A Multivocal Review on Derivation Games

Diego Castro, Claudia Werner
Programa de Engenharia de Sistemas e Computação - COPPE
Universidade Federal do Rio de Janeiro
Rio de Janeiro, Brazil
email: {diegocbcastro, werner}@cos.ufrj.br

***Abstract*—Games have emerged as a prominent form of entertainment, hence establishing the gaming business as a highly lucrative sector. Nevertheless, the process of developing a game can be extremely complex, involving a multitude of activities, components, and team members, making some games take a long time to be produced. The game community has already engaged in the creation of its own games. This practice is commonly referred to as modding. The application of mods in game development can provide several benefits, including enhanced longevity of games, reduced production expenses, and accelerated creation of diverse games within reduced timelines. Nevertheless, the existing mod development process predominantly lacks a structured framework. Hence, the objective of this study is to conduct a comprehensive evaluation that clarifies the main attributes, benefits, difficulties, and methodologies employed in the creation of mods.**

***Index Terms*—Game, Mods, Derivation, Structured review, Multivocal review.**

I. Introduction

By conducting a brief search, it is feasible to identify numerous games available for purchase, as well as several websites that offer modifications for these games. A modification, also referred to as a "mod," involves one or several alterations or adjustments made to a game, which could be related to its mechanics, dynamics, or any other basic element. The classification of it may differ contingent upon the degree of modification: these categorizations involve terms such as patches, tweaks, add-ons, and other designations [1–3]. This method of modifying games can result in a variety of advantages for the company that created the original titles. Among the primary advantages there are: an increase in the number of users, the number of sales, and the longevity of the game [4].

With this is mind, the paper presents a review of the use of mods in development of games. The method used in this research was Multivocal review that is a more complete examination of the literature that aims to elicit as much information as possible about a specific subject; hence, it incorporates data from both white (academic papers, books, etc.) and gray (blogs, websites, videos, etc.) sources. This strategy is typically utilized when there is substantial community support for the study subject and it is necessary to verify practical knowledge on a particular subject [5].

The subsequent sections of this paper are outlined as follows: Section 2 provides a concise overview of the research procedure used in this study. Section 3 includes a comprehensive analysis of the data encountered during the search process. Finally, Section 4 offers a concluding summary of the paper.

II. Research protocol

As mentioned, this research conducted two literature reviews applying distinct search strategies (one for each review): a Mapping Literature Review (MLR) and a Multivocal Review (RM). The first study's objective was to collect data from white literature (academic papers), while the second served as a supplement by collecting data from gray literature (websites, blogs, etc.) [5].

In the initial phase of the investigation, three search databases were utilized, following the recommendation of B. Kitchenham et al. [6]. The search string was executed on the main search engines: Scopus, ScienceDirect, IEEEXplore.

In order to facilitate the execution of this study, a fundamental search string was formulated based on the PICOC framework, which covers the following components: Population, Intervention, Comparison, Outcome, and Context [6]. Combining domain-specific keywords with the logical operator "OR" and fields with the logical operator "AND" produced the search string. This string was utilized for the duration of the search. To validate the search string, two control papers (Modding as part of game culture [3], Serious mods: A case for modding in serious games pedagogy [7]) were used to generate and execute the string in the Scopus database, the first database to which the string was applied. This validation technique seeks to ensure the quality of the search string by returning only relevant articles and author knowledge.

According to some scholars, snowball processes can mitigate the lack of other search engines and supplement the strategy by conducting search through the references and citations of the papers. In order to minimize the loss of some papers and broaden the scope of the search, the forward and backward (one-level) snowballing procedure was employed to verify the references and citations of papers for relevance [8]. The procedures, inclusion and exclusion criteria, quality criteria and research questions will be described below.

The research execution procedure consisted of the following steps:

1) Execute the search string. For searches in gray literature, it was searched for each search string up to page 10 of

 ISBN: 978-1-68558-089-6

google. The search strings were formed by combining the keywords of population and intervention;
2) Apply the inclusion / exclusion criteria based on the title;
3) Apply the inclusion / exclusion criteria based on the abstract;
4) Apply the inclusion / exclusion criteria based on the full text;
5) Apply the quality criteria; Apply snowballing backward;
6) Apply snowballing forward. For searches in gray literature, the snowballing was performed on site references, on links contained within the site.

