Coding Strip: A Pedagogical Tool for Teaching and Learning Programming Concepts through Comics

Sangho Suh, Martinet Lee, Gracie Xia, Edith Law

UNIVERSITY OF WATERLOO

WATERLOO HCI
CHERITON SCHOOL OF COMPUTER SCIENCE

Table of Contents

- Motivation
- Related Work
- Methodology & Results
- Summary & Future Work

Motivation

Teaching kids to code is the next frontier in 21st century education

By Helena Game | September 15, 2017 | @helenagame

Japan Makes Coding Mandatory for All Students Starting in Elementary School

Posted on March 28, 2019 by Ted

China Pushes Coding for Kids in Effort to Tackle Innovation Gap

A young boy learns to code at the Tarena Learning Center in Beijing. NBC News

EDUCATION

Teaching coding in Canadian schools: How do the provinces measure up?

BY ALYSSA JULIE • NEWS TALK 770

Posted August 24, 2017 2:23 pm
Updated August 28, 2017 12:10 pm

File photo. *AP Photo/Stephan Savoia*

A computing revolution in schools

Rory Cellan-Jones
Technology correspondent
@BBCRoryCJ

1 September 2014 | Share

Coding at school: a parent's guide to England's new computing curriculum

From the start of the new term, children as young as five will be learning programming skills in the classroom

Coding is on the curriculum for primary and secondary school pupils in the UK. Photograph: Alamy

Programming remains difficult to master because

it involves **concepts** and **procedures** that are **abstract**

Learning Concepts in Programming

"*in learning programming,* ***[students] need to imagine*** *... many* ***abstract terms*** *that do not have equivalents in* ***real life****: how does a variable, a data type, or a memory address relate to a real-life object?*"

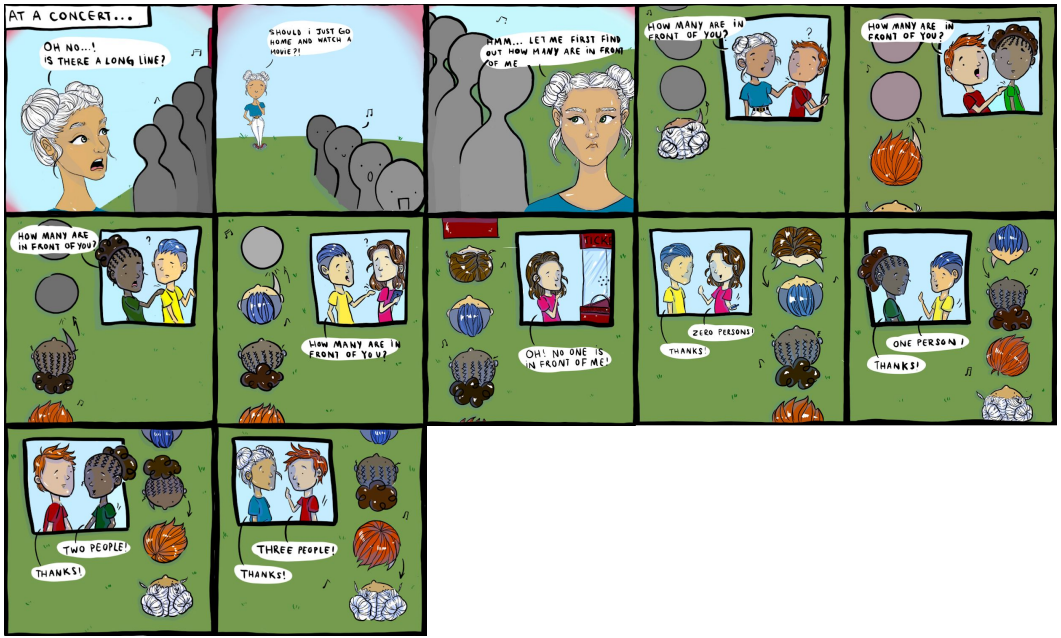
L. M. Giraffa, M. C. Moraes, and L. Uden, "Teaching object-oriented programming in first-year undergraduate courses supported by virtual classrooms"

Learning Procedures in Programming

Procedures in computing are often presented as an abstraction (e.g., loop), i.e., without showing the steps and thereby obscuring and making the procedures **abstract** for novice learners

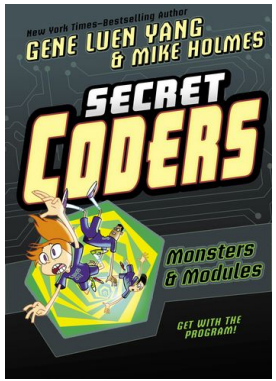
Making Abstract More Concrete with Comics

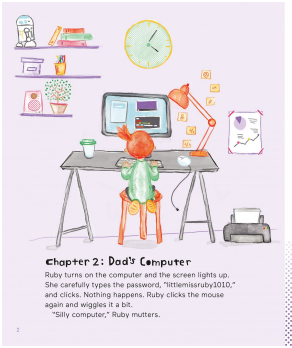
- *The **sequential** nature of the medium and its ability to express complicated concepts and procedures through **visual storytelling** provide reasons to believe that it can be an effective medium*



Limitation with Comics for CS Education

- While there have been several comic books for CS education, they are formatted as storybooks without correspondence to code
  - This misses out on the opportunity for students to transfer what they learn to traditional text-based programming





Coding Strip

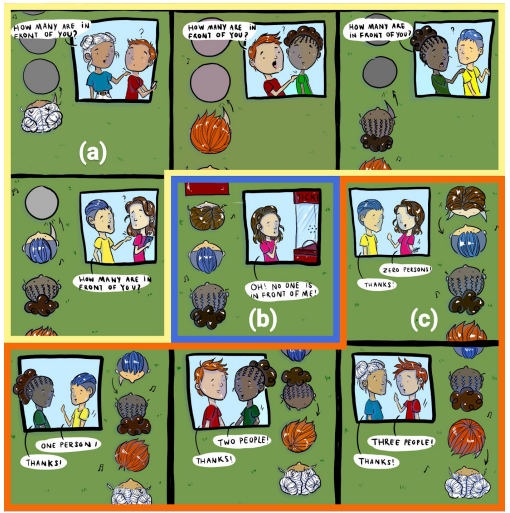
abstract

concrete

```
def how_many_in_line (num_of_people):
    if num_of_people == 0:
        return 0
    else:
        return 1 + how_many_in_line (num_of_people - 1)

how_many_in_line (4)
```

(a) (b) (c)



Coding Strip

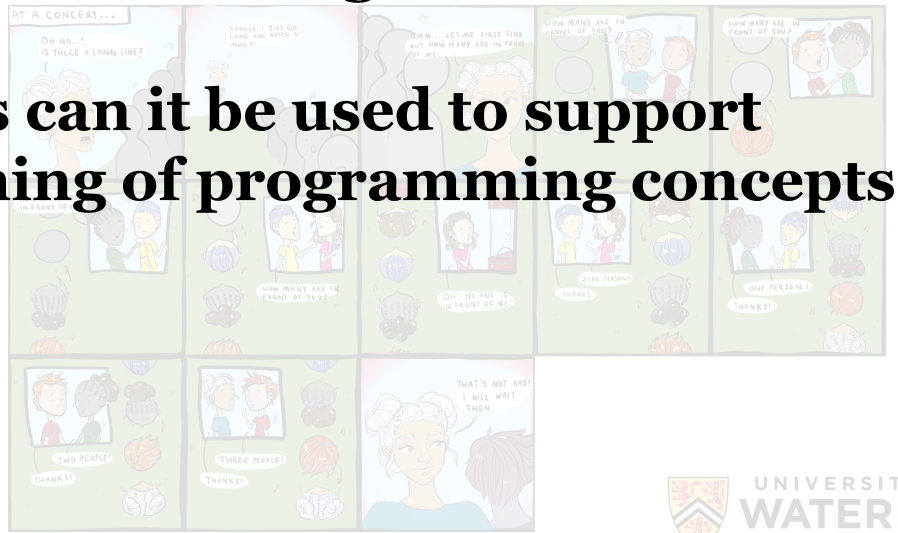
abstract

concrete

```
def func(x):
  if x == 0:
    return 0
  else:
    return 1 + func (x - 1)
```

How do we design it?

In what ways can it be used to support teaching and learning of programming concepts?

