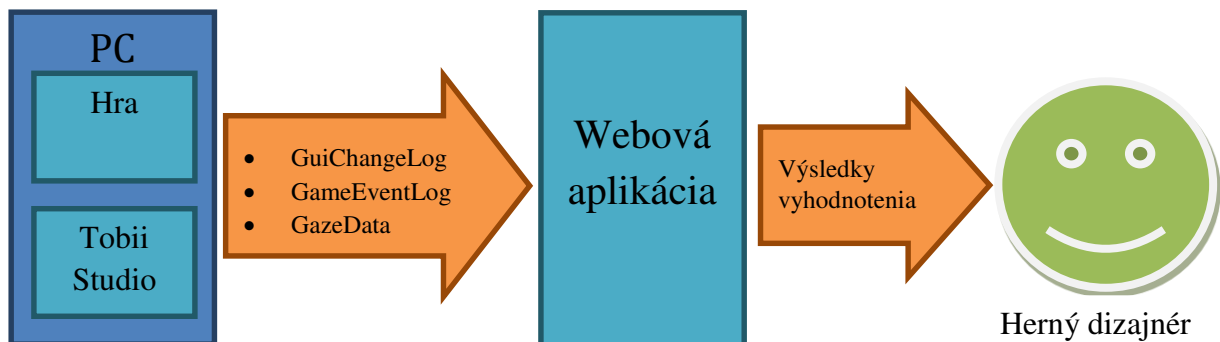
Prvoradou motiváciu implementácie nášho riešenia bolo experimentálne overenie LCM metódy. Zároveň sme ale chceli, aby naše riešenie bolo možné po overení v prípade záujmu čo najjednoduchšie prispôbiť tak, aby sa dalo sprístupniť na použitie herným dizajnérom na vyhodnocovanie naučiteľnosti ich vlastných hier. Technické riešenie sprístupňujúce našu metódu muselo byť preto jednoduché, dostupné a použiteľné pre akéhokoľvek záujemcu.

Ako ideálne technológie pre podporu týchto požiadaviek sme na implementáciu nášho riešenia zvolili jazyk Ruby a webový rámec Ruby on Rails. Hoci sme pre naše prvoradé potreby implementovali iba samotný algoritmus na dolovanie prípadov naučenia, táto voľba technológií nám umožňuje jednoduché sprístupnenie LCM metódy cez webovú aplikáciu. Výsledný nástroj, obsahujúci funkcie implementujúce LCM metódu, založený na webových technológiách, ale s aktuálne neimplementovaným webovým rozhraním sme pomenovali Geddit. Webovú aplikáciu sme ako najvhodnejšie riešenie zvolili preto, že je centralizovaná a jednoducho udržiavateľná, a zároveň by (po naimplementovaní webového rozhrania) umožnila prístup k našej metóde komukoľvek s internetom a webovým prehliadačom.

Na obrázku 5.1 uvádzame schému použitia webovej aplikácie Geddit. Počas hrania hry účastníkom hra generuje logy o zmenách prvkov grafického rozhrania aj o vlastných logických udalostiach, ktoré boli pre účely testovania navrhnuté herným návrhárom. Na zabezpečenie údajov o pohľade používateľa je použitý nástroj Tobii Studio, ktoré po ukončení merania umožňuje export týchto údajov vo formáte súboru XLSX.

Po ukončení testovania sú údaje z hrania hry buď odoslané na aplikačné rozhranie Geddit, alebo sú zozbierané herným dizajnérom a manuálne odovzdané na server.

Po poskytnutí všetkých potrebných údajov metóde herný dizajnéer spustí vyhodnotenie a cez webovú aplikáciu získa prístup k celkovým výsledkom testu.



Obr. 5.1 Schéma použitia webovej aplikácie Gedit

Jazyk Ruby a rámec na tvorbu webových aplikácií Ruby on Rails sme pri implementácii vybrali jednak pre veľkú vyjadrovaciu silu jazyka Ruby a preto, že obe tieto technológie sú dostupné voľne ako *opensource*. Navyše sú podložené živou komunitou, ktorá ich používa a ďalej zveľaduje, a preto sú dobre zdokumentované a jednoducho sa pre nich vyhľadávajú riešenia problémov.

5.2 Tobii Studio

Pre účely experimentov so sledovaním pohľadu používame v tejto práci nástroj Tobii Studio. Tobii Studio je rozsiahly softvérový nástroj od spoločnosti Tobii, ktorá je jednou z vedúcich komerčných organizácií v oblasti výskumu a aplikácie technológií na sledovanie pohľadu.

Tobii Studio umožňuje experimentátorom vytvoriť si svoj projekt, v rámci ktorého si experimentátor dokáže zdefinovať jednotlivých účastníkov a pre nich vykonávať sledovania. Sledovanie je možné vykonávať nad obrázkami, videami, webovými stránkami, pdf súbormi, alebo celou obrazovkou monitora bez obmedzení na obsah, aj keď v tomto prípade je vhodné zabezpečiť, aby testovaná aplikácia pokrývala plné rozmery obrazovky.

Po vykonaní sledovania môže experimentátor v Tobii Studiu prehliadať video, v rámci ktorého je mu na nahrané zábery obrazovky vykresľovaná poloha pohľadu používateľa v danom čase. Tobii Studio ďalej poskytuje vizualizácie ako skenovacia dráha (nazývané *Gaze Plot*), tepelná mapa a strapce (en. *Clusters*, AOI vypočítané na základe reálnych dát, funguje pokiaľ sa v dátach počas testu nachádzajú jednoznačné oblasti s najvyššou návštevnosťou).

Nad záznamom z experimentu je možné kedykoľvek zdefinovať rozloženie AOI (oblasť záujmu na obrazovke, spravidla obdĺžniková), ktoré je ďalej možné zaradiť do skupín.

Z Tobii Studia je možné ďalej vyexportovať sledovateľný videozáznam, vizualizácie, aj surové tabuľkové údaje z experimentu uložené v súbore napríklad vo formáte XLSX. Tieto údaje obsahujú pre každú pohľadovú udalosť, udalosť myši a udalosť klávesnice jej identifikátor, časovú pečiatku, dĺžku v milisekundách, jej typ a rôzne spracované či nespracované pozície pohľadu pre obidve oči. Tieto údaje sú zvlášť zaujímavé pre možnosti ich využitia pri implementácii nástroja potrebného pre aplikáciu našej metódy.

5.3 Vstupné údaje metódy

Naša metóda je založená na porovnávaní sekvenčných logov herných akcií s prípadmi naučenia zadefinovanými hernými dizajnermi. Aby implementácia našej metódy v aplikácii Geddit mohla spracovať a analyzovať ich podobnosť, herný dizajner musí Geddit poskytnúť všetky potrebné údaje vo forme, ktorú rozhranie Geddit pre všetky potrebné vstupy predpisuje.

Ako sme ich popísali v kapitole návrhu, prípady naučenia sú orientované grafy modulujúce správanie používateľa pri učení sa herných mechaník alebo dynamík. Používateľské akcie v nich je možné usporiadať pomocou operácií zreteľovania, rozhodovania, paralelizmu a cyklu.

Diagram, ktorý spĺňa vyššiu špecifikáciu a poskytuje odraz každej z potrebných operácií, je diagram aktivít, špecifikovaný ako súčasť jazyka UML. Jazyk UML je široko známy v kruhoch softvérových inžinierov a existuje široké množstvo nástrojov na tvorbu jeho diagramov v grafickom prostredí. Konkrétne jeho diagram aktivít je ľahko čitateľný a pochopiteľný aj pre UML laikov. Pre tieto príčiny sme ako základ pre zápis prípadov naučenia stanovili UML diagram aktivít.

Nástroj Umlet⁴ je voľne dostupný nástroj na tvorbu UML diagramov v grafickom rozhraní. Umlet umožňuje vytvorené diagramy ukladať v štandardnom formáte UXF. Pre jeho dostupnosť a jednoduchú prenositeľnosť sme Umlet zvolili ako nástroj na tvorbu prípadov naučenia. Ako vstup do metódy reprezentujúci prípad naučenia potom používame export diagramu aktivít vo formáte UXF.

Čistý diagram aktivít však neposkytuje pre našu metódu dostatok informácií o udalostiach, ktoré potrebuje herný návrhár definovať v rámci opisu prípadov naučenia. Rozhranie aplikácie Geddit preto predpisuje nasledovný syntax pre zápis herných udalostí:

TYP NazovUdalosti param1 param2 ...

Udalosť môže mať tri typy:

- **GAZE** – pohľad hráča na prvok herného rozhrania. Názov udalosti obsahuje identifikátor prvku rozhrania, ktorého sa fixácia pohľadu týka. Ako parametre môžu byť určené parametre daného prvku rozhrania.
- **INPUT** – vstup od hráča zo vstupného zariadenia. Názov obsahuje názov vstupu. Pokiaľ je vstup spojený so súradnicami na obrazovke (napríklad kliknutie myši), parametre môžu byť určené ako identifikátor a parametre daného prvku rozhrania.
- **CUSTOM** – vlastné logické udalosti zadefinované herným dizajnerom. Názov udalosti je ľubovoľný, určený herným dizajnerom, a parametre obsahujú tiež ľubovoľné parametre určené herným dizajnerom.

