*Article*

# Redefining the MDA Framework—The Pursuit of a Game Design Ontology

**Rogério Junior [1,*] and Frutuoso Silva [2,3]** (ID)

1    Department of Arts, University of Beira Interior, Rua Marquês d'Ávila e Bolama, 6201-001 Covilhã, Portugal
2    Department of Computer Science, University of Beira Interior, Rua Marquês d'Ávila e Bolama, 6201-001 Covilhã, Portugal; fsilva@di.ubi.pt
3    Instituto de Telecomunicações, University of Beira Interior, Calçada Fonte do Lameiro, 6201-001 Covilhã, Portugal
*    Correspondence: rogerioblesa@gmail.com

**Abstract:** In computer science, an ontology is a way of showing the properties of a subject area and how they are related by defining a set of concepts and categories that represent the subject. There have been many attempts to create a widely accepted ontology for the universe of games. Most of these attempts are defined based on an analytical perspective: few have found frequent use outside universities, as they are not easily translated to the development of games, which is a design perspective. There are some core aspects of the domain that turn this task into a difficult goal to achieve. In addition, game designers tend to refuse a methodology or a structured way of developing a game; the main concern is that it can impair creativity in a field that could not survive without it. A defined ontology would improve and mature the growing industry of digital games, both by enhancing the understanding of the domain and by supporting a structured methodology for designing games. This paper describes the properties of digital games and shows how they make it difficult to create an ontology for that field of study, especially when it comes to a design perspective. It clarifies the closest approach to a unified ontology that there is for the game domain: the mechanics, dynamics and aesthetics framework (MDA). We propose the redefinition of MDA's taxonomy, calling it Redefining the MDA (RMDA), providing better use for the approach from a designer's perspective, embracing the design properties of the domain, and overcoming issues found in the literature of the game domain. The main purpose of this paper is to clarify the MDA framework by redefining its main components, mechanics, dynamics and aesthetics, as a way to make the tool more understandable and useful for game designers. Understanding aesthetics and how developers can invoke them by correctly defining mechanics and creating dynamics is the main focus of the paper. Thus, some examples are provided in order to explain the applicability of the RMDA as a methodology to produce games.

**Keywords:** MDA; RMDA; game design; game ontology; game design methodology; game mechanics; game dynamics

## 1. Introduction

Game designers tend to refuse a methodology or a structured way of developing a game because it is commonly believed within the domain that it could not survive without creativity [1–4]. Nevertheless, there are many specific methodologies/frameworks to assist in the design of games. However, most of them are more oriented to the analysis of games and not to the design process.

An ontology can help by defining the properties, concepts and categories that represent the game area. It would improve and mature the growing industry of digital games, both by improving the understanding of the domain and by supporting a structured methodology for designing games.

Currently, there is neither a structured ontology that is widely accepted in the industry of games nor one that is used in an academic environment to help in the design and development of games. This is not only because the area is a relatively new field of work, but also due to some specific aspects that make it difficult to create an ontology that supports the domain. The lack of an ontology for this domain decreases the efficiency of research in games, and this inefficiency scales when it comes to designing games.

Games are difficult to design and develop, and the difficulty increases every year, as the industry continues to grow and game technology becomes more complex. To keep producing games that can keep up the pace with quality standards, companies need to improve the efficiency and effectiveness of the development process. There are several guides, methodologies and theories that have been created over the last years to help in the analysis, design or documentation of games. Some can be used as design tools, others as documentation tools, and other ones for game analysis, but most of them fail in one aspect or another. Sometimes, they contradict themselves in describing even the basic concepts of the game design domain; for example, game mechanics, which are considered the building blocks of digital games, do not have a single and clear definition. Note that, according to Hunicke et al. [5], games are subdivided into their distinct components: rules, systems, and fun, which are related to their design counterparts, mechanics, dynamics, and aesthetics, respectively. Therefore, the mechanics are considered the building blocks of the games because they are associated with the game rules.

There have been some attempts to define an ontology for the game universe. A project called The Game Ontology (GOP) [6] is a "wiki-based framework that tries to develop a game ontology, which identifies the important structural elements of games and the relationships between them, organizing them hierarchically. The GOP provides a framework for exploring, dissecting and understanding the relationships between different game elements. It is a hierarchy of concepts abstracted from an analysis of many specific games" [7]. Although the project was trying to define an ontology for a domain that desperately needs one, the last entry in the GOP project was in 2015, and it seems to have been abandoned. Another attempt was the ambitious project "The 400 Rules of Game Design" [8]. This project aimed at collecting 400 rules of game design, but only 112 rules were listed; additionally, the last rule was added in 2006, and the project was also abandoned.

The lack of an ontology within the game design domain is a characteristic that impairs the analysis of the field and, to an even great extent, the design of games [4]. "Within the game industry, and to a lesser extent within game research too, there is no fixed vocabulary unified vocabulary for describing existing games and thinking through the design of new ones" [9]. "Many concepts are used quite informally, and terminology frequently overlaps or even conflicts" [4]. This issue makes the growth of the field difficult in many aspects, by both impairing the understanding of the domain from an academic perspective and by leaving the design process reliant upon subjectivity and the previous knowledge of its stakeholders. Because of this, there is an urgent need to define an ontology that would increase the understanding of the domain, especially one that can be useful when it comes to a design perspective of it.

Consequently, a clarification of the MDA framework will be useful to clarify the relations between all the abstraction layers and an emotional response that can be invoked in the player.

## 2. Related Work

Here, we mainly present studies related to methodologies and frameworks for designing games or serious games, although most of them are used more for analytical purposes. However, there are some related works about the concept of design patterns [10], which so far have proven successful in object-oriented design and software engineering. Nevertheless, the application of design patterns to games usually targets game programming rather than game design because most of the games are based on the object-oriented paradigm.

Other works introduce gamification patterns as a semi-formal description of game rules based on the analysis of commonalities in existing gamified applications [11,12]. For example, Kelle et al. [13] described design patterns for learning games, where they explained how game design patterns can be linked with educational functions. However, the focus of this paper is to clarify the use of a framework that makes the game design process more predictable for designers.

According to Salen and Zimmerman [14], one must consider that the design of a game is an iterative process that consists of three steps: design, prototype and playtest. Similarly, Robson et al. [15] described a framework of gamification principles that is also a cycle involving the mechanics, dynamics and emotions. As will be seen below, some of the frameworks for the design of games use concepts which are similar to this one.

Hunicke et al. [5] introduced the MDA framework, which is a formal approach to understanding games. It breaks games into their distinct components and establishes their design counterparts as mechanics, dynamics and aesthetics (MDA). Thus, the mechanics describe the main components of a game, the dynamics focus on its run-time behaviour, while aesthetics correspond to desirable emotional responses of the player. This is a general framework for game design and, by itself, it is not enough to help the design of games. Although it is viewed more as an analysis tool of games, it can be detailed to be used as a production tool, as will be shown in this paper.

Kiili [16] introduced an experiential gaming model that is based on experiential learning theory, flow theory [17] and game design. This model focuses on immediate feedback to the player, on posing challenges and on a progression that is designed to increase the level of difficulty of each task while the learner/player goes through them. This model can be used to design and analyse educational video games. However, as said by the author, the model is insufficient for guiding the entirety of a game design project.

De Freitas and Jarvis [18] presented a framework for describing game-based learning scenarios, in which an approach to the analysis that effectively profiles the learner within the learner group concerning game-based learning is outlined. This framework is based on four dimensions: Representation, Context, Pedagogy, and Learner, and it is centred on the learner, who requires that their preferences and differences are taken into consideration during the development process. This option results in a better relationship between learning outcome and the game. However, it is not easy to translate it into the game design iterative process, for example, mechanics, dynamics and aesthetics are better understood by game designers.

