

Serious Mods: A Case for Modding in Serious Games Pedagogy

Victoria McArthur
ICCID
University of Toronto
Mississauga, Canada
victoria.mcarthur@utoronto.ca

Robert J. Teather
Dept. of Computing and Software
McMaster University
Hamilton, Canada
teather@mcmaster.ca

Abstract— In this paper, we present a case study for the use of modding as a pedagogical practice in a humanities-based game design course. In particular, this approach has been extremely beneficial as it allows students to sidestep technological barriers. In our case study, we show how two different mods of the same platformer game can allow students to engage with game design in order to explore the relationship between mechanics and meaningful play.

Keywords—*Game Design, Serious Games, Modding, Pedagogy*

I. INTRODUCTION

In this paper, we discuss the use of modding – or “modifying” – existing computer games as a pedagogical tool when exploring the design of serious games. Here, modding refers to the editing of an otherwise functional, pre-existing game. While the term “serious game” has come to encompass a broad variety of games, including but not limited to advergames, art games, and training simulations, we focus exclusively on persuasive games [1] and educational games [2] in this paper.

Game design courses offered by the humanities often differ substantially from game design courses offered by engineering schools and departments. Engineering-centric departments generally focus on technical aspects of game development. In contrast, the humanities must focus not only on the practical components of game design, but also on the significance of the game as a designed artifact [3]. To this end, we argue that such courses should encourage students to engage both with theory and praxis, the process of “practicing” theory [4]. Praxis is achieved by not only reading and writing about games, but through the simultaneous design and development of one.

In humanities academic streams, it is not uncommon for instructors to be faced with diverse classroom populations, in terms of both disciplinary background and technical capabilities. This increases the difficulty in targeting classroom material to the students: assuming that students have a highly technical background, for example, will leave behind those with limited technical skills. In order to address this limitation, we have introduced modding as a pedagogical tool. This allows students with limited technical capabilities to side-step some of the lower level aspects of game programming, while not “punishing” advanced students (i.e., those with more technical

background) with boring introductory-level content. This allows both classes of students to engage with the higher level forms of expression and representation in their course projects. We present a case study investigating two mods of the same platform game developed in two offerings of the same humanities game design course. Our case study helps to illustrate how modding can be used as an effective pedagogical exercise, allowing students in non-technical programs to study serious games by designing them.

II. RELATED WORK

There are numerous pedagogical benefits of modding existing games [5], rather than developing them from the ground up. The primary benefit is that modding allows students to side-step technical barriers to focus on more meaningful design activities. This way, they can engage more directly with game design concepts and content without getting bogged down in minute details that might otherwise stymie their efforts early on. The pedagogical benefits of modding have been explored in a variety of contexts, including history education [6, 7], as a collaborative exercise in the classroom [8], and to motivate young girls to learn IT concepts [9]. In this paper, we illustrate the above in the context of a humanities game design course, showing how two different mods of the same platformer game can allow students to engage with game design in order to explore the relationship between mechanics and meaningful play.

Games such as Firaxis Games’ *Civilization III* and Bioware’s *Neverwinter Nights* provide advanced editing tools, allowing users to create original content. For example, modders can create original game maps, new in-game objects, and new enemies and obstacles to challenge players. Squire demonstrated that through modding and subsequently playing *Civilization III*, students are able to better understand the processes that contribute to historical events, rather than conceiving of them as entirely narrative accounts [7]. Similar work by Moshirnia [6] indicates that students who played mods of *Civilization IV* in a controlled experiment were more prepared to attempt higher-order-thinking questions on exams than those who did not.

The game developer kit (GDK) for *Neverwinter Nights*, allows users to easily develop their own custom games using the *Neverwinter Nights* engine. The GDK has been used by

researchers to sidestep technical barriers and create their own educational or serious games [10].

III. CASE STUDY

We present a case study on the use of modding in the classroom to develop serious games. We discuss two different mods of the same game. The mods were generated by two different groups of students from two offerings of a humanities game design course with entirely different outcomes. The first is a persuasive game – a game which conveys an argument through its rules and mechanics [1]. The second is an educational game [2] and was developed with the intent to teach players about the management of diabetes.

