# Abstract

Winning or losing a game session is the final consequence of a series of decisions and actions made during the game. The analysis and understanding of events, mistakes, and flows of a concrete game play may be useful for different reasons: understanding problems of gameplay, data mining of specific situations, and even understanding educational aspects in serious games. Mistakes made by players may result in failure to complete the game objectives. These mistakes, which are usually difficult to spot or to reproduce in subsequent trials, directly jeopardize the learning capabilities of the serious games. In order to solve this issue we introduce a novel approach based on provenance concepts in order to model and represent a game flow. We model the game data and map it to provenance in order to generate a provenance graph, used for analysis. As a proof of concept, we also instantiated our proposed framework and graph generation in a Software Engineering game, allowing players to identify their mistakes and learn through them by analyzing the generated provenance graph from collected gameplay data.

# Resumo

Ganhar ou perder uma sessão de jogo é a consequência final de uma série de decisões e ações feitas durante o jogo. A análise e compreensão dos eventos, erros, e os fluxos de um jogo concreto pode ser útil por diversos motivos: compreender os problemas de jogabilidade, mineração de dados de situações específicas, e até mesmo entender os aspectos educacionais em jogos sérios. Erros cometidos pelos jogadores podem resultar em falha para completar os objetivos do jogo. Estes erros, que normalmente são difíceis de detectar ou reproduzir em ensaios subsequentes, prejudicar diretamente as capacidades de aprendizagem dos jogos sérios. Para resolver esta questão, apresentamos uma nova abordagem baseada em conceitos de proveniência, a fim de modelar e representar um fluxo de jogo. Modelam-se os dados do jogo e mapear a proveniência, a fim de gerar um gráfico proveniência, utilizada para análise. Como prova de conceito, também instanciado nosso quadro proposto e geração de gráfico em um jogo de Engenharia de Software, que permite aos jogadores a identificar seus erros e aprender com eles, analisando o gráfico proveniência gerado a partir de dados coletados de jogo.

# – Introduction

## Motivation

A game session is composed of a series of decisions and actions that were made throughout the game. In many situations, analyzing and understanding the events, mistakes, and flows of a concrete gameplay[[1]](#footnote-1) experience may be useful for understanding the achieved results from the session. A game flow analysis may also be fundamental for detecting symptoms of problems that occurred due to wrong decision-making or even bad gameplay design. Besides that, without any formalized process, this type of analysis may be subjective and, depending on the game dynamics and its complexity, it would require playing the game successively, making the same decisions, to intuitively guess which ones were responsible for generating the observed effects. Thus, reproducing the same state can be unviable, making it difficult to replay and identify, in a trial and error approach, the source of the problem. In addition, examining the game flow allows the identification of good and bad attitudes made by the player.

Game flow analysis techniques involve two phases: data logging and visualization. The data logging phase is responsible for collecting gameplay information during a gaming session. The visualization phase display the gathered gameplay information in the form of graphics and graphs. These graphs are used by developers and game designers to study certain aspects of the game, providing insights about player behavior or game issues.

Therefore, almost all AAA[[2]](#footnote-2) game titles have some form of game development telemetry (GDT) due to the importance of a game flow analysis (ZOELLER, 2010). According to ZOELLER (2010), GDT is an “*automatic measurement and transmission of data from game executable, build pipeline and development tools for recording, analysis, and workflow improvement*”. GDTs are mainly used by the game industry to analyze gameplay data to understand the customer’s experiences in the game, to identify issues post launch, and to understand the market for future games releases (ZOELLER, 2010).

Thus, researches that focus on game flow analysis are gaining strength along the years. However, the main usage of existing approaches for game flow analysis is to improve quality assurance (QA) and game testing[[3]](#footnote-3). Common approaches involves gathering information about players behavior (DIXIT; YOUNGBLOOD, 2008; HOOBLER *et al.*, 2004; LIU *et al.*, 2011; WALLNER, 2013), statistical data mining (AMBINDER, 2009; ROMERO, 2008; THOMPSON, CLIVE, 2007; ZOELLER, 2010), and aiding in the playtesting[[4]](#footnote-4) phase (DANKOFF, 2011; ZOELLER, 2010).

All these approaches are developer-oriented, which means that the game analysis is done by developers to improve their games. However, […]

This work main motivation is to facilitate player’s understanding about how events transpired during the game session and how each action influenced the outcomes. […] learn […] This knowledge can be used in future game sessions to avoid making the same mistakes or even to adjust gameplay features.