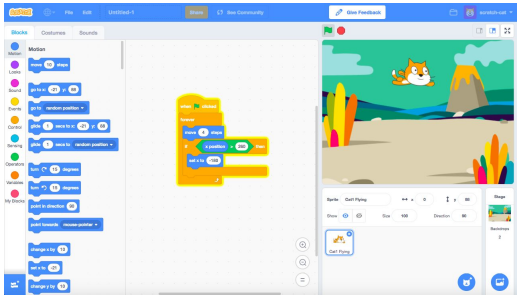


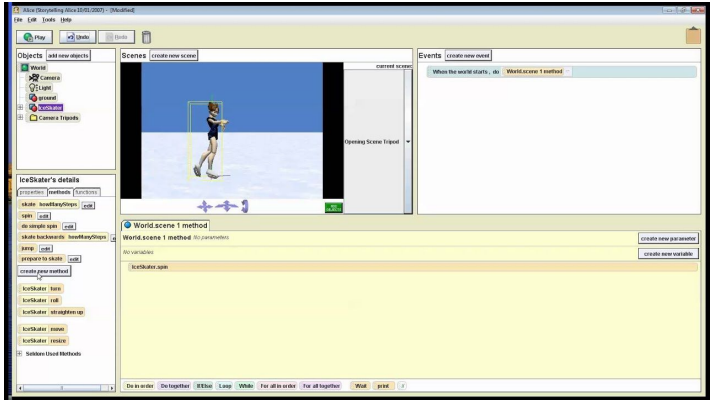
UNIVERSITY OF WATERLOO

Related Work

Making Programming More Concrete

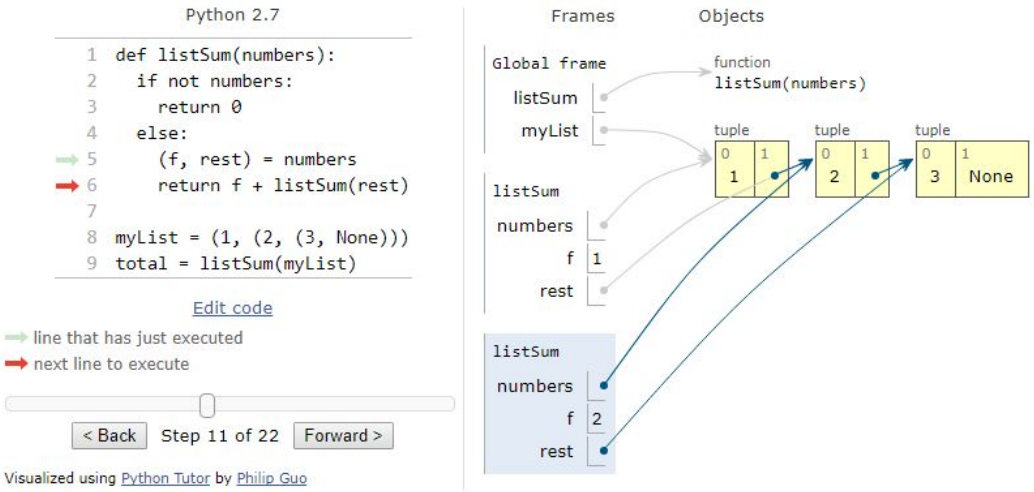
- Hands-on activities (e.g., CS Unplugged)
- Block-based / Tangible programming languages
- Storytelling (e.g., Storytelling Alice)





Making Programming More Concrete

- Visualization tools (e.g., Python Tutor)



Making Abstract More Concrete with Comics

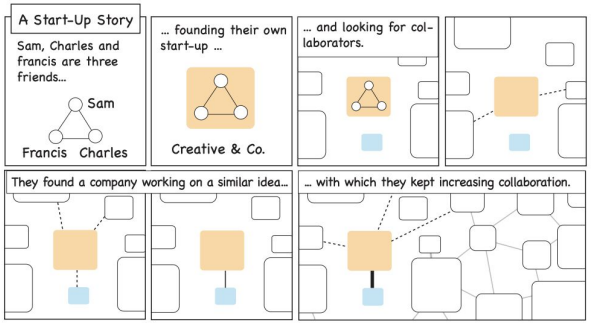
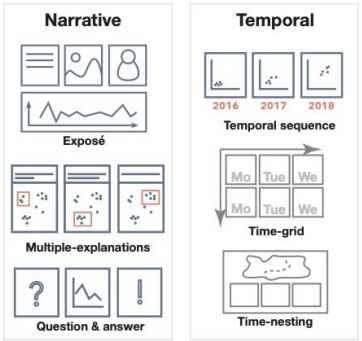
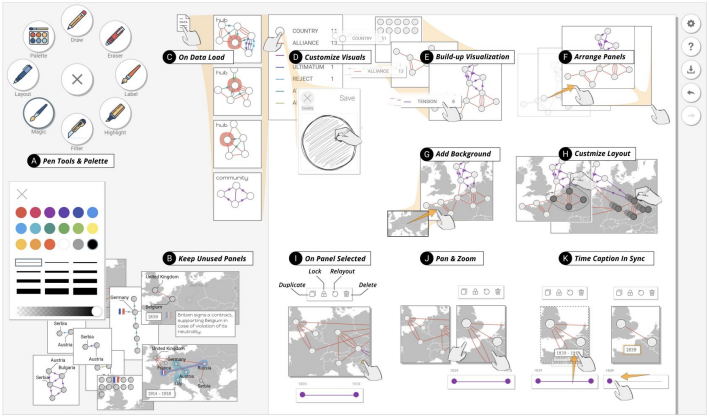


Figure 1. Graph comic: Evolving fictitious business relations.

Data-driven storytelling



Design patterns
for data comics



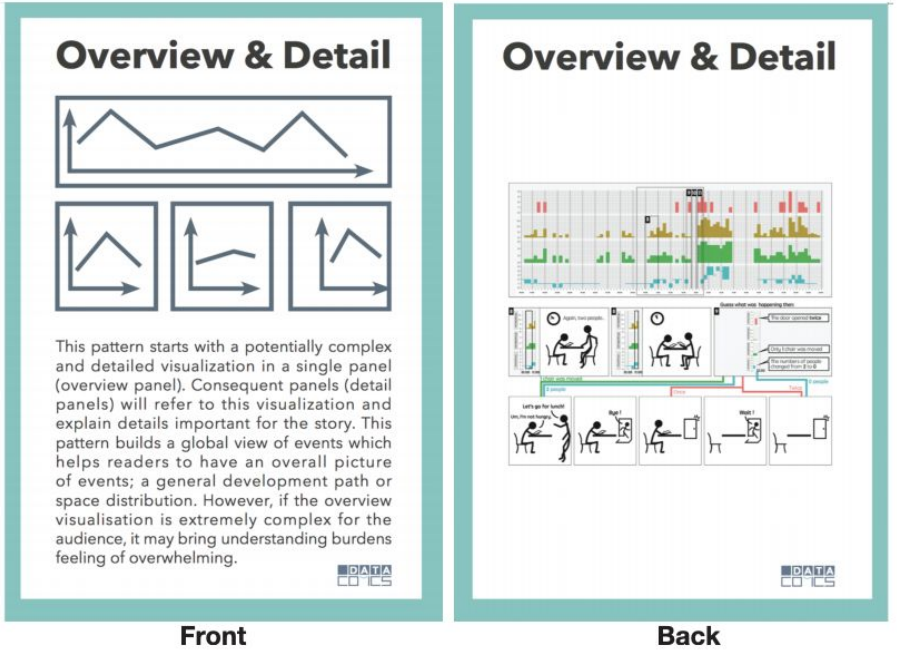
Authoring tool
for data comics

Bach et al, *Data comics*.

Design Process & Tools

Mora, Simone, Francesco Gianni, and Monica Divitini. Tiles: a card-based ideation toolkit for the internet of things.

Design Process & Tools



Front Back

Bach et al, *Design Patterns for Data Comics.*

Figure 8. Data-Comics workshop: (left) designing with pattern cards, (right) storyboarding with post-its.