The inclusion criteria, exclusion criteria, quality criteria, and research questions used in the study were:

Inclusion Criteria:

- Viability Study: The document must be in the context of Mods;
- The document must be in the context of Games and Software Reuse;
- The document must provide data to answer at least one of the research questions;
- The paper must be written in English.

Exclusion Criteria:

- Conference call;
- Studies that can not be fully accessed;
- tudies that are not in the area of Computer Science or Engineering.

Quality Criteria:

The quality criteria employed are derived from Lincoln and Guba, with the objective to evaluate the author's credibility, the transferability of ideas to the new paper, the reliability of the information, and the confirmability of the information [9].

- Is the publishing organization reputable?
- Has the author published another work in the area?
- Does the author have expertise in the area?
- Is the article clear?
- Are the references documented?
- Does this enrich the research?

Research Questions:

- **Q1:** What modifiers are used to create games from other games?
- **Q2:** What characteristics are needed to derive a game?
- **Q3:** What are the advantages and difficulties of creating games from others?
- **Q4:** What tools strategy or frameworks support these changes?

III. Results analysis

The first phase provided a total of 923 papers. This number was reduced to 14 after the inclusion and exclusion criteria were applied to the publications. From these studies, the snowballing process was carried out, and a total of 245 more papers were evaluated. After this approach, 9 papers were included, totaling 23 papers read and assessed.

Based on the findings of the initial phase, it was determined that the gaming community is quite active in terms of development, enhancements, and modifications. Consequently, a new phase was introduced to the study. In addition to the investigation, a search for gray literature was conducted.

TABLE I
Search string of mutator and games.

TITLE-ABS-KEY ((*game*) AND (mutator OR variant OR mods OR modification OR conversion OR add-on OR tweak OR modding) AND (tools OR approach* OR method* OR ideas OR framework* OR mechanics OR interpretation*) AND (creation OR production OR development OR elaboration OR generation OR practice)) AND (LIMIT-TO (SUBJAREA, "COMP") OR LIMIT-TO (SUBJAREA , "ENGI"))

The gray literature search encompassed up to page 10 of Google for each of the search keywords, resulting in 700 links that required validation. The inclusion and exclusion criteria were implemented after visiting each link, resulting in the selection of 21 links for the quality criteria step. Ten links were selected and approved based on the following criteria. The snowball effect was achieved by utilizing backlinks (website reference connections). As a consequence, the entire procedure was restarted for the authorized connections, and 335 additional links were validated. Lastly, fourteen additional documents were added in the search the snowballing process. Table I shows the search string used in the search. The appendix demonstrates the papers and links that were analyzed in this review.

Q1: What modifiers are used to create games from other games?

Increased accessibility to personal computers and the expansion of the Internet, which is disseminating an increasing quantity of content, are closely related to the rise of the mod trend [10]. The community and academy are increasingly generating game adaptations, which help game developers in a variety of ways, such as recruiting new players, prolonging the life of a game, providing new perspectives for the game, and fixing bugs. In general, modifications are referred to as mods and can be viewed as alterations to the original game [1]. In general, a mod is an original game that has had one or N alterations or modifications made to its mechanics, dynamics, rules, or some of its components [11].

Mods are as diverse as the games themselves. They vary in size and complexity and can make minor adjustments to the original game or completely alter its visual design [12]. Modding is the process and technique of modifying or adapting video games. It is frequently a "Do It Yourself" (DIY) strategy that teaches social and technical skills affiliated with innovation by reusing the concept of an existing game. Numerous aspects of the game, including the user interface, game items, bug fixes, characters, and regulations, are modifiable [4]. By altering the rules of a game, for instance, players are able to construct a unique gaming experience [13].

Developing mods is possible by applying mutators to a game. A mutator is a modification to an existing game; for instance, applying mutator M to game G results in the creation of a new game named G [M] [14]. Depending on the number

 ISBN: 978-1-68558-089-6

of mutators utilized, a game may be classified in a variety of ways. There are numerous adaptations and modifications, each serving a distinct purpose [1, 4]. Each of them will be described in more detail in the following [1-3][8][16-27].

