1. **Introduction** 
   1. **Statement of the problem** 
      1. **General Problem**

**Learners who engage in basic technical computer programming lessons find it hard to learn its concepts in comprehensive way.**

In order to successfully learn about something doesn’t only require academic skills but it also requires full interest of the learner. In fact according to John Dewey’s book—The Interest and Effort in Education—“it plays a big part in anyone’s learning ability about a certain topic.” Without developing an interest to the lesson, it would be almost impossible for the learner to absorb the essence of the topic.

One of the course/subjects that suffers with learner’s loss of interest is computer programming. Computer programming lessons is composed of highly technical terms and concepts that, most of the time, intimidates students and cause them to flee from the lessons. These concepts require practical understanding or application in order for students to absorb its core ideas.

* + 1. **Specific Problems**

**Non-academic virtual games consume young learners’ opportunity to learn.**

Virtual games have been one of the most consuming past time of our generation. According to Washington Post, our generation of teens consumes an average of 7½ hours per day with mobile devices. Unlike the past generations, gadgets are most accessible in our times for any age. Currently according to the survey we have conducted with the students of STI Ortigas-Cainta, there are 33/70 students who spend plays virtual game 4 or more days a week. While 30/70 students spend 5 or more hours a day playing video games. Virtual games consume the time that they should’ve been spending with their studies. As a result, more and more students neglect their studies and lose their interest for it.

**Intimidations from technical computer programming ideas affect learner’s performance.**

Salil Sethi, Founder & CEO at GoSchoolWise.com, McKinsey alum, and MIT Sloan graduate, stated that first reason why most students Computer Science dropout is because of the new language system that is necessary for computer programming. 2nd is that workload is too overwhelming because students are studying different languages at the same time.

Computer programming is composed of collective technical terms that most of the time overwhelms new learners. Scenarios like these sometimes make it hard for the learners to follow the lesson. It makes it harder for them to cope with the succeeding lessons because of they haven’t yet understood the logic of the previous lessons.

According to Tech Occult—the article Problem With Training The Next Generations of Scientists, Engineer and Mathematicians—“roughly about 30%-60% of Computer Science students dropout after failing their first computer programming class.”

A research, Why Students Fail To Graduate ICT-Related Curricula at University Level, conducted surveys to ICT dropouts about what their former university should change. 22% said, “Subjects shouldn’t be too hard”. 19% answered that “More practical studies and less theories and technicalities. 8% requested to “Lessen study load.”

Due to computer programming’s technicality, many students fail to perform well. Learners are being overwhelmed by the technical terms used in programming lessons that is why they easily lose their interest in the discussion. Most of the students fail to see the practical sense of the lessons because they seldom see how the logic works. As a result some students totally lose interest in studying programming and some, misinterprets the situation to be inability to comprehend with the lesson.

**Learners pursuing computer programming in college without any prior knowledge of what they’re taking is often overwhelmed by heavy loads of new things to study.**

According to the survey the proponents have conducted at STI Ortigas-Cainta, 67/70 students said that it would be a great help for them to gain prior knowledge/idea of what their incoming subjects are basically about. TechOccult also stated that, “60 percent of Science, Engineering and Computer major students either dropout or change their major.” Salil Sethi, Founder & CEO at GoSchoolWise.com, McKinsey alum, and MIT Sloan graduate also stated that, “the 5th of all reasons why computer major students dropped out is the unpreparedness of the course structure.” They never had even the basic idea of what computer programming subjects are about.

* 1. **Current State of Technology**

Media plays out a huge part in our generation. Our current generation of teens uses a large portion of their time in playing various kinds of video games. Washington Post stated that an average teen spends 7½ hours a day with their mobile phones. We are even labeled as the Digital Natives by Rappler and other journalism companies.

According to ESA, “35% of games played worldwide are on smartphones”. Smartphones are utilized by different operating systems such as android, IOS, windows and etc. These devices use touchscreen technology, as it’s basic navigation. Games designed for smartphone uses different styles of controls. Examples are analog swipes, swipes to move, touch, commands and etc.

