JDCS
디지털콘텐츠학회논문지
Journal of Digital Contents Society
Vol. 19, No. 12, pp. 2297-2304, Dec. 2018

컴퓨팅 사고력 강화를 위한 소프트웨어 교육과 비즈니스 모델

박재흥 · 박순화 · 서영건[*]
경상대학교 컴퓨터과학과, 대학원 문화융복합학과

Software Education and Business Model for Enforcing the Computational Thinking Ability

Jae Heung Park · Sunhwa Park · Yeong Geon Seo[*]
Dept. of Computer Science and CCBM, Graduate School, Gyeongsang Nat'l University, Gyeongnam 52828, Korea

[요 약]

본 연구에서는 소프트웨어 교육의 국제적 현상을 분석하고 현재 소프트웨어 교육에서 강화되어야 할 요소를 분석한다. 또한 컴퓨팅 사고력을 향상시키고 비즈니스 모델 캔버스로 제시하는 소프트웨어 교육 모델을 제시한다. 이 모델은 단순한 교육에서만이 아니라 교육의 결과물인 스마트 북을 새로운 비즈니스 컨텐츠로 유통시킬 수 있게 함으로서 교육의 효과를 극대화 시키는 것이 목적이다. 제안된 스토리텔링 코딩 교육은 익숙한 이야기로 다양한 프로그램 도구를 사용하여 대화식 스마트 북 형식으로 교육 결과를 만드는 것이다. 먼저 동화를 부분으로 분해하고 패턴, 경향 및 규칙을 숨기고 중심 아이디어 또는 개념으로 추상화 한 다음, 마지막으로 문제를 해결하기 위해 코딩 알고리즘을 적용한다. 언플러그드 컴퓨팅 및 알고리즘, 블록 기반 교육용 프로그래밍 언어, 물리적 컴퓨팅, C 및 파이썬과 같은 텍스트 기반 프로그래밍 언어를 혼합하여 플립 학습을 기반으로 한 점진적 융합 교육 방법을 제안한다.

[Abstract]

In this study, we examine the international situation of software education and analyze the factors that should be reinforced in current software education. We also present a software education model that enhances computational thinking and presents a business model canvas. This model can maximize the effect of education by not only staying in the education but also by distributing the result of education into the new business contents. The proposed storytelling coding education makes the result of education in the form of an interactive smart book by using various program tools with a familiar story. First, it decomposes the assimilation into parts, finds patterns, tendencies, and regularities hidden in them, abstracts them with a central idea or concept, and finally applies a coding algorithm to solve the problem. We propose a progressive convergence training method by blending a text-based programming language such as unplugged computing and algorithm, blocked based educational programming language, physical computing, C and Python.

색인어 : 소프트웨어 교육, 컴퓨팅 사고, 스토리텔링, 비즈니스 모델, 언플러그드 러닝
Key word : Software education, Computational thinking, Storytelling, Business model, Unplugged learning

http://dx.doi.org/10.9728/dcs.2018.19.12.2297

Received 05 September 2018; **Revised** 17 September 2018
Accepted 23 December 2018

***Corresponding Author; Yeong Geon Seo**

Tel: +82-55-772-1392
E-mail: young@gnu.ac.kr

pISSN: 1598-2009 eISSN: 2287-738X

Ⅰ. Introduction

Today, the development of information and communication technology(ICT) is creating new trends such as Internet of Objects (IoT), Artificial Intelligence(AI), big data and so on, making a big difference in our lifestyle. This has created a new environment and business with a destructive and innovative impact on existing business systems and stakeholders. The so-called fourth industrial revolution began. Klaus Schwab, President of the World Economic Forum at Davos Forum(2016), insists that the paradigm that has a major impact on society and industry over the next 50 years is the fourth industrial revolution[1]. The fourth industrial revolution meant a fusion of manufacturing and ICT in Germany's 'Industry 4.0 Strategy'. Recently, however, the concept of a broader concept of changing the lifestyle of mankind by intelligently connecting and fusing objects and humans based on platforms[2]. The commercialization of autonomous driving cars based on AI, robot advisors, and chatbots, which is hot in 2016, shows that the 4th industrial revolution is progressing gradually in society as a whole.