Interface customization: The interfaces are designed to emphasize the visual component of the game in order to enhance the experience. This customization entails making changes to the visual element, such as remodeling the accessories, skin, shader, or animation of a character or a game map, altering the game's colors, or altering the information displayed on the screen;

Partial Conversions: Add a new map, a new character, and a new item; increase the game's pace; add small mechanisms, bots, and rules. It is still possible to classify partial modifiers according to the modifications they execute. **(1) Mutators/tweaks:** Modify or add restricted features that have no effect on the game's functionality or mechanics. They may include modifying the game's theme song, increasing the game's speed, or modifying some graphic elements and minor rules. **(2) Add-ons:** They serve as supplementary elements within the game, performing minor adjustments such as modifying the theme's music, accelerating the game, or adjusting minor graphical components and rules. **(3) Mod: Mods:** They are the intersection of the previous two, as they retain the capacity to change rules and configurations.

Total Conversions: Certain changes are so drastic that they result in the creation of new games. A well-known conversion is the CounterStrike mod, which was based on Half-Life. In general, the number of modifiers used differentiates a partial conversion from a complete conversion. When a significant number of modifiers are applied to the point where something new is generated, a complete conversion occurs.

Others: Machinima: It could be seen as the outcome of changes that influence the visual replay of game usage sessions. In this type of modification, games are used for other purposes, such as telling a story, making a movie, or replicating a gaming experience. **Patch:** They frequently concentrate on addressing unresolved problems and creating technical enhancements. This modification is known as an unofficial or fan patch when it is created by a community.

Q2: What characteristics are needed to derive a game?

A game is a type of software development in which designers, developers, and software engineers work together to create an experience for players to live through the game [15]. Once the game is out, the contributors devote their time to updating and adding content to the main game. Modifications may include new game models, textures, music, and mechanisms, as well as complete remakes [15].

There are two primary methodologies for mod development. The first scenario occurs when there is a need for expansion in a particular game by introducing new elements, while the second scenario occurs while seeking games that offer similar characteristics to those wanted in the game under development [16]. Both need the same characteristics.

A game is made up of components that work together to generate the final output. The required qualities for their construction can be determined by defining games. Games are activities that occur in an abstract environment where decisions, actions, and rules are developed with the objective of accomplishing a leisure activity in the form of entertainment or amusement [17]. On this premise, the following aspects must be decided prior to the construction of any game: rules, actions, behaviors, objective, game loop, difficulty, and rewards [16, 18].

Each of the characteristics necessary for the interpretation and evolution of a game will be exemplified below. These features were divided into four broad categories that capture the attributes of the games at a higher level of abstraction. It should be noted that game mechanics were previously divided into actions and behaviors [17-19][23][24][29][32].

[Avatar]

Operation rules: Rules about the player. E.g.: the player can only carry one weapon at a time [4][14][18][28][33-35]; **Transition rules/states**: Understanding the character's state transitions. E.g.: the player can only shoot if he/she has a weapon in his/her hand [16, 19–23]; **Actions**: Commands that can be executed by the character. E.g.: shooting and walking [4, 19, 20, 22–24].

[Game world]

Levels: The game's stages. Strongly influenced by the gameplay that can change from one stage to the next [16, 19–23, 25]; **Rules of objects**: Rules of the objects contained in the world. E.g.: when an object must be locked or unlocked [4, 16, 20, 23, 24, 26]; **Behavioral rules**: Rules of behavior that the world can exhibit. Eg .: if the player collects a specific item it can start to rain [15, 16, 23–25, 27]; **Temporal states**: It works like a state machine; depending on the world's state, it can only go to a specific one [4, 16, 23, 25, 27]; **Mission**: What you want to achieve/complete [4, 15, 16, 19, 20, 23, 25]; **Obstacles**: What you must overcome in the game, its difficulties [7, 15, 20, 22, 23, 25, 27].

