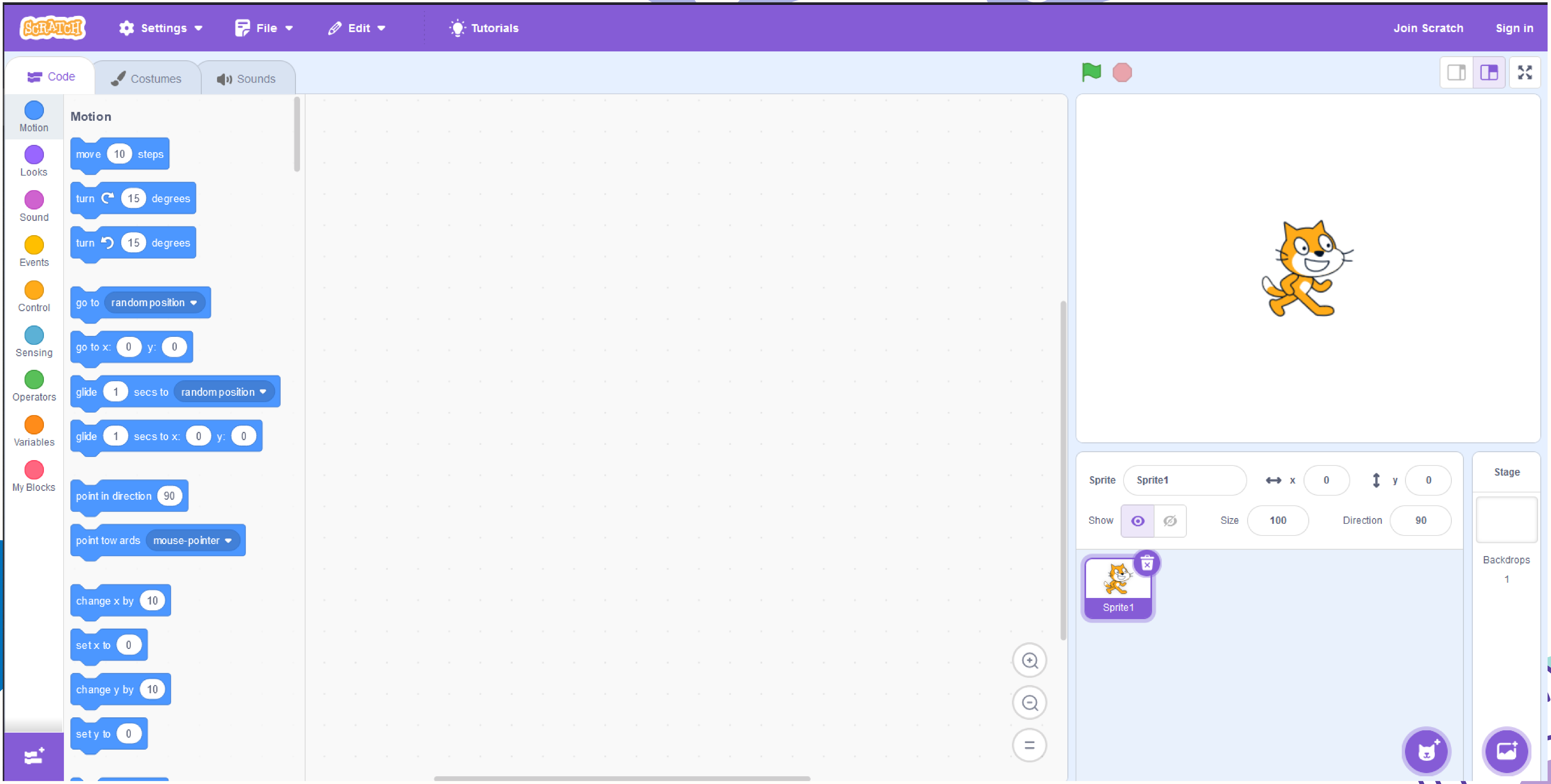
CHATSCRATCH:

AN AI-AUGMENTED SYSTEM TOWARD AUTONOMOUS Visual Programming Learning for Children Aged 6-12

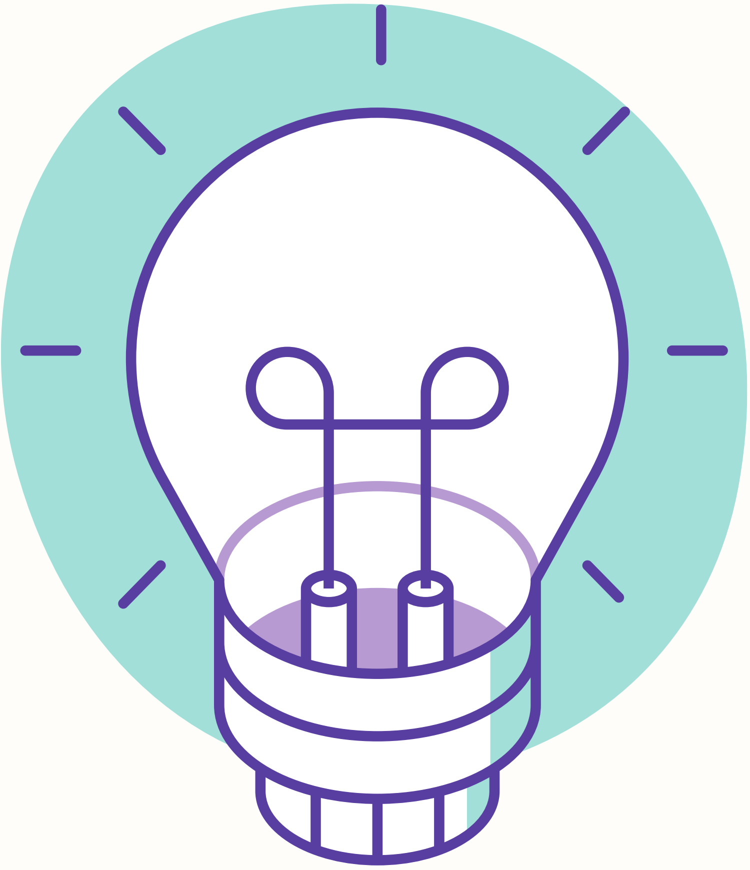
Written by Liuqing Chen, Shuhong Xiao, Yunnong Chen, Ruoyu Wu, Yaxuan Song, Lingyun Sun

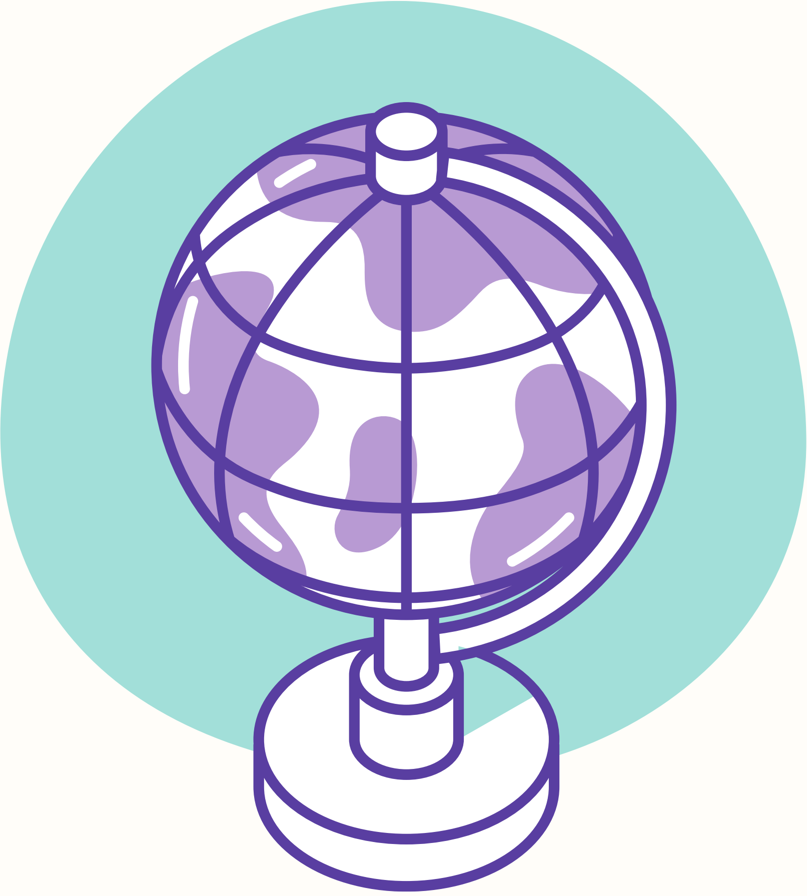
Presented by Zubair Alnoor & Neha Prabu

INTRODUCTION


Scratch
Settings
File
Edit
Tutorials
Join Scratch
Sign in
Code
Costumes
Sounds
Motion
Looks
Sound
Events
Control
Sensing
Operators
Variables
My Blocks
Motion
move 10 steps
turn 15 degrees
turn 15 degrees
go to random position
go to x: 0 y: 0
glide 1 secs to random position
glide 1 secs to x: 0 y: 0
point in direction 90
point towards mouse-pointer
change x by 10
set x to 0
change y by 10
set y to 0
Sprite
Sprite1
x
0
y
0
Show
Size
100
Direction
90
Sprite1
Stage
Backdrops
1

