

Research on Programming Teaching Strategy Based on Gamified Learning in Primary School

Ma Zong-bing

South China Normal University, Guangzhou, China

Programming teaching in primary and secondary schools of China has experienced a process from prosperity to decline, and now it is going to prosperous. With the development of artificial intelligence, the application of some gamified programming tools has accelerated the development of various primary programming curriculums. While it is still in its infancy. Due to the complex teaching situation and lacking effective teaching strategies, and thus the teaching effect cannot be guaranteed. This study proposes a primary school programming teaching strategy based on gamified learning and has practiced in a subsidiary foreign language school in Guangzhou for a term. It adopts the action research method to iteratively improve the strategy, supplemented by the questionnaire method to verify the effect. The study has shown that primary school students are generally very interested in programming courses, and the teaching strategy can encourage primary school students to learn programming knowledge and enhance their computational thinking ability.

Keywords: gamified learning, programming in primary, visual programming, teaching strategy, child programming

Introduction

In the 21st century, information technology reconstructs social productivity and production relations all the time. It can be said that the underlying systems of modern systems on which modern society operates are nested within algorithms. In the face of the future, the countries of the world pay special attention to the corresponding education system design, advocate the promotion of computational thinking to the level of national literacy, and open relevant courses in different sections to train students' future-oriented computational thinking ability (Ren, 2018). Countries, such as the United States and Britain have successively issued articles to emphasize the importance of cultivating students' computational thinking ability in the K-12 stage, and therefore set up computer programming education courses in primary and secondary schools to carry out relevant practices.

The New Media Alliance's "2017 Horizon Report (Basic Education Edition)" proposes "programming as a professional literacy", stating that "many educators believe that programming can help children understand how computers work and express themselves through structure and logic. Ideas, critical thinking, and success in an increasingly digital work environment". "Computer science is still one of the fastest growing industries in the world, and doing a good job of human resources is a key to future economic development. Programming literacy is in all industries. It becomes more and more necessary... Basic level programming literacy can help students build transferable skills. Programming can also develop problem-solving skills, collaboration skills,

and even interaction with games, robots, and animations. To cultivate students' interest in learning" (Beijing Open University Horizon Reporting Project Team et al., 2017). In fact, programming education is spreading around the world. Currently, more than 24 countries around the world have programmed programming into the basic education system.

Since 2012, Japan has begun to popularize programming in primary and secondary schools. Lesson:

1. Since 2013, when President Barack Obama called on all American students learning programming;
2. From 2018, the official annual allocation of 200 million US dollars to support the popularization of programming education, currently more than five million young people in the United States are learning to program; EU countries from the primary school to program programming into the general curriculum;
3. Since 2014, the United Kingdom will be programming as compulsory courses for students aged 5-16;
4. From 2017, Singapore has added programming subjects to the primary and secondary school exams; Australia has compulsory courses in the fifth grade.

Children's programming has grown nearly 10 abroad and more than 60 million children worldwide are receiving programming education. The feasibility of children's learning programming has been fully verified through years of development practice (Children's Programming Network, 2017).

In 2017, the State Council issued the "New Generation Artificial Intelligence Development Plan", which clearly stated that it is necessary to "extensively carry out artificial intelligence science activities", "implementing the National Intelligence Education Project, setting up artificial intelligence related courses in primary and secondary schools, gradually promoting programming education, and encouraging society. Power to participate in educational programming software is for learning and learning, game development, and promotion". Shandong Province has added programming content to the textbook of sixth grade in primary school. In 2017, Zhejiang Province would use information technology (including programming) as the college entrance examination candidate (7 out of 3). One of them has the same status as the traditional physical and chemical subjects. It can be seen that children's learning programming has become a trend, but because programming education is not in the K-12 stage, especially the programming education in primary school is in its infancy, and the difficulty of programming learning itself makes the real teaching situation especially complicated and lacks effective. The targeted teaching strategy does not guarantee the teaching effect. Therefore, exploring effective teaching methods and strategies for children's programming at this stage has become an important research direction.

Gamification-Enabled Children's Programming Learning

Visual Programming Dramatically Reduces Programming Learning Thresholds

Computer programming languages (programming languages) have evolved from machine language, assembly language, high-level languages, to non-procedural languages (4GL, encoding only requires "What to do", without describing the details of the algorithm) and the emergence of visual programming tools. The difficulty of programming learning is greatly reduced.

