Proposal of Game Design Document from Software Engineering Requirements Perspective

Mario Gonzalez Salazar, Hugo A. Mitre, Cuauhtémoc Lemus Olalde
Computer Science Department
Research Center in Mathematics (CIMAT)
Zacatecas, Zacatecas, México
remylebv@cimat.mx, hmitre@cimat.mx,
clemola@cimat.mx

José Luis González Sánchez
Departament d'Informàtica i Enginyeria Industrial
Escola Politècnica Superior de la Universitat de Lleida
Lleida, España
joseluisgs@diei.udl.cat

Abstract— **The Game Design Document (GDD) plays a key role in the design phase of every game development. A poorly elaborated GDD can lead to rework and loss of investment in production and postproduction phases. To address these issues, an analysis of several available GDDs found in the literature was performed, contrasting our findings with the best practices from Software Requirements Specification (SRS). Our improved GDD incorporates a common understanding of terms, quality assurance, decision making, definition of relations, boundaries, limitations and knowledge of game elements. Finally, our GDD is put side by side with a commercial GDD.**

Keywords-component; Game Design; Game Design Document; SRS; Software Engineering Requirements; Video Game.

I. Introduction

Major software companies are interested in video game development due to its high industry revenues and its growing capability [1]. The design is the cornerstone of video games companies. The terms video game and game are used as synonymous for the sake of variety in the article. Video game development has three main stages. Video games are designed at the preproduction stage, which deliverable is commonly named Game Design Document (GDD). In production stage, the GDD is used for software design, development and validation. In the postproduction stage video games are distributed and monitored after delivery, with the purpose to take corrective actions, along with analysis of company´s expectations on the sales and performance of the video game product. Therefore, a GDD plays a key role throughout the video game development process.

The complexity of video game design has been addressed by the scientific community, specifically in formalizing the preproduction stage. Some argue that applying requirements engineering best practices may avoid rework during production stage [2]. Some others point out that a lack of formality in the GDD reduces investment [3]. In our opinion, requirements engineering best practices may support preproduction and production stages, by bringing structure, detail and establishing relationships among video game elements in order of improving the best experiences during the play time.

A formal GDD may provide support to the transition between the preproduction stage and production stage, reducing rework . We propose an improved GDD based on a comparative analysis of GDD documents found in the literature. Moreover, the resulting GDD template is formalized with other well-known Software Requirements Specification (SRS) standards. Eventually, the proposal was compared with a commercial GDD to extract conclusions.

The article is organized as follows: Section II describes the research method used in our research; section III presents a brief summary of the literature reviewed for our work; section IV presents our analysis results of GDD proposal and SRS; section V summarizes the comparison between our improved GDD with a commercial GDD; and finally, section VI presents the conclusions of our work.

II. Research Method

This section addresses the issue of creating a *GDD using requirement engineering principles to avoid rework and loss of investment*. Thus, we focus on the following research question:

- What characteristics should a GDD have in order to avoid rework and loss of investment in the production stage?

We made a literature review in order to find information that helps us answer the previous question. Our literature review used editorials such as New Riders Publishing, Addison Wesley, Wordware Publishing, Thomas Course Technology PTR and electronic resources such as Google Scholar, Wiley, IEEE, ACM, Springer, Elsevier, Gamasutra, and GameDev.

III. Literature Review

Game design and the use of a GDD are common topics in the video game field. Regarding the GDD many authors agree that there is no established structure for a GDD, since there are significant differences from game to game, specially by video games genre [4–7]. However, there is a set of common elements of game design. We use these common elements to create a structure for a GDD template and the SRS best practice to formalize the template.

A. Game Design Document Structure

The purpose of this section is to identify the principal sections and structure of a GDD. The elements identified are the following: ***overview***, ***mechanics***, ***dynamics***, ***aesthetics***, ***experience*** and ***assumptions and constraints***.