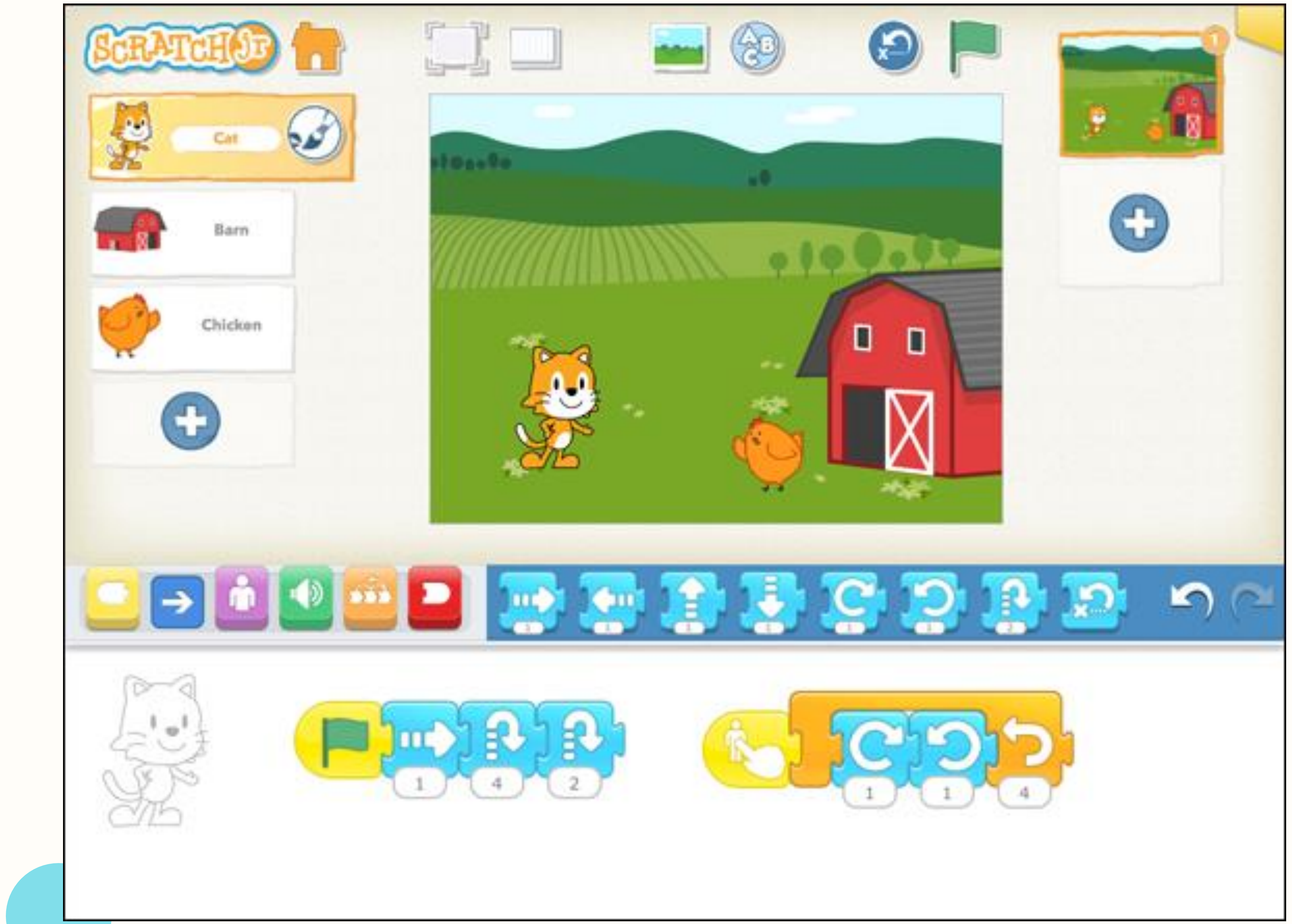
Limitations of Scratch



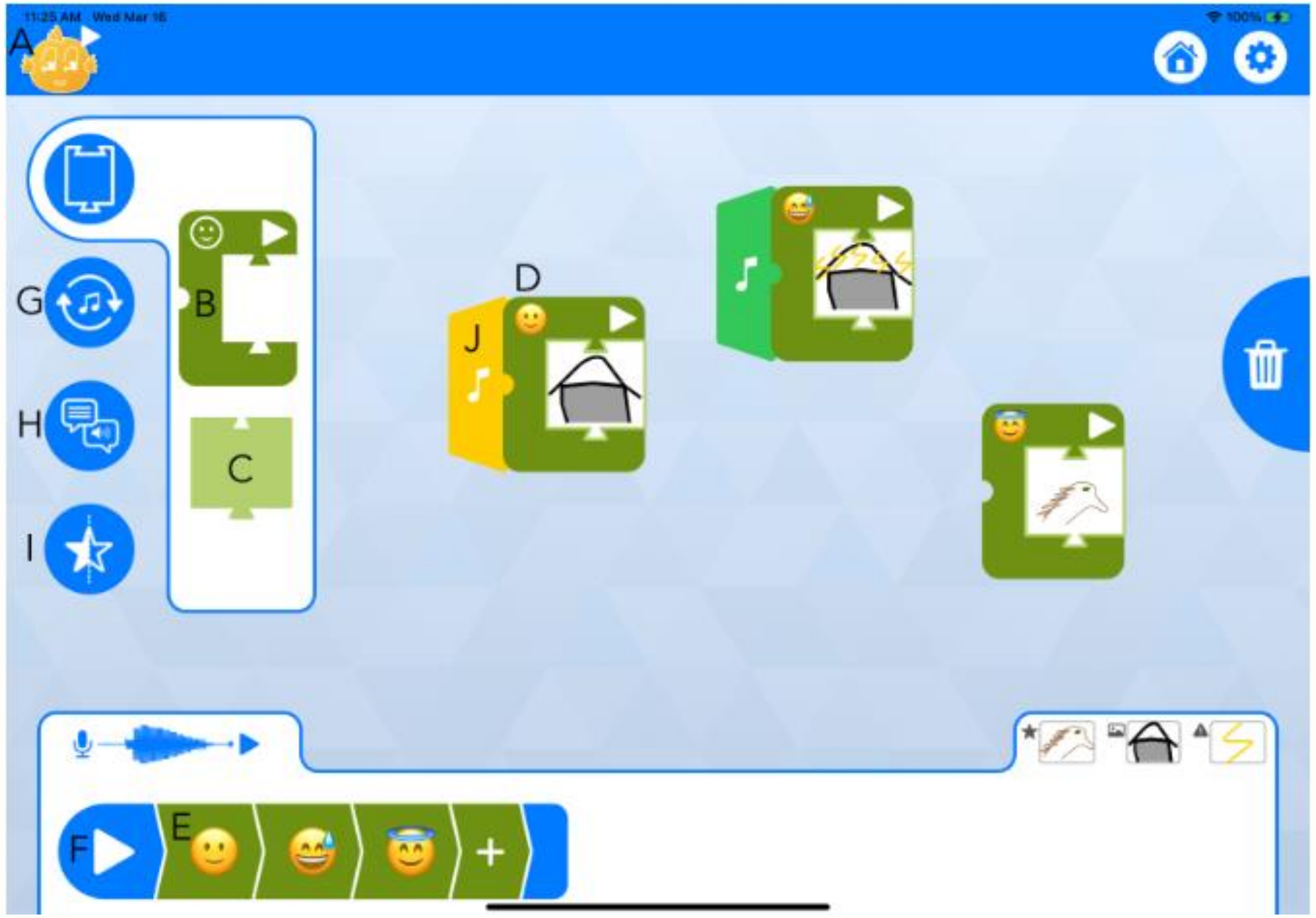


- Computational thinking is starting to get introduced at a much younger age.
- Children are limited by:
    - Literacy Skills
    - Arithmetic capabilities
    - Fine-motor skills
- Tailored for classroom environments, guided by professional educators, and a structured curriculum.