Beyond typical game design issues (e.g., interface, controls, etc.), students designing serious games must also consider how the “serious” elements of their game are expressed. For example, students designing a persuasive game must be mindful of how alternative play styles can yield alternative understandings of their game. Treanor, for example, cautions that “emergent effects of the procedural rule system can and often do undermine its intended meaning” [11]. Additionally, students designing educational games must ensure that gameplay is not overshadowed by the game’s educational content [12].

We note that the modding activities described in this paper are specific to Yoyo Games’ *Game Maker* engine¹. As such, prior to presenting the actual case study, we will briefly discuss choosing a game engine for students of mixed or low programming capabilities.

A. Choosing a Game Engine

There are many free game engines available, each with different strengths and weaknesses. The engine we have used in multiple offerings of this course is *Game Maker*. There are numerous advantages to using *Game Maker* when teaching students with limited technical skills. For example, *Game Maker* works on both Mac and PC platforms and has an online community that hosts numerous tutorials and help articles. The developers of *Game Maker* also have published *The Game Maker’s Apprentice* [13], which includes detailed instructions for creating 10 different (types of) games. The greatest advantage of *Game Maker* is that it is easy enough to use that it allows students to create basic games without having to learn complex programming languages. However, for students who wish to have a greater degree of control, it also supports scripting using the specialized Game Maker Language (GML) – a proprietary full-featured object-oriented programming language included with the engine. Finally, *Game Maker* also provides the ability to compile games to an executable file. This enables easy sharing of completed projects with others, for example, by hosting games in online portfolios.

Fig. 1 illustrates *Game Maker*’s WYSIWYG (what you see is what you get) tools. WYSIWYG interfaces, based on the idea of direct manipulation [14], have long been known to offer enhanced productivity for novice users [15]. Such interfaces

replace commands with actions, while allowing novices to immediately understand the outcome of their actions.



Fig. 1. Modding the platform game using *Game Maker*’s WYSIWYG tools.

In our experience, students with little to no programming experience have found the engine to be easy to use. However, some felt it was best suited to the development of specific kinds of games (the kinds presented in *The Game Maker’s Apprentice*). Genres such as platformer or maze games are comparatively easy to create. Other styles, such as role-playing games (RPGs) are considerably harder to develop with *Game Maker*. Moreover, while it is easy to create simple games following the tutorials, some students felt that they were not skilled enough to implement the sophisticated mechanics they had envisioned when first proposing their serious game prior to developing it.

Other platforms that were explored in later offerings include *Game Salad*, *RPG Maker*, *Adobe Flash*, and *Unity*. While students found that *Game Salad* was easier to learn than *Game Maker*, they discovered that projects could not be easily exported for sharing (and later submitting to the instructor). *Flash* and *Unity*, while more powerful than *Game Maker*, have much steeper learning curves, and can take considerably longer to get a project off the ground. We thus conclude that for instructors interested in exploring procedural rhetoric through game design, *Game Maker* is the easiest engine to learn, teach, and use for students without a strong technical background. Furthermore, the tutorials offered in the aforementioned text can be easily modded by students.

B. Modding as Praxis

Since technical ability can vary greatly between students in humanities programs, we felt it was important to offer a “modding” option in our game design courses. The idea of modding is not dissimilar to the pedagogical concept of constructionism, or “learning by building” [16]. The pedagogical benefits of modding have been presented by others [3, 5] as a way to allow students to side-step game development barriers in order access meaningful design activities. This practice may also be particularly beneficial when working with younger students.

After working through a number of tutorials, students choose a game and modify it in an attempt to create a new serious game. This usually involves two fundamental steps: re-

¹ Available from <http://www.yoyogames.com/studio>

skinning objects in the game with new sprites, and modifying the game's mechanics. Bogost cautions that re-skinning alone is a problematic way to explore procedural rhetoric, since, at best we are playing with "graphical logics" [1, pp. 89]. In other words, when exploring procedural rhetoric, students must move beyond purely re-skinning the game and make adjustments to other mechanics (e.g., scoring, level design, etc.) in order to express their argument. A change of imagery alone would, in most cases simply be an exploration of visual rhetoric. Only through modding of game mechanics can students actively explore the concept of procedural rhetoric. Once the mechanics are in place, the graphical aspects of the game can easily be redesigned to suit the theme of the new game.

