Game Live Logs

Creating a Conversation Platform to Reduce Game Development Conflicts

Tiago Lemos de A. Machado
Centro de Informática
UFPE
Recife, Brazil
tlam@cin.ufpe.br

Geber Lisboa Ramalho
Centro de Informática
UFPE
Recife, Brazil
glr@cin.ufpe.br

Carina Frota Alves
Centro de Informática
UFPE
Recife, Brazil
cfa@cin.ufpe.br

***Abstract*— One of the main problems observed in the game development process is related to the artifacts supporting the production [5]. There is no standardization about what is needed, what their content is and how they should be presented. Traditionally, according to the literature, the Game Design Document (GDD) should be the main artifact, the guide for the production team to create the game, almost like a movie script or an architectural design blueprint [28]. However, analyzing game industry practices, the importance of such an artifact is a contradictory issue, mainly because it does not accomplish its functions and it generates conflicts. This paper presents the results of interviews and a survey with game insdustry that led us to a detailed diagnosis of the problems related to the GDD. Then, the paper introduces Game Live Logs, a novel tool that extends the conventional GDD format, content and role in order to minimize conflicts, maximize productivity and, mainly, foster more dialog among the development team. The tool has been incrementally tested and improved, yielding to encouraging results.**

I. INTRODUCTION

Although its presence in development cycles is considered as one of the most important elements for game production, we have observed that the importance of the GDD has proved increasingly contradictory. In some works, it is said that initially the GDD was not necessary (at least in formal terms) because all the game view was in the mind of some individual that could accomplish all the development work [24]. However, due to the growth of the game industry, the need for documentation in game projects arose out of the rising complexity of such projects which are currently comparable to that of major film productions [11]. Considered in the literature as the primary artifact in the game production process, we have observed in the course of our research that in practice this document is in the heart of a great number of conflicts between the pre-production and production stages. Based on our literature study we develop a survey to collect answers for questions about the GDD importance. The survey was answered by more than 30 industry members and resulted in guidelines to help other professionals to avoid the pitfalls between the development stages.

In order to confirm our findings we continue our research. We conducted interviews with 12 game industry professionals that answered about their problems and necessities in their process. The gathered opinions was used as principles to build a prototype which aimed help to solve the related conflicts.

Thus, this paper presents a two-fold work. First, it discuss the results of the application of questionnaires (survey) to games industry members in order to identify preproduction vs. production conflicts. Then, the paper goes deeper in this discussion by presenting the results of an interview conducted with industry experts. Using their opinios to built and introduces the Game Live Logs (GLL), a possible solution for the identified documentation problem.

GLL is a novel tool that extends the conventional GDD format, content and role. The tool has been incrementally tested and improved. The results obtained so far encouraging shows that GLL mitigates some conflicts, maximizes productivity and, mainly, fosters more dialog among the development team.

II. DOCUMENTATION LIMITATIONS IN GAME DEVELOPMENT

To serve as a means to conceptualize, to describe and communicate the record of ideas about what should be the game to be produced is the definition of Rouse III [24] for the GDD. Observing the work of other professionals, such as [4; 28; 27; 17], we see that such a definition has varied little and the presence of such a document within the production processes, at least in the view of literature, is unquestionable.

However, although the role of *conceptualizing, describing, and communicating ideas about the record of the game to be produced* is almost a consensus among the authors, the form and content of the GDD varies greatly, being far from a standardization. What makes the task of extracting requirements in a challenge for each new document that is found. Both in [1] and in [5] this is a fact that plays a crucial role for the later stages of the project because the difficulty of correctly interpreting the document has led to decisions that affect the team productivity.

What is evident in a work like [10], based on the RUP, his process of game creation fits an entirely methodology for software development, in which all the phases: Inception, Elaboration, Construction and Transition are reviewed, expanded and updated from the perspective of the game production. Including a set of tools and techniques to facilitate the work of developers. However, such a proposal can only be used in essence if the Inception phase (software) receive accurate and correct information from artifacts like the GDD and similars, which, as indicating by other researchers [35], is far from happening.

Works such as [31] and [12] have long revealed how expensive is negligence in obtaining precise requirements for later phases of software development. Moreover, if this does not occur as the first step in a proposal as previously mentioned the risk of rework would be immense.

An alternative to this problem would be to create something like the one proposed by [6], the work deals with the creation of a game predictive process for personal computers. A major inspiration for the model was also the RUP. The author creates an orientation for Game Design Decisions, which means that all tasks only can be accomplish if it has a prior level of detail present in the GDD. The criticism is exactly the level of detail, which is a definition of the responsible for the Game Design and the answer if it is sufficient or not, only occur during the production stage.