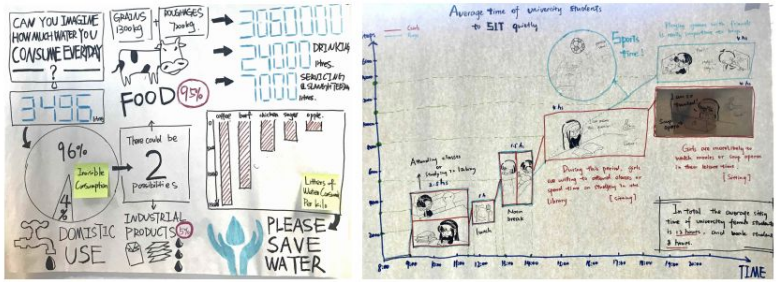


Figure 9. Data-Comic from the workshop: (left) annual water consumption (©Xudong Jiang, Yi He), (right) how long students sit (©Yuchen Ye, Wenqi Cai).

Methodology

Methodology

- **Phase I. Develop Design Process & Tools**
- **Phase II. Evaluate through Design Workshops**

Design Process

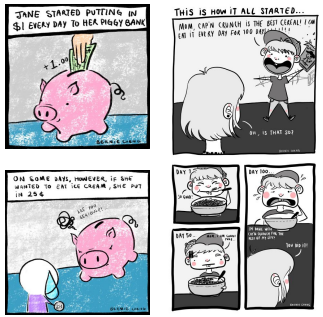
- **Step 1. Identify main programming concepts**

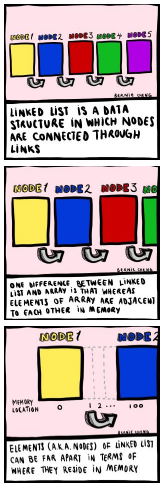
TABLE I: Concepts used in our study. Bolded are the concepts that workshop participants chose for their coding strips.

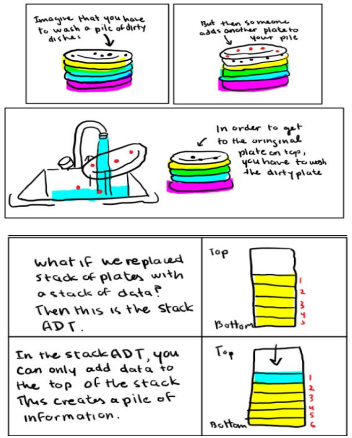
Constructs	Data Structures	Algorithms	Problem Solving Techniques
Variable	Array	Selection Sort	Greedy
Boolean	Linked List	Insertion Sort	**Divide & Conquer**
Condition	Queue	Merge Sort	**Recursion**
Counted Loop	Stack	Bubble Sort	
Conditional Loop	Tree	Linear Search	
Function	Graph	Binary Search	
	Dictionary		

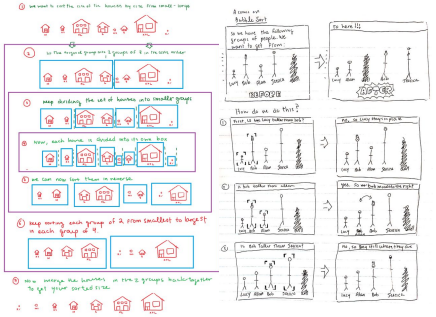
Design Process

- **Step 2. Design coding strips for concepts**









Design Process

- **Step 3. Identify process & support (tool) required**

Design Process

- **Step 3. Identify process & support (tool) required**

Design Process

- **Step 3. Identify process & support (tool) required**

Stage	**Concept Formulation**
Goal	"identify ideas behind a given concept"
Example	"**loop** is used to repeat sth. until certain condition is met"

Design Process

- **Step 3. Identify process & support (tool) required**

Stage	Concept Formulation	→	Story Development
Goal	"identify ideas behind a given concept"		"develop stories to explain the ideas"
Example	"**loop** is used to repeat sth. until certain condition is met"		"Henry ate capt'n crunch cereal for 100 days as a challenge"

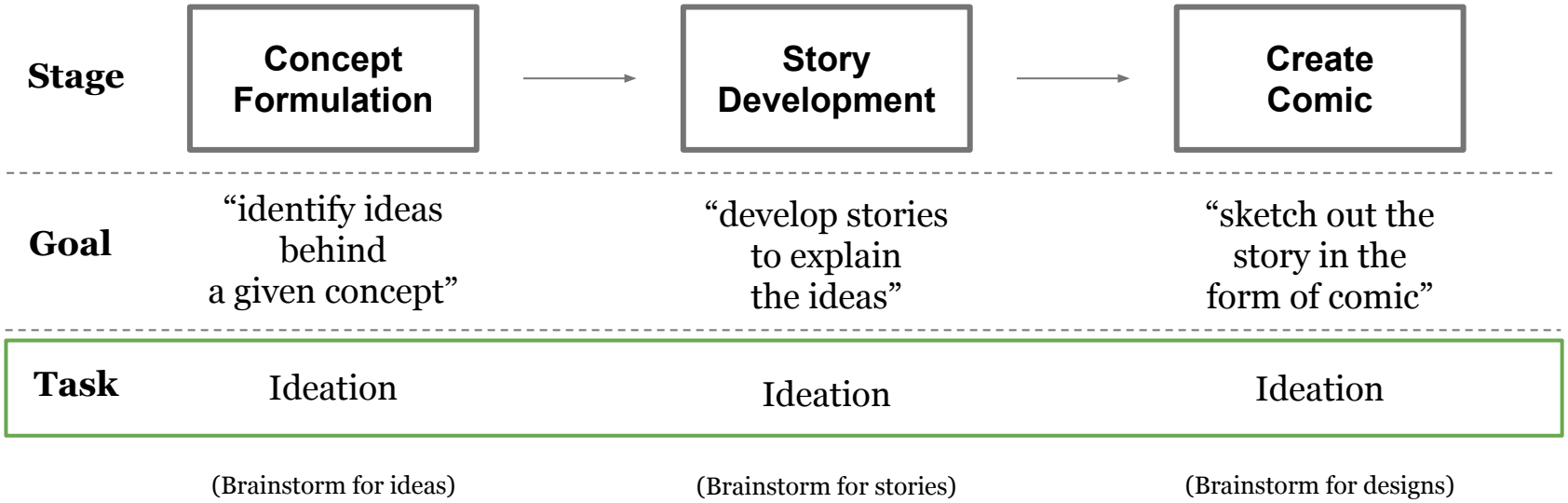
Design Process

- **Step 3. Identify process** & support (tool) required

Stage	Concept Formulation	→	Story Development	→	Create Comic
Goal	"identify ideas behind a given concept"		"develop stories to explain the ideas"		"sketch out the story in the form of comic"
Example	"**loop** is used to repeat sth. until certain condition is met"		"Henry ate capt'n crunch cereal for 100 days as a challenge"		

Design Process

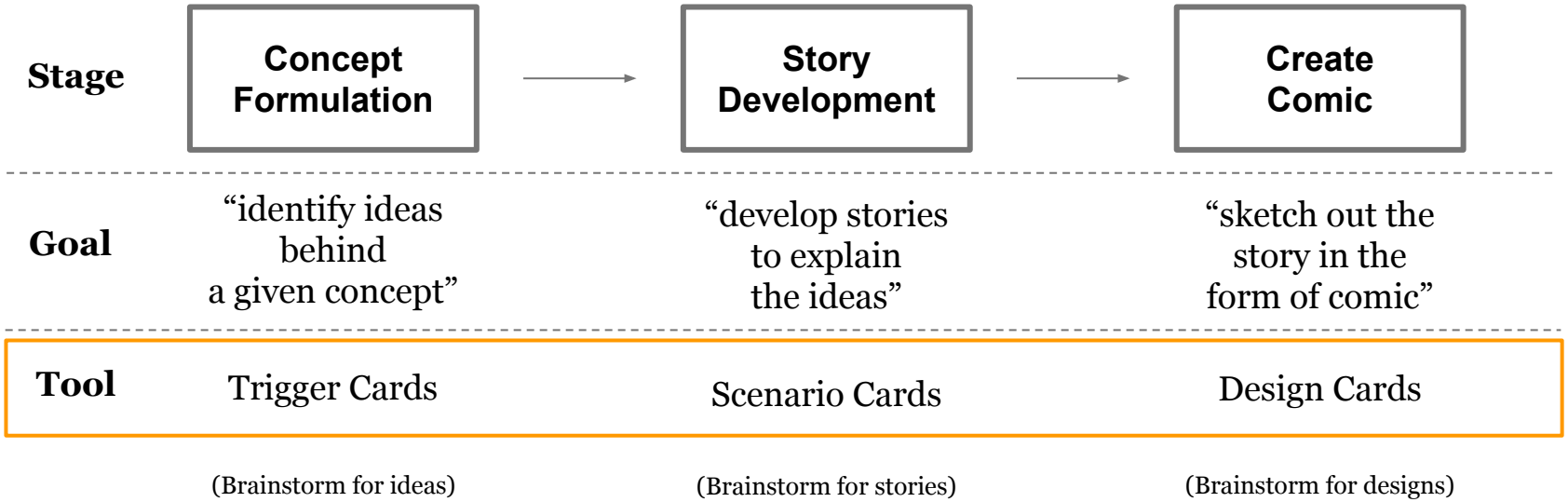
- **Step 3. Identify process & support (tool) required**



