Implementation of Gamification and Digital Game-Based Learning into STEM Related Subjects

Dr. Linda B. Hayden
Principal Investigator
Elizabeth City State University
1704 Weeksville Road, CB 672
Elizabeth City, NC
haydenl@mindspring.com

Steffi Walthall
Mentor
Elizabeth City State University
1704 Weeksville Road, CB 672
Elizabeth City, NC
Steffid2113@gmail.com

Joselyn Hathaway
Elizabeth City State University
hathawayjoselyn@gmail.com

Tangee Beverly
Elizabeth City State University
tlbeverly96@gmail.com

***Abstract*— Gamification is a concept that has been utilized as a supplementary tool in and out of classrooms since before video games, and even its own definition, were as established in mainstream conversations today. Gamification is a common tactic used in educational, workplace, and commercial settings to promote engagement of participants, otherwise known as "players." A term coined in 2002 by Nick Pelling and later redefined in 2014 by Gartner, gamification refers to the integration of core gameplay mechanics-- such as rules, level progression, challenges, and rewards-- and tends to result in an increase in engagement and investment in the different areas of activity it is applied to [15]. According to Gartner, gamification "focuses on enabling players to achieve their goals" by aligning task goals with "player" goals which leads to the consequence of the task being completed because of this alignment [16].**

Gamification can occur in both digital and analog experiences, digital referring to engagement via the incorporation of computers, smartphones, and associated applications while analog references non-digital components such as loyalty cards or best attendance in a classroom. For the context of this team's goals, the focus was on the implementation digital gamification. Gamifying an everyday task or topic increases an individual's enjoyment and personal investment in a task because the incorporation of these gameplay elements breaks the mundanity of a simple activity and elevates it by adding a driving motivation of reward.

When applied to an educational context, enjoyment, engagement and investment in a subject have been shown to result in higher retention of the information learned [17]. Several studies have argued that gamification in education has a positive effect on learning and student achievements in classrooms [18]. By gamifying difficult subjects that have a tendency to be seen as "dry" and "technical" by younger participants, such as math and science, students have the potential to comprehend them at a faster rate.

Digital game-based learning, a similar concept to gamification with the express difference of actually incorporating "learning principles into immersive video game environments into immersive video game environments in an effort to provide a new tool for education that is as modern and adaptive" according to Prensky [19]. Video games represent a unique potential for educators because of their interactivity, accessibility, and the format's tendency for modification especially in relation to the implementation of different curriculums across grade levels.

The goal for this team was to encourage the incorporation of digital gamification of STEM concepts in K-6 educational settings through digital game-based learning. To accomplish this goal, the team set the objectives of conceptualizing and developing a digital, educational video game that can be further modified to incorporate different STEM-related curriculum and grade levels. This was completed utilizing Unreal Engine 4 to build and develop graphical and interactive components. The team also referenced the North Carolina Common Core Curriculum to develop educational questions to implement into the video game.

***Key Terms*— Gamification, STEM, Unreal Engine, UE4, Game Development, Cognitive Development, Digital Game-Based Learning, Game Development, Game Design**

I. INTRODUCTION

A. Gamification

Gamification is the process of implementing gameplay elements into non-gaming environments, such as classrooms and workplace [9]. The emergence of the Internet and computing has digital components, such as, videos and computer games to be used in classroom activities for the enhancement of learning [10]. Khan Academy and IXL are examples of interactive educational programs used globally in multiple schools that represent gamification. Originally, Khan Academy was a resource of educational videos in different subjects, such as, mathematics, science, history, art, and more. Khan Academy has gamified by combining their videos with interactive questions that give students points and rewards based on their correct answers. The new gamified features with Khan Academy are similar to IXL, which provides an environment that hosts a variety of subjects with questions that provide instant feedback. Students can earn points with each question answered correctly, which will be rewarded with trophies. The feedback about interactive programs, from students of different age groups, was positive. Some liked the independence while others enjoyed how much they were learning [11].

Kahoot is a free educational game development of the Lecture Quiz Research Project in 2006 at the Norwegian University of Science and Technology(NTNU). The purpose of Kahoot is to make learning fun across all subjects in any language, and can be used on different digital devices. It enables the connection between teacher and student interaction in a classroom environment of different group sizes. Kahoot can potentially promote motivation and learning among students, because of the embedded graphical interfaces and audio elements.

Teachers can create their own quizzes and surveys and/or use existing quizzes and surveys made available for the public to use. The scores are displayed at the end of each session and teachers are able to save the information in a digital document. Players are not required to register for a Kahoot account, but will be given the game pin prior to joining a specific game as lead by the game host [11]. A question will be displayed on screen and students have to answer the question correctly and quickly. It is a time based game. The more questions answered correctly in least amount of time, increase the chances of students competing against each other of becoming the top winner.

B. Digital Game-Based Learning

Digital game-based learning refers to how learning principles can be put into immersive video game environments to provide a new medium for education [5]. Unlike Gamification which takes hallmarks of gameplay, like game mechanics and rewards systems, and storytelling and puts them into a non-game context, digital game-based learning takes educational practices and puts them into a game made for the distinct purpose of teach the student within a specific immersive environment. Digital game-based learning can come in many forms, such as apps, browser games, or computer/console games.

Kahoot is a free educational game developed of the Lecture Quiz Research Project in 2006 at the Norwegian University of Science and Technology(NTNU). The purpose of Kahoot is to make learning fun across all subjects in any language, and can be used on different digital devices. It enables the connection between teacher and student interaction in a classroom environment of different group sizes. Kahoot can potentially promote motivation and learning among students, because of the embedded graphical interfaces and audio elements.

Teachers can create their own quizzes and surveys and/or use existing quizzes and surveys made available for the public to use. The scores are displayed at the end of each session and teachers are able to save the information in a digital document. Players are not required to register for a Kahoot account, but will be given the game pin prior to joining a specific game as lead by the game host [11]. A question will be displayed on screen and students have to answer the question correctly and quickly. It is a time based game. The more questions answered correctly in least amount of time, increase the chances of students competing against each other of becoming the top winner.

II. Research Questions

The following research questions guided this study:

1. What is Gamification?
2. What is Digital Game-Based Learning?
3. How do education and game development connect?
4. What are the benefits of gamification and digital game-based learning in the classroom?

III. Literature Review

A. Gamification

Regular computer gaming dates back to as far back of the 1970's and later evolved into educational games. These educational games started with two subjects: mathematics and science. They were initially based on imagination, programmed by teachers, and were educational without the student's realizing so [2]. In the past, having teachers develop and design computer games for their students was the ultimate step ahead in engaging students. Later in this portion of the research process, we will evaluate present day approaches to gamification in education.

As gaming pertains to present day education, there are certain goals that you must keep in mind in order to maintain the educational factor. When creating a game, you need to consider if it will keep your students motivated to do the work, increase their learning, and develop their skills. The game development industry is continuously growing and continuing to develop. Developers are creating games that they know will reach younger audiences everywhere. Since this audience is responding to this tactic, the same approach can be made toward developing educational games. As an educational game designer, the first step is to consider playful versus serious games and follow the correct design process. Playful games are those that focus on the aesthetics, visual contexts, and have the goal of getting an emotional response from the user [1]. When students play games that are considered playful, they tend to not gain any educational value from the game. Serious games are those that focus on contexts such as economics, education, health, industry, military, engineering and politics. These contexts

Simulate real-world situations and imply that there is a risk involved [1]. Even with the main ideas of making a game educational, it is still possible to make learning fun. According to Borges, the most common term linked to educational gamification is motivation. This term leads to seven main objectives for an educational game:

1. Mastering skills: improving student ability
2. Challenging: increasing the learning process
3. Engagement: engaging student interest in the learning process and activating interest
4. Improving Learning: activating the acquisition of knowledge
5. Behavioral Change: providing reward for correct responses and penalties for incorrect responses

6. Socialization: allowing room socialization and group learning with peers
7. Guidelines: demonstrating benefits in order to motivate students and address learning process problems

[1]. All seven objectives work simultaneously to reach a primary goal, which is to benefit the student's educational gain.

There are factors to consider as well that focus on the learning styles of students. Since each student does not learn the same, games should have adjustable elements in order to cater to the educational needs of the individual. The various learning styles include:

1. Visual-looking at images
2. Auditory-hearing sound, music, and rhythms
3. Physical-learning by doing
4. Verbal-speaking and writing
5. Logical-using reasoning
6. Social-group learning
7. Solitary-learning by themselves

[3]. Knowing the types of learning styles and the targeted audience will help game developers provide a game that is beneficial for the student's educational value. By taking the enjoyable elements of a game and combining them with education, the teaching process evolves in a positive and beneficial manner [5].

