# Adopting Backward Design into a Constructionist Curriculum Design for IoT Skill Development in High Schoolers

Ananya Ipsita
Mechanical Engineering, C Design Lab
Purdue University
West Lafayette, Indiana, USA
aipsita@purdue.edu

Terrell Kendall Glenn
School of Mechanical Engineering - C Design Lab
Purdue University
West Lafayette, Indiana, USA
glenn3@purdue.edu

Disha Ajay Bhagwat
Purdue University
Indianapolis, Indiana, USA
dbhagwat@purdue.edu

Nielsen Pereira
Purdue UNiversity
West Lafayette, Indiana, USA
npereira@purdue.edu

Kylie A Peppler
School of Education / Creativity Labs
University of California
Irvine, California, USA
kpeppler@gmail.com

Karthik Ramani
School of Mechanical Engineering
Purdue University
West Lafayette, Indiana, USA
ramani@purdue.edu

## Abstract

Despite the growing demand for experienced Internet of Things (IoT) professionals across industrial establishments, most secondary education institutions do not offer a curriculum to empower students' knowledge and skills in IoT. Enrichment programs and vocational workshops can be considered as potential solutions to equip students with necessary IoT-related skills and assist them in planning and making conscious career choices. Through this research, we aim to utilize the principles of Backward Design and constructionism in designing an IoT curriculum for enrichment programs for high school students, while incorporating electro-mechanical concepts from electronics, programming, connectivity, and design. The curriculum was used to teach IoT concepts to 28 high school students during two enrichment programs. It was found that students with hardly any prior knowledge in IoT could acquire the necessary skills to design and prototype IoT applications.

## CCS Concepts

• **Computer systems organization** → **Robotic components**; • **Applied computing** → **Computer-assisted instruction**.

## Keywords

Learning Theories, Backward design, Constructionism, Internet of Things, Curriculum design, Vocational education;

**ACM Reference Format:**
Ananya Ipsita, Terrell Kendall Glenn, Disha Ajay Bhagwat, Nielsen Pereira, Kylie A Peppler, and Karthik Ramani. 2025. Adopting Backward Design into a Constructionist Curriculum Design for IoT Skill Development in High Schoolers. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems (CHI EA '25), April 26–May 01, 2025, Yokohama, Japan.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3706599.3720210


*CHI EA '25, Yokohama, Japan*

ACM ISBN 979-8-4007-1395-8/25/04
https://doi.org/10.1145/3706599.3720210

## 1 Introduction

With the advent of Industry 4.0, major transformations have been noticed in the industrial sector, wherein the workplace and processes have been revolutionized by Internet of Things (IoT) through enabling connectivity between man and machine [18, 40]. This rapidly growing technological transformation in workplace environments is placing new demands on employees' know-how, which apparently is resulting in a skills-gap in the industrial sector that requires immediate attention [41]. To address this concern, IoT-based solutions must be integrated into education and training curricula to prepare students and future employees for the evolving IoT skill requirements in the job market [17, 31].

Prior research has highlighted the mental preparedness of high school students for introducing IoT-related concepts and the benefits that come along with this exposure [32, 43]. However, it is often the case that the majority of secondary schools do not provide a curriculum to advance students' IoT knowledge and abilities [46]. Moreover, the current STEM curricula in these educational establishments do not have enough room to introduce IoT-related courses [23]. This encourages educators and experts to provide part-time training in IoT technology in the form of vocational courses and enrichment programs to better prepare their students for the future [17]. However, these efforts have not been accessible due to their limited guidance on developing outcome-oriented instruction for IoT education. Moreover, there is usually a lack of structure on how instruction relates to and scaffolds the prior experience of students in a gradually challenging manner. This research uses Backward Design to inform the design of an IoT-based outcome-oriented curriculum for an enrichment program for high school students.

IoT, usually defined by smart products, is typically based on combinations of electro-mechanical designs with advanced sensors, onboard intelligence, and connectivity [6]. These multidisciplinary concepts originate from the STEM-based curriculum and thus can be adopted easily into the learning methodologies for young novices in the field of IoT education [18]. Additionally, these concepts have been widely explored in MAKER-based education, enabling students to build computational thinking and problem-solving skills while applying the concepts to real-world problems [26]. Being able

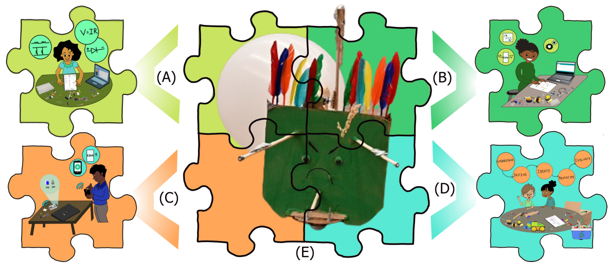



**Figure 1: IoT curriculum designed using Backward design model. Using scaffolding and learning by doing approach, the learning modules (A) Basic Electronic Components and Wiring, (B) Basics of Microcontroller Programming, (C) Connecting Devices to the Internet, and (D) Design of Physical Things and Interfaces, can provide necessary skills to E) design and implement Smart Toys and Robots.**

to relate the concepts to the real world can help enhance analytical thinking abilities and encourage lifelong learning. Therefore, our approach to building a MAKER-based learning curriculum was aimed at the development of smart tangible products by providing a foundation in electro-mechanical interconnected systems.

The use of robots is one of many approaches used to stimulate students' interest in studying electro-mechanical systems [21]. Bringing robots into classroom settings not only makes learning fun and exciting, but it also plays a key role in helping students learn how to solve problems and work collaboratively [16, 29]. Moreover, with increased accessibility to low-cost hardware and open-source software platforms such as Arduino, teaching techniques have been explored and found to be effective in enhancing students' interest in learning about electronics and computer programming to build and design robots [30].

With the above background, we present the curriculum design of ***Design and Prototyping of Smart Toys and Robots*** for teaching IoT concepts to high school students. The objective of the curriculum design is to provide students with a hands-on engineering design experience covering all aspects of IoT, from designing smart robots to the hardware and software that allow them to connect to the Internet, as shown in Figure 1. Project-based and structured tutorials are provided in the curriculum to enable students to learn and develop smart products while completing a variety of projects. A number of learning activities are designed, allowing students to express their opinions and ideas while gaining more knowledge through peer interaction and discussion. Finally, pilot testing of the curriculum was performed on 28 high school students during two enrichment programs, and it was observed that students with minimal prior knowledge were able to design and prototype IoT applications in the form of smart toys and robots.