Therefore, as early as the 1980s, China had carried out programming teaching of BASIC language and LOGO language in primary and secondary schools. With the development of computer technology and the deepening of computer teaching research, people realize that primary and secondary school students do not need to fully master a programming language. Therefore, programming languages are gradually fading out of teaching (Wu, 2006). However, due to the graphical programming attributes of the LOGO language and the

teaching methods of teaching and learning, the threshold of learning programming is greatly reduced. This enables primary and secondary schools to carry out programming teaching activities. People have not forgotten the advantages and contributions of the LOGO language.

In recent years, with the rapid development of information technology, such as artificial intelligence, people are increasingly aware of the importance of computer programming. The state has issued a series of documents to promote computer programming in primary and secondary schools. In 2017, Zhejiang Province will be included in the College Entrance Examination. By April 2018, Nanjing and Tianjin have also included programming in the Entrance Examination Enrollment Program. Various localities have also developed various forms of computer programming education and practice in primary and secondary schools.

Gamification Programming Based on the Concept of Visual Programming Allows Children to Learn Programming Happily

An important reason for the development of programming education from primary to secondary in primary and secondary schools is the gamification programming based on the concept of visual programming. Throughout the successful domestic children's programming education case, the graphical and modular programming language based on the visual concept is the most popular among programming educators (Grover, & Pea, 2016). For children programming learners, it is very difficult and meaningless to master the basic terminology of programming. Therefore, by using graphical and modular programming concepts in gamification for children's programming learning, it is feasible to exercise their computational thinking path. In a general sense, gamification is the application of game design elements to non-gaming situations (Bao & Zhao, 2015) to achieve user engagement, user retention, and user loyalty (Wang, 2015).

Gamification learning is to cultivate goals and development and evaluation methods in the process of teaching design. Learn from the game's psychological characteristics and teaching strategies, design and select appropriate development tools, evaluation methods, and teaching strategies (Niu & Ren, 2006). Gamification teaching is based on the concept of the game's doubts, challenges, autonomy, etc., and hides the teaching objectives in the game activities. According to the characteristics of the learners and the teaching content, the corresponding gamification teaching strategies are adopted, so that the learners are relaxed. In the state, gain knowledge, improve skills, and cultivate sentiment from fun. The games here include video games and learning activities (J. L. Zhang & B. H. Zhang, 2013).

Gamification learning is a gamification-based way of learning, mainly including digital games and game activities. Therefore, gamification programming is to learn programming in a gamified way, often based on visual programming. In summary, there are three advantages to learning programming in a gamified manner: Firstly, the game incentive mechanism maintains learning motivation. The game has strong interactivity, allowing users to immerse themselves in exploration and trial and error, and the incentive mechanism of the game can enhance the learner's interest in learning and keep it. Secondly, the game can better train students' thinking. The feedback provided by the game is not a simple "right" or "wrong", but a more complete real strategy to help solve problems or improve the path to achieve goals, deepen the understanding of knowledge, and overcome more complex challenges. And thirdly, learn real programming through real games. In addition to the incentive mechanism, programming games often set the challenge to attract players. The learner must solve the problem and complete the challenge through the real programming intermediary, and gradually cultivate the learner's self-confidence and improve the ability to solve the problem.

Development and Application Status of Gamified Children's Programming

As early as 1968, Papert's child-oriented programming language LOGO language allowed students to learn programming by means of "drawing", which opened the curtain for children to learn programming in an entertaining way. Since then, children's programming gamification tools and applications have developed steadily, such as Squeak Etoys, HANDS, Alice, and many other tools (Wang et al., 2013).

In 2007, Scratch, a graphical open source programming software developed by MIT Multimedia Lab, was officially released. Scratch is widely used globally because of its simple operation, friendly interface, and the ability to fully utilize children's creativity and develop computational thinking skills. Children's gamification programming has entered a stage of rapid development. At this stage, the representative and widely used gamification programming tools in China are mainly Scratch developed by MIT, the first graphical application developed by Google engineers, Android application development tool App Inventor, open source electronic prototype platform Arduino, Apple Developed 3D gamification, visual command programming tools Swift Playgrounds, and online role-playing, related programming learning platform CodeCombat (China NetEase exclusive agent), and domestically developed Scratch developed by Shenzhen Point Cat Technology, more successful programming cat platform, etc. These tools are aimed at children of all ages and each has its own characteristics and focus.