Today, students are known as "digital natives" due to their birth in the digital era [3]. When you compare present day generations to older generations, the access to technology is comparably different. Since the generations today are supporters of technology, the idea of combining technology and education would be ideal. Just like any other situation and implementation in education, there are cons. The negative side of gamification in education includes:

1. Inadequate access to technology
2. Lack of professional expertise in integrating new technologies
3. Resistance to change

Research also says that the human attention span has decreased from 12 seconds to 8 seconds due to an increase in the digital world. In order to increase this time, the suggestion has been made for teachers to use gamification to transfer new knowledge and information to students [3]. Using gamification in education is a way to defeat boredom with textbooks lectures [4]. In the past, students had no choice but to use textbooks and lectures to learn due to the lack of technological advancement. In present day education, the thought of conforming to the technological world for the advancement of student learning would be logical.

B. *The Benefits of Gamification and Digital Game-Based Learning in the Classroom*

Game development has expansive potential in relation to education as the immersive environment can greatly benefit students as well as creating an experienced based learning environment. The objective of a game is to increase the difficulty of the levels as the game goes on by providing solvable problems that require the player to make measured, tested, and learning-based decisions [6]. In a traditional classroom, teachers aim for their students to approach their work the same way. Researchers have found that complex video games that include survival and simulation provide the most benefits for the player. These benefits also overlap with students in the classroom. Some of the benefits include:

1. Aids in overcoming dyslexia
2. Improves balance
3. Increases decision making speed
4. Reduces stress and anxiety levels
5. Encourages teamwork
6. Improves motor skills
7. Enhances socialization and career-building skills

[6]. One important observation made about the list of benefits is that it addresses a learning disability; dyslexia. Another branch of dyslexia is dyscalculia, which is a person's difficulty with mathematical computations. Through the interaction of a mathematics game, all of the benefits discussed still qualify and can be applied to students in a mathematics classroom.

Psychologists emphasize a process called "deep practice" where one fully focuses on the task at hand and blocks out all other distractions around oneself. Video games provide a unique format that allows students to learn through games is also a way for them to learn from their mistakes try new approaches [7]. An educator's priority is to becomes their students "educational facilitator". This means that instead of telling the students all of the answers right away, the teacher allows room for the students to think and understand concepts on their own. This is particularly important as this relates to mathematics. An example would be when students are doing mathematics to solve for one or more unknown variables. This type of mathematics problem allows student to try trial and error, figure out where they made a mistake, and make corrections along the way.

Teachers can also be involved with their students during game play to maintain the in-classroom collaboration aspect. Doing so encourages teachers to maintain the instructional importance of the game and modify their approaches with their students. Parents are also encouraged to be involved during with the students' game by communicating with the teacher for updates on their progress [8]. Educational games in the classroom are a way to ensure inclusion and diversity of all students. Students will not only be engaged in the game but engaged in the classroom and the learning process.

IV. Methodology

A. *A Brief Summary of the Game Development Pipeline*

The game development pipeline follows a general pattern of procedure but is by no means a purely linear process. A typical game development pipeline is comprised of four major Milestones:

1. Concept
2. Pre-Production
3. Production
4. Post-Production

Within each of these milestones there are several different sub-processes that a development team will complete simultaneously to accomplish the milestones, from asset creation to programming, to testing, and bug fixing. For the purpose of this team's research, the concept, pre-production and early production phases were the active objectives.

B. Conceptualization

1) Brainstorming Concept and Defining Target Audience

The team began the development process by defining their target audience and brainstorming concepts for the game. It was decided to create a game that would be targeted towards fourth grade students with an emphasis on North Carolina Common Core Standards for mathematics. This specific grade level and subject matter was determined because of the transitory curriculum given to fourth grade students builds upon basic concepts developed by the lower grades while also beginning to implement more complex concepts such as multiplication, fractions, and decimals and the team's collective knowledge of the subject. The standards used in the development were [17].

2) Unreal Engine

For the game project, the team had two options for game engines, Unity and Unreal Engine 4. Unity is a cross-platform, real-time engine that was developed by Unity Technologies. Unreal Engine 4 (UE4) is also a cross-platform, real-time engine that was developed by Epic Games [19]. Both engines have similar capabilities and benefits, however it was ultimately decided that UE4 would be utilized for the development process because of the expansiveness of its documentation of the engine by the company itself as well as outside resources, the accessibility of the UE4 user interface in comparison to Unity, and the engine's Blueprint Visual Scripting, furthermore referred to as "blueprint," "blueprints," or "blueprinting". The blueprint system is a complete, node-based interface gameplay scripting system that is used to create gameplay elements from within Unreal Editor. The scripting language is utilized to define object-oriented classes or object in the engine that can have gameplay and mechanics implemented into them [19] The node-based nature of UE4 allowed for quicker implementation of the project's code and a faster grasp of the programming in comparison to Unity's C# based engine. The engine also possesses C++ capability. The specific engine version used for this project was 4.20.3.

C. Pre-Production

Game design documents (GDD) are used to map out the game. Topics that are usually discussed in these documents are the audience, platform, genre, gameplay, visuals, characters, level design, and story. These documents are live references for the development team to use to remain consistent to the progress of building the game. A GDD is meant to be thoroughly detailed over time with the constant addition of information.

The game development team's game design document outlined the premise of the game: a third-person space explore called Alien Invasion in which aliens have begun to invade the player's solar system and the player must reach their spaceship to defeat them. While going to their ship, the player encounters alien ship bases and must solve math equations to defeat destroy the alien shields and "blast" the aliens off of the planet.

The team also defined the game's mechanics inside of this document. Game mechanics provide gameplay by providing a set of rules or procedures that guide the player and the game response to the player's moves or actions [18]. The game mechanics for alien invasion were that: (1) the player could solve the problems by interacting with the kiosk answer spheres and if the problem was right it would take away health, and if it was wrong the player would be taunted by the aliens and (2) when the player answered all of the questions correctly the aliens would leave the planet, resulting in player progression through the game.

D. Production

1) Developing the Shield Blueprint

After creating the game project, the team began developing the Shield Blueprint which would house the alien characters that the player needed to eliminate from the planet. A blueprint was created with a shield mesh that would be destroyed after certain conditions were met [fig. A.]. A representational health bar was created through a widget blueprint and connected as a component inside the Shield Blueprint so that the player could see when the shield's health was affected, indicating how many more actions must be taken for the shield to disappear and the aliens to be "banished" from the planet. Later on a box collision volume was added to implement functionality for story elements inside of the Story Robot Blueprint.

2) Developing the Shield Health System

The Shield Blueprint health system was constructed inside of the shield blueprint event graph. An event tick was set to run a check every 3 milliseconds was connected to a cast to widget which targeted the health progress bar that was previously established. The 'cast to' runs through a 'set percent' that targets the health progress bar and runs a division equation of the health, a float variable established in shield blueprint, divided by 100 to determine the percentage of the health. This percentage is run through a branch statement, otherwise known as a Boolean, whose conditions are 'health' variable == 0. If those conditions are met the blueprint runs a sequence (ordered list of events) that will deactivate the shield mesh and the health bar [fig. B.]. In the Level Blueprint, a branch statement connected to a keyboard input sets the health. When the key is pressed the shield's health is reduced by 50%. For initial testing purposes, this was implemented as only a keyboard input connected to a 'set health', which determined the health value. When functionality was implemented in the Kiosk Blueprint, the branch statement was added with the condition referencing the Shield Blueprint that was connected to a local variable inside of the Shield Blueprint. If this condition is met the branch runs the health reduction once. This code is specifically implemented for the answer spheres that the team created to be the true answers [fig. C].

3) Kiosk Blueprint

The Kiosk Blueprint was developed as an object that the player could interact with in-game that would affect the Shield Blueprint. The Kiosk Blueprint comprised of two major parts: the problem hub and the three answer spheres. The problem hub displays the math problem that the player must solve and is always visible. The answer spheres are comprised of a sphere collision volume component, a widget component that stores the answer, and a sphere static mesh component.

Each Kiosk consists of one true/correct answer and two false answers, similar to a multiple choice question. The true and false answers are similar in construction; when the collision volume component has an overlap it 'casts to' the player character and enables input for the player character within the collision volume as well as setting the 'is enabled 'x'' variable and setting the visibility of the answer widget to true. When the component overlap ends, the player key input is disabled and the 'is enabled 'x'' variable and widget visibility are set to false [fig. D] [fig. E]. For the 'true/correct' answer sphere, if key input occurs, the game will take away health as described in the 'Developing section Shield Health System [fig. C]. For the 'false' answer spheres, a print string runs across the screen saying 'is False' [fig. F].

4) Implementing Characters

The player character model and animations were created in outside programs (Autodesk Maya and Mixamo, respectively) and then imported into the engine. UE4 already has a preexisting Third Person Character Blueprint available for use and the player character replaced the default character in this blueprint [fig. G]. A double jump was implemented into the Third Person Blueprint [fig. H] as well as custom animations for the player character.