Possible Alternatives?



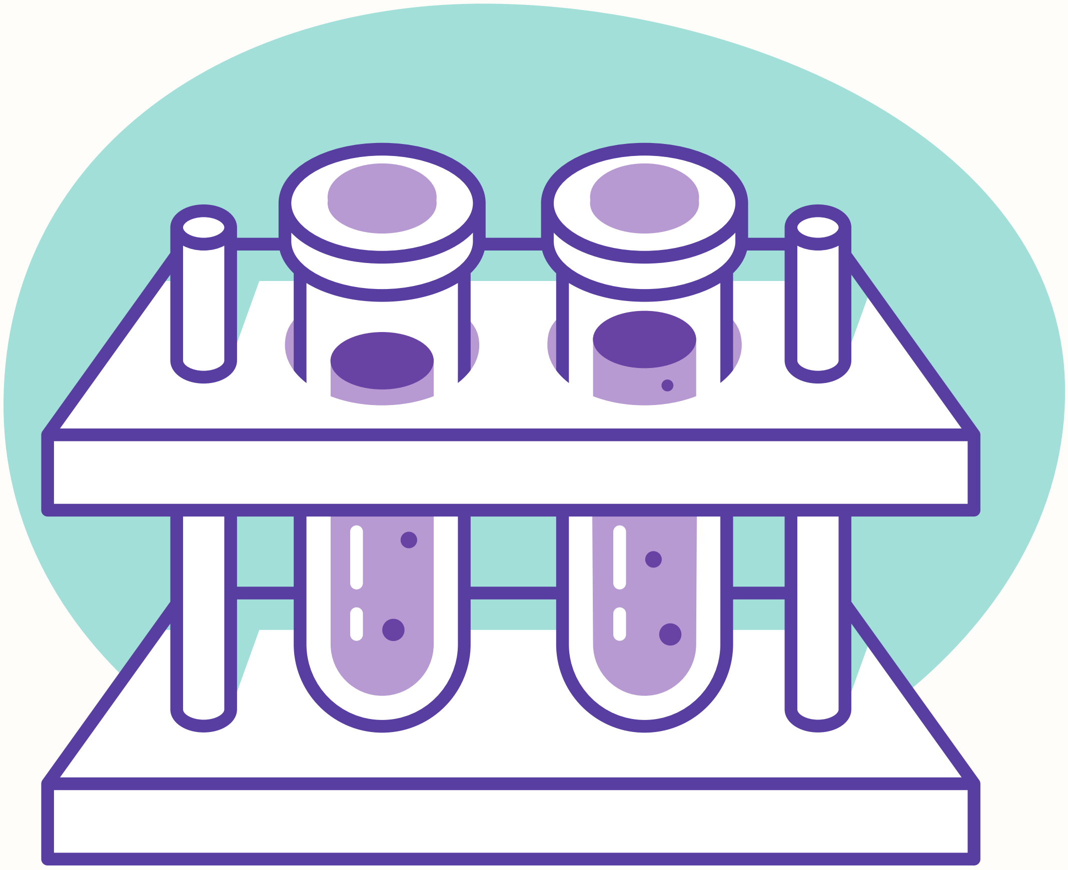
https://www.scratchjr.org/learn/interface



https://doi.org/10.1145/3544548.3580981

Learning Strategies

- Focus on learning strategies rather than tool.
- Project based learning is promising for a number of reasons
    - Personally meaningful projects enhance engagement and dedication
    - Aligns exceptionally well with Scratch in autonomous learning environments
- How to enhance effectiveness of project-based learning with Scratch?



RELATED WORK

Learning Computational Thinking Through Programming

- Programming is the optimal medium for CT education.
  - Understanding the foundational principles and practices of programming has demonstrated benefits for an individual's academic and professional lives.
  - The nuances of CT can be effectively evaluated through programming exercises.
- Many schools have attempted to teach computational thinking through the use of programming languages such as Python and C, but this can prove very difficult for younger students.
- Led to the creation of new methodologies that involved graphical interfaces, block-based programming, and little to no syntax requirements.

Enhancing Creativity in Scratch Learning Environments

- Scratch is not just for teaching computational thinking, but is also a tool for freedom of expression through the use of storytelling, drawing, and creating games.
- The nurturing of CT can be greatly enhanced in programming environments that facilitate creativity.
- Multiple cases of tools for supporting creativity leading to improved computational thinking:
  - Narrative Thread (Howard et al.), a storytelling-based game design tool that streamlines the development of more complex game characters, dialogues, and events
  - Robertson et al. implemented Adventure Author, a creative process model in game design, which provides children with enhanced support and feedback during open-ended programming tasks

Supporting Children’s Personally Meaningful Projects through Child-AI Collaboration

- Personally meaningful projects reflect a child’s personal interests, emotions, and experiences. Can come in the form of stories, animations, and games.
- Can be difficult to adapt in the teaching environment, as a child’s creativity often surpasses their technical skill. Often leads to standardized projects, which inhibits the child’s creativity.
- The use of technology to bridge the gap between a child’s creativity and ability is already well established
  - PopBots, RoBoHoN, StoryCoder, Teachable Machine, StoryDrawer
- LLMs can take this to the next level by allowing for interactivity, and can be trained on the large number of available educational content for Scratch.

FORMATIVE INVESTIGATION

Objectives

1) Identify the challenges children face when using Scratch for autonomous programming learning through theme-based creative programming.

2) Discover design opportunities for ChatScratch that address these challenges.

Methodology

- Recruited six expert Scratch educators (3 females, 3 males) of age 26-39 (M=31, SD=4.44) with Scratch teaching experience range from 4 to 12 years (M=6.67, SD=2.96)
- Educators were affiliated with a tutoring company in China that specializes in programming for children.
- Observed teaching sessions conducted by these educators.

Methodology cont.

- Performed semi-structured interviews after about 2 weeks of observation.
- Interviews structured on workflow of children’s theme-based creative programming.
- Focused on three key phases: project planning, assets creation, and code implementation.
- Discussed format of theme-based creative programming and how AI can assist in children’s programming.

FINDINGS

Artist’s Block in Project Planning

- Prior to actually coding a project, students generally go through a process of creative thinking and conceptualization.
- As young children are at still in the midst of cognitive development, they are often limited in their abstract reasoning and ability to construct a cohesive project plan.
- Four interviewees mentioned employing a simple storyboard to assist in overcoming this obstacle.

Bounded Creativity in Assets Creation

- While Scratch does offer a large variety of sprites and backgrounds to be used in creative projects, students often have difficulty finding the exact thing they need, and instead settle on something generic. This limits their creativity and how personal the project is to them, reducing engagement and attachment to their projects.
- *"I have seen many kids use the Scratch Cat as their character, not by choice, but due to the lack of appropriate assets. Such limitations frequently lead to their frustration as their original creative intent is not truly represented."*
- There is a feature that allows the user to draw or import their own sprites, but is seldom used by children either due to undeveloped drawing skills or low digital literacy.