1) All authors suggest that a GDD should include a section that summarizes the key elements of the game to keep the eyes on the road [3–10]. Some authors even include a subsection of goals or objectives of the game [6], [7]. We call this section ***overview***.
2) The term mechanics is used to describe both, elements of the game like a player character or an intended interaction like a challenge. We decided to separate them in order to achieve a better order in the structure of the game and a good potential for reuse. Among the authors the way of describing the game elements have some commonalities. Sections like mechanics, characters or assets list[3–10] but the description sections are not always the same. We call ***mechanics*** to the section referring to the game elements.
3) As mentioned earlier, we divided game elements from game interactions. All authors have common sections that contain the game elements interacting between them and with the player, such as interfaces, levels or artificial intelligence [3–10]. We call ***dynamics*** to the section referring to the game interactions.
4) What the player perceives by his senses has two main aspects, the visual and the auditory. Most authors cover the visual aspects in a document called the art bible. Mark Baldwin [5] suggests an art section abbreviating the art bible in his template. The auditory is mentioned by some authors [6–8]. We call ***Aesthetics*** section to what the player perceives by his senses.
5) Creating enjoyable experiences for the player is fundamental for the game success [11]. Player experiences are enriched by mechanics, dynamics and aesthetics of the game. Playability can be used to link game design to player experience [12]. Therefore, defining the expectations of player experiences may lead to the improvement of the game and to the establishment of a base line to test the experiences in production. No author includes this issue in their GDD. We decided to include a section called ***experience*** to address the expected experiences.
6) The technical limitations are covered by some authors by including a summary of the technical bible in the GDD [5], [6], [10]. We include a section called ***assumptions and constraints***, because they affect directly the game design decisions.

B. *Requirements Engineering Applied to Game Design*

In this section we discuss about the formality on a GDD understood as the structure, relations and detail it contains. We used the IEEE Std 830-1998 [13] (reaffirmed in 2009) for the purpose of comparing the SRS with a GDD.

About the structure, the SRS has three main sections: a) an introduction that provides an overview of the SRS, b) an overall description that contains the general factor that affects the product and its requirements; it provides background for the requirements, and c) details of the specific requirements that a designer and a tester can use for designing and testing a system.

About the relations, there are different types of requirements and the relations between them need to be clear, and cross-referenced to related documents. The specific requirements should be uniquely identifiable. Wiegers [14] gives a hierarchical description of the requirements types dividing them into business, user and system levels.

About the detail, all the specific requirements should be valid in conformance with the following characteristics: correct; unambiguous; complete; consistent; ranked for importance and/or stability; verifiable; modifiable; traceable.

As shown in TABLE I bringing SRS structure, detail and relations may improve a GDD by:

1) The structure of the specific requirements section introduces best practices, like ***organizing specific requirements*** in which the ***object organization*** described in the SRS standard can be used for bringing together the mechanics.

2) The ***document references*** from the introduction section. Would be an aid to game developers in identifying how documents are connected. Therefore all decisions can be traced backward and forward along the development process.

3) The SRS overall description section may enrich the relations of the GDD by making explicit descriptions of the ***assumptions and dependencies***. In this way, developers would know which parts of the game have to be doubled checked later in the project. The ***constraints*** can help the game designers to know the limitations or boundaries to take into account when designing the game.

The SRS specific requirements section may enhance the relations in a GDD. ***Traceability*** and ***stability*** over game elements may help to estimate the effort needed when changes arrive. Ranking ***importance*** and ***stability*** of game elements are crucial for decision-making tradeoffs.

The GDD detail can be improved by the SRS with a ***definitions, acronyms and abbreviations*** section. Making the document easy to read and establish a common language for the creation of a GDD among stakeholders during the development process.

TABLE I. IMPROVEMENT OF GDD WITH SRS

SRS Sections	**Characteristics of SRS for GDD improvement**		
	Structure	***Relations***	***Detail***
Introduction	No improvement.	- Document references.	-Definition acronyms and abbreviations.
Overall Description	No improvement.	-Assumptions and dependences. -Constraints.	-User characteristics.
Specific Requirements	-Organizing specific requirements.	-Characteristics: Traceable, ranked for importance and/or stability.	-Characteristics: Correct, unambiguous, complete, consistent, verifiable, and modifiable. -Software system attributes.

User characteristics part of the overall description on the SRS may improve the GDD detail by allowing game designers to know who may play the game and to adjust the interface complexity for different gamer profiles.

Most of the characteristics of the specific requirements from the SRS help to improve the *detail* in the GDD. A detailed GDD with ***correct***, ***unambiguous***, ***complete***, ***consistent***, ***verifiable*** and ***modifiable*** game elements is more suitable to be used to design the software of the game. Considering ***software system attributes*** in a GDD can help to identify requirements not based on functionality. Besides video games have special requirements not addressed right now by any standard [15], additions can be made to consider requirements such as feelings [16] or any other based on the user experiences [17].