5) Implementing Story Elements

The story functionality was employed into the game differently than some of the other gameplay elements by using a Blueprint Interface (BPI). A BPI is a collection of one or more functions - name only, no implementation - that can be added to other Blueprints" [14]. The team created a BPI called 'BPI Story Interface' with two primary functions, 'Pass thru Collision Story 'x'' and 'Left Collision 'x''[fig. I]. This BPI, when added to a blueprint in the Class Settings, allowed the respective blueprints (Story Robot and Shield Blueprint) to utilize nodes and create events based off of the functions inside of the BPI. The BPI is especially useful because it allows events to happen without having to initiate them in the level blueprint, meaning that the user can attach specific events to components within different blueprints as opposed to generally referencing the entire blueprint.

In the Shield Blueprint, the team placed a box collision and created a 'on component begin overlap' event. Instead of using the 'cast to' actor, the team attached a 'does implement interface' node that looks for blueprints that run the 'BPI Story Interface' BPI, which makes the resulting event specific to those two actor class blueprints. 'Does Implement Interface' is the condition requirement for the overlap event and if true sends the message "Passed through Collision Story 'x'" [fig. J]. The 'Component End Overlap' event fulfills the same purpose but sends the message 'Left Collision.'

The Story Robot Blueprint can now receive these messages and use the events 'Event Passed thru Collision Story 'x'' and 'Event Left Collision 'x''. 'Event Passed thru Collision Story 'x'' sets the visibility of the story widget and 'Event Left Collision 'x'' turns off the visibility [fig. K].

For the sake of efficiency, one default blueprint was developed for each major actor class blueprint and then later duplicated and the variable updated as needed. The blueprints were then placed into the level.

V. Conclusion

In conclusion, the team found that it is possible to implement games in the classroom. In order for this connection to be effective, there must be an understanding of the game development process, the curriculum of note, and how to implement the information in a meaningful way. As research mentioned before, teachers were the developers and forerunners as it pertained to the original execution of digital game-based learning and gamification in education. In the future, the intersection of the education and gaming worlds will be an ideal inclusion in common curriculums once the collaboration components are present. Based on our research, the gamification of STEM-related subjects would highly benefit students. The process would address students and classrooms in every way such as catering to multiple learning styles, creating challenges, keeping students engaged, and improving student learning.

As this research project relates to the creation of the game, it should be noted that the team did have complications and challenges during the process. The first complication was the allotted amount of time used for the training of the Unreal Engine. The first challenge was the restricted time frame. The actual coding to make objects interactive within the environment and responsive to player input took about three months. After the coding and logic problems were solved, we were able to include the animations, which provided their own inconsistencies. Another challenge was troubleshooting and finding bugs in the code outside of the team's control in relation to specific nuances of the engine's blueprint functionality. Some of the errors we received during the coding process were minor while others were major. Lastly, the limitations during this research included the need of dual screen computers to work on multiple tabs and windows.