Although the latest works have used RUP as inspiration, we realized in the researches of [18;19;20] and [30] that even in agile methods the lack and incoherence documentation is a problem factor pointed by professionals. The first one reveals a list of critical factors and good practices related to an analyze study conducted in game postmortem documents and the other one, based on the same data, lists the current industry challenges. In both studies, the transition between pre-production and production guided by artifacts was mentioned like one of the most frequently claim in game development process.

III. GDD Issues According To The Industry: A Survey

As the literature is not clear enough about documentation problems in game development, and it may have academic bias, we have decided to run a survey with 38 game professionals working in Brazil and abroad in order to go deeper into this problem according to the industry point of view. We saw that many of the problems are repeated both in smaller companies, working with teams of a few developers and shorter timelines, as in large companies with many employees, long development cycles and high investments (AAA).

In these problems, there is a set of them related to the conflict between the stages of pre-production and production, which many participants mentioned that is caused by poorly artifacts, which should clarify the project to the team, but has proven to be inadequate, especially the GDD. Because of this, many survey respondents recommended that is more practical to have the fully attendance of a Game Designer in all phases of the project, since the interpretation errors are frequent and many developers do not usually read the documents [Fig. 1]. This fact was also pointing out in the interview presented by the Designer Tim Lang [13] to the Game Carrer Guide, in which he ironically answered that his job in the game industry is *"write documents that nobody reads"*.

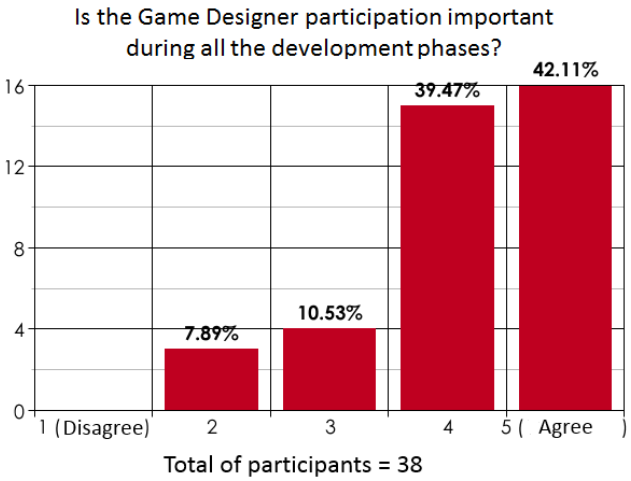

Fig. 1. Participants' answers about the game designer participation during the development phases.

The problem of not using any documentation and requiring full presence of the game designer is that the he or she will not always have the availability to promptely help the team all the time and for all the details. The period between the doubt request and the designer prompt attendance may be not short. This is potentially a source of delay, misunderstanding, fall in the productivity that can cover several tasks.

The survey participants also suggested that the creation of the GDD auxiliary artifacts could attract more users (readers) and mitigate many problems related to the design and production. According to them, the use of state machines and prototypes are examples of rich and attractive resources that has a greater power to explain situations that GDD, only, cannot make it clear [Fig. 2].

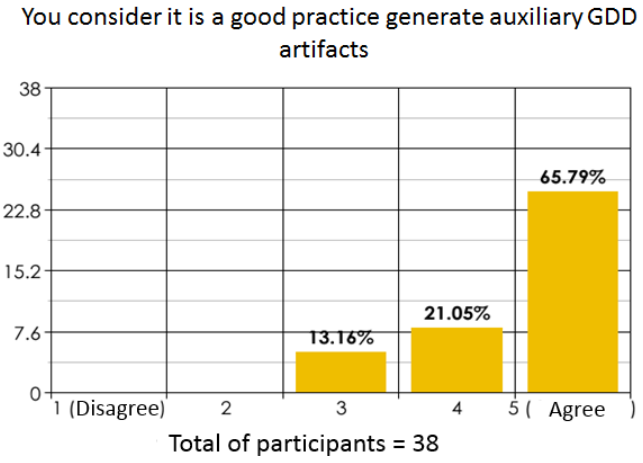

Fig. 2. Participants agree to be a good practice develop GDD auxiliary artifacts.

One concern about these complementary artifacts, as well as the GDD itself, is that the designer needs to define an appropriate structure of presentation for each artifact, which implies in choice documentation tools that can be appropriate to the team members' multidisciplinary activities. For instance, AI programmers, network programmers, graphic artists, sound designers, market people, level designers, and so on, each of these developers requires different information, possibly in different formats. In short, the GDD should be aplied to its different audiences. Keeping a broad and general scope in a long and detailed GDD may explain why people do not want to read it.

Finally, the participants also reaveled a need of the GDD be more able to evolve during production instead of having a definite or complete version of it at the beggining of game development. The majority of the respondents agreed to be able to start the production of any game from a preliminary document, which can iterate during the production [Fig. 3].