## Goals

Given the above, the aim of this work is to present a new approach to game flow analysis, from data logging to the data visualization. The goal is to improve the player’s understanding of the game flow, providing insights on how the story progressed and the influences in the outcome. In order to improve understanding, this work provides the means for analyzing the game flow by using provenance. The provenance analysis is done by processing collected gameplay data and generating a provenance graph, which relates the actions and events that occurred during the game session. This provenance graph allows the player to identify critical actions that influenced the game outcome and helps to understand how events were generated and which decisions influenced them. This process may also aid in the identification of mistakes, allowing the player to reflect upon them for future interactions.

Thus, this work proposes a conceptual framework that collects information during a game session and maps it to provenance terms, using digital provenance (FREIRE *et al.*, 2008) concepts for representing the game flow. The conceptual framework provides the means for a game flow analysis by using and manipulating a provenance graph to represent the gathered game information.

Even though the scenario used in this work is over a serious game, we believe that the concepts discussed here are applicable to any kind of game and are useful to support advanced game flow analysis, such as gameplay design, gameplay balancing, gameplay metrics, data mining, and even for storytelling.

## Research Questions

The construction of the approach and the evaluation of results through experiments have as main objective to answer the following research questions:

* Does provenance analysis help to understand events that emerged during the game?
* Is provenance analysis faster than only watching a replay of the game session?
* Is provenance analysis more accurate than only watching a replay of the game session?

## Contributions

This work introduces new perspectives on game flow analysis, to a level where the game provenance can produce and consolidate knowledge. This knowledge can help on (1) confirming the hypotheses formulated by players on how events affected the game, (2) supporting tutors for a better guidance when applying in a serious game, (3) motivating group dynamics around some case studies, and (4) extracting behavior patterns from individual sessions or groups of sessions. The provenance gathering also opens new research possibilities for behavior pattern data mining, provenance in storytelling, detecting gameplay design issues, gameplay metrics, and gameplay refinement and balance.

The provenance visualization can occur both on-the-fly or in post-mortem sessions. It allows the discovery of issues that contributed to specific game flows and results achieved throughout the gaming session. This analysis can be used on games to improve understanding of the game flow and identifying actions that influenced the outcome, aiding the player to understand why they happened the way they did. It can also be used to analyze a game story development, how it was generated, and which events affected it. Currently, we do not make inferences to the user, but let the user decide what he wants to infer. Studies in this area can be made in order to identify information that can be automatically omitted from the user without affecting the overall analysis.

## Organization

This work is organized in six other chapters, beside this introduction. Chapter 2 outlines part of the knowledge base necessary for understanding this work. It presents existing usages for gameplay data logging and analysis in the game industry. Then it describes known approaches for gameplay data logging and gameplay data visualization. However, these approaches are designed for aiding developers to identify player strategies and gameplay issues during testing phases.

Chapter 3 outlines the other part of the knowledge base necessary for this work. It describes the concepts of provenance in order to gather historical information about objects for further analysis. It also presents both existing provenance models (OPM and PROV) that can be used for provenance of digital information. Lastly, it presents a comparison between models, pointing out their similarities.

The proposed conceptual framework is presented at Chapter 4, denominated as *provenance in games*. Subsections describe how the gameplay data is gathered, in a game using the conceptual framework, and structured to be used in a provenance graph. Then it describes rules to interpret the gathered data for the provenance graph. Lastly, it describes features to distinguish information in the graph and existing visualization features to aid in the analysis.

Chapter 5 presents the materialization of the conceptual framework presented at Chapter 4, encompassing both the collection and visualization phases. The subsections provide an overview of serious game, describing the game and how the gameplay data is gathered, and implementation details for the provenance graph visualization tool developed in this work. It also presents a guiding game session example, which is the same game session used at Chapter 6 experiment. Lastly, it shows details about the provenance graph generation from the gathered information, the graph representations, and analysis features available at the developed tool by using the guiding example scenario.

After describing the approach and the graph visualization tool, Chapter 6 describes the evaluation performed on the usage of provenance analysis to understand game events. The planning and execution of the experiments are mentioned in subsections indicating how we obtained and processed the data for the assessment. The analysis is detailed through hypothesis testing. Finally, it presents the threats to validity of experiments.

Finally, Chapter 6 concludes this work, listing contributions, limitations and future work.