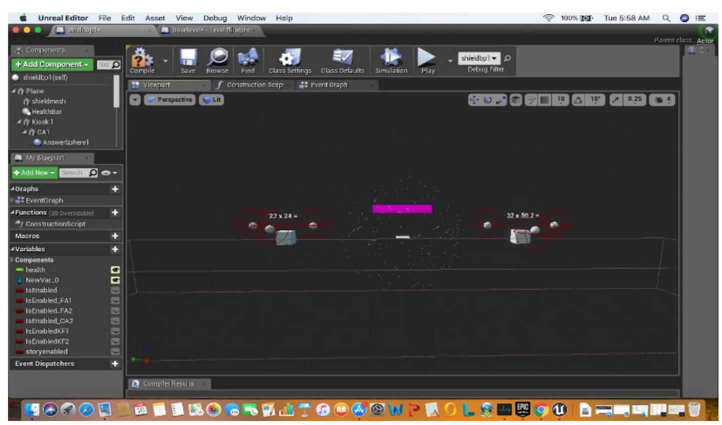


Fig. 1. Shield Blueprint

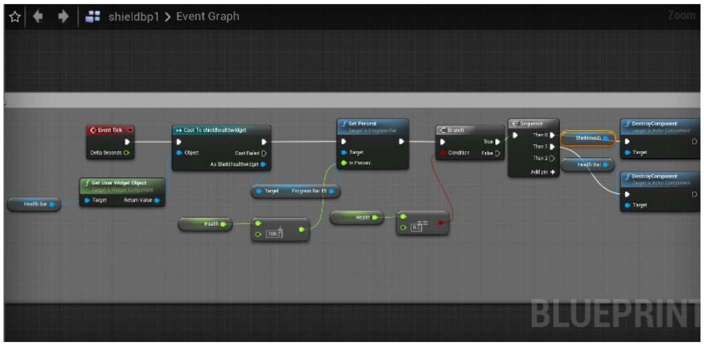


Fig. 2. Shield Health System

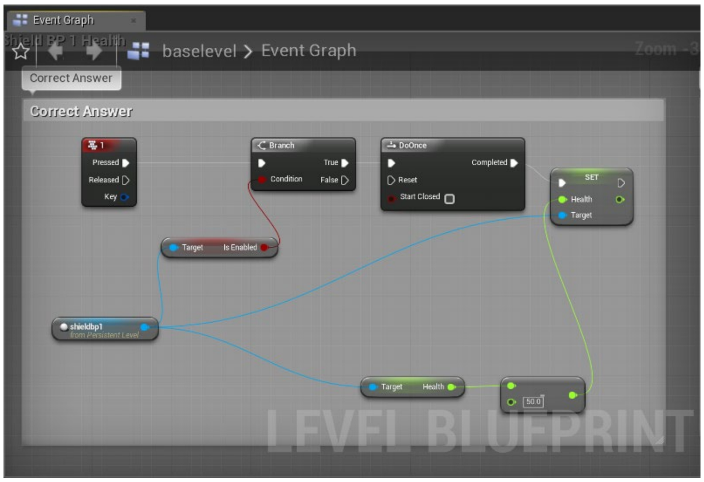


Fig. 3. Shield Health System

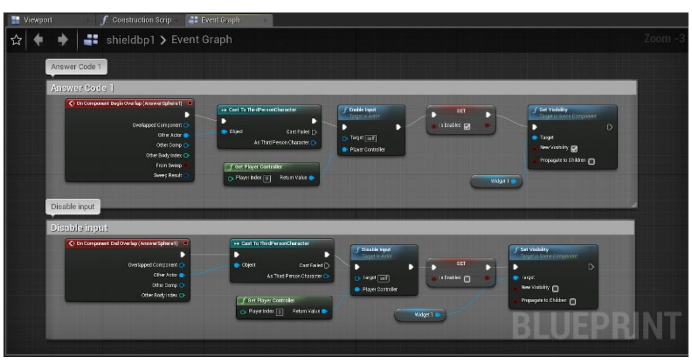


Fig. 4. Kiosk Blueprint

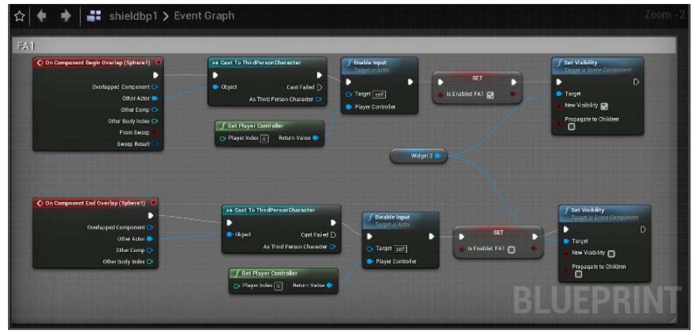


Fig. 5. Kiosk Blueprint

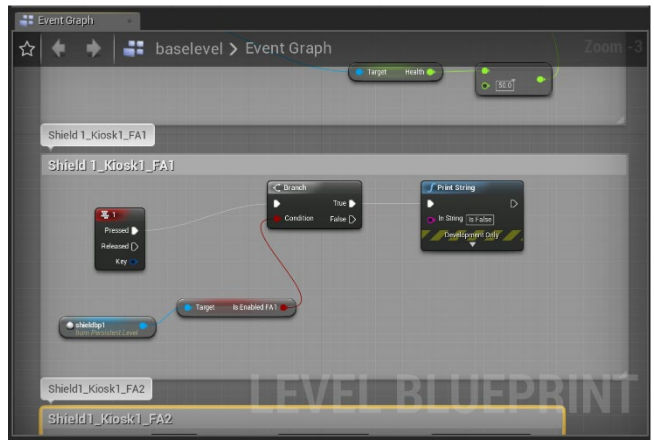


Fig. 6. Kiosk Blueprint

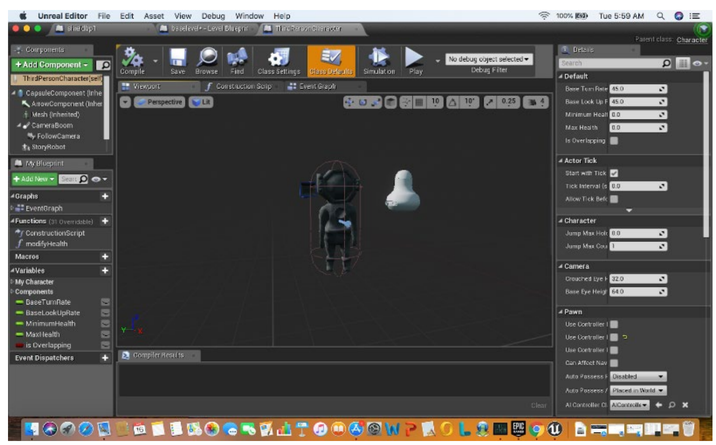


Fig. 7. Character Blueprint

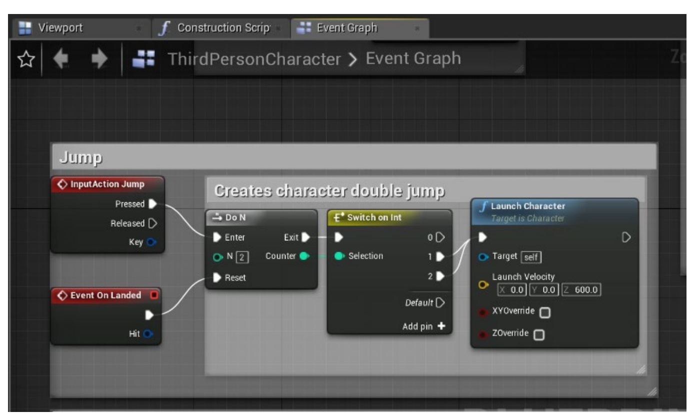


Fig. 8. Third Person Blueprint

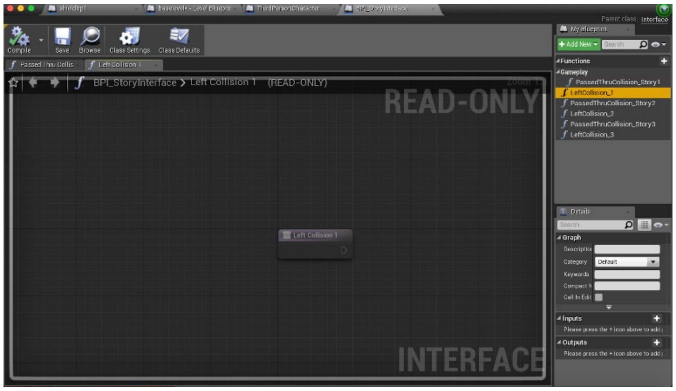


Fig. 9. Collision Blueprint

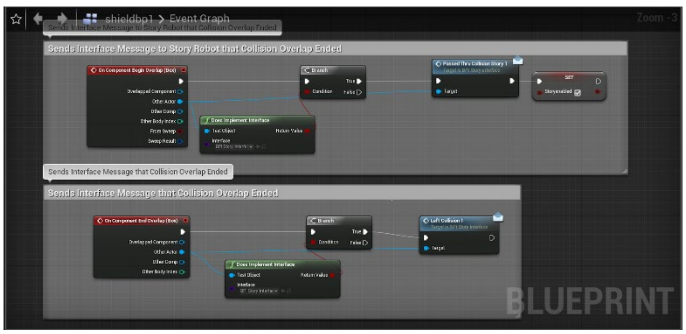


Fig. 10. Collision Blueprint

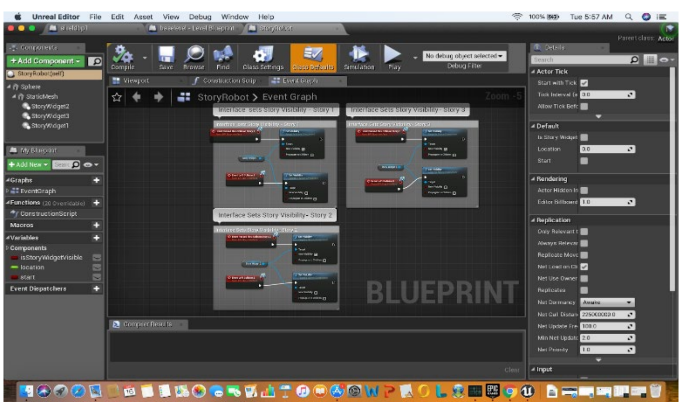


Fig. 11. Story Blueprint

VI. Future Work

In the future, the team would like to further continue the development of the game to achieve a more immersive experience in terms of functionality, interaction, and aesthetics. We would also want to implement more game mechanics to increase the visuals and interaction of the game. The team would also push the game to alpha and beta versions. An alpha version would allow for quality testing within the development team. With a completed beta version of the game the team would collaborate with local educators to begin conducting beta-tests on a focus group of students. The team would develop a survey for the participants on the educational content and enjoyment of the game. Simultaneously, the team would collaborate with the educators to determine if the game strengthens and reinforces the participants' comprehension of the subject matter over time. Lastly, team would also continue modify their game by expanding the availability to other grade levels and incorporate more STEM-based curriculum.

Acknowledgments

The team would like to acknowledge Dr. Linda B. Hayden who has provided the necessary resources for this research project. In addition, the team wishes to recognize Steffi Walthall for her guidance, input, and support throughout the duration of this project.

References

[1] S. D. S. Borges, V. H. S. Durelli, H. M. Reis, and S. Isotani, "A systematic mapping on gamification applied to education," *Proceedings of the 29th Annual ACM Symposium on Applied Computing - SAC 14*, Mar. 2014.

[2] R. J. Jones, "From little Acorns – a brief history of computer games in the classroom," *The Conversation*, 19-Sep-2018. [Online]. Available: https://theconversation.com/from-little-acorns-a-brief-history-of-computer-games-in-the-classroom-76746.

[3] 10, 390, and 65354, "How Does Gamification Effect the Learning Process?," *The Edvocate*, 14-Aug-2018. [Online]. Available: https://www.theedadvocate.org/how-does-gamification-effect-the-learning-process/.

[4] C. Miller, "Gamification as a Business Strategy," *The Gamification of Higher Education*, vol. 40, 2013.

[5] J. Simões, R. D. Redondo, and A. F. Vilas, "A social gamification framework for a K-6 learning platform," *Computers in Human Behavior*, vol. 29, no. 2, pp. 345–353, Mar. 2013.

[6] "Game Changer: Video Games and Real-World Problem Solving, Part One," *Game Changer: Video Games and Real-World Problem Solving, Part One | Information Technology | University of Arizona*. [Online]. Available: https://it.arizona.edu/blog/game-changer-video-games-and-real-world-problem-solving-part-one.

[7] T. Zimmer, "How Video Games Can Help to Improve Problem Solving and Creativity," *Agents of Geek*, 12-Apr-2019. [Online]. Available: https://www.agentsofgeek.com/2017/07/how-video-games-can-help-to-improve-problem-solving-and-creativity/.

[8] S. Slota, "Well-Designed Video Games Can Enhance Problem-Solving Skills and Make Learning More Effective," *Neag School of Education*, 29-May-2013. [Online]. Available: https://education.uconn.edu/2013/05/29/well-designed-video-games-can-enhance-problem-solving-skills-and-make-learning-more-effective/#.

[9] A. Scott and C. Neustaedter, "Analysis of Gamification in Education - cLab." [Online]. Available: http://clab.iat.sfu.ca/pubs/Stott-Gamification.pdf.

[10] D. Tan, M. S. M. Singh, and M. Ganapathy, "(PDF) Kahoot! It: Gamification in Higher Education," *ResearchGate*, Mar-2018. [Online]. Available: https://www.researchgate.net/publication/320182671_Kahoot_It_Gamification_in_Higher_Education.

[11] M. Lynch, "Will gamification be the future of education?," *The Edvocate*, 24-Jan-2017. [Online]. Available: https://www.theedadvocate.org/will-gamification-be-the-future-of-education/.

[12] "STANDARD COURSE OF STUDY," *Mathematics Standard Course of Study*, 2018. [Online]. Available: http://www.ncpublicschools.org/curriculum/mathematics/scos/current/#unpacking.
[13] "The Beginner's Guide to Game Mechanics," *The Ultimate Resource for Video Game Design*, 03-Apr-2019. [Online]. Available: https://www.gamedesigning.org/learn/basic-game-mechanics/.
[14] "Unreal Engine," *What is Unreal Engine 4*. [Online]. Available: https://www.unrealengine.com/en-US/.
[15] "What is the Definition of Gamification and What Does it Mean?," *Growth Engineering*, 12-Mar-2019. [Online]. Available: https://www.growthengineering.co.uk/definition-of-gamification/.
[16] B. Burke, "Gartner Redefines Gamification," *Gartner Blog Network*, 04-Apr-2014. [Online]. Available: https://blogs.gartner.com/brian_burke/2014/04/04/gartner-redefines-gamification/.
[17] M. Prensky, "Digital Game-Based Learning," *Digital Game-Based Learning*, vol. 1, no. 1, Oct. 2003.
[18] M. E. O'Riley, "The Question of Digital Game Based Learning: An Investigation into the Potential Promises and Perils of Education's Golden Goose," *The College at Brockport: State University of New York Digital Commons @Brockport*, 2016.
[19] M. Prensky, "The Games Generations: How Learners Have Changed by Marc Prensky," *Digital Game-Based Learning*, 2001.
[20] "Digital Game-Based Learning," *Digital Game-Based Learning - ETEC 510*. [Online]. Available: http://etec.ctlt.ubc.ca/510wiki/Digital_Game-Based_Learning.
[21] U. Technologies, "Unity," *Unity*. [Online]. Available: https://unity.com/.
[22] "Unreal Engine," *What is Unreal Engine 4*. [Online]. Available: https://www.unrealengine.com/en-US/.
[23] "Blueprints Visual Scripting," *Unreal Engine*. [Online]. Available: https://docs.unrealengine.com/en-us/Engine/Blueprints.
[24] "Game Design Documents," *The National STEM Video Game Challenge*. [Online]. Available: http://stemchallenge.org/resources/game-design-documents/.