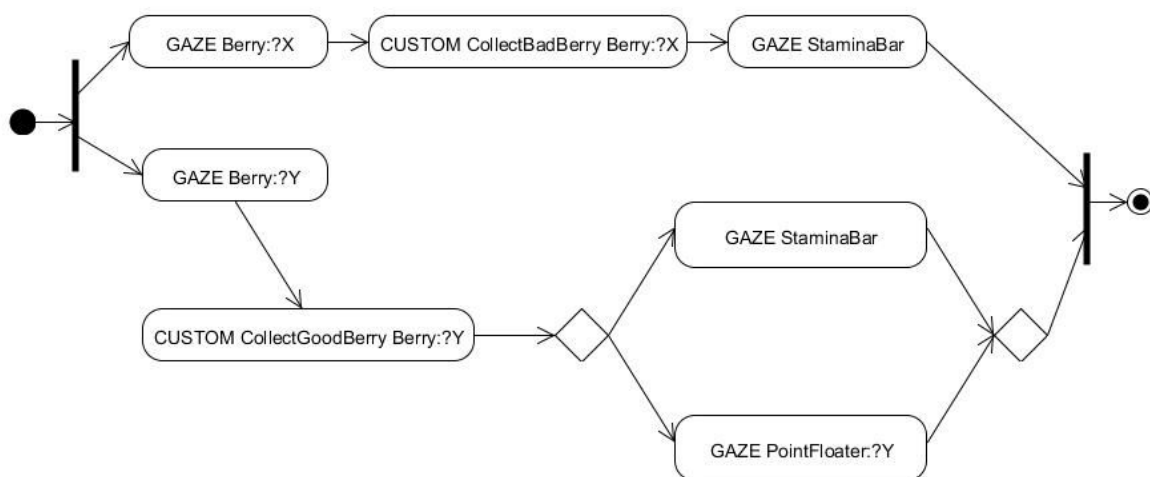
Ako sme uviedli, opisy udalostí v prípade naučenia disponujú identifikátormi a parametrami. Parameter môže byť buď konštantný reťazec, alebo variabilný identifikátor označený vo formáte:

⁴ <http://www.umlet.com/>

Class:?id

Class označuje triedu herného objektu, ktorého sa identifikátor týka, a ?id predstavuje zástupný symbol identifikátora. Príklad použitia identifikátorov vidíme na obrázku 5.2, ktorý zobrazuje ukážkový prípad naučenia vytvorený v nástroji Umlet.

?X a ?Y sú zástupné symboly pre dva identifikátory priradené dvom objektom bobuľ v rámci hry. ?X je pritom zlá bobuľa a ?Y je dobrá bobuľa. Na začiatku prípad naučenia očakáva, že sa hráč pozrie na objekty označené týmto identifikátorom a že ich následne zozbiera. Hráč by sa mal následne pozrieť na notifikátory, ktoré hovoria o tom, či hráč za pozbieranie bobule buď pri dobrej bobuli získal body a výdrž, alebo pri zlej bobuli stratil výdrž. Identifikátory nám umožňujú v postupnosti herných udalostí určiť, či hráč pozrel a následne pozbieral tú istú bobuľu. Pri dobrej bobuli je identifikátor použitý aj na označenie notifikátora o získanom počte bodov, ktorý sa zobrazil po zozbieraní bobule s daným identifikátorom.



Obr. 5.2 Prípad naučenia – Zbieranie bobuliek

Druhým vstupom našej metódy je sekvencia herných udalostí. Na získanie všetkých typov herných udalostí potrebuje aplikácia Geddit tri typy vstupov: log vlastných logických udalostí, údaje zo sledovania pohľadu a log zmien prvkov grafického používateľského rozhrania. Všetky tieto vstupy sú pre jednoduchosť vo formáte CSV, kde oddeľovač stĺpcov predstavuje znak čiarky.

Log vlastných logických udalostí obsahuje chronologicky usporiadané vnútorné herné udalosti, tak, ako si ich zadefinuje samotný herný dizajnér. Jedna udalosť sa nachádza na jednom riadku, a má formát:

NazovUdalosti,CasovaPeciatka,param1,param2,...

Časová pečiatka udalosti musí byť zapísaná v štandardnom formáte ISO 8601 a to s presnosťou na milisekundy, nakoľko herné udalosti vrátane udalostí pohľadu operujú na takejto časovej presnosti. Príklad logu vlastných logických udalostí s časom v štandardnom formáte:

CollectGoodBerry,2014-10-16 13:11:29.7866004+01:00,Berry:1

CollectMushroom,2014-10-16 13:11:36.1980815+01:00

Okrem vlastných udalostí od herného návrhára naša metóda narába aj s udalosťami zo vstupných zariadení a s udalosťami pohľadu. Ako sme opísali v návrhovej kapitole, naša metóda sa spolieha na schopnosť herných návrhárov logovať pozície prvkov grafického používateľského rozhrania, aby bolo možné akcie kurzora a fixácie pohľadu z nástroja Tobii Studio prepojiť priamo s jednotlivými prvkami grafického používateľského rozhrania, na ktoré sa hráč v danom momente pozerá.

Export údajov o pohľade a vstupoch je dodávaný vonkajším komponentom (Tobii Studio). Obsahuje údaje o fixáciách (sústredenia pohľadu na jednu oblasť) a sakádach (prechody medzi nimi) pohľadu používateľa, spolu s údajmi o stlačených klávesách na klávesnici a tlačidlách myši. Každý údaj je zaopatrený časovou pečiatkou v štandardnom formáte ISO 8601. Rozhranie je prispôbené na spracovávanie XLSX exportu konvertovaného do súboru vo formáte CSV, kde oddeľovač predstavuje znak čiarky. Nakoľko sa tu jedná o predurčený formát ktorý herný návrhári nemusia pri aplikácii metódy žiadnym spôsobom modifikovať, ďalší opis týchto údajov nie je potrebný.

Log zmien prvkov grafického používateľského rozhrania loguje pre všetky prvky grafického používateľského rozhrania iba zmeny ich pozícií, čím je dosiahnutá efektívnosť záznamu. Log je zapísaný vo formáte CSV, a každý riadok predstavuje inú zmenu grafického používateľského rozhrania. Formát jedného riadka je v takomto prípade nasledovný:

TypZmeny,IdentifikatorPrvku,x,y,w,h,CasovaPeciatka,param1,param2,...

Typ zmeny môže byť troch typov:

- Begin – zavádza prvok do grafického používateľského rozhrania
- Change – mení pozíciu prvku
- End – prvok mizne z grafického používateľského rozhrania

Po type zmeny nasleduje identifikátor prvku, ktorý predstavuje buď pri jedinom výskyte prvku na rozhraní ľubovoľný konštantný reťazec neobsahujúci znak dvojbodky, alebo v prípade výskytu viacerých rovnakých prvkov, ktoré je potrebné medzi sebou odlišovať identifikátor prvku vo formáte, ktorý sme definovali pri opise syntaxe prípadu naučenia.

Ďalšie štyri stĺpce riadka (nepovinné pri zmene typu End) definujú pozíciu a rozmery obdĺžnika, v ktorom sa nachádza herný prvok. Nasleduje časová pečiatka zmeny v štandardnom formáte ISO 8601, a v prípade, že má herný prvok parametre, voliteľne dlhý zoznam jeho parametrov. Príklad logu zmien prvkov grafického používateľského rozhrania:

Begin,StaminaBar,408,20,768,80,2014-11-06T18:57:10.2402260+01:00

Begin,Berry:1,508,596,0,0,2014-11-06T18:57:14.5294713+01:00

Change,Berry:1,502,591,12,10,2014-11-06T18:57:14.5624732+01:00

5.4 Spracovanie vstupných údajov

Ako sme opísali v kapitole 5.3, nástroj Geddit prijíma pri základnom príklade vyhodnocovania jedného prípadu naučenia nad údajmi z hrania jedného hráča štyri rôzne zdroje údajov. Jedným z nich je zápis prípadu naučenia. Zvyšné tri reprezentujú časovo opečiatkované logy o priebehu hry v predpísanom formáte.

Každý z týchto logov nesie odlišnú základnú informáciu – kde sa hráč pozeral a čo robil na úrovni vstupov (údaje z nástroja Tobii Studio), čo sa v hre dialo na vyššej logickej úrovni (log vlastných logických udalostí) a kde sa nachádzali jednotlivé herné prvky na obrazovke. Je zjavné, že naše riešenie potrebuje na ďalšiu prácu so zozbieranými údajmi vykonať ich zjednotenie do jedného zdroja herných udalostí.

Túto úlohu spĺňa v aplikácii Geddit modul predprocesora. Predprocesor transformuje logy o priebehu hry do jedinej sekvencie herných udalostí ktorá neobsahuje žiadne redundantné údaje, nepotrebné pre účel porovnávania nameraných údajov s prípadom naučenia.

Predprocesor spracúva logy o priebehu hry sekvenčne – vždy spracuje zo všetkých troch súborov riadok s najnižšou časovou pečiatkou, a podľa typu zdrojového logu buď eviduje zmenu grafického používateľského rozhrania, eviduje vlastnú logickú udalosť, alebo spája pohľadovú či vstupnú udalosť s prvkom grafického používateľského rozhrania ktorého sa udalosť týka.