# – Conclusion

## Contributions

This work introduced a new approach for gameplay data logging and visualization, denominated *provenance in games*. This approach collects gameplay information from executed actions and events by logging related information, including the characters who executed the action and those affected by it. After the completion of the gaming session, the collected information, denominated as *game flow log*, is exported to an external provenance graph visualization tool, *Prov Viewer*. The *game flow log* contains the provenance information from the gaming session and is used by *Prov Viewer* to plot the game’s provenance graph.

The provenance graph allows post game analysis to discover issues that contributed to specific game flows and results achieved throughout the gaming session. This analysis can be used on games to improve understanding of the game flow and to identify actions that influenced the outcome, aiding the player to understand why they happened the way they did. It can also be used to analyze a game story development, how it was generated, and which events affected it.

The provenance graph visualization tool *Prov Viewer* utilizes graphic features to distinguish information for faster comprehension of the events. These features affect the displayed graph by transforming vertices and edges, changing their shapes and color according to the information type. Another important feature present is the information filter, which omit displayed information that is not relevant for the analysis. This filtering is important for analysis because it reduces the amount of displayed information to only those desired to be seen by the user. This allows for faster identification of the influences in the game, which is possible due how the provenance graph is structured.

The usage of provenance graph for analysis provides faster and more precise answers when determining the reasons of events outcomes in a game, in comparison with watching a replay of the game session. Both precision and agility were evaluated by an experiment where players watched a game session and answered a questionnaire with specific questions about certain events that occurred in the game.

While the main application of provenance in this work is over a serious game and is used to assist players in understanding how events affected the story, we believe that the concepts discussed are applicable to other kinds of games and useful to support advanced forms of analysis. The concepts presented may be useful for gameplay balancing and design, behaviors data mining, storytelling, gameplay metrics, and to detect gameplay issues.

Thus the contributions are:

[…]

## Limitations

One limitation in the *Prov Viewer* is related to scalability, which involves performance. Algorithms in *Prov Viewer* are not optimized for manipulating a graph with thousands vertices and edges. Thus, its performance may be compromised when dealing with such graph sizes.

Another current limitation is based on inferences, which *Prov Viewer* does not automatically make. All inferences must be done by the user. This might cause visualizations problems at a first glance due to the size of the graphs at their full extension. However *Prov Viewer* supports inference rules and provides the necessary features for implementing them.

Currently, *Prov Viewer’s* is not compatible with other provenance applications due to the input file format. The current input format uses an unspecified and undefined structure from the SDM game. Nevertheless, it can be adapted for known formats, such as XML, by modifying how *Prov Viewer* reads a file.

## Future Work

After developing the conceptual framework for data logging and analysis, along with the provenance graph viewer tool *Prov Viewer*, it is possible to describe new research possibilities for the proposed approach. The following paragraphs describe possible researches and improvements in the *provenance in games* conceptual framework and the *Prov Viewer* provenance graph visualization tool.

In relation to accessibility, we propose to make *Prov Viewer* less context sensitive, allowing the user to customize filters (edge filters and attribute status visualization) without the need of hard coding it in the application. For example, allowing the user to provide a configuration file that specifies the type of each filter. Thus, this would make *Prov Viewer* compatible with other games or provenance applications without the need of tinkering with the source code.

Another change would be related to *Prov Viewer’s* input file format. Currently, the *game flow log* is a simple tab separated value text file. However, there are plans to modify the structure to use some semi-structured format such as JSON or XML for greater compatibility with other applications. Thus, it would allow *Prov Viewer* to be more accessible by other applications due to the usage of semi-structured format.

One hypothesis for improving accessibility for *Prov Viewer* would be in creating an extension for existing game engines (i.e. Unity3D). This extension, when enabled, would automatically capture gameplay information when executing events, thus generating a *game flow log* without the need of changing the game’s source code.

One of the current draw-backs of *Prov Viewer* is related to inferences. Currently, we do not make inferences to the user, but let the user or developers decide what needs to be inferred. However, we provide the necessary tools to create inference rules, like filters and collapses (both for vertices and edges). Studies in this area can be made in order to identify information that can be omitted from the user without affecting the overall analysis, while at the same time not being context dependable. Thus it would provide generic inferences rules that can be used in any game.