Code Assistance as a Scaffold

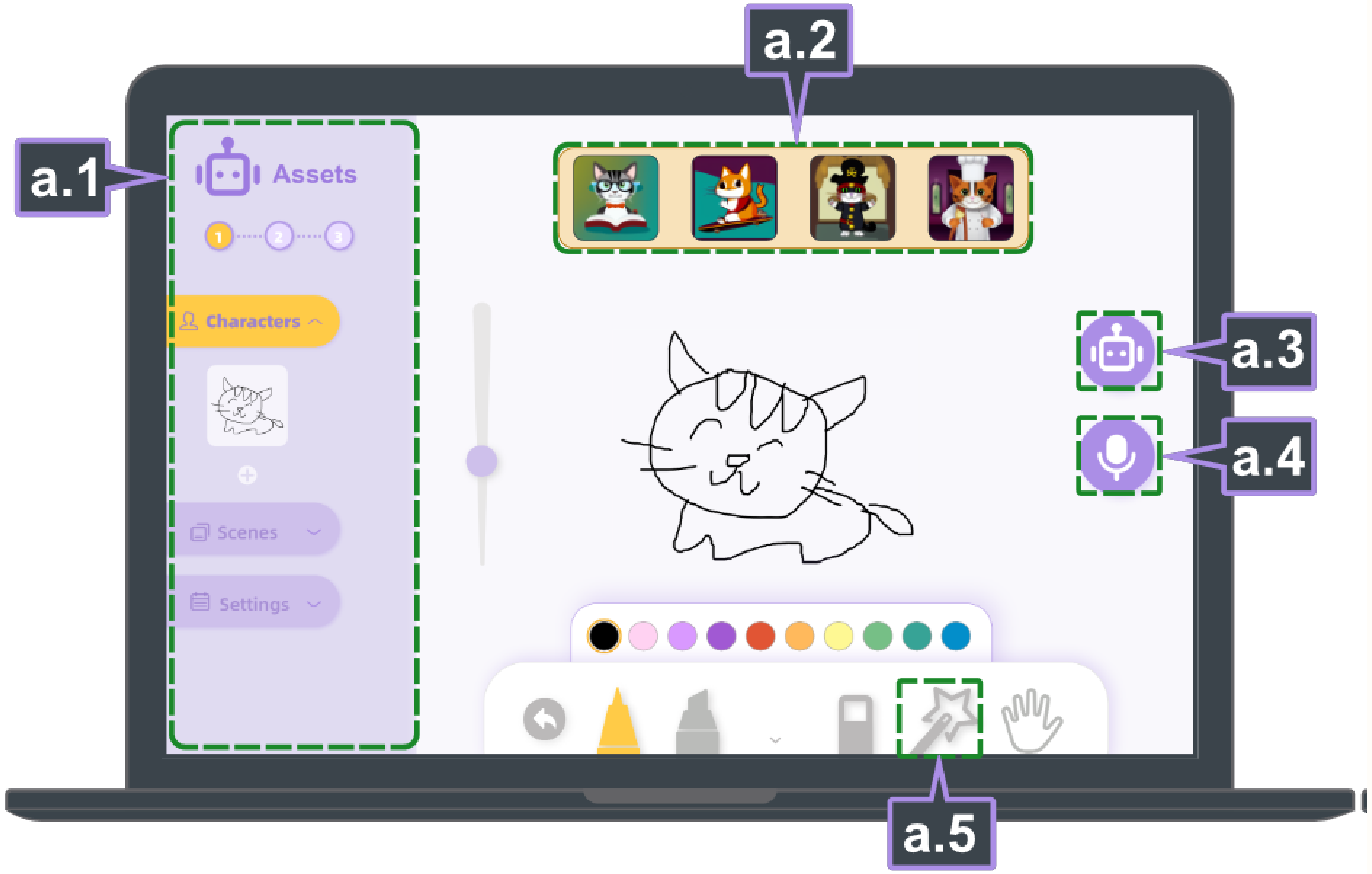
- While the color coded nature of Scratch blocks makes it easier to find a block once a child has seen it, young children still struggle with determining where a block might be based on its function rather than color.
- Educators typically deal with this difficulty by providing the foundation of the projects (i.e. enough blocks to get them started, but not so much to make the task trivial).
- This can give students a sense of confidence as the initial anxiety of how to begin is eliminated, and can lead to much more impressive projects.

Support Personally Meaningful Projects.

- All six interviewees agreed that students show a much stronger affinity for projects that have some personal meaning to them over standardized projects that focus on completing a specific task.
    - Significantly enhances children’s engagement, which encourages creative behavior, deepening their understanding, and motivating them to explore and innovate.
    - Children exhibit a strong sense of pride and accomplishment when they engage with personal meaningful projects. *“This is something I created”*