UNIVERSITY OF WATERLOO

Design Process

- **Step 3. Identify process & support (tool) required**

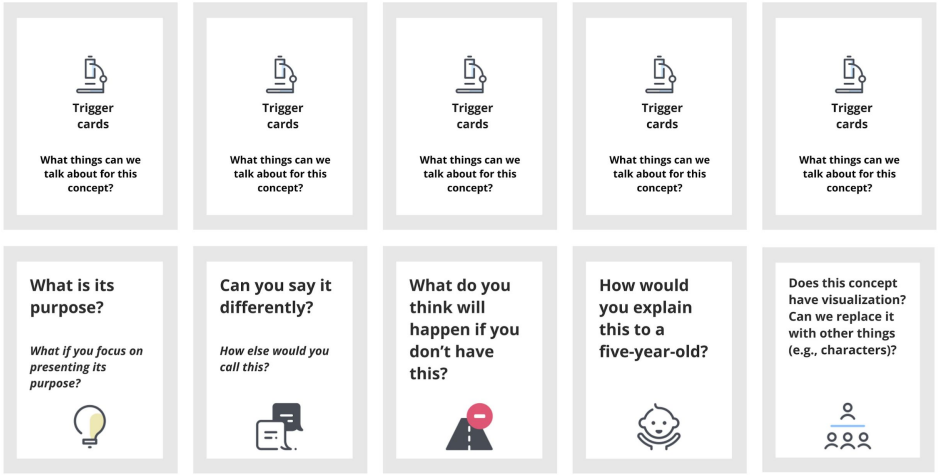


UNIVERSITY OF WATERLOO

Ideation Cards

- **Step 4. Create supporting tools**

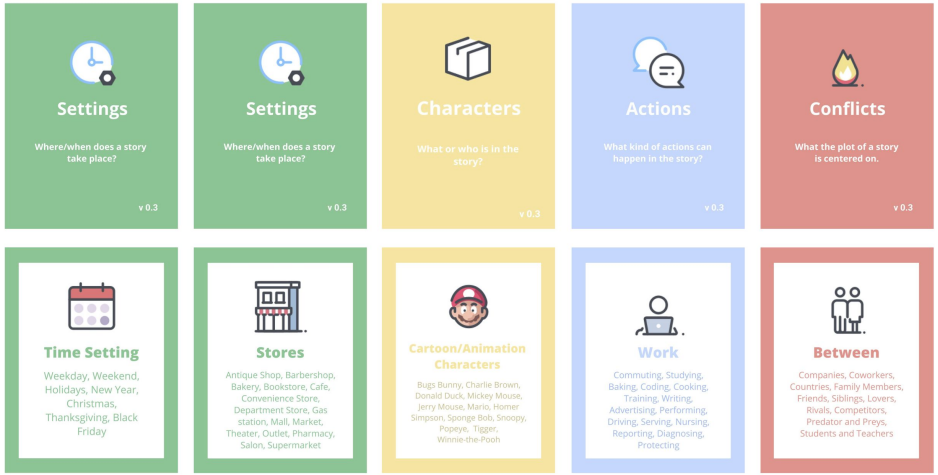
Stage	**Concept Formulation**
Goal	“identify ideas behind a given concept”
Task	Brainstorming
Tool	16 trigger cards



Ideation Cards

- **Step 4. Create supporting tools**

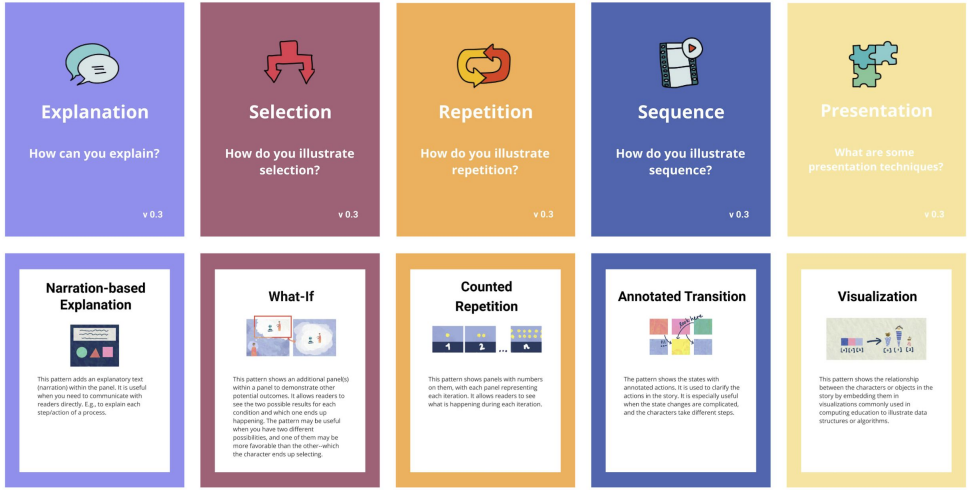
Stage	**Story Development**
Goal	“develop stories to explain the ideas”
Task	Brainstorming
Tool	19 scenario cards



Ideation Cards

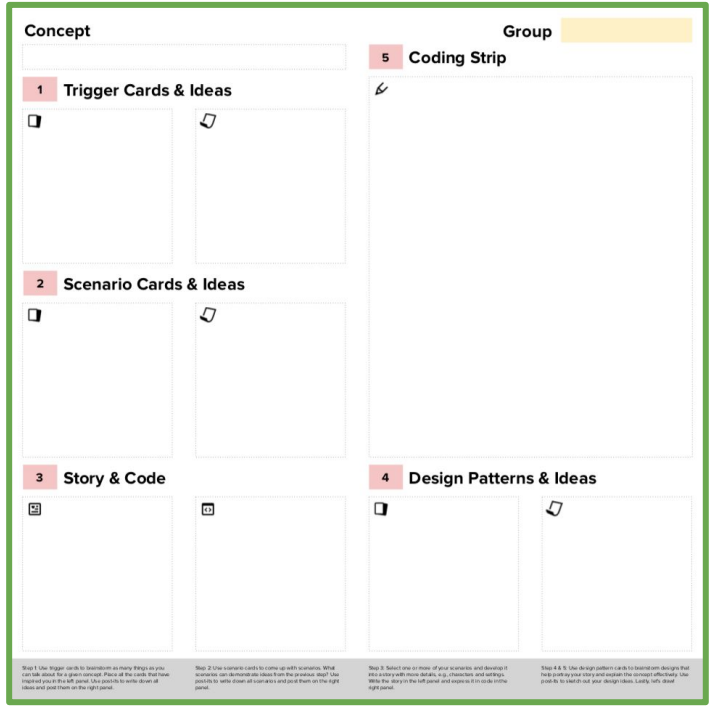
- **Step 4. Create supporting tools**

Stage	**Comic Illustration**
Goal	“sketch out stories in the form of comic”
Task	Brainstorming
Tool	30 design cards