Winn [19] introduced the framework DPE for the design of serious games for learning as an expansion of the MDA framework. It describes the relationship between the designer and the player, where the designer Designs the game and the player Plays it, which results in the player's Experience (DPE). Then, to consider the learning games, he extends this framework including for each component (i.e., Design, Play and Experience) four levels: Learning, Storytelling, Gameplay and User experience. However, this framework is more oriented to the analysis of existing games. It defines a uniform language for the team to discuss and critique a game's design, but it is not clear how four levels are converted into game elements, making it more difficult to use in the design process of the game (i.e., how the learning level is translated into game elements).

Schell [20] also presented a methodology, designated Design Lenses, that proposes a set of questions about several aspects of the game that must be considered in its design. However, this methodology is more oriented to analyse the design of a game than for its production. Nevertheless, it is very useful for the analysis of the game design options because it allows the designer to validate their decisions.

There are other frameworks and methodologies more oriented to serious games and educational games which try to include the learning contents in the game design process. For example, Amory [21], who presented a framework for educational game development, extended the original game object model (GOM), included a new social space object and called it GOM version II. It considers the fact that an educational game consists of a

set of components (objects) each described by an abstract and concrete interface. The pedagogical constructs are associated with the abstract interface, while the design elements are associated with the concrete interface. However, this framework is based on the Object-Oriented Programming paradigm that makes it too theoretical and not easy to be understood by the designers, i.e., team members that are not programmers.

Yusoff et al. [22] described a conceptual framework for serious games, where they enumerate a list of components that must be considered for learning through a serious game. The diagram defines the learning activity as the central component having on one side the learning contents and, on the other side, the game. However, it does not explain the relationship that is established between the learning contents and the game, which makes this framework less useful in its design process.

Marne et al. [23] introduced a conceptual framework for serious game design, called the six facets, where the type of expertise is identified for each facet. In addition, they defined a serious game design pattern library to help the teams solve some design problems and foster communication between stakeholders. Thus, this methodology is useful to understand the role of each expert within the six facets framework and to facilitate the communication between design team members by using the design pattern library. However, it does not define properly a methodology to design a serious game.

Barbosa et al. [24] presented a new methodology of design and development of serious games that facilitates the integration of educational content in games. They introduced the concept of a learning mechanism that must be included in the game, either in storytelling or in the gameplay. For example, within puzzles, mini-games and throughout the game mechanics, as a way to include educational content in the game without disfiguring it (i.e., keeping the fun factor in the first place).

Arnab et al. [25] described a framework to relate Learning Mechanics with Game Mechanics, called LM-GM. It works more as an analytic tool to study the mechanisms joining pedagogical and game features and not as a methodology to design serious games. However, it is useful because it gives an overview of learning mechanics and game mechanics that can be incorporated into the gameplay.

Carvalho et al. [26] presented a conceptual model to better understand the relationships between serious game components and educational goals of the game. This model is based on the activity theory to comprehend the structure of a serious game, called the Activity Theory-based Model of Serious Games (ATMSG). It considers three components, i.e., gaming components, learning components and instructional components, where each one is subdivided into actions, tools and goals. The model is composed of two phases, the analysis of activities and analysis of actions, and it is applied in four steps: identifying and describing activities; representing game sequence; identifying actions, tools and objectives; and describing the implementations. Hence, this model is more useful as an analysis tool than a game design tool.

Roungas [27] described a conceptual model for educational serious games. This model identifies all elements of an educational game and presents a class diagram for an educational serious game. An interesting aspect of this model is that it relates the game level to the learning outcome, i.e., each level requires the achievement of learning outcomes. In addition, it has a web implementation, making it a tool for the design process and a way to keep the game design document (GDD) updated and accessible. This implementation can also overcome the difficulty of other members of the team that are non-programmers to perceive the conceptual model. However, this model is very detailed, and therefore also more complicated to be understood by all members of the team (i.e., game designers, programmers and experts).

Lope et al. [28] presented a high-level methodology where the phases of design, production, and test are similar to the iterative cycle of the design process referred by Salen and Zimmerman [14]. It is divided into five phases: start up, design, production, test, and post-production. It considers in the design phase a set of key components (e.g., scenario, characters, educational competencies, educational challenges, etc.) and the game structure

follows the theatre metaphor (i.e., act, scene, action and dialogue). Besides, it identifies the main design tasks and relates them to different roles (i.e., project manager, computer analytic designer, client, writer and educator).

Pesántez et al. [29] did a methodical review of the approaches for serious game design and identified eleven approaches according to their selection criteria and period. Based on that, they identified four phases for the design of a serious game, which are: analysis, design, development and evaluation. For each phase, they recognized a set of factors and pedagogical and didactic aspects that must be considered in the design of a serious game. However, they do not explain how these aspects and factors must be combined to design a serious game.

Alexiou and Schippers [30] tried to delineate the relationship between game design elements and engagement, intrinsic motivation and user dispositions, as a way to blend pedagogy with gaming technology. While digital games provide multiple opportunities for educators as educational tools, combining pedagogy with gaming technology is not easy. This combination requires a good understanding of the interplay between game mechanics, motivation, engagement, and their effect on learning behaviour.

Spyridon and Ioannis [31] introduced a conceptual methodology based on a task repository, where it is possible to define task dependencies and the skills required for each task. The main idea is to have an adaptive scheme based on artificial intelligence that will prioritise the next learning tasks according to the prior knowledge of the player. Therefore, it is not a methodology to help the design of serious games, but a way to create an adaptive scheme for the learning process inside of games.

Furthermore, Silva [32] presented a methodology to support the design of educational serious games. This methodology is more all-encompassing because it identifies all the main steps that are needed to define the learning mechanisms in an educational serious game, from topic choice to user experience. In addition, it also separates the game's learning contents from other mechanics used to keep the game fun to play.

Recently, Viudes-Carbonell et al. [33] proposed an iterative design for serious games. This methodology defines the concepts to be taught, and then uses them as building blocks for the game design. It includes the educative part at the top and the game design part at the bottom. The game design part is based on the MDA framework, but only considers the game mechanics because they are the building block of the games. Thus, the concepts to be taught should be considered or designed from the practical point of view, as a way to acquire and master them as skills, because skills are developed by practising. This strategy is based on the creation of core game mechanics, but leaves the dynamics and aesthetics out.

There are several frameworks or methodologies to help the design of games or serious games, but most of them are theoretical and they are difficult to translate into the game design process while others are more oriented to game design analysis. Hence, we decided to concentrate on the most used framework, the MDA, and propose its redefinition to make it more useful for the game design process (i.e., from the game designer's perspective). This does not mean that other methodologies, namely those that are more analysis-oriented, cannot be used in the design process too.

## 3. RMDA—Redefining the MDA Concepts

The closest that we have from a widely accepted ontology is the framework proposed by Hunicke et al. [5], the MDA. It has been quite influential and frequently used in universities all over the world. The MDA framework divides the game into three elements and proposes an order of influence between them (see Figure 1):

1.  **Mechanics:** describes the particular components of the game, at the level of data representation and algorithms;
2.  **Dynamics:** describes the run-time behaviour of the mechanics acting on player inputs and each other's outputs over time;

3. **Aesthetics:** describes the desirable emotional responses evoked in the player when interacting with the game system.
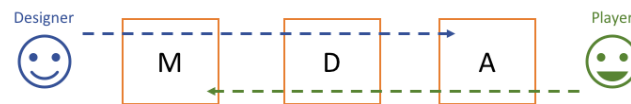


**Figure 1.** MDA framework order of influence.

Although somewhat accepted in the field, mostly used in universities for analytical purposes, MDA is not in use in the games' industry for helping game design work. The main issue that surrounds this framework, according to the literature, is that its concepts lack scrutiny and accuracy and can even contradict themselves in the definitions. This framework has been criticised for leaving certain aspects such as narrative, graphics, sound and interface of the game outside its definition, which can impact the invoked aesthetics [34]. Thus, we propose extending it and clarifying some contents to make it more useful for game designers.