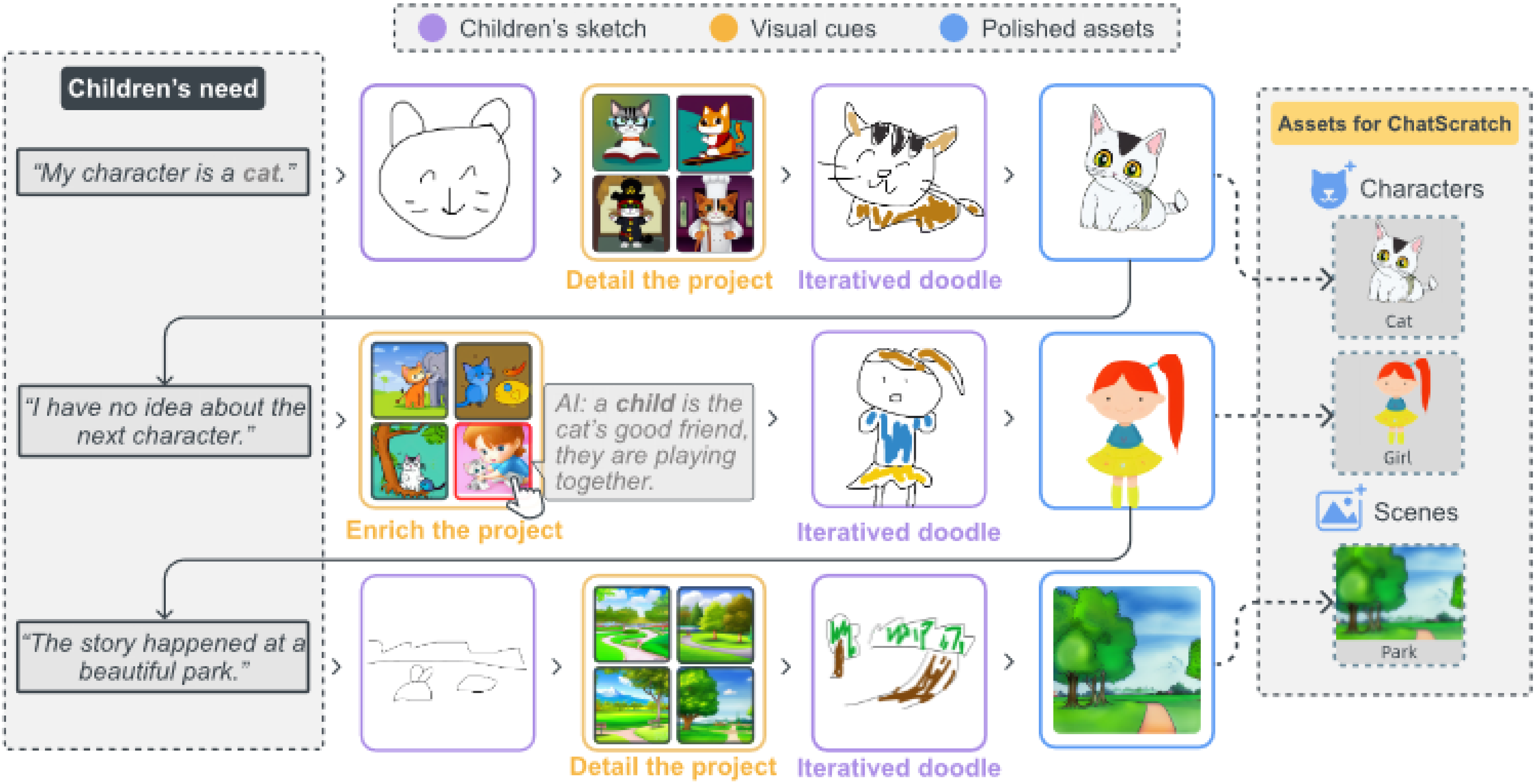
DESIGN

Goal 1: Interactive Structured Storyboard and Visual Cues

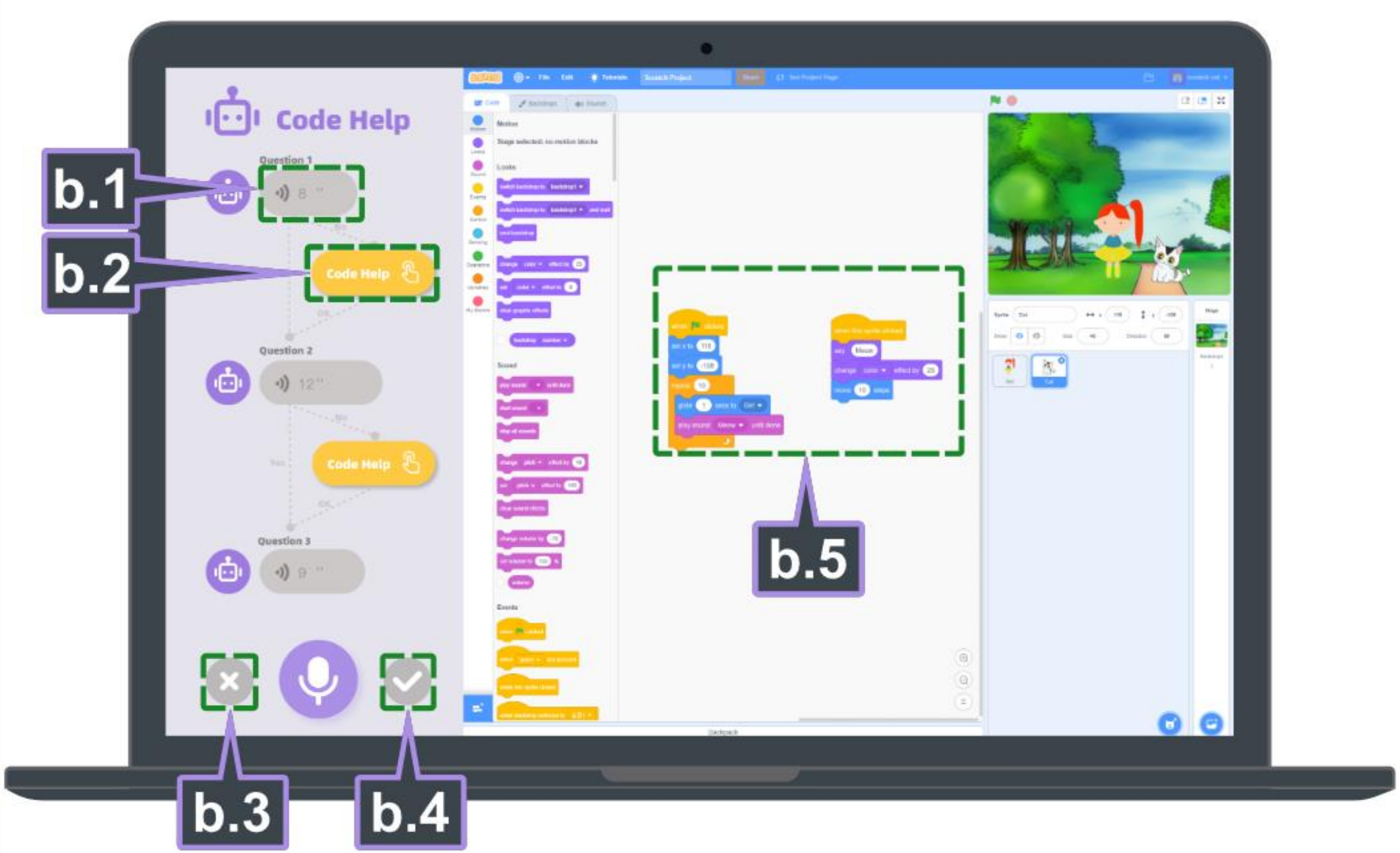


a. AI-empowered interactive storyboard

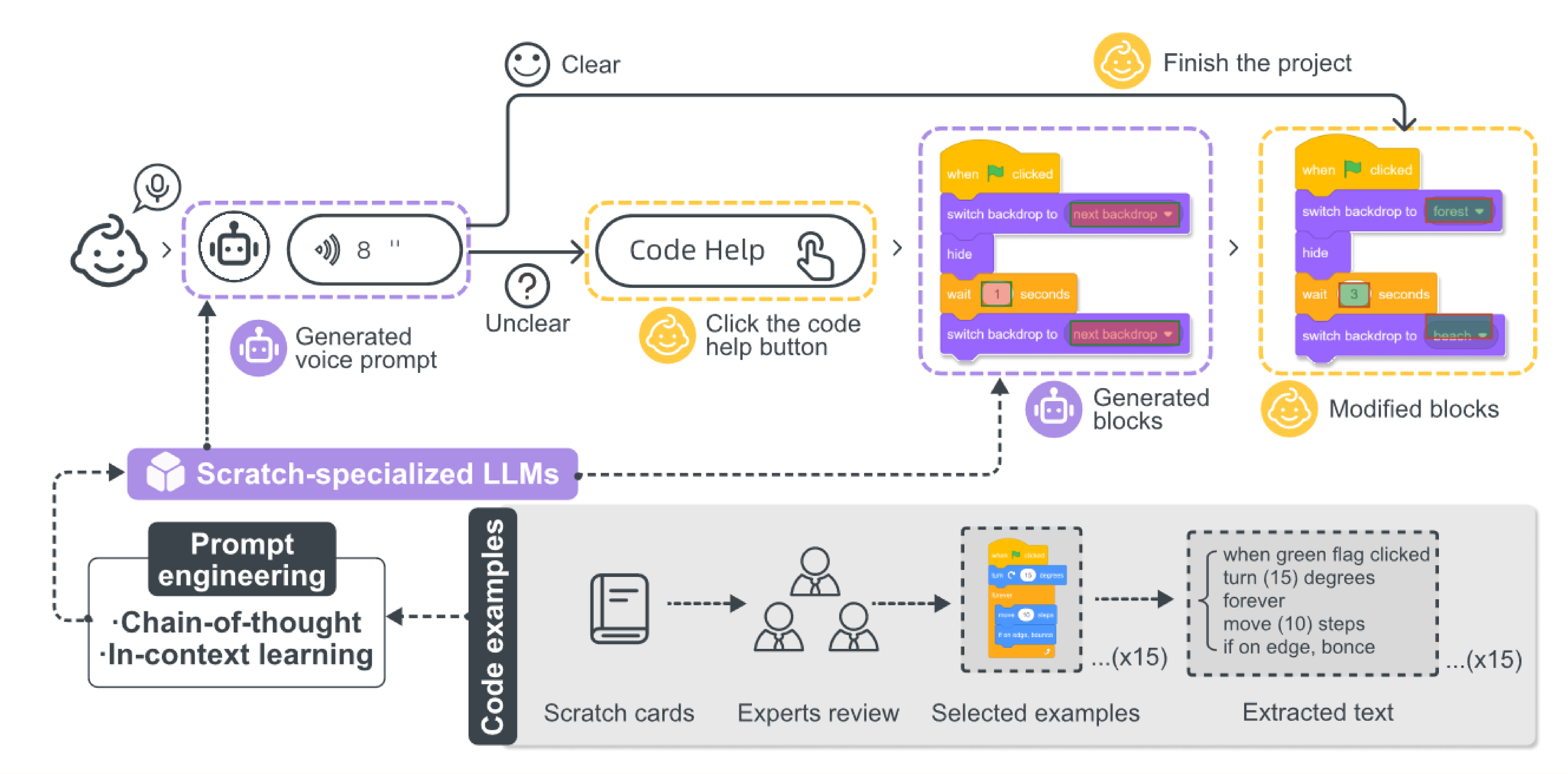
Goal 2: Drawing-based Assets Creation with Advanced Image Generation