C. Modding the Platformer

We first describe the original platformer game that was subsequently modified by students into serious games. The original platformer is shown in Fig. 2. Detailed tutorials for the game, along with incremental builds (versions of the game at various stages of development) are hosted on the *Game Maker* website². Like many platformer games, this one tasks the player with moving a character through a 2D environment while jumping over and onto platforms, shooting and avoiding enemies, and attempting to collect items scattered throughout the environment.



Fig. 2. Original version (unmodified) of the platformer tutorial game, developed using Game Maker.

The following sub-sections discuss the specific student mods produced using the platformer game described above.

1) Equality Game

In the first mod, a female athlete wished to explore gender inequality in sports. As a woman, she was ever aware of the lack of opportunities and unequal pay between the sexes in the athletic profession. She had formed a thesis for her persuasive game, but found it difficult to design an original game to present her thesis. As a pedagogical experiment, she was invited to try modding a game. Using the platformer game described above as a starting point, the student began

modifying the graphics, levels, and mechanics of the game. The resulting modified game is shown in Fig. 3.

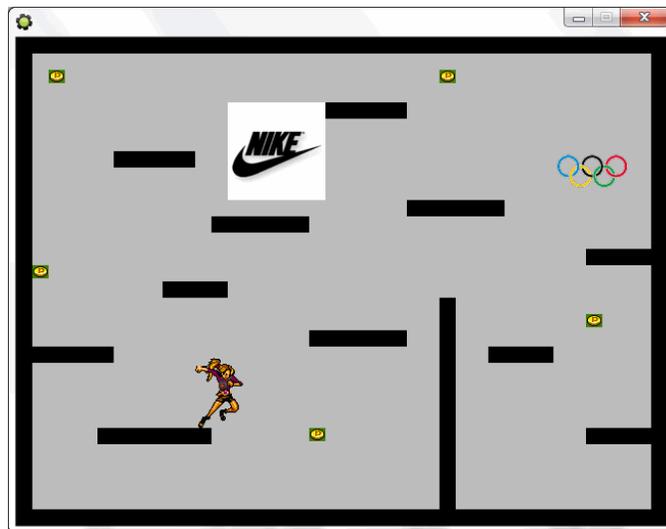


Fig. 3. Equality Game - a game exploring inequality between the sexes in athletics.

In the modified game, the player first plays as a female character. The player jumps from platform to platform, attempting to collect each of the athletic opportunities (money, endorsements, etc. represented as company icons, e.g., the Nike logo shown in Fig. 3). Next, the player plays as a male character, attempting to collect all of the same objects using the same interaction techniques. As the female, the character cannot reach all of the objects as easily and they are worth fewer points. As the male, there are more objects to collect, they are easier to collect, and they are worth more points than they are when the female sprite collects the exact same objects. In this fashion, the game presents the argument that female athletes are disadvantaged relative to their male peers.

The premise of this mod is quite simple, and gameplay doesn't take very long (both levels are quite short), however, through gameplay mechanics, the student is able to invite others to consider the ways in which gender has such a tremendous impact on success and opportunity within the world of athletics. Regardless of the player's ambition, when they play as the female avatar, career opportunities are few and they are undervalued. The player's ability to experience the differences between the male and female avatars in quick succession further makes this argument more accessible.

2) Diabetes Game

Another student wanted to design a serious game to teach people how diabetics must manage their blood sugar levels throughout the day. Using the same platformer game, the student modified it into a game in which a young diabetic child must maintain appropriate blood glucose levels while progressing through various stages of their day. See Fig. 4. Each level of the game contains a several collecting items, including food and powerups representing insulin. Players must make mindful choices about the foods they collect such that they maintain a healthy blood sugar level. Collecting food items increases the blood sugar level, while insulin decreases it. The player must not go above or below a healthy blood

² Available at <http://sandbox.yoyogames.com/make/tutorials>

sugar level. Game levels cannot be passed without consuming *some* foods – else the player character becomes weak and their sugar levels plummet. Additionally, players must be careful to only collect as many insulin powerups as needed, otherwise they may over- or under-correct for their desired blood glucose level.



Fig. 4. Diabetes Game – a game designed to teach players about blood glucose management.