According to the way of interacting with the machine, children's gamification (visualization) programming learning tools can be divided into two categories: graphical programming and imperative programming. Graphical programming is relatively easy. Simply dragging and combining the code blocks (graphic blocks) of the corresponding function can realize the students' creativity, represented by Scratch, App Inventor, and programming cat. For imperative programming, you need to input programming code. Learning is close to real programming, but with good gamification design and auxiliary support (such as 3D games), so that primary school students can also learn, Swift Playgrounds and CodeCombat are typical representatives. The application of these tools greatly reduces the difficulty of programming learning, and makes children have a strong interest in programming learning. At the same time, they have cultivated and trained their logic and computational thinking skills, making it possible to carry out programming education in the basic education stage.

Primary School Programming Teaching Strategy and Application Based on Gamification Learning

The Development Status of Gamification Primary School Programming Teaching Strategy

After years of practice, gamification-based programming learning has proven to improve the computational thinking ability of primary school students and even enhance students' participation in learning. Hsu and Wang's (2018) research confirms that an online puzzle game learning system (TGTS) can significantly enhance primary school students' thinking ability and problem-solving ability, and can improve students' willingness to participate and experience. Sáez-López, Román-González, and Vázquez-Cano (2016) used Scratch in five primary schools in Spain for two-year practical teaching, which proved that Scratch is effective in teaching programming and pointed out that programming learning through project-based learning can enhance students' learning fun and motivation. Choi, Y. Lee, and E. Lee's (2016) research indicates that algorithmic learning based on puzzles effectively enhances students' computational thinking.

Domestic related research and practice are mainly based on Scratch and App Inventor. In the form of starting classes, it is often embedded in information technology courses, "second class" or combined with

maker education, and proves that gamification programming improves students' interest in learning (Yuan, 2014) and computational thinking ability (Zhang, 2018). All in all, the research on primary school programming mainly focuses on the application practice test of some gamification programming tools, and the research related to the primary programming teaching method is still paid.

Roughly speaking, based on graphical programming tools, teachers often organize students to carry out project-based learning and problem-based learning to complete a project and solve a problem in a way that motivates students to “go forward” and ultimately achieve goals in the process. Exercise students' clear thinking and efficient way of thinking about solving problems, that is, improve students' thinking skills. With imperative programming tools, teachers usually guide students through autonomous learning to acquire programming knowledge, master the corresponding skills, and train computational thinking skills. The game is more interesting and immersive. As the current domestic programming education has just started, the actual teaching situation is relatively complicated. The teaching strategies adopted should be different for different teaching situations.

Primary School Programming Teaching Strategy Based on Gamification Learning

As it can be seen from the foregoing, gamification learning is a feasible approach to primary programming. Specific to the Chinese primary school programming education environment, based on theoretical analysis and practical teaching practice, this study is constructed and updated through two iterations to form a primary school programming teaching strategy based on gamification learning, as follows:

The basic information-based teaching environment is the premise of implementing gamification programming teaching in primary schools. The information-based teaching environment is divided into two parts: hardware facilities and software environment. The information-based teaching environment of a subsidiary foreign language school in Guangzhou is good. As shown in Figure 1, in the hardware facilities, each classroom has an independent wireless LAN, with good wireless Internet access conditions, in addition to the traditional whiteboard in the classroom is also equipped with a touch-one machine. In addition, a part of the third grade primary school class was known as the “iPad class”, these classes or schools have unified procurement equipment, or students bring their own equipment (BYOD) to ensure that an iPad is available. In the software environment, the most important thing is to choose a major child's programming learning tool or platform. The author chooses the iPad-based Swift Playgrounds App which developed by Apple according to the basic situation of the students and the teaching environment of the school. 3D passing through checkpoints game is the main learning content, supplemented by a variety of flexible robot-driven programming creation of the imperative children's programming gamification learning platform. And secondly, based on the learning process recording software Seesaw, not only formed the student's personal learning space, but also formed a “sheng-teacher-school” interconnection platform to facilitate home-school communication and better support children's learning.

Evaluate the school's teaching situation and determine the programming curriculum organization. At present, the teaching of elementary school programming is still in the initial stage of exploration. The common form of course can be divided into two forms: A complete programming course and a module course embedded in other courses as shown in Figure 2. The former appears in the form of electives and interest classes. It is still rare as a formal course, but it is a relatively complete program based on programming. It can be offered in a school elective course or “four points”. Semi-classroom (student's interest class after 4:30 pm),

students choose whether to study according to their hobbies. The latter is generally embedded in other courses as a relatively complete module, such as embedding an information technology course, or combining interest courses, such as the Maker class to organize relevant programming learning and practical activities. We need to comprehensively analyze the school’s curriculum to set the actual situation and the learner’s overall interests and hobbies, to choose whether to open a complete course or a modular course to determine the course capacity. The school where the author is located has opened a large number of elective courses for students to choose. In addition, it is easy to accept that students in the third-grade and above are more likely to accept iPad-based learning (one-to-one digital learning). Elective courses are scheduled for one class per week for a total of 15 class hours. There are 23 enrollment courses (the school stipulates that more than 18 people can start classes), of which 19 are third-grade students, accounting for 82.6%, and the remaining four are fourth-grade.

