Penerbit Akademia Baru

International Journal of Advanced Research in Future Ready Learning and Education

https://www.akademiabaru.com/submit/index.php/frle/index
ISSN: 2821 - 2800

Review of Design Thinking Approach in Learning IoT Programming

Salbiah Zainal[1,*], Rasimah Che Mohd Yusoff[1], Hafiza Abas[1], Suraya Yaacub[1], Norziha Megat Zainuddin[1]

[1] Razak Faculty of Technology and Informatics, Universiti Teknologi Malaysia, Kuala Lumpur, Malaysia

ABSTRACT

Internet of Things (IoT) is one of the keys for the Fourth Industrial Revolution (IR4.0) which refers to the use of intelligently connected devices and systems by embedded sensors and actuators in machines and other physical objects. With the pace of rapid IoT adoption, the demand for skilled IoT developers rises in tandem. This paper review the integration of design thinking as a pedagogical in IoT programming education. Topics on challenges in developing IoT systems, models, or frameworks for learning IoT programming, design thinking, design thinking approach in IoT system prototype development and teaching strategy technique have been discussed.

Keywords:
Industry 4.0; design thinking; IoT; IoT programming

1. Introduction

In the Fourth Industrial Revolution (4IR) or Industry 4.0 (IR4.0), there will be a fundamental paradigm shift caused by the combination of Internet technologies and future-oriented technologies. Since Germany announced IR4.0 in 2011 and promised to bring remarkable benefits to the manufacturing industry worldwide, many countries have rushed to launch similar initiatives [1]. For example, the United States of America has launched a similar initiative called Smart manufacturing. In 2014, China revealed a national 10-year vision called “Made in China 2025” to transform China into a world manufacturing power [2, 3]. In 2016, Japan went far beyond Industry 4.0 and they shared the vision of what is called Society 5.0. Society 5.0 focuses on the digitalization of all life sectors of Japanese society [4]. Malaysia also introduced National Industrial Revolution 4.0 (IR4.0) which expect to improve the country’s productivity by 30% across all sectors by the end of 2030 where Malaysia government has identified five core technologies to strengthen local ability, namely artificial intelligence (AI); Internet of Things; blockchain; cloud computing; and big data analytics.

Internet of Things (IoT) is one of the keys for IR4.0 which uses a network of interconnected devices to deliver data via the Internet. Devices, machines, sensors, and people are able to communicate with each other by using IoT [5]. IoT is used across a broad range of industries such as healthcare, manufacturing, automotive, retail, and building automation, among others. Malaysia is well-

* *Corresponding author.*
E-mail address: salbiahzainal@yahoo.com.my

positioned to become a regional hub for IoT, as the country has strong foundations in the electronics and semiconductors manufacturing industries. The pace of rapid IoT adoption will lead to rising demand for expertise to help companies in specific industries that incorporate IoT technology. Higher demand in these areas infers an increased need for qualified software developers. So, this paper aims to review the integration of design thinking as pedagogical in IoT programming education which not only focuses on problem-solving also encourages innovative ideation in generating new ideas among students. Section 2 will be on the related literature and Section 3 will be the conclusion.

2. Literature Review

2.1 Design and Development of IoT System

IoT programming is different since it involves connected devices which comprise of IoT device, gateway, and a cloud or server. IoT device transmits sensing data acquired by sensors to the IoT gateway via the sensor network. The IoT gateway transmits the sensed data to the cloud via the internet. The sensing data are processed on the cloud with visualization and analysis, causing IoT device actuators to actuate via the IoT gateway. Consequently, with IoT systems, plural elements work together to establish an effective system. So, learning and teaching IoT programming will be more complex since IoT devices are implemented using both hardware and software components. Dedicated hardware components are used to implement the interface with the physical world, and to perform tasks which are more computationally complex. Microcontroller's development board is used to execute software that interprets inputs and controls the system. For students to create IoT system ideas, they should experience the system through IoT system construction. Among the most favourites microcontrollers are Arduino, Raspberry Pi and Microbit that help beginners and developers to build IoT projects. Basically, Arduino microcontrollers are designed for hardware development while the Raspberry Pi models are developed for software development. The Microbit is designed for wearable hardware-oriented purposes. In order to properly select the right platform, the users must understand what kinds of IoT systems to be applied or implemented based on the specifications and capabilities of each platform [6]. Consequently, there are five interconnected subsystems in an IoT systems which consist of sensors, gateways, back-end, actuator and end-user.