Appendix

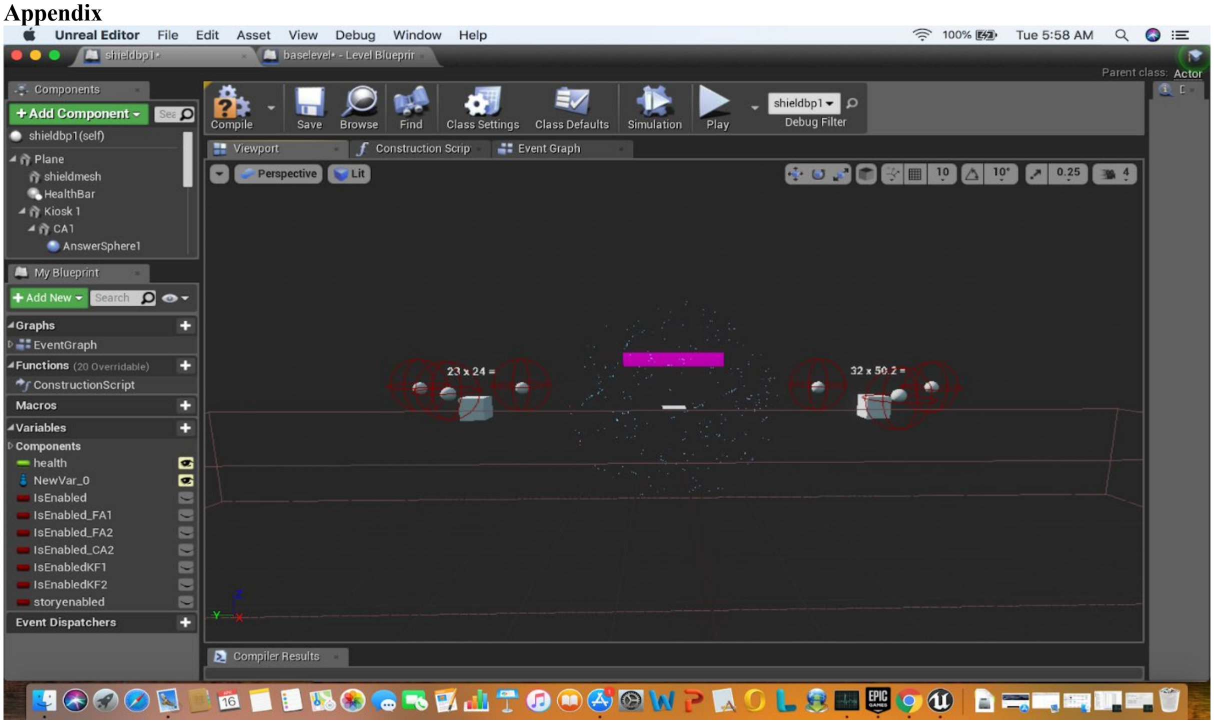


Fig a

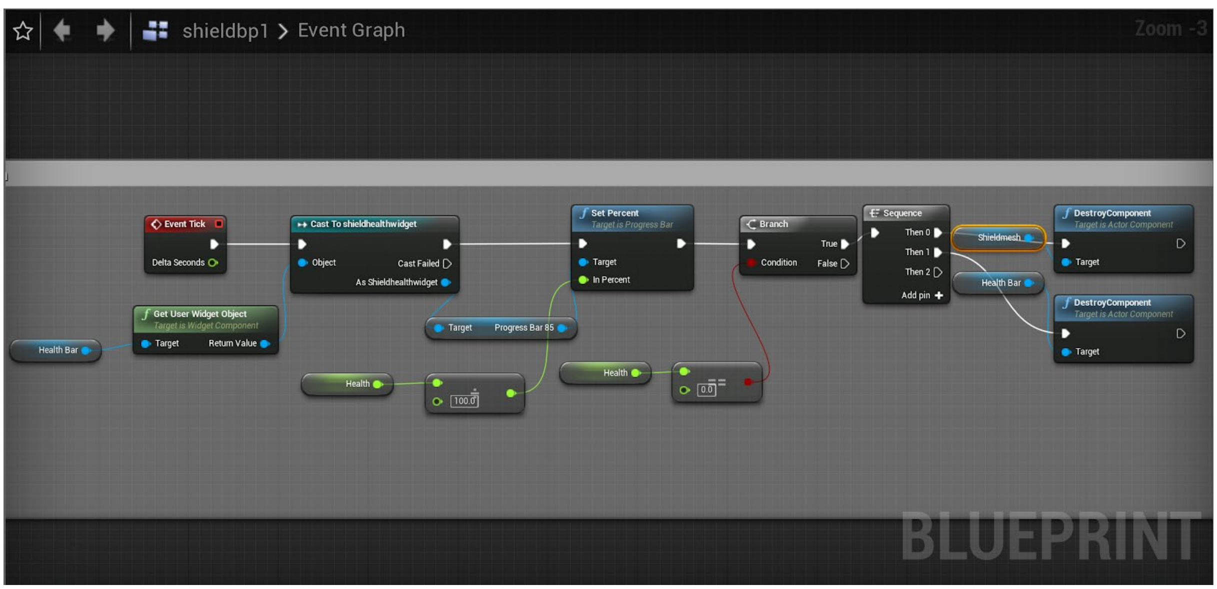
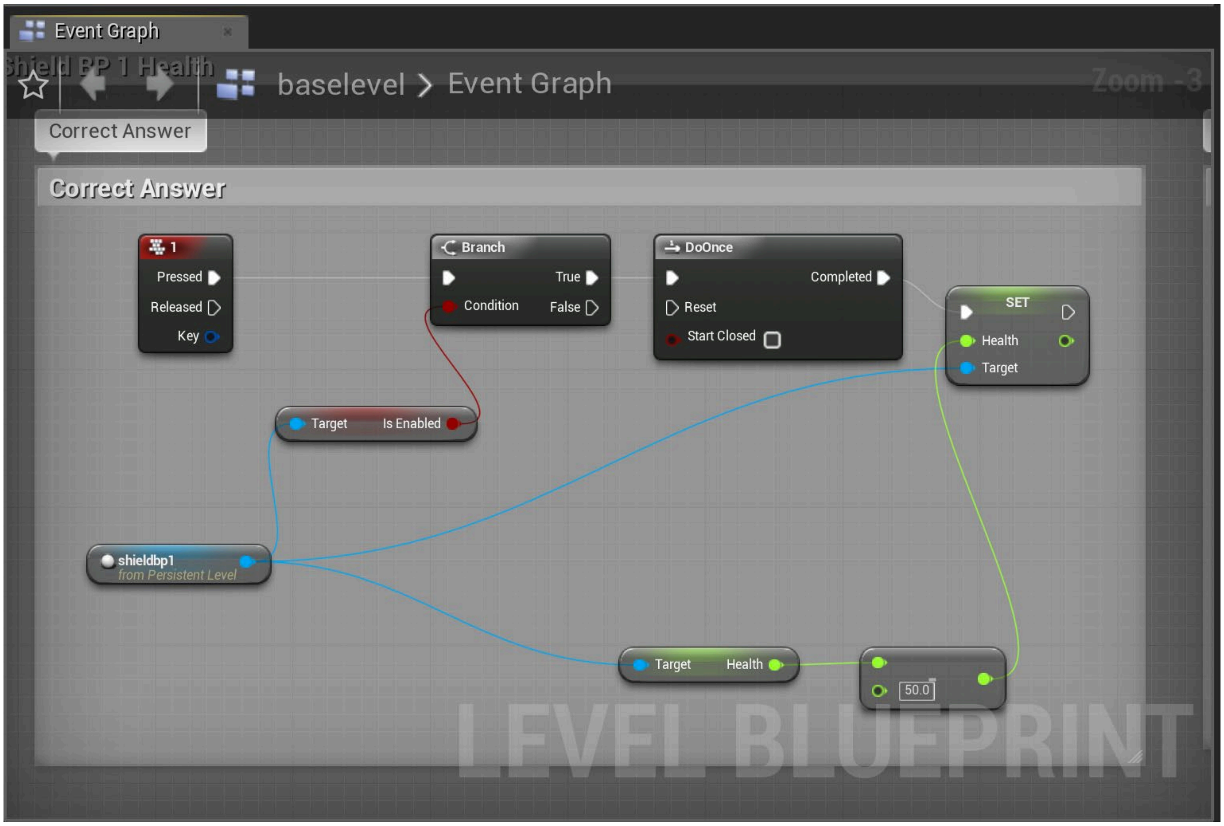