The world is paying attention to software as a core technology that can lead the market order in the fourth industrial revolution era. In the field of education, the changes in information science and software education are clearly seen in many countries around the world, reflecting the demands of this age. In order to grow into a core talent that leads society in the era of fusion, it is important to develop the ability to solve the problems creatively and efficiently by learning the computational thinking ability that penetrates the core of convergence[3]. In the UK, students will be required to study computing from age 5 to 16 years from 2014, and India will receive one to two hours of software training a week for six years in elementary school. In addition, the countries such as the US, Japan, Estonia, and Israel are competing with each other to make software education mandatory[4-7]. In Korea, as well, software education will be applied to the 5th grade elementary curriculum from 2018 in 2015. In 2019, it will be adopted as regular curriculum, and all students will receive software education as compulsory education for 17 hours for elementary school and 34 hours for middle school. Although the software capability is emphasized as the core competence required by the current world, the Korean government has begun to organize the regular courses in only one year after the declaration of software-centered society in 2014, The difficulty of the field is serious. There is a shortage of teachers, educational programs and dioceses to teach[8].

Software education is a method of teaching computational thinking based on computer science and programming principles, which are fundamental concepts of computing. Computational thinking refers to the overall process of creating a solution to a problem when it occurs and using that method to produce the desired result. Of course, there may be various solutions for the same problem, and various procedural methods to solve the problem are called algorithms in software education. The main purpose of software education is to develop computational thinking ability including various algorithms to solve the problem, and further to promote fusion thinking. Thus, the goal of our software education is not education of simple skills or software developers but education of creative problem solving ability that fosters informational thinking ability as basic competency of the fourth industrial revolution era.

In the actual education field, most of the curriculum for elementary and high school software education curriculum is composed of block based educational programming language(EPL) software education such as Scratch, Entry, App Inventory and E-sense board. Also as an essential culture subject of software leading university, some of them are being taught in the course of “computational thinking and software coding”[8]. This study investigates the international situation of software education and analyzes the factors that should be reinforced in our education. And we propose storytelling coding education to enhance computational thinking ability. We aim to create a book-like form that allows anyone to familiarize with a fairy tale or story through abstraction, algorithm, and automation process of the computational thinking thought process, and to make the existing textbooks interactive with multimedia elements. A fairy tale or a story that combines story and coding as a fusion factor can double the humanistic imagination because it can create various creative stories depending on the result of selection or game. This educational model can maximize the effect of education by not only staying in the education but also by distributing the result of education into the new business contents.

The proposed learning model is divided into the different levels of difficulty, and the process of reviewing each team's story in the off-line education after basic prerequisite learning through online using the flipped learning method is completed. This process is expected to contribute greatly to the achievement of team collaboration ability and communication result. Storytelling coding education leads to the development of natural storytelling, which is combined with the context of the story, which enables deepening the progressive convergence education. In this paper, we present a software education model that integrates human factors called storytelling and coding to enhance the software capability, ie, computational thinking, and presents a business model for commercialization. It is learning to write a story to

Table 1. Computational Thinking Models

Researcher (year)	Wing(2006,2008)	CSTA(2011)	Barr& Stephenson(2011)	KeyStage3(2016)	Google(2016)
Components of Computational Thinking	Analysis	Data Acquisition	Data Acquisition	Analysis	Data Acquisition
	Conceptualization	Data Analysis	Data Analysis	Pattern Recognition	Data Analysis
	Abstraction	Data Representation	Data Presentation	Abstraction	Pattern Recognition
	Recursive Thinking	Problem Disassemble	Problem Disassemble	Algorithm	Data Representation
	Parallel Processing	Abstraction	Abstraction		Disassemble
		Algorithm and Procedure	Control Structure		Abstraction
		Automation	Algorithm and Procedure		Pattern Generalization
		Simulation	Analysis and Model Validation		Algorithm Design
		Parallelization	Automation		Automation
			Parallelization		Parallelization
			Test and Verification		Simulation
			Simulation		

make a computer to be a learner's favorite story. It is not a story as a means to learn a command of a specific program but the purpose of the story is to become familiar with the commands of the program and become a software education model.

