Lessons Learned from the STEAM Project: Development and Deployment of Digital Learning Games for Effective Learning Improvement

William Young II1, Teresa Franklin2, Chang Liu3 and David Chelberg3

1Department of Industrial and Systems Engineering, Russ College of Engineering and Technology, 270 Stocker Center, Ohio University, Athens, OH 45701-2979, USA
2Department of Educational Studies, The Gladys W. and David H. Patton College of Education and Human Services, Ohio University, 315 McCracken Hall, Athens, OH 45701-2979, USA
3School of Electrical Engineering and Computer Science, Russ College of Engineering and Technology, Ohio University, 329 Stocker Center, Athens, OH 45701-2979, USA

This paper presents the experiences of engineering graduate students that developed and deployed digital learning games for middle school science classrooms in Appalachian Ohio. The STEAM project (Science and Technology Enrichment for Appalachian Middle-schoolers) serves as a background for the integration of digital games containing much-needed standards-based science content in the classrooms. Learning technology development of content-rich 3D learning games is one major topic of this paper. The concerns of using STEAMIE, an 'in-house' 3D software; Second Life, a partially open source software system; and Flash, a commercial software tool are presented to share our lessons learned with other technologists seeking to develop digital learning games with similar software technologies. Post-development and deployment activities and a sample of the initial results obtained from selected learning games are presented to illustrate the learning effectiveness of STEAM's learning games and program.

Keywords: interactive learning environments; teaching/learning strategies; virtual reality; simulations

1.Introduction

Children spend a significant time playing games outside of the classroom setting. However, the use of games as an educational medium within the K-12 school setting is limited [1]. To provide digital natives the digital content that they desire, educators must use progressive means within the areas of computer science, engineering, and instructional technology to integrate digital content within our nation’s schools. In particular, educational games and simulations can engage students in higher-order thinking, which is useful when learning difficult to teach concepts.

This paper provides a comprehensive summary of a National Science Foundation (NSF) GK-12 (Graduate teaching Fellows in K-12 education) project called STEAM (Science and Technology Enrichment for Appalachian Middle-schoolers). The primary focus of this experience report is to summarize the lessons learned by the project's team of graduate students, or Fellows, in developing and deploying game-based learning (GBL) technologies in middle schools of Appalachia. After a review of related work is presented, the article summarizes the STEAM project, as well as the games that were developed by the graduate students, which are all freely accessible to the public (http://vital.cs.ohiou.edu). In addition to this discussion, this paper quantitatively summarizes a few select games and their educational benefits.

There are three goals of this paper, which include 1) informing others about the STEAM project and freely accessible games, 2) sharing the insights learned from the Fellows in developing and deploying GBL technologies so that others may benefit from their experience, and 3) presenting quantitative evidence of a few selected STEAM games and a discussion of their educational benefits.

2. Digital Games in Education

The video game industry has grown rapidly over the last decade. In 2010, the revenue generated in the video game industry in the United States reached the $20 billion mark [2], which was twice that of the movie industry [3]. A recent large-scale investigation found that 97% of teens 12 to 17 play games regardless of gender, age, or socioeconomic status [4]. Cummings and Vandewater [5] found that 36% of adolescents play video games for around one hour every weekday and one and a half hours every weekend. The National Endowment for the Arts found that 15 to 24 year-olds spent only seven minutes a weekday reading, while spending over an hour of watching TV daily [6]. With these statistics, one can conclude that today's youth is more accustomed to receiving information in a digital format instead of printed material.

With the widespread use of information being presented in the digital form, one could assume that it would be advantageous for educators to use this form of medium in their classrooms. Prensky [7] states “today’s students think and process information fundamentally differently from their predecessors” and that GBL aids are needed for “even the
most serious content.” Though GBL, there are several examples of its educational benefit. For example, over four-hundred schools participated in playing educational games designed for the Sony PlayStation® [8]. The studies found that games attributed to a 24% increase in vocabulary skills, a 25% increase in language arts, a 51% increase in math problem solving skills, and a 30% increase in math algorithm scores. Educational games have also been developed for other console-based technologies such as the Super Nintendo® [9]. Packy and Marlon demonstrated that playing such a video game not only improves a child’s self-efficacy, but can also improve outcomes, communication skills, and behaviors about one’s health. GBL technologies have benefited students with severe reading problems [8].

The academic debates that are opposed to GBL in the classroom have diminished over time [10]. More recent trends in GBL issues study the cost of development, easy of deployment, and the need to assess and learn what attributes to the educational benefits that are reaped [11]. Simulating or building educational modules to teach complex or costly subject matter are just some of the reasons why GBL is significant to the research community. It has been demonstrated that today’s students are adapted to new learning techniques and have learned parallel processing skills, quick task-switching abilities, change in reasoning from the abstract to the concrete, and information organization skills while playing video games, which advocate the need for a change in traditional lesson plans [12].