Goal 3: Code Assistant Supported by Scratch-Specialized LLM

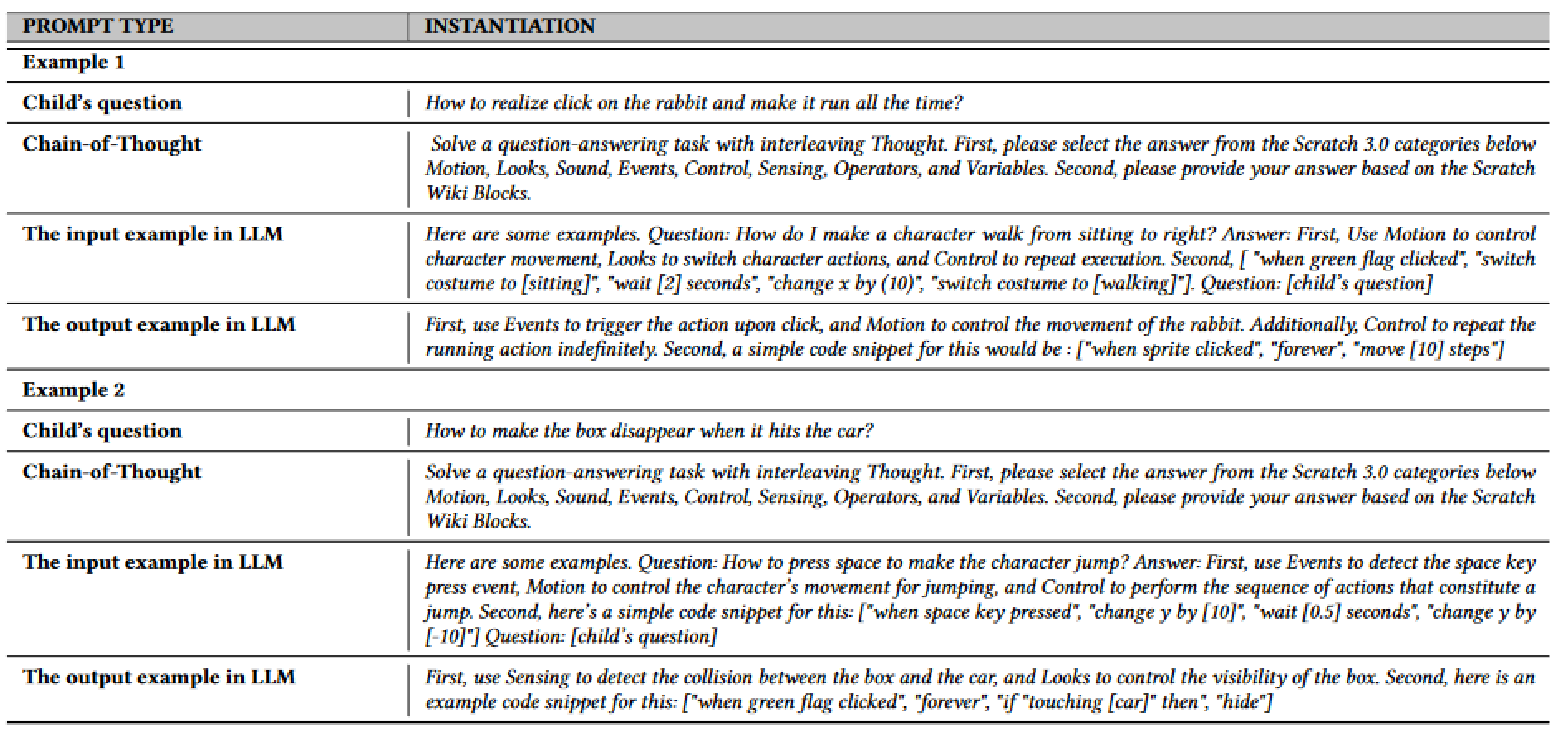


Goal 3: Code Assistant Supported by Scratch-Specialized LLM



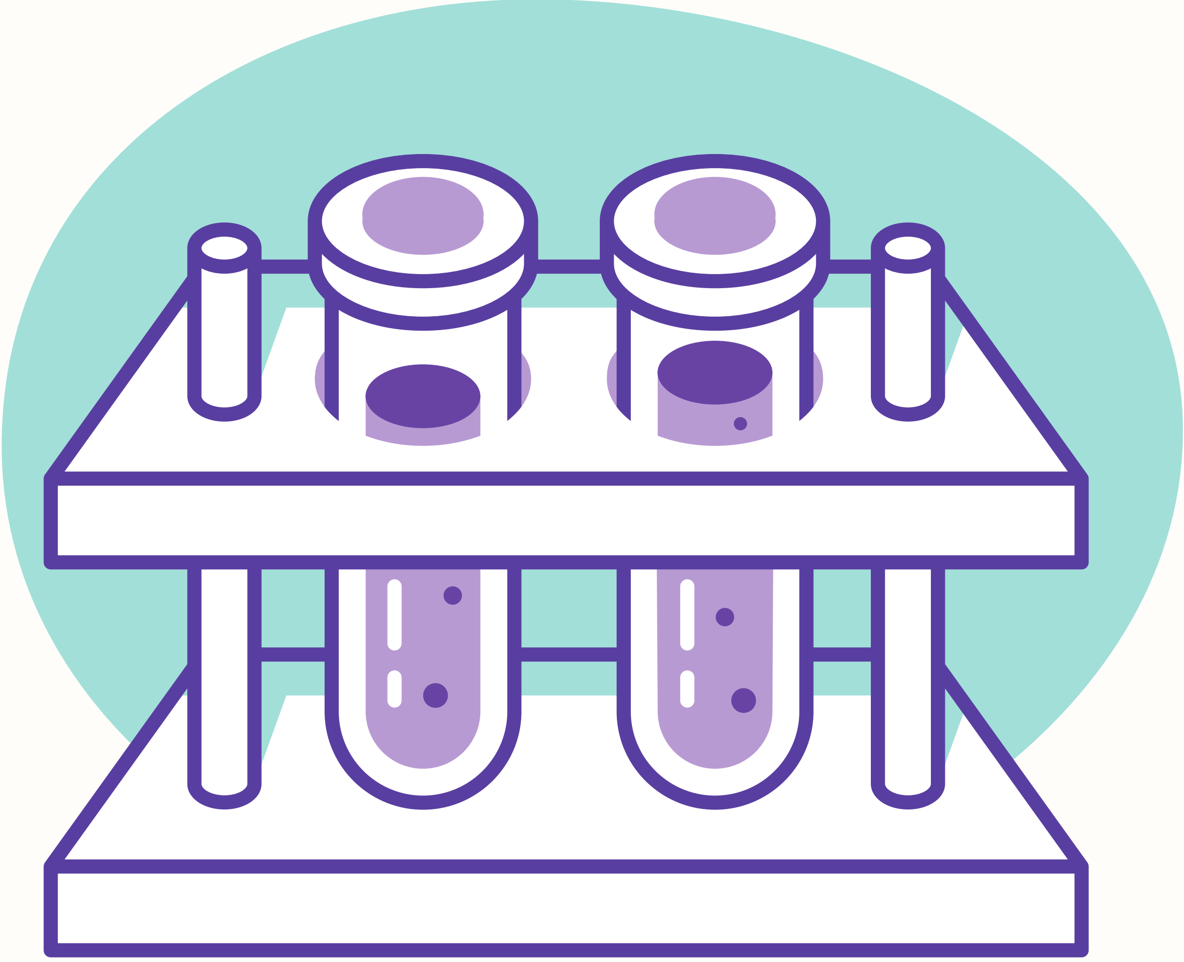
```
when green flag clicked
turn (15) degrees
forever
move (10) steps
if on edge, bonce
```

Goal 3: Code Assistant Supported by Scratch-Specialized LLM



PROMPT TYPE	INSTANTIATION
Example 1	
Child's question	*How to realize click on the rabbit and make it run all the time?*
Chain-of-Thought	*Solve a question-answering task with interleaving Thought. First, please select the answer from the Scratch 3.0 categories below Motion, Looks, Sound, Events, Control, Sensing, Operators, and Variables. Second, please provide your answer based on the Scratch Wiki Blocks.*
The input example in LLM	*Here are some examples. Question: How do I make a character walk from sitting to right? Answer: First, Use Motion to control character movement, Looks to switch character actions, and Control to repeat execution. Second, ["when green flag clicked", "switch costume to [sitting]", "wait [2] seconds", "change x by (10)", "switch costume to [walking]"]. Question: [child's question]*
The output example in LLM	*First, use Events to trigger the action upon click, and Motion to control the movement of the rabbit. Additionally, Control to repeat the running action indefinitely. Second, a simple code snippet for this would be : ["when sprite clicked", "forever", "move [10] steps"]*
Example 2	
Child's question	*How to make the box disappear when it hits the car?*
Chain-of-Thought	*Solve a question-answering task with interleaving Thought. First, please select the answer from the Scratch 3.0 categories below Motion, Looks, Sound, Events, Control, Sensing, Operators, and Variables. Second, please provide your answer based on the Scratch Wiki Blocks.*
The input example in LLM	*Here are some examples. Question: How to press space to make the character jump? Answer: First, use Events to detect the space key press event, Motion to control the character's movement for jumping, and Control to perform the sequence of actions that constitute a jump. Second, here's a simple code snippet for this: ["when space key pressed", "change y by [10]", "wait [0.5] seconds", "change y by [-10]"] Question: [child's question]*
The output example in LLM	*First, use Sensing to detect the collision between the box and the car, and Looks to control the visibility of the box. Second, here is an example code snippet for this: ["when green flag clicked", "forever", "if "touching [car]" then", "hide"]*

Evaluation



The paper evaluated the performance of ChatScratch The paper included 24 (10 female, 14 male) children as participants, from ages 6 to 12.

