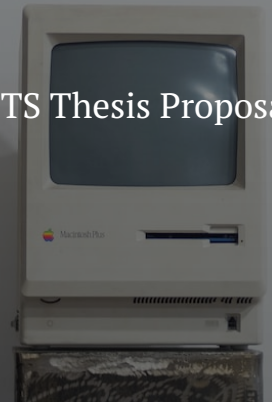
Contextualizing Programming with Algorithmic Art Practices

Computational Thinking, Constructionism, and Script Theory

PHDDTS Thesis Proposal →



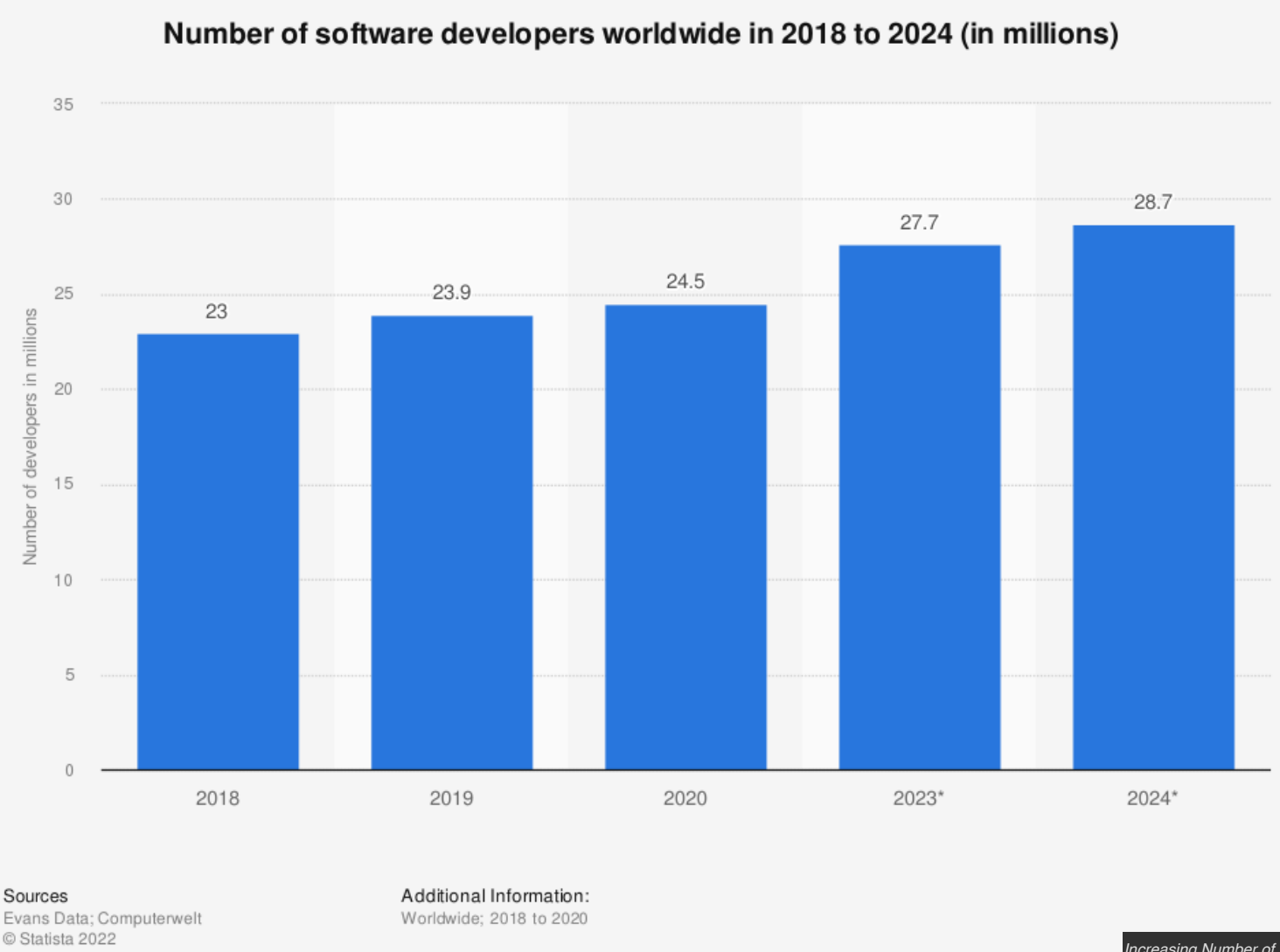
Overview

- Theoretical Basis → Constructionism ~ Pragmatism
- Point of View → ANT ~ Script Theory
- Methods to Contextualize → Computatonal Thinking ~ Programming Fundamentals