It is undeniable that today’s children are growing up in a radically different environment than just a decade ago. These new influences have ushered in a new way of absorbing information for children and educational games are an effective method of reaching these future leaders of the world. However, “many skills that new technologies have actually enhanced … are almost totally ignored by educators [8].” This further demonstrates the importance of projects like STEAM, which are able to overcome development and deployment issues surrounding the use of GBL technologies in the classroom, as a significant field of research.

3. STEAM Overview

The National Science Foundation's Graduate Teaching Fellows in K-12 Education, or GK-12 program, “provides funding for graduate students in NSF-supported science, technology, engineering, and mathematics (STEM) disciplines to acquire additional skills that will broadly prepare them for professional and scientific careers in the 21st century. Through interactions with teachers and students in K-12 schools and with other graduate Fellows and faculty from STEM disciplines, graduate students can improve communication, teaching, collaboration, and team building skills while enriching STEM learning and instruction in K-12 schools [13].”

Seamlessly integrating technology in the classroom can be difficult [14]. However, through integration, technology can serve as a great medium to teach difficult scientific content to students. STEAM was a multi-year project (i.e. 2006 to 2010) that sought to provide classroom instructional content in digital form to teach content that is often seen as ‘difficult to teach’ and ‘difficult to learn’ by middle school science teachers [1]. STEAM consisted of a diverse set of engineering graduate students, or Fellows, from Ohio University. The project had as one of its primary goals to increase middle-schoolers content knowledge found in the Ohio’s Academic Content Standard for Science [15].

3.1 Appalachian Schools

The Appalachian school districts of rural Ohio are unique to others in the state because most were established during the coal mining years and are situated in former coal mining communities. Most students in these schools can trace their ancestry back to coal miners or have parents still working in the coal mining industry. The expectations for high school graduation or even college attendance have historically been low. Most schools in the region have a graduation rate of less than 80% [16], which is less than the national average. Appalachian schools also have large portions of their students enrolled on free or reduced lunch program. For example, 33% to 70% of the students participating on the STEAM project were a part of the free or reduced lunch program. Standardized testing scores are also among the lowest in the state with science achievement scores ranging from 42% to 72%, which are all below the state established minimum score of 85% [17].

Appalachian schools have remained underfunded even with the establishment of a Supreme Court Order in 2002 that declared Ohio’s school funding unconstitutional [18]. Taking the lack of school funding into account and the isolation of many of these communities, it is understandable that the technology within these schools is limited. For example, some of the schools participating on the STEAM project in 2006 were still running PCs with Windows 95® with no immediate plans or resources to upgrade. Due to the STEAM grant, schools with older labs equipment were upgraded to machines that were suitable for Windows XP®. In terms of Internet access, all of the participating schools were required to have access to E-Rate funding. However, the speed of the access varied from school to school and in most cases, the available bandwidth was not sufficient to support broadband applications.

3.2 Development and Deployment Overview

During four years of the STEAM project, many challenges were overcome in developing successful digital games for the middle school classrooms. One design principal in all of STEAM's games was that they all had to focus on Ohio’s
Content Standards for Science. With the Fellows being paired with a middle-school teacher, they had to identify a way to integrate game play with the educational content that teachers demanded.

The development of the digital games started each academic year with a week of Professional Development and Training at the start of summer. In these sessions, the entire project team of teachers, Fellows, and the Directors came together and engaged in conversations related to examining and establishing goals for the development and deployment of the games. During this initial training session, the following goals were identified and continuously improved or revised:

- An orientation for the Fellows and middle school teachers
- A review of all previous digital games for technical accuracy and completeness
- Discussion of new topics for digital games
- Development of lesson plans for existing digital games for curriculum implementation
- Technology training for teachers to integrate the digital games in their classrooms
- Development and refinement of pre-tests and post-tests
- Establishment of the research design for the upcoming year

Fellows had varying degrees of technical expertise and were encouraged to work in multi-disciplinarian teams. Through grant funding, each Fellow was provided a PC notebook for their development needs. Server space, repositories, wiki web sites were all utilized for development efforts. Through these means, Fellows and administration could share documents, organize working schedules, supply weekly reports, and follow software release protocols. Software was also purchased for the project such licenses included popular productivity, developing environments, and creativity and design tools.