Sensors monitor End-user activities and detect changes in the environment by measuring variables such as temperature, humidity, and occupation, among others. They generally refer to wearable devices and environmental sensors.

Gateways gather the data coming from the Sensors and perform computation and reasoning tasks over it. If more computing or storage capacity is required, the Gateways communicate with the Back-End subsystem and delegate the most demanding tasks. Furthermore, Gateways also interact with the actuators. They control the acting devices based on the outputs from their computations or based on the instructions that they receive from the Back-End subsystem. In the projects developed by novice IoT developers, this subsystem typically consisted of single-board computers such as Raspberry Pis.

Back-end groups third-party services APIs, the main application server, and the persistence component. The functionalities provided by the application server and the persistence component are typically exposed to the Gateways subsystem through RESTful web services. Finally, third-party service APIs are commonly used to interact with the wearable devices belonging to the Sensors subsystem.

Actuators span actuating devices that trigger changes in the physical environment. However, they also encompass push notifications through which end-users are informed about the occurrence of a given event. Acting devices are generally controlled by gateway devices via Bluetooth or Wi-Fi, while push notifications are commonly generated in the Back-End subsystem through the Android and iOS push notifications platforms APIs.

End-user refers to the interfaces with which the End-users are enabled to interact with the IoT system. These interfaces typically consist of mobile and web applications through which user preferences can be configured, Actuators can be activated or deactivated, and Sensors can be monitored and managed.

Figure 1 shows an example of IoT systems reference architecture which consists of different subsystems (sensors, actuators, gateways, backend, end-user) connected among each other and with the end-users [7].

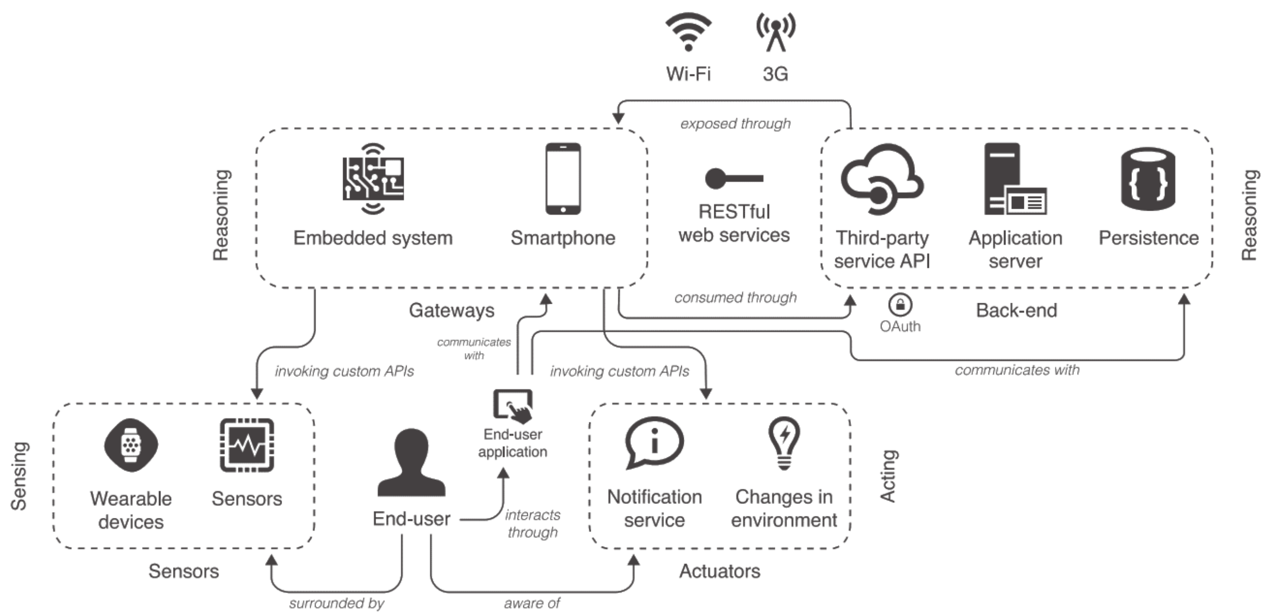

Fig. 1. Example of IoT systems reference architecture which consists of different subsystems (sensors, actuators, gateways, backend) connected among each other and with the end-users [7]