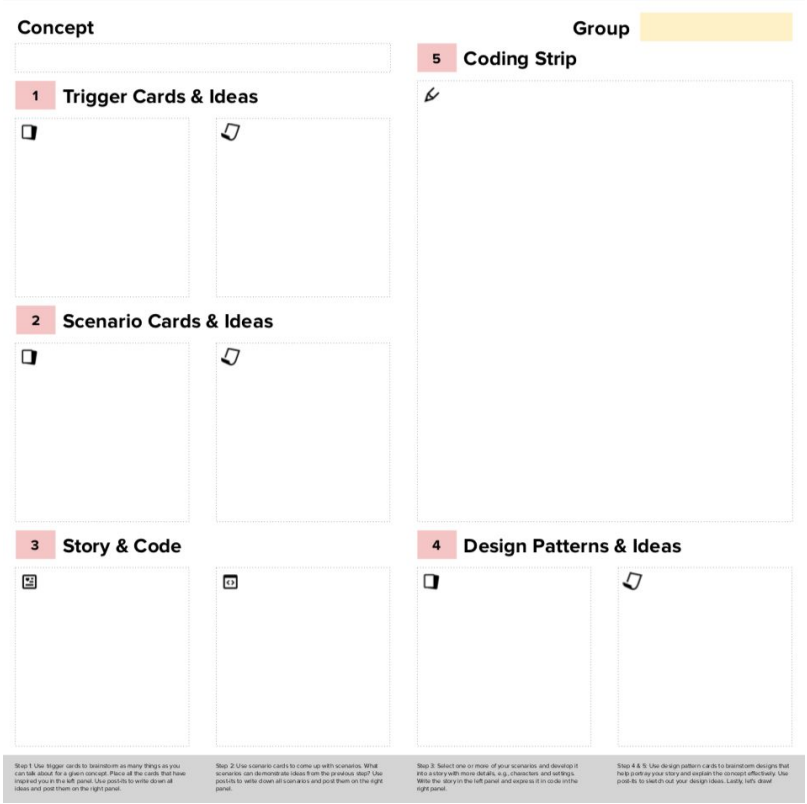
Design Board

- **Step 4. Create supporting tools**



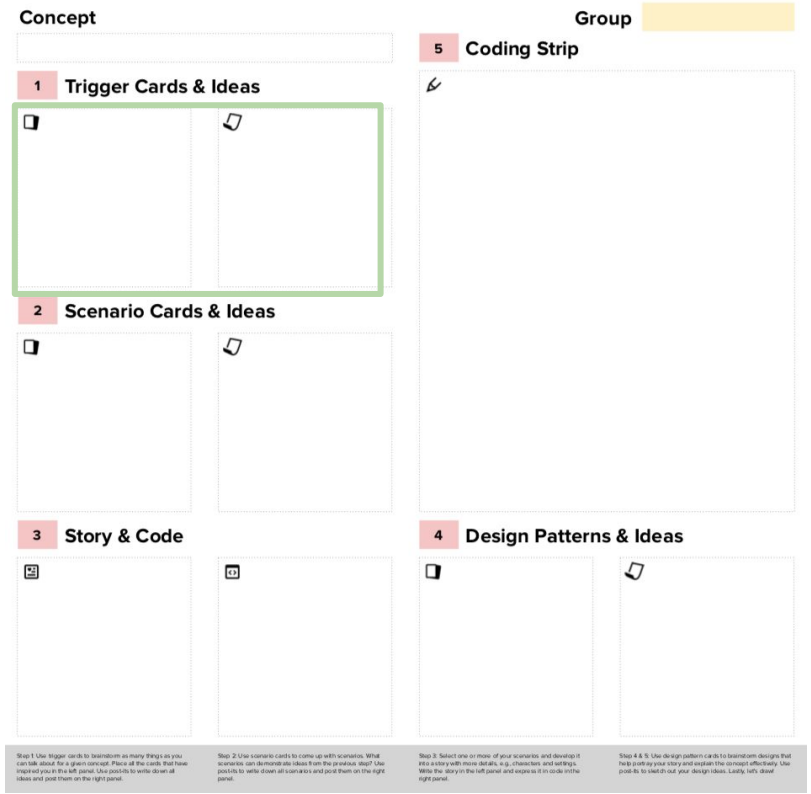
Tool design board

Design Board



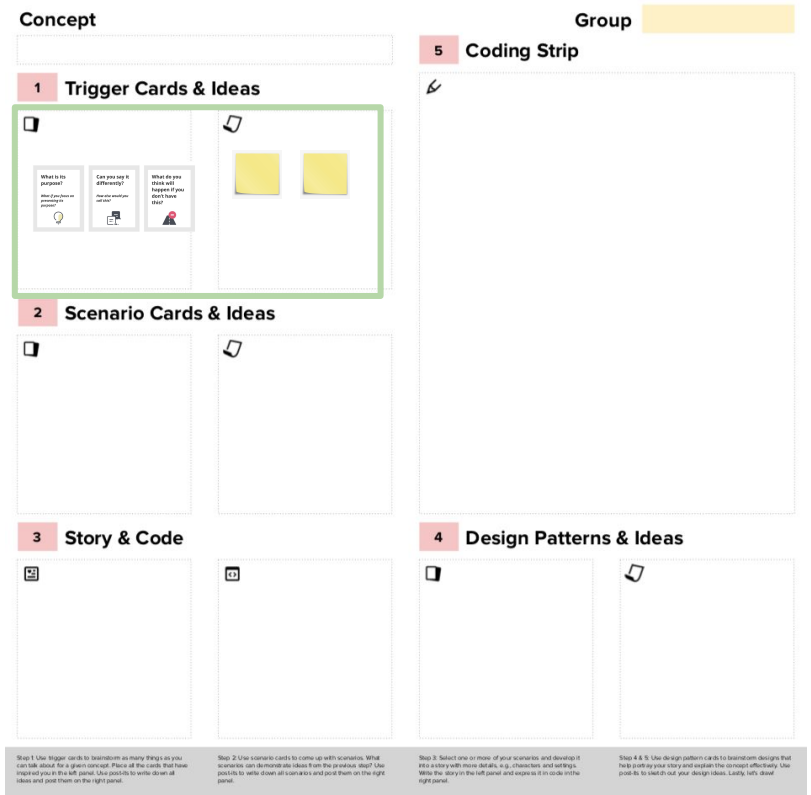
Design Board

Concept Formulation →



Design Board

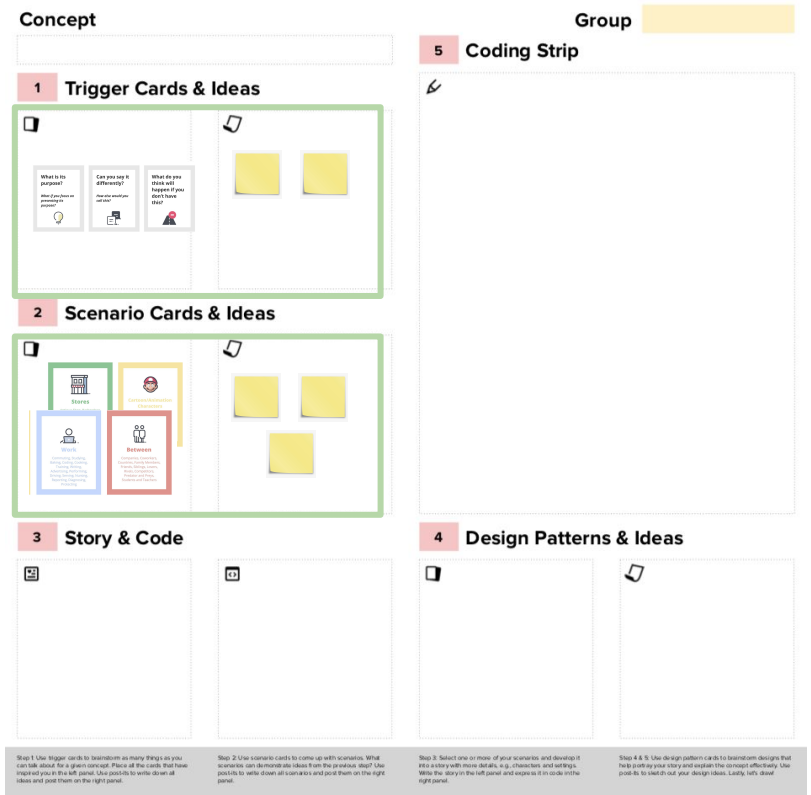
Concept Formulation



Design Board

Concept Formulation →

Story Development →

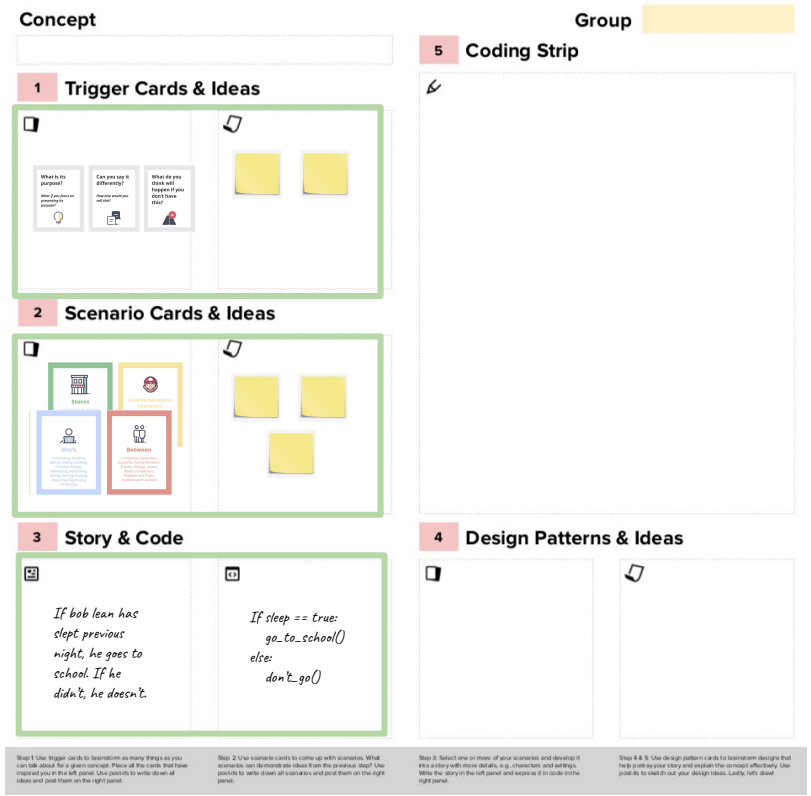


Design Board

Concept Formulation →

Story Development →

Story-to-Code Mapping →

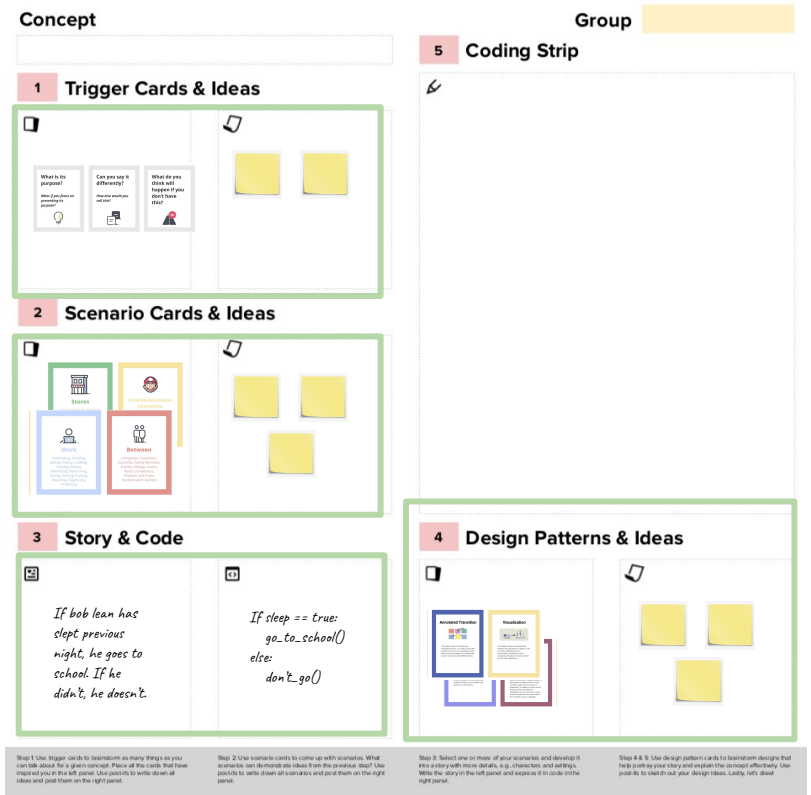


Design Board

Concept Formulation →

Story Development →

Story-to-Code Mapping →

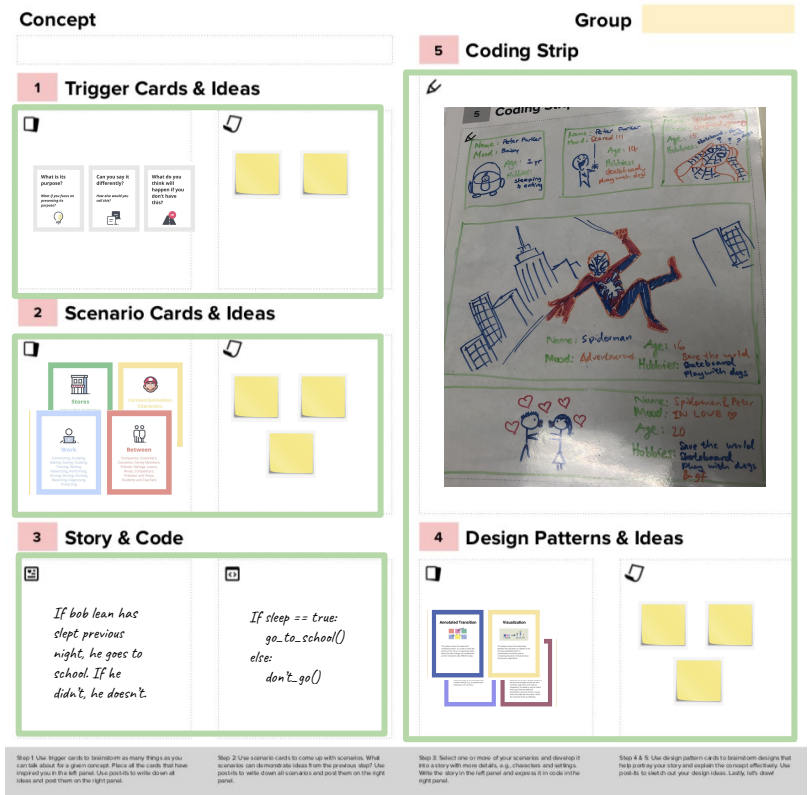


Design Board

Concept Formulation

Story Development

Story-to-Code Mapping



Comic Illustration

Methodology