Each Fellow was asked to work full time on game development during the summer months of the academic year. Once school started, Fellows were asked to spend 1 to 2 full days per week (10 hours) at the middle school. The remainder of their time was spent refining games. In cases where games where not deployed in the middle school, Fellows were asked to supplement classroom activities, lead discussions, and develop science based lessons plans. In total, there were eight participating middle school and eight Fellows, which provided many opportunities for Fellows to demonstrate the collection of games that were developed. Early deployment sessions allowed students to have a voice in how the games were created, which often would result in a discussion about how games are created and the educational background that is needed. It provided teachers an opportunity to provide feedback about the game's design and educational accurateness. Aside from early demos at the middle school, several other opportunities were exercised to obtain valuable feedback to enhance the games. Fellows engaged in peer-review sessions called ‘Game Nights.’ During these sessions, the community was invited to listen to the Fellows give presentations and demonstrations of the digital games they were creating. These sessions gave middle-school students the opportunity to visit the campus with their parents. Fellows were directly involved with educational workshops that were designed to educate teachers on how to deploy games in the classroom.

Once games were ‘testable’, Fellows were asked to apply a ‘research protocol’, which was mixed-methods, quasi-experimental approach based on convenient samples. The goal of this approach was to collect data for quantitative and qualitative analysis. In brief, students were randomly assigned to one of two groups and given a pre-test. A strategy was developed that allowed all students to experience the game being deployed. Some students were given the treatment of the game being deployed as well as the teacher’s instruction, while the control group was only provided the teacher’s instruction. Later, students not in the treatment group were provided the opportunity to experience the game. A comparison between groups could then be made between a treatment and control group using a post-test. More about the testing protocol will be addressed in a later section.

4. Technology Overview

STEAM developed educational modules using three primary technologies: Adobe Flash, Linden Lab’s Second Life, and Ohio University’s STEAMiE game engine [19]. The following sections provide an overview of the technologies used by STEAM.

4.1 Adobe Flash

Flash games have the advantage of working on nearly all computers within the school environment. Games developed using this technology are robust with respect to changing technologies, and are in general, easier to play due to some of their design limitations. Flash has been used to significantly improve testing scores [1], but the team decided that it was not an ideal technology to develop enriched virtual world environments for the designs identified by the teacher team. This is significant because middle school students were much more interested in playing the 3D games than the 2D games, which were developed with Flash technology. This preference of playing more immersive games is supported more recently by other educational researchers [20]. At the time, Flash could not provide true support for 3D game design, a social networking environment, and other more complex gaming support.
4.2 Linden Lab’s Second Life

Second Life (SL) is a 3D virtual world complete with social interaction, scripting, 3D building, camera and movement controls, and a helpful community for instruction and development. A typical SL game allows students to control an avatar, which is a virtual representation of their controller that features customizable options to improve game-play immersion [21]. In addition to SL’s immersive environment, which is an ideal setting to increase students’ motivation to play educational games [22], SL has a vast collection of building and scripting support using Linden Lab's Scripting Language. In addition, the programming syntax is similar to the C programming language [23], which allows relatively inexperienced programmers to develop GBL aids that address topics that are only possible within a virtual world environment [24].

SL works with high-end computer and networking systems. In the schools where such technology was available, the STEAM project chose to explore this more immersive environment. The lack of control and ability to use SL in the low-end computing environments in some rural schools increased the need for STEAM to investigate other options for game development.

4.3 Ohio University’s STEAMiE

In an attempt to incorporate all of the advantages of Flash and SL and to overcome their disadvantages, STEAM Fellows developed an in-house game engine called STEAMiE [19]. STEAMiE allows developers to have complete control for software development and emphasizes flexibility and modularity from the developers’ perspective. STEAMiE was programmed using C++ and is designed to be built and/or deployed on a wide variety of computers and operating systems, including Apple's Macintosh, Microsoft's Windows, and the Linux operating system. STEAMiE utilizes the OpenGL standard for graphics, together with several other open source libraries to implement sound, images, networking, avatars, etc.

Customized game engines allow developers to have complete control of their games. Internet communication and social interaction can be included in a game, which is beneficial for students working together as they explore a virtual environment. Some teachers do not always desire this feature and view it as a distraction. If a game does not lend itself to social interaction or if the school playing the game does not want their students’ actions going over the Internet, it is possible for all Internet communication to be removed from the game with the full control that a customized game engine can offer.

5. STEAM’s Digital Games

During the four years of funding, the STEAM project produced nearly 30 educational games, which are all freely available on the project's website (http://vital.cs.ohiou.edu). The purpose of this section is to summarize some of the games that were more popular with teachers and students. Thus, a brief description of these games is presented below.

**Furry Family**, shown in Figure 1, is a life science Flash game. This game was designed to address the hard-to-teach concepts of sexual reproduction using Punnett squares. In this game, students follow a storyline while they are asked to create a family of bunnies with certain traits when they are given information about their parent’s phenotypes and genotypes.

