http://www.educacioneningenieria.org

Video games to learn programming

Jorge Iván Fuentes-Rosado

Academia de Ingeniería en Sistemas, Instituto Tecnológico Superior Progreso, Progreso, Yucatán, México. jfuentes@itsprogreso.edu.mx

Abstract— In this paper I present three environments for learning programming. These platforms utilize puzzles to engage users to solve them, making the users to code the avatar's "intelligence" or in some cases, coding the video game itself. Some platforms create links between the industry and their users, allowing the employers to see the performance and offering them job positions as software developer.

Keywords— video games; learning management systems; software developers.

Recibido: 29 de marzo de 2019. Revisado: 26 de junio de 2019. Aceptado: 17 de Julio de 2019.

Video juegos para aprender a programar

Resumen— En este artículo se presentan tres plataformas de video juegos cuyo objetivo en enseñar a los infantes a programar. Estas plataformas emplean actividades que se basan en la solución de acertijos, donde la usuaria puede programar la "inteligencia" del avatar en el juego, hasta la lógica del video juego. De acuerdo al tipo de usuario, las plataformas ofrecen a las empresas la posibilidad de ver el rendimiento de los usuarios para ofrecerles empleo como desarrolladores de software.

Palabras Clave— video juegos; plataforma educativa; aprender a programar.

1. Introduction

The mind behind computer application is a software developer. Software developers are magicians who tell computer what to do. A software developer analyzes user's needs, design, develop and test a solution to those needs using the best engineering process to ensure quality.

According to U.S. Department of Labor [1] software developers' jobs will grow faster than any other occupation in US. Department of Labor predicts that all industries will need and use a software solution [1]. Society is changing in this century, and jobs are doing too. Society is no longer waiting for consumers but also producing them. Business are creating new needs to be satisfied than before we, as individuals, did not have.

Gaming is a medium that encourages interaction, discovery, and trial-and-error. A good game challenges the player to master skills over time, which is the same critical process students go through as they learn (Starr, 2018). Games excel at rewarding "productive struggle" the kind of struggle that results in learning, that is engaging and motivating, not tedious [2].

1.2. Software developers skills

There are several skills required to be a software developer. These skills are attention to detail, Logical and Structured Thinking, solving problem and technical skills like knowing a programming language, data structures knowledge among others.

Logical and Structured Thinking is one of the mainly skills needed to this job. This skill is related with the ability to solve problems in an organized way. We solve problems every day, like making a sandwich, but if we take time to think what are all the steps included in this task we can be surprised, but what if we need to tell a computer to make a sandwich, we need to tell all the steps and we have to be sure all steps are covered. When we are coding a new software solution we need to tell all steps to solve a problem to the computer. The computer will follow all the instructions without asking if it is right or not (This paper does not include new advances in Artificial Intelligence). Having a Logical Thinking and organized allow us to think how the steps will be executed, we do not put mayo if we do not have the bread first. If we want to delete a record in a database we need to be sure we have the record we want to delete and be sure that record exists in the database.

There is a saying, "The devil is on the details". Attention to detail is another skill we need to have or develop in order to be a successful software developer. Let's retake the sandwich example, if we ignore expiration date of the ingredients this sandwich could be ruined. We do not want to ruin a good software solution because we forget that a division can't be done if the divisor is zero. We need to think in most of the ways this software can be used or fail and try to catch or block those incorrect paths.

Logical, Structured thinking and Attention to details are skills needed to solve problems. Solving a problem in software engineering is different to other fields. When we are coding most of the time we are the one creating the algorithms to be executed by the computer and that algorithm should solve the problem. Even though there is something called software engineering, that is the process of applying engineering into software, when we are actually dealing in the routine to solve something we are the ones who are writing detailed instructions for the computer.

Technical skills like data structures, databases or a programming language are important, these skills can be developed studying a computer science curriculum during

Como citar este artículo: Fuentes-Rosado, J.I., Video games to learn programming. Educación en Ingeniería, 14(28), pp. 119-123, Marzo - Julio de 2019.

Revista Educación en Ingeniería, 14(28), pp. 119-123, Marzo - Julio de 2019, Bogotá. ISSN 1900-8260
DOI: http://doi.org/10.26507/rei.v14n28.1006

college. Industries are not worried about any particular programming language because if a person has logical and structured thinking, attention to details, data structure, some data base knowledge and knowledge in any programming language it is easy to move or learn a new one.