Up to this point, from the identified SRS best practices a GDD should include:

- Relations with other documents.
- Common language for common understanding.
- Knowledge of game parts for reviews.
- Decision making based on tradeoffs of game parts.
- Limitations or boundaries of video game.
- Relation of complexity with gamer profile.
- Organization of game requirements.
- Requirements traceability for decision making.
- Requirement characteristics for formality required in GDD.
- Quality attributes on video games (e.g. feelings and experiences).

IV. AN IMPROVED GAME DESIGN DOCUMENT

This section shows how our proposal is improved with SRS characteristics. TABLE II shows our initial proposal for a GDD based on our analysis of reviewed GDDs and incorporating identified SRS best practices. Noteworthy that our GDD is best suited for video games with a progression path with begin and end, divided in elements like levels, missions or chapters. Next, we present the descriptions for each GDD improved section.

Overview: There are two principal additions to this section. One is a reference subsection relating GDD with others documents in the project. The second subsection establishes a common understanding for the document reader by adding definitions, abbreviations and acronyms of the document.

Mechanics: This section is used to describe objects in the game such as the player avatar or an enemy. Hence, by using the object organization of requirements, mechanics can be described in a practical way. Categories are used to classify and to specify general attributes and behavior that share the elements that belongs to the category, such as "enemy". Elements of the game are described by defining its attributes and its behavior and the rules of how elements can interact with others.

Dynamics: There are ways of organizing requirements that can be adapted to support the dynamics specification of the game, such as features, responses, functional hierarchy and system mode. The gamer profile added in this section can be used to adjust the interfaces and challenges of the game. The elements described in the mechanics section are added to a field or levels in the game world. Objectives, rewards and challenges are described. How the player will learn to play the game and how the game can be balanced are sections that concern to the dynamics too.

Aesthetics: There is no support of SRS elements for defining/capturing what a gamer will hear and see.

Experience: In this section, on one hand, we add the importance and stability of the different parts of the game. Therefore a decision change can be taken knowing the impact of it. On the other hand, we add quality attributes, due that a video game is a software product, some of these attributes are taken into consideration by current standards and other attributes may be incorporated by recent work on feelings and experiences attributes [18].

TABLE II. DESCRIPTION OF GDD

General Characteristics of SRS	**GDD**		
	Section	***Description***	***SRS Characteristics***
Traceable. Correct. Unambiguous. Complete. Consistent. Verifiable. Modifiable.	Overview	In this section are subsections that describes briefly the important aspects of the game.	Relations with other documents. Common language for common understanding.
	Mechanics	In this section are the subsections describing the various elements of the game.	Organization of game requirements (objects organization).
	Dynamics	This section includes subsections that describe how the elements of the game will take action in the game.	Organization of game requirements. Relation of complexity with gamer profile.
	Aesthetics	This section includes subsections that describe what the player perceives directly through their senses. Like what he sees and hears.	There are no sounds and images on SRS.
	Experience	This section includes subsections that highlight important aspects of the game and what you hope to achieve from these aspects.	Decision making based on tradeoffs of game parts. Quality attributes on video games.
	Assumptions and constraints	In this section are the subsections dealing with aspects of the design assumptions and limitations of the game, either technical or business.	Knowledge of game parts for reviews. Limitations or boundaries of video game.

Assumptions and constraints: We add this part to make explicit the assumptions and constraints of the game. Both affect directly the design decisions on the game. Assumptions had to be checked later in the project and may bring changes to the game and constraints limit the game design.

A complete list of all GDD sections can be found on our current template at:

http://www.cimat.mx/~hmitre/GDDTv0.3.pdf

We filled our GDD template with the mechanics and dynamics sections, based on the Donkey Kong game to show a way of how specify mechanics and dynamics for a given game. The document can be seen at:

http://www.cimat.mx/~hmitre/GDD-DKS.pdf

There are other characteristics that affect the whole GDD and not only a specific section as mentioned in TABLE II. Characteristics such as traceable, correct, unambiguous, complete, consistent, verifiable and modifiable are considered the key drivers for a good definition of a GDD.