**Fruit Fly Genetics**, shown in Figure 2, is also a life science Flash game, where students observe the process of fruit flies passing on their traits to their offspring. Students choose the colors for their fruit fly's body and eye colors and the game simulates the breeding process and shows the created offspring, which helps to reinforce issues related to heredity.

**Drift**, shown in Figure 3, is an Earth science Flash game that teaches the concept of continental drift by allowing players to examine fossil, rock, and plant clues to determine how the continents once fit together. Players must then use convection currents, which is the primary factor for plate tectonic movement, in order to continents back to their original locations.

![Figure 1: Furry Family.](image1)

![Figure 2: Fruit Fly Genetics.](image2)

![Figure 3: Drift.](image3)
Weather Challenge, shown in Figure 4, is an Earth science SL module where students control avatars to fly around and hunt for raindrops in different cloud types. As they find the rain drops they are asked questions to identify the cloud type and what conditions are needed to produced a particular weather event as the students follow along a fun storyline [25].

Rafting Adventures, shown in Figure 5, is another earth science game in SL. Students use an avatar to ride a raft down a virtual mountain to watch simulations and learn how agents of erosion and deposition work, which reinforces how landforms are created.

The Redi Experiment, shown in Figure 6, is a scientific methods game in SL that reproduces Francesco Redi’s experiment that disproves spontaneous generation by experimenting with virtual flies, maggot larva, and meat separated into different containers. Students step through the experiment by interacting with a HUD (heads up display) and are asked questions related to the scientific method.

Mystery School, shown in Figure 7, is another scientific methods game in SL that students control avatars to walk around a virtual schoolhouse. Along the way, students find clues and answer questions based on if the information is an observation or an inference.

Energy Park, shown in Figure 8, is a physical science SL module that represents a virtual amusement park where students can ride a roller coaster, a pirate ship, or a terror tower in order to learn concepts about the transformation of potential and kinetic energy.

The Great Invertebrate Race, shown in Figure 9, is a life science SL module where students race cars around a track and learn about invertebrates with interactive screens. During the race, students collect information about various invertebrates, which they must use the information to correctly identify characteristics about invertebrates.

Mystery Minerals, shown in Figure 10, is a STEAMiE module intended to serve as a replacement for a physical geology lab. The goal of the game is to determine various minerals by using different geology tools. Students create a digital notebook filled with testing results when they apply the various tools and use the Dichotomous Key and Mohs hardness scale [26].

Wave Hero, shown in Figure 11, is a STEAMiE module that teaches concepts about the propagation of energy through a wave. In particular, Wave Hero familiarizes students with key concepts such as transverse and compression waves, wavelength determination, velocity, frequency, resonance, and other particle properties. These concepts are reinforced using visual demonstrations where students see how key parameters affect the wave and its medium [27].

Career Exploration, shown in Figure 12, is a STEAMiE module that teaches students about science careers. Students explore a 3D immersive world that contains obstacles that must be completed by a virtual vehicle to unlock biographies that contain multimedia that can be accessed that include descriptions of an science projects and career advice [28].
The Question Set Editor, shown in Figure 13, is a Java application that was developed to allow users to design and edit their own stand-alone question sets for use with the STEAMiE educational games. This capability allows teachers to control the content featured in the games to better suit their educational needs.

Stunt Park, shown in Figure 14, is a STEAMiE game that was designed to be a test-reviewing tool that features a challenging obstacle course. Whenever students complete stunts, they collect points by correctly answering questions that are created from the Question Set Editor. Teachers can easily create a question set and use the game as a fun and effective review tool for upcoming exams or standardized tests [26].

Space Racer, shown in Figure 15, is a multiplayer STEAMiE game that is also configured to work with the Question Set Editor. Space Racer, which can also be used as a reviewing tool, presents questions and instantly provides feedback to the teachers [26].

6. Lessons Learned from Development

Developing education games is a difficult task because developers are bound by technologies and environmental constraints. The next section describes development certain aspects of game developed based on the Fellows prospective of game development.

61. General Issues with Development

In designing an educational game, it must be fun for the students, while meeting educational standards required by the teachers. These are competing goals, which lead to conflicting game design decisions. The teachers believed that educational digital games are a novel way to explain hard to teach concepts. In comparison, the students often focused on the entertainment value of the games. Students were asked to provide qualitative feedback about the game, which helped the Fellows create products that were attractive to the students. Students' overwhelmingly commented and requested features that would allow them to control the look of the characters or vehicles they controlled in the played a game. Teachers were far less concerned about these options and more concerned about content presented in the games.