*3.1. Mechanics*

There are many definitions for mechanics in game design. For example, Sicart [35] presented a definition of game mechanics as a way to clarify the vocabulary for defining game structures and systems that allowed a formal and precise description of games. Thus, he defined "game mechanics as methods invoked by agents for interacting with the game world". This definition is based on Object-Oriented Programming (OOP), where the rules and mechanics can be designed as methods and properties, and where the agents are the players. This lack of conceptual precision points to a definitional problem: it is unclear what game mechanics are, and this fact would support neither the definition nor the industry's acceptance of a game domain ontology. Hoping to overcome this issue, the following definition is proposed:

**"Doing responsibilities of Entities, with a purpose to invoke Dynamics."**

In OOP, an entity is defined as any singular identifiable and separate abstraction of an object. It mainly has two responsibilities: knowing and doing. Doing is understood as the actions afforded to an entity, while knowing is the information it possesses. Within the game domain, in a hypothetical first-player-shooter (FPS) game, the player, a gun, an enemy, the map, all are easily identifiable entities, since they are some of the core objects of the game. Other components of the game are also entities, but harder to define since they belong to a lower abstraction level, such as the game camera, gravity, or the UI.

The first part of the definition avoids that entities are mistakenly defined as mechanics. In the previous FPS example, the player, a gun, an enemy, and all the entities described would not be defined as mechanics. If we use the well-known game Super Mario World [36] as an example, the player is an entity that can jump—an action, or a doing responsibility— and also knows its current movement speed, or its current position in the world—knowing responsibility. The knowing responsibilities would already be discarded and not defined as mechanics, while the doing responsibility needs to go through one more step to be defined as mechanics: check its purpose.

The second part of this definition makes it clear that it has to have a purpose: to invoke dynamics. This is important to avoid many unwanted actions of the game's entities being considered as mechanics, such as a game camera controlling the aspect ratio, or the UI responding to a pointer click. These doing responsibilities do not have a direct purpose of invoking any kind of dynamics—they are just necessary to allow the user to interact with the game.

The importance of defining mechanics is clear: it is what the designer can directly control to achieve the wanted emotional objectives, or aesthetics, through the invocation of dynamics. Identifying them is the core of the design of a game and should neither be

overlooked nor cost more time than necessary. This paper hopes to propose a definition that would make it clear what should be defined as mechanics during the design process, and by doing so avoiding unnecessary complexity and/or a huge amount of mechanics defined while developing a game.

To further enhance the step of identifying and defining mechanics, a sub-division of mechanics into three categories is proposed: **Implied mechanics**, **Core mechanics** and **Extra mechanics**. The first stands for doing responsibilities usually contained within the game genre—such as run, jump and die in a 2D platform. If all these actions were to be defined as mechanics, the purpose of avoiding huge amounts of defined mechanics would not be fulfilled. We have to deal with these kinds simply: if it can be tuned to explicitly invoke dynamics, then it should be defined as mechanics.

Core mechanics is the main action or responsibility usually deferred to the main entity in the game genre, such as the player shooting in an FPS or attacking enemies in a fantasy Role Playing Games (RPG). These mechanics are implied ones and also represent the most important in invoking dynamics, and as such, they should always be explicitly defined.

The last category is the Extra mechanics. These are the mechanics that are usually defined later, sometimes after prototyping. This is the category of mechanics that we could call the "extra" of a game or the ones that will create the difference between similar genre games. One example would be the camera shaking or have a blur effect in terror games: it is an extra mechanics that can be defined since it has a clear purpose of invoking one or more dynamics contained within the game.

Mechanics are the only thing that designers have full control over when trying to make the game fulfil its emotional purpose, or aesthetics, through the creation of dynamics. Understanding how they work is essential: It shows where the team should spend time and resources working and how it can affect the player experience when interacting with the game.

*3.2. Dynamics*

The RMDA framework proposes a new definition to dynamics in pursuit of a structured ontology for the domain, one that would not only be used in an analytical perspective, but one that could be translated to the real game design world. The lack of a precise and defined taxonomy does not support this objective. How can a development team work on achieving the aesthetics purposes of a game if they do not explicitly work with the dynamics that invokes them? There are two common ways the industry deals with this: by relying on previous similar games' dynamics and/or by prototyping it in an ad hoc manner until wanted aesthetics somehow emerge from it.

The first way is more common, and it brings as a consequence to game design the impairment of creativity: the team usually does not have a clear knowledge of how dynamics work together to invoke the wanted emotional purpose (aesthetics), so as a risk avoidance technique they rely on copying dynamics from previous similar games, especially if the team does not fully understand how it emerges, or how it works on invoking aesthetics. This is not a healthy process for the domain, as it can lead to an evident resemblance between different titles. How many RPGs do not have the dynamics of hunting monsters for in-game currency rewards (e.g., gold coins)? This dynamic is so present in this genre that it is hard to imagine games without it, but they do exist. A game called Pokemon Yellow [37] for Game Boy was a huge success and created another dynamic that would encourage the players to hunt monsters: they could capture them, and the player's captured monsters could increase their level by earning experience.

The second way is an inefficient way of working with dynamics for a game: although some useful dynamics emerge only in prototyping and play-testing, without a clear understanding of how mechanics work to create them, the team will hardly be able to invoke all dynamics that could be created to enhance the wanted aesthetics, and it will certainly be more time costly. If the taxonomy was well defined and understood, the development team could intendedly work on specific mechanics to create expected dynamics, without

relying on "luck" and hoping for it to emerge during play-testing. Furthermore, of course, there is a greater risk of unwanted dynamics to be shipped in the released game.

The importance of a coherent definition for dynamics is evident but not yet achieved by the game domain, especially when it comes to a definition that can be used in the design process due to its complexity and virtually infinite outcomes, there is no apparent benefit for the team to waste time trying to find them. This concept is mostly used in an analytical perspective: analysing already existing games and dynamics seems a more achievable task.

To achieve the purpose of creating a framework usable in a design perspective, this paper proposes the following definition for dynamics:

**"Dynamics are the predictable runtime behaviours that emerge from Mechanics, with a purpose to invoke Aesthetics."**

It is important to note that since this is a design-based taxonomy in a domain that has absorbed the iterative process, it should be considered as predictable the dynamics found in any iteration step of the process. The proposed definition hopes to support the following core aspects found in the literature, while clarifying the relationship with mechanics to avoid misinterpretation and filling the gaps surrounding the lack of support to the complex nature of dynamics.

1. **Its runtime behaviour.** Dynamics can only emerge while the interaction player-game is happening;
2. **The close relationship between mechanics and dynamics.** By acknowledging that dynamics emerge from mechanics, it is clear how designers can create/control dynamics: by working on the mechanics that creates them;
3. **The unpredictable nature of dynamics.** Due to its number of possibilities, it can be harsh to find all dynamics while developing a game and, as a consequence, causing this concept not to be very useful within a designer perspective—which is the objective of this paper. The proposed taxonomy hopes to avoid this issue by only identifying as dynamics the predictable behaviours, giving more emphasis to the designer's point of view of the taxonomy. Since the concepts here are iterative design based, we consider as predictable dynamics the ones that are found either before or after prototyping, as the complexity of the games nowadays can make it almost impossible to predict all dynamics that one user can identify within the game world, before setting it in motion;
4. **The complexity of dynamics.** There are virtually unlimited possibilities afforded to the player backed up by the exponentially growing complexity of games nowadays, making the finding of all dynamics unachievable. This complex nature of dynamics can sometimes impair the team to work towards describing and redefining them during the design process. To clarify the possible dynamics and how to work with this taxonomy, we propose a division into two categories of dynamics: **Simple dynamics**, or dynamics that only emerge directly from mechanics; and **Complex dynamics**, dynamics that emerge from other dynamics as well as from mechanics.