Vzhľadom na nedokonalú presnosť zariadení na sledovanie pohľadu, náš Geddit predprocesor určuje, či fixácia pohľadu dopadá na prvok grafického používateľského rozhrania na základe maximálnej tolerovanej vzdialenosti bodu fixácie a obdĺžnika okupovaného herným prvkom. Táto hodnota maximálnej tolerovanej vzdialenosti je parametrom našej metódy a je možné ju nakonfigurovať podľa charakteristík zariadenia na meranie pohľadu, vlastností nášho grafického používateľského rozhrania, alebo iných potrieb merania.

Preprocesor tiež zabezpečuje transformáciu prípadov naučenia vo formáte UXF na množinu všetkých možných ciest jeho diagramom aktivít. Predprocesor najskôr transformuje grafický zápis diagramu aktivít (kde sú hrany a uzly diagramu určené svojimi vektorovými súradnicami) na logický model diagramu. V druhom kroku preprocesor exploratívnu rekurziou cez všetky rozhodovania a paralelizmy vypočíta všetky možné cesty diagramom vo forme samostatných sekvencií.

5.5 Výpočet metrík podobnosti

Komponent predprocesora vykonáva transformáciu vstupných údajov na log čistých nameraných herných udalostí a na množinu všetkých možných ciest cez diagram prípadu naučenia. Logicky nasledujúcim krokom je samotný výpočet metrík podobnosti týchto reťazcov, a výber tej najmenej vzdialenej cesty cez prípad naučenia, vyskytujúcej sa v logu nameraných herných udalostí.

Štandardný algoritmus na výpočet Damerau-Levenshteinovej vzdialenosti je prispôsobený na výpočet jedinej metriky vzdialenosti ako súčtu Damerau-Levenshtein operácií. Ako sme však diskutovali v kapitole návrhu našej metódy, tento prístup nie je v našom prípade vhodný

v súvislosti rôznou váhou jednotlivých Damerau-Levenshtein operácií na vzdialenosť cesty cez prípad naučenia od nameraných údajov. Okrem toho je tiež tento algoritmus priestorovo náročný – $O(N^2)$ – kde N dĺžka vstupných reťazcov.

Vzhľadom na tieto fakty sme museli vytvoriť našu vlastnú špecializovanú verziu algoritmu na výpočet množstva Damerau-Levenshtein operácií. Z nameraných udalostí sa najskôr odstráni všetky udalosti nevyskytujúce sa v porovnáwanej ceste cez prípad naučenia a spočítajú sa ako adície pre nasledujúcu relevantnú hernú udalosť. Všetky predpisy akcií vyskytujúce sa v ceste cez prípad naučenia a chýbajúce v nameraných akciách po priradení hodnôt identifikátorov do zástupných symbolov sa spočítajú ako delécie. Počet transpozícií sa vypočíta porovnaním cesty cez prípad naučenia s výslednou upravenou sekvenciou, pričom počet transpozícií pre každú akciu je rovný počtu pozícií v prechode prípadom naučenia o ktoré sa posunul.

6 Overenie LCM metódy

Vzhľadom na výskumný charakter nami vyvinutej metódy dolovania prípadov naučenia (ďalej „LCM metódy“), jej tvorba bola od začiatku sprevádzaná viacerými experimentami. Pri iničiálnom návrhu metódy sme vykonali jednoduchý experiment s nástrojom Tobii Studio, ktorého jediným cieľom bolo získanie skúseností s technológiou sledovania pohľadu, jej možnosťami a ohraničeniami. Na začiatku implementácie nástroja Geddit sme vykonali experiment, ktorého predmetom bol zber jednoduchých ale reálnych údajov z priebehu hrania hry pre umožnenie testovania medzi-výstupov počas vývoja.

V nasledujúcich sekciách opisujeme experiment, ktorým sme overovali uplatniteľnosť LCM metódy pre automatizované vyhodnocovanie naučiteľnosti počas *playtestov* (overovanie hier na základe prirodzenej interakcie medzi hrou a hráčom). Výsledky ktoré sme prostredníctvom metódy LCM dosiahli popisujeme a následne podrobne diskutujeme.

6.1 Experiment

Vzhľadom na motiváciu návrhu našej metódy sme určili dve hlavné hypotézy, ktoré sú predmetom nášho overovania:

H1: Na základe našej metódy dokážeme určiť, či majú herné mechaniky alebo dynamiky nedostatočnú naučiteľnosť

H2: Na základe našej metódy dokážeme ukázať na príčinu nedostatočnej naučiteľnosti herných mechaník alebo dynamík

Keďže naše hypotézy sú všeobecné tvrdenia týkajúce sa hier v abstraktnej rovine, je prirodzené, že našu metódu sme museli overovať na minimálne dvoch alebo viacerých hrách. Pre každú z týchto hier sme si ďalej definovali aspoň jeden prípad naučenia o ktorom sme vedeli, že má nedostatočnú naučiteľnosť, a jeden prípad naučenia, o ktorom očakávame, že je dobre naučiteľný, s predpokladom, že touto voľbou prípadov naučenia overíme, či LCM metóda neposkytuje falošné pozitívne alebo falošné negatívne výsledky.

Nás experiment sme vykonali ako *playtest*, počas ktorého boli zbierané údaje o priebehu hry potrebné na uplatnenie LCM metódy. Experiment prebiehal na zariadení vybavenom softvérom Tobii Studio a hardvérovým vybavením na sledovanie pohľadu. Pre viac detailov o príprave aj priebehu hlavného aj pilotného experimentu, pozri prílohu D. V nasledujúcich častiach odôvodňujeme najdôležitejšie rozhodnutia v návrhu experimentov, ako aj analyzujeme vybrané hry a prípady naučenia, na ktorých sme vykonávali overenie.

Keďže pre potvrdenie našich hypotéz sme potrebovali ukázať, že LCM metóda umožňuje správne vyhodnotiť problémy naučiteľnosti, alebo aspoň na ne poukázať porovnateľne s technikami na zber spätnej väzby bežne používanými počas *playtestov*, pri našom *playteste* sme uplatnili aj takéto techniky. Pridanou hodnotou LCM metódy je, že neruší normálny priebeh hry ako napríklad pozorovanie a protokol premýšľania nahlas, preto sme ako jednu z techník na zber explicitnej spätnej väzby použili review videa z behu *playtestu*, pri ktorom účastníci dostali priestor rozprávať o svojom vnútornom prežívaní počas hrania hry. Ako druhú

techniku pre doplnenie dodatočných informácií o úspešnosti naučenia jednotlivých herných mechaník sme použili interview.

Pre vykonanie overenia sme vybrali dve jednoduché hry, ktorých vývoj nebol dokončený, a sú v takom stave, že je pre ne reálne užitočné vykonať playtesting.

Prvá hra, na ktorej sme vykonali overenie našej metódy sa nazýva TailTrails (obrázok 6.1). Jedná sa o detskú edukačnú-logopedickú hru, v ktorej hráč ovláda na obrazovke postavičku hadíka a snaží sa nazbierať čo najviac bodov. Hadík má určité množstvo energie ktorá sa mu postupne míňa. Hráč môže hadíkovi energiu doplniť zbieraním bobuliek, ktoré hovoria slová obsahujúce konkrétnu hlásku, alebo môže energiu ešte aj stratiť, pokiaľ zbiera nesprávne bobule, ktoré túto hlásku nevyslovujú. Hra sa skončí, ak hadík narazí do svojho chvosta, alebo stratí všetku energiu.