From a design standpoint, the logic acquired from students correctly or incorrectly answering questions is a convenient way to drive game-play logic. It is not necessarily required. Career Exploration did not force students to answer questions to progress through the game. Instead, a worksheet was created, which was intended to allow students to explore the virtual world for information related to the questions that were asked on the worksheet. In contrast, Furry Family required students to answer questions correctly in order to progress through the game. Both games were very successfully in the eyes of students and teachers; however, they were designed with different strategy in terms of logical control. Teachers consistently stated that they had no preference on whether worksheets were used to collect
information about the students' understanding, or information was collected in the game itself. Students had very strong feelings that their work should be captured ‘in-world’ where they would not have the leave the immersive world and use writing utensils or calculators.

As a result of Professional Development sessions, teachers indicated that they wanted additional options for reviewing material for exams or standardized tests. Teachers did not often have the technical skills to modify educational content that appeared in games without the aid of a Fellow. An effort was made to provide an easy-to-use system for teachers to develop question sets on their own. In conjunction with the Question Set Editor, games like Space Racer and Stunt Park allowed teachers to create and utilize created question sets. These games were designed intelligently where question sets could vary in the number and types of questions that teachers wanted to create. These types of games have a limitation in the fact that they are ‘generic’ by design. Space Racer and Stunt Park served their purpose by allowing teachers to review material, which was fun for the students.

Games that rely on Internet connectivity are often difficult to deploy due to the unreliability of service. The power of the Internet should not be avoided. Games that were designed to work with the Question Set Editor were designed with flexibility that should be considered for future game development. Games accessed the server and made the various questions sets available when Internet access was available. Question sets could be saved and accessed from local resource. Educational games that are intended to be deployed in the classroom should consider this type of flexibility.

A common feature missing within educational games design is a meaningful feedback system. If students are missing common questions in a game, a reporting system should capture this particular problem so that the teacher can address the issue in class. Feedback should be provided with every question a student answers incorrectly. The feedback system should restate the question and inform the student why their answer was incorrect.

Educational games should be designed with students of varying ability in mind. Though students admit to playing games of some variety frequently, not all students play all game genres. In public school settings, it is common to have a mixture of students with various abilities and deficiencies. Wave Hero was designed with difficulty settings, which allowed users to control game play, Career Exploration consisted of timers that allow students to access the educational content that appeared in games without the aid of a Fellow. An effort was made to provide an easy-to-use system for teachers to develop question sets on their own. In conjunction with the Question Set Editor, games like

6.2 Issues with Technologies

Developing games with commercialized system, such as Flash, has the advantage of reference materials being widely available. SL shares this advantage because Linden Lab’s provides an existing library for its 3D virtual world environment. Due to the extensive online reference material, virtual world objects or even scripts can be purchased from many online sites, which expedite the game development process and results in better looking, more professional solutions. However, since commercial engines are constantly evolving, maintaining games can be difficult, especially in the case where physics calculations are required. Games like the Great Invertebrate Race utilized SL’s physics engine. This can cause problems if updates were enforced. Slight changes to the physics engine would cause a once drivable car, to float uncontrollably in the air, rendering the game unusable. Fellows constantly had to check their games, when SL released and required users to update their software to a new version even if nothing changed within their code.

Certain features of SL hindered game development. The built-in scripting language was often not powerful enough to handle the needed physics of the game. The power of a customized game engine, like STEAMIE, could be too powerful for inexperienced programmers. Games like Career Exploration contained hundreds, if not thousands, of physics-enabled objects, which were continuously monitored for collision detection. Though an experienced programmer could easily develop a system of disabling and enabling relevant objects for collision detection, an inexperienced programmer would wonder why the frame rate performance was hideously slow and unplayable with older computer technology. A fundamental knowledge of game development is needed to develop enjoyable games for students to play.

7. Lessons Learned from Deployment

Bitner and Bitner [14] outline eight keys of success in terms of integrating technology in the classroom. A summary of these keys are provided below:

1. Teachers are likely to have anxiety about integrating technology into their classroom because it would require making changes to their lesson plans and would require learning new technology skills
2. Teachers must have a rudimentary understanding of how computers operate and it may be naïve to assume that they have adequate knowledge or skills. Thus, the importance of training cannot be understated.
3. Teachers are encouraged to use personal productivity applications in their everyday life. The skills that are developed help to build confidence and overcome fears about using technology as part of their lesson plans.
4. Teachers must be good evaluators of software. They must be able to identify that software is useful and helps to promote problem-solving skills, provide relevant information, and stimulate discussions.
5. Teachers should embrace the opportunity for students to assist and help change the learning paradigm of teachers being the sole source of the knowledge in the classroom.

6. Administration must provide an environment that allows teachers to make mistakes that do not make teachers feel like failures.