Another area to be worked upon is related to the graph scalability. Depending on the game style, a game session might take several hours to complete, or even days in case of RPGs. This makes the size of the provenance graph to be overwhelming to the user, even when removing unnecessary information during the generation of the *game flow log* to reduce the log size. One way to avoid such situations is to show the provenance graph with some filters selected instead of its full extension. For example, before showing the graph to the user, it is possible to use collapses to reduce the graph’s size. Combats can be identified and collapsed into a single vertex for each instance. Places visited in the game can also be collapsed into a single vertex, containing all interactions made in that location, even combats. It is also possible to have collapses inside collapses. In this case, a collapsed combat inside a collapsed area visited by the player may contain other actions aside from the combat, such as interactions with the ambient. This gives an impression of a map from the player’s journey, showing vertices for each location visited by the player, while allowing the player to expand only the situations he desires to analyze. It is similar to *google maps*, where it shows the entire world and allows the user to zoom into specific locations. However in this case, it shows instances of the journey taken by the player.

It is also possible to go beyond that. Instead of collapsing all combats and locations, filters can be used to decide which combats or locations were not relevant to the story, or had no noticeable impact in the player’s journey, while keeping important events visible to the player. This is possible because provenance is analyzed from the present to the past. This way, combats outcomes are known and can be used to decide if it they are relevant or not. If the player was victorious with minor challenge, did not suffer severe wounds, or barely used any resources at his disposal, then the entire combat can be simplified into just one vertex representing the combat with the enemy. However, if the combat was challenging or the player lost, it is interesting to display all actions in it for analysis, allowing the player to identify important facts that influenced the combat outcome.

Although the proposed approach was used to assist the players to understand game events, it can also be used to open new researches in the field of behavior patterns data mining, gameplay design, detecting gameplay issues, and adjusting gameplay features. It might also be possible to modify the proposed approach to use in artificial intelligence (AI) algorithms. The AI would collect gameplay information, according to its perception limitations (if any), and analyze provenance information in order to improve its performance and adapt through past experiences.

# References

AMBINDER, Mike. Valve’s approach to playtesting: The application of empiricism. *Game Developer Conference (GDC)*, 2009. Available: <http://www.gdcvault.com/play/1566/Valve-s-Approach-to-Playtesting>.

BATES, Bob. *Game design the art & business of creating games*. 2. ed. Cambridge, Mass.: Thomson Course Technology, 2004. Available: <http://site.ebrary.com/id/10073606>.

DANKOFF, Jonathan. Game telemetry with playtest DNA on Assassin’s Creed. *Ubisoft Montreal*, 2011. Available: <http://engineroom.ubi.com/game-telemetry-with-playtest-dna-on-assassins-creed/>.

DIXIT, Priyesh; YOUNGBLOOD, Michael. Understanding playtest data through visual data mining in interactive 3D environments. *12th International Conference on Computer Games: AI, Animation, Mobile, Interactive Multimedia and Serious Games (CGAMES)*, p. 34–42, 2008.

FREIRE, J. *et al.* Provenance for Computational Tasks: A Survey. *Computing in Science Engineering*, v. 10, n. 3, p. 11 –21, jun. 2008.

HOOBLER, Nate *et al.* Visualizing Competitive Behaviors in Multi-User Virtual Environments. *Proceedings of the conference on Visualization (VIS)*, p. 163–170, 2004.

ISBISTER, Katherine. *Game usability advancing the player experience*. 1. ed. San Francisco, Calif.; Oxford: Morgan Kaufmann ; Elsevier Science, 2008. Available: <http://www.sciencedirect.com/science/book/9780123744470>.

LIU, Yun-En *et al.* Feature-based projections for effective playtrace analysis. *Proceedings of the 6th International Conference on Foundations of Digital Games (FDG)*, p. 69–76, 2011. Accessed: 24 jun. 2013.

ROMERO, Ramon. Tracking attitudes and behaviors to improve games. *Game Developer Conference (GDC)*, 2008.

SCHULTZ, Warren. *AAA Game*. Available: <http://gameindustry.about.com/od/glossary/g/Aaa-Game.htm>. Accessed: 3 jul. 2013.

THOMPSON, Clive. Halo 3: How Microsoft Labs Invented a New Science of Play. *Wired Magazine*, v. 15, n. 9, 2007. Available: <http://www.wired.com/gaming/virtualworlds/magazine/15-09/ff\_halo>.

THOMPSON, Jim *et al.* *Game Design: Principles, Practice, and Techniques - The Ultimate Guide for the Aspiring Game Designer*. 1. ed. United States: Wiley, 2007.

WALLNER, Günter. Play-Graph: A Methodology and Visualization Approach for the Analysis of Gameplay Data. *Proceedings of the 8th International Conference on the Foundations of Digital Games (FDG)*, p. 253–260, 2013.