Thus, our contributions are as follows:

(1) Development of the learning curriculum for an enrichment program to teach high school students about IoT-related concepts by using the Backward design approach from learning sciences, and
(2) Design of a MAKER-based instruction model as a part of the learning curriculum by using theories of constructionism

In order to meet the demands of a society that is constantly changing, we believe that this research will provide insights to the HCI community in building learning curricula that go beyond traditional classroom requirements.

## 2 Related Work

***Adoption of learning strategies in IoT education:*** As IoT drives digital transformation, the industrial revolution continues to reshape workplace conditions and skill requirements. Education must prepare future workers for these shifts. To equip students with relevant competencies beyond traditional classrooms, vocational training and enrichment programs often follow outcome-based education [19, 42]. This approach, known as Backward Design, is flexible and adaptable to individual student needs, making it both efficient and engaging [34, 45]. Studies show that Backward Design outperforms traditional lesson plans by accelerating learning, improving retention, and enhancing motivation [27, 39]. However, integrating established learning principles into IoT education remains underexplored in HCI literature, particularly in providing educators with guidelines for outcome-oriented instruction [1, 11]. Our research presents a detailed curriculum using the Backward Design model for an enrichment program that teaches IoT to high school students. This approach can help educators create relevant, timely content for computing education across different educational levels.

When designing learning content for short-term vocational training courses, it is crucial to engage young students and facilitate their understanding through hands-on experiences, allowing them to construct new knowledge. Prior research emphasizes aligning instructional methods with learning theories to enhance effectiveness in IoT education [2, 46]. Constructionism has been widely explored for designing instructional models that help students comprehend and build IoT systems [2, 8, 12, 24]. It posits that learners actively construct knowledge by creating tangible products, using prior knowledge as a foundation, and working collaboratively in social settings [22]. Hughes et al. implemented a constructionist framework in a week-long maker-oriented camp for K-12 students to teach IoT concepts [24]. The Make2Learn workshop promoted IoT education through free exploration and tinkering with creative materials [12]. Constructionist-driven educational toolkits have also been developed to facilitate IoT learning [8, 38]. These studies highlight how constructionist environments support students in acquiring IoT knowledge by designing and building artifacts while fostering interest in STEM. Given these benefits, our curriculum integrates constructionism principles into the IoT instructional model.

***Integrating IoT education with STEM and MAKER culture and Robots:*** IoT, according to the IETF, is the network of physical objects embedded with electronics, software, sensors, actuators, and connectivity to exchange data [37]. Understanding and implementing IoT systems require interdisciplinary skills in hardware and software [44]. Open-source platforms like ESP32 and Raspberry Pi, combined with programming tools such as C, Python, MIT App Inventor, and Ardublock, enable affordable and accessible IoT projects [13]. Instructors can use these platforms to design cost-effective curricula.

Despite accessible resources, introducing IoT concepts to high school students remains challenging [44]. Effective curriculum development is essential for integrating multidisciplinary IoT concepts and equipping students with relevant skills [23]. To address this, blending STEM and MAKER-based education into IoT learning frameworks has been proposed [9, 48]. This approach fosters creativity, critical thinking, and compensates for the lack of theoretical knowledge in MAKER-based courses [48]. Chen et al. [9] introduced a teaching model integrating STEM into IoT-MAKER courses, emphasizing interdisciplinary content, inquiry-based learning, iterative processes, and talent development.

Integrating computational devices (Raspberry Pis, Arduinos) and robots into MAKER-based settings enhances learning experiences [10, 35]. Learners engage in these activities with excitement and motivation [5, 16]. These techniques facilitate rapid skill acquisition while minimizing training costs. Thus, modular and affordable robotic devices should be incorporated into STEM education at secondary, tertiary, and vocational levels [4]. Extensive research has examined their use in primary and secondary education [49]. More recently, studies have explored IoToys, where toys not only interact with users but also form a network, commonly referred to as the Internet of Toys (IoToys) [50]. These toys aid IoT learning through creative play and have been used in workshops and curriculum development for effective learning environments [25].

## 3 Curriculum Design

Utilizing the Backward design model from learning sciences, the IoT curriculum was developed for high school students to impart IoT-related skills and knowledge. Backward design, as proposed by Wiggins and McTighe [33], involves creating instructional curriculum and assessments based on desired learning outcomes. Learning objectives are defined first, followed by the design of formative and summative assessments aligned with those objectives. The curriculum design ensures alignment between learning objectives, instruction, and assessment, leading to effective learning activities and outcomes. A MAKER-based instruction model, designed for this curriculum, was built on constructionist principles, where learners actively construct knowledge through hands-on experiences [22].

### 3.1 Design of the learning curriculum

The Backward design-based curriculum was developed by two university students with approximately five years of experience in IoT education and curriculum development (See Table 1 in Appendix). One has extensive experience creating IoT educational content for high school students, while the other has designed an IoT curriculum for an undergraduate course. Both have prior experience teaching workshops for high school students.

*3.1.1 Identify the desired result.* First learning outcomes were identified with an aim to equip high school students with the knowledge and skills to understand IoT concepts and create basic IoT systems, such as smart toys and robots. Students were expected to learn to design and troubleshoot electronic circuits, program microcontrollers, utilize sensors and actuators, connect devices to the internet, and be familiar with prototyping tools and resources. The learning concepts were subsequently worked upon by the experts by reflecting on their own understanding of the topics, and reviewing reputable resources, including peer-reviewed literature, government reports, and educational standards. These concepts were organized into four topic areas of (A) Electronics, (B) Programming, (C) Connectivity, and (D) Design to align with essential aspects of IoT technology [7]. The necessary learning objectives and skills were then determined to guide learners towards achieving the desired outcomes.

*3.1.2 Determine assessment techniques.* After establishing the learning objectives and skills, the subsequent step involved devising appropriate assessment strategies to effectively measure and foster the learning process at various stages. A combination of assessment of/for/as learning techniques were employed, aiming to create both formative and summative assessment methods [15]. Striking the right balance among these three types of assessments is crucial for enhancing the learning experience and overall outcomes [20].

*Assessment of learning* involves using evidence of students' learning to measure their performance against set outcomes and standards. For instance, in the curriculum, after completing the basic electronic prototyping modules, students' understanding of electronic circuitry was assessed through quizzes on debugging circuits.

*Assessment for learning* occurs when teachers utilize inferences about students' knowledge, understanding, and skills to inform their teaching. Examples of these assessment practices used in our curriculum include using Q&A, play-based activities during the classroom sessions, where the instructor actively engages with students to augment or clarify their understanding of concepts during the teaching process.