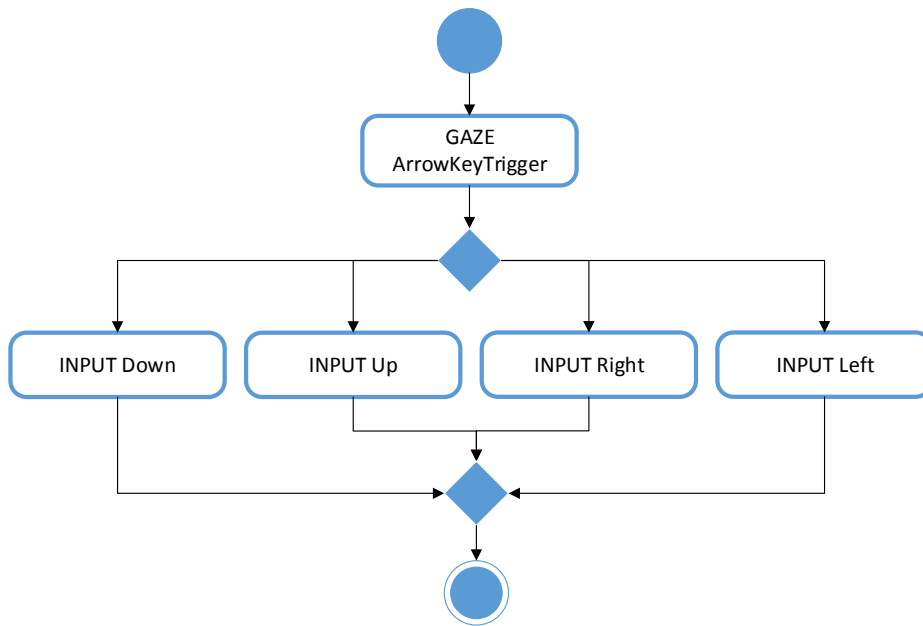


Obr. 6.1 Hra TailTrails – prvá hra použitá na overenie

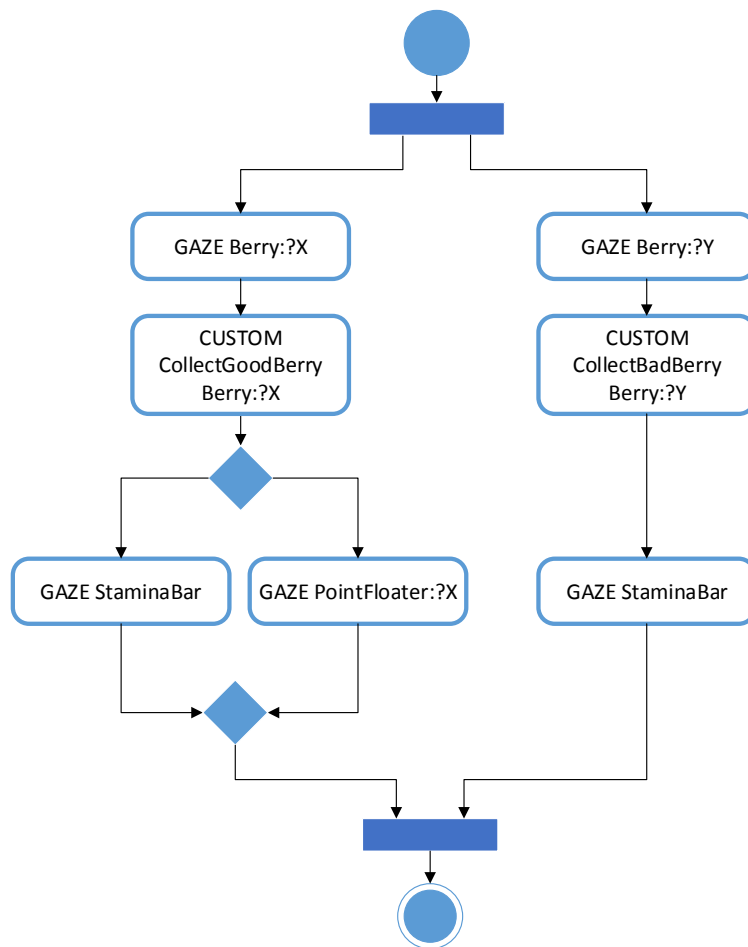
Pre hru TailTrails sme navrhli dva prípady naučenia predstavujúce dve základné herné mechaniky, ktoré musia hráči zvládnuť – ovládanie šípkami a mechanika zbierania bobuliek

Prípado naučenia #1 Ovládanie hry šípkami (obrázok 6.2). Pre mechaniku ovládania hry šípkami očakávame, že je ľahko naučiteľná, nakoľko predchádzajúci hráči s jej pochopením nemali problémy. Jej prípad naučenia pozostáva z pohľadu na indikátor ukazujúci šípky na klávesnici, a stlačenia jednej zo šípok na klávesnici, čo spustí pohyb hadíka po obrazovke.

Prípado naučenia #2 Mechanika zbierania bobuliek (obrázok 6.3). Táto mechanika, tak ako sme si ju zadefinovali, sa sústreďí na hráčovo pochopenie, že na hracej ploche sa vyskytujú dva typy bobuliek, ktoré majú rôzne efekty po zozbieraní - podľa toho či sú dobré alebo zlé. Toto je hlavná príčina, prečo tvorí zbieranie dobrých a zlých bobuliek spojený prípad naučenia, ktorý nebol rozdelený do dvoch samostatných prípadov naučenia pre každú situáciu zvlášť. Pre tento prípad naučenia predpokladáme horšiu naučiteľnosť hlavne v oblasti vnímania získavania a strátenia energie hráčmi, nakoľko predchádzajúci hráči mali tendenciu energiu si veľmi nevšimáť. Všimnime si, že samotné pochopenie hráčov, čo robí dobré bobulky dobrými a zlé zlými, nie je predmetom tohto prípadu naučenia a takáto vedomosť známa dopredu by mohla spôsobiť, že hráč by sa vyhýbal už prvému zbieraniu zlej bobule, čo protirečí tomuto prípadu naučenia.

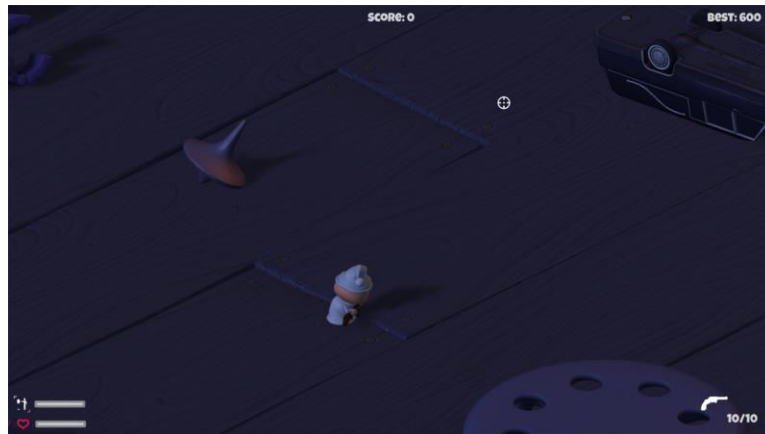


Obr. 6.2 Prípad naučenia #1 Ovládanie hry šípkami



Obr. 6.3 Prípad naučenia #2 Mechanika zbierania bobuliek

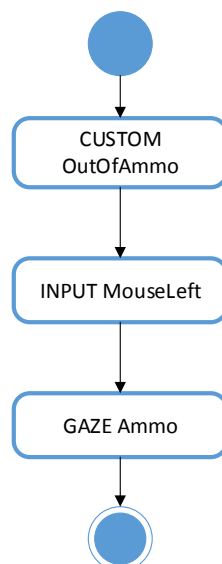
Druhá hra, ktorú sme využili na overenie sa nazýva Nightmares (obr. 6.4). Jedná sa o vylepšenú verziu hry o prežitie vytvorenej na základe tutoriálu⁵ na tvorbu hier v rámci Unity. V rámci tejto hry sa hráč pohybuje po ohraničenej mape a snaží sa prežiť vlny objavujúcich sa nepriateľov, po ktorých môže strieľať pištoľou. Ďalšie relevantné herné mechaniky diskutujeme v opise prípadov naučenia



Obr. 6.4 Hra Nightmares – druhá hra použitá na overenie

Pre hru Nightmares sme definovali celkovo tri prípady naučenia:

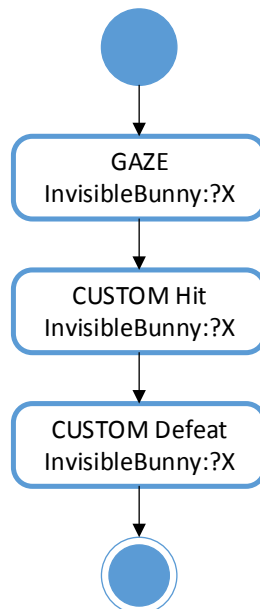
Prípad naučenia #3 Míňajúce sa náboje (obrázok 6.5). Hráč má obmedzený počet nábojov, po ktorých vystrieľaní ich musí opätovne nabiť, čo sa vykoná automaticky, ale trvá to niekoľko sekúnd. Počas tejto doby očakávame, že hráč vyhladá v rozhraní informáciu o počte nábojov. Od tohto prípadu naučenia očakávame dobrú naučiteľnosť, nakoľko strieľanie je v hre okrem pohybu hlavnou akciou hráča a indikátor o počte nábojov je pomerne dobre čitateľný.



Obr. 6.5 Prípad naučenia #3 Míňajúce sa náboje

⁵ Survival Shooter Unity tutoriál: <http://unity3d.com/learn/tutorials/projects/survival-shooter>

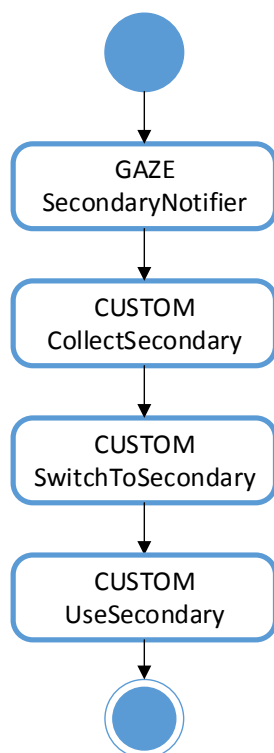
Prípado naučenia #4 Neviditeľní nepriatelia (obrázok 6.6). Na mape sa občas objavia polopriehvitní nepriatelia, ktorí sa správajú rovnako ako obyčajní nepriatelia, ale sú slabšie viditeľní. Prípado naučenia od hráčov očakáva, že si týchto nepriateľov hráči všimnú a budú po nich strieľať až kým ich neporiazia, tak isto ako pri bežných nepriateľoch. Od tohto prípadu naučenia očakávame slabšiu naučiteľnosť ako v prípade naučenia #3, nakoľko hráči nie sú na prítomnosť slabšie viditeľných nepriateľov žiadnym spôsobom upozorení.