Ⅱ. Related Studies

2-1 Software Education and Computational Thinking

The textbook definition of software education is defined as "education to cultivate talents to solve problems through computational thinking". The computational thinking here starts with computer scientists thinking about problem solving and the basic problem solving techniques associated with it. The programming process for solving everyday problems with software can be summarized as the followings.

1. Understanding and analyzing the problem accurately (Problem understanding and analysis)
2. Extracting the key elements from the problem, redefining the problem with the extracted elements (Abstraction and modeling)
3. Designing the methods and procedures for problem solving (Algorithm)
4. Writing a program so that the computer can do the work according to the designed method and procedures (Writing the source program) and confirm that the result is correct (Automation).
5. If the result is incorrect, find out where it went wrong in the previous step and correct the error (Error correction)
6. Repeating the remaining steps until the desired program is completed (Debugging)

The ability to perform these series of the processes is called computational thinking. The thinking model of it is shown in <Table 1>.

2-2 Foreign Software Education Cases

The United States educates the computer science subjects at Level 1, Level 2, and Level 3. In the first stage, they learn basic principles of information science through computational thinking, experience how much computer is used and important in everyday life, and link with various curriculum and induce learning and creation. In the second step, computational thinking is used as a tool for problem solving. They use communication, and collaboration, as well as the use of computational thinking in learning, creation, and inquiry. In the third stage, three subdivision processes are provided to deepen learning, solve real-world problems using computational thinking, and carry out the projects together to encourage collaboration and effective communication. The difference from Korea is that the curriculum includes community, global, and ethical influences. The collaboration is the most important thing in creating software because nobody can do programming by oneself.

No matter how skillful a programmer is, it is a reality that it is difficult to evolve and develop in the market with the value of a commodity. In other words, it is necessary to communicate with many people until one software is created. In some cases, meetings that coordinate and prioritize their work take longer to design and make the actual software code. Therefore, it is necessary to look community awareness in understanding digital world.

From september 2014, the UK announced a new national curriculum reorganization that would teach programming as an integral part of new computing subjects that replace traditional ICT education at all levels of education between the ages of 5 and 16 years. It consists of 4 steps and the computing subjects are included in the basic curriculum, and the students study computing for 11 years from elementary to middle school. The goals of the computing curriculum are largely captured by three computer science, information processing technologies, and digital literacy skills. It is especially noteworthy that it provides creation, reuse, revision, and modification of digital artifacts to improve them during the course of computing. We are going to take a look at expanding to the point of creating and modifying current softwares and providing them to users.

This process involves experiencing a series of processes where softwares are created and communicated to users to interact with real users and make further improvements. The UK understands technical safeguards on information protection and includes how to see how it is changing, in the field of ethics and personal information protection. Currently, it is managed by strictly limiting personal data protection act, but there is still much information available to software programmers. Therefore, educating the security consciousness and ethical awareness of information is of the utmost importance from the point of view of the providers of software, like the UK.

India taught coding in C++ and Java in high school and has already become a software powerhouse. Estonia, the hometown of Skype, the world's largest internet phone, has become a jurisdiction of coding education with mandatory coding education for all students since the 90s[7]. The students who have difficulty coding directly learn the coding principles using multicolored codes, and see the strong commitment of the Estonian government to draw one's attention to the world of coding. Since december 2001, China has set up a pilot institute for seminars at 35 universities nationwide including Tsinghua university, Beijing university and Harbin institute of technology[4]. In this way, the world is looking for the future of the fourth industrial revolution in education. In the era of the fourth industrial revolution, what is needed is a national education system capable of cultivating talented people who can analyze data and extract value[5].

2-3 Digital Storytelling and Flipped Learning

Stories have been important teaching and learning materials and methods since ancient times. In particular, fun, an element of story, can be a factor that allows learners to voluntarily participate in the learning process and immerse themselves in activities. The story that can become an important material in teaching and learning has been reborn as a combination of digital technology including various multimedia elements such as sound, video, animation, and game through ICT growth.

