Game development life cycle guidelines

2 authors, including:

Yani Widyani

Bandung Institute of Technology

21 PUBLICATIONS   **29** CITATIONS

Some of the authors of this publication are also working on these related projects:

Project   PHD Research in Situational Method Engineering View project

Game Development Life Cycle Guidelines

Rido Ramadan and Yani Widyani
Data and Software Engineering Research Group
School of Electrical Engineering and Informatics, Institut Teknologi Bandung
Email: rido.ramadan@gmail.com, yani@informatika.org

Abstract— **Game is a kind of software with goal to provide entertainment. However, during the real game development practice, simply adopting the software development life cycle (SDLC) is not enough, as the developers face several challenges during its life cycle. To address the problem, game development uses a kind of specific approach called game development life cycle (GDLC) to direct the game development. However, none of the existing GDLCs explicitly address how to successfully deliver a good quality game. This paper presents a new game development life cycle model and guidelines to successfully deliver a good quality game. Several quality criterias are explicitly considered at each phase.**

I. Introduction

THE birth of video games has slowly shifted the meaning of traditional games into a digitalized multimedia games. The term of games refer to the meaning of video games. Nowadays, games can be played in almost any device, and that is why developing games can be a profitable industry. To support the growth of gaming industry, several original equipment manufacturer (OEM) publicly distribute their software development kit (SDK) and application programming interface (API) to attract people to become "indie developer" [1].

According to Pressman, game is a kind of software which provides entertainment [2]. However, game development using only software development life cycle (SDLC) faces several challenges [3][4]. While SDLC is a systematical process of engineering to develop software [2], game is not purely a product of pure engineering. Game also is not just pure art, a creation of creativity and imaginative thinking, but game is more like a craft, created from the combination of interleaving, multidiscipline aspect, from art, music, programming, acting, and the management and integration of those aspects [3][5]. Therefore, a game development requires specific guidelines which govern its development process, the game development lifecycle (GDLC).

The GDLC in question appears in many forms. There are many practices on how a GDLC applied in project, published by both independent (indie) game studio and well-known companies. However, there is no silver bullet, no single GDLC is perfect. There are three questions arise:

1. What are the steps and the phases of a game development process?
2. What are quality criterias that must be considered during each phase?
3. What kind of GDLC which can be the best practice in a proper game development and deliver a good quality product?

The goal of this research is to propose a new GDLC and the guidelines. The guidelines itself is used to properly conduct the proposed GDLC and successfully deliver a good quality game.

II. Term and Definition

A. Video Games

Video games, or simply called games in this paper, is a type of play activity, conducted in the context of a pretended reality, in which the participant(s) try to achieve a pre-determined goal and mediated in a form of digital media [4].

To play a game, one must played it on a proper platform. Gaming platform can be categorized into three types: console, mobile, and cross-platform. Each platform has different characteristics and SDK distribution method. Console SDK has closed distribution method, while most of the mobile SDK can be downloaded freely, although with several constraints. That's why mobile games development has become increasingly more popular [1].

B. Software Development Life Cycle

Software development life cycle (SDLC), also known as software process models, is a development strategy that encompasses the process, methods, and tools which is used to do the software development [2][6]. The typical SDLC phases are shown in Fig. 1.

Fig. 1. Typical software development life cycle.

Analysis is related in gathering and measuring the user requirements to create the software requirement

specifications. In **design** phase those requirements are translated into more detailed models and software modules representation. During **code generation** or **implementation**, the models are translated into source code and executable application. Finally **testing** is conducted to ensure that all elements work properly and meet the specification.

C. Game Prototype Usability Quality Criteria

In this research, the criteria used to assess the game quality is based on Fullerton's game prototype usability quality criteria[7]. Prototype is categorized into four prototype stages in which each stage is related to quality criteria shown in Table I.

Four prototype stages are: **(1) Foundations**, that is the most basic prototype which represents the gameplay basic concepts in the form of either low fidelity prototype or incomplete game; **(2) Structure,** that is a refined version of foundations which already has the core gameplay logic, mechanics, and game rules; **(3) Formal details**, the refinement of structure that includes necessary rules and procedures to make the game fully functional; and **(4) Refinement**, the refined and almost finished game.