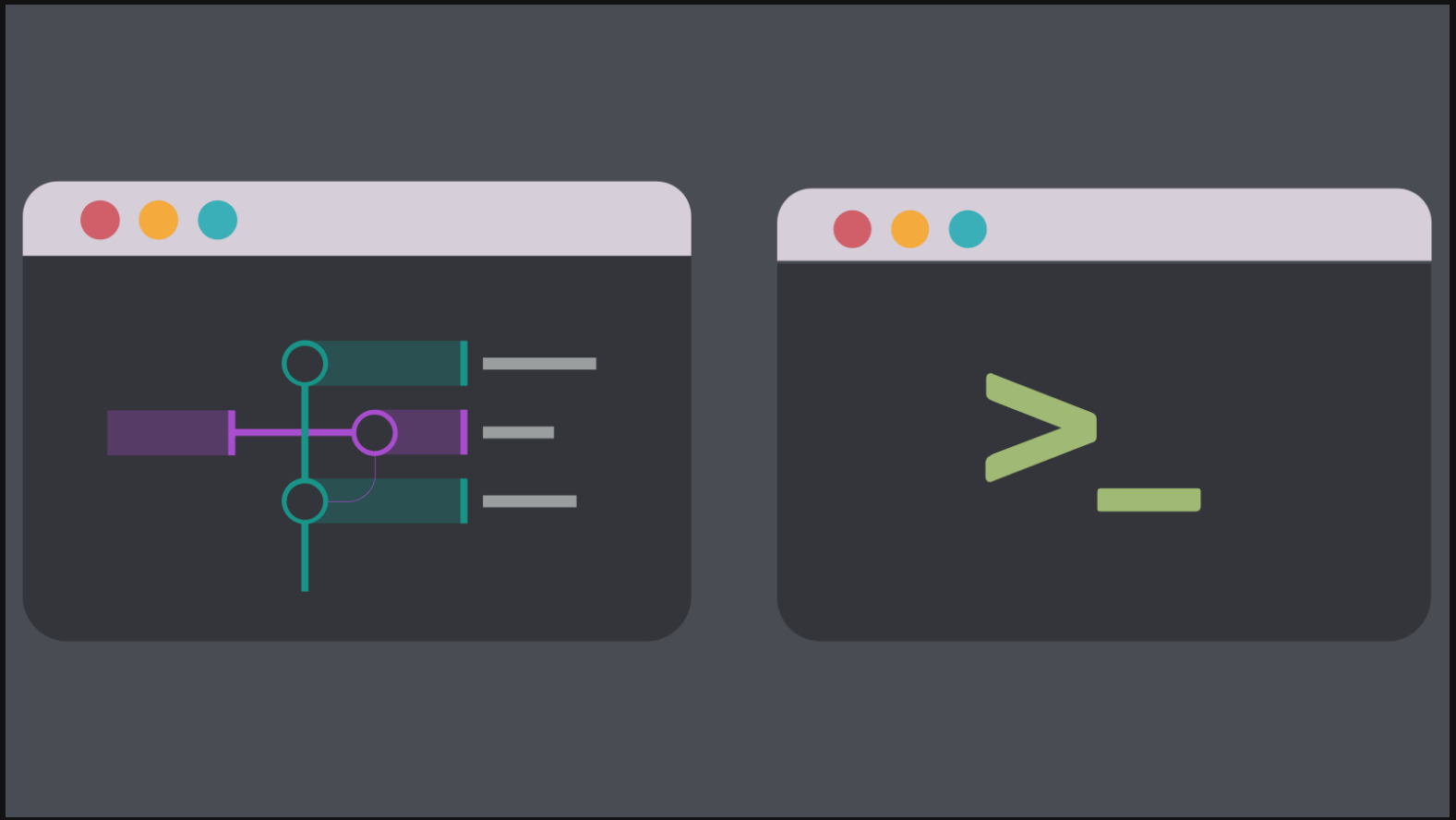
Introduction & Background

Programming knowledge becomes a required skill more than before.