Video games have many varieties of genre like puzzle, educational, strategy, first-person shooters, war, artillery and many more.

Though video games have been a very helpful for experiencing temporal entertainment, we can’t deny that it dealt negative side effects to those who got hooked with it—especially youth. Some examples are the effects that violent games such as first-shooter games, “Violent games are significantly associated with: increased aggressive behavior, thought, and affect increased physiological arousal; and decreased prosocial helping behavior.” (Anderson, 2003, Myths and Facts, para. 1). A study in 2008 considers the correlation between increasing interactive digital media usage and unhealthful behaviors (Escobar-Chaves & Anderson, 2008). The researchers looked at five major areas of risky behavior. These include obesity, smoking, drinking, violence, and early sexual activity. These categories were chosen because the Center for Disease Control and Prevention (CDC) has identified these areas among the activities that “contribute to the leading causes of death and disability in the United States among adults and youth” (Escobar-Chaves & Anderson, 2008, p. 148).

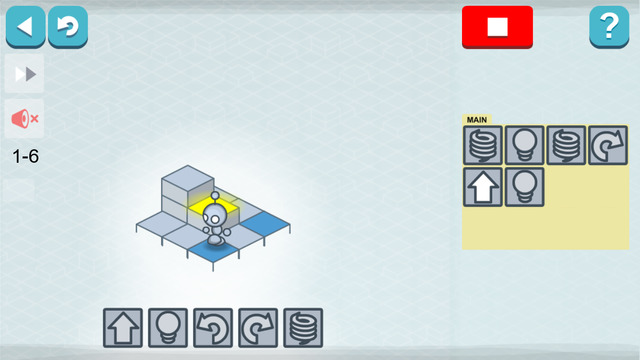
While most researches indicate negative effects of video games–primarily the violent and roleplaying games, there are genres that seem to give light to positive impact on our youth today. Some of these genres are educational, puzzle, logical and others that stimulate learning and personality development through the use of video games.

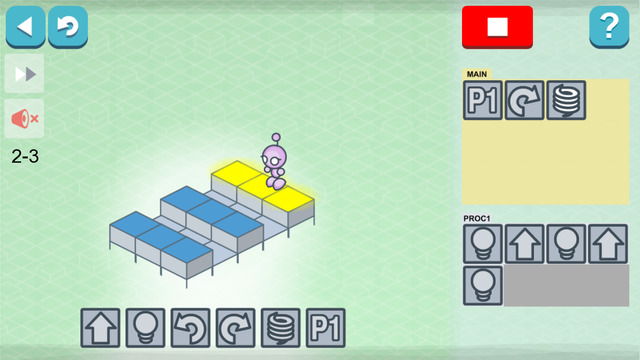
Puzzle games are one of the most played games in smartphones. Some examples are 2048, Tetris, The Room, Candy Crush Saga, Cut the Rope, Lightbot and etc. One of a good example of puzzle game is Lightbot. Lightbot is a puzzler that teaches kids concepts used in computer programming. An undergraduate student who’s been coding since he was a kid himself developed this programming app for kids. Brief instructions are included at the beginning of each level - what you need to know when you need to know it. The level of challenge ramps up very quickly, making this best suited for older kids and teens.

Educational video games are those that aid learners to learn specific areas in their academe. Best examples of these are the mathematical, vocabulary and programming games. Educational video games are a properly laid informative set of data contained in a specific scope of a single academe. It is properly guided path of succeeding information that aims to aid students/learners with a more comprehensive educational experience.

Lightbot Screenshots:







* 1. **Objective of the Study**
     1. **General Objective**

**To develop a virtual game that would will stimulate interest and help learners to engage basic technical computer programming lessons.**

The proponents aim to help learners to grasp the ideas of basic technical computer programming lessons through the use of video games. Through the use of a video game that is guided by specific lesson plan; players will be introduced to programming concepts, which will serve as skills that could be used by users to pass different stages. The video game Geek vs. Bugs is also designed to enhance problem-solving skills of gamers.