- **Phase I. Develop Design Process & Tools**
- **Phase II. Evaluate through Design Workshops**

Methodology

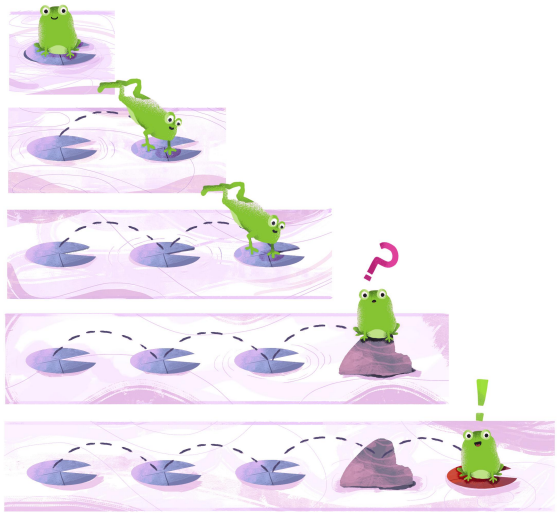
- **Phase I. Develop Design Process & Tools**
- **Phase II. Evaluate through Design Workshops**

Design Workshops (W1 & W2)

- **W1: Undergraduate and graduate students**
  - **13 participants (5M, 8F; age: 17-29, mean 21)**
- **W2: High school computer science teachers**
  - **6 participants (2M, 4F)**
- **2 artists**

	Description	Duration
Session 1 (90 min)	Consent form	5 min
	Pre-study survey	5 min
	Presentation	10 min
	Warm-up: sketch	10 min
	Warm-up: story to code	10 min
	Design session #1	50 min
	Break	15 min
Session 2 (90 min)	Design session #2	50 min
	Discussion	35 min
	Post-study survey	5 min

Design Workshops (W1 & W2)