Fig b

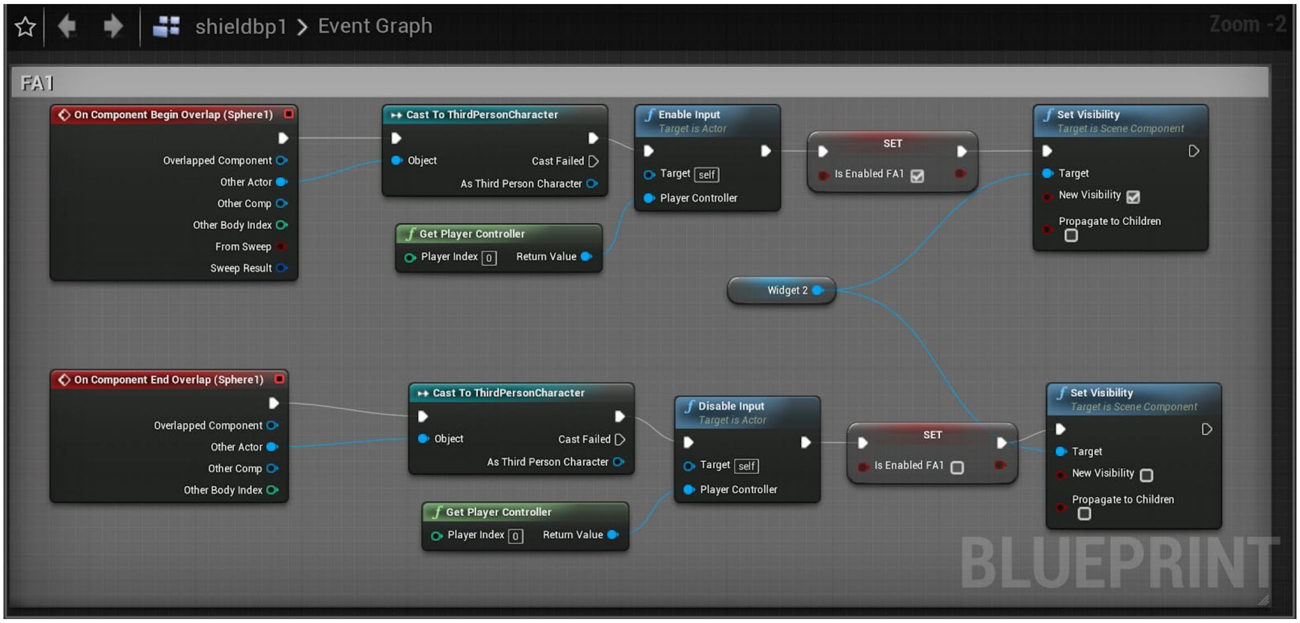


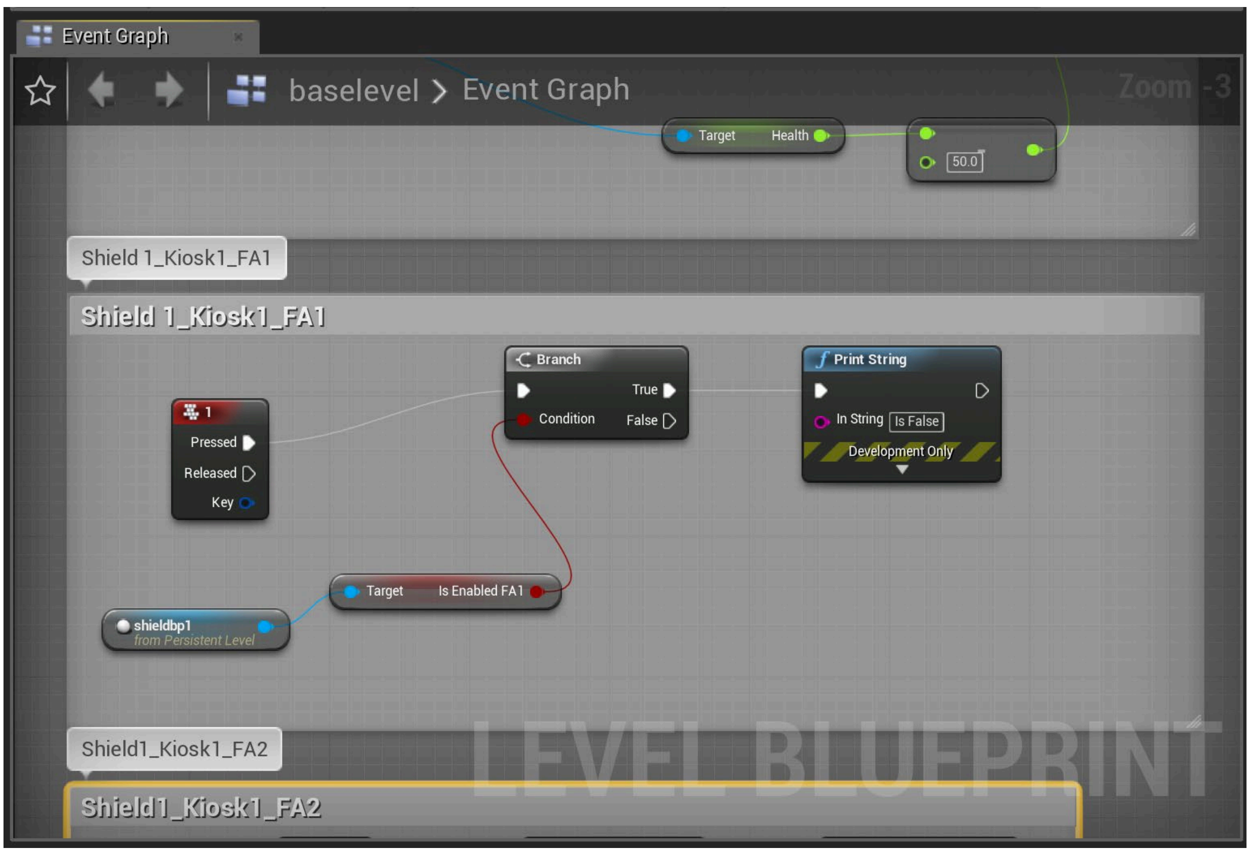
Event Graph
baselevel > Event Graph
Zoom -3
Correct Answer
Correct Answer
1
Pressed
Released
Key
Branch
True
Condition
False
DoOnce
Completed
Reset
Start Closed
SET
Health
Target
Target
Is Enabled
shieldbp1
from Persistent Level
Target
Health
50.0
LEVEL BLUEPRINT

Fig c

Viewport
Construction Scrip
Event Graph
shieldbp1 > Event Graph
Zoom -3
Answer Code 1
Answer Code 1
On Component Begin Overlap (AnswerSphere1)
Overlapped Component
Other Actor
Other Comp
Other Body Index
From Sweep
Sweep Result
Cast To ThirdPersonCharacter
Object
Cast Failed
As Third Person Character
Get Player Controller
Player Index
Return Value
Enable Input
Target is Actor
Target
self
Player Controller
SET
Is Enabled
Set Visibility
Target is Scene Component
Target
New Visibility
Propagate to Children
Widget 1
Disable input
Disable input
On Component End Overlap (AnswerSphere1)
Overlapped Component
Other Actor
Other Comp
Other Body Index
Cast To ThirdPersonCharacter
Object
Cast Failed
As Third Person Character
Get Player Controller
Player Index
Return Value
Disable Input
Target is Actor
Target
self
Player Controller
SET
Is Enabled
Set Visibility
Target is Scene Component
Target
New Visibility
Propagate to Children
Widget 1
BLUEPRINT

Fig d


shieldbp1 > Event Graph
Zoom -2
FA1
On Component Begin Overlap (Sphere1)
Overlapped Component
Other Actor
Other Comp
Other Body Index
From Sweep
Sweep Result
Cast To ThirdPersonCharacter
Object
Cast Failed
As Third Person Character
Get Player Controller
Player Index
Return Value
Enable Input
Target is Actor
Target
self
Player Controller
SET
Is Enabled FA1
Set Visibility
Target is Scene Component
Target
New Visibility
Propagate to Children
Widget 2
On Component End Overlap (Sphere1)
Overlapped Component
Other Actor
Other Comp
Other Body Index
Cast To ThirdPersonCharacter
Object
Cast Failed
As Third Person Character
Get Player Controller
Player Index
Return Value
Disable Input
Target is Actor
Target
self
Player Controller
SET
Is Enabled FA1
Set Visibility
Target is Scene Component
Target
New Visibility
Propagate to Children
BLUEPRINT
Fig e