*Assessment as learning* involves students being their own assessors, monitoring their learning, asking questions, and using various strategies to gauge their knowledge and progress. An example of this in our curriculum is providing programming exercises to students after teaching them basic programming constructs, microcontrollers, and sensors. The assessment requires students to critically analyze the given problem and devise creative solutions independently or with minimal assistance.

*3.1.3 Design of instructional model.* After determining the assessment strategies, the final step was to plan learning experience and instruction using the following constructionist approaches.

*Scaffolding* in constructionism involves starting learning from accessible contexts and gradually progressing to more challenging ones [22]. We applied this by leveraging students' prior knowledge of basic electrical circuits asuch as power sources, conductors, and loads. Beginning with an introduction to basic electronics, we built up to complex topics like Ohm's Law. This sequential approach enables students to construct new knowledge based on their existing understanding. For example, learning electrical wiring in the first module prepares students for programming electro-mechanical circuits in the second.

*Experiential Learning:* This 'learn by doing' approach involves active participation of learners in connecting theories to real-world situations. It entails a continuous cycle of inquiry, reflection, analysis, and synthesis [28]. Our instruction model offers students hands-on learning experiences through physical and logical experiments. Technology, such as simulation environments like TinkerCAD, is integrated to create authentic learning contexts, allowing students to learn, build, and debug electronic circuits.

*Project-based learning:* This is a learner-centered approach where students spend time investigating and solving a complex problem or challenge, in order to build knowledge and skills [36]. This allows students to take more ownership of their learning by giving them more authority over what they learn. Our lesson plan includes numerous project-based activities, both at individual and group level, encouraging students to provide solutions to practical problems.

*Collaborative learning:* Social interaction can play a central role in the cognitive development in learners, simulate brainstorming, and also improves learners' motivation [47]. Collaborative environments require strategies for group size, instructional goals, communication, assignments, and evaluation. Medium-sized groups of 3-5 students are ideal; larger groups risk unequal participation, while smaller groups may lack diversity and limit divergent thinking. Setting clear group goals before starting assignments helps maintain focus and build a shared purpose. This approach was evident during the final group project, when the goal was to create robots that could connect to the internet and carry out specific tasks.

## 3.2 Design of Content

The instructions consisted of 4 learning modules and a final project.

*3.2.1 Learning Modules.* The content design comprised four modules: (A) Basic Electronic Components and Wiring, (B) Basics of Microcontroller Programming, (C) Connecting Devices to the Internet, and (D) Design of Physical Things and Interfaces. The lesson plan included various learning activities such as lectures, guided demonstrations, hands-on practicals, and assignments.

*Basic Electronic Components and Wiring:* This module aimed to teach students about fundamental circuit elements, circuit connections in series and parallel, electronic components for circuit prototyping, basic circuit analysis, creating and debugging circuits using TinkerCAD and real breadboards, and the laws governing circuit parameters that guide electronics circuit design.

*Basics of Microcontroller Programming:* This module introduces students to Arduino and programming basics. Hands-on experience involves building circuits with ESP32 microcontrollers and coding using the Arduino programming language. Topics covered include variables, loops, conditions, and functions to enable students to implement desired functions in robot programming. Coding challenges using TinkerCAD and real breadboards allow students to perform tasks virtually and in physical setups. Microcontroller programming and interfacing with sensors and actuators are also taught. Sensors like photo resistors and ultrasonic sensors for light intensity and distance detection are included. For example, a coding challenge involves using the ultrasonic sensor to create an electronic yardstick to measure distance from an obstacle. Actuators such as DC motors and Servo motors are covered, with coding challenges to control their speed and direction using microcontrollers.

*Connecting Devices to the Internet:* This module provided hands-on experiences to students to connect their devices to the Internet. Some examples of learning activities include: blinking LEDs through WiFi server, controlling the robots and toys using WiFi through the Blynk application on smartphone.

*Design of Physical Things and Interfaces:* This module introduces the design thinking process, covering problem understanding, definition, ideation, prototyping, and testing. Students learn about rapid prototyping methods like 3D printing and LASER cutting through guided demonstrations. The module concludes with a project-based activity where students work in teams to design, build, and test their prototypes for the final project, i.e., the Battlebots challenge.

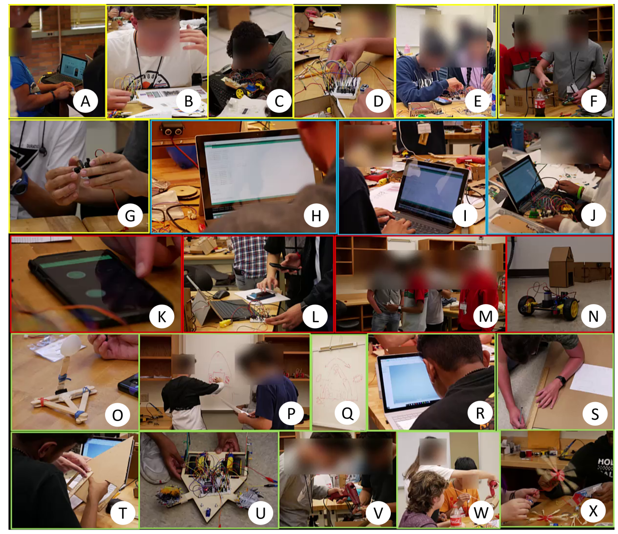
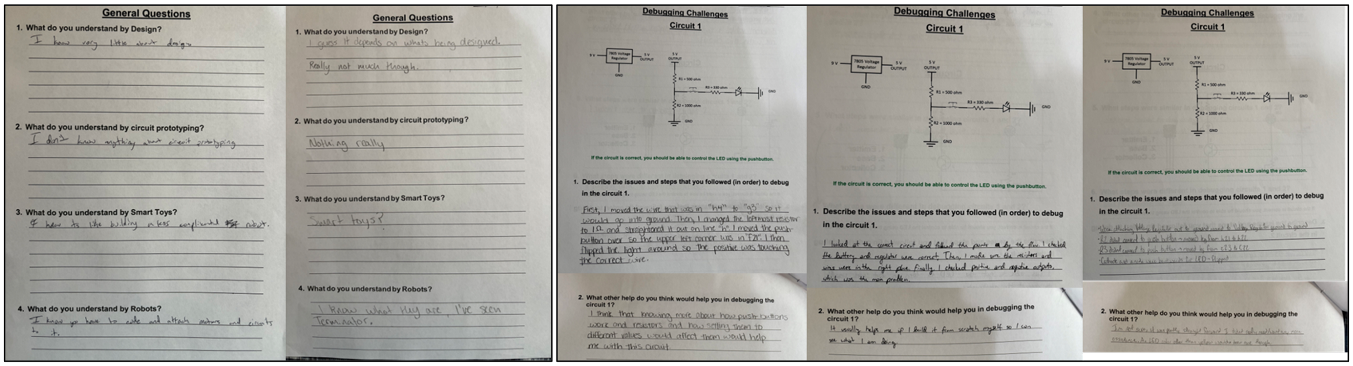