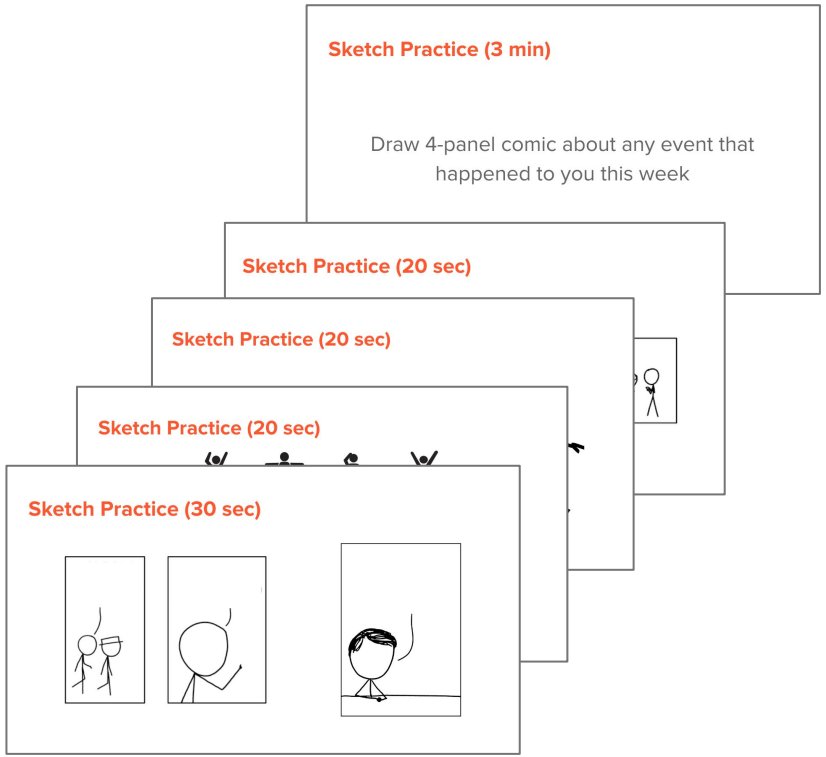


```
String[ ] lilypad = ["green", "green", "green",
"rock", "green", "red"];

While ( lilypad [i] != red) {
      frog.jump();
      if ( lilypad[i] == "rock" ) {
          frog.confused();
          frog.jump();
      }
      i++;
}
```

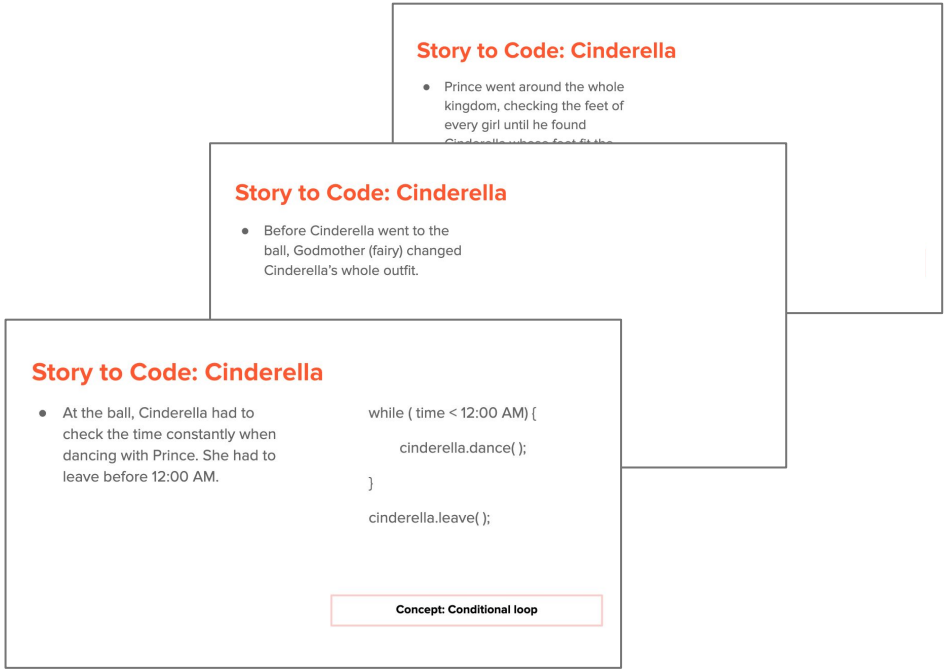
	Description	Duration
Session 1 (90 min)	Consent form	5 min
	Pre-study survey	5 min
	Presentation	10 min
	Warm-up: sketch	10 min
	Warm-up: story to code	10 min
	Design session #1	50 min
	Break	15 min
Session 2 (90 min)	Design session #2	50 min
	Discussion	35 min
	Post-study survey	5 min

Design Workshops (W1 & W2)



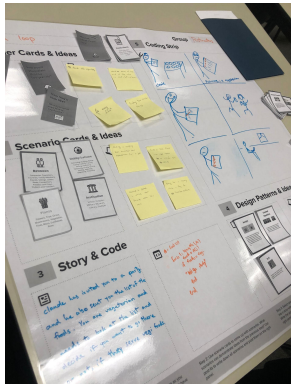
	Description	Duration
Session 1 (90 min)	Consent form	5 min
	Pre-study survey	5 min
	Presentation	10 min
	Warm-up: sketch	10 min
	Warm-up: story to code	10 min
	Design session #1	50 min
	Break	15 min
Session 2 (90 min)	Design session #2	50 min
	Discussion	35 min
	Post-study survey	5 min

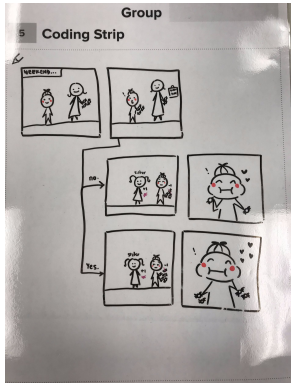
Design Workshops (W1 & W2)

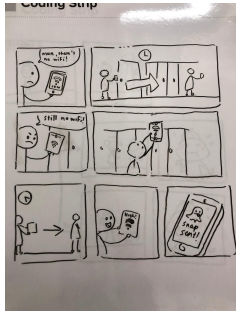


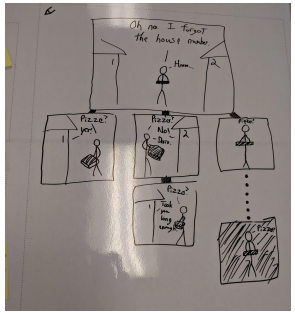
	Description	Duration
Session 1 (90 min)	Consent form	5 min
	Pre-study survey	5 min
	Presentation	10 min
	Warm-up: sketch	10 min
	Warm-up: story to code	10 min
	Design session #1	50 min
	Break	15 min
Session 2 (90 min)	Design session #2	50 min
	Discussion	35 min
	Post-study survey	5 min

Design Workshops (W1 & W2)



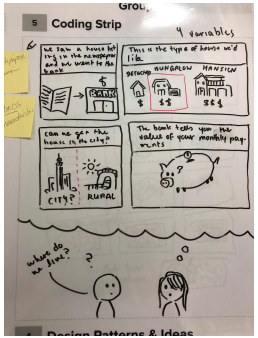


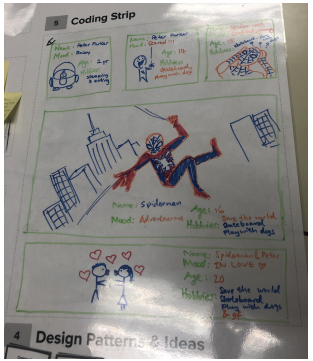


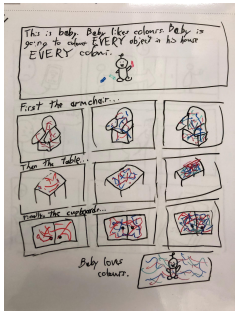


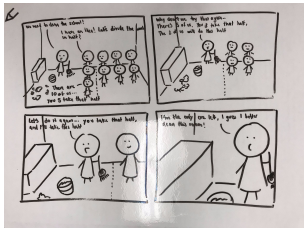
	Description	Duration
Session 1 (90 min)	Consent form	5 min
	Pre-study survey	5 min
	Presentation	10 min
	Warm-up: sketch	10 min
	Warm-up: story to code	10 min
	Design session #1	50 min
	Break	15 min
Session 2 (90 min)	Design session #2	50 min
	Discussion	35 min
	Post-study survey	5 min