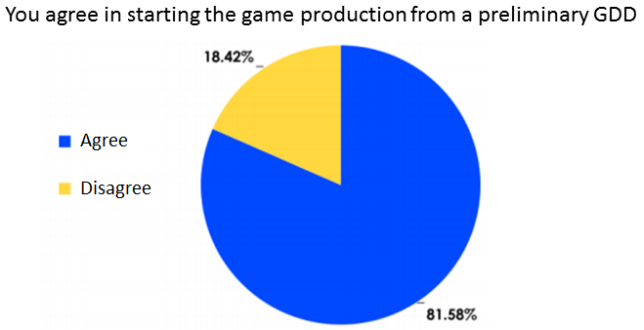

Fig. 3. Participants agree in start the game production from a preimanary GDD which can be iterate during the production.

From the revealed in this section and based upon the results of our research and the literature study, we detail the inadequacies and conflicts generated by the GDD, below we list what we understood as a diagnosis of the problem.

1) GDD Target Audience

The problem of misunderstanding of the roles and functions reflects a classic case of multidisciplinary conflict [14] we observe in GDDs. While in many cases the same document is passed on to all team members, perhaps the best solution would be thinking its creation and subsequent distribution to different audiences.

The multidisciplinary nature of game projects makes that the information are many, but not all team members need to consume all of them to conduct the project development, in other words, an artist may not need to get all the details about a software that the programmers must create and vice versa.

2) GDD Format and Content

Another inadequacy factor concerns the GDD content and format, the traditional text document which intends to explain everything in the minimum details tends to be ignored by many members. But the summary document brings lapses that creates problems of interpretation, blockages and delays in production. There are alternative formats [32], widely used, such as presentations, wikis [23] and storyboards [26], but all brings his advantages and disadvantages.

3) GDD Evolution

One of the most common conflicts between pre-production and production is the GDD evolution. As previously seen in one of the testimonies presented in the work of [19], one of the interviewees clearly revealed that the team delayed the production of the game due lapses in the document that was ignored and did not followed the project updates. In our research, we noticed similar disconnection between the game described in the document and the game in development.

4) The GDD and The Project Plan

Another conflict between pre-production and production concerns the inadequacy of the connection between the Project Plan and the GDD. As the second contains the specifications of the game, it would be the ideal material for the construction of such a plan, but the number of interpretation problems already mentioned makes that the GDD not a good candidate for such a task and implies in create a mismatch between the production intents and the design perspectives.

As a result, we present a list of guidelines to help developers avoid the problems reported on the experiences of the professionals who attended the survey. In order to confirm our findings we decided to conduct further study through interviews to understand, from the expert opinions, what are the real needs of the industry about the research topic and how we can contribute to present a more accurate solution.

IV. EXPANDING THE DISCUSSION: REVISITING THE DIAGNOSIS

As a result of our findings in the previous research and what we found in the literature, we decided to start a new study. In which we broaden the scope of the discussion and understand, according to the views of the game industry members, the conflicts between pre-production and production, which have historically been a direct consequence of the GDD [5]. The objective was to evaluate our initial considerations in order to confirm it and find out if there was some unidentified issues that could be inserted to have a more complete and diagnosis. For this, we made contact with the game companies and their professionals in Brazil and worldwide. Our goal was to establish a more accurate industry diagnosis that allow us better understand their main problems and needs, to propose a solution based on their views.

For the participants' selection, our interest was in the following specific groups [22]: Administrative Group - comprises people who deal with the administration and

development resources allocation. In addition, the roles are responsible for the majority decision in a game project.

• Producer / Project Manager;

• Technical Software and Art Leaders / Game Designer;

Operational Group - are in charge of, from the decisions of the Administrative Group, transform design ideas into a product, in this case, a digital game.

• Developers

• Artists

• Designers

According to the desired profiles, we selected twelve (12) participants. The average age was around twenty nine (29) and time of experience in the game industry reached an average of seven (7) years, many respondents also gained experience in other activities in the software industry, most of them as programmers, interface designer or project managers. Our intention was that our employees answer about your needs and learning about problems between pre-production and production resulting from artifacts that surround them.

The respondents followed a protocol of questions, which was conducted in a semi-structured manner to attend the conversation possibilities allowed by the technique as described by [16;25]. From the transcripts of the interviews, we analyzed the produced material according to the Coding Theory, which consists of generate conclusions in a clear and systematically way based on the collected data to be coded into categories that help the researcher to develop his theoretical bases [25]. To assist this task, we use the evaluation version of the software Atlas.ti [2], which provides a number of features for navigating in the transcribed material, the creation and editing of codes, and various forms of visualization and queries that allow quick word locations and quotations in the text, as well as creating relationships between them.

A. Interview Results: New Diagnosis

We decided in this work categorize the interviewees according to the points discussed in our first diagnosis. Below the more representative quotes raised in the interviews are putting accompanied by our analysis on these data considering the survey results (section II) and our literature review.