1.2. Video Games and skills development

Video Games have been on the sight of many researches either way to talk about how bad are or how they can be used to develop some skills. There are video games or simulations to help surgeons to improve their abilities on surgeries [3], or rescue training [4].

Psychology and Education researchers have identified some skills that can be developed or improved for playing video games. Video Games can reinforce executive functions such as planning, organization, solving problems, working memory and decision making [5]. Expert Gamers can track moving objects more precisely and perform better in task with visual-short memory [6]. According to Dye et. al. video game play improves attentional resources, allowing gamers to better allocate their attention across both space and time [7]

Planning helps to think about something before doing it, complete a project on time, do things step-by-step. Kulman in [8] define a set of skills related to planning like Setting goals, there are many video games that make gamers to figure out what do in order to achieve something in the future [8]. For example, *Age of the Empires*, players must analyze situations to think what is best for their cities, like, getting more metal, wood, attack other cities, etc. Another skill that is part of planning is sequencing and ordering skills, for example, in *Where is my water*, the gamer has to come up with the best steps to make the water run into *swampy's* pipes. Other skill defined by Kulman is understanding both short-term and long-term goals, in *Tetris*, a gamer could be thinking in removing one line but if she organizes blocks in a way she could remove several lines at the same time. Prediction and foresight. Part of planning is being able to estimate possible outcomes. While no one can predict the future, the ability to identify the most important issues helps in knowing what happens next. Strategy games such as *StarCraft II* practices this, like when choosing what upgrades to buy first, deciding how to reinforce their defenses or estimating the angle from which the enemy will attack [8].

Decision Making refers to choose among a list of options one. Bluelow mentioned in her work people who have played several hours before a gambling game they tend to be more impulsive in their decision-making process, they prefer small instant reward than bigger and distant rewards from those people without playing video games before gambling [5]

2. Gamifying projects to learn programming

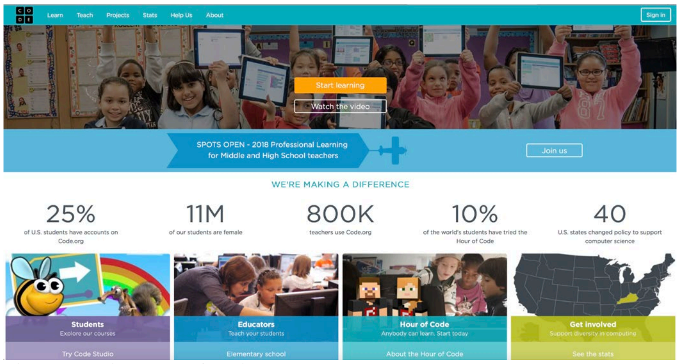
As I portrait in the previous sections there is an overlap between the skills needed to develop code and the skill than can be developed by playing video games. This theoretical correlation has led to create several community projects related to play video games while programming. In this section I review three projects: One hour of code initiative, Code Combat and CodinGame.

2.1. One hour of code initiative

Code.org is a non-profit organization advocating to teach computer science among children and underrepresented minorities. According to its website, from the population of students they have, 45% are girls, 48% belongs to a underrepresented minority and 49% are on free or reduce meal [9]. They proclaim: "Code.org increases diversity in computer science by reaching students of all backgrounds where they are — at their skill-level, in their schools, and in ways that inspire them to keep learning."

They offered a set of courses organized in three categories, Grades K-5, Grades 6-12 and University+. For Grades k-5 they have two courses, Course A: Introduction to Computer Science for Pre-readers, for students between 4 and 7 years old, Course B: similar to the previous course but with more variety for older students for students between 5 and 8 years old, Course C: Learn the basics of computer science and create your own art, stories, and games, for students between 6 and 10 years old. Course D: Quickly cover concepts from Course C, then go further with algorithms, nested loops, conditionals and more, for students between 7 and 11, Course E: Quickly cover concepts in Course C &D and the go further with functions, for students between 8 and 12, and the last course, Course F: Learn all the concepts in Computer science Fundamentals and create your own art, story or game, for students between 9 and 13 years old. They also offer two express courses, for kindergarten and first grade students (4-8 years old) and a second one for older students (9-18 years old). In case of Grades 6-12, they are mainly focused in Javascript, HTML and CSS, and they have three categories, "App Lap", "Game Lap" and "Web Lap". App Lap, allows students to code with blocks or Javascript and they have the possibility to share their work. Game Lab, as they say, it is a more challenging environment where students can make animations and game with characters. Web Lab is for students who want to create websites with HTML and CSS. For University+ code does not offer any activities so far, but they suggest specific online free courses in different universities.