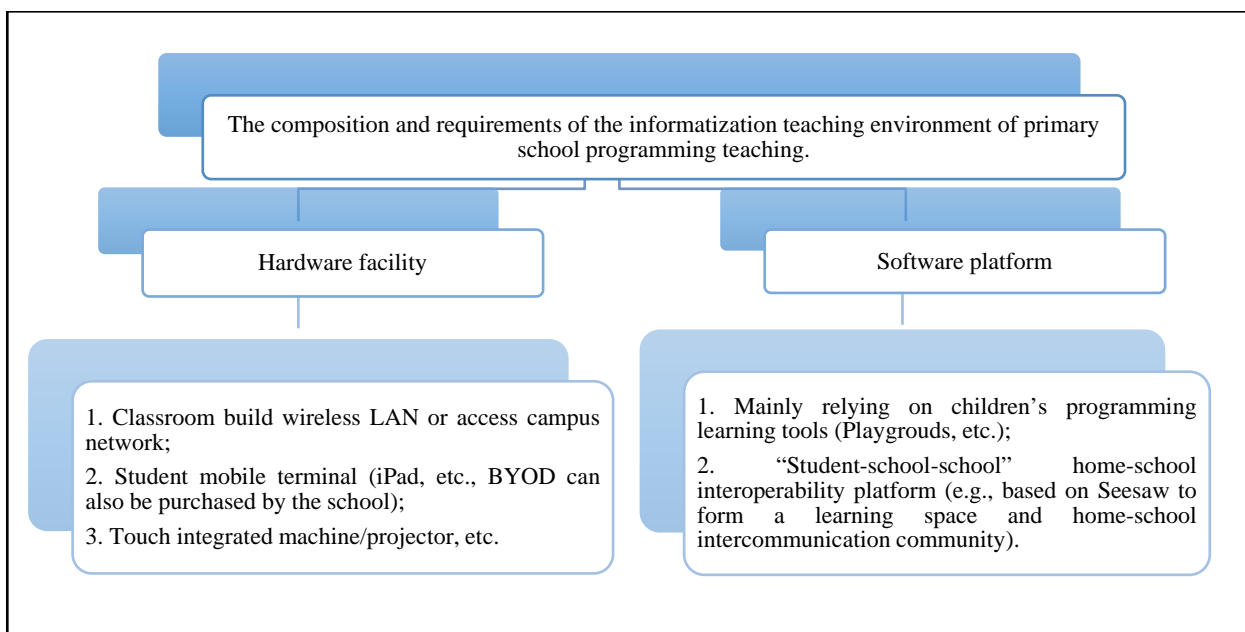


Figure 1. The composition and requirements of the informatization teaching environment of primary school programming teaching.

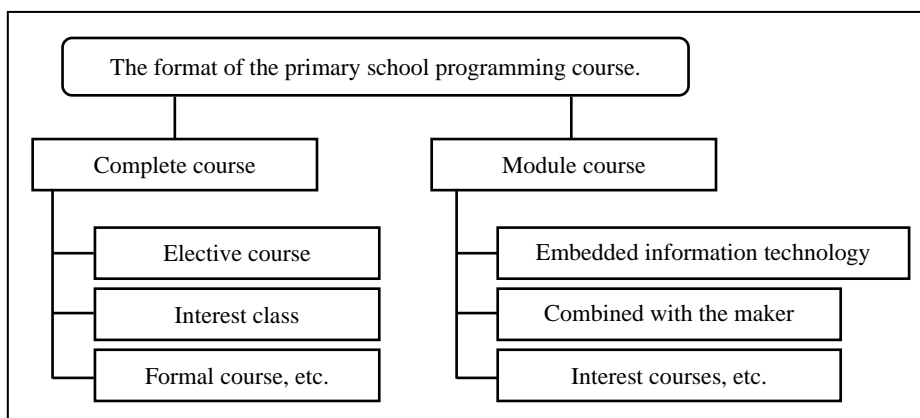


Figure 2. Primary school programming course general course organization form of primary school programming teaching.