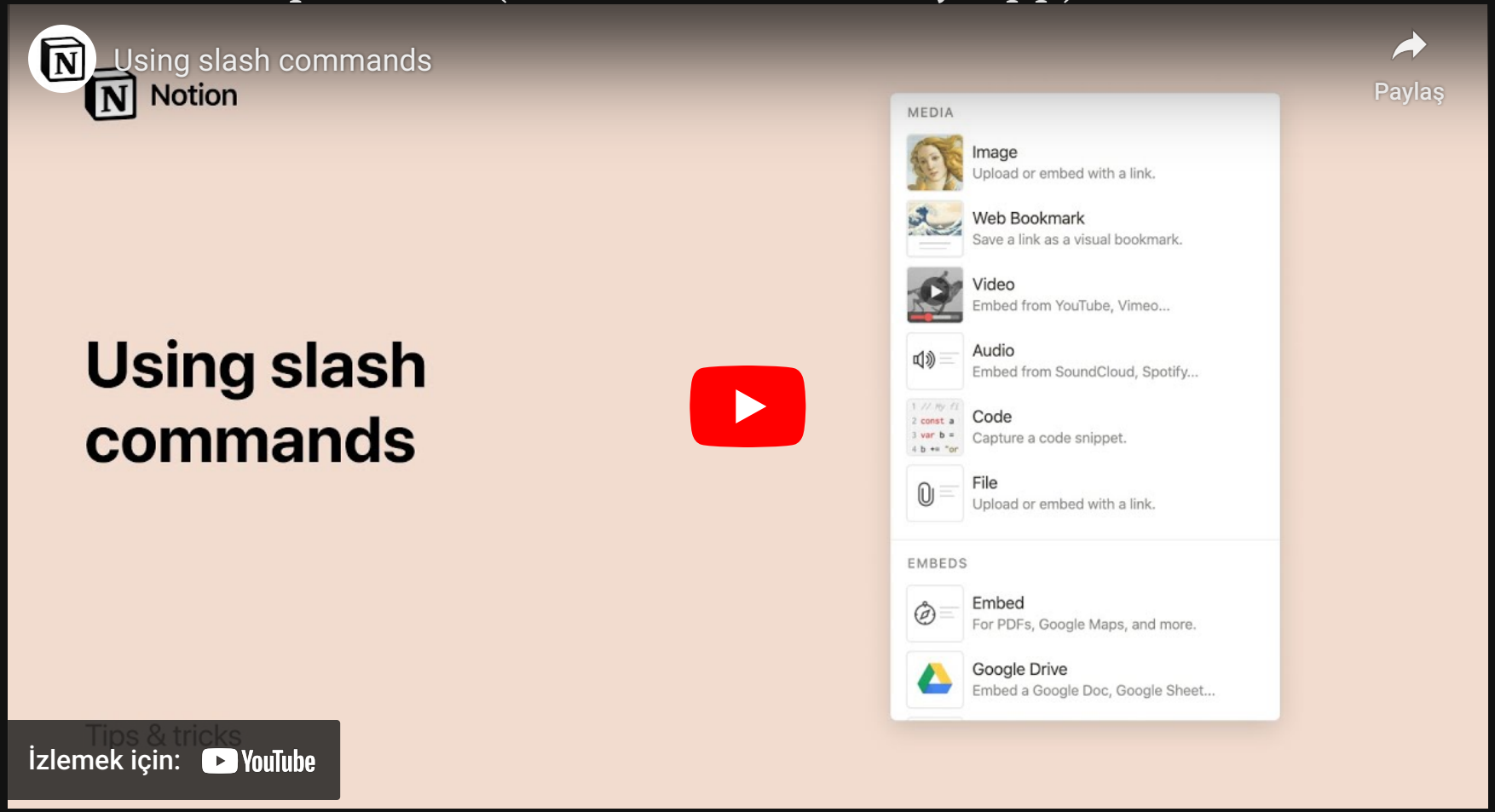
- Paradigmatic Shift in Technology Usage… But Why?
- Increasing Number of Programmers (1.1. The 4th R)
- Changing Grammars in Technology (GUI → CBI) (1.2. Emerging Tendencies)
- Fresh Problem-Solving Paradigms (1.3. Computational Creativity)