[Game play]

Winning and losing conditions: Conditions to win or lose the game [7, 12, 16, 20, 23, 27]; **Strategic dilemmas**: Strategies that can be used in the game. E.g.: combo attacks [7, 12, 16, 20, 23, 27]; **Chains of actions**: Chain of actions that can be combined. Eg .: player action with a map action [16, 20, 22, 27].

[General features]

Rules: Encapsulates the logic inside the system [4, 13, 16, 20, 21, 23, 24, 28]; **Score**: The points obtained by the player throughout the game [16, 20–23, 27]; **Behaviors**: Commands that are executed by the system [4, 16, 20, 21, 23, 27]; **Goal**: What you want to achieve/complete [4, 16, 19, 20, 23, 25]; **Challenge**: What must be accomplished to achieve the goal [4, 7, 16, 20, 21, 23]; **Rewards**: reaching the goal [4, 16, 20, 23, 24, 28]; **Game loop**: Flow of engagement of the game. It is the execution of the game where the player seeks a goal by executing a challenge and being rewarded with something [4, 20–22, 27]; **Interface**: The visual of the game, the game's

 ISBN: 978-1-68558-089-6

sprites, and graphics [7, 19, 21, 23–25, 28]; **Entities**: Objects and elements instantiated within the game [12, 21, 23, 24, 27].

Q3: What are the advantages and difficulties of creating games from others?

Generalizing mod developers' intentions is difficult. There are several elements that contribute to a user producing a mod. Attempting new things, resolving bugs, creating new characters, increasing the difficulty of the game, gaining advantages in the game, extending the game's life cycle, the software was originally designed for a significantly different environment and may require improvement, the official developer is unable to deal with the problems, and so on are among the most significant ones [4, 29].

Modifiers, like games, are complex and time-consuming to create. The time it takes to create a mod varies greatly. The construction process might vary in duration, ranging from a few days to a somewhat longer period, while offering the advantage of using reusable components. As previously mentioned, creating a game can be incredibly time consuming and can take years. However, the time necessary to release a mod is far shorter [4]. Mods allow the community to add to the original game. Depending on the nature of the mod, it may only require one or several releases. For example, a mod that improves the texture of a game may only require one version.

The potential to increase the longevity of games is another advantage that can be ascribed to the employment of modifiers. Every game has an effective life cycle. Modifiers, on the other hand, can extend the life of the game by adding additional instructions, characters, levels, and other factors, giving players more areas to explore [4, 20, 30]. Using the same logic, modifications may help boost sales, income, and profits for original games, as many people purchase the original game in order to play the mod [4, 11, 29].

Another significant advantage of modifications is their ability to draw new players to the game, so extending its longevity. For example, Dota 2 was a Warcraft mod that reached 450,000 daily players five years after its debut and 16 years after the original game's release. As a result, the game's player base and longevity grow [4, 31].

Despite all of the benefits stated so far, there are still issues and hurdles to be aware of when building mods. The first and most serious issue is the initial investment required to create a mod, which is required to comprehend the source code, reverse engineer it, and extract its functionality [21, 32]. Following this line of thought, various investigations have already been conducted with the product line. However, this strategy necessitates an initial expenditure to understand the project's first characteristics [21, 32].

Q4: What tools or frameworks support these changes?

A variety of frameworks and tools make it easier to construct customizations. However, cloning and do-it-yourself have been the most popular methods of mod development thus far. The modder chooses the base game to be modified, confirms the properties he or she wishes to change, and then creates the new game [32]. This less complex approach, known as opportunistic reuse or ad hoc reuse, consists of cloning, copying, and stretching. Opportunistic reuse gives immediate benefits and achieves the intended result. However, project quality is not a priority; extensive reworking results in unanticipated behavior and an unstable software structure [23].

Typically, technologies that allow access to an unencrypted internal representation of the game software are used to modify games. While it may appear that game makers want to dissuade players from customizing their games, this is not the case. In order to increase sales and market share, video game developers are increasingly providing software tools for personalizing their products [29]. Software Development Kits (SDKs) for games/domains provided by game development studios to users represent a current business method for engaging users and assisting in product creation outside the company [1, 15, 33]. In addition to SDKs, which are the most popular manner of accessing the game's source code, various other platforms enable access to and allow alterations to the game's source code. The Creation Kit, GECK, Construction Set, MODKit, REDKit, Modbuddy, and D'jinni are among the most important ones [4].