**Figure 2: Learning activities across four topic areas of Electronics (yellow borders), Programming (blue), Connectivity (red), and Design (green) include: (A) Replicate TinkerCAD circuits on real breadboards, (B) Construct circuits based on pinout diagrams, (C-E) Verify and build circuits on toy car assemblies, individually and in groups, (F) Use ultrasonic sensors, (G) Utilize servo motor actuators, (H) Program obstacle-sensing code with ultrasonic sensors, (I) Work on code individually, (J) Verify code and operation of toy car assemblies, (K) Design user interfaces with the Blynk IoT application, (L) Test circuit functionality via WiFi, (M, N) Control toy car assemblies using Blynk IoT, (O) Create a catapult from recyclable materials to launch a marshmallow projectile, (P, Q) Early design ideation and sketching for the Battlebot challenge, (R) Refine designs using digital prototyping, (S, T) Build low-fidelity prototypes, (U) Test the final Battlebot design and (V-X) Design and test weapon mechanisms**

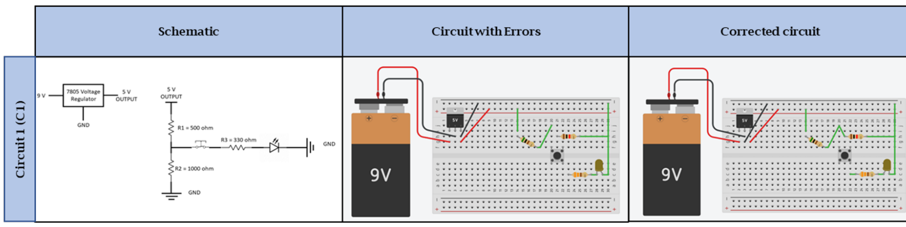
*3.2.2 Final Project:* The final project allows students to apply their acquired knowledge to design and construct smart toys or battlebots in groups of 3-5. The challenge is to create battlebots with the goal of eliminating the opponent's balloon located at the rear of their robot. The battlebots must adhere to certain restrictions, including the use of BlynkIoT for control, inclusion of at least one defense mechanism, and fitting within a 16 in X 16 in X 16 in cubic volume while static. The balloons should not be shielded, and each team has a license plate for identification. Specific limits are set for electronic components, like 2 ESP32 microcontrollers, 4 DC motors, and 3 Servo motors. Students can choose from materials like cardboard, hot glue guns, wooden skewers, popsicle sticks, recycled materials, and balloons to build their battlebots.

## 4 Pilot testing of the Curriculum

We conducted two 2-week enrichment programs for high school students with limited IoT experience (Figure 3A), with 18 students in the first and 10 in the second, ranging from grades 9-12. Three students belonged to the age range of 9-14 years, while the remaining 25 belonged to the age range of 15-17 years. The workshops ran

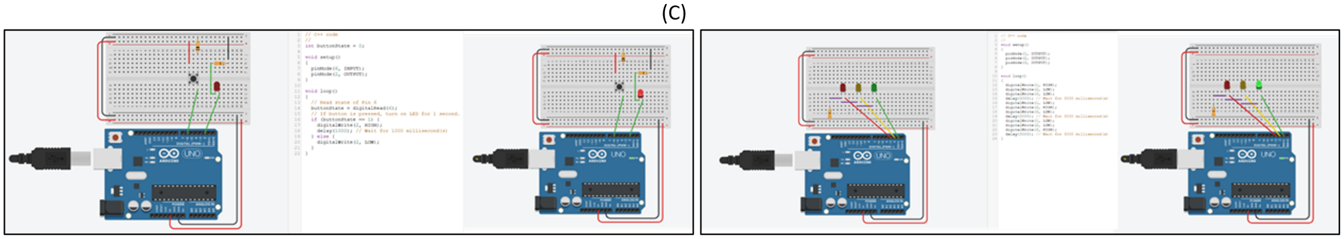



(A) (B)

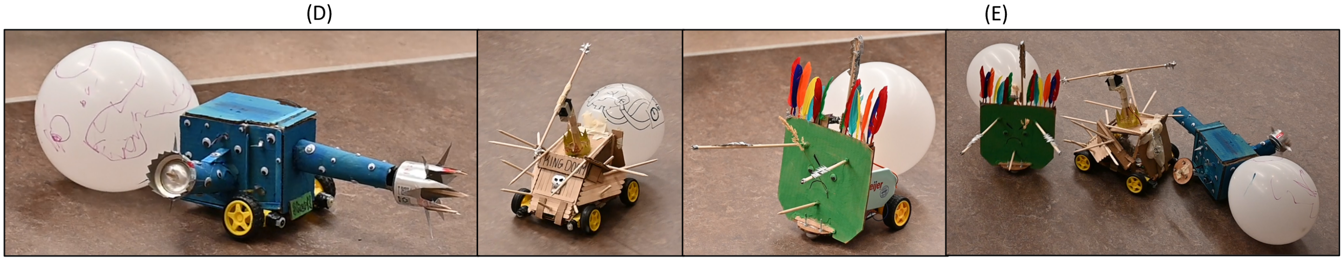



| Check for Errors | | |
|---|---|---|
| Name | Description | Result (%) |
| C1E1 | 7805 Voltage Regulator GND pin is connected correctly? | 94.12 |
| C1E2 | R3 (330 ohm) is connected to the other terminal of the pushbutton. | 100.00 |
| C1E3 | Polarity of LED is connected correctly? | 58.82 |

(C)



(D) (E)



(F)

**Figure 3: Assessment and artifacts from the enrichment program: (A) Exercise showing limited experience of students in the domain of IoT before the workshop, (B, C) Assessment using debugging quizzes, student reflection and instructor evaluation, (D) Formative assessments for learning programming constructs like if-else, (E) Connection of concepts to real world application (In this case, a LED display using while loop has similar connections to a Traffic signal), and (F) Battlebots during the second enrichment program**

five days a week, with three-hour daily sessions. Each program had a main instructor, who was also one of the curriculum designers.