2.2 Challenges in developing IoT systems

Current exposure to IoT concepts, programming and practices in higher education institutions is still little. Students have lack of holistic understanding of what IoT is [8]. In Malaysia, despite the increase of IR4.0 initiatives by the government and industry, more than half of the students and graduates were unable to articulate what IR4.0 entails. Not many students took the initiative to take courses to get additional certifications relevant to the IR4.0 workplace, citing time constraints and a lack of value as their reasons [9].

In higher education institutions, programming is one of the basic competencies that students should have in computer science and engineering programs and also in other programs such as mathematics, computer education and instructional technologies. In electrical and computer engineering disciplines, students are exposed to embedded programming, cyber-physical programming, real-time system programming and sensor programming which are closely related to IoT programming. However, these programming concepts are new to non-computer engineering students [10]. IoT programming is different since it involves connected devices which comprise of IoT device, gateway, and a cloud or server. Consequently, with IoT systems, plural elements work together to establish an effective system. So, learning and teaching IoT programming will be more complex since IoT devices are implemented using both hardware and software components. Dedicated hardware components are used to implement the interface with the physical world, and to perform tasks which are more computationally complex. Microcontrollers development boards are used to execute software that interprets inputs and controls the system. For students to create IoT system ideas, they should experience the system through IoT system construction.

Students face difficulty in writing programming since it requires higher-order thinking skills [4,11]. To produce a skilled programmer requires a lot of experience solving program problems [12]. So lab exercises and hands-on are very important for teaching programming and go beyond normal in-class exercises. Besides technical skills, soft skills such as critical thinking, problem-solving, leadership skills and lifelong learning are also important to prepare future talents to be part of the IR4.0 workforce. There is a student that developed an IoT prototype system but during the programming practice presentation, they were unable to ascertain what they did not understand well and were unable to present their thoughts clearly because some of the students did not understand the basic programming concepts [13].

Technology-oriented learning-teaching processes lack of pedagogical aspects that lead to inefficiency example for programming education [14]. There are several approaches for programming education. IoT related programming being taught using problem-based learning (PBL) [13,15,16]; project based [17]; cooperative learning [18] and other methods [19, 20]. However, these approaches focus on problem-solving through analysis and lack of innovative ideation which is the creative process in generating new ideas. Some researchers used the design thinking approach to teach Java Object-Oriented programming [22] and programming for gifted students [14] but the design thinking approach is not being implemented for IoT programming education. Besides that, current approaches lacking the inter-play between analysis (breaking problems apart) and synthesis (putting ideas together) and did not emphasize participants to generate insights from different domains through drawing, prototyping and storytelling [25].

2.3 Models or Frameworks for Learning IoT Programming

The PBL approach has been implemented in a blended learning environment for student's self-construction of the IoT prototype system [13]. Among important topics related to the basic configuration of an IoT system are IoT devices, IoT gateway, cloud or server and actuator.

The model consists of four steps by self-study and PBL style class to cover all topics. Students using tutorial agents to support the construction of an IoT prototype instead of a teacher. They are a "wiring support agent" that supports the wiring work of electronic parts, a "programming support agent" that supports the creation of a program to acquire and transmit sensor data, and a "wireless setting support agent" that supports the wireless communication setting. Procedure manuals are also being used to help them construct a prototype system through self-study. In the class, students present the results of self-study and ideas of the IoT system and discuss them. After students experience IoT system construction, they present ideas and discuss them. Fig 2 shows the PBL approach for student's self-construction of IoT prototypes [13].

Other research proposed a model that uses approaches of Design-Based Learning (DBL) and Cooperative Learning (CL) in order to improve the algorithmic thinking on how hardware and software components can be combined in the construction and implementation of designs [18]. In this model, there are four components in the sequence of learning of a student: conceptualizing, analyzing, designing, implementing and debugging, which are articulated with the DBL approach. The first part of the course explored the concepts of programming with Python and Arduino platforms. In the second part, contents associated with the Raspberry Pi platform, employing sensors and actuators will be introduced. Fig 3 shows the general model of blended learning using Design-Based Learning (DBL) and Cooperative Learning (CL) to enhance students' algorithmic thinking.