Each stages related to quality criterias. **Functional** means the game's feature is playable and operating well. Functional is tested via the accomplishment of each playtest scenario. **Internally complete** indicates all rules, branches, and conditions has been properly addressed. It is tested via observation of inexistencies of three types of errors during playtest. **Balanced** indicates the game's difficulty is just fit, not too hard and not too easy. Balanced is tested via discussion or questionnaire about the game difficulty and game progression. **Fun** means the game is engaging, entertaining, challenging, and makes player keeps coming and coming. Fun is very subjective, therefore it is tested via questionnaire or direct feedback from players. **Accessible** means the game is easy to understand, easy to learn and intuitive enough. Accessible can be tested by observing the player capability to navigate and grasp the control of the game and the time needed to learn the user interface.

III. RELATED WORKS

Game development life cycle (GDLC) is a guideline which encompasses the game development process [8]. Several GDLC have been proposed by different organization, but none of them properly address how to ensure the qualities and successfully deliver good quality games. There are four GDLCs which become the consideration in developing a new GDLC guidelines.

TABLE I
RELATIONSHIP BETWEEN PROTOTYPE STAGE AND QUALITY CRITERIA

Prototype Stage	Functional	Internally Complete	Balanced	Fun	Accessible
Foundations				✓	
Structure	✓			✓	
Formal Details	✓	✓	✓		
Refinement				✓	✓

A. Blitz Games Studios GDLC

Blitz Games Studios [9] defines six steps of their GDLC. Those steps are shown in Fig. 2.

Fig. 2. Blitz Games Studios GDLC consists of 7 phases.

A game development is started from **pitch (1)** to create the initial design and game concept. After the concept is made, it is refined through the **pre-production (2)**, in which the game design, concept art, and the game design document is made. The realization of the concept is done through the **main production (3)** process. Through a lengthy process of main production, the game build then is tested by internal team members, called **alpha (4)** testing. When the build has satisfied the needs of alpha testing, a next phase testing called **beta (5)** testing is conducted to the third party tester. The game is launched in **master (6)** phase.

B. Arnold Hendrick's GDLC

Arnold Hendrick [8] defines five steps of developing a game, shown in Fig. 3.

Fig. 3. Arnold Hendrick's GDLC consists of 5 phases.

Starting point of creating a game is to create the initial design, concept arts, and several prototype in **prototype (1)** phase. The next step, **pre-production (2)**, is to make the documentation in a form of game design document. **Production (3)** is related to the construction of assets, source code, and the integration of those aspects. When the build is ready, **beta(4)** testing is conducted to draw users' feedback. **Live(5)** is when the game has already been pass the testing and ready to play.

C. Doppler Interactive GDLC

Joshua McGrath [10] from Doppler Interactive defines seven steps in game development process. Those steps are shown in Fig. 4.
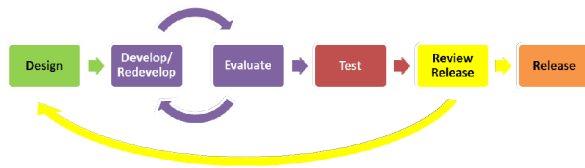
Fig. 4. Doppler Interactive GDLC consists of 6 iterative phases.

This GDLC applies an iterative approach to develop a game. **Design (1)** is related to the creation of game initial design and the game design document. After the design is ready, start develop a game engine for current game in **develop (2)** phase, then test it out in **evaluate (3)** phase. If the build is not satisfying, **redevelop (2)** it. If it passes the evaluation, advance to **test (4)** phase to test the game (not just the engine) to the internal team and doing the bug fixing. After that, the game is released to the third party in **review release (5)**. Repeat the whole process from (1) to (5) until the game is ready to launch in **release (6)** phase.