Increasing Number of Programmers (Statista, 2020)



CBI → Example Case (Notion Productivity App)



This Presentation is another proof-of-concept

Text-based presentation tool → Sli.dev

slides.md — 2023_Phd_Proposal

break.md | slides.md ● | style.css

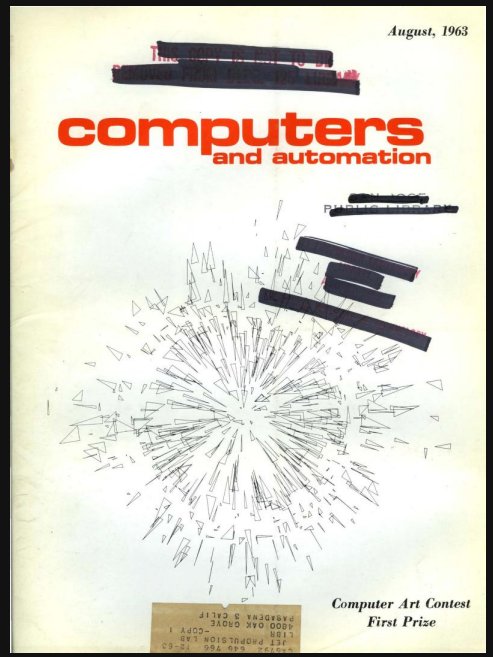
slides.md > # Introduction & Background > ## layout: section

```
58
59  # Introduction & Background
60  Three main reasons on how programming will become more wide-spread.
61
62  - Paradigmatic Shift...
63
64  - Increasing Number of Programmers (1.1. The 4th R)
65
66  - Changing Grammars in Technology (GUI → CBI) (1.2. Emerging Tendencies)
67
68  - Emergence of New Problem-Solving Paradigms (1.3. Computational Creativity)
69
70  ---
71  class: bottom-0
72  ---
73
74  <div scale-90>
75  <img src='/fp1_statistic_id627312_developers-population-worldwide-2018-2024.png'/>
76  </div>
77
78  <div class='caption' top-0>
79  Increasing Number of Programmers (Statista, 2020)
80  </div>
81
82  <!--Growing Population of specific actors in a network results in domination of the crowded
    group. E.g. Democracy in our country. It is not the reflection of Tech. Determinism.-->
83
84  ---
85  class: bottom-0
86  transition: slide-left
87  ---
88
89  <div scale-90>
90  <img src='/git-cli-hero.png'/>
91  </div>
92
93  <div class='caption'>
94  Changing Grammars in Technology GUI → CBI (Command-based Interface)
95  </div>
96
97  <!--New Approaches in Human-Computer Interaction-->
98
99  ---
100 transition: slide-left
101 ---
102
103 ## CBI → Example Case (Notion Productivity App)
104 <Youtube id='cBdyHp_XVFQ?t=32' width='100%' height='100%'/>
105
106 ---
107 layout: section
108 transition: slide-left
109 ---
110
111 ## This Presentation is another proof-of-concept
112
```

New Paradigms For Computational Creativity

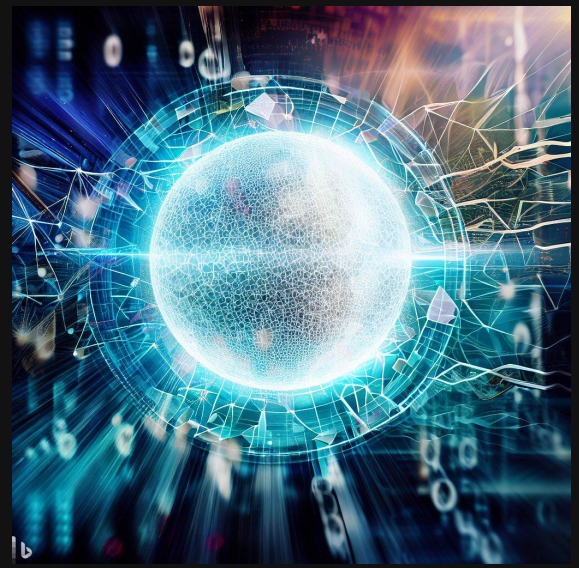
Evolution of computational environments as instruments