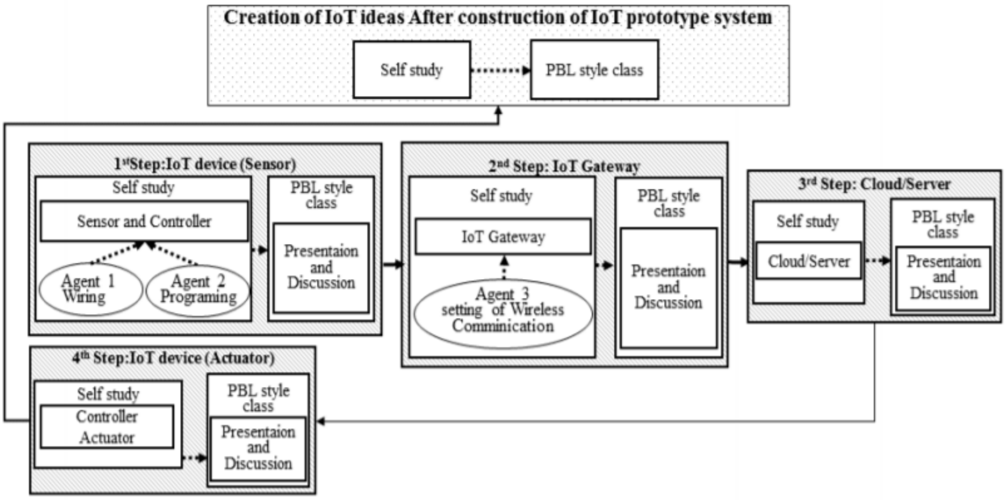

Fig. 2. PBL approach for student's self-construction of IoT prototypes [13]

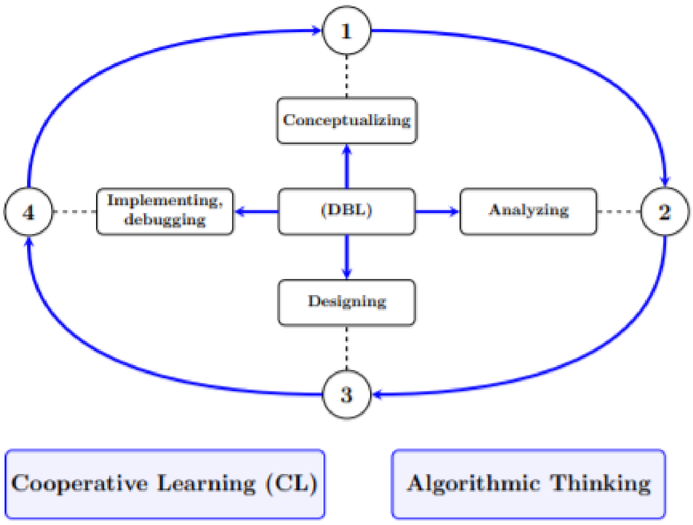

Fig. 3. The general model of DBL and CL to enhance students' algorithmic thinking [18]

Most of these approaches focus on problem-solving through analysis and, however lack of innovative ideation which is the creative process in generating new ideas. Some researchers used the design thinking approach to teach Java Object-Oriented programming [22] and block-based, text-based, physical, and mobile programming for gifted students [14] but the design thinking approach is not being implemented for IoT programming education yet.

2.4 Design Thinking

Design is an interdisciplinary domain that employs approaches, tools, and thinking skills that help designers devise more and better ideas toward creative solutions [28]. The term "design thinking" refers to cognitive processes of design work or the thinking skills and practices designers use to create new artifacts or ideas and solve problems in practice [17]. Design is the creative process of intentionally developing something that does not yet exist. Thus, both analytical thinking and divergent creative thinking are key to design processes. Design lies at the intersection of art and science and applies to a wide range of human-centered disciplines through creative work. A designer's work is iterative and often idiosyncratic, but designers' creativity and design choices are scaffolded and informed by common processes [10]. These design thinking skills give flexible support and grounding to the open-ended arena of creative practice [19]. Design involves directing creativity towards goals, actions, and purpose around real-world issues [12,19]. This situates design as a creative problem-solving and thinking approach for human-centered professions, such as doctors, nurses, engineers, and others–most notably, educators. Design thinking is considered as a human-centered approach that aims to find creative and innovative solutions to various social and commercial problems by using design tools and mindsets [26]. During the process of design thinking, learners work on targets that are not clearly defined and unstructured problems that have no solutions stated yet.