7. Administration should provide motivation for teachers to try to integration technology in the classroom. The motivation must help teachers endure the frustration and unease of adopting new technology.

8. Administration must be provide teachers with ongoing and onsite support that take both time and money.

The majority of these issues are focused on teachers not having adequate knowledge or experience using technology in the classroom. As Bitner and Bitner explain, teachers often need to see positive outcomes of using educational technology first hand before they are willing to adopt and change their existing practices [14]. It is unlikely that teachers can observe the positives outcomes if there is a significant technology barrier that must be overcome. Having a Fellow in the classroom reduced many of the issues related to deploying the games in the classroom and provided teachers an experience of observing the benefits of integrating educational technology in the classroom. Fellows exposed teachers to new technologies, which caused teachers to become proactive in learning additional skills on their own.

7.1 General Issues with Deployment

There are many general issues involved with deploying a game successfully. Computer labs must be reserved well in advance, supplemental material must be printed beforehand, and the games needed to be installed on the computers at school. Fellows had either to coordinate with the schools’ technology staff or work with the schools’ computers themselves. Sufficient lead-time was needed if the school's technology staff was used. In some cases, even if the technology administrators did not have to install a specific game on their schools computer, they would have to be informed to allow certain TCP/IP ports and traffic to be accessible. The amount of foresight and testing required by the Fellows to develop and deploy a game successfully is substantial. Some of STEAM's games utilized a ‘high score’ feature, which was very effective in enticing students to play a game more than once. To create this feature, an accessible database was needed which resided on STEAM's server. If this feature was implemented in a particular game, web connectivity could not be blocked by the school's technology staff in order for the system to function correctly.

Computer hardware and software changes rapidly, expecting that all middle schools will have the latest hardware specifications may be naive. When developing digital games, designers must be aware of the limited technologies with and should consider developing games that do not require advanced hardware or software support. For a project like STEAM, borrowing a school's computer may be one of the better ways a Fellow can prepare for a school visit. For example, a problem occurred during a deployment event using Mystery Minerals. Though this particular game was successfully deployed at other schools, a Fellow was surprised that the game crashed on the majority of a school's computers. Unable to solve the problem at the school, the Fellow borrowed a computer from the school that was crashing and one that was not. The Fellow spent several hours trouble-shooting source code before coming to the conclusion that the driver for the graphics card was out of date on the machine that was crashing. After taking the computers back and installing updated drivers on all machines, the game still crashed on a handful of machines. Frustrated and puzzled, the Fellow finally arrived to a solution and found that the computers that were crashing had hand resolution settings that were too low. Though this was a relatively an obscure problem, it serves as an example of why testing and good coding practices cannot be understated.

Students, especially in the Appalachian area, do not often read very well. One constant theme emerged, which was that game play information or scientific concepts should not be buried within lengthy text because most students could not maintain sufficient focus to read the material. For example, Stunt Park featured a very brief instruction set. Students often quickly ‘clicked through’ the relevant information on their screen and missed basic concepts, which hindered the educational benefits. To prevent this situation from occurring, some games featured timers that would not allow students to ‘click through’ the instructions. Even a balanced approach of presenting the right amount of information could not ensure students would read the material. More successful deployments efforts would begin by the Fellows giving a live demonstration of the game before allowing students to play the game on their own. Though this simple strategy took some time away from the class period, it was effective.

7.2 Issues due to the School

Technology (or the lack there of) was the greatest obstacle to overcome when deploying educational media in the classroom. In general, the computer hardware was outdated by almost a decade, with the processors alone being several generations behind current technologies. With these limitations, it was difficult to deploy technologies even when games were created on engines that do not require advanced hardware support [19]. The problems encountered when designing and testing, may not be the same problems found when deploying.

Older school systems in the Appalachian area are likely to have gone through consolidation or are forced to share undersized facilities with many grade levels. Many of the schools had one computer lab per school and at times, one
computer lab for both the middle and high schools. Scheduling was always an issue. It was advantageous for Fellows to
develop and use efficient methods for installing the games, which included creating executive files that could be access
via the project's website, have sufficient log-on identities for the students, and, when applicable, having a version of the
game that ran directly from an Internet web browser.

Games that depend heavily on the internet are affected by slow internet access. Students were paired in teams in
schools with slow access. Students really enjoyed playing educational games that were online and multiplayer by
design. Part of the draw for games like Weather Challenge is that students see their peers flying around and being
immersed in the 3D world. Even if schools had a reliable high-speed Internet connection, a class size of 20 to 30
playing the game at once would often cripple the ability to enjoy the game without lengthy rendering times or lag,
which negatively affected the students' interest. Games like Rafting Adventures feature media that requires a high-speed
internet connection. Videos or simulations that demonstrate events like the effects of agents of erosion are not as
meaningful to the students if they are slow to render.