Analyze learner characteristics and form study groups. After the course form and the students are

determined, it is necessary to start to fully understand the students. You can start with the students' starting ability, learning style, learning attitude, and information literacy. In addition, if it is an elective course, there may be students' equipment. In a unified situation, it is necessary to confirm in a timely manner whether the students' equipment specifications meet the minimum requirements for programming teaching. For example, the Playgrounds App needs to run on iOS 10 or higher systems, to ensure that each student's device can run the teaching tools. After fully understanding the basic situation of the students, because the programming gains itself to promote teamwork, in order to develop the students' sense of collaboration, students can be grouped according to the principle of "homogeneity within the group and heterogeneity within the group". Each group of 3-5 people should be suitable. The author divides each student into a group. Each group has their own group name and slogan, forming a sense of belonging and initial cooperation awareness.

Give a certain class "pre-" guidance, let students try self-study. There is no doubt that the current primary school students belong to digital aborigines. They have amazing talents in programming and learning, and they can realize rich ideas. What the teachers have to do is not to explain the steps step-by-step and explain the programming terminology, but to use the course before the official start of the course. A well-crafted and comprehensive study task guide students to self-study, allowing students to take advantage of the task list, learning partners, and teachers and parents to overcome the challenges and find out the difficulties they have encountered. The author first tried to flip the classroom teaching, but soon found that the third-grade students need more personalized guidance in learning programming in this teaching mode. Parents and companions have their own "quality" and time and energy. The limitations of the conditions do not provide help to meet their needs, and it is obviously unrealistic to produce a complete guiding resource that meets the needs of each student. Therefore, when the strategy is iteratively adjusted, the teaching method of "intra-school small flip" is boldly adopted. The original flip teaching needs to complete the part of the task before the class in the first 10 minutes of the class (the length of time can be customized), so that students can collectively "self-learn" in a small team, the teacher "tours" in the classroom at this time. Answer the students' doubts in time and guide the students to complete the self-study tasks as shown in Figure 3.

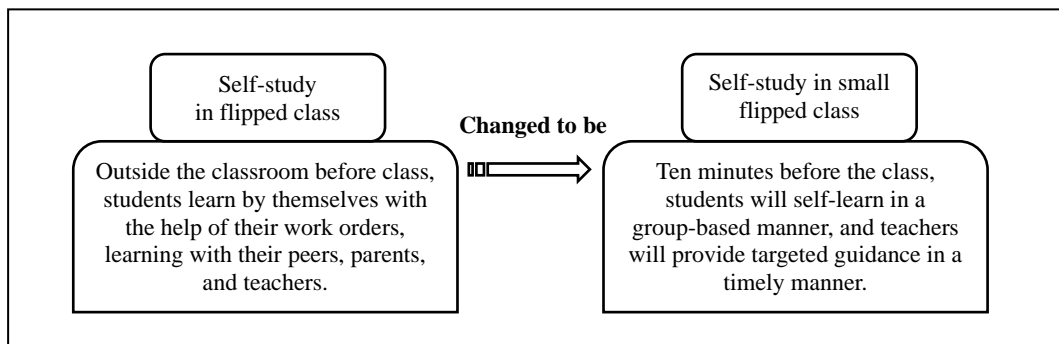


Figure 3. The application of the classroom teaching method in the primary school programming curriculum school programming teaching.

Problem-based/project-based gamification learning, focusing on teamwork. The teaching of children's programming courses often creates certain game situations, giving students the challenges and tasks to be completed. Students need to combine the knowledge base of self-learning before class, and cooperate with peers to determine problems and solutions according to task requirements. Learn programming skills and improve your thinking skills while overcoming challenges. It should be noted that in the actual teaching, some

students find that they do not like to cooperate with others in the process of completing the programming challenge. When they encounter difficulties, they will not actively seek help, which may result in “falling behind”. At this time, teachers and teaching assistants must discover this situation in time and adopt certain strategies to guide students to collaborative learning. For example, students who can complete a certain challenge faster in the group can help those who have difficulties and use their own “discourse” to jointly overcome difficulties. For some difficult-to-understand programming terms, you can set the corresponding unplugged programming game activities to promote students’ understanding and experience. For example, for playgrounds teaching, you can set up some “role-playing” game activities for students to simulate. The role of the game, the experience of the code instructions, can help students understand the relevant terms more effectively.