Flipped learning is a way to 'flip' an existing method. Before real class, lecture videos for the students to study online are provided, and the students can solve problems that the students can not solve and the more intensive learning activities are conducted through discussions with the peer learners or with the assistance of assistants and instructors. Like these, they can do it all by themselves. When all of them are together, it is the most efficient way to do activities that can be done together. This is efficient in time as well as streamlining. The passive learners of the class actively become the subjects of the active class, and the supervisor who is the control supervisor or the knowledge communicator plays the role of assisting the learning of the students. In other words, the role of the teacher in the education of the flipped learning method must become the facilitator.

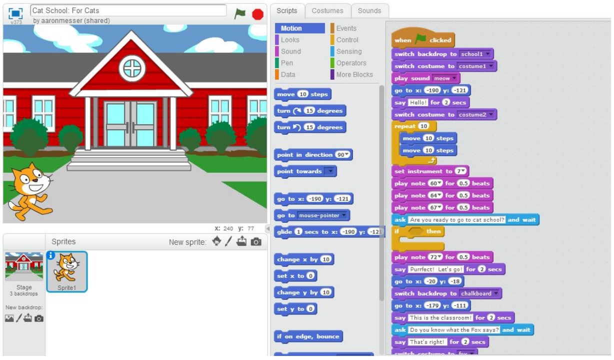
Ⅲ. Convergence of Computational Thinking and Business Model

In this chapter, we describe a gradual convergence education model to improve computational thinking ability based on the theoretical background. Here we refer to convergence in two aspects. The first is the fusion of story and coding, and the second is the fusion of various tools for coding. In other words, for coding, a single tool specification is avoided, and a variety of coding tools are fused to provide education.

3-1 Thinking and Coding Procedure

Fun is a factor that enables learners to voluntarily participate in the learning process and engage in activities. In this educational model, we intend to foster software capability, that is, computational thinking, by combining human factors called storytelling with coding. By using the familiar stories that users are already familiar with, it is possible to develop various creation according to the selection factors of the coding as well as the fun factors, thereby helping to improve the creativity. Thinking and coding procedure is a process of decomposing story into each individual process, recognizing repeated pattern of problem to be solved in each process, abstraction into main concept to reduce complexity of problem, and a step of automating the coding process by using a computer.

To link thinking and coding processes, it is necessary to format the story, which is called story coding. The story coding is a process of fusing the coding based on a funny story that everyone knows, such as the sun and the moon, the rabbit and turtle, and the baby rat. Creativity can be improved by expressing the action of a scene or an object realistically by animating the action of the scene or an object to express an interesting event in the storyline of the story, or by creating the ending of another creative story in an optional situation. In addition, in the process of conflict resolution and development, the game can be fused to move away from one-sided fairy tale and improve the thinking ability of computing by combining various methods in the form of a smart book that can interact with each other. Figure 1 is a part of story development and coding of “Cat school” using Scratch. Such a coding method is not a one-sided tool-based education, but rather a method that can progressively improve the thinking power of a computer by finding a portion to be coded by oneself and accumulating experience of problem solving.



```
when clicked
switch backdrop to school1
switch costume to costume1
play sound meow
go to x: -190 y: -121
say Hello! for 2 secs
switch costume to costume2
repeat 10
    move 10 steps
    move 10 steps
set instrument to 7
play note 60 for 0.5 beats
play note 64 for 0.5 beats
play note 67 for 0.5 beats
ask Are you ready to go to cat school? and wait
if then
play note 72 for 0.5 beats
say Purrfect! Let's go! for 2 secs
go to x: -20 y: -18
switch backdrop to chalkboard
go to x: -179 y: -111
say This is the classroom! for 2 secs
ask Do you know what the Fox says? and wait
say That's right! for 2 secs
```

Fig. 1. Story development using Scratch

3-2 A Progressive Convergence of Coding Tools

The progressive convergence education model of the program tool is gradual deepening by the process of 1~5 steps which fused each independent tool. Figure 2 is a list of the programs that can be used for convergence. For progressive convergence there are so many methods like the followings. Unplugged computing and algorithm for first step, relatively easy visual(block) programming for education, unplugged programming and algorithm for second step, visual programming, unplugged and physical programming for third step, visual and text programming for fourth step, and real and general development programming language, text and visual programming, unplugged programming, algorithm and physical programming for fifth step. These models give flexibility by allowing users to access various languages or to select and integrate a specific language they are familiar with. Also, the advantages and disadvantages of each language can be learned by themselves and the computational thinking can be improved. The result of education is to make digital storybooks. Therefore, this study uses this digital storybook as a business model to be discussed in the next chapter. This digital storybook creates an inflow customer of the platform. Many customer subscription can lead to advertising revenue, off-line training in the form of lectures, and support from key partners.