Design thinking has become a pedagogical phenomenon in higher education due to its widespread relevance across many disciplines [23]. Teaching-learning approach can be enriched with design thinking since design thinking relies on an interplay between analysis (breaking problems apart) and synthesis (putting ideas together) and allow participants to generate insights from different domain. The effectiveness in bringing 21st-century skills and characteristics to students creates the educational value of design problems. Since there is no common description of design thinking in literature, there is no single way to follow the design thinking process. Institutions such as Stanford University Hasso Plattner Design School, IDEO and Design Council have developed many designs process models. In all models; collecting information in order to understand the problem, using creative thinking skills in the process and being experiential during the process were always highlighted [27]. The four principles of design thinking are:

i. The human rule: No matter what the context, all design activity is social in nature, and any social innovation will bring us back to the "human-centric point of view".
ii. The ambiguity rule: Ambiguity is inevitable, and it cannot be removed or oversimplified. Experimenting at the limits of your knowledge and ability is crucial in being able to see things differently.
iii. The redesign rule: All design is redesign. While technology and social circumstances may change and evolve, basic human needs remain unchanged. We essentially only redesign the means of fulfilling these needs or reaching desired outcomes.
iv. The tangibility rule: Making ideas tangible in the form of prototypes enables designers to communicate them more effectively.

2.5 Design Thinking Process

The design thinking process relies on the principles of empathizing in order to understand user needs, defining the needs, making trials, prototyping, receiving feedback from users, redesigning the process and expressing one's thought through creative ways besides using words and symbols [14]. The design thinking process can be broken down into five steps (empathize, define, ideate, prototype and test) [17]. Figure 4 shows five steps in the design thinking process (empathize, define, ideate, prototype and test).

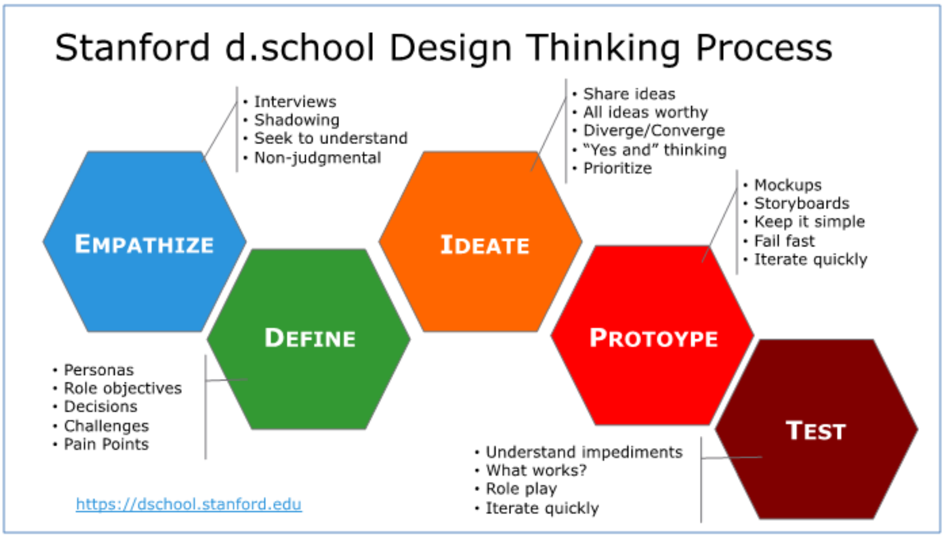

Fig. 4. Stanford design thinking model [17]

Empathize: Empathy is at the foundation of human-centered design as an essential starting point for any design work. In this mode, designers observe users and their behaviors, interact with and interview them, and immerse themselves in understanding the experience and perspective of the user. These insights allow designers to approach the rest of the process with a stronger understanding of the context and problem.

Define: designers use insights gathered from empathizing to focus on the problem. They aim to go beyond a simple definition as they describe the complexities of the user, the problem, and the context. In this mode, designers articulate a problem statement based on details and understandings they gained previously. They focus on and frame the problem, to guide design efforts moving forward.