Dynamics are the bridge that connects the designers to players. It is what emerges from the designer's actions and creates the emotional response in players. Having a clear understanding of how mechanics are the foundation for dynamics, and how dynamics work on creating aesthetics, the development team can have a clearer path to follow in order to achieve the game's emotional purposes. Correctly defined dynamics will support the development team by showing them where and how to work, increasing efficiency in the design process.

*3.3. Aesthetics*

There is no clear way of determining what makes a game fun. First, even defining fun is a hard task: one can dive deep into philosophy and psychology and still fail to find a clear definition for it. From psychology, we know that intrinsic and extrinsic motivation is what boosts people to do things. Playing a game to have fun, that is intrinsic motivation.

However, games have the capabilities to evoke intrinsic and extrinsic motivations. A good game must motivate the player through extrinsic rewards.

According to Csikszentmihalyi [17], when a person performing an activity is immersed with focus, involvement and enjoyment in the process of that activity, losing the sense of space and time, this person reaches the mental state of flow. Reaching the state of flow in a certain activity makes it an 'optimal experience', since the user gets a high gratification from it. In order to keep a person in a state of flow, the activity needs to reach a balance between the challenges of the activity and the abilities of the user [38]. In video games, to keep the players interested (i.e., having fun), the game should be designed in order to maintaining the players in the state of flow. Similarly, Lazzaro [39] presented "The 4 Keys to Fun", where she describes the main reasons why people play games: 1—Novelty; 2—Challenge; 3—Friendship; 4—Meaning. These four keys are related to the game mechanics, namely by the hard fun, easy fun, the people factor and altered states, respectively. Thus, it is crucial to have it in mind during the game design.

Developers need a way of better understanding the emotions their games are meant to invoke, and this is where aesthetics can help. At the time of MDA's introduction in 2004, this involved the concept of 'fun', but as Hunicke et al. explain: "defining fun is not the point here, but to create a vocabulary that can be used as a compass to lead the team towards the expected player's emotional responses" [5]. These authors sidestep the difficulties in defining fun by introducing a taxonomy to rationalise fun with aesthetics, based on Le Blanc's website, "The 8 kinds of fun" [40]:

1.  Sensation: Game as sense-pleasure. Games that have a strong characteristic of engaging senses—either by visual art style or sound design.
2.  Fantasy: Game as make-believe. Games that create a make-believe world, an alternative reality for the player.
3.  Narrative: Game as drama. Games that have a well-written narrative, with defined characters and/or world.
4.  Challenge: Game as obstacle course. Games that have a competitive feeling, that invoke the thrill of competition. It should be noticed that it can also happen in single player games, the fun arises upon overcoming a difficult challenge.
5.  Fellowship: Game as social framework. Games where one of its aspects is to engage the player into social relations, with friends, family or other gamers.
6.  Discovery: Game as uncharted territory. Games that motivate the player to explore and discover new features contained in them.
7.  Expression: Game as self-discovery. Games that enable the player to find ways of expressing himself.
8.  Submission: Game as pastime. Games that focus on creating a distraction for the player.

Based on this idea, the RMDA framework proposes that the aesthetics definition should make it clear that the player is ultimately responsible for creating their own emotions, and the following taxonomy is proposed:

**"Aesthetics describe the desirable emotional responses that the player can invoke when interacting with the game system."**

The first aspect to notice in this definition is how there is no direct relation with the visual or art style of a game, but with the emotions that the player can invoke. The second point around this definition is how it contains the runtime characteristic of the game domain merged with it: "when they interact with the game system". By embracing the runtime aspect, aesthetics deals with the experience that emerges when the player interacts with the game, and the RMDA framework proposes to make it clearer that the player is ultimately responsible for creating their own emotions: a game does not directly invoke emotions—it offers tools and rules, in a virtual world that allows the player to create their own emotions.

Understanding aesthetics and how developers can invoke them by correctly following the proposed pattern (e.g., Figure 2) will ease the process of dealing with unexpected end emotional results: it becomes clear which mechanics should be directly changed to improve

the end result. Not only would it help dealing with unexpected results, but also increase efficiency when creating new aesthetics due to the clarity of how one layer would affect the next layer—mechanics (Core, Implied, Extra) creates dynamics (Simple, Complex) that invokes aesthetics. Thus, the relation between mechanics, dynamics, and aesthetics is redefined and game designers have more control to design an emotional response the player can invoke.
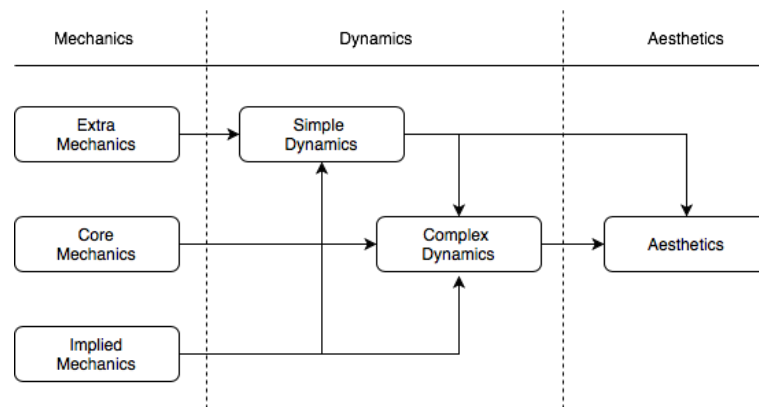
**Figure 2.** Diagram of the RMDA framework.

## 4. Discussion and Applicability of RMDA

To prove the analytical use of RMDA, we discuss the design process and illustrate some situations with commercial games.

The first step when developing a game is to define the core aesthetics of it—or what is the main experience the game would allow the player to invoke. Sometimes this is a subjective decision: when the game idea arises from the inspiration or dream of a small indie development team, for example. The main aesthetics here comes from the designers, and its definition may be supported by analysing similar games that were sources of inspiration. There are cases where the aesthetics is pre-determined by the stakeholders. If a team is hired to create a sequel, an advergame or an educational serious game, for example, the aesthetics can be fixed by the contractors. The idea here is to define one main aesthetics to support the developing process.

After the main aesthetics are fixed, the team should work on defining one or more secondary aesthetics, as a weight to be counted when decisions are to be made. This step should use some information regarding what is needed to define them, which are, but not limited to:

1.  The knowledge: The development team has to be completely aware of its capability. With an experienced graphic design team, it would be wise to select Sensation as secondary aesthetics. A small indie team who are mostly software programmers should not select Narrative as a priority, since (probably) they would not have the necessary skills to efficiently achieve it.
2.  The Target: Understanding what type of player the game focuses on is crucial for its success, and it is extremely important when defining aesthetics. By identifying the player's type, the team can analyse already existing games that share the same target and use them as inspiration when defining aesthetics. Good knowledge of the target is useful in many ways, such as understanding what is the main platform they use for games: it can be counterproductive to focus on sensation aesthetics and create extremely detailed 3D graphics if the target mainly plays on mobile phones, as they probably do not have the graphical capacity to render it.
3.  The Market: Using business related information, the team could focus on which aesthetics is usually expected by its target. What is the mainstream game's aesthetics? Are there any new and common dynamics being introduced in games? If so, which

are the aesthetics they hope to evoke? Do players nowadays care more about graphics (Sensation) or story (Narrative)?

