4 RAZVOJ VIDEO IGER

Veliko je interesa na temo raziskovanja discipline metodologij razvoja video iger (Ampatzoglou in Stamelos 2010, 889). Opaziti je bilo povečano število raziskav na tem področju. Poleg tega aktivnost raziskovanja inženiringa video iger v primerjavi z raziskovanjem razvoja programske opreme raste hitreje (Ampatzoglou in Stamelos 2010, 893). Od leta 2008 je bilo opaziti več študij na temo upravljanja razvoja video iger (Ampatzoglou in Stamelos 2010, 894), ki zajema tehnike upravljanja s procesi (Novak 2012, 351). To potrdi tudi sistematičen pregled literature, ki so ga izvedli Aleem in kolegi (2016).

Razvidno je da je število raziskav doseglo vrhunec med leti 2008-2012 (PRILOGA A). Kljub porastu raziskav smo opazili, da se organizacije poslužujejo različnim pristopom in njihovim hibridom predstavljenih v prejšnjem poglavju. Našli smo le nekaj raziskav, ki so definirale razvojni proces, metodologijo ali življenjski cikel izključno za razvoj video iger. Morda je krivo to, da v literaturi še ni bilo podrobneje predstavljenih najboljših praks ali strategij za razvoj iger. Poleg tega je bila disciplina okrnjena orodja za optimizacijo procesa, ki je ključno za izboljšanje procesa (Aleem, Capretz, in Ahmed 2016, 26). Menimo, da celovito podedovanje procesnih modelov in metod iz standardnega razvoja programske opreme za igre ni primerno. Zaradi svoje agilnosti, tendence po zmanjšanju dokumentacije, multidisciplinarnosti, kreativnosti, abstraktno definiranih zahtev (Ruonala in YLIOPISTO, 5 2016), krajšega življenjskega cikla (O’Hagan in O’Connor 2015, 3) posedujejo lastnosti, ki jih evidentno ločuje od standardnega razvoja programske opreme.

4.1 ZNAČILNOSTI RAZVOJA IGER

Za razvoj iger je jasno, da poseduje značilnosti, ki niso prisotne v ostalih disciplinah inženiringa programske opreme (Abrahamsson in dr. 2017, 1).

Igre se osredotočajo na igranje (Esposito 2005, 3). Igranje predstavlja element, ki emocionalno pritegne igralce. Je nekakšno stanje, ki je podobno zabavi (Fullerton 2008, 91). Posledično se igre smatrajo za najbolj ekstremne hedonične informacijske sisteme, pa vendar se pri utilitarnih namenih uporabe uporabna motivacija premakne iz zabave v uporabnost (Hamari in Keronen 2017, 136).

Igre lahko vodijo tako intrinzične kot ekstrinzične motivacije (Novak 2012, 198). Številne študije kažejo pozitivne učinke učnih iger na področju motivacije do učenja in učnega izkoristka. Motivacija in pozitivne emocije sta pomembna faktorja pri učenju (Imlig-Iten in Petko 2014, 151), katerih uravnoteženost igre predstavlja efektivnost, ki je pridobljena s konsistenco zahtevnosti in zabave pri igranju (Novak 2012, 202). To predstavlja zabavno izkušnjo, ki je najpomembnejše merilo uspeha iger (Cooper in Scacchi 2015, 12). V razvoju iger se uporablja termin uporabniška izkušnja[[1]](#footnote-1), ki jo z iterativnim procesom upoštevanja uporabniških povratnih informacij oblikovalci izboljšujejo tekom procesa razvoja (Fullerton 2008, 2). Potrebno je razumevanje, da igra ni izkušnja vendar je izkušnja nekaj kar igra omogoča (Schell 2008, 10).

S povratnimi informacijami je povezana tudi pozornost na defekte. Zavoljo tega osrednja aktivnost razvoja video iger navadno postane konstantno testiranje[[2]](#footnote-2) igranja (Cooper in Scacchi 2015, 10). To aktivnost predstavlja testni protokol in velja za iterativni proces med načrtovanjem, razvojem in testiranjem (Sylvester 2013, 295). Zato se razvoji iger nagibajo od tradicionalnih življenjskih ciklov razvoja k bolj inkrementalnim, kateri na podlagi uporabniških povratnih informacij izpopolnjujejo ali izboljšujejo verzije programa (Cooper in Scacchi 2015, 10).

4.2 PROCESI V PRAKSI

Po podatkih spletne ankete v Avstrijski industriji iger 23% podjetij razvija igre z ad-hoc pristopi, 77% pa jih uporablja Scrum ali XP. Verjetno je k rezultatu botrovala velikost samih podjetij, saj ima 85% podjetij vsaj 4 zaposlene medtem, ko ima le 15% podjetij 15 ali več zaposlenih. Ne glede na rezultate so vsa podjetja nakazala uporabo nekakšnih fleksibilnih, sekvenčnih ali agilnih pristopov (Musil in dr. 2010, 5).

Raziskava, ki so jo naredili O'Hagan in kolegi je pokazala, da se pri razvoju iger uporablja 47% agilnih in 53% hibridnih procesov. Izmed 404 študij so identificirali 23 procesnih modelov med katerimi so bili vidnejši: XP, Scrum, Kanban, rapidni, inkrementalni in komponentni razvojni modeli. Medtem, ko so prvi trije našteti strogo agili, vsi bazirajo na iteracijah. Vsi modeli raziskave so se razlikovali le po številu iteracij, katerih število je bilo večje pri agilnih in manjše pri hibridnih procesih (O’Hagan, Coleman, in O’Connor 2014, 187). Prav tako agilne pristope podpira Fullerton, saj meni, da je Scrum primeren za reševanje zapletenih problemov oblikovanja iger (Fullerton 2008, 369).