Again, the premise of the modified game is quite simple, but through a little research the student was able to help players learn some lesser-known facts of blood sugar management. For instance, it is obvious to most people that sugary treats can cause blood sugar to raise, but fewer people may realize that carbohydrates are broken down into sugar when digested. This information is shared not through text, but through the player's interactions with various food objects. If the player character consumes milk, they will notice a small spike in their glucose levels. This spike can be countered with an insulin powerup. Other foods, such as broccoli, carrots, and peppers do not cause a noticeable spike in the player character's blood sugar levels, but contribute to their food energy, helping them reach the end of the level.

In the early levels, a variety of foods are generously littered throughout the level, allowing the player to play around more with food choices and see how these affect their blood glucose levels. In later levels, food choices are spaced out more into healthy and unhealthy groupings. If the player does not consume enough healthy food when available, they may not be able to keep the two variables (energy and blood sugar) balanced enough to complete the level. In this case, they must start the level over and be more mindful of the choices they make.

IV. DISCUSSION AND CONCLUSIONS

Both examples are very simple modifications and both provided rich writing opportunities for the students. Many students in the course found the idea of creating a game from

scratch to be overwhelming. Modding afforded these students the same academic opportunity: to engage theory in a meaningful way.

The mods presented in this paper were two mods of the same game: a platformer tutorial hosted on the *Game Maker* website. Similar tutorials are offered for shooting games, maze games, and puzzle games. These other genres are also very easily modded by students.

The beauty of modding as a pedagogical practice is not only its accessibility, but also that it can be made to fit the instructional needs of the course. Modding can be introduced as a collaborative practice or as an individual assignment. We have also found that modding can be highly scalable, providing rich opportunities for reflection as an in-class activity or a semester-long project.

REFERENCES

- [1] I. Bogost, *Persuasive games: the expressive power of videogames*. Cambridge, Mass.: MIT Press, 2010.
- [2] D. Charsky, "From edutainment to serious games: a change in the use of game characteristics," *Games and Culture*, vol. 5, pp. 177 - 198, 2010.
- [3] R. J. Teather and V. McArthur, "Teaching User Interface Evaluation Methods with Games," *Proceedings of the IEEE Consumer Electronics Society Conference on Games, Entertainment, and Media - GEM 2014*, pp. 204-207.
- [4] D. W. Shaffer, "Pedagogical praxis: The professions as models for post-industrial education," *Teachers College Record*, vol. 10, pp. 1401-1421, 2004.
- [5] M. S. El-Nasr and B. K. Smith, "Learning through game modding," *Computers in Entertainment (CIE)*, vol. 4, p. 7, 2006.
- [6] A. Moshirnia, "The educational potential of modified video games," *Issues in informing science and information technology*, vol. 4, pp. 511-521, 2007.
- [7] K. Squire, "Replaying history: Learning world history through playing Civilization III," *Indiana University, Indianapolis, IN*, 2004.
- [8] S. George, E. Lavoué, and B. Monterrat, "An Environment to Support Collaborative Learning by Modding," in *Scaling up Learning for Sustained Impact*, ed: Springer, 2013, pp. 111-124.
- [9] I. Yucel, J. Zupko, and M. Seif El-Nasr, "IT education, girls and game modding," *Interactive Technology and Smart Education*, vol. 3, pp. 143-156, 2006.
- [10] C. S. Loh and J. H. Byun, "Modding Neverwinter Nights into serious games," *Digital Simulations for Improving Education: Learning Through Artificial Teaching Environments: Learning Through Artificial Teaching Environments*, 2009.
- [11] M. Treanor and M. Mateas, "Newsgames: Procedural rhetoric meets political cartoons," *Proceedings of DiGRA 2009*.
- [12] J. P. Gee, *What video games have to teach us about learning and literacy*. New York, NY: Palgrave MacMillan, 2003.
- [13] J. Habgood and M. Overmars, *The Game Maker's Apprentice*: Springer, 2006.
- [14] B. Shneiderman, "Direct manipulation: A step beyond programming languages," in *ACM SIGSOC Bulletin*, 1981, p. 143.
- [15] B. A. Myers, "A brief history of human-computer interaction technology," *Interactions*, vol. 5, pp. 44-54, 1998.
- [16] I. E. Harel and S. E. Papert, *Constructionism*: Ablex Publishing, 1991.