

**Cowboys, Ankle Sprains, and Keepers of
Quality:
How Is Video Game Development
Different
from Software Development?**

Emerson Murphy-Hill
Thomas Zimmermann and Nachiappan
Nagappan

Guilherme H. Assis

Abstract

- Jogos para video-game correspondem a uma parte importante na indústria de software.
- Engenharia de Software estuda pouco esse tema.
- Desenvolvimento para jogos difere do de softwares em geral.
- Verificar as diferenças entre as duas áreas.
- 14 entrevistas e 364 respostas para um questionário.

Introdução

- Jogos estão se tornando uma parte importante da indústria de software.
- Apesar da importância, é pouco estudado em estudos de engenharia de software.
- Estudos passados de engenharia de software afetam pouco os jogos.
- Desenvolvimento de jogos é diferente da tradicional engenharia de software?

Trabalhos Relacionados

- Existem diversos livros sobre o assunto em alto nível.
- Não contextualiza o desenvolvimento de jogos como um tipo especial de engenharia de software.
- Recentemente surgiram trabalhos parecidos que tentaram mostrar as semelhanças e diferenças.

Metodologia

- 2 partes: entrevistas qualitativas e pesquisas quantitativas.
- Foram entrevistados pessoas com experiência nas duas abordagens (LinkedIn).
- Nos últimos 10 anos, pelo menos 2 anos com software tradicional e 2 anos com games.
- 38 pessoas selecionadas, 14 entrevistadas (saturação). 9 atualmente em games e 5 não.
- Pesquisa de 10 min: 300 que trabalham com jogos, 300 do Microsoft Office e 300 da Microsoft que não trabalhe com Office. 84 responderam.

Resultados

- Os resultados foram organizados de acordo com o tópico da entrevista.

1) Software Requirements

- Essencialmente, jogos tem uma exigência: diversão.
- *“You are designing an experience, an emotional experience... It’s something supposed to be fun which is very subjective”.*
- Os requisitos de uma feature são passados em alto nível.
- Os requisitos são mais subjetivos em jogos.

2) Software Design

- Pouca atividade de design e planejamento. Os desenhos mudam com muita frequência.
- Com exceção de jogos de série, não há muito reuso de código.
- Criatividade é mais valorizada do que conhecimento técnico.
- *“...doesn't want you [the developer] to go and spend a whole bunch of time planning out how you're going to do this thing that he's asked for because he might change his mind in a week or two. Knowing that, he knows deep down that designing is useless because he's going to be constantly changing things...”*

3) Software Construction, Tools, and Methods

- Não há muito reuso, há performance deve estar sempre melhorando.
- Reuso implica em similaridade de software, e jogos devem ser inovadores.
- Um framework bem específico pode ser reaproveitado.
- Em jogos em série, algumas features podem ser reaproveitadas.
- *“The thing that makes video games unique is gamers want unique experiences...”*

4) Software Testing and Quality

- A qualidade é importante nas duas abordagens.
- Testes automatizados não são utilizados.
- Testes automatizados são frágeis com mudanças frequentes.
- Testes com humano são mais baratos que com softwares.
- *“It’s almost like watching a movie... so that experience for you to immerse in the game experience, it has to not create anything that would take you out of that immersion.”*

5) Software Maintenance

- Geralmente softwares de jogos não são mantidos, ao contrário dos demais.
- Desenvolvedores têm a impressão de que são os últimos a tocarem no código.
- A popularidade de jogos baseados na nuvem tem aproximado este ponto dos demais softwares.

6) Software Configuration Management

- É mais importante que no desenvolvimento de software tradicional.
- O jogo deve rodar em vários tipos de máquinas.

7) Software Engineering Management

- Gerentes na área de desenvolvimento de jogos geralmente não possuem conhecimento técnico.
- Em empresa de jogos, quando um conceito de engenharia de software é introduzido, é pelos engenheiros, e não pela gerência.
- *“Engineering management, in my time with [two large game companies], the person that I always reported to... didn’t contribute code.... half of them had no engineering experience at all.”*

8) Software Engineering Process

- Desenvolvimento de software ágil.
- *“When I worked in games, I got exposed to Agile, Sprint, Scrum... In the more traditional [large company] way, ... they're the opposite.... One year long, they know exactly what's going to happen every week. It's very different.”*

Survey Results

- Os resultados confirmaram vários pontos levantados nas entrevistas.
 - Requisitos são mais obscuros no desenvolvimento de jogos do que nos demais.
 - Desenvolvimento ágil é mais utilizado em empresas de jogos do que nas demais.
 - Criatividade é muito valorizada.
 - A habilidade em comunicar com gerentes é mais importante nas empresas de jogos.
 - Times de desenvolvimento de jogos precisam ser mais diversos.
 - Pessoas ficam mais “impressionadas” com desenvolvedores de jogos do que de sistemas.

Limitações

- As entrevistas e pesquisas foram pouco numerosas.
- Os questionários foram respondidos apenas por funcionários da Microsoft.
- As perguntas foram focadas nas diferenças e não nas similaridades, pode ter influenciado as respostas.

Conclusões

- Video-games ocupam uma parte importante no desenvolvimento de software.
- A maioria das pesquisas em Engenharia de Software não contemplam os jogos.
- Desenvolvimento de jogos e o software tradicional possuem muitas diferenças.