Another possibility for the development of modifiers is through free software games, in which the end user has complete access to the game's source code and may modify it as desired [1]. However, this strategy is used by small businesses or anonymous developers.

Finally, there are companies that help and support the development of adjustments with the purpose of decreasing difficulties, enhancing game quality and consistency, and developing new ideas. This strategy seeks to improve the game sold by leveraging the suggestions of consumers. The Unreal engine was designed to provide users access to all of its technological components. This allowed it to conduct a number of tournaments known as Unreal Tournaments, in which the developer could express his or her creativity while producing mods [4, 12]. Other firms allow alteration construction as well, although without direct access to the components. Blizzard Entertainment's World of Warcraft, for example, provides a User Interface (UI) modification tool that allows add-ons to modify the user interface panel, resulting in a better gameplay experience.

IV. Conclusion

Game companies are growing in size, earning billions of dollars per year, releasing a significant number of titles each year, and attracting fans of all ages and genres. However, as previously mentioned, designing a game may be a time-consuming process that can take years to finish. The gaming community, on the other hand, is expanding on a daily basis. With such a vast user base, some members may be concerned or disgruntled about having to wait so long for a game to be launched.

In light of this consideration, the current paper presents a multivocal study of papers that involve information associated

 ISBN: 978-1-68558-089-6

with the development of modifications (mods). The primary objective was to find the essential characteristics required for mod creation, as well as to explore the associated challenges and advantages. Additionally, the study aims to identify the most commonly employed techniques in this process.

Through the review, it is possible to conclude that the mod process is widely utilized in the gaming community, with numerous benefits and few disadvantages, but numerous challenges. The current strategy to mod development lacks a standardized protocol, as it is being conducted in an ad-hoc manner. Therefore, the primary objective of this research paper is to categorize the fundamental attributes, techniques, and resources employed in mod development.

In subsequent research efforts, it is anticipated that the insights gleaned from this comprehensive analysis will serve as a foundation for the development of an innovative mod creation methodology.

REFERENCES

[1] W. Scacchi, "Modding as a basis for developing game systems," *Proceedings of the 1st international workshop on Games and software engineering*, pp. 5–8, 2011.

[2] ——, "Modding as an open source approach to extending computer game systems," *Proceedings of IFIP International Conference on Open Source Systems*, pp. 62–74, 2011.

[3] A. Unger, "Modding as part of game culture," *Computer Games and New Media Cultures*, pp. 509–523, 2012.

[4] D. Lee, D. Lin, C.-P. Bezemer, and A. E. Hassan, "Building the perfect game–an empirical study of game modifications," *Empirical Software Engineering*, pp. 1–34, 2020.

[5] V. Garousi, M. Felderer, and M. V. Mäntylä, "Guidelines for including grey literature and conducting multivocal literature reviews in software engineering," *Information and Software Technology*, vol. 106, pp. 101–121, 2019.

[6] M. Petticrew and H. Roberts, *Systematic reviews in the social sciences: A practical guide*. John Wiley & Sons, 2008.

[7] V. McArthur and R. J. Teather, "Serious mods: A case for modding in serious games pedagogy," *Proceedings of IEEE Games Entertainment Media Conference (GEM)*, pp. 1–4, 2015.

[8] R. C. Motta, K. M. de Oliveira, and G. H. Travassos, "Characterizing interoperability in context-aware software systems," *Proceedings of VI Brazilian Symposium on Computing Systems Engineering (SBESC)*, pp. 203–208, 2016.

[9] Y. Lincoln and E. Guba, "Naturalistic inquiry. encyclopedia of research design. 2455 teller road, thousand oaks california 91320 united states," 2016.

[10] O. Sotamaa, "On modder labour, commodification of play, and mod competitions," *First Monday*, vol. 12, no. 9, 2007.

[11] W. Scacchi, "Computer game mods, modders, modding, and the mod scene," *First Monday*, 2010.

[12] D. B. Nieborg, "Am i mod or not?—an analysis of first person shooter modification culture," in *Creative Gamers Seminar—Exploring Participatory Culture in Gaming, University of Tampere, Finland (14–15 January)*, 2005.