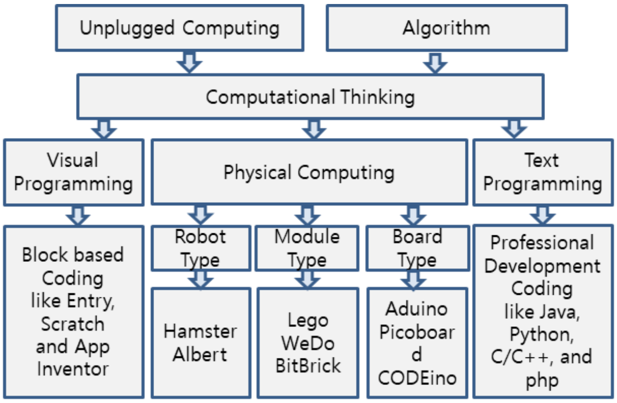


Fig. 2. Different tools for computational thinking

3-3 Business Model for Computational Thinking

In this paper, a software education business model using incremental convergence education to improve computational thinking power is shown in table 2 with 9 blocks proposed by [9].

- Customer Segments : To build an effective business model, we need to understand which customers the company needs to serve. The different types of customer segments are the followings. A person who wants to be a software education specialist or a coding educator, after-school teacher and who wants to produce digital storytelling contents exists as a productive customer and a consumer who wants to see digital storybook as a result of simple education. Learning of basic tool is based on learning through free video lectures provided by the platform.

- Value Propositions : They combine storytelling with software education to create easy and fun learning, and the results created in the course offer new business opportunities. The value propositions may be that quantitative ones are price and efficiency, and qualitative ones are overall customer experience and outcome.

- Channels : A company can deliver its value proposition to its

Table 2. Business Model for Computational Thinking

<table>
<tr>
<td rowspan="2">Key Partnerships :
– Coding experts training institutes
– National university
– Education support agency
– Fairy tale writer
– Character designer</td>
<td>Key Activities :
– Analysis of story development, coding elements and game element
– Storytelling coding
– Video production for software training</td>
<td rowspan="2">Value Propositions :
– Software program development for easy and funny computational thinking
– Support for creating content that can be a revenue model through software training</td>
<td>Customer Relationships :
– Face–to–face
– Internet
– Blended of them</td>
<td rowspan="2">Customer Segments:
– Those who want to be a software education specialist
– Those who want to create digital storytelling contents
– Those who want to view multimedia fairy tales for free
– Coding education job training
– Those who want to be after–school teachers</td>
</tr>
<tr>
<td>Key Resources :
– Those who have more than 5 years of edu–cational experience
– Computer related workforce</td>
<td>Channels :
– Multiple SNSs
– Off–line</td>
</tr>
<tr>
<td colspan="2">Cost Structures :
– Cost for making movies
– Cost of holding storytelling coding competition or camps</td>
<td colspan="3">Revenue Streams :
– Lecture fee
– Content, production commission income
– Story board kit sale</td>
</tr>
</table>

targeted customers through different channels such as web site, SNS channel and off-line. Effective channels deploy the company value propositions in a fast, efficient and cost-effective way.

- Customer Relationships : A good customer relationship can lead to the success of the business model. We use growth hacking, an online marketing technique that monitors and immediately reflects improvements in products and services through off-line lectures and on internet. Different forms of customer relationships include personal assistance, dedicated personal assistance, self service, automated services, communities and co-creation.

- Revenue Streams : This means the way a company makes income from each customer segment. Revenue through off-line lectures, content production import orders, and sales of storyboard kits is expected.

- Key Resources : This menas the resources that are necessary to create value for the customer. It has human, technical and intellectual resources with university lecturers and professors in the computer field.