D. Heather Chandler's GDLC

Heather Chandler [11] defines the four steps in game development process. The corresponding steps is called the production cycle and can be seen in Fig. 5.
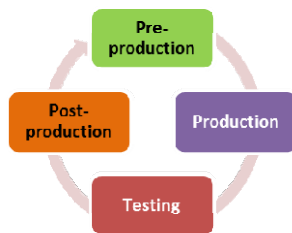

Fig. 5. Heather Chandler GDLC consists of 4 phases in a cycle.

A game development consists of several production cycle started from **pre-production (1)** which defines game design and project planning. After the design and plan has been fixed and approved, it is time move into action to **production (2)** which is related to creation of both technical and artistic aspects. Then, **test (3)** the game and fix the bugs. When a build is considered finished for a single cycle, **post-production (4)** is conducted to deliver the current documentation and post-mortem activities.

IV. ANALYSIS ON GDLCS

Each GDLC has different characteristics and several pros and cons. Blitz Games Studios GDLC [9] and Arnold Hendrick's GDLC [6] apply linear approach in games development. However, the former is more complete since it includes internal (alpha) and external (beta) testing. The trade-off is Blitz Games Studios GDLC [9] takes a longer phase than Arnold Hendrick's GDLC [8].

Doppler Interactive GDLC [10] and Heather Chandler's GDLC [11] are iterative process. However, Doppler Interactive GDLC [10] emphasizes on engineering aspect rather than Heather Chandler's GDLC [11]. Both of them includes internal and external testing, though in the latter, internal testing is included as testing activity rather than explicitly appear as a development phase.

From the four mentioned GDLCs, it can be inferred that the game development process consists of three key activities: **(1) Design and prototype**: the process of creating initial game design, game concept, and put it into a form of playable prototype, **(2) Production**: the process of making the source code, creating the assets, and integrating them as one, **(3) Testing**: the process of playtesting, whether it is conducted by internal team members or third party testers.

The relationship between each GDLC's phase and the three key actvities above are shown in Table II. From the comparison between GDLCs' activities and SDLC's activities, the most prominent difference between SDLC and GDLC is the assets management during game design and production phase. Haddad & Kanode [3] explain that game is created from the synergy of multidiscipline aspects, one of them is the creative aspect. What makes assets significant is the fact that software emphasizes functionality, game emphasizes both functionality and user engagement.

TABLE II
GAME DEVELOPMENT KEY ACTIVITIES

Linear GDLC		Iterative GDLC		
Blitz Games Studios	Arnold Hendrick	Doppler Interactive	Heather Chandler	Generic Phase
Pitching				
Pre-production	Prototype	Design	Pre-production	Design & Prototype
	Pre-production			
Main production	Production	Develop/ Redevelop	Production	Production
		Evaluate		
Alpha	Beta	Test	Testing	Testing
Beta		Review release		
Master	Live	Release	Post-production	

In order to properly handle the game quality, each prototype stage should be addressed in the appropriate development phases. Prototype stages span from the beginning up to near the completion of game development. The relationship between development timeline and prototype stages are shown in Fig. 6.