Ob uporabi agilnih razvojnih procesov delujejo bolje tudi mobilne igre. Razvijalci označujejo te metode za metode inkrementalnih in iterativnih komponent (Unger in Novak 2011, 178).

4.3 PROCESNI MODELI ZA IGRE

Widyani (2013) na podlagi ključnih aktivnosti različnih organizacij in raziskovalcev predlaga GDLC[[3]](#footnote-3). Pristop predlaga faze: iniciacijo, pred produkcijo, produkcijo, testiranje, beta in izdajo. Iniciacija služi za stvaritev koncepta igre in enostaven opis specifikacije. Pred produkcija zajema revidiranje oblikovanja in izdelava prototipa. Po tej fazi je končan GDD[[4]](#footnote-4) na podlagi katerega je izdelan prototip. V prvi iteraciji produkcije služi prototip za temelje in strukturo v nadaljnjih pa se ga izboljšuje. Produkcija se konča z izdelanim prototipom, ki predstavlja celoto(Ramadan in Widyani 2013, 98). Sledi testiranje, ki izda poročilo defektov in seznam izboljšav. Rezultat testiranja odloča ali se razvoj nadaljuje v beta fazo. Slednja predstavlja identifikacijo defektov in povratnih informacij uporabnikov. Iz beta lahko ponovno sledi faza produkcije z namenom izboljšanja igre ali faza izdaje v kolikor je rezultat beta testiranja zadovoljiv (Ramadan in Widyani 2013, 99).

Aslan in Balci (2015) predstavita metodologijo za kompleksni razvoj iger GAMED[[5]](#footnote-5). PRILOGA B prikazuje življenjski cikel DEG[[6]](#footnote-6), ki je osnova metodologije. DEG cikel je sestavljen iz štirih faz: oblikovanja igre, oblikovanja programske opreme, implementacije (izdaja) in učenja na podlagi iger (povratne informacije) (Aslan in Balci 2015, 309). Metodo označujeta za iterativno, saj pričakuje povratne tranzicije. V kolikor se pojavi potreba se postavimo en korak nazaj in ponovimo delo. Tipično se pomikamo naprej in nazaj med procesi dokler ne dosežemo zadovoljive kvalitete delavnih produktov (Aslan in Balci 2015, 310). Kot podporni proces metodologija uporablja spiralni model, ki ga aplicira v fazo oblikovanja igre (PRILOGA C). Oblikovanje poteka po spiralnem vzorcu pri čemer vsaka iteracija pomeni izpopolnjen dokument specifikacije ideje. Iteracija vsebuje aktivnosti: prototipiranja, testiranja igranja, evalvacija in analize tveganj. Vzorec se nadaljuje dokler kvaliteta oblikovanja ni potrjena v aktivnosti evalvacije. Končni dokument te aktivnosti predstavlja specifikacija oblikovanja igre (Aslan in Balci 2015, 313). Specifikacijo oblikovanja se v aktivnosti zbiranja zahtev nadgradi v dokument specifikacije potreb. Specifikacija potreb se po potrebi zahtevne arhitekture nadgradi v dokument specifikacije arhitekture. V zadnji aktivnosti faze oblikovanja se predhodni dokument (specifikacija arhitekture ali specifikacija potreb) posodobi v dokument specifikacije oblikovanja programske opreme iger (Aslan in Balci 2015, 315–316). Na podlagi tega dokumenta se v fazi implementacije izvede aktivnost programiranja v kateri se zgradi DEG ali njene komponente. Metoda GAMED predvideva vzdrževanje, ki se izvaja na podlagi povratnih informacijah. Po potrebi posodobitve se ponovno izvede celoten življenjski cikel (Aslan in Balci 2015, 317).

Barbosa (2017) predstavi Game-Scrum, ki je hibrid Scrum in XP procesnih modelov (PRILOGA D). Scrum skrbi za upravljanje projekta medtem, ko XP zagotavlja inženiring. Procesni hibrid je primeren za razvojne ekipe z malo ali nič izkušnjami (Barbosa 2017, 293). Proces se deli na faze: pred produkcija, faza izdelave GDD, produkcija in post produkcija. V pred produkciji poteka iskanje dejavnika zabave, izdelava idealnega koncepta in oblikovanja. Delo poteka po principu metode poskušanja in popravljanja napak. V tej fazi se predvideva izgradnja enostavnega prototipa, ki je zaradi narave svoje hitre konstrukcije navadno zavržen. Sledi izgradnja GDD, ki je v produkcijski fazi preveden v dnevnik zaostankov. Produkcija poteka iterativno, priporoča Kanban metodo za kreacijo umetnin in podajanje časovnih okvirjev, mejnikov njihove izdelave (Barbosa 2017, 293). V kolikor je v ekipi več programerjev se priporoča uporaba Scrum in XP tehnik. Po končani igri se uporabi testiranje igranja za zagotavljanje kvalitete in dejavnika zabave. Zadnje dejanje v razvoju predstavlja izdelava dokumenta obdukcije razvoja, ki služi za identifikacijo pomanjkljivosti v preteklem razvojem procesu (Barbosa 2017, 294).

4.4 POMANJKLJIVOSTI PREDLAGANIH PROCESOV

1. (angl.) User Experiece-UX. [↑](#footnote-ref-1)
2. (angl.) Playtesting. [↑](#footnote-ref-2)
3. Game development lifecycle. [↑](#footnote-ref-3)
4. Game design document. [↑](#footnote-ref-4)
5. diGital educAtional gaMe dEvelopment. [↑](#footnote-ref-5)
6. Digital education game. [↑](#footnote-ref-6)