[13] G. Cheung and J. Huang, "Remix and play: lessons from rule variants in texas hold'em and halo 2," in *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work*, 2012, pp. 559–568.

[14] B. T. Claude Chaunier, Torben Mogensen, "Mutators," http://www.di.fc.ul.pt/~jpn/cv/mutators.htm, online; accessed 04 March 2021.

[15] L. Poretski and O. Arazy, "Placing value on community co-creations: A study of a video game'modding'community," in *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*, 2017, pp. 480–491.

[16] D. Abbott, "Modding tabletop games for education," *Proceedings of International Conference on Games and Learning Alliance*, pp. 318–329, 2018.

[17] G. Xexéo, A. Carmo, A. Acioli, B. Taucei, C. DIpolitto, E. Mangeli, J. Kritz, L. Costa, and R. Monclar, "What are games?" *LUDES. Rio de Janeiro*, vol. 1, pp. 1–30, 2013. (In Portuguese).

[18] J. Ehrmann, C. Lewis, and P. Lewis, "Homo ludens revisited," *Yale French Studies*, no. 41, pp. 31–57, 1968.

[19] R. Al-Washmi, J. Bana, I. Knight, E. Benson, O. A. A. Kerr, P. Blanchfield, and G. Hopkins, "Design of a math learning game using a minecraft mod," in *European conference on games based learning*, vol. 1. Academic Conferences International Limited, 2014, p. 10.

[20] K. Bilińska, A. Dewalska-Opitek, and M. Hofman-Kohlmeyer, "To mod or not to mod—an empirical study on game modding as customer value co-creation," *Sustainability*, vol. 12, no. 21, p. 9014, 2020.

[21] W. D. Mendonça, W. K. Assunção, and L. Linsbauer, "Multi-objective optimization for reverse engineering of apo-games feature models," in *Proceedings of the 22nd International Systems and Software Product LineConference, vol. 1*, 2018, pp. 279–283.

[22] S. Zuppiroli, P. Ciancarini, and M. Gabbrielli, "A role-playing game for a software engineering lab: Developing a product line," in *2012 IEEE 25th Conference on Software Engineering Education and Training*. IEEE, 2012, pp. 13–22.

[23] C. Lima, W. K. Assunção, J. Martinez, W. Mendonça, I. C. Machado, and C. F. Chavez, "Product line architecture recovery with outlier filtering in software families: the apo-games case study," *Journal of the Brazilian Computer Society*, vol. 25, no. 1, pp. 1–17, 2019.

[24] R. Damaševičius and D. Ašeriškis, "Visual and computational modelling of minority games," *TEM J*, vol. 6, no. 1, pp. 108–116, 2017.

[25] M. F. Shiratuddin and W. Thabet, "Utilizing a 3d game engine to develop a virtual design review system," 2011.

[26] J. Åkesson, S. Nilsson, J. Krüger, and T. Berger, "Migrating the android apo-games into an annotation-based software product line," in *Proceedings of the 23rd International Systems and Software Product LineConference, vol. A*, 2019, pp. 103–107.

[27] G. A. Cignoni, "Reporting about the mod software process," in *European Workshop on Software Process Technology*. Springer, 2001, pp. 242–245.

[28] S. George, É. Lavoué, and B. Monterrat, "An environment to support collaborative learning by modding," in *European Conference on Technology Enhanced Learning*. Springer, 2013, pp. 111–124.

[29] S. Agarwal and P. Seetharaman, "Understanding game modding through phases of mod development." *Proceedings of ICEIS*, pp. 114–121, 2015.

[30] O. Sotamaa, "When the game is not enough: Motivations and practices among computer game modding culture," *Games and Culture*, vol. 5, no. 3, pp. 239–255, 2010.

[31] H. Postigo, "Of mods and modders: Chasing down the value of fan-based digital game modifications," *Games and Culture*, vol. 2, no. 4, pp. 300–313, 2007.

[32] J. Krüger, W. Fenske, T. Thüm, D. Aporius, G. Saake, and T. Leich, "Apo-games: a case study for reverse engineering variability from cloned java variants," in *Proceedings of the 22nd International Systems and Software Product LineConference, vol. 1*, 2018, pp. 251–256.