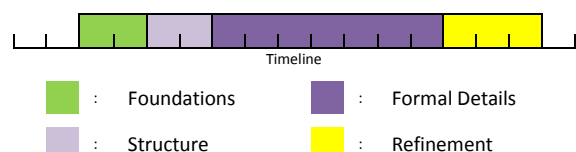

Foundations
Structure
Formal Details
Refinement

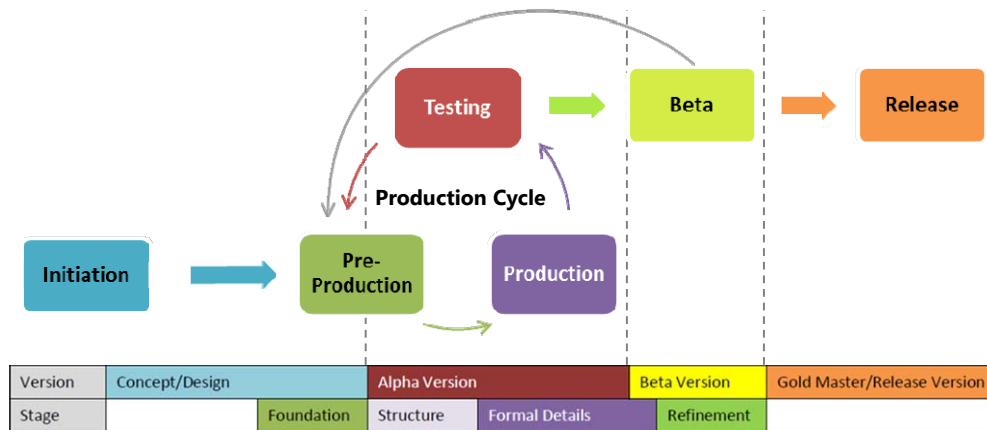Fig. 6. Game development timeline and prototype stages

Fig. 7. The Proposed GDLC model. It consists of 6 development phases. Production cycle consists of Pre-production, Production, and Testing.

V. THE PROPOSED GDLC

The new GDLC is proposed to answer the three research questions: what steps needed to develop a game, what are the quality criteria that must me considered during each step, and how to create a good quality game.   The main principles of the newly proposed GDLC are as follows:
1. The proposed GDLC is developed from the analysis above and derived from the key activities in the relevant GDLC phases.
2. The proposed GDLC applies an iterative approach to enable higher degree of flexibility towards changes during game development.
3. The proposed GDLC is created to address the quality criteria of each prototype stage in order to maintain the quality of the final product.

The proposed GDLC consists of six development phases is shown in the Fig. 7.

A. Initiation

The first step to do in creating a game is to create a rough concept what kind of game that will be created. The output of initation is the game concept and a simple game description.

B. Pre-production

Pre-production is one of the first and foremost phase in the production cycle. Pre-production involves the creation and the revision of **game design** and the creation of **game prototype**. Game design focuses on defining game genre, gameplay, mechanics, storyline, characters, challenges, fun factors, technical aspects, and its elements documentation in game design document (GDD).

After the GDD has been made, a form of prototype is made to assess the game design and the whole idea. In the first iteration of production cycle, the created prototypes are foundations and structure, while in the next iterations, the related prototypes to be refined are formal details and refinement.

Foundation, the first prototype, is related to **fun** quality criteria. Foundation is used to show the mock-up of core gameplay and game capabilities. The **fun** quality criteria in foundation is tested via questionnaire or discussion.

Structure, the refinement over Foundation, and related to **fun** and **functional** quality criteria. The main characteristic of structure is showing both the core gameplay of the game and its related core mechanics such as arithmetic, logic, and game rules. Questionnaire and discussion is used to test the **fun** quality criteria. The **functional** quality criteria is tested via playtesting, where the tester are given some task and goals to achieve according to the testing scenario.

Pre-production ends when the revision or changes of game design has been approved and documented in the GDD.

C. Production

Production is the core process which revolves around the assets creation, source codes creation, and the integration of both elements. The related prototypes in this phases are formal details and refinement.

Formal Details is a refined structure with more complete mechanics and assets. The production activities which are related to the creation and the refinement of **formal details** are balancing the game (related to **balanced** quality criteria), adding new features, improving overall perfomance, and fixing the bug (related to **functional** and **internally complete** quality criteria). Game balancing means adjustments related to game difficulty to make the game's difficulty fit just right.

Refinement is a complete prototype which is the subject of game polishing. The related quality criteria are **fun** and **accessible**. The activities during the refinement are directed to make the game more fun, challenging, and easier to understand. Only minor changes allowed in this phase.

D. Testing

Testing in this context means internal testing conducted to test the game usability and playability. The testing method is specific to each prototype stage.

Formal Details Testing are conducted using playtest to assess the features functionality and the game difficulty (related to balanced). The method to test **functional** quality criteria is via features playtesting. To test the **internally complete** quality criteria, it can be done via playtesting simultaneously with functionality test. When a tester discover bugs, loopholes, or dead-ends during playtesting, the causes and scenarios to reproduce the error needed to be documented and analyzed. To test the **balanced** quality criteria, playtesting with several different treatments is used to categorized whether a treatment is too difficult, too easy, or just fine.