V. COMPLETENESS, BENEFITS AND DEFICIENCIES OF OUR PROPOSED GDD

We obtained the benefits or deficiencies of our GDD comparing it with a GDD example taken from International hobo´s company, the GDD of the Fireball video game[19], we use this document as it is one of the few GDDs available of a commercial game and is a well known sample of a GDD published on Gamasutra. The purpose of this comparative analysis is to verify for completeness and to identify the benefits and deficiencies of our proposal.

The content in the sections of the Fireball GDD were analyzed to see if they fit in any section of our GDD. TABLE III shows the sections in the Fireball GDD and with the equivalent on our GDD proposal.

About *completeness*, almost all sections of Fireball and their descriptions were alike with our GDD. The only exception was the version control, not included in our GDD, Fireball call it Delta Log.

Referring to *deficiencies*, we identified several improvement areas for our proposal. One is to include a section to describe an example when the game mechanics are complicated or difficult to understand. Another area of improvement is to include a guiding section on how to create missions / levels / chapters of the game. And the last improvement is to add a GDD versions control to manage changes in video games stages.

TABLE III. GDD FIREBALL COMPARISON

Section	Fireball	Section	Our GDD proposal
1.1	Overview	1.1	Game abstract
1.1	Overview	1.3	Game features
1.1	Overview	1.5.3	Target platform
1.2	Vision statement	1.4	Core gameplay
1.3	Branding choices	1.6	Player characteristics
2.1	Game subsystems	2.2	Core game elements
2.2	Avatar	2.2	Core game elements
2.3	Controls	3.6	Controls interface
2.3.1	Jump profile	2.3.1	Interaction rules
2.3.2	Slam profile	2.3.1	Interaction rules
2.4	The players goal	3.2.1	Objectives
2.4	The players goal	3.2.2	Rewards
2.5	Gaining Temperature	2.3.1	Interaction rules
2.6.1	Chains - Overview	3.2.1	Objectives
2.6.2	Chains - The chain counter	2.5	Game log elements
2.6.3	Chains - Font Size of the chain counter	3.5	Game interface
2.7	Field reset	3.5	Game interface
3.1	Environment - Components	2.1	Game elements categories
3.2	Gravity	2.3.1	Interaction rules
3.3	Types of blocks	2.2	Core game elements
3.4	Burning	2.3.1	Interaction rules
3.5	Melting	2.3.1	Interaction rules
4.1	Structure - Overview	3.1.2	Missions/levels/chapters flow
4.2	Ash	2.5	Game log elements
4.3	Rewards	3.2.2	Rewards
4.4	Front End	3.5	Game interface
4.4.7	The hub	3.2.3	Challenges
4.5	Auto-saving	3.5	Game interface
4.6	High Level States	3.5	Game interface
4.7	Overlays	3.5	Game interface
4.8	Path progression	3.1.2	Missions/levels/chapters flow
4.9	Options	3.5	Game interface
4.10	Field list	3.3	Missions/levels/chapters description
5	Audio	4.9-4.5	Aesthetics sound
6	Templates	3.3	Missions/levels/chapters description
6.2	Stages	3.7	Game learning
7	Target Audience	1.6	Player characteristics
8	Delta log	NA	No equivalences where found

Regarding *benefits*, we found that a part of our proposal had no equivalence in the GDD of Fireball. Though, useful parts such as game objectives, game justification, initial scope, game balance, experience, constraints and assumptions, and document information are important for preproduction, production and postproduction stages. Details such as scope, constraints, assumptions, or goals of the game are not treated in Fireball GDD even though these have a direct influence on the game design decisions. On the other hand, the balance and experience in the Fireball GDD are not mentioned, so in future stages such as implementation, we could not measure these experiences or balance the game easily.

Finally we note that some Fireball GDD parts are scattered and not classified correctly, making it difficult to track parts, such as game interface, rewards or the interaction rules of game elements. This may hinder the usefulness of the document especially in the implementation stage.

VI. Conclusions

In this proposal we analyzed various GDD templates to obtain an *initial structure* of a GDD, and then we study a SRS standard template looking for *improvements of structure, relations and details* of GDD. Finally we conducted a comparative analysis of our GDD proposal with a commercial GDD. The purpose of this analysis was to find the degree of completeness, lacks and benefits of our proposal for production and postproduction stages.

Designing a video game with our GDD can offer:

- SRS improvements such as: common understanding; quality assurance; decision making driven impacts; and relations, boundaries, limitations and knowledge of game parts.
- Facing commercial example, the proposal offers completeness: game objectives, game justification, initial scope, game balance, experience, constraints and assumptions, and document information.