The first six days of each program included lecture sessions, guided demonstrations, and individual projects to teach various multi-disciplinary skills related to prototyping circuit hardware, programming, and connecting devices to the internet. The last four days involved students working in groups to design and build smart toys for the final Battlebots challenge. A survey conducted during the last day of the second program received five responses. The 50 percent response (5 out of 10 students) is possibly due to students focused on the competition. The pilot testing procedures for the curriculum were approved as exempt under the IRB protocols.

***Study results:*** The instructor successfully covered all planned topics during the first week of the workshop. Various assessment methods provided insights into students' understanding. For example, as shown in Figure 3B, C, the quizzes on debugging circuits (assessment of learning) were evaluated by the instructor to check if the students' understood the concepts of basic electronics. Each day's formative assessments (assessment as learning) encouraged students to assess their own progress. For example, in one TinkerCAD activity, students received a rough draft of an if-else program and had to modify it to blink an LED for two seconds. By verifying their corrections through simulation, students saw tangible results as shown in Figure 3D.

HCI literature highlights how maker-based approaches, FabLabs, and design tinkering help students develop familiarity with IoT concepts [11]. As shown in Figure 2, the maker-based activities provided students with an efficient and engaging way to reinforce their understanding of the learned topics while constructing tangible artifacts. These hands-on activities ranged across the four major areas, including electronics, programming, connectivity, and design, that constitute an IoT system. Furthermore, the progressive scaffolding of the learning content helped students gradually develop their skills and knowledge during the first week of the workshop. By the end of the course, participants could conceptualize, prototype, and build basic IoT applications using Arduino, electronic components, and network connections.

During the survey, students identified familiar electronic components, including resistors, LEDs, breadboards, sensors, motors, motor drivers, transistors, and diodes. They confirmed building circuits, commenting: "*Turns on LEDs*", "*Drives RC car*", and "*H circuit*". All participants gained programming experience with Arduino, mentioning familiarity with constructs like "*If-else for digitalWrite, Serial.println*". When asked to comment on the programs they wrote, some students responded: "*Robot control program, Morse code, Wi-Fi control, light show*", "*To make the LED light up, to make the car move*". They learned how to connect their devices to the Internet, responding: "*With code*", "*By using the example, you can stream the device to Wi-Fi using the IP address*", "*By programming on the computer first and creating a web server with a given address*", "*Using Blynk*". Students learned about IoT, identifying key modules such as "*Wi-Fi, and all hardware needed to receive and transmit Wi-Fi signals*", "*The different modules needed are the data streams and buttons and device IDs*". When asked to briefly explain what smart robots they made during the workshop, some students responded: "*Making a car and having it move*".

The instruction modules facilitated connections between IoT concepts and real-world applications. For example, after completing a TinkerCAD activity on a while loop to blink three LEDs (red, green, and yellow for 2, 1, and 2 seconds, respectively), students were asked for real-world examples as shown in Figure 3E. One quickly responded: "*Traffic signal*". Such connections can reinforce long-term learning in learners. Digital platforms like TinkerCAD also enabled students to build and debug circuits and code before actually implementing on real breadboards [3]. When asked to comment on what they used it for, some students responded: "*Circuit simulations*", "*Virtual circuits*", "*For coding*".

As shown in Figure 2 P-X, student activities during the final projects ranged from design ideation to low-fidelity prototyping to design refinement and iteration to final concept generation, prototyping, and testing. The problem statement ensured deliverables concerning different aspects of the IoT learning module that were covered during the first week. The successful execution of the final BattleBots shows that the student groups developed understanding of the individual modules of IoT technology. The battlebots built during the second enrichment program are shown in Figure 3F. When asked to comment about the workshop, some students responded: "*It was great and I liked it so much*", "*It is a nice workshop, especially for people who are interested in STEM classes*".

Collaborative work allowed them to share ideas and learn progressively, building on prior knowledge and insights. Students actively participated, presenting their thoughts, asking relevant questions, and engaging in continuous reflection and progress. Their ability to apply concepts in their final projects demonstrated their problem-solving skills. Survey results confirmed that they participated in group projects. When asked to comment on the projects they worked on in teams, some students responded: "*RC car*", "*Making a car*". It was also observed by the instructor that when students would get done with their own assignments, they went out of their way to help his teammates who were faced with some trouble.

## 5 Conclusion

With the Internet of Things (IoT) revolutionizing the industrial sector through facilitating connectivity between man and machines, there is a need for increased awareness about the technology in the minds of young adolescents to prepare them for the changing workplace. Utilizing the principles from Backward Design and constructionism, we presented an IoT curriculum for an enrichment program to teach high school students about IoT-related concepts and skills. Pilot-testing the curriculum was conducted with 28 students during two enrichment programs. Although the results from the pilot testing indicate direct short-term results on the skill development of novice high schoolers, we acknowledge that the study results are limited in terms of sample size and rigorous techniques to evaluate the learning gains. Further analysis is required to quantitatively evaluate the effect of the design methods on the development of learning outcomes. Furthermore, future work should also consider longitudinal studies to evaluate the lasting impact of the curriculum.

As a scope of future work, we also recommend exploring the effect of the proposed design methods on the development of Maker's mindset in learners. As shown by prior work, this mindset that encourages (A) curiosity and exploration, (B) collaboration, communication and sharing of ideas, (C) making mistakes and reflecting on them, and (D) critical thinking and problem-solving, is best developed in a learner in an interdisciplinary setting [14]. With an interplay of multidisciplinary concepts, we believe that the IoT curriculum can nurture these skills by making learners the protagonists of the learning experience. We hope that our work provides insights to educators and researchers to explore the use of such pedagogical approaches for developing early interest in STEM careers and beyond.

## Acknowledgments

We thank the reviewers for their invaluable feedback. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding agency.

## References

[1] Pramod Abichandani, Vidhusheidhaa Sivakumar, Deepan Lobo, Craig Iaboni, and Prateek Shekhar. 2022. Internet-of-things curriculum, pedagogy, and assessment for stem education: A review of literature. *Ieee Access* 10 (2022), 38351–38369.

[2] Mariana Aki Tamashiro. 2021. How do we teach Emerging Technologies in K-9 Education? Using design fiction and constructionist approaches to support the understanding of emerging technologies' societal implications in formal K-9 education. In *Interaction Design and Children*. 637–640.