4.  The Cost: By having a clear understanding of its capacity, the team should be aware of the costs to achieve each aesthetics. Which aesthetics would be easier/less costly to be worked by the team? Which aesthetics will not be worth to spending time working on?

When first defining the prioritized aesthetics, the designers know where to put a bigger effort into the design process. It would not be optimal (for most game companies) to hire an entire orchestra to compose and record the soundtrack of a puzzle game that is to be played as a pastime. Submission is the aimed aesthetics here, not sensation. Neither to allocate all the graphic design team to create an extremely well-detailed environment world in an online competitive racing game where challenge is the priority over fantasy and discovery. In other common scenarios, the team might decide to work more in a detailed 3D world rather than redefining the mathematical progression of the player's attack stats in an RPG because the main aesthetics could be fantasy or discovery rather than challenge.

Let us suppose that a fictional company has an FPS code template and wants to create a game from it (Knowledge, Cost). Due to some understanding of the market (Market), they decide that the game will be cartoonish and multiplayer co-op, aimed at teenagers under 18 (Target). The main aesthetics is fellowship, since it is a co-op game. Secondary is sensation (cartoonish) and challenge (FPS online). To support this aesthetics, dynamics should be defined, and from there, create/tune mechanics that would support them.

After the aesthetics are defined, dynamics should be proposed in order to achieve those emotional objectives—and from these dynamics, the team can define and tune mechanics that invoke them. Based on the proposed aesthetics and genre (FPS), the team can think of defining simple dynamics such as "killing enemies" and complex dynamics such as "defeating boss as a party" and "group loot hunting". From there, it is time to work on mechanics that are contained within these dynamics in a way that would support the intended aesthetics.

Starting with the simple dynamics of "killing enemies", the team could work on the mechanics that would support this dynamic, such as "getting shot", afforded to the enemy's entity. In these mechanics, the team could create different and more detailed animations so that enemies got damage if being shot by more than a specified number of players: it would improve not only fellowship aesthetics, but also sensation. If this change is not viable to the team for some reason (such as not having enough animators, or the possibility that engine used would not support it), the team could create different audio-visual effects for the moment in which the enemies get shot, as it would also improve aesthetics. Another option would be to make enemies more vulnerable from one side than the other, so it would be easier to kill with more players together, improving both fellowship and challenge aesthetics.

Moving on to the complex dynamics, the team could work on the defined dynamics "defeating boss as a party", which would need both mechanics (such as "player shooting") and dynamics ("killing enemies") in its creation. By always having the wanted aesthetics in mind, this dynamics could be enhanced by working on both the mechanics that directly invoke these dynamics and the mechanics that indirectly support it—in this example, the ones that invoke "killing enemies". Some ways to enhance it could be tuning the "getting shot" mechanics that indirectly supports this dynamics provided to the boss, and make it receive extra damage when it gets shot by two or more players in a short period of time (Fellowship and Challenge). To improve the dynamics that is directly involved ("player shooting"), they could make the UI show to all other party members how many times they have to shoot the boss and cause this extra bonus damage after one party member hits it, as it would also increase fellowship and challenge.

Another complex dynamic is "group loot hunting", also common in co-op games. By defining the mechanics of "picking up items" provided to the players (a mechanic that would directly influence the defined dynamics), the game could make it support the

fellowship aesthetics by improving the UI in order to show all players when someone finds a rare item (as it would also improve challenge). A dynamic that also supports this one is "killing enemies", and as such it could be tuned to enhance this one as well. The team could, for example, make the enemies give more rewards when they get killed by more than one player.

During play-testing, if the team notices an unexpected dynamics emerge, party members can sometimes get really far apart from their friends, and it decreases the fellowship aesthetics of the game. When an unexpected dynamic emerges, the team has some choices: they could remove it from the game, ignore it or maintain it, based on the effects that it has on the proposed aesthetics. If an unexpected dynamic does not influence the wanted emotional responses, developers can choose to ignore it. One example is a game called Elders Scrolls Skyrim, by Bethesda Studios [41]. In this game, the player controls an avatar in an extremely well-detailed 3D world, full of creatures to be defeated, caverns and secrets to be discovered and quests to be fulfilled. Exploring and fantasy are certainly priorities in this game. Regardless of how many dynamics the designers expected and created for the player, we must remember that the player is the ultimate creator of their own experience: the game is nothing but a tool. The player can create a dynamic such as trying to reach the highest point of the map, only for the sake of doing it. This is (probably) not an expected dynamic, and it can surely be ignored since it neither has the potential for being redefined to support the wanted aesthetics nor impairs the player to achieve them.

If the dynamics goes against the proposed aesthetics, they should remove it from the game, by removing or changing the mechanics that invoke it. One example is a Massively Multiplayer Online Role-Playing Game (MMORPG) called Tibia, by Cipsoft [42]. The game has a dynamic of hunting where the avatar increases its level by defeating monsters that give the player experience points. In this world, when a player kills a monster, he has to wait some predetermined time for the monster to respawn again, and this time ranges from some seconds to several hours. This respawn time can make it difficult for the player to keep killing the same monster at the same spot and increase their level, and from this the hunting dynamic has a characteristic that is the fact that the player has to keep looking for new places to find other monsters in order to level up, enhancing the discovery aesthetics. Another characteristic of some monsters is that they can summon other creatures to help them in battle. With the monster's mechanics to summon, an unexpected dynamic was found: the player can choose not to kill the summoning monster, and only kill its summoned creatures, for as long as he wants. Since the summoning time is way less than the re-spawning time, the player would increase their level faster than those who do not use this dynamics, drastically impairing the challenge aesthetics by allowing this unfair advantage between players. This dynamic also goes against the exploring aesthetics: the player could keep on hunting without leaving the same place.

In this game, the dynamics and mechanics that invoke this unwanted behaviour belong to the core of the game genre, and as such it is not possible to remove one or more in order solve this issue. The designers should not remove the summoning nor the re-spawn time-delay mechanics from the monster, and less so to remove the dynamics of killing monsters or hunting from the player. This issue is a complex case, since it is a massive multiplayer game with a strong challenge aesthetics: it had to be solved to avoid misuse and creation of an unfair advantage for some player using it. These complex cases require extra caution: even small changes in mechanics can cascade through dynamics [5,35] and changes in dynamics can change and/or create new unexpected behaviours inside the game—an exponential complexity. The design team could have come up with a solution like limiting the number of monsters that could be summoned. That could work for the unwanted behaviour of hunting the same summoned monster over and over again but could also elicit other problematic behaviour. For instance, one player with a higher level could keep killing the summons until the monster reaches its limit, then let a lower level player—who should not be capable of killing the monster—kill it, since there are no

summons to support the monster. Furthermore, this new dynamic would also impair the challenge aesthetics.

In this example, we can see how a detailed analysis around redefining dynamics is important. Having to predict possible dynamics is not an easy task—but it can be simplified if the team understands the basic concept of how it emerges and how it influences the final aesthetics. The solution created by the designers was to keep all mechanics and dynamics that emerged the behaviour the same, but to remove a characteristic of the summoned monster entity: it would no longer afford experience points and loot to the player. This change completely removed the unwanted dynamics of levelling up only using summons, since it is no longer possible while keeping the mechanics and dynamics that are considered important to the game.