Ideate: explores a wide variety of solutions and ideas. The goal is to go beyond the obvious to brainstorm, incubate and generate far-ranging ideas, solutions, and approaches connected to the problem. Designers must go wide with ideas, keeping the problem in mind, but also letting flights of fancy bring up novel, creative ideas. Deferring judgment on evaluating ideas allows the unconstrained development of ideas.

Prototype: After designers have generated ideas, they put those ideas into action in the fourth mode of Prototype, by creating a possible prototype or a model of a solution(s) to the problem (which can later be tested). It is not an attempt to arrive at a final solution, but an opportunity to try making ideas concrete.

Test: Designers test the prototype with actual or representative users/stakeholders. Designers may interview users, observe them interacting with the prototype, or use other methods to gather feedback for refinement of the solution(s). Testing may show that a designer must refine the prototype, or redefine and re-examine the original point of view. They may revisit the empathize mode to understand users or return to the ideate mode to explore alternative solutions. Design is iterative, and at any point, a designer might repeat or reconsider a phase.

2.6 Design Thinking Approach in IoT System Prototype Development

A typical design thinking process consists of a cycle of (i) empathizing and observing, (ii) defining the problem, (iii) creating ideas, (iv) prototyping, and (v) testing [25].

Phase 1: Empathise
Empathy provides the critical starting point for design thinking. The first stage of the process is spent getting to know the user and understanding their wants, needs and objectives. In order to develop insights, students observe how people behave, and how they interact with other people and the environment. Also, they can record projections regarding the answers to questions asked. This way, empathy will be established with the user.

Phase 2: Define
In this phase, the need is defined. Action-based problem statements will be stated after analyzing and synthesizing the data obtained during the empathy phase. Problem situations are expressed as Point of View (POV) statements which are formed by combining “User + User’s Need + Insight”. The problem-defining phase supports creative thinking skills in the context of evaluating a situation or problem from different angles, redefining present models and enabling the production of new information by developing multiple points of view.

Phase 3: Ideate
With a solid understanding of users and a clear problem statement in mind, it’s time to start working on potential solutions. The third phase in the design thinking process is where creativity happens, and it’s crucial to point out that the ideation stage is a judgment. Brainstorming is a phase aimed at producing many ideas in various categories devoted to finding a solution for the defined problem. Students can participate in brainstorming processes in different groups or individually.

Phase 4: Prototype
This phase is about experimentation and turning ideas into tangible products. A prototype is basically a scaled-down version of the product which incorporates the potential solutions identified in the previous stages. This step is key in putting each solution to the test and highlighting any constraints and flaws. Any kind of thing that has a physical component such as an object, role-play activity, an interface, a visual scenario draft inherits the feature of being a prototype.

Phase 5: Test
In the testing phase, the user is able to experience the developed prototype and give feedback to the designers. Solutions developed according to user feedback are evaluated and will be improved accordingly. Additionally in this phase, gathering more information about the user and the improvement of POV statements after testing can be realized.

In all design thinking tasks; design thinking digital presentation, Empathy Map Template, Point of View (POV) Template, User Feedback Template, Online Stopwatch Web 2.0 Tool, A4 papers, sticky note papers, color pencils and Ideate Cards were used. As for programming environments and physical programming tools; Scratch Block-Based Programming Environment, Python Text-Based Programming Environment, Ardunio Uno Ultimate Set and Arduino IDLE Text-Based Physical Programming Environment, Lego Mindstorm EV3 Education Kits and Add-on Kits, Legomindstorms Education EV3 Teacher Edition Block-Based Robot Programming Environment and App Inventor 2 Mobile Programming Environment were used.

2.7 Teaching Strategy for Design Thinking in IoT System Prototype Development

During the implementation of design thinking in prototype development, students claimed that the period to conduct the task was to limited. So, there are some techniques that can be applied in the practice of the design thinking approach in IoT system development in order to give enough time for students to conduct the task. IBIA (Initiator, Before, In, After) teaching strategy can be used which is interactivity and embraces inductive learning environment [29].