ZOELLER, Georg. *Development telemetry in video games projects*. Available: <http://gdc.gulbsoft.org/talk>. Accessed: 28 jun. 2013.

# Appendix A

* 1. **INTRODUÇÃO**

Este apêndice descreve a construção, a distribuição e a análise de um survey sobre ferramentas de *diff* e merge. O principal objetivo é a coleta de opiniões sobre características de ferramentas destinadas à comparação de artefatos no âmbito acadêmico e industrial.

O mapeamento sistemático, descrito no Apêndice A, foca em artigos científicos sobre algoritmos para detecção de diferenças. Complementarmente, para uma análise do estado da prática, este *survey* concentra no que está sendo usado no dia a dia pelos desenvolvedores.

* 1. **CONSTRUÇÃO E APLICAÇÃO**

O formulário, apresentado no Apêndice C, é composto por questões referentes à experiência e ao uso de diversas ferramentas de comparação. Além disso, possibilita a indicação de características não enumeradas através de campos de texto de livre preenchimento. O *survey* foi construído e disponibilizado no Google Drive e a solicitação de colaboração foi realizada para um público alvo de 80 pessoas e esteve disponível no prazo de 15 dias. Dentre os participantes estavam alunos da Universidade Federal Fluminense estudantes de Ciência da Computação, graduandos e Pós-graduandos, além de pessoas do mercados de trabalho, de empresas públicas e privadas.

# Appendix B

**Formulário de Consentimento**

**Estudo**

Este estudo visa avaliar o quanto as técnicas de Proveniência em Jogos, apresentadas através de um cenário de jogo do SDM, são beneficentes para o aprendizado, tanto para um aluno experiente quanto para um aluno novo no assunto.

**Idade**

Eu declaro ter mais de 18 anos de idade e concordar em participar de um estudo conduzido por *Troy Costa Kohwalter* da Universidade Federal Fluminense.

**Procedimento**

Este estudo acontecerá em uma única sessão, que incluirá a apresentação de um vídeo do jogo SDM, documentos auxiliares para o entendimento do procedimento, um questionário, e em alguns casos a utilização da ferramenta *Proof Viewer*. Eu entendo que, uma vez que o experimento tenha terminado, os trabalhos que desenvolvi serão estudados visando entender a eficácia do modelo proposto.

**Confidencialidade**

Toda informação coletada neste estudo é confidencial, e meu nome não será divulgado. Da mesma forma, me comprometo a não comunicar os meus resultados enquanto não terminar o estudo, bem como manter sigilo das técnicas e documentos apresentados e que fazem parte do experimento.

**Benefícios e liberdade de desistência**

Eu entendo que os benefícios que receberei deste estudo são limitados ao aprendizado do material que é distribuído e apresentado. Eu entendo que sou livre para realizar perguntas a qualquer momento ou solicitar que qualquer informação relacionada à minha pessoa não seja incluída no estudo. Eu entendo que participo de livre e espontânea vontade com o único intuito de contribuir para o avanço e desenvolvimento de técnicas de ensino para a Engenharia de Software.

**Pesquisador responsável**

Troy Costa Kohwalter

Instituto de Computação – Universidade Federal Fluminense (UFF)

**Professores responsáveis (Orientadores)**

Prof Leonardo Paulino Murta

Instituto de Computação – Universidade Federal Fluminense (UFF)

Prof Esteban W. Gonzalez Clua

Instituto de Computação – Universidade Federal Fluminense (UFF)

**Nome:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Assinatura: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Data:\_\_\_/\_\_\_/\_\_\_**

**Este formulário está dividido Formulário de Consentimento, Questionário de Caracterização e um Questionário de Conteúdo.**

**Desde já, agradecemos a sua disponibilidade.**

# Appendix C

**Questionário de Caracterização do participante**

1. Formação Acadêmica

( ) Doutorado

( ) Doutorando

( ) Mestrado

( ) Mestrando

( ) Graduação

( ) Graduando

Ano de ingresso: \_\_\_\_\_\_\_\_ Ano de conclusão (ou previsão de conclusão): \_\_\_\_\_\_\_\_\_

1. Formação Geral
   1. Qual é sua experiência em Engenharia de Software? (marque aqueles itens que melhor se aplicam)

( ) Nunca aprendi Engenharia de Software.

( ) Já li material sobre Engenharia de Software.

( ) Estou fazendo uma disciplina sobre Engenharia de Software.

( ) Já fiz uma disciplina de Engenharia de Software.

( ) Dou aula de Engenharia de Software.