Code.org offers all created material, like tutorials, videos under Creative Commons license. All material can be accessed and used freely on non-commercial purposes.



WE'RE MAKING A DIFFERENCE

25% of U.S. students have accounts on Code.org

11M of our students are female

800K teachers use Code.org

10% of the world's students have tried the Hour of Code

40 U.S. states changed policy to support computer science

Figure 1. Index page of Code.org
Source: [9]

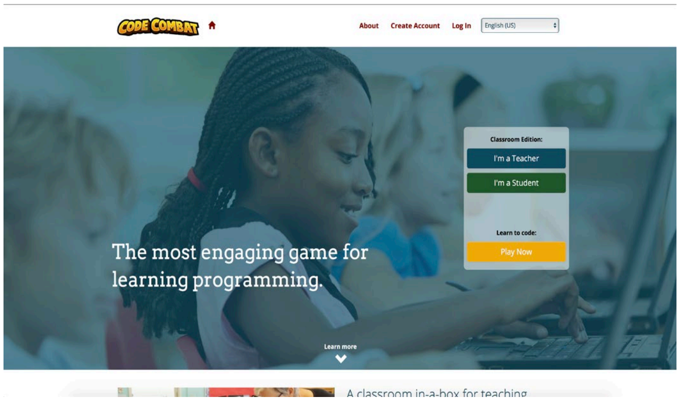
2.2. Code combat

"Our mission: make programming accessible to every student on Earth" [2]. I wanted to start the section of *Code Combat* with their mission, because it provides a clear idea about what I write. *Code Combat* is a community project with hundreds of players and volunteering to create levels, testing, fixing bugs and translate in order to achieve the goal. *Code Combat* has been translated to 50 languages so far. Keeping the idea of community this project is open source, code and art can be downloaded from their github repository. *Code Combat* has two programming languages to learn and practice so far, Python and Javascript.

Code Combat, at codecombat.com, allows three different users, teachers, students and unregistered participants. As a Teacher, we can create a class for students. We can choose the programming skill's level of the students in a class. This level goes from no experienced to advanced. It is optional to write the range of age of the students. Once the class is created according the platform provides you with a set of activities for the students enrolled in the class. As a student, we can enroll into a specific class with a specific curriculum.

Code Combat offers three curriculums organized by levels. Computer science, Web development and Game development. Computer Science offers six different levels that goes from learning the basic concepts of a language, to more advanced algorithms, data structures, and computation. Web development has two levels starts with an introduction to HTML and CSS to finish writing scripts to make interactive webpages. Game development has three levels to learn Basic Syntax, Arguments, Place game objects, Construct mazes, Create a playable, sharable game projects.

The game start choosing an avatar for the game. It provides with a puzzle to solve and a list of methods that can be used to solve it. In the right of the view we can see the puzzle to solve, in the center of the view a menu with the methods and on the left the IDE to write code. In the bottom part of the IDE there is a button run to execute the code and see how the puzzle is solved with the instructions/code we can write.



The most engaging game for learning programming.

A classroom in-a-box for teaching

Figure 2. Index page of CodeCombat
Source: [10]

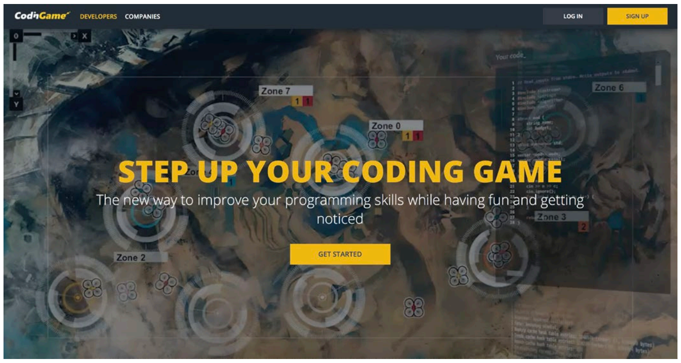
2.3. CodinGame

"It has never been so exciting to be a programmer", [11]. As a programmer I can only agree with this quote from one of the creators of *CodinGame*. Nowadays being a programmer represents not only a job, but also a hobby. As I presented in previous sections, programming requires many skills that can be used in our daily life including in work, but sometimes at the beginning of this career it can be overwhelming. Programming languages and paradigms move so fast and keeping track of them is hard to do. *CodinGame* tries to help to programmers no matter the level to improve their skills presenting an easy way to do it.