Obr. 6.6 Prípado naučenia #4 Neviditeľní nepriatelia

Prípado naučenia #5 Sekundárne zbrane (obrázok 6.7). Na mape sa počas hry zvyknú časom zobrazit' dodatočné zbrane, ktorými si hráči dokážu pomôcť (puška, samopal, bomba). Tento prípad naučenia predpokladá, že hráč si všimne, keď na mape sa objaví takáto zbraň, vyhľadá ju, prepne si ju a bude ju používať. Pre tento prípad naučenia predpokladáme najhoršiu naučiteľnosť, nakoľko hráči sa žiadnym spôsobom nedozvedia, že môžu používať iné zbrane okrem pištole, a aj po nájdení ďalších zbraní ich hra žiadnym spôsobom neinformuje, ako si ďalšiu zbraň prepnúť a použiť ju.

Pri zbere explicitnej spätnej väzby cez interview po playteste sme zber informácií o priebehu hrania hry cielili tiež prevažne na vyhodnotenie úspešnosti vyššie menovaných prípadov naučenia. Pri review videa pred interview boli hráči ponechaní rozprávať o svojom vnútornom prežívaní počas behu hry. Pokiaľ tento monológ sám nespresnil, či sa u hráča nevyskytol počas hry problém naučiteľnosti v konkrétnych herných mechanikách, nasledovali otázky v interview, ktorých cieľom bolo dozvedieť sa viac. Pri kladení otázok sme dbali na to, aby otázka nevnucovala účastníkovi odpoveď, ktorú by sami nepovedal (Príklad: *Aké typy nepriateľov si z hry pamätáš?* namiesto *Videl si tých neviditeľných nepriateľov?*)

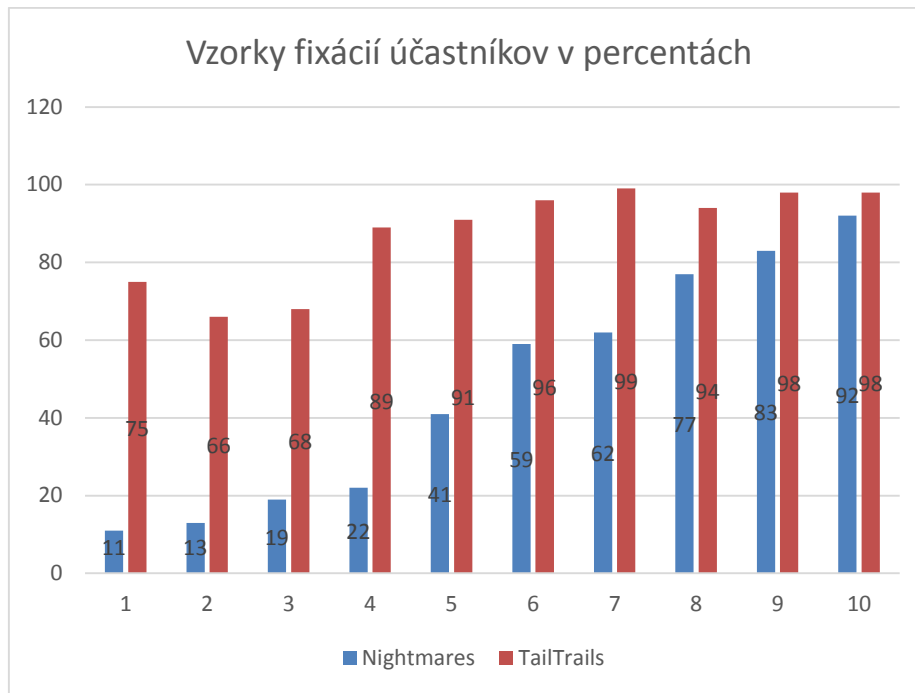


Obr. 6.7 Prípád naučenia #5 Sekundárne zbrane

Experiment sme celkovo vykonali s desiatimi účastníkmi, čo je počet, ktorý sme zvolili prevažne vzhľadom na rozsiahle časové nároky na zber a vyhodnocovanie explicitnej spätnej väzby od účastníkov. Tento počet je tiež podobný s rôznymi odlišnými experimentami zameranými na sledovanie pohľadu, tak ako boli vykonané inými stranami (viď. kapitola 3, sekcia 3.3).

6.2 Vyhodnotenie výsledkov metódy

Uskutočnením experimentu sme získali dostatočné množstvo údajov na kvantitatívne vyhodnotenie prípadov naučenia na základe LCM metódy. Zaujímavosťou je, že pri hre Nightmares nástroj Tobii Studio zachytil znížený pomer fixácií pohľadu (obrázok 6.8), hoci podmienky pre účastníkov boli pri hraní oboch hier rovnaké. Ako však ukážeme v tejto kapitole, až na niektoré výnimky (ktorým sa venujeme ešte viac v diskusii) bolo možné uplatniť metódu LCM aj na takúto vzorku údajov a dosiahnuť tak výpovedné výsledky.

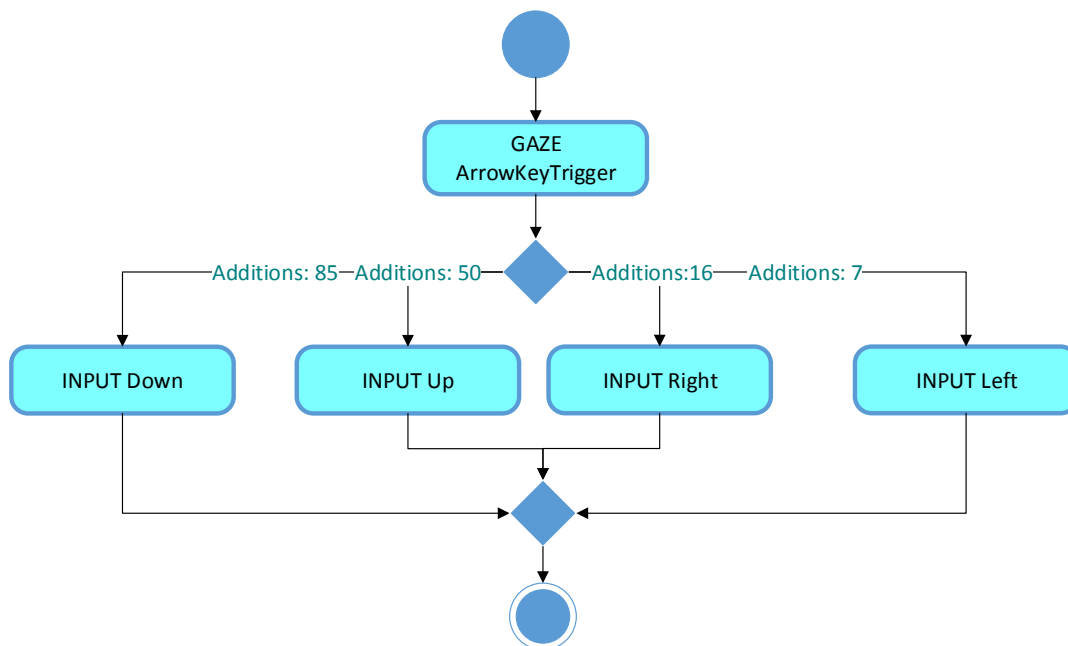


Obr. 6.8 Pomer zachytených fixácií jednotlivých účastníkov je pri hre Nightmares spravidla nižší, ako pri hre TailTrails

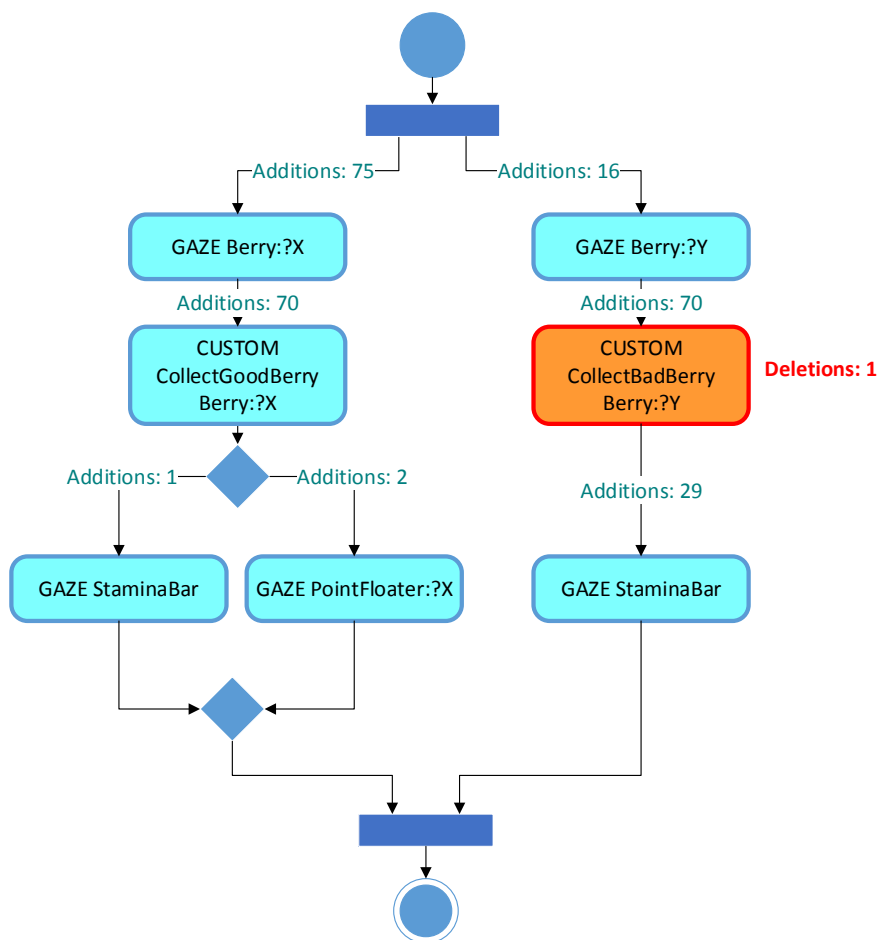
Po aplikácii LCM metódy sme získali vyhodnotenú metriku adícií, transpozícií a delécií, ktoré reprezentujú aký je minimálny potrebný počet takýchto operácií pre transformáciu sekvencie udalostí z hrania hry na prechod prípadom naučenia. Tieto metriky sme zosumarizovali pre jednotlivé predpisy udalostí v prípade naučenia a vizualizovali sme ich na prípade naučenia, kde zvýšené hodnoty metrík znázorňujeme teplými farbami. Všimnime si, že hodnoty metrík žiadnym spôsobom nenormujeme, nakoľko mierny rozdiel medzi hodnotami metrík pri jednom páre udalostí môže mať vyšší význam než rádovo väčší rozdiel, ktorý bol však aj dopredu očakávaný (Príklad: *V strategickej hre medzi postavením budovy kasárni a stavaním armády hráč vykonáva mnoho akcií, lebo čaká na dokončenie stavby kasárne*).