8. Selected Testing of STEAM Games

One of the arguments that are consistently made in GBL literature is the lack of qualitative and or quantitative evidence
to support the benefits of educational games. To overcome this limitation, and more importantly, to measure the
implementation of STEAM's games, quantitative results from selected games are presented in this section. It should be
noted that more robust experimental procedure (i.e. randomized treatment and control groups) could not be
implemented due to an agreement between the participating schools and STEAM. This mutual agreement was based on
not withholding any instructional material from the students, which would have needed if control and treatment groups
would have been evaluated. Thus, for this investigation, a quasi-experimental model using a convenient sample and a
'research protocol' developed by the evaluators, teachers and administrators was followed and is summarized below.

- Consent forms were sent and were collected earlier in the academic year.
- Team developed pre- and post-test were finalized and located two weeks before the teacher taught the
  educational content found in the module being tested.
- Students were randomly divided into two groups, where special attention was given to the number of females
  and males and ethnicity in each group
- Group 1 was considered the treatment group
- Group 2 was considered the control group
- Stage 1 of Testing/Data Collection
  - Group 1 took the pre-test
  - Group 1 then went to the computer lab in order to play the game being tested least 2 or 3 times
  - Group 1 then took the post-test (which was identical to the pre-text except for the order of the questions.
  - Group 2 remained in the classroom and took the post-test (which was the same as the post-test given to Group
    1 without playing game
  - Group 2 would then go to the lab after Group 1 has finished their post-tests
  - Tests are then graded and an item analysis was performed.
  - The purpose of this pre/post testing strategy was to check the reliability of the instrument, to identify material
    already known by the students and removed from the pre/post testing in the next round if desired and
    finally to identify any unique characteristics of the software and its questions compared to the expected
    curriculum and unit testing presented by the classroom teacher.

- Stage 2 of Testing/Data Collection
  - After at least two weeks from the first day of testing, both groups were given the pretest
  - After the content was taught, but before the teacher gave the official test over the subject matter, both groups
    were to play the game and then given the post-test
  - Tests are then graded and an item analysis is preformed again to identify if any questions were not measuring
    the expected conditions.
  - Comparisons were made between 1) pre/post of experimental group in stage 1, posttests of the control and
    experimental groups in Stage 1 and the pre/posttests of the experimental and control groups in Stage 2.

8.1 Rafting Adventures Testing

In an investigation of Rafting Adventures, a two-tailed test was performed on 38 students in the first week of the study.
The results are presented in Table 1. The average score for the pre-test mean was 3.37 and the post-test mean was 4.13
(an average difference of 0.763). The test indicated that the students' average achievement significantly improved by
playing the game with a significance level (p-value) of less than 0.05.
Table 1: Rafting Adventures Results

<table>
<thead>
<tr>
<th></th>
<th>Mean</th>
<th>Std. Deviation</th>
<th>Std. Error Mean</th>
<th>t</th>
<th>df</th>
<th>Sig. (2-tailed)</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.763</td>
<td>1.83</td>
<td>0.298</td>
<td>-2.561</td>
<td>37</td>
<td>0.015</td>
<td></td>
</tr>
</tbody>
</table>

It should be noted that from the initial analysis, the factor for ‘class period’ was not significant (i.e. p<0.05). In a follow-up to the game play, classroom visitation occurred the following day in which the teacher verbally quizzed the students on the concepts presented the day before in the Rafting Adventure game. The teacher was visibly pleased at the abilities of the students to describe the geology presented in the Rafting Adventure. While this is antedotal in nature, the teacher felt that the ability of the students to describe and name the features presented in the game the following day to be an improvement over previous experience of introducing these concepts in the classroom. As a result, in a teacher-developed unit-test over the content, the classes scored on average 12% higher than in the previous year.

8.2 Mystery Minerals Testing

The second game investigated for this paper was Mystery Minerals. For this test, the results did not show significance (p > 0.05) at a particular school (School A) when it was first deployed. When another school played the game (School B), a significant difference was obtained between the means of the pre-test and post-test, see Table 2.

Table 2: Mystery Minerals Results

<table>
<thead>
<tr>
<th></th>
<th>Mean</th>
<th>Std. Deviation</th>
<th>Std. Error Mean</th>
<th>t</th>
<th>df</th>
<th>Sig. (2-tailed)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.354</td>
<td>1.907</td>
<td>0.275</td>
<td>4.919</td>
<td>47</td>
<td>0.000</td>
<td></td>
</tr>
</tbody>
</table>

One possible explanation may be in the socio-economics of the two locations. School A had a more affluent population (free-and-reduced lunch = 32%) while School B was considered to be in the lowest of the socio-economic bracket for free-and-reduced lunch (78%). The disparity in state achievement scores in science followed a similar pattern. Research [29] indicates that technology has the greatest impact on students with low achievement levels.