[3] V Alavés-Baeza and JJ Galiana-Merino. 2022. Expanding the possibilities of the electronics laboratory with TinkerCAD. In *INTED2022 Proceedings*. IATED, 7777–7783.

[4] Saira Anwar, Nicholas Alexander Bascou, Muhsin Menekse, and Asefeh Kardgar. 2019. A systematic review of studies on educational robotics. *Journal of Pre-College Engineering Education Research (J-PEER)* 9, 2 (2019), 2.

[5] Raidell Avello, Jari Lavonen, and Miguel Zapata-Ros. 2020. Coding and educational robotics and their relationship with computational and creative thinking. A compressive review. *Revista de Educación a Distancia (RED)* 20, 63 (2020).

[6] Ranbir Singh Batth, Anand Nayyar, and Amandeep Nagpal. 2018. Internet of robotic things: driving intelligent robotics of future-concept, architecture, applications and technologies. In *2018 4th International Conference on Computing Sciences (ICCS)*. IEEE, 151–160.

[7] Barry Burd, Lecia Barker, Félix Armando Fermín Pérez, Ingrid Russell, Bill Siever, Liviana Tudor, Michael McCarthy, and Ian Pollock. 2018. The internet of things in undergraduate computer and information science education: exploring curricula and pedagogy. In *Proceedings Companion of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education*. 200–216.

[8] Attapan Chan-In and Arnan Sipitakiat. 2020. Designing an Educational Internet of Things Toolbox: A case-study of a learner-centric tool design. In *Proceedings of the FabLearn 2020-9th Annual Conference on Maker Education*. 90–93.

[9] Rongjun Chen, Yani Zheng, Xiansheng Xu, Huimin Zhao, Jinchang Ren, and Hong-Zhou Tan. 2020. STEM teaching for the Internet of Things maker course: a teaching model based on the iterative loop. *Sustainability* 12, 14 (2020), 5758.

[10] Pao-Nan Chou. 2018. Skill development and knowledge acquisition cultivated by maker education: Evidence from Arduino-based educational robotics. *EURASIA Journal of Mathematics, Science and Technology Education* 14, 10 (2018), em1600.

[11] Geert de Haan. 2016. A hands-on approach to making in the internet of things and creative technology. In *Proceedings of the European Conference on Cognitive Ergonomics*. 1–4.

[12] Monica Divitini, Michail N Giannakos, Simone Mora, Sofia Papavlasopoulou, and Ole Sejer Iversen. 2017. Make2Learn with IoT: Engaging children into joyful design and making of interactive connected objects. In *Proceedings of the 2017 Conference on Interaction Design and Children*. 757–760.

[13] Michael V Doran and George W Clark. 2018. Enhancing robotic experiences throughout the computing curriculum. In *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*. 368–371.

[14] Dale Dougherty. 2013. The maker mindset. *Design, make, play: Growing the next generation of STEM innovators* (2013), 7–11.

[15] Lorna M Earl and M Steven Katz. 2006. *Rethinking classroom assessment with purpose in mind: Assessment for learning, assessment as learning, assessment of learning*. Manitoba Education, Citizenship & Youth.

[16] Amy Eguchi. 2017. Bringing robotics in classrooms. In *Robotics in STEM education*. Springer, 3–31.

[17] Hicham El Mrabet and Abdelaziz Ait Moussa. 2021. IoT-school guidance: A holistic approach to vocational self-awareness & career path. *Education and Information Technologies* 26, 5 (2021), 5439–5456.

[18] Kerem Elibal and Eren Özceylan. 2020. A Review of Logistics 4.0 Literature for Curriculum Development for Vocational High Schools. In *Proceedings of the 2nd African International Conference on Industrial Engineering and Operations Management Harare, Zimbabwe*.

[19] Jan Emory. 2014. Understanding backward design to strengthen curricular models. *Nurse Educator* 39, 3 (2014), 122–125.

[20] National Forum for the Enhancement of Teaching and Learning in Higher Education. 2017. Expanding our Understanding of Assessment and Feedback in Irish Higher Education.

[21] Mariano Garduño-Aparicio, Juvenal Rodríguez-Reséndiz, Gonzalo Macias-Bobadilla, and Suresh Thenozhi. 2017. A multidisciplinary industrial robot approach for teaching mechatronics-related courses. *IEEE Transactions on Education* 61, 1 (2017), 55–62.

[22] Idit Ed Harel and Seymour Ed Papert. 1991. *Constructionism.* Ablex Publishing.

[23] Jing He, Dan Chia-Tien Lo, Ying Xie, and Jonathan Lartigue. 2016. Integrating Internet of Things (IoT) into STEM undergraduate education: Case study of a modern technology infused courseware for embedded system course. In *2016 IEEE frontiers in education conference (FIE)*. IEEE, 1–9.

[24] Janette Hughes, Jennifer Anne Robb, and Margaret Lam. 2020. Designing and Learning with IoT in a Passion-Based Constructionist Context. Springer, 760–771.

[25] PIRITA Ihamäki and K Heljakka. 2018. Smart, skilled and connected in the 21st century: Educational promises of the Internet of Toys (IoToys). In *Proceedings of the 2018 Hawaii university international conferences, arts, humanities, social sciences & education, Prince Waikiki Hotel, Honolulu, Hawaii*. 1–19.

[26] Yu-Lin Jeng, Chin-Feng Lai, Sheng-Bo Huang, Po-Sheng Chiu, and Hua-Xu Zhong. 2020. To cultivate creativity and a maker mindset through an internet-of-things programming course. *Frontiers in Psychology* 11 (2020), 1572.

[27] Lynn Marie Kelting-Gibson. 2003. *Preservice teachers' planning and preparation practices: a comparison of lesson and unit plans developed using the backward design model and a traditional model.* Montana State University.

[28] David A Kolb. 2014. *Experiential learning: Experience as the source of learning and development.* FT press.

[29] Beng Yong Lee, Lee Hung Liew, Mohd Yazid Bin Mohd Anas Khan, and Azlina Narawi. 2020. The effectiveness of using mbot to increase the interest and basic knowledge in programming and robotic among children of age 13. In *Proceedings of the 2020 The 6th International Conference on E-Business and Applications*.

[30] Dimitrios Loukatos, Eleftherios Chondrogiannis, and Konstantinos G. Arvanitis. 2020. A Low-Cost Example, Combining MIT App Inventor, Arduino Specific Components and Recycled Materials to Foster Engineering Education. In *24th Pan-Hellenic Conference on Informatics*. 367–371.