* + 1. **Specific Objectives**

**The study aims to provide a guided opportunity of learning in forms of entertaining virtual games.**

Through the use of video games, the proponents aim to develop an environment for learners where they would be introduced to the basic concepts of computer programming. The game flow will be guided by specific lessons, which will lay a path of learning for the gamers.

**To develop a game that would explain technical computer programming concepts into a practical, more understandable idea for learners to easily grasp.**

 Geeks VS. Bugs is a virtual game that will be used as a channel of education. The game will illustrate the basic computer programming concepts in layman’s term as game elements.

The game will be guided by a lesson plan that will be influenced by different curriculums for learning different programming languages. This lesson plan will lay a guided path of basic programming concept learning.

**To provide a tool that would allow incoming computer major student to gain prior knowledge of basic computer programming concepts before entering any colleges or institutes.**

Incoming computer programming major students will be able to acquire basic knowledge about what they’re pursuing. They will be able to prepare for what’s coming for them in college through the guided education method of Geeks VS Bugs. This tool could also be a bridge to those young people who have undiscovered skills in computer programming due to lack of exposure. And by having an experience of being introduced to basic concepts of computer programming, some players might develop an interest with software development.

* + 1. **Scope and Limitations**

**Scopes**

**Basic Computer Programming Concepts**

The basic computer programming concepts will guide parts of the virtual game such as game gameplay, tools, story line and game flow. This method will be used to introduce basic computer programming concepts to player through

**Game Story**

Game’s flow will be run through the story that was inspired by the discovery of the first bug that has ever been encountered. This story will be narrated throughout the 6 different stages

**Game Modes:**

The game will be contained with two different modes of game, which are story, and tutorial mode.

**Logical Compilers**

This module will serve as a compiler for the commands that are combined by the gamer. It will allow the user to check for instructions that are not logically executable. Before executing, he will first have to check if there is any logical errors or conflict within the gamer’s chosen set of commands for the Nanobot—the character in the game. If the compiler found any error, alert messages will be prompted to inform the geek—the player of the game. These prompt messages will be exited by the gamer through ok button.

**Stages and Levels**

The game will be broken down into 4 stages, which are also broken down into 6 levels each. Each stage will have different maps, new basic computer programming concept for the gamers to learn and some other new. Stages will require a minimum number of medals acquired by the player from current and previous stage-levels. Levels will also vary with each of its difficulty.

**Saving Game Status In Local Database**

User/Player will be allowed to save the current data of their game. This module will allow players to create another game without overwriting the previous records. The game will allow a maximum of 4 saved games that could be overwritten from time-to-time by the player.

**High Scores/Medal**

The game will be able to rate the player’s performance through recording their scores. Scores will be based on the path the Nanobot was commanded to take in order to accomplish each level requirement. These awards will vary on the difference between the shortest path possible that was indicated by the game for the current level and the length of the path taken by the player. Each level will be able to award a maximum of 3 medals.

**Artificial Intelligence**

Artificial intelligence could be experienced in the game through stage 4. It will be applied to the bugs in the last stage in a way that they will be fighting with the gamer. It will decide when to attack and when to flee.

**Shortest Path Analysis**

The game will be loaded by a shortest path analysis engine that will enable the application to compute the minimum possible steps that the Nanobot is able to take towards it’s goal. This engine will provide the game a dynamic standard of computing scores.

**Augmented Reality**

Augmented Reality will be added in the game at stage 4. This will allow the player to experience new environment within the game. This technology will be applied as the environment where the bugs and Nanobot would fight.

**Dynamic Allocation of Bugs and Tools Within the Map**

The game has will be created with the ability to allocate bugs and tools in dynamic points within the map. This will give user a larger possibility of different experience even with repeating a same level.

**Limitations**

**Multiplayer**

Multiplayer will not be included in the game