Event Graph
baselevel > Event Graph
Zoom -3
Target
Health
50.0
Shield 1_Kiosk1_FA1
Shield 1_Kiosk1_FA1
Branch
True
Condition
False
Print String
In String
Is False
Development Only
1
Pressed
Released
Key
Target
Is Enabled FA1
shieldbp1
from Persistent Level
Shield1_Kiosk1_FA2
Shield1_Kiosk1_FA2
LEVEL BLUEPRINT
Fig f

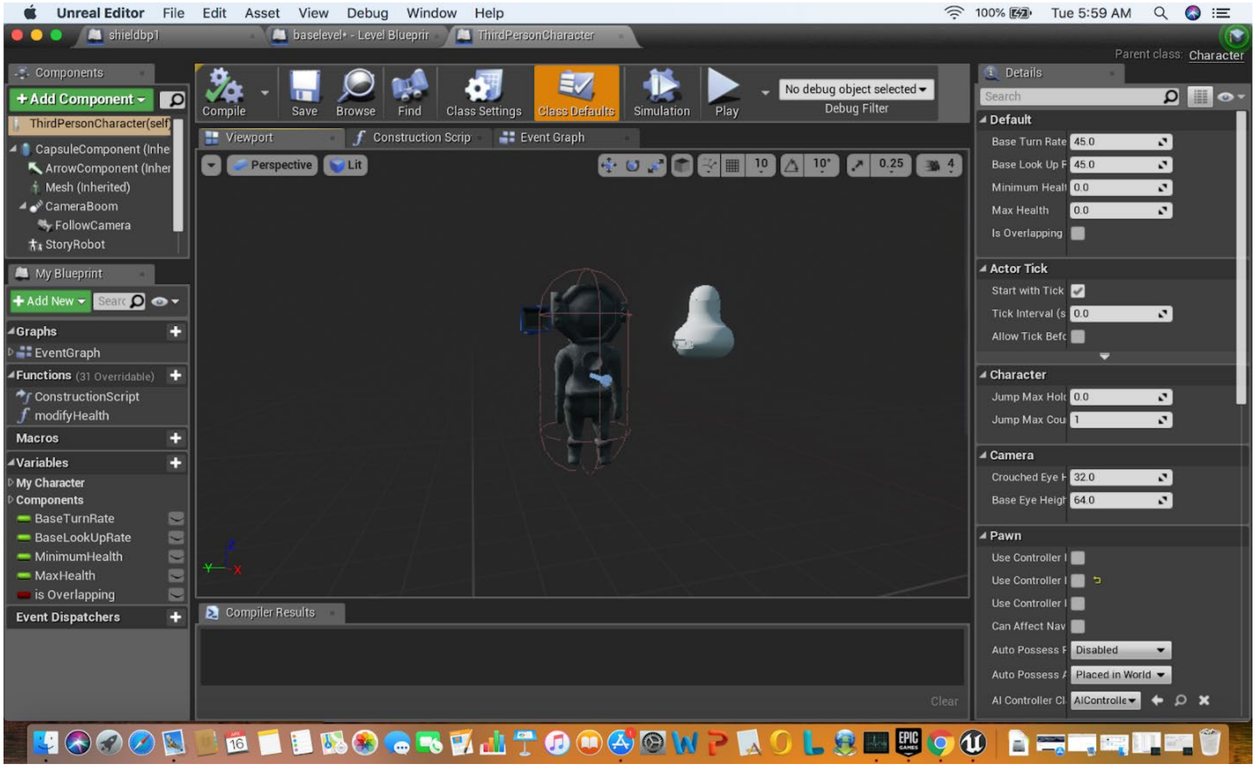
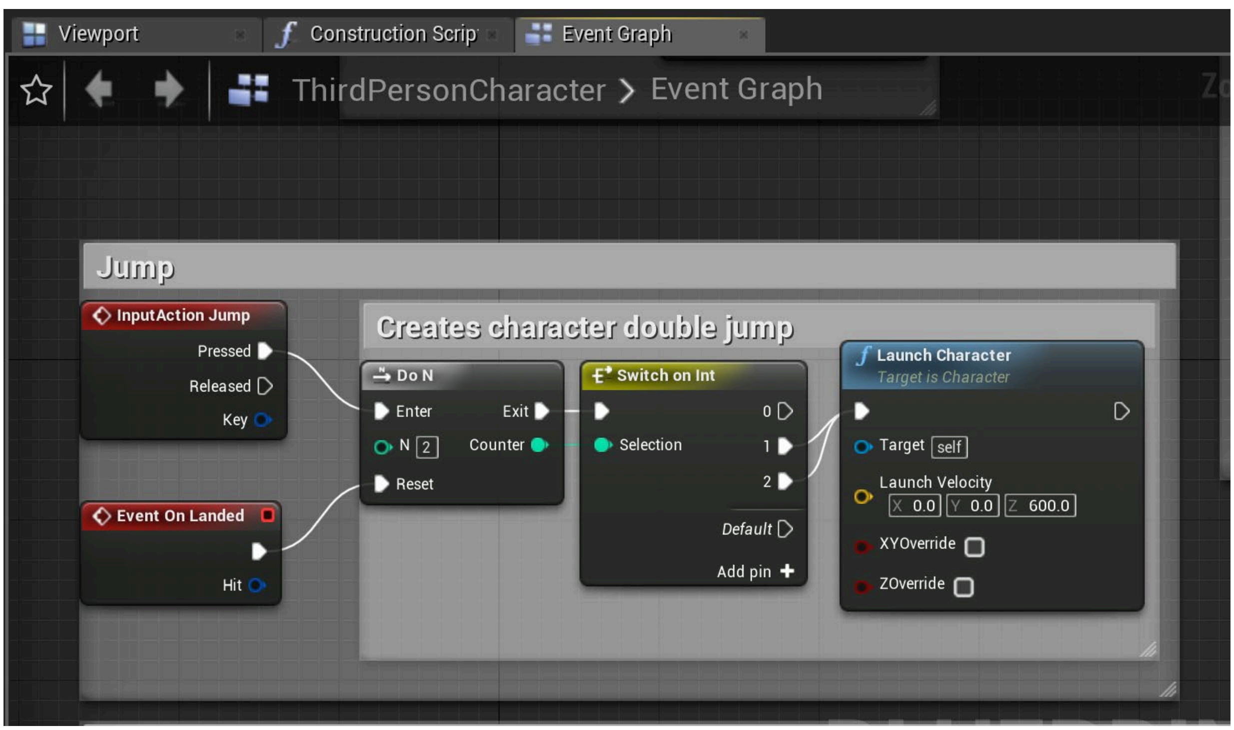


Fig g



Viewport
Construction Scrip
Event Graph
ThirdPersonCharacter > Event Graph
Jump
InputAction Jump
Pressed
Released
Key
Creates character double jump
Do N
Enter
Exit
N 2
Counter
Reset
Switch on Int
Selection
0
1
2
Default
Add pin
Launch Character
Target is Character
Target self
Launch Velocity
X 0.0 Y 0.0 Z 600.0
XYOverride
ZOverride
Event On Landed
Hit