[33] M. M. Hofman-Kohlmeyer, "Players as content creators. the benefits of game modding according to polish users," *International*

Scientific Journal News, vol. 2, pp. 8–26, 2019.

[34] J. Debbiche, O. Lignell, J. Krüger, and T. Berger, "Migrating java-based apo-games into a composition-based software product line," in *Proceedings of the 23rd International Systems and Software Product LineConference, vol. A*, 2019, pp. 98–102.

[35] T. Dey, J. L. Massengill, and A. Mockus, "Analysis of popularity of game mods: A case study," in *Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play Companion Extended Abstracts*, 2016, pp. 133–139.

[36] T. Tengtrirat and N. Prompoon, "Applying exception handling patterns for user interface customization in software games modification," in *Proceedings of the International MultiConference of Engineers and Computer Scientists*, vol. 1, 2013.

[37] R. F. Patterson, "Mod (video gaming)," https://civilization.fandom.com/wiki/Mod_(video_gaming), online; accessed 05 March 2021.

[38] C. Weeke, "Appropriation & motivation in game modification," *Erasmus University Thesis Repository*, 2020.

[39] R. Ramadan, "Does game modding require programming?" https://www.quora.com/Does-game-modding-require-programming, online; accessed 09 March 2021.

[40] N. Poor, "Computer game modders' motivations and sense of community: A mixed-methods approach," *New media & society*, vol. 16, no. 8, pp. 1249–1267, 2014.

[41] E. Champion, *Game mods: design, theory and criticism*, Lulu.com, 2013.

APPENDIX

A. First Stage

(1:) Building the Perfect Game – An Empirical Study of Game Modifications [4]; (2:) To mod or not to mod—an empirical study on game modding as customer value co-creation [20]; (3:) Modding tabletop games for education [16]; (4:) Migrating Java-based apo-games into a composition-based software product line [34]; (5:) Product line architecture recovery with outlier filtering in software families: the Apo-Games case study [23]; (6:) Apo-games-a case study for reverse engineering variability from cloned Java variants [32]; (7:) Multi-objective optimization for reverse engineering of apo-games feature models [21]; (8:) Visual and computational modelling of minority games [24]; (9:) Placing value on community co-creations: A study of a video game 'modding' community [15]; (10:) Analysis of popularity of game mods: A case study [35]; (11:) Serious mods: A case for modding in serious games pedagogy [7]; (12:) Design of a math learning game using a Minecraft mod [19]; (13:) Applying exception handling patterns for user interface customization in software games modification [36]; (14:) An environment to support collaborative learning by modding [28]; (13:) Reporting about the Mod software process [22]; (15:) A Role-Playing Game for a Software Engineering Lab: Developing a Product Line [22]; (16:) Remix and play: Lessons from rul ts in texas hold'em and halo 2 [13]; (17:) Modding as part of game culture [3]; (18:) Utilizing a 3D game engine to develop a virtual design review system [25]; (19:) Modding as an open source approach to extending computer game systems [2]; (20:) When the game is not enough: Motivations and practices among computer game modding culture [30]; (21:) Modding as a basis for developing game systems [1]; (23:) Of mods and modders: Chasing down the value of fan-based digital game modifications [31]; (23:) Am I Mod or Not? - an Analysis of First Person Shooter Modification Culture [12].

B. Second Stage

(1:) Mod (video gaming) [37]; (2:) Appropriation & Motivation in Game Modification [38]; (3:) Players as Content Creators the Benefits of Game Modding According to Polish Users. [33]; (4:) Understanding Game Modding through Phases of Mod Development [29]; (5:) Does game modding require programming? [39]; (6:) Computer game modders' motivations and sense of community: A mixed-methods approach [40]; (7:) Game Mods: Design, Theory and Criticism [41]; (8:) Computer game mods, modders, modding, and the mod scene [11]; (9:) On modder labour, commodification of play, and mod competitions [39]; (10:) Am I Mod or Not? - An analysis of First Person Shooter modification culture [12].

 ISBN: 978-1-68558-089-6