[31] Petre-Daniel Mătăsaru, Luminiţa Scripcariu, and Felix Diaconu. 2018. Analysis of proposed innovative methods for training students in IoT and computer networks within the frame of current trends in technical and vocational education. In *2018 International Conference and Exposition on Electrical And Power Engineering (EPE)*.

[32] Anna Mavroudi, Monica Divitini, Francesco Gianni, Simone Mora, and Dag R Kvittem. 2018. Designing IoT applications in lower secondary schools. In *2018 IEEE Global Engineering Education Conference (EDUCON)*. IEEE, 1120–1126.

[33] Jay McTighe and Ronald S Thomas. 2003. Backward Design for Forward Action. *Educational leadership* 60, 5 (2003), 52–55.

[34] Jay McTighe and Judy Willis. 2019. *Upgrade your teaching: Understanding by design meets neuroscience.* ASCD.

[35] Aditya Mehrotra, Christian Giang, Laila El-Hamamsy, Anthony Guinchard, Amaury Dame, Géraldine Zahnd, and Francesco Mondada. 2021. Accessible Maker-Based Approaches to Educational Robotics in Online Learning. *IEEE Access* 9 (2021), 96877–96889.

[36] Paniti Netinant, Punnipa Narad, and Meennapa Rukhiran. 2021. A case study of project-based learning on Internet of Things course. In *Proceedings of the 7th International Conference on Frontiers of Educational Technologies*. 126–131.

[37] The Internet of Things at the IETF. 2022. The Internet of Things at the IETF.

[38] Am-Suk Oh. 2018. A Study on Maker Kit based on Internet of Things. *International Information Institute (Tokyo). Information* 21, 1 (2018), 231–238.

[39] Mónica Ontaneda Rea and Jorge Luis Sánchez Román. 2018. Implementing backward design to improve students' academic performance in EFL classes. *Centro Sur* 2, 2 (jul. 2018), 37–49.

[40] Maria Rita Palattella, Mischa Dohler, Alfredo Grieco, Gianluca Rizzo, Johan Torsner, Thomas Engel, and Latif Ladid. 2016. Internet of things in the 5G era: Enablers, architecture, and business models. *IEEE journal on selected areas in communications* 34, 3 (2016), 510–527.

[41] Shaoping Qiu, Malini Natarajarathinam, Michael D Johnson, and Elizabeth A Roumell. 2021. The Future of Work: Identifying Future-ready Capabilities for the Industrial Distribution Workforce. In *2021 ASEE Virtual Annual Conference Content Access*.

[42] Heather L Reynolds and Katherine Dowell Kearns. 2017. A planning tool for incorporating backward design, active learning, and authentic assessment in the college classroom. *College Teaching* 65, 1 (2017), 17–27.

[43] Natalia Spyropoulou, Dimitrios Glaroudis, Athanasios Iossifides, and Ioannis D Zaharakis. 2020. Fostering Secondary Students' STEM Career Awareness through IoT Hands-On Educational Activities: Experiences and Lessons Learned. *IEEE Communications Magazine* 58, 2 (2020), 86–92.

[44] Ana-Maria Suduc, Mihai Bîzoi, and Gabriel Gorghiu. 2018. A Survey on IoT in Education. *Romanian Journal for Multidimensional Education/Revista Romaneasca pentru Educatie Multidimensionala* 10, 3 (2018).

[45] Carol A Tomlinson and Jay McTighe. 2006. *Integrating differentiated instruction & understanding by design: Connecting content and kids.* ASCD.

[46] Maarten Van Mechelen, Rachel Charlotte Smith, Marie-Monique Schaper, Mariana Tamashiro, Karl-Emil Bilstrup, Mille Lunding, Marianne Graves Petersen, and Ole Sejer Iversen. 2023. Emerging technologies in K–12 education: A future HCI research agenda. *ACM Transactions on Computer-Human Interaction* (2023).

[47] L Vygotsky. 2019. Collaborative learning. *Collaboration, communications, and critical thinking: A STEM-inspired path across the curriculum* 43 (2019).

[48] Xingwei Wang, Wenwen Xu, and Liang Guo. 2018. The status quo and ways of STEAM education promoting China's future social sustainable development. *Sustainability* 10, 12 (2018), 4417.

[49] Ioannis D Zaharakis, Nicolas Sklavos, and Achilles Kameas. 2016. Exploiting ubiquitous computing, mobile computing and the internet of things to promote science education. In *2016 8th IFIP International Conference on New Technologies, Mobility and Security (NTMS)*. IEEE, 1–2.

[50] Bieke Zaman, Maarten Van Mechelen, and Lizzy Bleumers. 2018. When toys come to life: considering the internet of toys from an animistic design perspective. In *Proceedings of the 17th ACM Conference on Interaction Design and Children*.