1) Target Audience: For the target audience, respondents were clear in specifying three areas as the most representative in the game development field. Although references to the Marketing and Advertisement have appeared in some answers, Game Design, Art and Programming were always mentioned as the main disciplines interested in the artifacts that surrounding the game production, as evidenced by the comment below.

"The team is divided into three basic views: Game Design, Art and Programming. So we need to think in elements to represent these areas... Here, we divide the work according to this view... "

2) GDD Format and Content: The GDD Format and content continue to vary widely, some companies still use the traditional text-based format and still seeking other alternatives, as indicated by the comment below:

"I feel the need in terms of having a GDD that is more practical because it contains a lot of text, so it's easy to forget something. Perhaps, I believe it can be more interesting to have something like a checklist. "

Meanwhile, other companies have invested in other formats, which seems to be very concise and objective as shown below.

"...we use a single page of Game Design Document to all game modules (...) we defined what needs to simplify on it due to the team capacity and to reach the priorities like release date..."

3) GDD Evolution: One of the major problems cited was the difficulty in maintaining the artifact (s) updating. According to participants, the idea of a complete document that defines everything about the game is wrong and unreal as shown by the employee in the comment below:

"I think the GDD is not the only process that has the game essence... in fact the game essence should be at the team's head. I say this because I see a great concern with the game design document. Everyone tries to develop it tightly closed before starting the production and in my opinion the game design document is never closed. I always say this in classes, lectures, for teams that are starting... It is very common we develop 10 features over the original one that the customer proposed and because of it, to me, the record of a game design should be done in post-it notes or paper. Because these things are volatile and easy to manipulate ... You record your ideas and know where to go... game is essentialy iteration ... then you do, forehead, does it again, see if it looks good and go on."

Such cases were reported by experient participants and dealing with the intensity changes that occur in the game development process, what requires flexibility in its documentations, which unexists in the classic (texts) formats. For this, participants have continued to invest in alternative means to reach those needs in the comment below:

"We are using a model called Game Design Logs, it was created to support the 'live' format of a GDD and it has been very productive for our projects. All documentation is done by small records, with few details and shared with the whole team, everyone can view and make changes in real time or by records that are always saved. So it has been very easy to work with and attend many of our needs, especially to keep a GDD updating... Before, the design and the game (production) were always asynchronous. "

4) The GDD and The Project Plan: The intensity changes and the difficulty of keeping the GDD updating reveal how far the relationship between the artifact and project planning is. The number of iterations present in the current process was considered one of the factors which implies in changes the game specifications all the time as mentioned in the following comment:

"The main problem is the number of iterations we have to do during our process... Basically, you have to make changes all the time..."

What is most evident when such specifications have never been well defined at a first moment, as informed by the comments of another participant:

"All definitions of pre-production are important and help in the production. Even with them being iterated during the project, have a starting point and an ultimate goal is essential."

The above comments reinforce the need of having a flexible document which can be adapt to changes and that can explain in each iteration the objectives to be achieved, otherwise problems between the design and development can be accumulated during the project.

5) Game As Document: One participant, a member of a traditional company recognized for its AAA titles, revealed that the company is using prototypes as a GDD replace, according to him, it happened due the power of prototypes expression in explain how a game should be implemented:

"We stopped to create documents to explain how the game should be, we now have developed prototypes and show how the games should be! We reduced our documentation process and we made the things very simple to be specified and concluded."

However, despite the success cited by the previous participant, there are those who do not agree with the drastic decrease of the documentation (which is more one of many conflicts in which the GDD is inserted):

"Some sees it as an unnecessary work because they "know what they are doing." However, this documentation is important to improve the chance that the challenging parts of the game design can be implemented successfully."

We have seen that even with a broader discussion, which was focused to address the game professional's vision about the problems surrounding the game pre-production and production, the GDD, by the interviewees' speech, continues to ba a contradictory artifact leading conflituos opinions about its uses, effectivity, formats and etc. Such contrary is reflected mainly because the task of guiding the team during the dynamic game production is still not answered. Revisiting the diagnosis, we found in the interviews many data that reinforce it and allow us to incorporate a new element in the discussion: the possibility of the game prelimary versions (prototypes) can serve as its own document (subsection 6 of this chapter).

V. GAME LIVE LOGS

We believe that as a result of the raised discussion, the points of inadequacy and conflicts presented in the GDD, should be seen as principles to reduce problems of the game Pre-Production and Production, considering the use of an artifact. However, updated and corrected according to the professionals' view that collaborated in this study.

We use the principles as a basis for building a prototype that allowed us to verify our considerations in a real context. The first step was to map the speeches in requirements, for this task we adopt a similar approach created by Sari Kujala [12] and already applied in a many researches like [9].

Based on our findings we believe that the Game Live Logs must provide the features below, which can iterate and evolve to new ones according to the Lean methodology.

Log – It is the document edition atom. Every entry in the document is made by a log, which consists of a text box that accepts any content type. The purpose is that a log represent a rich, clear and concise game design information.