# Appendix D

**Experimento piloto: Questionário de Avaliação do Conteúdo**

1. Hora de inicio: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
2. Utilizou o *Proof Viewer*?

( ) Sim.

( ) Não.

1. Considerando que nos dias três e quatro, o funcionário Urias exerceu a mesma tarefa (Elicitação sem protótipo), por que seu desempenho foi quase um terço (1/3) no dia quatro em comparação ao dia três?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Descreva os motivos que levaram a funcionária Emmy pedir demissão no dia 15.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Identifique a semana com o maior índice de produtividade. Aponte os fatores que levaram essa conclusão.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Identifique a semana com menor índice de produtividade. Aponte os fatores que levaram essa conclusão.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Identifique os fatores que levaram a demissão de diversos funcionários durante a quinta e sexta semanas (dias 26 a 34).

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Identifique os fatores mais contribuintes que levaram a falta de Creditos apresentada na quarta semana (dias 20 a 26).

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Hora de término: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Appendix E

**Experimento: Questionário de Avaliação do Conteúdo**

1. Hora de inicio: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
2. Utilizou o *Proof Viewer*?

( ) Sim.

( ) Não.

1. Qual foi o motivo responsável pela redução do moral do funcionário *Arden* que consequentemente levou a seu pedido de demissão?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Qual foi o motivo responsável pela redução do moral do funcionário *Daniel* que consequentemente levou a seu pedido de demissão?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Por que nos dias 9, 10, e 11 o funcionário *Tornik* não obteve progresso na sua função de *elicitação* (não teve aumento nos requisitos de cliente)?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Por que nos dias 10 e 11 o rendimento de Daniel na sua função de *especificação* teve uma discrepância muito grande (342 *validation* vs 34 *validation*)?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Mesmo entregando o projeto com *bugs* não encontrados/corrigidos, qual foi o maior fator contribuinte que permitiu entregar o projeto a tempo?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Identifique os dois fatores mais contribuintes que levaram a falta de Creditos apresentada a partir do dia 11.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Um funcionário ficou sem nenhuma tarefa durante quatro dias. Quem foi esse funcionário?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Hora de término: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Se tiver alguma sugestão, escreva no verso. Obrigado.

**Questionário de Avaliação do Conteúdo**

**Gabarito**

1. Hora de inicio: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
2. Utilizou o *Proof Viewer*?

( ) Sim.

( ) Não.

1. Qual foi o motivo responsável pela redução do moral do funcionário *Arden* que consequentemente levou a seu pedido de demissão?

Falta de pagamentos

1. Qual foi o motivo responsável pela redução do moral do funcionário *Daniel* que consequentemente levou a seu pedido de demissão?

Exaustão

1. Por que nos dias 9, 10, e 11 o funcionário *Tornik* não obteve progresso na sua função de *elicitação* (não teve aumento nos requisitos de cliente)?

“Fez elicitação através de revisão de requisitos” ou “Falta de protótipos”

1. Por que nos dias 10 e 11 o rendimento de Daniel na sua função de *especificação* teve uma discrepância muito grande (342 *validation* vs 34 *validation*)?

Influencia negativa

1. Mesmo entregando o projeto com *bugs* não encontrados/corrigidos, qual foi o maior fator contribuinte que permitiu entregar o projeto a tempo?

Negociação por mais tempo (estender deadline)

1. Identifique os dois fatores mais contribuintes que levaram a falta de Creditos apresentada a partir do dia 11.

Contratação e treinamento

1. Um funcionário ficou sem nenhuma tarefa durante quatro dias. Quem foi esse funcionário?

Arden

1. Hora de término: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Se tiver alguma sugestão, escreva no verso. Obrigado.

1. Gameplay is defined as “the total experience provided by a game’s structure and mechanics” (THOMPSON, JIM *et al.*, 2007). [↑](#footnote-ref-1)
2. AAA, or triple-A, game is a game developed by a large studio and funded by massive budget (SCHULTZ, 2006). [↑](#footnote-ref-2)
3. Game testing is a software testing process for detecting defects, also known as bugs, in the game and is directly related to quality control in games (BATES, 2004). [↑](#footnote-ref-3)
4. Playtesting is the process of exposing the game, before its release, to the intended audience in order to find bugs and design flaws (ISBISTER, 2008). Playtesting differs from QA testing because it does not use professional testers, but actual players. It can be run as an open (commonly known as Beta test in games) or closed (commonly known as Alpha test in games) playtesting. [↑](#footnote-ref-4)