Prípad naučenia #1 Ovládanie hry šípkami sa po vyhodnotení (obr. 6.9) javí taký bezproblémový, ako sme očakávali. Hráči nemali problém všimnúť si indikátor, že hra sa ovláda šípkami, a krátko nato šípky aj stlačali. Zaujímavosťou je, že 8 z 10 hráčov začali hru pohybom po vertikálnej rovine, a len dvaja hráči začali hru stlačením šípok na horizontálnej rovine, teda doprava alebo doľava. Mechanika ovládania je teda dobre naučiteľná.

Prípad naučenia #2 Mechanika zbierania bobuliek po vyhodnotení (obr. 6.10) obsahuje jednu deléciu udalosti zbierania zlej bobule. Môžeme teda predpokladať, že jeden z účastníkov pochopil, že existujú dobré a zlé bobule pred tým, ako prebehlo učenie pokusom a omylom. V tomto prípade delécia nie je nutne závažná.



Obr. 6.9 Vyhodnotenie prípadu naučenia #1



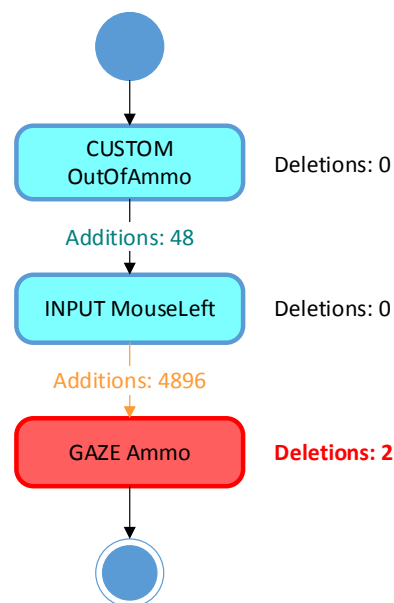
Obr. 6.10 Vyhodnotenie prípadu naučenia #2

Dodatočnou informáciou, ktorá je vo vyhodnotení prípadu naučenia #2 zaujímavá z hľadiska naučiteľnosti, ale nepochádza priamo z nami definovaných metrik je, že pri vetvení hernej spätnej väzby, ktorú mali hráči spracovať po zozbieraní dobrej bobule, väčšina hráčov sústredila pozornosť na získané body, a nie na získanú energiu.

Z tohto pozorovania si môžeme odvodiť dva odlišné závery:

1. Pokiaľ by nás pri vetvení zaujímalo, ktorú cestu si hráči často nevyberajú, mohli by sme naše metriky doplniť o počet navštívení každého predpisu udalosti. Takáto metrika by nebola informačne totožná s metrikou delácií, nakoľko tá by bola predpisu udalosti na vetve zvyšovaná iba v prípade, že hráč nevykonal udalosť na žiadnej z vetiev. Takáto metrika však nemá pôvod v podobnosti reťazcov, ale skôr v ceste grafom, a preto túto úvahu ponechávame ako podnet na ďalšiu prácu.
2. Keď zväžíme fakt, že pôvodným úmyslom vetvenia v prípadoch naučenia nebolo odlišovať vetvy, ale umožniť hráčom správať sa rôznymi spôsobmi, pričom každý by bol akceptovaný ako správny, potom žiadnu dodatočnú metriku netreba pridávať. Na vyhodnocovanie naučiteľnosti mechaniky energie by bolo vhodné zaviesť samostatný prípad naučenia. Toto je náš finálny záver, nakoľko potvrdzuje, aký dôležitý je mať premyslený návrh prípadov naučenia pred spustením playtestu.

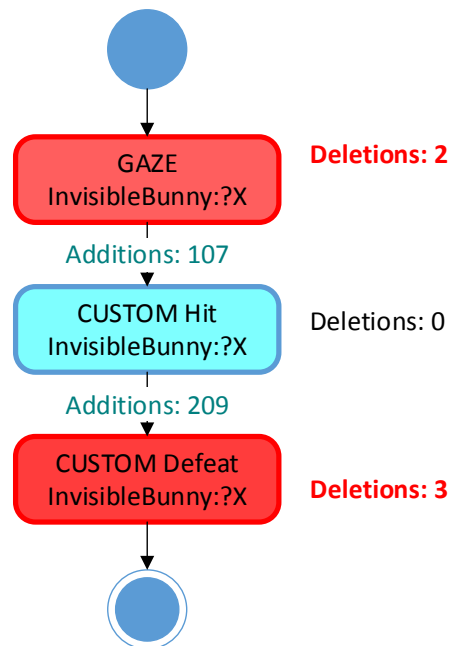
Prípad naučenia #3 Míňajúce sa náboje sme pôvodne považovali za hráčmi dobre naučiteľný, ale po jeho vyhodnotení (obr. 6.11) vidíme, že aj jeho naučiteľnosť má svoje nedostatky. Konkrétne mali hráči problémy so všimnutím si počtu nábojov na obrazovke, kde dvaja hráči si ukazovateľ nábojov nevšimli vôbec, a polovica ostatných používateľov ho hľadala významne dlhšie, o čom svedčí aj zvýšená sumatívna hodnota adícií pred predpisom udalosti pohľadu na ukazovateľ nábojov. Čisto na základe LCM by sme preto mali zväžiť zvýraznenie tohto ukazovateľa, a to zvlášť pri úplnom minutí nábojov a počas nabíjania zbrane.



Obr. 6.11 Vyhodnotenie prípadu naučenia #3

Prípado naučenia #4 Neviditeľní nepriatelia sa po svojom vyhodnotení(obr. 6.12) prejavil ako problematický v dvoch bodoch:

1. Dvaja hráči si neviditeľných nepriateľov nevšimli vôbec počas celého playtestu. To, že po nich aj tak vystrelili je možné pripísať faktu, že hráči boli počas hry často prenasledovaní veľkým množstvom nepriateľov, a preto niektoré hráčove strely určené pre viditeľných nepriateľov sa mohli ujsť aj tým neviditeľným. V tomto prípade by sme mohli zvážiť vložiť do hry sekciu, v ktorej sú hráči upozornení na prítomnosť neviditeľných nepriateľov, pričom ich stupeň priesvitnosti môže byť zachovaný, a tým sa zachová aj výzva, ktorú slabšie viditeľní nepriatelia predstavujú.
2. Trom hráčom sa počas playtestu vôbec nepodarilo poraziť žiadneho z neviditeľných nepriateľov. Keďže iba dvaja z hráčov nevedeli o prítomnosti neviditeľných nepriateľov, tento fakt nám napovedá, že by sme mali u nepriateľov posilniť spätnú väzbu, ktorú hráčovi poskytujú, keď sú strelení. Počas sekcie, počas ktorej sú hráčovi títo špeciálni nepriatelia predstavení by tiež mal hráč dostať priestor čeliť iba jednému takémuto nepriateľovi, a overiť si, že je ho možné dostupnými zbraňami poraziť.