All participants had less than 1 year of experience with Scratch and were native Mandarin Speakers. Each child participated in two theme based creative programming sessions, with a 48 hour interval between sessions.

Methodology

Each participant were required to undertake two theme based creative programming sessions.

The session covered two specific themes, which were “Memorable Experiences” and “Animal Stories.”

Each session was for 70 minutes, with 20 minutes for understanding each platform. Researchers were allowed to assist children with unexpected difficulties.

Video recordings and project files in sb3 format were collected.

Afterwards, participants filled out the Creativity Support Index questionnaire and had an interview on user experiences.

Collected Date and Metrics

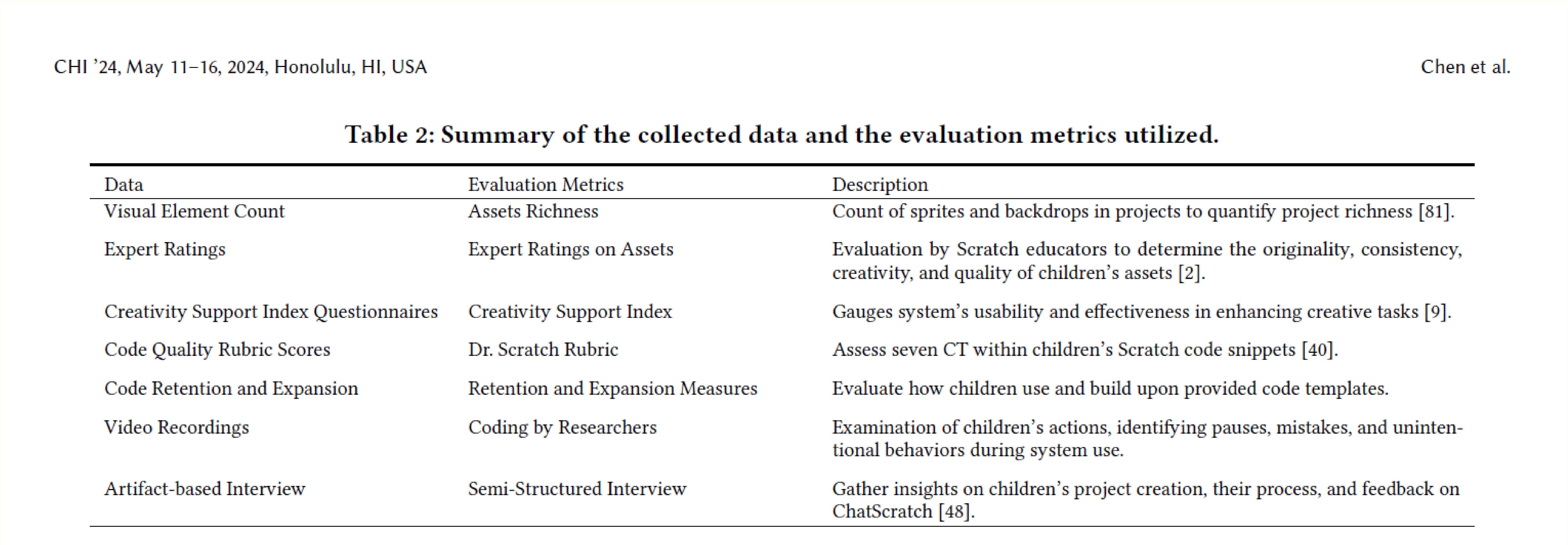


Table 2: Summary of the collected data and the evaluation metrics utilized.

Data	Evaluation Metrics	Description
Visual Element Count	Assets Richness	Count of sprites and backdrops in projects to quantify project richness [81].
Expert Ratings	Expert Ratings on Assets	Evaluation by Scratch educators to determine the originality, consistency, creativity, and quality of children’s assets [2].
Creativity Support Index Questionnaires	Creativity Support Index	Gauges system’s usability and effectiveness in enhancing creative tasks [9].
Code Quality Rubric Scores	Dr. Scratch Rubric	Assess seven CT within children’s Scratch code snippets [40].
Code Retention and Expansion	Retention and Expansion Measures	Evaluate how children use and build upon provided code templates.
Video Recordings	Coding by Researchers	Examination of children’s actions, identifying pauses, mistakes, and unintentional behaviors during system use.
Artifact-based Interview	Semi-Structured Interview	Gather insights on children’s project creation, their process, and feedback on ChatScratch [48].

Results

The results of the paper were based on answering the following Research Questions (RQ):

- How does ChatScratch support creative tasks?
- How does ChatScratch support high- quality code implementation in Scratch Programming?
- How does ChatScratch support children to create personally meaningful projects autonomously?

Research Question 1 (RQ1)

The authors implemented the visual element count to evaluate the efficacy of planning and visual cues. Using visual question and image polish helped children:

- Create more freely (originality)
- Produce content that aligned with their expectations (consistentcy)

The data showed that the collaborative support meant that children’s efforts were better represented, providing them with visually rich and detailed projects.

Research Question 2 (RQ2)

The results from the interviews showed that children were more comfortable with “Loops.” The ambiguity in the “If else” statements made it challenging to achieve the full potential of their projects.

Additionally, the authors evaluated the efficacy and usability of the code produced by the code assistant and the degree to which this was used by the children.

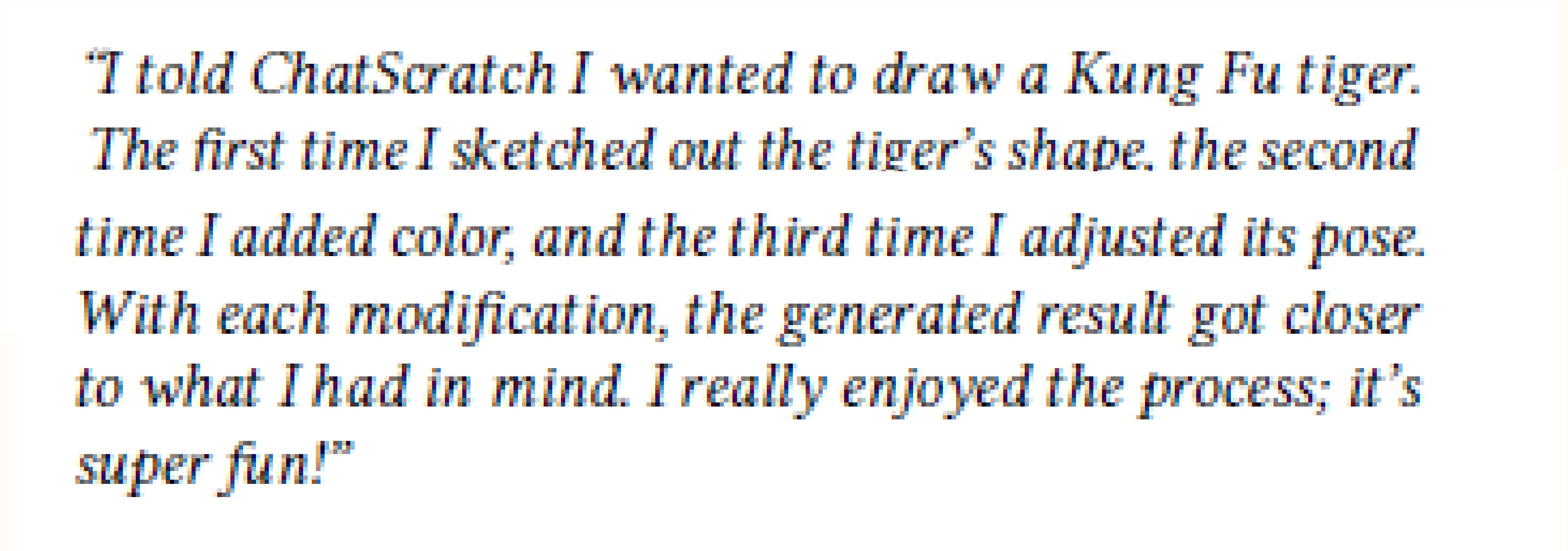
The data shows that about **3 quarters of the children retained the suggested code**, which indicated it’s effectiveness.