- Key Activities : This is the most important activities in executing a company's value proposition. Story analysis, game element analysis, storytelling coding training, and software education video production are used to select familiar fairy tales and stories and apply them to coding education.

- Key Partnerships : To optimize operations and reduce risks of a business model, they usually cultivate buyer-supplier relationships so they can focus on their core activity. These are after-school coding experts, education support offices, fairy tale writers, part time lecturers and character designers.

- Cost Structures : This represents the most significant monetary results while operating under different business models. These are video production costs, personnel expenses, and the cost of holding a camp or storytelling coding competition.

Ⅳ. Conclusion

Recently, there has been a demand for teaching and learning that can develop core competencies such as information literacy[10], science and mathematical literacy through the reform of the curriculum to improve computational thinking ability. Software education has become an important turning point in securing Korea's competitiveness in the future. In this paper, we presented a computational thinking based coding education model and problem oriented software education based on the concepts and elements of computational thinking and instruction of software education management through the literature review. This educational model aims to cultivate convergent talents as a software education model that enhances computational thinking by integrating human factors called storytelling and coding. Also, we presented a business model canvas for the interactive digital storybook business. Future studies will further reinforce the ideas and suggestions that were not expressed in the model canvas, and will supplement the validation statistics for this model. We will also design and implement a real platform for future research.

References

[1] E. World and M. Magazine, “CEO column 4th Generation Industry Innovation, Industry 4.0,” Vol. 33, pp. 3233, 2016.

[2] S. Y. Park, “A study on the new Business Area of the Telecommunications Company to be the First Mover in the Fourth Industrial Revolution”, *Yonsei University Master Thesis*. 2016.

[3] T. Kim, “STEAM Education Program Based on Programming to Improve Computational Thinking Ability”, *Jeju National University Ph.D Thesis*, Korea. 2015.

[4] D. H. Yoon, "Software Coding, How do you Train in Foreign Countries?", http://www.newstomato .com/*ReadNews.aspx?no=618515*, 2016.

[5] J. H. Kim, “Estonia, Jumping into the GDP Budget as a Teaching Education.”, *Dong-a Newspaper*, 2017.

[6] S. G. Shin and Y. G. Bae, “Review of Software Education based on the Coding in Finland”, *Korea Information Education Society*, Vol. 19, No. 1, pp. 127-138, 2015.

[7] S. G. Shin and Y. G. Bae, “Study on the Implications about Curriculum Design through the Analysis of Software Education Policy in Estonia”, *Korea Information Education Society*, Vol. 19, No. 3, pp. 361-372, 2015.

[8] N. G. Lee, “Survey of Software Education in Software Leading Schools and Research for Improving Elementary School Software Education System”, *Sookmyung Women University, Graduate School of Education Master Thesis*, 2016.

[9] Alexander Osterwalder and Yves Pigneur, “Business Model Generation”, *Wiley*, 2010.

[10] J. H. Han, “Analysis on Creativity and Solving-Problem Ability with Hackathon-based Elementary SW Education”, *Journal of DCS,* Vol. 18, No. 5, pp. 995-1000, 2017.

Jae Heung Park

1989 : Dept. of Computer Science of Joongang Univ. (Ph. D. Degree)
1983-now : Professor in Gyeongsang National University(GNU), Dept. of Computer Science

Research Interests: Software Engineering, Computer Education. Medical Imaging

Sunhwa Park

1996 : Gyeongnam Science and Technology University of Computer Science(B.S Degree)
2000 : GNU of Computer Science(M.S Degree)
2015 : Doctoral Candidate in GNU, Graduate School of Cultural Convergence Business Model(On Ph.D Degree)
Research Interests: Multimedia, Medical Imaging and IT Cultural Convergence

Yeong Geon Seo

1987 : GNU, Dept. of Computer Science (B.S Degree)
1989-1992 : Software Developer in Trigem Computer Inc.
1997 : Soongsil University, Dept. of Computer Science (Ph.D Degree)
1997-now : Professor in GNU, Dept. of Computer Science and Graduate School of Cultural Convergence Business Model
2011-now : Member of Engineering Research Institute in GNU
Research Interests: Multimedia, Medical Imaging, Machine Learning and Computer Network

http://dx.doi.org/10.9728/dcs.2018.19.12.2297