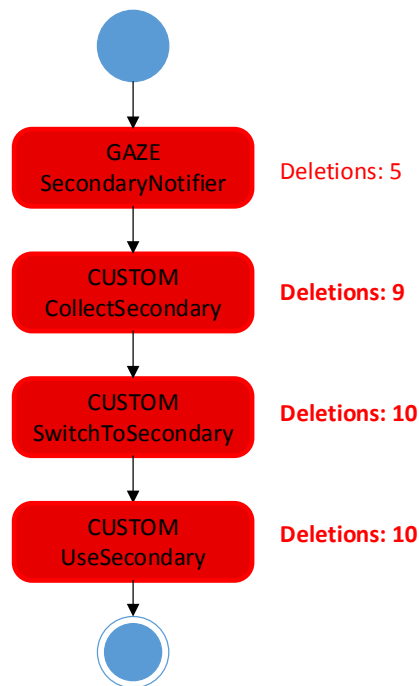


Obr. 6.12 Vyhodnotenie prípadu naučenia #4

Prípado naučenia #5 Sekundárne zbrane sa po vyhodnotení(obrázok 6.13) ukázal tak problematický, ako sme pred vykonaním experimentu očakávali. Iba polovica z hráčov si vôbec všimla, že na mape sa zjavila zbraň, ktorú by mohli zobrať. Iba jednému sa pritom podarilo takúto zbraň zobrať, avšak nevedel si ju prepnúť a ani si ju tým pádom nevyskúšal použiť.

Na základe výstupu LCM zvažujeme, že by bolo vhodné pri učení mechaník sekundárnych zbraní hráčovi poskytnúť zbraň niekde v jeho blízkosti, a tiež tento objekt na mape značne zvýrazniť, minimálne v rámci tejto iniciálnej sekcie, aby si hráč zbraň nepomýlil so súčasťou

nehybného pozadia. Hráčom by malo byť tiež po prvom zozbieraní zbrane odkomunikované, ako si zbraň prepnúť.



Obr. 6.13 Vyhodnotenie prípadu naučenia #5

Pokiaľ si spätne prezrieme vyhodnotenia všetkých piatich prípadov naučenia, všimneme si, že obsahujú iba operácie adícií a delácií. Toto je spôsobené faktom, že pre naše zvolené prípady naučenia buď udalosti chýbali úplne, alebo sa v celkovej postupnosti udalostí našiel taký prechod prípadom naučenia, ktorý bol usporiadaný v správnom poradí. Predpokladáme však, prítomnosť transpozícií je stále možná, a závisí od viacerých faktorov, ako je samotná herná mechanika, návrh prípadu naučenia, ale aj dĺžka samotného playtestu

6.3 Porovnanie s explicitnou spätnou väzbou a diskusia

V predchádzajúcej sekcii sme analyzovali výsledky, ku ktorým sme dospeli aplikovaním metódy LCM počas playtestu. Túto sekcii venujeme konfrontácií zistených výsledkov s vyhodnotením naučiteľnosti na základe explicitnej spätnej väzby zozbieranej cez review videa a interview po ukončení playtestu.

Pri vyhodnocovaní explicitnej spätnej väzby sme pre každú hernú mechaniku, pre ktorú sme mali navrhnutý prípad naučenia samostatne zozbierali a spracovali informácie o tom, ako ju hráč zvládol, a porovnali výsledky s jeho najlepším nájdeným prechodom cez prípad naučenia.

Mechanika ovládania hry TailTrails šípkami bola na základe explicitnej spätnej väzby naučená hráčmi veľmi jednoducho – väčšina hráčov okamžite siahla po šípkach. Iba jeden z hráčov sa pokúsil najprv kliknúť na indikátor šípok na obrazovke, a aj on rýchlo pochopil, že musí použiť šípky na klávesnici. Tento výsledok je prakticky totožný s výsledkom, ktorý sme dosiahli aplikovaním LCM.

Mechanika zbierania bobúľ v hre TailTrails ukázala tiež podobné výsledky ako vyhodnotenie na základe LCM. Väčšina hráčov pochopila, že hra obsahuje dobré a zlé bobule, Jeden z účastníkov z kategórie *hardcore* hráčov (hráčov s najviac skúseností s hrami), ktorý podľa interview vedel, že sa jedná o logopedickú hru sa dokonca bez potreby učenia pokusom a omylom hneď od začiatku vyhýbal bobuliám, o ktorých si myslel, že sú zlé, čo sa odráža vo vyhodnotení LCM jednou deléciou v príslušnom prípade naučenia.

Iným významným prípadom je iný účastník z kategórie *casual* hráčov (hráčov s málo skúsenosťami s hrami) u ktorého zjavne LCM metóda nad týmto prípadom naučenia vyvolala falošne pozitívny výsledok. Jeho prechod cez prípad naučenia neobsahoval žiadne problematické časti, avšak na základe interview s hráčom vieme, že hráč z hry nepochopil, že hra obsahuje dobré a zlé bobule. Tento výsledok však môžeme pripísať ako chybu nie samotnej LCM metódy, ale iba chybu v dizajne tohto konkrétneho prípadu naučenia. Pri pohľade na prípad naučenia #2 si môžeme uvedomiť, že obe paralelné cesty, ktoré mal byť hráč na playtestu konci schopný odlíšiť sú si veľmi podobné, a predpisy akcií ktorými sa odlišujú nemajú nič spoločné s vnímaním hráča (jedná sa o zbieranie bobúľ, ktoré sa pri dobrých aj zlých bobuliach koná rovnako). Navrhujeme preto prípad naučenia vylepšiť tak, aby obsahoval pohľadové udalosti na bobule v čase, kedy poskytujú hráčovi vizuálnu spätnú väzbu o tom, či boli dobré alebo zlé.

Počas hrania hry Nightmares sme pozorovali u hráčov znížený pomer zachytených fixácií než pri hre TailTrails. Hoci u väčšiny hráčov sa nízke množstvo fixácií neprejavilo ako závažný problém, u jedného hráča konkrétne tento jav spôsobil, že všetky udalosti pohľadu boli u neho označené ako delécie. Na základe explicitnej spätnej väzby sme zistili, že všetky prvky, ktoré si tento hráč na základe LCM nevšimol, si hráč v skutočnosti všimol. Na základe tejto skúsenosti môžeme tvrdiť, že kvalitný záznam pohľadu je pre udalosti pohľadu dôležitý, a v opačnom prípade môže spôsobiť falošné negatívne výsledky, zvlášť ak by sa problém opakoval u viacerých hráčov.

Mechanika míňajúcich sa nábojov v hre Nightmares bola hráčmi naučená rovnako dobre na základe oboch zdrojov spätnej väzby – hráči ktorí tvrdili, že si nevšimli indikátor nábojov si ho nevšimli ani podľa LCM. U hráčov s vyššími skúsenosťami s hrami sme tiež spozorovali vyššiu tendenciu pamätať si, koľko nábojov mali v zásobníku.

Podobnosť medzi výsledkami LCM a video review a interview sme pozorovali aj pri vyhodnocovaní naučiteľnosti mechaniky neviditeľných nepriateľov. Väčšina hráčov, od ktorých sme zistili, že si neviditeľných nepriateľov všimli, si ich všimla aj na základe LCM. Pri tejto mechanike sme však identifikovali aj falošné pozitívne LCM vyhodnotenie, nakoľko keď hráči sledovali pohľadom hordu nepriateľov, ktorá ich prenasledovala, často sa fixácia ich pohľadu nachádzala v blízkosti neviditeľných nepriateľov, ktorých si však nevšimli. U jedného z hráčov, u ktorého sme cez LCM zistil, že neporazil žiadneho neviditeľného nepriateľa sme sa dozvedeli, že si myslel, že týchto nepriateľov nedokáže poraziť, čo tiež korešponduje so závermi, ktoré sme urobili na základe vyhodnotenia LCM metódou.

Napokon, mechanika sekundárnych zbraní sa na základe explicitnej spätnej väzby ukázala z hľadiska naučiteľnosti taká chybná, ako ju vyhodnotila metóda LCM. Spravidla všetci *core*

a *hardcore* hráči si všimli notifikátor, že na mape sa zjavila nová zbraň. Všetci až na jedného však tieto zbrane nezobrali, keďže buď v hre prioritizovali iné úlohy (utekanie pred nepriateľmi), nepodarilo sa im zbraň nájsť, alebo ich prospekt novej zbrane jednoducho dostatočne nezaujal. Jediný hráč ktorému sa podarilo zozbierať sekundárnu zbraň v danom momente nevedel, ako túto zbraň použiť a použitie skrolovania na myši mu napadlo až počas priebehu interview. Závěry, ku ktorým sme dospeli na základe explicitnej spätnej väzby sú v podstate totožné so závermi odvodenými na základe LCM: prvýkrát je potrebné hráčovi poskytnúť sekundárnu zbraň niekde v jeho blízkosti, aby si ju mohol v pokoji vyskúšať, a je potrebné mu vysvetliť, ako ju má použiť.

Keďže sme vo väčšine prípadov prišli na základe explicitnej spätnej väzby k rovnakým problémom naučiteľnosti ako za pomoci LCM, a nesúlady medzi LCM a konvenčnými technikami používanými v rámci playtestingu dokážeme zdôvodniť nedokonalým návrhom prípadov naučenia alebo nedostatkom údajov zo sledovania pohľadu, prichádzame k záveru, že náš experiment potvrdil naše hypotézy, a LCM metóda skutočne dokáže poukázať na problémy naučiteľnosti herných mechaník alebo dynamík, ako aj presne poukázať na bod, v ktorom daná herná mechanika či dynamika zlyháva.