Refinement Testing are related to **fun** and **accessibility** quality criteria. In refinement testing, **fun** is tested via playtest and direct feedback from fellow developers, whether it is boring, frustrating, challenging, etc. **Accessibility** can be tested via observing the tester behavior. If tester find it difficult to play and understand the game, it means that the game is not **accessible** enough.

The output of testing is bug report, change request, and development decision. The result will decide whether it is time to advance to the next phase (**Beta**) or reiterate the production cycle.

E. Beta

Beta is phase to conduct third-party or external tester called beta testing. Beta testing still using the same testing methods as the previous testing method, since the related prototypes in the beta testing are both formal details and refinement. The tester selection method comes in two types: closed beta and open beta. Closed beta is only allow invited individuals to be the participant, while open beta allow anyone who register become the participant.

The quality criteria in beta are closely related to the current prototype stage. In formal details testing, the testers are demanded to discover the bugs (related to **functional** and **internally complete** quality criteria). In refinement testing, the testers are given more freedom to enjoy the game, as the goals are more directed to get the feedback (related to **fun** and **accessibility** quality criteria).

The output of beta testing are bug reports and user feedbacks. Beta session is closed mainly due to 2 reasons, either the beta term ended or the number of specified beta tester has already given their test report. From here, it may lead to production cycle again to refine the product or continue to releasing the game if the result is satisfactory.

F. Release

It is time when the game build has reach final stage and ready to be released to public. Release involves product launching, project documentation, knowledge sharing, post-mortems, and planning for maintenance and game expansion.

VI. GAME DEVELOPMENT GUIDELINES

The proposed GDLC mentioned before are just steps taken to create a game. In order to successfully create and deliver the game, the guidelines are made to accompany the GDLC application. It consists of introduction of game development, role management, initiation, pre-production, production, testing, beta testing, and release [12].

Role Management chapter provides explanation, importance, and responsibilities of each roles.

Initiation chapter provides methods on how to generate ideas and game concepts. To help the ideation of game, the guidelines provides brainstorming help section in a form of questions.

Pre-production chapter provides explanation of each game design elements, such as game description, characters, storyline, control, features and concept arts, documentation in form of game design document (GDD), making **prototypes** and the pre-production phase deliverable checklist. The prototypes made are evaluated using the methods specified in the guidelines.

Production chapter focuses on programming and assets creation. The guidelines provides different kind of assets, method to achieve specified quality criteria, and examples of changes in game archicture. All of them are compiled in a form of deliverable checklist.

Testing chapter provides testing methods related to each quality criteria on each prototype stage and example of each testing method.

Beta chapter explains the importance of beta testing, beta testing type, and provides the methods, checklist, and questionnaire sample in playtesting.

Release chapter explains how to release game package, post-production activities, and planning for game package.

VII. EVALUATION

In order to verify the proposed GDLC and its guidelines, both of them are used in a game development project. The success parameter of this research are the validity of proposed GDLC and the application of this GDLC successfully deliver a good quality game [13].

The proposed GDLC is applied in a project called Feline Project, a project to create a mobile game. Feline Project had lasted for 8 months with so many changes occured, either change request for asset style, main feature, or control method. The Feline Project was done in four production cycle [13].

At first iteration, the initiation of Feline Project produced the game concepts, then it was being refined into game design in pre-production phase. The game design was translated into Feline GDD and a foundation prototype was made. The testing showed the curiosity of each team member (related to fun), so the prototype was refined into structure prototype. The

structure prototype had several core features and showed the game is both functional and fun, so the development was continued. During the development of formal details, first iteration showed there was two lack of quality achievement, so the second and third iteration game design was focused to add functionalities and fix the quality lacking. Formal details from third iteration passed the internal quality testing related to balanced, functional, and internally complete.

The build was beta tested to 60 testers, with 5 testers complained about the game progression felt empty, the rest was good. According to the result, game balanced was not achieved so the focus of the next iteration was to fix it.