| Learning Outcomes | Learning Objectives | Skill Identification for IoT prototyping | Skill Assessment Criteria |
|---|---|---|---|
| · Build and debug electronic circuits | · Identify different types of circuit - open vs. closed, and connections - series vs. parallel<br>· Identify and locate circuit components - Power Source, Conductors, Loads (resistors, LEDs, motors, Buzzer, and Potentiometer) and additional components (button, voltage regulators, capacitor, transistor, motor driver, breadboard)<br>· Understand the imporatnce of circuit components - e.g. breadboard<br>· Understand laws governing the relation between different circuit elements (Ohm's Law) and circuit fundamentals - e.g. Volatge division<br>· Learn color coding in resistors to calculate resistance values<br>· Follow safety practices related to wiring during prototyping basic electronics circuits<br>· Learn about polarity and pin configuration of electronic components<br>· Learn virtual electronic simulation and prototyping using TinkerCAD<br>· Learn about circuit symbols and reading schematic diagrams<br>· Learn about debugging tips and tricks in basic circuitry and wiring | · Ability to read component values directly from electronic components; e.g., color coding of resistors, capacitance values<br>· Ability to connect components based on the pin configuration, polarity of terminals, etc.<br>· Knowledge about the fundamental principles governing electron flow in any circuit<br>· Ability to identify and select electronic components based on the circuit requirements<br>· Ability to debug circuits in case the circuit does not function properly<br>· Ability to read and draw circuit schematics diagram | · Connect electronic components correctly in the circuit while taking care of polarity, component values, pin configuration, etc.<br>· Debug non-functioning circuits to find errors in the TinkerCAD circuits<br>· Completing the circuit to make it work and show output to the coaches<br>· Locate and identify the learning elements (e.g. resistors, polarity of terminals, etc.)<br>· Read/measure/calculate the component values correctly<br>· Duplicate schematic using virtual TinkerCAD or real breadboard prototyping |
| · Program logic for electromechanical devices using microcontrollers, sensors, actuators | · Understand basics about microcontrollers, sensors and actuators<br>· Learn about Arduino - IDE, boards, basic functions<br>· Learn about programming contructs - variables, datatypes, controls (if-else), loops (for, while), functions<br>· Understand type of signals (analog vs. digital), logic levels (HIGH/ON/5V, LOW/OFF/0V), Analog-to-Digital Conversion (ADC) | · Knowledge about the ESP32 features, I/O pins and interfacing<br>· Knowledge about basic Arduino functions to read/write data from/to ESP32/Serial Monitor/sensors/actuators<br>· Knowledge about the binary system, and its relevance in digital electronics<br>· Ability to read, process and make sense of raw sensor data<br>· Ability to apply computational logic using programming constructs to code functions to control actuators | · Completing the script or pseudocode to get required outputs<br>· Configure the Arduino IDE and board correctly for ESP32<br>· Configure the Serial Monitor correctly<br>· Compile the code successfully without errors and upload to ESP32 board<br>· Connect the sesors and actuators correctly to the ESP32 I/O pins |
| · Connect physical things over WiFi | · Control ESP32 from web interface using HTTP request and headers<br>· Understand usage of BlynkIoT application | · Knowledge about the built-in WiFI capabilities of ESP32<br>· Ability to use ESP32 to connect and control devices wirelessly<br>· Ability to use ESP32 as client using BlynkIoT | · Control the devices over WiFi as expected and show the outputs<br>· Configuration of BlynkIoT correctly<br>· Explain understanding of code and circuit |
| · Design physical things (smart toys) and UI controls | · Learn about the design thinking process<br>· Understand the importance of prototyping<br>· Learn about different low fidelity prototyping techniques<br>· Learn about different methods used for rapid prototyping<br>· Design user interfaces for IoT applications using BlynkIoT | · Knowledge about the various techniques used in rapid prototyping methods - addtive and subtractive<br>· Ability to use the low fidelity prototyping techniques in the early design phase<br>· Ability to design UIs using contol widgets in BlynkIoT | · Explain the design of physical structures or virtual controls when asked<br>· Correctly assemble the components and parts to make the final design<br>· Use sketches in the early design phase |

**Learning Modules**

| Lectures | Practice |
|---|---|
| · Circuit fundamentals, types of circuit (open vs. closed), connections in circuit (series vs. parallel), circuit components (power source, conductors, and loads)<br>· Basics about the different learning elements involved in basic electronics circuitry:<br>A) Breadboard - importance, connections, and how to connect components on a breadboard<br>B) Resistors - sybmols, color coding, Ohm's Law, connections in series and parallel<br>C) Potentiometer - symbols, pin configuration, working, voltage division<br>D) LEDs - polarity, symbols, safety practices while connecting LEDs in circuit<br>E) Motors - importance, terminals, effects of reversal of terminals connection to power source<br>F) Switch and button - relation to binary, importance, symbols<br>G) Transistors - types (npn vs. pnp), pin configuration, symbols, importance in digital electronics<br>· Using TinkerCAD circuits and simulation to prototype and test circuit designs | · Locate and identify the learning elements in the component box (e.g. resistors, capacitors)<br>· Locate and identify the polarity and pin configuration of the electronic components<br>· Find resistance values of resistors using color coding, Ohm's Law, and in series/parallel connections<br>· Prototype circuits using TinkerCAD and/or real breadboard:<br>A) Turn ON an LED using battery and proctective resistor to prevent current overload through the LED<br>B) Connect LEDs in series and parallel<br>C) Using potentiometer to change light intensity of an LED and speed of a motor<br>D) Change the direction of rotation of motors by reversing polarity of the terminals<br>E) Use transistor as a switch -Tturn an LED ON and OFF using pushbutton and a transistor<br>F) Using a switch, kill all functions of a robot<br>· Debug electronics circuit to find errors. Fix the errors to make the circuit work |
| · Introduction to Arduino IDE, boards, ESP32 features - I/O pins, builtin WiFi<br>· Arduino IDE - Configuration Settings, structure of Arduino Sketch, basic Arduino functions<br>· Serial Monitor - Baud Rate settings and functions<br>· Basic programming constructs - variables and data types, comments (single line vs. multi line), controls (if - else), loops (for vs. while), operators (comparators, increment, decrement)<br>· Types of sensors (analog vs. digital), corresponding Arduino pins and functions, ADC<br>A) Photoresistors - working and principle, Interfacing with ESP32<br>B) Ultrasonic Distance Sensors - working and principle, pin configuration, Interfacing with ESP32<br>· Actuators, their importance, commonly used actuators in hobbyist robotics:<br>A) DC Motors, H-bridge circuits, Motor drivers (L293D), pin configuration, Interfacing with ESP32<br>B) Servo motor - understand the working of servo motors, pin configuration, Interfacing with ESP32 | · Program and prototype electromechanical circuits using TinkerCAD and/or real breadboard:<br>A) Blinking LEDs using ESP32 in delayed intervals and printing status to Serial Monitor<br>B) Using if-else controls, turn a LED ON when a pushbutton is pressed<br>C) Using loops, program a traffic light simulator, turn a buzzer ON for 10 times<br>D) Using a photoresistor, create a mock prototype for smart lighting<br>E) Using Ultrasonic Distance Sensors, design an electronic yardstick and print out distance ranges<br>F) Controlling direction and speed of motors using Arduino<br>G) Control the movement of a toy car by implementing functions - forward, backward, left, right, stop.<br>H) Change the shaft position of a servo motor from 0 to 180 degrees, and then from 180 to 0 degrees. |
| · Configuring BlynkIoT to control ESP32 using smartphone over WiFi | · Wirelessly control LED, toy car by pressing buttons on the BlynkIoT application<br>· Controlling LED pins wirelessly from a web interface using ESP32 |
| · The various stages of design thinking process, starting from understanding the users' needs, to defining the problem statement, ideating different approaches for tackling the problem, prototyping solutions, and finally testing the prototypes to check if the design suits requirements<br>· rapid prototyping methods used in engineering - additive, subtractive<br>· demonstration of 3D printer, laser cutter in action, understand the working | · Assemble a toy car from scratch using components and parts.<br>· Design UIs using control widgets in BlynkIoT application |

**Table 1: Design and development of the Curriculum using the backward design principles**