Design Workshops (W1 & W2)









	Description	Duration
Session 1 (90 min)	Consent form	5 min
	Pre-study survey	5 min
	Presentation	10 min
	Warm-up: sketch	10 min
	Warm-up: story to code	10 min
	Design session #1	50 min
	Break	15 min
Session 2 (90 min)	Design session #2	50 min
	Discussion	35 min
	Post-study survey	5 min

Design Workshops (W1 & W2)

- **Discussion**
  - Use case scenarios for learning & teaching
  - Suggestions for design process & tools
- **Post-study survey**
  - Effectiveness of design process & tools
  - Perceived usefulness of coding strip

	Description	Duration
Session 1 (90 min)	Consent form	5 min
	Pre-study survey	5 min
	Presentation	10 min
	Warm-up: sketch	10 min
	Warm-up: story to code	10 min
	Design session #1	50 min
	Break	15 min
Session 2 (90 min)	Design session #2	50 min
	Discussion	35 min
	Post-study survey	5 min

Results

Effectiveness of Design Process

- Participants found the design process and tools **useful**
  (M=4.1, SD=0.8 for W1, and M=4.2, S=0.8 for W2)
- Participants found designing coding strip **engaging**
  (M=4.2, SD=1 for W1 | M=4.3, SD=1 for W2)
- Participants did not need much help from the artists

	Description	Duration
Session 1 (90 min)	Consent form	5 min
	Pre-study survey	5 min
	Presentation	10 min
	Warm-up: sketch	10 min
	Warm-up: story to code	10 min
	Design session #1	50 min
	Break	15 min
Session 2 (90 min)	Design session #2	50 min
	Discussion	35 min
	Post-study survey	5 min

Effectiveness of Ideation Cards

- *"How many ideas was each card able to help generate?"*

1.6 ideas per card

1.4 ideas per card

# of trigger cards

M=4.2 SD=1.7

# of ideas

M=6.7, SD=2.9

# of scenario cards

M=3.6 SD=1.7

# of ideas

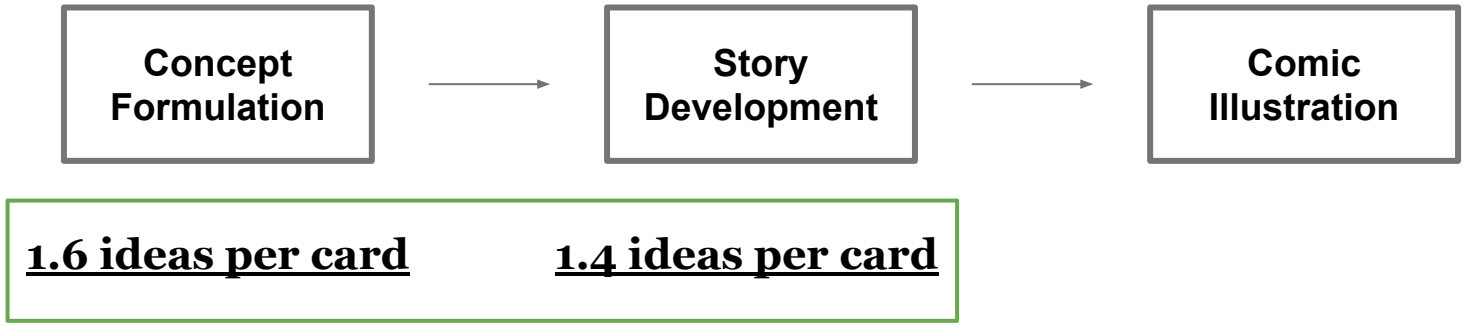
M=5, SD=1.7

# of design cards

M=5.2 SD=2.4

Effectiveness of Ideation Cards

- *"How many ideas was each card able to help generate?"*



# of trigger cards

M=4.2 SD=1.7

# of ideas

M=6.7, SD=2.9

# of scenario cards

M=3.6 SD=1.7

# of ideas

M=5, SD=1.7

# of design cards

M=5.2 SD=2.4

Effectiveness of Ideation Cards

- *"How many ideas was each card able to help generate?"*

1.6 ideas per card

1.4 ideas per card

# of trigger cards

M=4.2 SD=1.7

# of ideas

M=6.7, SD=2.9

# of scenario cards

M=3.6 SD=1.7

# of ideas

M=5, SD=1.7

# of design cards

M=5.2 SD=2.4

Perceived Utility for Learning & Teaching

- **Both teachers and students want to learn with coding strip**

  (5 “yes”, 1 “not sure” for teachers; 10 “yes”, 2 “no” for students)

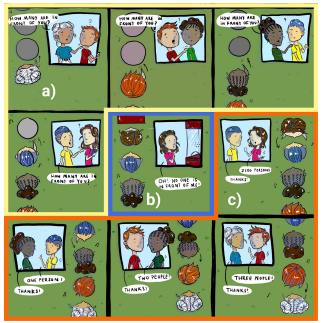
- **While teachers want to teach with coding strip, students are divided**

  (5 “yes”, 1 “not sure” for teachers; 6 “yes”; 7 “not sure” for students)

	Description	Duration
Session 1 (90 min)	Consent form	5 min
	Pre-study survey	5 min
	Presentation	10 min
	Warm-up: sketch	10 min
	Warm-up: story to code	10 min
	Design session #1	50 min
	Break	15 min
Session 2 (90 min)	Design session #2	50 min
	Discussion	35 min
	Post-study survey	5 min

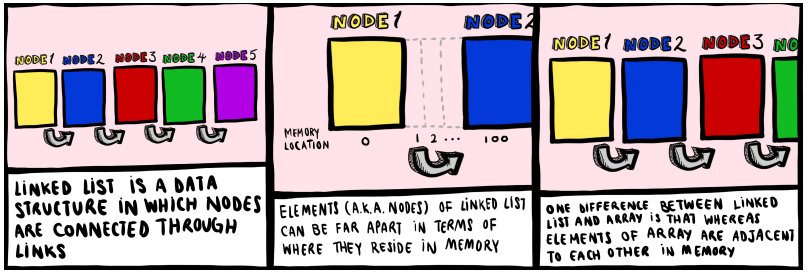
Panel-to-Execution Mapping

- Analysis of 18 coding strips revealed 3 *panel*-to-*execution* mapping patterns



1-to-1

1-to-many



many-to-1

Use Case Scenarios

- Design activity
- Instruction
- Complementary resource in textbooks
- Coding exercise
  - write code based on the presented comics

	Description	Duration
Session 1 (90 min)	Consent form	5 min
	Pre-study survey	5 min
	Presentation	10 min
	Warm-up: sketch	10 min
	Warm-up: story to code	10 min
	Design session #1	50 min
	Break	15 min
Session 2 (90 min)	Design session #2	50 min
	Discussion	35 min
	Post-study survey	5 min

Summary

- The proposed design process and tool appear to be effective at supporting the design of coding strip
- Teachers and students are interested in using coding strip for learning and teaching

Future Work

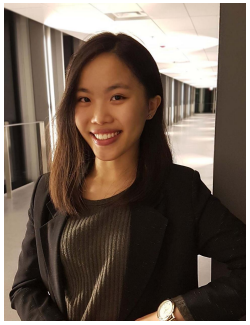
- Develop authoring tool to facilitate the production of coding strips
- Explore how our design process and tools can generalize to work with concepts from other domains (e.g., machine learning, data science, etc.)

Acknowledgements

Edith Law

Martinet Lee

Gracie Xia



Bernadette Cheng

LITE Grant, Center for Teaching Excellence

UNIVERSITY OF WATERLOO

WATERLOO HCI
CHERITON SCHOOL OF COMPUTER SCIENCE

Image References

- https://www.studyinternational.com/news/coding-kids/
- https://www.bbc.com/news/technology-29010511
- https://www.nbcnews.com/news/china/china-pushes-coding-kids-effort-tackle-innovation-gap-n641966
- https://globalnews.ca/news/3693932/teaching-coding-in-canadian-schools-how-do-the-provinces-measure-up/
- https://www.strata-gee.com/japan-makes-coding-mandatory-for-all-students-starting-in-elementary-school/