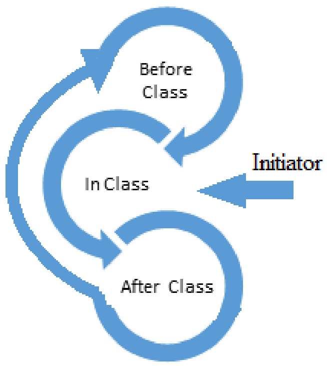

Fig. 5. The IBIA teaching strategy [29]

The IBIA teaching strategy is conducted as follows:
Initiator: The lecturer prepares the task-related of any challenge that affected communities' daily lives and discusses how such challenges could be addressed by technology such as cyber-physical systems and/or the internet of things.
In Class: Briefly discuss with students the purpose of the assignment/task and discuss the envisaged Output. The lecturer emphasizes to the students on the tools that can be used to address the challenge. This tool can be used after class.
After Class: The students use the tools needed to carry out their tasks. Furthermore, students identify issues that need discussion.
Before Class: The students complete their assignments after class. Then, they took notes and tried briefly to analyze the problems identified.
In Class: The students present their solutions, lecturers and students discuss and analyze the solutions proposed by students including problems identified by students. If the solution is good enough, it can be leveraged on different platforms.

3. Conclusion

Despite the design thinking pedagogical phenomenon in higher education, still a lack of implementation design thinking for programming education especially for IoT programming. This paper review the integration of design thinking as a pedagogical in IoT programming education. Topics on challenges in developing IoT systems, models, or frameworks for learning IoT programming, design thinking, design thinking approach in IoT system prototype development and teaching strategy have been discussed.

References

[1] Basilaia, Giorgi, and David Kvavadze. "Transition to online education in schools during a SARS-CoV-2 coronavirus (COVID-19) pandemic in Georgia." *Pedagogical Research* 5, no. 4 (2020).

[2] Bao, Wei. "COVID-19 and online teaching in higher education: A case study of Peking University." *Human Behavior and Emerging Technologies* 2, no. 2 (2020): 113-115.

[3] Ali, Wahab. "Online and remote learning in higher education institutes: A necessity in light of COVID-19 pandemic." *Higher education studies* 10, no. 3 (2020): 16-25.

[4] Baist, Abdul, and Aan Subhan Pamungkas. "Analysis of Student Difficulties in Computer Programming." *VOLT: Jurnal Ilmiah Pendidikan Teknik Elektro* 2, no. 2 (2017): 81-92.

[5] Aldowah, Hanan, Shafiq Ul Rehman, Samar Ghazal, and Irfan Naufal Umar. "Internet of Things in higher education: a study on future learning." In *Journal of Physics: Conference Series*, vol. 892, no. 1, p. 012017. IOP Publishing, 2017.

[6] Noor, Nur Qamarina Mohd, Azi Azwady Jamaludin, Azizul Azizan, Hafiza Abas, Kamilia Kamardin, and Mohd Fitri Mohd Yakub. "Arduino vs Raspberry Pi vs Micro Bit: Platforms for Fast IoT Systems Prototyping." *Open International Journal of Informatics (OIJI)* (2018): 96-107.

[7] Corno, Fulvio, Luigi De Russis, and Juan Pablo Sáenz. "On the challenges novice programmers experience in developing IoT systems: A survey." *Journal of Systems and Software* 157 (2019): 110389.

[8] Suduc, Ana-Maria, Mihai Bîzoi, and Gabriel Gorghiu. "A Survey on IoT in Education." *Romanian Journal for Multidimensional Education/Revista Romaneasca pentru Educatie Multidimensionala* 10, no. 3 (2018).

[9] New Strait Times (2019). https://www.nst.com.my/education/2019/10/526409/are-our-students-ready-ir40-workplace.

[10] Ridwan, Intisar Ibrahim, Izzeldin Ibrahim Mohamed, Rosmah Ali, and Nazar ElFadil. "Embedded systems: Teaching and design challenges for nonhomogeneous classes." *International Journal of Multimedia and Ubiquitous Engineering* 11, no. 10 (2016): 221-244.

[11] Alayan, Allie. "Making Pedagogic Sense of Design Thinking: Author Interview." *AACE Review* March (2020): 1-9.

[12] Özmen, Büşra, and Arif Altun. "Undergraduate Students' Experiences in Programming: Difficulties and Obstacles." *Turkish Online Journal of Qualitative Inquiry* 5, no. 3 (2014).