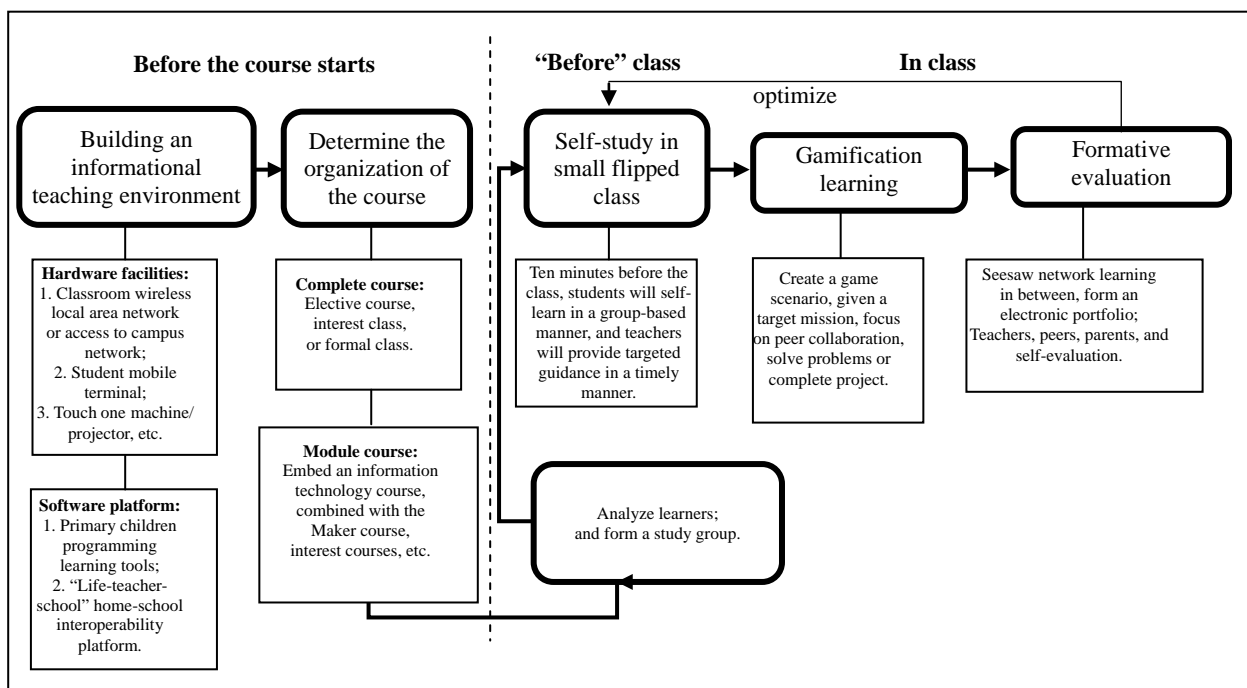


Figure 4. Programming teaching strategy based on gamified learning in primary school.

Formative evaluation is an important evaluation method for primary school programming courses.

Primary school students learn programming, what is important is not the acquisition of programming skills, but the improvement of thinking ability, that is, the way of thinking to solve problems efficiently. In addition, learning programming can also develop students’ teamwork and communication skills, enhance their self-confidence, patience, and concentration, and enhance students’ ability to solve real problems. The evaluation of these abilities is difficult to achieve through traditional diagnostic evaluations, so the formative evaluation of students’ learning process is particularly important. In the process of course implementation, the author uses Seesaw to form a professional student learning space and communication platform. Students upload their learning processes and results to the online learning space by means of pictures, images, and texts. Other students include parents of children. Students can be seen, encouraged, and commented on. Most importantly, each student has formed a personal portfolio that allows teachers to comprehensively assess each student’s learning outcomes. For some basic intellectual content, of course, a small amount of summative testing can also be taken. Finally, in addition to teacher evaluation, do not overlook peer evaluation, parent evaluation, and

student self-evaluation.

Combined with the above description and two iterations of modification in the teaching implementation process, the finalized gamification learning based on the primary school programming teaching strategy is shown in Figure 4. It consists of two parts: preparation before the course start and implementation of the course. The strategy separates that the preparatory work before the start of the course as a part to illustrate its importance. The establishment of the primary school programming course is an innovative attempt, and it requires sufficient preparation work. The construction of the hardware environment and the construction of the software platform are prerequisites for programming teaching condition. In the implementation part of the course, in order not to increase the burden of students, embodying the concept of “happy programming”, adjust the pre-class self-study course of the flip classroom to 10 minutes before the class, so students can also collaborate on self-study to solve the difficulties encountered in self-study. At the same time, it is also convenient for teachers to give targeted guidance. Classroom teaching adopts gamification learning mode, allowing students to collaborate on completed tasks in the created game context, and subtly acquire knowledge and training thinking. Throughout the learning process, students are guided to use Seesaw to save and upload the learning process and results to the online learning space, form an electronic portfolio, encourage students to grow, and become the most important material for teachers to evaluate students. Finally, pay attention to the evaluation. A combination of evaluation methods.