There are cases where the unexpected dynamics can actually enhance the proposed aesthetics of the game, and in this case, the team can leave it as it is or change its mechanics to improve its capability of achieving the emotional purposes even more. If we use as an example a classic MMORPG game, the dynamics of hunting for loot or levelling up is usually defined at the beginning of the development process. If the game has exploring and challenge as an aesthetics priority, the team should work on these dynamics towards the wanted emotional responses: they could create different hunt areas and monsters as an encouragement for the player to explore uncharted territory and balance the enemies and their rewards to a fair challenge between players that hunt in different places. In games like this, it is common to give rise to a dynamic where the player would only hunt a specific monster with a specific weapon or specific skills due to easier rewards when comparing it to other hunting places and as such directly impairing the discovery aesthetics, since the players would not have the incentive to find new hunting areas. This dynamic would also go against the challenge aesthetics: players who are not aware of this behaviour will be at an unfair disadvantage. In this scenario, it is not optimal to remove the dynamics of hunting that specific monster using a specific weapon: the design team worked on creating and modelling them, and reducing the variety of monsters and weapons in a game can reduce the discovery aesthetics. A solution that is usually adopted is to change some entity characteristics (or knowing responsibilities) involved, but not the mechanics afforded to them, and as a consequence, the dynamics that emerged from them will be kept. In this case, the designers could change the health of the monsters to increase difficulty or lower the experience points given to players that kill it—in both cases, the dynamics of hunting that monster will be kept—but the unfair advantage is no longer within it.

In our FPS fictional game, the team decided to remove the unexpected dynamics of the player getting too far away from party members, by creating new mechanics provided to the player: they can teleport to party leaders when out of combat. These new mechanics would certainly remove the unwanted dynamics of players getting far apart from other party members and should not negatively influence the proposed aesthetics for the game. It can even be tuned to enhance other proposed aesthetics, such as allowing the creation of new mechanics of making the player that dealt more damage in the last combat becoming the leader, and as such enhancing the challenge aesthetics as well.

Although this is a really simple example and it may sound obvious for developers familiar with the FPS genre, it should be clear how RMDA would support this fictional game: it is a fact that the team will work on animating the enemies getting shot, but for how long? With what priority? The same applies for the boss getting defeated by a group: co-op games usually give more rewards to the players, but should the team spend more time in audio–visual details of these rewards, or do the mathematical probabilities of rare items drop? Is it worth hiring professional voice actors to turn dialogues into more realistic ones? These choices should be made to understand how they can affect the end emotional results (Aesthetics) of the game—not by blindly copying from other games of a similar genre. This framework can also support dealing with unwanted dynamics that would emerge after play-testing, prototype, or release of the game, by clarifying when and how to ignore/maintain/remove it by working on the mechanics that invoke it.

By understanding the flow of RMDA and its proposed taxonomies, developers can lead the design/update process towards more explicit objectives with ease. It can be difficult to bring new players to already existing multiplayer FPS co-op online games based on challenge aesthetics since old players will have more understanding of its concepts and it would not be exciting to help new players nor to go against them. Game designers can create new dynamics as an incentive to bring these new players to the game: some more common, such as making new players receive an in-game bonus (such as attack or defence), or more complex, such as creating new weapons that would not require as much aiming skills as regular ones (grenade launchers, weapons that can heal your team, guided weapons) and/or creating exclusively rewards for when a player saves or helps another, creating a reason for old players to play with new ones.

To further explain how RMDA can be useful when designing games, some ways that it could be viewed as supporting some well-known games are proposed. It has to be made clear that the authors do not claim that this solution was supported or endorsed by the creators of these games.

A game called Diablo III was launched in 2008 by Blizzard, as a sequel of its famous Diablo series. Diablo is a game where the player controls an avatar in an alternative terror world, where he has to kill monsters and find better equipment to get stronger so that he is able to defeat the ultimate boss: Diablo. There was a dynamic in Diablo II where the player had to repeatedly hunt some hard bosses in order to "farm" or "loot" (i.e., find) rare weapons and equipment, so they would get stronger and finish the game in a more difficult setting.

This "hunting for loot" dynamic invoked challenge aesthetics and was at the core of Diablo II—players loved it. When Diablo III was released, Blizzard introduced a new system in the game, an auction house, where players could buy and sell equipment from other players using real money. This created a dynamic where some players would farm items for real-life profit, and impaired the dynamics of loot hunting from other players: they could easily buy rare items in the auction house, and as such there was no real incentive to loot hunting, and as a consequence, this created a huge impact on the challenge aesthetics.

The diagram of Figure 3 represents how this scenario could have been translated to RMDA: some of the mechanics and dynamics that are related to this issue have been defined and presented in a way that would make it easier for developers to understand how to deal with the problem. They had to remove this dynamic of the game after complaints of its users, and to do so, they removed the mechanics that would invoke it (buy and sell items in the Auction house), without changing the other wanted mechanics. Blizzard removed the auction house from the game, reinserting the "hunting for loot" dynamics, and it is a success to this day.

Another example is a game called League of Legends, by Riot Games, where the player can choose among several heroes to play as a team against other online players. These heroes have stats, like attack damage and defence. They also had a percentage of chance to dodge an incoming attack, which can be enhanced by obtaining items that increase this chance. One of the heroes that the player could select had a particular skill that could also increase their dodging chance, and it would add to the chance by obtaining items. This created a dynamics where players could select this hero, buy particular items and achieve a 100% dodging chance, therefore making them invulnerable to attacks from other players (see Figure 4).
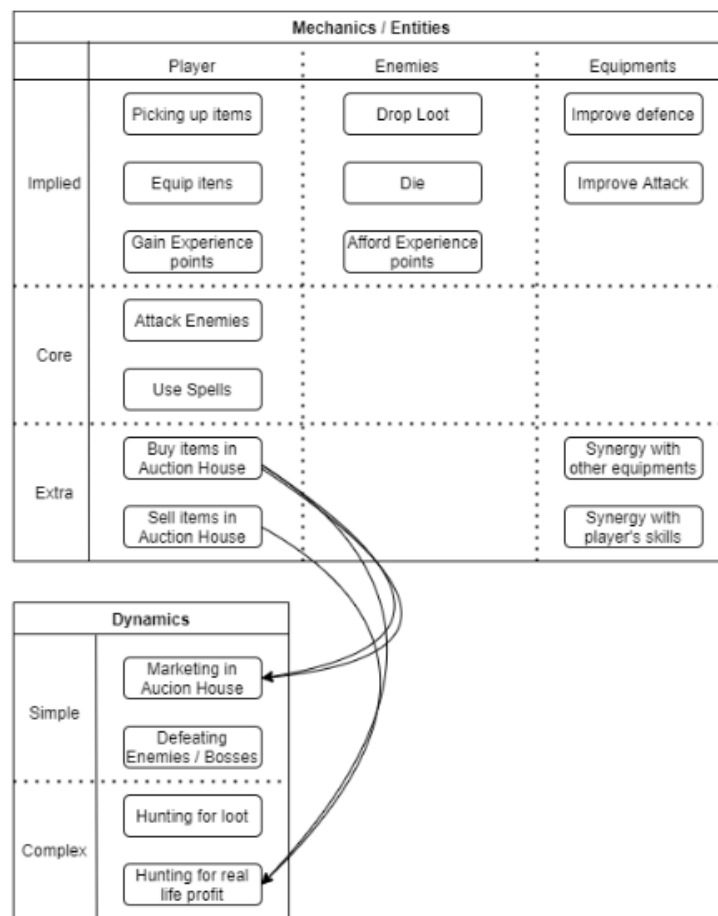
**Figure 3.** Diagram of the scenario translated to RMDA.