[13] Kennedy, Felbin C., and S. U. Malini. "Role of Octagonal Fuzzy Numbers in Solving Some Special Fuzzy Linear Programming Problems." In *International Forum for Interdisciplinary Mathematics*, pp. 152-163. Springer, Cham, 2015.

[14] Akiyama, Koji. "Problem-based Learning Style IoT System Education Method by Student's Self-construction of Prototype System." *International Journal of Internet of Things* 7, no. 2 (2018): 30-36.

[15] Avcu, Yunus Emre, and Kemal Oğuz ER. "Design thinking applications in teaching programming to gifted students." *Journal of Educational Technology and Online Learning* 3, no. 1 (2020): 1-30.

[16] Vogel, Bahtijar, Bo Peterson, and Blerim Emruli. "Prototyping for internet of things with web technologies: A case on project-based learning using scrum." In *2019 IEEE 43rd Annual Computer Software and Applications Conference (COMPSAC)*, vol. 2, pp. 300-305. IEEE, 2019.

[17] Looi, Hong Cheong, and Afzaal H. Seyal. "Problem-based learning: An analysis of its application to the teaching of programming." *International Proceedings of Economics Development and Research* 70 (2014): 68.

[18] Ramalho, J. F. C. B., R. A. S. Ferreira, R. Amante, M. Amaral, J. Cardoso, I. Ferreira, J. Almeida, and P. S. André. "Promoting IoT Education for Pre-university Students With Coloured QR Codes: Colour multiplexed QR codes." In *2018 3rd International Conference of the Portuguese Society for Engineering Education (CISPEE)*, pp. 1-5. IEEE, 2018.

[19] Ariza, Jonathan Álvarez. "Towards education alternatives to teaching and learning of programming: A course experience using open hardware tools." In *2018 IEEE Frontiers in Education Conference (FIE)*, pp. 1-8. IEEE, 2018.

[20] Chochiang, Kitsiri, Kullawat Chaowanawatee, Kittasil Silanon, and Thitinan Kliangsuwan. "Arduino visual programming." In *2019 23rd International Computer Science and Engineering Conference (ICSEC)*, pp. 82-86. IEEE, 2019.

[21] Ghazal, Mohammed, Fasila Haneefa, Samr Ali, Yasmina Al Khalil, and Ahmed Sweleh. "A framework for teaching robotic control using a novel visual programming language." In *2016 IEEE 59th International Midwest Symposium on Circuits and Systems (MWSCAS)*, pp. 1-4. IEEE, 2016.

[22] Henriquez, Yuranis, Kevin Pedroza, and Kevin Vega. "Yet another robot platform for teaching programming: YarpTp." In *2018 IEEE 2nd Colombian Conference on Robotics and Automation (CCRA)*, pp. 1-5. IEEE, 2018.

[23] Rajashekharaiah, K. M. M., Manjula Pawar, Mahesh S. Patil, Nagaratna Kulenavar, and G. H. Joshi. "Design thinking framework to enhance object oriented design and problem analysis skill in Java programming laboratory: An experience." In *2016 IEEE 4th International Conference on MOOCs, Innovation and Technology in Education (MITE)*, pp. 200-205. IEEE, 2016.

[24] Beligatamulla, Gnanaharsha, Janice Rieger, Jill Franz, and Megan Strickfaden. "Making pedagogic sense of design thinking in the higher education context." *Open Education Studies* 1, no. 1 (2019): 91-105.

[25] Panke, Stefanie. "Design thinking in education: Perspectives, opportunities and challenges." *Open Education Studies* 1, no. 1 (2019): 281-306.

[26] Lor, Rex. "Design thinking in education: A critical review of literature." (2017).

[27] Chesson, Dani. "Design thinker profile: Creating and validating a scale for measuring design thinking capabilities." (2017).

[28] Henriksen, Danah, Carmen Richardson, and Rohit Mehta. "Design thinking: A creative approach to educational problems of practice." *Thinking skills and Creativity* 26 (2017): 140-153.

[29] Malele, Vusumuzi, and Manthiba E. Ramaboka. "The Design Thinking Approach to students STEAM projects." *Procedia CIRP* 91 (2020): 230-236.