Colaborative – Every team member can view, insert, edit and comment a log whenever wants.

Content Flexibility – The Game Live Logs should accept any content type, even prototypes or games, and provide visualizations and iteractions to them within the system.

Tags – The users can include tags in their logs to categorize and direct it to specified disciplines (Programming, Arts, Design, etc.) and team members. Thus, the users can read only the document 'pieces' that they are interesting.

Queries – In order to facilitate the document navigation, the system must provide many query types (by tags, by user names, by date, by content, etc.).

The features presented brings a lot of functional requirements like insert and edit logs, as non-functional requirements like easy navigation. Attention to these requirements, especifically the non-functional are important. Only a good usability can provide to developers a system-document in which the communication can occur in a more clearly way than the traditional ones, that can be easy to keep and upgrade, that meets specific preferences of team members and can supports various content types and presentation formats.

To build our solution we decided to take as benchmark, an implementation of the Game Design Logs concept, proposed by Daniel Cook and presented to us by one of the respondents. From all the methods that we had contact by read the literature or due to our survey and interview's answers this one was the more apropriate to follow in order to be considered easy to keep and register game design issues. The original approach has no system and runs inside an online Google Drive's document. Our implementation extends the proposal creating a specific tool for it, adding the requirements listed based on our researches and correcting some of its criticisms, presented in the author publication [7]. We named our prototype as "Game Live Logs" by inspiration of a research collaborator who mentioned the need for a game documentation be an "alive" artifact.

For the prototype implementation, we follow a joint approach using the process Running Lean [15] associated with agile practices [34]. Thus, the mapping listed above contains the definition of our Minimum Viable Product (MVP). From the prototype evaluations we conduct our learning cycle, in which we update our proposal with several versions, following the recommendations of [8] and [33] who indicate the creation of many prototypes and the preparation for frequent, quick and agile trials as factors to achieve more efficient solutions.

VI. GAME LIVE LOGS EVALUATIONS

The first version of the prototype was built using a trial version of Axure [3], it had many similarities with the proposal on which we were based, the Game Design Logs. We did a "Quick and Dirty" evaluation, recommended to collect informally opinions about system elements under development [29]. Two volunteers, both research participants during the interviews phase, agreed to evaluate the prototype's first version [Fig. 4].

Fig. 4. A print of the Game Live Logs first version.

They mentioned that they liked the log ideas, principally beacause "it is easy to follow". However, they pointed a fail in separating text from content. In this version, the GDD text was maintened in a page and the attached game content associated were refered by a link. Users indicated that it was confused writing a GDD with this separation, because it brings usability problems in the system navigation. They also highlited that the tag and attachment fields were not eventually viewed and caused omission in some inserts.

According to the evalution, we modify the forms to insert and view the logs. By the reviews we see that these two actions (insert and view a log) must be in the same page in order to improve the system navigation and improve the log readability. We also highlighted the tag and attachment fields and let the log creation with just one text area, the previous version had two, one to write about an issue and another to propose related notes with suggestions, solutions and so on, but the evaluations considered it unacessary and mentioned that it can be done in a single text area.

The prototype changes of insert [Fig. 5] and view [Fig. 6] a log are present in following.

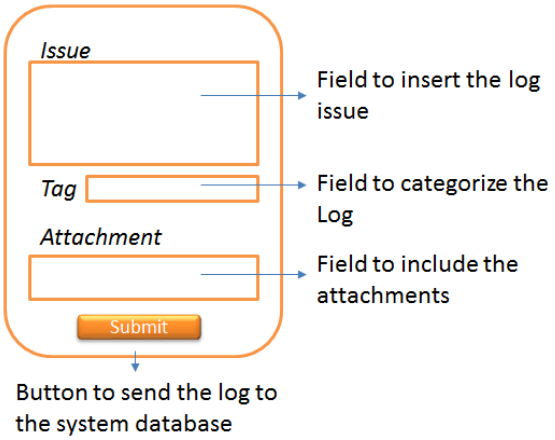

Fig. 5. The prototype user interface to create a new log.

Fig. 6. The prototype user interface to view a log.

With the first experiment results, we decided to implement the new version using an appropriate tool to the creation and maintenance of web systems and for this task we adopt the Google App Engine. For this version we had the cooperation of Manifesto, a Porto Digital do Recife's company, which has several titles released and about eight years of experience. More specifically, the test was conducted by one of the Manifesto's team development for a period of four weeks. The team was formed by a Game Designer, an Artist and a Programmer and it were selected because they were in the early stages of a new game development.

Below, the UI, based on our first version findings, to insert [Fig. 7] and view [Fig. 8] logs as built in order to start the evaluation of the Game Live Logs in the selected company.