Research Question 3 (RQ3)

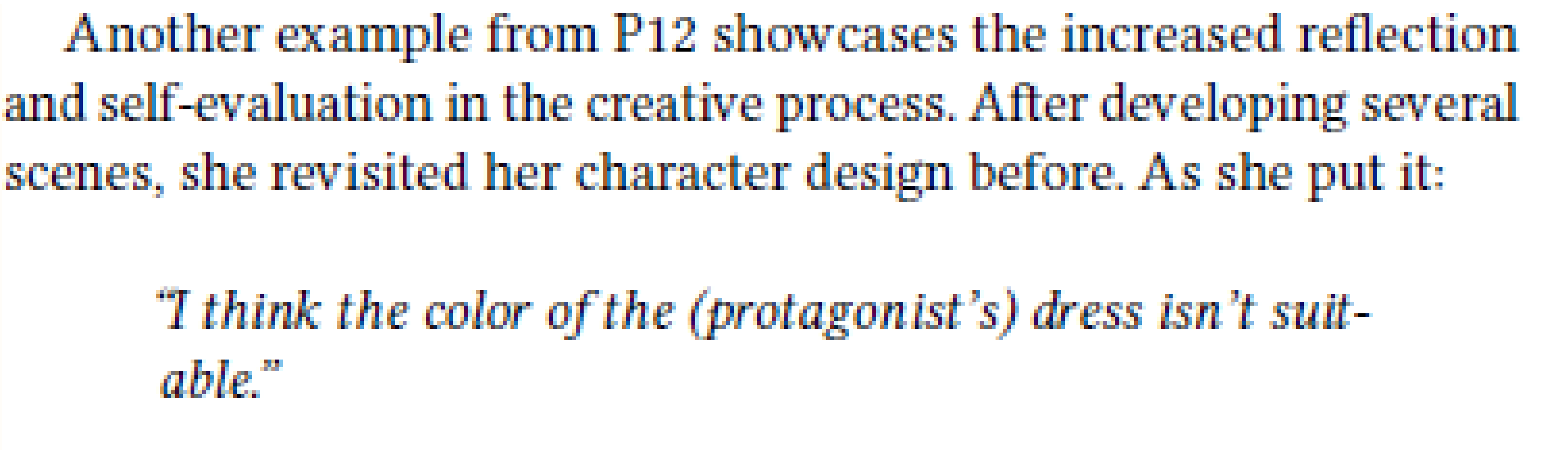
The platform aims to craft meaningful projects by **incorporating visual cues, image polish and code templates.** The combination of all of these features gives children the opportunity to receive ongoing feedback on their projects, and refine them accordingly.

The projects additionally became more meaningful as children focused on their own preferences and passions. Some children additionally decided to develop projects based on classic fables, such as the Tortoise and the Hare.

Research Question 3 (RQ3)



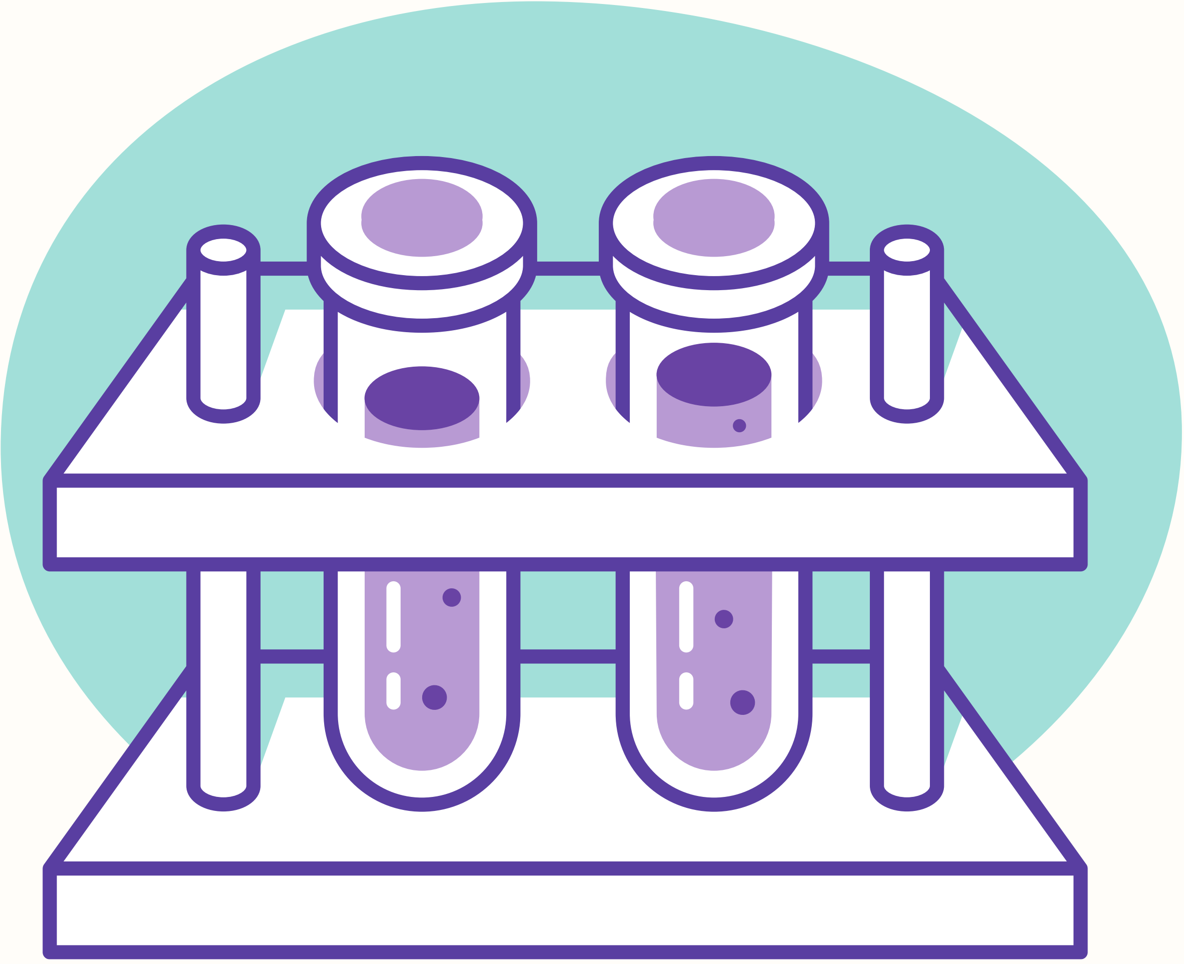
"I told ChatScratch I wanted to draw a Kung Fu tiger. The first time I sketched out the tiger's shape, the second time I added color, and the third time I adjusted its pose. With each modification, the generated result got closer to what I had in mind. I really enjoyed the process; it's super fun!"



Another example from P12 showcases the increased reflection and self-evaluation in the creative process. After developing several scenes, she revisited her character design before. As she put it:

> *"I think the color of the (protagonist's) dress isn't suitable."*

Limitations



While the voice features helped address potential literacy problems, the absence of visual input feedback may lead children to incorrectly assume that they are ones making the mistakes.

In the future, ChatScratch can be changed to include rephrasing or refining children’s output or ask clarifying questions.

Limitations

Similarly the platform was made based on separating children’s projects into three main acts, which included characters, scenes and events. This helped them conceptualize ideas. This however, reduced the generalizability.

As children focused on recreating stories or telling ones of their own, it may be favorable to change the three act design, as children may want to work on projects such as music and games.

General Discussion

The ability to simplify the access to code implementation addresses the issues with learning barriers for children, especially younger children trying to access computational thinking.

Children not only retained the code suggestions but more than half of the final code was implemented by the children themselves.

Driven to make changes based on their vision, children were additionally more resilient and less likely to give up. This additionally promoted children to have a greater sense of participation and enthusiasm.

General Discussion

The authors postulate that constructing with Scratch is like building with LEGO bricks, as children coded based on storytelling. ChatScratch helped children to create diverse ideas, create imaginative characters and make visual changes based on their own preferences.

ChatScratch has leveraged the use of GAI to help assist children with **project planning,** supporting their creative projects and **providing guidance** in programming.

The usage of GAI provides children with the **ability to understand, reason and think creatively**. These skills are **helpful to guide and enrich the creative process in children**, especially when teaching resources may be limited.

THANK YOU!