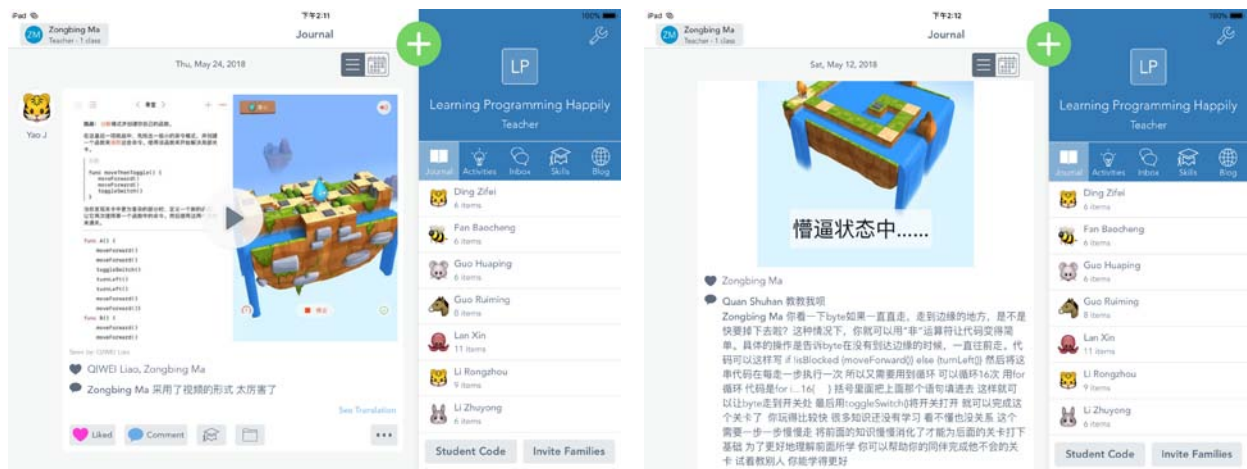


Figure 5. Screenshot of Seesaw web learning space interface.

Discussion on the Application Effect of Primary School Programming Teaching Strategy Based on Gamification Learning

Formative data analysis. The “Happy Learning Programming” elective course taught by the author created an online learning space based on Seesaw, and 23 students and a few parents joined the virtual space. From the interactive data point of view, Seesaw has about 10 sessions (15 sessions), and students have generated 105 works or results sharing. Among them, very active students (10 or more works share) account for 26.1%. It is more active (the work is shared seven times or more), accounting for 91.3%. In general, most students are willing to share their learning experiences in the space, but at the same time, because students’ collaborative learning awareness is not strong, the space is liked and the number of comments is relatively small. From the point of view of sharing content, most of the students completed the task challenge of the

course, and gradually developed the preliminary calculation thinking ability, such as “clear problem-decomposition step-algorithm implementation-acquisition and troubleshooting”. Figure 5 is the interface of Seesaw learning space Screenshot. Some students have completed challenging tasks and levels ahead of time, and actively consulted teachers during their spare time, showing a strong interest in programming learning.

Summative data analysis. According to the results of the course completion test, the average score of the students is 87.83 points, which indicates that most of the students have a good grasp of the module knowledge, such as “command, function, cycle, and selection” in the programming. The average score of 15 boys is 91, the average score of 8 girls is about 82, and the independent sample *t*-test shows $p = 0.255$. Therefore, there is no statistically significant difference among male and female students. In the stage, gamification-based programming is a good vehicle for developing students’ thinking (It is also possible that the sample size is too small and the data is not objective).

Questionnaire results discussion. The student’s preference for programming course is ranked three levels of A, B, and C. The results of the questionnaire show that 60.9% are considered to be A, 34.8% are considered to be B, and only 4.3% are selected as C, indicating that more than 90% of students are programming learning is very interesting. And from the level of scores and the degree of preference for programming courses, the more you can think of, the more you like programming, the better the results. From the feedback of parents in the home school WeChat group, some children said that he hopes to learn programming courses next semester.

In general, the application of gamification-based programming teaching strategies has done a good job in teaching tasks, which has made primary school students have a keen interest in programming learning, mastered the basic knowledge of children programming, and exercised and developed their initial computational thinking skills.