Even though, our proposal present issues during design stage including:

- The skills required to prepare a GDD may need SRS experience and training.
- Formalization of the GDD may hinder creativity.
- No versions control.
- Do not show examples of mechanics.
- We don’t have a guide on how to make levels.

Currently, we are using the proposed GDD in different groups of students with different levels of video game development skills to validate its benefits in avoiding rework. As future work case studies will be performed to confirm a reduction of rework and loss of investment.

Acknowledgments

Consejo Nacional de Ciencia y Tecnología (CONACYT), Fondo Mixto de Fomento a la Investigación Científica y Tecnológica CONACyT – Grupo de Ingeniería de Software – CIMAT Unidad Zacatecas.

References

[1] “Video Game Industry Statics,” 2010 [Online] Available: http://www.esrb.org.

[2] D. Callele, E. Neufeld, and K. Schneider, “Requirements engineering and the creative process in the video game industry,” in Requirements Engineering, 2005. Proceedings. 13th IEEE International Conference on, 2005, pp. 240–250.

[3] E. Bethke, Game Development and Production. Wordware Publishing Inc.; Pap/Cdr edition, 2002, p. 412.

[4] R. Rouse, Game Design, Theory and Practice. Wordware Publishing Inc.; 2nd Revised edition edition, 2004, p. 704.

[5] M. Baldwin, “Game Design Document Outline.” 2005.

[6] S. Rogers, Level Up!: The Guide to Great Video Game Design [Paperback]. John Wiley &amp; Sons, 2010, p. 514.

[7] M. K. Oxland, Gameplay and Design [Paperback]. Addison Wesley; 1 edition, 2004, p. 368.

[8] C. Taylor, “Design Template.” 1999 Available: http://www.runawaystudios.com.

[9] A. Rollings, Andrew Rollings and Ernest Adams on Game Design. New Riders; 1 edition, 2003, p. 648.

[10] B. Bates, Game Design [Paperback]. Premier Press; 2nd Revised edition edition, 2004, p. 450.

[11] J. Schell, The Art of Game Design: A book of lenses [Paperback]. Morgan Kaufmann, 2008, p. 512.

[12] L. E. Nacke, A. Drachen, K. Kuikkaniemi, and Y. A. W. D. Kort, “Playability and Player Experience Research,” Proceedings of the IEEE, 2009.

[13] S. Engineering and S. Committee, “IEEE Recommended Practice for Software Requirements Specifications,” Electronics, vol. 1998, no. October. 1998.

[14] K. Wiegers, Software Requirements 2. Microsoft Press, 2003, p. 544.

[15] D. Callele, E. Neufeld, and K. Schneider, “Requirements engineering and the creative process in the video game industry,” in Requirements Engineering, 2005. Proceedings. 13th IEEE International Conference on, 2005, pp. 240–250.

[16] D. Callele, E. Neufeld, and K. Schneider, “Emotional Requirements in Video Games,” 14th IEEE International Requirements Engineering Conference (RE’06), pp. 299-302, Sep. 2006.

[17] J. L. González Sánchez, N. Padilla Zea, and F. L. Gutiérrez, "Human Centered Design," vol. 5619. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 65-74.

[18] J. L. González Sánchez, F. Montero, N. Padilla Zea, F. L. Gutiérrez, “Playability as Extension of Quality in Use in Video Games,” in 2nd International Workshop on the Interplay between Usability Evaluation and Software Development (I-USED), 2009.

[19] B. I. Down, “Design Document: Play With Fire,” 2007. [Online]. Available: http://www.gamasutra.com.

Proposal of game design document from software engineering requirements perspective

Salazar, Mario Gonzalez; Mitre, Hugo A.; Olalde, Cuauhtémoc Lemus; Sánchez, José Luis González

01 Ellen Haas Page 1
13/2/2019 14:38

02 Ellen Haas Page 1
13/2/2019 14:38

03 Ellen Haas Page 2
13/2/2019 14:38

04 Ellen Haas Page 2
13/2/2019 14:38

05 Ellen Haas Page 2
13/2/2019 14:38

06 Ellen Haas Page 2
13/2/2019 14:38

07 Ellen Haas Page 2
13/2/2019 14:38

08 Ellen Haas Page 2
13/2/2019 14:38