CodinGame is an online platform to code on real time video games. The platform presents a history and explain the input you will have and the output they expect, if you can solve the puzzle they ask you to write actual and real code in a programming language you can see how your code affects the output of the video game. *CodinGame* has 26 programming languages to be used, like python, c, c++, java, go, JavaScript, among others. It offers a Linux environment with a 64bit multi-core architecture with 768MB of memory.

When you open www.codingame.com you see a set of videos and code inviting you to step up your coding game. The platform provides a go through tutorial, where it shows the general structure of a game. A game o puzzle to solve comes with a background story to set the mood for the game. It provides the necessary information to solve your mission, examples of this information are: the input, expected output format, and sometimes, it gives you variable names to be used in your code. You can select a programming language to use to code your answer. An experienced programmer could think in the advantages of using a particular language in particular missions, however, all the provided compilers are of general purpose. Once you have selected the language to use, it is time to code, the platform provides you with a method signature for the function you have to code, with all the needed parameters, you can't change any parameters. After finishing you code, you have a set of test cases to verify how your solution works in different environments of the mission. This is when magic happens, you can see you code acting in the video game. The last part, after clearing all test cases, is submitting your code to keep record of your progress.

When you log into the system, it offers you four options: Practice, Compete, Contribute and Learn. It provides a general



STEP UP YOUR CODING GAME

The new way to improve your programming skills while having fun and getting noticed

Figure 3. Index Page of CodinGame
Source: [11]

message board where you can chat with other users. Practice view is organized in six sections, easy, medium, hard, very hard, community puzzles, and machine learning. Easy puzzles are meant to help users to remember or learn basic programming concepts like loops, hash tables, conditions, arrays, string management, etc. Medium Puzzles help you with more complex data structures like list, graphs, trees, and some specific algorithms like BFS, DFS, etc. In Hard puzzles, missions require a better understating of algorithms and specific complex solutions like path finding, back tracking, greedy algorithm, etc. In very Hard Problems you can find problems related to cryptography, binary search, recursion, memoization, dynamic programming, pattern recognition, etc. Community puzzles are created and shared by others users. The last section is Machine Learning, it requires tensor flow library.

CodinGame is an online platform to help programmers no matter the level of expertise to practice, learn and have fun at the same time. Time to time *CodinGame* and its community open contest to participate among other programmers into missions. These missions can be coding bots to protect or attack your adversaries. It can be solutions that requires your bot to optimize or find the shortest path before other users for example. *CodinGame* contests are online programming competitions, where software developers / programmers from all over the world can compete for fun, or to get in contact with companies they like and that are recruiting. *CodinGame* consist in developing computer programs to solve complex problems in a limited time.

The goal of the platform is not only to provide a funny way to learn, practice, and improve programming skills but also to help participants and employers to contact each other for consideration. Employers can see the performance of the participants they want and communicate with them for further interviews if participant wants it. *CodinGame* provides a showcase for programmers no matter they level of education. It is a way to close the bridge between employers and candidates.

3. Analysis and comparison

I presented three platforms for solving video games puzzles. Three of them provide puzzles and a run button to see how the written code is affecting the video game. The Three of them are free to use and have an organized way to use it. However, *Code* and *Code Combat* are focused into having a school curriculum and can be used in a computer science class, while *CodinGaming* is an open platform focused more in programmer that might not attend to school.

CodinGame offers more programming languages of the three, with 26 compilers available on its core. *Code Combat* offered 2 languages, python and javascript while *Code* offers drag and drog programming for kids who do not know how to write.

CodiGame has an extra aim, that is closing the gap between employers and employees by having the possibility to have the user performance while *Code* and *Code Combat* aim is to teach computer science to kids and students. This difference is show on the views, while *CodinGame* has a dark pallet of colors as shown in Fig. 3. In the other hand *Code* and *Code Combat* have a clear one as we can see on Figs. 1 and 2 respectively.