**Figure 4.** League of Legends game. Image taken from Reddit forum post by IoNJohn. https://www.reddit.com/r/leagueoflegends/comments/13jlxo/why_is_this_still_here/, accessed on 6 March 2020.

This dynamics turned out to create extremely unbalanced advantages to the players who did it, which hugely affected the challenge aesthetics of the game. In this case, the workaround for this was to take away this dynamic from the game by removing the mechanics that would invoke it: the avatars no longer have the mechanic of dodging, and

the items do not have the mechanics to improve dodging skills. As a result, the unwanted dynamic was successfully removed.

A great example of unbalanced aesthetics was given by Philip Tan [43] analysing a game called BioShock Infinite, by Irrational Games. In this game, the player is sent to a fantastic and beautiful world, with well detailed Non-Player Characters (NPC) living in it, with an engaging, carefully constructed narrative, supported by small details like posters, NPC' dialogue, and well-designed architecture, implying fantasy aesthetics. Another characteristic of this game is that the player is constantly engaged in combat (see Figure 5), being attacked and having to defend himself, which is part of the challenge aesthetics. The player has the mechanics of gathering food from the ground to heal and to pick money and loot from corpses to improve their equipment, enhancing their combat victory chance. This resulted in a dynamic where the player would keep looking for more food and items laying on the ground or in corpses and ignoring the beautiful modelled city and all the small details that tell the story of this fantastic world—and were carefully designed. Since the game team worked hard on bringing this fantasy aesthetics as a priority over challenge, this is not an optimal result for the player experience.



**Figure 5.** BioShock Infinite game. Image taken from: Video game Bioshock Infinite, posted by Pavan Shamdasani, 2013. https://www.scmp.com/lifestyle/technology/article/1207642/video-game-bioshock-infinite, accessed on 6 March 2020.

To keep the gameplay as it is, it is probably not a good choice to remove the dynamics of looting nor the combat dynamics, but they should be better balanced to make the best of the detailed 3D world and narrative of the game. Understanding RMDA would help developers to correctly map the dynamics that would invoke this unbalance in aesthetics and define which mechanics they should change/remove, or even create, to fix this issue. The dynamics of combat could be changed in a way that it can become easier, so that the player will need to rely less on items laying on the ground and absorb the narrative more. The map entity(ies) could be remade in order to separate combat areas from narrative/immersion areas. They could create new mechanics provided to "interest points" related to narrative, such as awards for finding posters or NPCs and unlocking new dialogues, to create an incentive to players to look for them.

The aim of RMDA is not to create rules of what is right or wrong in game design. Its purpose is to work as a guide, or a blueprint, that can help developers understand how each aspect of the game they can directly affect or change (Mechanics), how they work together with the players' input over a period of time (Dynamics) and the emotional response that the player achieves when interacting with the game (Aesthetics)—and by doing so, enhance the final quality of the product.

## 5. Conclusions

The difficulty of acceptance of a structured design methodology is a common attitude by designers that belongs to domains where creativity is at the core of the creation process—such as music, movies, literature and games. This fact, added to complex aspects of the design of games, can make it hard to create a design methodology. This paper tried to propose a structured methodology that is supported by a clear ontology that will not impair creativity and can in fact help it. By connecting the aesthetics objectives to all the abstraction layers of the game being developed, we hope to justify decisions of design—as subtle as camera mechanics in a terror game (e.g., shake and blur effects), or as big as changing the core mechanics of a game—to an aesthetics objective, or to an emotional response the player can invoke. By working as a guide to support designers in their creative process, they can correctly aim their creation towards the expected result of the product.

The redefinition of the framework MDA concepts has been presented as a way to facilitate its use when designing games. Thus, we call it RMDA, whose main purpose is to eliminate the main problems with MDA and make it useful for game designers. It is a difficult task to achieve a structured methodology in a domain that has so many specific characteristics and aspects that have to be supported by it. Understandably, game designers would not easily adopt one. We hope that this paper moves the domain a little closer to this objective and allows designers to enhance the quality of the development process and the amount of fun players get with it.

The MDA authors pointed out that the eight kinds of fun are a starting point towards a vocabulary to be used as a guide to understand the player's emotions. By revealing more about this subjective area of the game domain, designers will have a better understanding of the emotional objective of the game and hopefully increase its quality. A paper written by Roberto Dillon [44] enhanced the way that designers could work on the player's emotion. The author created what he calls The 6-11 framework, a methodology that could be used alongside MDA which focuses on six emotions and eleven instincts that are recurrent in psychology: Fear, Anger, Joy/Happiness, Pride, Sadness and Excitement, and the instincts are Survival (Fight of Flight), Self-Identification, Collecting, Greed, Protection/Care/Nurture, Aggressiveness, Revenge, Competition, Communication, Exploration/Curiosity and Colour Appreciation. The extra detailing about the expanded modelling of player emotions is a reasonable way to further increase the efficacy of game design methodologies, and future work is needed in this area of the domain.

Of course, the use of RMDA can and should be complemented with other methodologies, namely some that are more analysis-oriented, as Design Lenses or even others like Game Patterns that allow solving specific issues of the game design.

Currently, the first author uses this methodology in their work as a game designer, which facilitates game development and communication with the development team. However, in the future, we want to use the RMDA in the design of new games, for example, with master students of game design studies, and evaluate their experiences.

# References

1. Bjork, S.; Lundgren, S.; Holopainen, J. Game Design Patterns. In Proceedings of the 1st International Digital Games Research Conference, Utrecht, The Netherlands, 4–6 November 2003.

2. Callele, D.; Neufeld, E.; Schneider, K. Requirements engineering and the creative process in the video game industry. In Proceedings of the 13th IEEE International Conference on Requirements Engineering (RE'05), Paris, France, 29 August–2 September 2005; pp. 240–250. doi:10.1109/RE.2005.58.

3. Winget, M.; Sampson, W. Game development documentation and institutional collection development policy. In Proceedings of the ACM/IEEE Joint Conference on Digital Libraries, Ottawa, ON, Canada, 13–17 June 2011; pp. 29–38. doi:10.1145/1998076.1998083.

4. Dormans, J. Engineering Emergence: Applied Theory for Game Design. Ph.D. Thesis, Interfacultary Research Institutes, Faculty of Humanities (FGw), Amsterdam, The Netherlands, 2012.

5. Hunicke, R.; Leblanc, M.G.; Zubek, R. MDA: A Formal Approach to Game Design and Game Research. In Proceedings of the AAAI Workshop on Challenges in Game AI, San Jose, CA, USA, 25–29 July 2004.

6. Zagal, J.P.; Mateas, M.; Fernandez-Vara, C.; Hochhalter, B.; Lichti, N. The Game Ontology Project. Available online: https://www.gameontology.com (accessed on 4 March 2020).

7. Zagal, J.; Mateas, M.; Fernandez-Vara, C.; Hochhalter, B.; Lichti, N. Towards an Ontological Language for Game Analysis. In Proceedings of the Digital Games Research Conference, Vancouver, BC, Canada, 16–20 June 2005.

8. Falstein, N.; Bardwood, H. The 400 Project. In Proceedings of the Game Developers Conference, San Jose, CA, USA, 20–24 March 2006.

9. Zagal, J.; Bruckman, A. The game ontology project: Supporting learning while contributing authentically to game studies. In Proceedings of the Computer-Supported Collaborative Learning Conference, CSCL, Xi'an, China, 16–18 April 2008; pp. 499–506.

10. Ampatzoglou, A.; Chatzigeorgiou, A. Evaluation of object-oriented design patterns in game development. *Inf. Softw. Technol.* **2007**, *49*, 445–454. doi:10.1016/j.infsof.2006.07.003.