The fourth iteration changes several features related to balanced. The prototype had reached refinement phase, so development also took on making the game more accessible. The second beta testing result to 25 testers had shown satisfying result, as all the quality criteria had already been achieved. Through four iteration, Feline Project succeeded in producing a good quality game.

The application of the proposed GDLC in Feline Project showed that it could succesfully create and deliver a good quality mobile game [13]. As for the GDLC validation, the proposed GDLC contains the key activities defined in section IV, with the mapping as follows: Pre-production equals to Game Design and Prototype, Production equals to Production, and Testing and Beta Testing equals to Testing [13].

VIII. Conclusion and Future Works

There are three key phases of game development. They are **design and prototype**, **production**, and **testing**. They combines both engineering process and artistics creational process in term of developing a game. The combination between engineering and arts is the aspect that a simple software development life cycle do not consider as a significant thing and become a challenge in software development.

The proposed GDLC consist of six phases, they are **Initiation**, **Pre-production**, **Production**, **Testing**, **Beta**, and **Release**. The proposed GDLC takes on the iterative approach to enable higher degree of flexibility towards changes during the development process. In order to deliver a good quality game, the GDLC product is assessed through 5 usability quality criteria, which are **fun**, **functional**, **balanced**, **internally complete**, and **accessible**. Finally, the application of the proposed GDLC by following our GDLC guidelines has successfully delivered a good quality game.

To further enchance the result of this research, it is necessary to analyze the relationship between game development life cycle and capability maturity models (CMM) [14] so that the process can be measured in a standard software engineering process measurements.

References

[1] Skillz, "Not Just Hype: The Rise of Indie Game Developers," (2013, March 15). [Online]. Available: http://skillz.com/corporate/2013/03/15/not-just-hype-the-rise-of-indie-game-developers.

[2] R. S. Pressman, Software Engineering: A Practioner Approach, 5th ed. (Book style), New York City: John Wiley & Sons, 2001.

[3] H. M. Haddad and C. M. Kanode, "Software Engineering Challenges in Game Development," in *Sixth International Conference on Information Technology: New Generations*, 2009.

[4] F. Petrillo, M. Pimenta dan F. Trindade, "What Went Wrong? A Survey of Problems in Game Development," in *ACM Computers in Entertainment*, vol. 7 no. 1, pp. 13.1-13.22, 2009.

[5] E. Adams, Fundamentals of Game Design, 2nd Ed (Book style), Berkeley: New Riders, 2009.

[6] S. R. Schach, Object-Oriented and Classical Software Engineering, 6th Ed., New York: McGraw Hill, 2002.

[7] T. Fullerton, Game Design Workshop - A Playcentric Approach to Creating Innovative Games, 2nd Ed. (Book style), Burlington: Elsevier, 2008.

[8] A. Hendrick, "Project Management for Game Development," (2009, June 15). [Online]. Available: http://mmotidbits.com/2009/06/15/project-management-for-game-development/.

[9] Blitz Games Studios, "Project Lifecycle," 2011. [Online]. Available: http://www.blitzgamesstudios.com/blitz_academy/game_dev/project_lifecycle.

[10] J. McGrath, "The Game Development Lifecycle - A theory for the extension of the Agile project methodology," (2011 April 3). [Online]. Available: http://blog.dopplerinteractive.com/2011/04/game-development-lifecycle-theory-for.html.

[11] H. M. Chandler, Game Production Handbook (Book style), Sudbury: Jones and Bartletts Publishers, 2010.

[12] R. Ramadan, The Game Development Life Cycle Handbook (Unpublished work style), unpublished.

[13] R. Ramadan, "Pengembangan Metode Pembangunan Game (Thesis style)," Undergraduate thesis, Informatics Engineering, Institut Teknologi Bandung, Bandung, 2013.

[14] Schorsch, T., "The Capability Im-Maturity Model," CrossTalk, (1996, November), Available: http://www.stsc.hill.af.mil/crosstalk/1996/11/xt96d11h.asp.