8.3 Energy Park Testing

The third, and final game investigated and presented in this paper was Energy Park. When deployed, this game showed a difference in the means from pre-test (mean of 6.38) and the post-test (mean of 6.63), there was not a significant difference at the p < 0.05 level given the size of the group (n = 90), see Table 3.

Table 3: Energy Park

<table>
<thead>
<tr>
<th></th>
<th>Mean</th>
<th>Std. Deviation</th>
<th>Std. Error Mean</th>
<th>t</th>
<th>df</th>
<th>Sig. (2-tailed)</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.242</td>
<td>2.301</td>
<td>-0.241</td>
<td>-1.002</td>
<td>90</td>
<td>0.319</td>
<td></td>
</tr>
</tbody>
</table>

Though a desired significance was not reached, students commented about how ‘fun’ the game was to play, but were unsure of ‘what the game was supposed to do in the class’. This indicated to researchers that the ‘fun factor’ was achieved, but a better job was needed to connect the science content with the game play.

Results from these three STEAM learning games were selected for this paper not because they produced the best results, but because their data sets were most complete. We have partial results from other STEAM learning games and will further collect data on their effectiveness. Those results will be submitted for publication in the future.

9. Conclusions

To create a fun and educational module, opinions from the university faculty researching the development, teachers, students, and Fellows must be combined to reach an acceptable implementation that satisfies all involved parties. Unsuccessful deployments are usually accredited to the steep learning curve required to build a program like STEAM. Game development takes time, and deployment is highly dependent upon the quality of game. Creating a breath of games that contain diverse content that can be used through an entire school year takes time and resources. For this reason, projects like STEAM need to be funded long term to allow for extended testing and revision of the games.

Educational computer games are an emerging and attractive technology in K-12 classrooms because today's students prefer learning techniques combining elements of education and entertainment in a digital format. Given the situation in
the Appalachian school districts of rural Ohio where students are from less-affluent families and their test scores are relatively low, it is important to investigate how alternative forms of content such as digital games can be created and used for improving learning, particularly science learning as in the case of STEAM.

The lessons learned through the experience and outcome of the STEAM project are summarized below.

- Students can be engaged in immersive learning opportunities provided by digital games. Teachers welcome the idea of integrating educational content into games and students feel more motivated to learn science. This should provide game designers in computer science and educators confidence in creating and using educational games in the future as an alternative to improve students’ learning skills.

- Creating digital learning games can be difficult in that it is delicate to balance between the educational and entertaining aspects of a game. In order to solve this issue, closely working with teachers and students appears to be an effective way that can be applied in educational game development. Teachers are experts in determining content suitability, content accuracy, and student engagement. The feedback from students is valuable because, while playing, students tend to find flaws of a game that have not been noticed during game development. They could even give specific suggestions regarding how to improve the game, the fun aspect in particular.

- Not all schools have up-to-date computers. Educational game designers must be aware of the technological situation of each school and should consider creating games that match the majority hardware or software capability, which is vital to a successful learning game deployment.

- Fellows played a significant role not only in game development, but in game deployment. Fellows being present in the classroom were very beneficial in that they could reduce the number of issues during game-play by testing the game in classroom in advance, and ensure the effectiveness of a game by incorporating it with the lesson plan precisely in terms of both content and timing. Fellows helped teachers become more technologically proficient and to feel comfortable when using the learning games in their classrooms.

- Though some of the modules deployed at the middle school showed statistical significance, not all of the games did. This could be due to several factors ranging from the technical skills the Fellows had while developing certain GBL technologies to the problems they encountered when deploying the games or representation of content.

- During all work within the project, the partner teachers were considered the experts in the development of content for insertion into the games being designed by Fellows. Teachers were asked to regularly review the content of each other’s games during and after development and again during deployment in the classroom to determine not only content accuracy, but also the understanding of the students when engaging with the content.

In summary, engaging, standards-based digital learning games were developed that were attractive to students and effective in improving learning. These innovations in learning technology development were only part of the factors that led to the success of this project. The team and creative and thorough approach in learning technology deployment was necessary to ensure that learners could actually reap full benefits of our learning technologies.

Acknowledgment The authors of this article would also like to acknowledge all of the graduate students and K-12 schools, teachers, and students who participated on the STEAM project. In particular, the authors would like to thank Steve Carroll, Tessa Cooper, Juan Flores, and Yanhui Fang for their contributions in preparing this manuscript.

This material is based upon work supported by the National Science Foundation under Grant No. 0538588. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References