7 Zhodnotenie

Predmetom tejto práce je metóda dolovania prípadov naučenia, ďalej LCM, ktorá je určená ako nástroj pre herných vývojárov na vyhodnocovanie iniciálnej naučiteľnosti hier počas ich vývoja. Okrem návrhu LCM metódy sme sa v tejto práci zaoberali aj jej zapojením do veľkého obrazu, ktorý predstavuje proces vývoja herného softvéru. Metóda LCM zapojená do playtestov (overovanie hier na základe reálnej interakcie hráčov s hrou) je novým cenným zdrojom spätnej väzby pre herných vývojárov. Dosiachnutie dobrej naučiteľnosti má pre hry kľúčovú rolu, zvlášť vzhľadom na množstvo dostupných hier, a na obmedzený čas, ktorý je hráč ochotný venovať na prekonanie bariéry pri učení sa novej hry.

Hlavným prínosom LCM je, že sa jedná o automatizovateľnú metódu, ktorá šetrí zdroje a čas potrebné na rigorózný zber spätnej väzby či už cez pozorovanie, rozhovory alebo review videa z playtestov. LCM je tiež založená na implicitnej spätnej väzbe, a teda neruší normálny priebeh hrania hry, čo je pri vyhodnocovaní hier zvlášť dôležité. LCM ako kľúčový zdroj implicitnej spätnej väzby používa technológie sledovania pohľadu, a tým využíva výskumne zaujímavý alternatívny zdroj informácií na pozorovanie mentálnych procesoch hráča.

Jadro metódy LCM je založené na porovnávaní modelového správania sa hráča pri učení sa hry s údajmi o reálnom správaní hráčov počas playtestov. Modelové správanie hráčov je hernými dizajnérmí definované pred začiatkom playtestu pre jednotlivé testované herné mechaniky alebo dynamiky, a to vo forme prípadov naučenia, ktoré sa zapisujú vo forme UML diagramov aktivít.

Našu metódu sme overili v experimente s 10 účastníkmi, kde sme dokázali, že metóda LCM dosahuje podobné výsledky ako techniky na zber explicitnej spätnej väzby bežne zaužívané počas playtestov. Všetky rozdiely medzi výsledkami z LCM, a výsledkami ktoré sme získali na základe review videa a interview s hráčmi dokážeme odôvodniť ako nedokonalý dizajn individuálnych prípadov naučenia, alebo nedostatočnou kvalitou nameraných údajov zo sledovania pohľadu.

Vyhodnocovanie hier je téma, ktorá je výskumne aj komerčne veľmi zaujímavá. V prípade, že ich dokážeme správne interpretovať, údaje zo sledovania pohľadu sa javia v tejto oblasti ako vysoko prínosné. V LCM metóde vidíme priestor na rozličnú ďalšiu prácu, ktorá by mohla ďalej rozšíriť jej možnosti. Jednou z možností je integrácia dodatočných metrik sledovania pohľadu, napríklad na vyhodnotenie výraznosti konkrétnych prvkov herného rozhrania, alebo detekciu situácií, kedy hráč vedel, že nasledujúca akcia súvisí s určitým herným prvkom, ale nevedel presne, ako by mal pokračovať. Pridanú hodnotu by malo aj doplnenie odlišných typov udalostí do LCM metód, ako napríklad udalosť skenovania istej oblasti obrazovky, ak by si herný dizajner želal, aby hráč v istom momente venoval pozornosť istej časti herného rozhrania.

Bibliografia

- [1] J. Schell, *The Art of Game Design: A Book of Lenses*. Morgan Kaufmann Publishers, 2008.
- [2] N. Hidayah, M. Zain, F. Hanis, and A. Razak, "Eye Tracking in Educational Games Environment : Evaluating User Interface Design through Eye Tracking Patterns," in *Visual Informatics: Sustaining Research and Innovations*, 2011, pp. 64–73.
- [3] M. Ernkvist, "Down Many Times , but Still Playing the Game : Creative Destruction and Industry Crashes in the Early Video Game Industry 1971-1986," *Hist. Insolv. Bankruptcy*, pp. 161–191, 2008.
- [4] M. D. Kickmeier-Rust, E. Hillemann, and D. Albert, "Tracking the UFO ' s Paths : Using Eye-Tracking for the Evaluation of Serious Games," in *Virtual and Mixed Reality - New Trends*, 2011, pp. 315–324.
- [5] M. Bétrancourt, "Assessing the educational potential of video games," no. October, pp. 1–18, 2005.
- [6] P. Dulačka, J. Šimko, and M. Bieliková, "Validation of Music Metadata via Game with a Purpose," in *Proceedings of the 8th International Conference on Semantic Systems*, 2012, pp. 177–180.
- [7] R. Hunicke, M. Leblanc, and R. Zubek, "MDA : A Formal Approach to Game Design and Game Research," in *Proceedings of the Challenges in Games AI Workshop, Nineteenth National Conference of Artificial Intelligence*, 2004.
- [8] C. Jennett, A. L. Cox, P. Cairns, S. Dhoparee, A. Epps, T. Tijs, and A. Walton, "Measuring and defining the experience of immersion in games," *Int. J. Hum. Comput. Stud.*, vol. 66, no. 9, pp. 641–661, Sep. 2008.
- [9] M. Csikszentmihalyi, *Flow: The Psychology of Optimal Experience*. New York: Harper & Row, 1990.
- [10] P. Sweetser and P. Wyeth, "GameFlow : A Model for Evaluating Player Enjoyment in Games," *Comput. Entertain. - Theor. Pract. Comput. Appl. Entertain.*, vol. 3, no. 3, pp. 1–24, 2005.
- [11] H. Korhonen, "Expert Review Method in Game Evaluations – Comparison of Two Playability Heuristic Sets," *Proc. 13th Int. MindTrek Conf. Everyday Life Ubiquitous Era*, pp. 74–81, 2009.
- [12] H. Desurvire, W. Blvd, M. Rey, and M. Caplan, "Using Heuristics to Evaluate the Playability of Games," in *Extended Abstracts on Human Factors in Computing Systems*, 2004, pp. 1509–1512.
- [13] J. Nielsen, L. A. Blatt, J. Bradford, and P. Brooks, *Usability Inspection Methods*. 1994.

- [14] T. Grossman, G. Fitzmaurice, and R. Attar, "A Survey of Software Learnability : Metrics , Methodologies and Guidelines," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2009, pp. 649–658.
- [15] H. Korhonen, "Comparison of playtesting and expert review methods in mobile game evaluation," *Proc. 3rd Int. Conf. Fun Games - Fun Games '10*, pp. 18–27, 2010.
- [16] W. S. Yue and N. A. M. Zin, "Usability evaluation for history educational games," *Proc. 2nd Int. Conf. Interact. Sci. Inf. Technol. Cult. Hum. - ICIS '09*, pp. 1019–1025, 2009.
- [17] S. Eivazi and R. Bednarik, "Inferring Problem Solving Strategies Using Eye-Tracking : System Description and Evaluation," *Proc. 10th Koli Call. Int. Conf. Comput. Educ. Res.*, pp. 55–61, 2010.
- [18] T. Tullis and B. Albert, *Measuring the User Experience*, Second Edi. 2013.
- [19] A. Faro, D. Giordano, C. Spampinato, D. De Tommaso, and S. Ullo, "An Interactive Interface for Remote Administration of Clinical Tests Based on Eye Tracking," in *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications*, 2009, pp. 69–72.
- [20] H. Wang, M. Chignell, and M. Ishizuka, "Empathic tutoring software agents using real-time eye tracking," *Proc. 2006 Symp. Eye Track. Res. Appl. - ETRA '06*, p. 73, 2006.
- [21] C. Hennessey, "Long Range Eye Tracking : Bringing Eye Tracking into the Living Room," in *Proceedings of the Symposium on Eye Tracking Research and Applications*, 2012, pp. 249–252.
- [22] M. A. Just and P. A. Carpenter, "A theory of reading: from eye fixations to comprehension.," *Psychol. Rev.*, vol. 87, no. 4, pp. 329–54, Jul. 1980.
- [23] B. Pan and G. K. Gay, "The Determinants of Web Page Viewing Behavior : An Eye-Tracking Study," in *Proceedings of the 2004 symposium on Eye tracking research & applications*, 2004, vol. 1, no. 212, pp. 147–154.
- [24] M. Bartels, "Eye Tracking Insights into Cognitive Modeling," *Proc. 2006 Symp. Eye Track. Res. Appl.*, vol. 1, no. March, pp. 27–29, 2006.
- [25] V. Sundstedt and E. Reinhard, "A Psychophysical Study of Fixation Behavior in a Computer Game," in *Proceedings of the 5th symposium on Applied perception in graphics and visualization*, 2008, vol. 1, no. 212, pp. 43–50.
- [26] E. L. Law, E. E. Mattheiss, M. D. Kickmeier-rust, and D. Albert, "Vicarious Learning with a Digital Educational Game : Eye-Tracking and Survey-Based Evaluation Approaches," in *HCI in Work and Learning, Life and Leisure*, 2010, pp. 471–488.
- [27] W. M. P. van der Aalst, "Conformance Checking," in *Process Mining: Discovery, Conformance and Enhancement of Business Processes*, Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 191–213.

- [28] F. J. Damerau, I. B. M. Corporation, and Y. Heights, "A Technique for Computer Detection and Correction of Spelling Errors," *Commun. ACM*, vol. 7, no. 3, pp. 171–176, 1964.