Conclusion and Outlook

Children’s learning programming can train their logical thinking ability, abstract thinking ability, ability to solve real problems, and computational thinking ability, can cultivate concentration, teamwork, and communication skills, and can enhance self-confidence to make it more suitable for future artificial intelligence. The requirements of the times, programming education in primary schools has become the trend of the times. It turns out that primary school programming education is feasible and effective with gamification learning.

This study proves that the programming teaching strategy based on gamification learning can enhance students’ interest in learning, help students master the basic knowledge of programming, and cultivate preliminary computational thinking ability. In the teaching of elementary school programming based on gamification learning, we also need to pay attention to the following points:

First, control the number of students as much as possible. Practice shows that each student’s progress is different, teachers need to provide more personalized guidance, so layered teaching is a worthwhile approach, too many people will make it difficult to achieve.

Second, do not rely solely on electronic tools and platforms. In addition to video games, gamification teaching should also include game activities. It is necessary to properly set up some unplugged programming activities, contact real life, and help students understand the core concepts and logic of programming.

Third, give full play to students’ autonomy. When appropriate, students should be guided to find ways to overcome challenges, rather than telling them directly, students will show amazing potential and creativity

when learning programming.

Fourth, programming emphasizes hands-on, so it is necessary to reserve a lot of time for students to “code”.

Fifth, make full use of all available resources. This includes communicating with educators in the online programming community and accessing educational materials on the Open Education Resources Website.

Sixth, become a learner. Primary school programming education is developing rapidly, and innovation in the field is overwhelming. Only by following up the status quo in time can we better organize our teaching.

Finally, the ultimate goal of primary school programming education is still the development of students' computational thinking ability, which is the only way to make it better adapt to the future society.

Therefore, the research of the corresponding computational thinking evaluation system has become an important research hotspot.

References

- Bao, X. Y., & Zhao, Y. X. (2015). Research progress and prospects of gamification learning. *Research on Audio-Visual Education*, 8, 45-52.
- Beijing Open University Horizon Reporting Project Team, Bai, X. J., Zhang, C. H., Ji, R. F., Wu, S. S., & Li, G. Y. (2017). New technology drives teaching innovation trends, challenges and strategies—2017 Horizon Report (Basic Education Chinese Edition). *Modern Chinese Education Equipment*, 18, 1-20.
- Children's Programming Network. (2017, Oct. 10). Children's programming education: To develop children's thinking and ability to benefit their lives. In *Children's Programming Network* [Online]. Retrieved from <http://www.shaoerbianchengwang.com/1639.html>
- Choi, J., Lee, Y., & Lee, E. (2016). Puzzle based algorithm learning for cultivating computational thinking. *Wireless Personal Communications*, 93(1), 1-15.
- Grover, S., & Pea, R. (2013). Computational thinking in K-12 a review of the state of the field. *Educational Researcher*, 42(1), 38-43.
- Hsu, C. C., & Wang, T. I. (2018). Applying game mechanics and student-generated questions to an online puzzle-based game learning system to promote algorithmic thinking skills. *Computers & Education*, 122, 73-88.
- Niu, Y. X., & Ren, W. (2006). A preliminary study of gamification teaching. *Software Guide*, 10, 4-5.
- Ren, Y. Q. (2018). Cultivate a new generation of “computational thinking +” for digital China. *China Information Technology Education*, 5, 7-7.
- Sáez-López, J. M., Román-González, M., & Vázquez-Cano, E. (2016). Visual programming languages integrated across the curriculum in elementary school: A two year case study using “scratch” in five schools. *Computers & Education*, 97, 129-141.
- Wang, D. P. (2015). Research on interactive design of online teaching based on gamification learning (Ph.D. dissertation, Northeast Normal University, Changchun, Taiwan).
- Wang, T. T., Wang, D. L., Lu, W., He, L., Wang, H. G., & Dai, G. Z. (2013). Graphical programming language and tools for children. *Journal of Computer-Aided Design & Computer Graphics*, 25(4), 584-591.
- Wu, J. (2006). The development of the logo language. *China's Electrification Education*, 7, 49-51.
- Yuan, Q. (2014). Introducing scratch into the elementary school information technology classroom. *China Education Informatization*, 20, 85-86.
- Zhang, J. L., & Zhang, B. H. (2013). Application research of gamification learning concept in flipping classroom teaching. *Journal of Distance Education*, 31(1), 73-78.
- Zhang, W. (2018). The development and application of primary school programming school-based textbooks to cultivate computational thinking—Taking the application of scratch creation “Maze Wars” as an example. *Digital Teaching in Primary and Secondary Schools*, 3, 55-57.