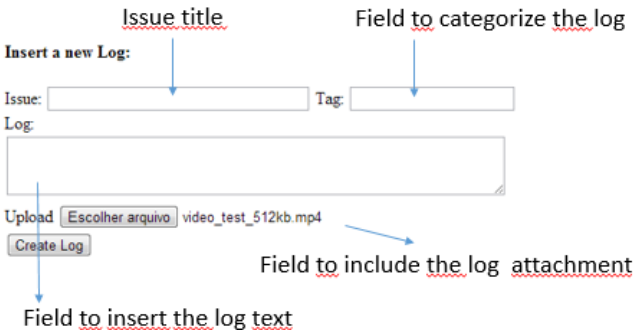

Fig. 7. The user interface to create a new log in the Manifesto's evaluation version.

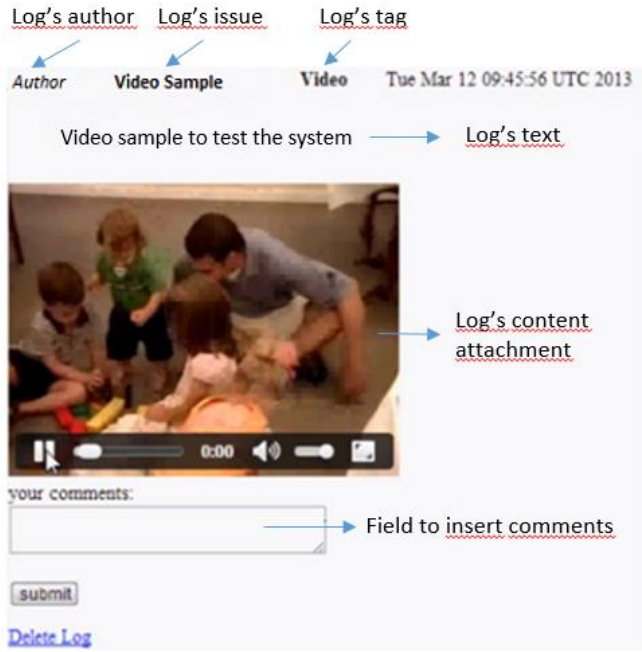

Fig. 8. The user interface to view a log in the Manifesto's evaluation version.

A. *Comparison with another tools*

Before starting the evaluation, we had a reunion with the test team members to present our research purposes and to know a bit of their daily professional routine. They mentioned the use of other tools used to document games. Initially they had an experience with Facebook groups, but it was not positive because as long as the project evolved more hard was to find older posts, the organization were not considered suitable to what they need, besides the social cycle interruptions.

After the experience with Facebook, users reported that they started to use phpBB [21], a free open source system designed to maintain web forums. The problems in this tool is that the forum organization was consider too heavy to raise game design issues via topics and suffers the same difficulties faced by Facebook's groups when the project evolves.

B. *Evaluation Results*

The system was tested for four weeks, and each week had a sprint to update the system according to the users' needs. To collect the recommendations a visit was made to the team every week and during the visits, the users always answered the following questions:

- What are the strengths and weaknesses found?
- What you need to improve your activities with the system?
- The system use has bothered your activities in the company?

The main strength revealed by the users was the easy of learning and use the system, they even mentioned that it was easier to manipulate than the tool normally used by the company (phpBB).

Regarding improvements, users asked for more support for content. Specifically they asked for a way to view swf files, because a great number of the team activities is built using Adobe Flash. They also felt the need for a text editor with support for markers [Fig. 9], hyperlinks and code highlight. Finally, they asked for a multiple tag system and a field to associate a log with a specific task reported in the project management tool used by the team, in case, the Atlasian Jira.

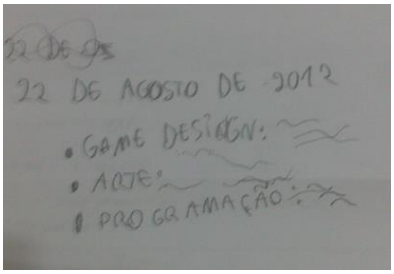

Fig. 9. User's sketch illustrating how he wants to edit his text with markers.

About inconvenience that our tool could bring to the users, they reported that they felt no problem in using it and that it did no harm to any of their daily tasks. Nevertheless, they were also emphatic that the experience could be even better with the suggested changes in the future.

C. *Evaluation Considerations*

During the test period, we observed that the evolution of the prototype followed many of the principles that we had previously defined. Throughout the evaluation cycles the users shaped many of the planned features. We received with enthusiasm the ask to include a field in the log to associate it

with a task definid in a project management tool [Fig. 10], because we do not know initially how to implement the the project plan comunication requirement. Due to our user test it was possible and in this case, with this tool, we could see a game design and a game in production step side by side.

Another enthusiastic moment was when the test users aswered *"You read our minds"* when we asked if the Flash content visualization that we implement was adequate to their needs.

Finally, we see that the GLL do not exclude any approach used to document games, but let the users to do it in a quick, easy and agile way. The concept is prepared to teams that like to write every game element in the minimum details as is prepared to work with teams who prefer visual and interactive approches like storyboards and prototypes.