Fig h

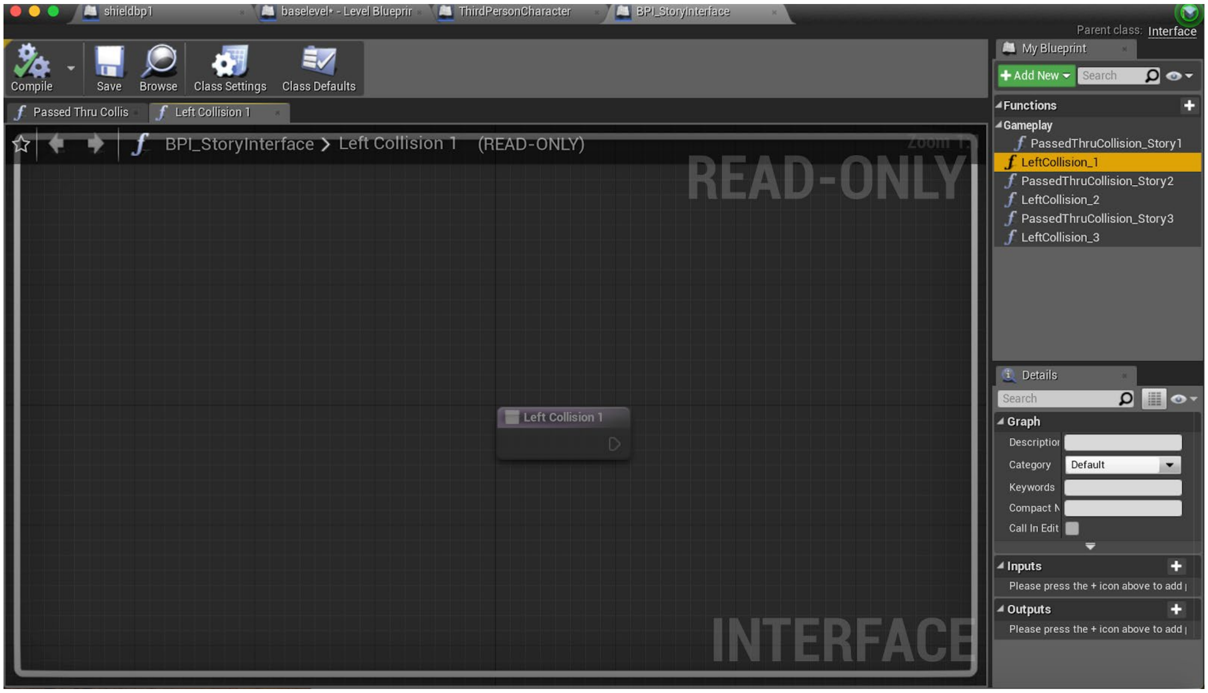

shieldbp1
baselevel* - Level Blueprint
ThirdPersonCharacter
BPI_StoryInterface
Parent class: Interface
Compile
Save
Browse
Class Settings
Class Defaults
Passed Thru Collis
Left Collision 1
BPI_StoryInterface > Left Collision 1 (READ-ONLY)
READ-ONLY
Left Collision 1
INTERFACE
My Blueprint
Add New
Search
Functions
Gameplay
PassedThruCollision_Story1
LeftCollision_1
PassedThruCollision_Story2
LeftCollision_2
PassedThruCollision_Story3
LeftCollision_3
Details
Graph
Category
Default
Keywords
Inputs
Please press the + icon above to add
Outputs
Please press the + icon above to add

Fig i

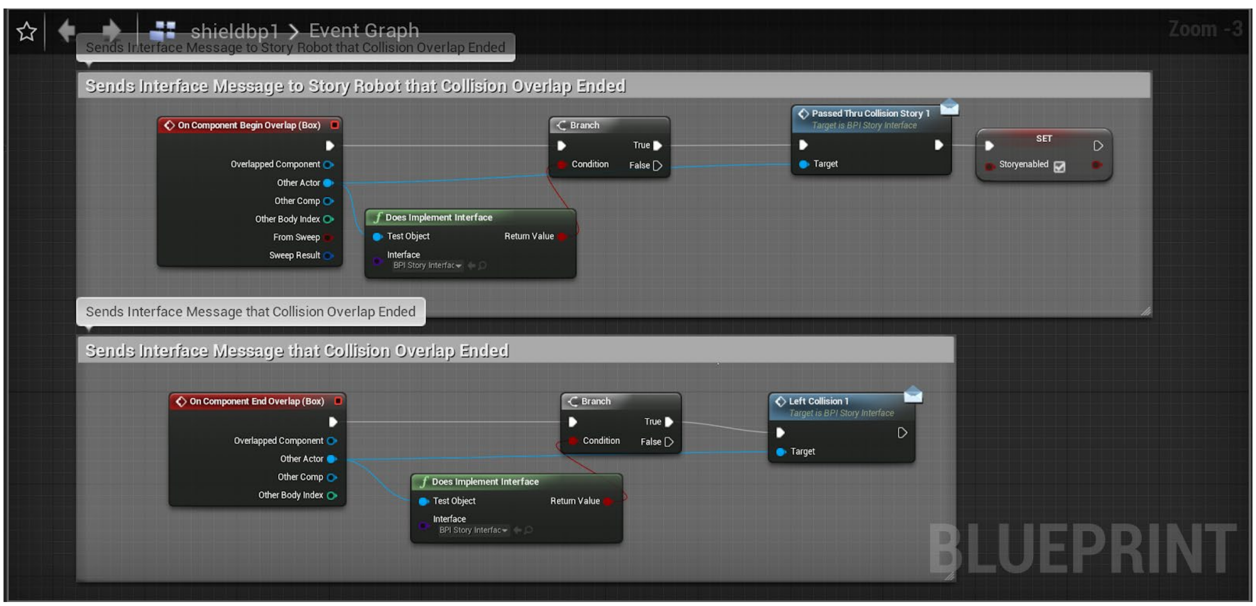
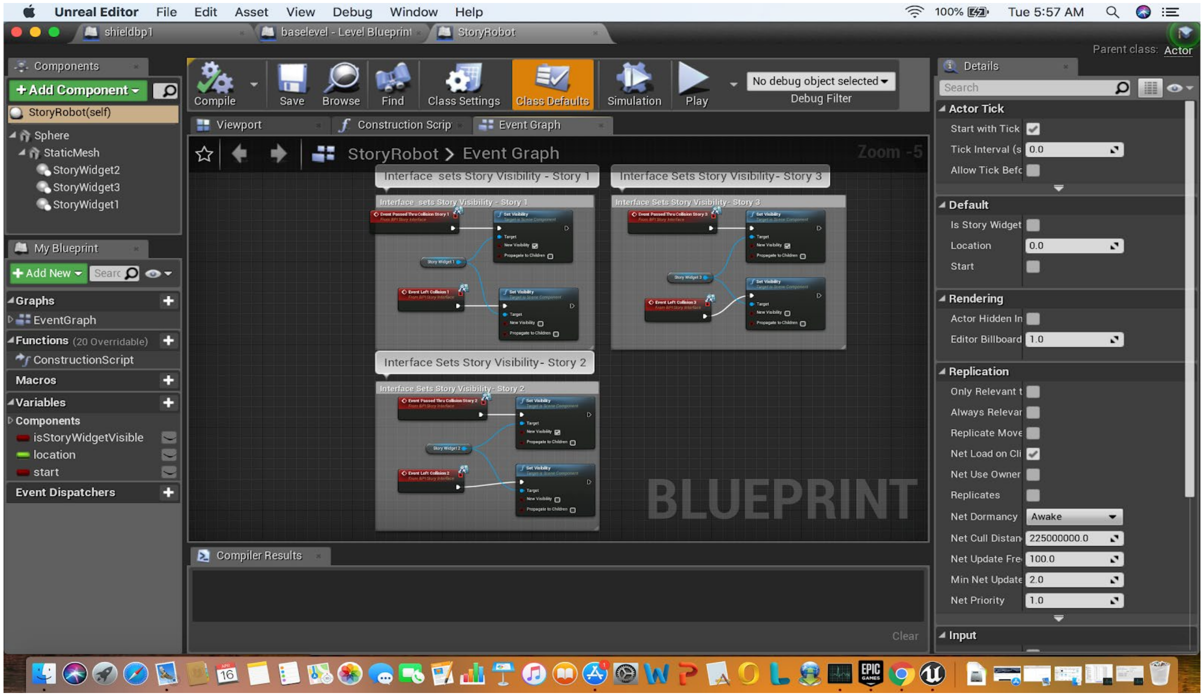

shieldbp1 > Event Graph
Zoom -3
Sends Interface Message to Story Robot that Collision Overlap Ended
On Component Begin Overlap (Box)
Overlapped Component
Other Actor
Other Comp
Other Body Index
From Sweep
Sweep Result
Branch
True
Condition
False
Passed Thru Collision Story 1
Target is BPI Story Interface
Target
SET
Storyenabled
Does Implement Interface
Test Object
Return Value
Interface
Sends Interface Message that Collision Overlap Ended
On Component End Overlap (Box)
Left Collision 1
BLUEPRINT

Fig j



Unreal Editor File Edit Asset View Debug Window Help
100% Tue 5:57 AM
shieldbp1
baselevel - Level Blueprint
StoryRobot
Parent class: Actor
Components
+ Add Component
StoryRobot(self)
Sphere
StaticMesh
StoryWidget2
StoryWidget3
StoryWidget1
My Blueprint
+ Add New
Graphs
EventGraph
Functions (20 Overridable)
ConstructionScript
Macros
Variables
Components
isStoryWidgetVisible
location
start
Event Dispatchers
Compile
Save
Browse
Find
Class Settings
Class Defaults
Simulation
Play
No debug object selected
Debug Filter
Viewport
Construction Scrip
Event Graph
StoryRobot > Event Graph
Zoom -5
Interface sets Story Visibility - Story 1
Interface Sets Story Visibility- Story 3
Interface Sets Story Visibility- Story 2
BLUEPRINT
Compiler Results
Clear
Details
Search
Actor Tick
Start with Tick
Tick Interval (s
0.0
Allow Tick Befo
Default
Is Story Widget
Location
0.0
Start
Rendering
Actor Hidden In
Editor Billboard
1.0
Replication
Only Relevant t
Always Relevar
Replicate Move
Net Load on Cli
Net Use Owner
Replicates
Net Dormancy
Awake
Net Cull Distan
225000000.0
Net Update Fre
100.0
Min Net Update
2.0
Net Priority
1.0
Input

Fig k