11. Ašeriškis, D.; Damaševičius, R. Gamification Patterns for Gamification Applications. *Procedia Comput. Sci.* **2014**, *39*, 83–90. doi:10.1016/j.procs.2014.11.013.

12. Ašeriškis, D.; Damaševičius, R. Gamification of a Project Management System. In Proceedings of the Seventh International Conference on Advances in Computer-Human Interactions (ACHI 2014), Barcelona, Spain, 23–27 March 2014.

13. Kelle, S.; Klemke, R.; Specht, M. Design Patterns for Learning Games. *Int. J. Technol. Enhanc. Learn.* **2011**, *3*. doi:10.1504/IJTEL.2011.045452.

14. Salen, K.; Zimmerman, E. *Rules of Play: Game Design Fundamentals*; The MIT Press: Cambridge, MA, USA, 2003.

15. Robson, K.; Plangger, K.; Kietzmann, J.H.; McCarthy, I.; Pitt, L. Is it all a game? Understanding the principles of gamification. *Bus. Horiz.* **2015**, *58*, 411–420. doi:10.1016/j.bushor.2015.03.006.

16. Kiili, K. Digital game-based learning: Towards an experiential gaming model. *Internet High. Educ.* **2005**, *8*, 13–24. doi:10.1016/j.iheduc.2004.12.001.

17. Csikszentmihalyi, M. *Flow: The Psychology of Optimal Experience*; Harper and Row: New York, NY, USA, 1990.

18. De Freitas, S.; Jarvis, S. A Framework for developing serious games to meet learner needs. In Proceedings of the Interservice/Industry Training, Simulation & Education Conference (I/ITSEC), Orlando, FL, USA, 4–7 December 2006.

19. Winn, B.M. Chapter LVIII The Design, Play, and Experience Framework. Handbook of research on effective electronic gaming in education. IGI Global: Hershey, PA, USA, 2007.

20. Schell, J. *The Art of Game Design: A Book of Lenses*; Morgan Kaufmann Publishers: Burlington, MA, USA, 2008.

21. Amory, A. Game object model version II: A theoretical framework for educational game development. *Educ. Technol. Res. Dev.* **2007**, *55*, 51–77. doi:10.1007/s11423-006-9001-x.

22. Yusoff, A.; Crowder, R.; Gilbert, L.; Wills, G. A Conceptual Framework for Serious Games. In Proceedings of the 2009 Ninth IEEE International Conference on Advanced Learning Technologies, Riga, Latvia, 15–17 July 2009; pp. 21–23. doi:10.1109/ICALT.2009.19.

23. Marne, B.; Wisdom, J.; Huynh-Kim-Bang, B.; Labat, J.M. The Six Facets of Serious Game Design: A Methodology Enhanced by Our Design Pattern Library. In *21st Century Learning for 21st Century Skills*; Ravenscroft, A., Lindstaedt, S., Kloos, C.D., Hernández-Leo, D., Eds.; Springer: Berlin/Heidelberg, Germany, 2012; pp. 208–221.

24. Barbosa, A.; Pereira, P.; Dias, J.; Silva, F. A New Methodology of Design and Development of Serious Games. *Int. J. Comput. Games Technol.* **2014**, *2014*, 817167. doi:10.1155/2014/817167.

25. Arnab, S.; Lim, T.; Carvalho, M.B.; Bellotti, F.; de Freitas, S.; Louchart, S.; Suttie, N.; Berta, R.; De Gloria, A. Mapping learning and game mechanics for serious games analysis. *Br. J. Educ. Technol.* **2015**, *46*, 391–411. doi:10.1111/bjet.12113.

26. Carvalho, M.B.; Bellotti, F.; Berta, R.; Gloria, A.D.; Sedano, C.I.; Hauge, J.B.; Hu, J.; Rauterberg, M. An activity theory-based model for serious games analysis and conceptual design. *Comput. Educ.* **2015**, *87*, 166 – 181.

27. Roungas, B. A Model-driven Framework for Educational Game Design. *Int. J. Serious Games* **2016**, *3*, 1–19. doi:0.17083/ijsg.v3i3.126.

28. De Lope, R.P.; Medina-Medina, N.; Soldado, R.M.; García, A.M.; Gutiérrez-Vela, F.L. Designing educational games: Key elements and methodological approach. In Proceedings of the 2017 9th International Conference on Virtual Worlds and Games for Serious Applications (VS-Games), Athens, Greece, 6–8 September 2017; pp. 63–70. doi:10.1109/VS-GAMES.2017.8055812.

29. Ávila Pesántez, D.; Rivera, L.A.; Alban, M.S. Approaches for Serious Game Design: A Systematic Literature Review. *ASEE Comput. Educ. J.* **2017**, *8*, 1–11.

30. Alexiou, A.; Schippers, M. Digital game elements, user experience and learning: A conceptual framework. *Educ. Inf. Technol.* **2018**, *23*, 2545–2567. doi:10.1007/s10639-018-9730-6.

31. Spyridon, B.; Refanidis, I. An adaptation and personalisation methodology for Serious Games design. In Proceedings of the European Conference on Games Based Learning, Odense, Denmark, 3–4 October 2019. doi:10.13140/RG.2.2.25266.45763.

32. Silva, F.G.M. Practical Methodology for the Design of Educational Serious Games. *Information* **2020**, *11*, 14. doi:10.3390/info11010014.

33. Viudes-Carbonell, S.J.; Gallego-Durán, F.J.; Llorens-Largo, F.; Molina-Carmona, R. Towards an Iterative Design for Serious Games. *Sustainability* **2021**, *13*, 3290. doi:10.3390/su13063290.

34. Walk, W.; Gorlich, D.; Barrett, M. Design, Dynamics, Experience (DDE): An Advancement of the MDA Framework for Game Design. In *Game Dynamics*; Springer: Cham, Switzerland, 2017; pp. 27–45. doi:10.1007/978-3-319-53088-8_3.

35. Sicart, M. Defining Game Mechanics. *Game Stud.* **2008**, *8*, 1–14.

36. Super Mario World, by Nintendo. 1990. Available online: https://www.nintendo.pt/Jogos/Super-Nintendo/Super-Mario-World-752133.html (accessed on 14 April 2020).

37. Pokemon Yellow, by Game Freak. 1999. Available online: https://www.nintendo.pt/Jogos/Game-Boy/Pokemon-Yellow-Version-Special-Pikachu-Edition-266142.html (accessed on 14 April 2020).

38. Nakamura, J.; Csikszentmihalyi, M. *The Concept of Flow*; Oxford University Press: Oxford, UK, 2002.

39. Lazzaro, N. Why We Play Games: Four Keys to More Emotion Without Story. In Proceedings of the Game Developer's Conference, San Jose, CA, USA, 24–25 March 2004; pp. 1–4.

40. LeBlanc, M. 8 Kinds of Fun. Available online: http://algorithmancy.8kindsoffun.com/ (accessed on 15 April 2020).

41. Elder Scrolls V Skyrim, by Bethesda Game Studio. 2011. Available online: https://elderscrolls.bethesda.net/en/skyrim (accessed on 14 April 2020).

42. Tibia, by Cipsoft. 1997. Available online: https://www.cipsoft.com/en/games/tibia (accessed on 15 April 2020).

43. Tan, P. MIT Game Lab—The Education Arcade. Available online: https://www.youtube.com/watch?v=pD74RdAMW8s (accessed on 5 May 2020).

44. Dillon, R. The 6-11 Framework: A New Methodology For Game Analysis And Design. In Proceedings of the Game-On Asia Conference Asia Conference, Singapore, 1–3 March 2011; pp. 25–29.