Fig. 10. The log creation changes after the evaluations. Among other features, this new version presents a text editor with more funcions (Bold, Italic, Markers, cold syntax highlight, etc.) and a field (Task Number) to associate a log with a task in a project management tool.

VII. CONCLUSIONS

In this paper we investigate the conflicts between pre-production and production of games, more precisely those generated by GDD artifact, which is in the heart of the transition between these two mentioned phases.

As a first contribution, we present a series of studies and empirical data that show clearly, from the industrial viewpoint, the inadequacy of the Game Design Document (GDD) in the game development process. The GDD is meant to help but, for a lot of reasons, it is not fulfilling the role of guiding teams and problems caused.

We take this knowledge and set principles that were used to create a collaborative tool, the Game Live Logs (GLL), our second contribution. GLL has been evaluated through some prototype versions, each of one incorporating suggestions obtained by the informal tests. A game development professional team has performaed a controlled evaluation of GLL, providing valuable suggestions on how to evolve it and confirming the utility of GLL as a tool for fostering dialog and productivity.

For future work, we must improve the supported contents and visualization tools to increase User Experience. Furthermore we will create integrations between the GLL and project management tools to bring closer the design issues and the project perspectives. Finally, due the concept test of GLL has been positive, we need to make it available for a large public in order to get more feedback and evolve it to a professional tool, aimed to the game industry worldwide.

ACKNOWLEDGEMENTS

We want to thanks Manifesto*, the research collaborators and companies which became this work possible.
* http://manifestogames.com.br/

REFERENCES

[1] Alves,C. Ramalho,G. and Damasceno, A. "Challenges in Requirements Engineering for Mobile Games Development: The Meantime Case Study," 2012 20th IEEE International Requirements Engineering Conference (RE), pp. 275-280, 15th IEEE International Requirements Engineering Conference (RE 2007), 2007

[2] Atlas.ti. 2012. Available at: http://www.atlasti.com [acessed june 08, 2013]

[3] Axure. 2011. www.axure.com [accessed june 10, 2013]

[4] Bates, B. and Robert, A. 2004. Game Design, 2nd Edition. Course Technology Press, Boston, MA, United States.

[5] Callele,D., Neufeld, E. and Schneider, K. 2005. Requirements Engineering and the Creative Process in the Video Game Industry. In Proceedings of the 13th IEEE International Conference on Requirements Engineering (RE '05). IEEE Computer Society, Washington, DC, USA, 240-252. DOI=10.1109/RE.2005.58 http://dx.doi.org/10.1109/RE.2005.58

[6] Carvalho, G. Uma Metodologia Preditiva para o Desenvolvimento de Jogos de Computador. Trabalho de Graduação. Graduação em Ciência da Computação. Centro de Informática. Universidade Federal de Pernambuco. Recife. 2006.

[7] COOK, D., 2011. Game Design Logs. Lost Garden. Available at: http://www.lostgarden.com/2011/05/game-designlogs.html [Accessed July 15, 2012].

[8] Dow, S. 2011. How prototyping practices affect design results. interactions 18, 3 (May 2011), 54-59. DOI=10.1145/1962438.1962451 http://doi.acm.org/10.1145/1962438.1962451

[9] Falcão, T. P. R.; GOMES, A. S.. Modelagem de soluções ubíquas para uso em salas de aula do ensino fundamental. In: GCETE 2004 - Global Congress on Engineering and Technology Education, 2004, São Vicente. Anais do GCETE'2004, 2004.

[10] Flynt, J.P., 2004. Software Engineering for Game Developers, Muska and Lipman Publishing.

[11] Hirsch, E., Stringfellow, R., Amer, P., Goodrich, B., Marshall, J., and Cliff Brett, L. 2007. Crossing the line: Moving from film to games and possibly back. In ACM SIGGRAPH 2007 courses (SIGGRAPH '07). ACM, New York, NY, USA, 1-94. DOI=10.1145/1281500.1281673 http://doi.acm.org/10.1145/1281500.1281673

[12] Kujala, S., Kauppinen, M., and Rekola, S. (2001). Bridging the gap between user needs and user requirements. In Avouris, N. and Fakotakis, N.,editors, Advances in Human-Computer Interaction I (Proceedings of the Panhellenic Conference with International Participation in Human-Computer Interaction PC – HCI 2001), pages 45–50. Typorama Publications.

[13] Lang, T., 2009. Four Ways to Write Your Design Docs. Game career guide, p.3. Available at: http://www.gamecareerguide.com/features/737/four_ways_to_write_your_design_.php?page=1 [Accessed June 9, 2012]

[14] Marco A. Lara, Shimon Y. Nof Computer-supported conflict resolution for collaborative facility designers
International Journal of Production Research Vol. 41, Iss. 2, 2003

[15] Maruya, A. 2012. Running Lean: Iterate from Plan A to a Plan That Works. O' Reilly Media; Second Edition edition (March 6, 2012).