4. Conclusions

Software development skills are a must have requirement in most job positions currently. Not only because It allows people to write their own scripts to solve problems but also programming create a new way to solve problems.

Nowadays It is easy to have access to best universities courses online, but sometimes having a class is not enough. The initiative to create video games to improve coding skills create this agency effect on interested people on programming.

The platforms I presented here have a clear goal, to help people to learn how to program and to improve their skills if they have. However, If the reader is interested in practicing programming for interviews, there are other kind of sites where you can practice and test her skills with real or alike interview questions. These websites like, https://coderbyte.com/, https://leetcode.com, https://codefights.com offers a set of questions with answers and an online compiler to test the solution.

Bibliography

[1] U.S. Department of Labor, Occupational Outlook Handbook, Software Developers, April 2018. [Online]. Available at: https://www.bls.gov/ooh/computer-and-information-technology/software-developers.htm.

[2] Starr, M., CodeCombat: learn to code through dungeon crawling, April 2018. [Online]. Available at: https://www.cnet.com.au/code-combat-teaches-you-to-code-through-dungeon-crawling-339346652.htm.

[3] Rosser, J.C., Lynch, P.J., Cuddihy, L., Gentile, D.A., Klonsky, J. and Merrell, R., The impact of video games on training surgeons in the 21st century, Archives of surgery, American Medical Association, 142, pp. 181-186, 2007. DOI: 10.1001/archsurg.142.2.181.

[4] Fanfarov'a, A. and Marivs, L., Simulation tool for fire and rescue services, Procedia Engineering, 192, pp. 160-165, 2017. DOI: 10.1016/j.proeng.2017.06.028.

[5] Buelow, M.T., Okdie, B.M. and Cooper, A.B., The influence of video games on executive functions in college students, Computers in Human Behavior, 45, pp. 228-234, 2015. DOI: 10.1016/j.chb.2014.12.029.

[6] Boot, W.R., Kramer, A.F., Simons, D.J., Fabiani, M. and Gratton, G., The effects of video game playing on attention, memory, and executive contro, Acta Psychologica, pp. 387-398, 2008. DOI: 10.1016/j.actpsy.2008.09.005.

[7] Dye, M., Green, C. and Bavelier, D., The development of attention skills in action video game players, Neuropsychologia, 47, pp. 1780-1789, 2009. DOI: 10.1016/j.neuropsychologia.2009.02.002.

[8] Kulman, R., Five ways that video games teach planning skills, Learning Works for Kids, LearningWorks, [online]. Available at: http://learningworksforkids.com/2012/05/five-ways-that-video-games-teach-planning-skills/.

[9] Code, Code, Code, 04 2018. [Online]. Available at: code.org. [Last access: 04 2018].

[10] Code Combat, Code Combat, [Online]. Available at: https://codecombat.com. [Last access: 1 04 2018].

[11] CodinGame, Play with programming, [Online]. Available at: https://www.codingame.com. [Last access: April 2018].

[12] Griffiths, M.D., The educational benefits of videogames, Education and Health - Schools Health Education Unit, 20, pp. 47-51, 2002.

[13] Mack, D.J., Wiesmann, H. and Ilg, U.J., Video game players show higher performance but no difference in speed of attention shifts, Acta Psychologica, 169, pp. 11-19, 2016. DOI:10.1016/j.actpsy.2016.05.001

J.I. Fuentes-Rosado, recibió el título de Lic. en Ciencias de la Computación en 2006 de la Universidad Autónoma de Yucatán, México, el título de MSc. en Ciencias con especialidad en Sistemas Inteligentes en 2008 del Instituto Tecnológico de Estudios Superiores de Monterrey, México. Inició sus labores docentes en el Instituto Tecnológico Superior Progreso en agosto de 2010 y es profesor de tiempo completo de asignatura C desde 2012. Fue líder de la línea de investigación "Desarrollo de Tecnologías de la Información y Comunicación". Actualmente se encuentra trabajando en la obtención del grado de Doctor en Ciencias de la Computación, en Texas A&M University, Estados Unidos de América. Sus intereses investigativos incluyen: aprendizaje máquina para procesos de optimización, innovación educativa, inteligencia artificial, bioestadística, bioinformática, machine learning.
ORCID: 0000-0002-3079-2323