[16] Merriam, S.B., 2009. Qualitative research: a guide to
design and implementation, San Francisco: Jossey-Bass

[17] Perry, D. David Perry On Game Design. 2009. Cengage Learning; 1 edition (March 24, 2009)

[18] Petrillo, F., Pimenta, M., Trindade, F. and Dietrich, C. 2008. Houston, we have a problem...: a survey of actual problems in computer games development. In Proceedings of the 2008 ACM symposium on Applied computing (SAC '08). ACM, New York, NY, USA, 707-711. DOI=10.1145/1363686.1363854 http://doi.acm.org/10.1145/1363686.1363854

[19] Petrillo, F., Pimenta, M., Trindade, F. and Dietrich, C. 2009. What went wrong? A survey of problems in game development. Comput. Entertain. 7, 1, Article 13 (February 2009), 22 pages. DOI=10.1145/1486508.1486521 http://doi.acm.org/10.1145/1486508.1486521

[20] Petrillo, F. and Pimenta, M. 2010. Is agility out there?: agile practices in game development. In Proceedings of the 28th ACM International Conference on Design of Communication (SIGDOC '10). ACM, New York, NY, USA, 9-15. DOI=10.1145/1878450.1878453 http://doi.acm.org/10.1145/1878450.1878453

[21] phpBB, 2012. Disponível em: https://www.phpbb.com/ [Acessed June 18, 2013]

[22] Raphael Lima Belém de Barros, Carina Frota Alves, and Geber Lisboa Ramalho. 2009. Investigating the Communication Process in Multidisciplinary Game Development Teams. InProceedings of the 2009 Simpósio Brasileiro de Sistemas Colaborativos (SBSC '09). IEEE Computer Society, Washington, DC, USA, 61-69. DOI=10.1109/SBSC.2009.21 http://dx.doi.org/10.1109/SBSC.2009.21

[23] Rollings, A. And Morris, D., 1999. Game Architecture and Design, Coriolis Group Books.

[24] Rouse III, R. 2004. Game Design: Theory and Practice. Wordware Publishing Inc., Plano, TX, USA.

[25] Runeson, P. and Host, M., 2008. Guidelines for conducting and reporting case study research in software engineering. Empirical Software Engineering, 14th ed

[26] S. Andriole. Storyboard prototyping for requirements verification. Large Scale Systems in Information and Decision Technologies, 12(3):231-247, 1987.

[27] Schell, J., 2008. Art of Game Design: A Book of Lenses.,San Francisco: Morgan Kaufmann Publishers Inc.

[28] Schuytema, P. 2006. Game Design: A Practical Approach (Game Development Series). Charles River Media, Inc., Rockland, MA, USA.

[29] Sharp,H., Rogers, Y. and Preece, J. 2007. Interaction Design: Beyond Human Computer Interaction. John Wiley & Sons.

[30] Sheffield, B., 2009. What Went Wrong? Learning From Past Postmortems. Available at: http://www.gamasutra.com/view/feature/4001/what_went_wrong_learning_from_.php [Accessed May 27, 2012].

[31] Sommerville, I. And Sawyer, P. Requirements Engineering: A Good Practice Guide (Paperback). Wiley (April 28, 1997).

[32] Souza, L.J.Et.Al., 2009. Análises de Documento de Game Design: Interpretação e Resultados Gerados. In VIII Brazilian Symposium on Games and Digital Entertainment. Rio de Janeiro: PUC – Rio

[33] Thomke, S. (2003). Experimentation Matters: Unlocking the Potential of New Technologies for
Innovation. Harvard Business School Press Books. Harvard Business School Press.

[34] Wang, X., Conboy, K. and Cawley. O. 2012. "Leagile" software development: An experience report analysis of the application of lean approaches in agile software development.J. Syst. Softw. 85, 6 (June 2012), 1287-1299. DOI=10.1016/j.jss.2012.01.061 http://dx.doi.org/10.1016/j.jss.2012.01.061

[35] Winget, M.A. and Sampson, W.W., 2011. Game development documentation and institutional collection development policy. In Proceeding of the 11th annual international ACM/IEEE joint conference on Digital libraries - JCDL '11. New York, New York, USA: ACM Press, p. 29. Available at: http://portal.acm.org/citation.cfm?doid=1998076.1998083 [Accessed July 18, 2012]. Yin, R.K., 2008. Case Sudie Research – Design And Methods 4th ed., SAGE PUBLICATIONS

NOTE: the screen captures in the figure 6 and 8 are respectively from the game *Ski Safari* ® [http://www.defiantdev.com/ski-safari.php] and the site [http://archive.org/details/Pbtestfilemp4videotestmp4]. Both were used in this work only for illustrative purposes.