Conceptual Art, Algorithmic Art and its sub-branches - 60s



1980-2000 Personal Computers, Discipline-specific software tools

2000-Today Internet, Blockchain, Digital Art, Artificial Intelligence



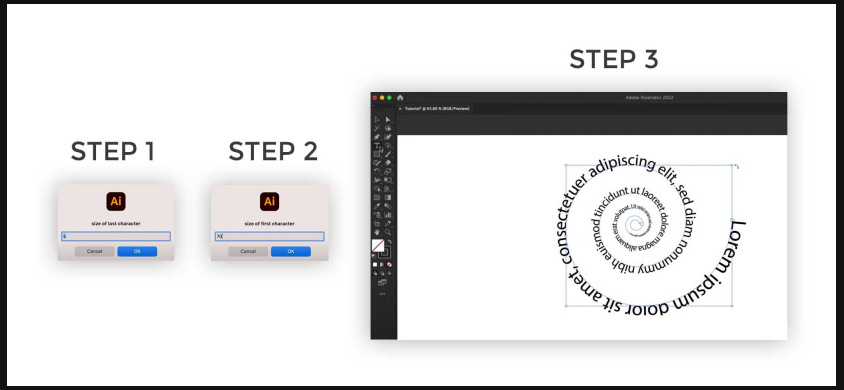
`CBI AI Tools`

Eg. Mid Journey, DALL-E

"Prompt: *A Panda fixing the rooftop.*"

`Scripting`

Eg. Adobe Illustrator




```
var aDoc = app.activeDocument;
if (aDoc.selection.length > 0) {
if (aDoc.selection.length < 2 && aDoc.selection[0].typename == "TextFrame") {
var aTFrame = aDoc.selection[0];
var theChars = aTFrame.characters;
var charLength = theChars.length;
var startSize = prompt("size of first character", 25,"start size");
var endSize = prompt("size of last character", 5,"end size");
var step = (startSize-endSize)/(charLength-1);
for (i = 0; i < charLength; i++) {
    theChars[i].size = (startSize - i*step).toFixed (2);
    redraw();
    }
} else {alert("Please select only one text frame")}
} else {alert("No selection")}
```

Conclusion

The act of being active actors rather than passive consumers. All actors within a social group equally affective.

Encourage individuals to open the Black-Box. How?

ANT → Script Analysis

Constructionist Learning Theories (Helper tools, involving learners, discovery).

Theoretical Background as `Computational Thinking` can help constructing `programming` knowledge.

Visual learners learn more effectively within visual stimuli.

Research Gap

- Contextualizing the knowledge works (Hansen, 2019; Guzdial, 2009).
- There is a need for elaborate approaches on programming educaiton (Brown & Wilson, 2018).
- Most studies focus on assesment of Computational Thinkig.
- Most studies claim that contextualizing programming practices are beneficial.
- But how?

Research Question

How can we contextualize programming fundamentals through algorithmic art practices to improve students' computational thinking skills in design departments?

Scope of the Research

- Design Students at Özyeğin University Faculty of Architecture
- Algorithmic and Conceptual works of art between 1960 - 2000s
- P5js built on Javascript (Hi-level programming language)

Research Contributions

- A dedicated software tool → modern implementation of Turtle p5Utils - pixelRuler.
- Enhancing previous research on programming education (Hansen, 2019).
- Taxonomy of Algorithmic Art Techniques.
- Technical Analysis Approaches (Opening the Black-Box, Problem-solving).
- Qualitative and Quantitative data about a specific social group.

Litereture Review

The literature review section will contain following topics;

- Computational Thinking and Programming
- Script Analysis and Computational Thinking
- Programming Education and Knowledge Creation
  - Programming is a tedious task for students
  - The Prejudice against programming classes
  - Computer Literacy
  - Pedagogical Issues (Wrong Choices)
  - Design Students are Visual Learners
  - Contextualizing Programming with Art. But how?
- Algorithmic Art Practices
  - Tracing Algorithmic Art Practices
  - Generative Art
  - Overview of Algorithmic Art Practices and Programming Fundamentals

Litereture Review Summary

- Programming is a tedious process for students
- The prejudice against coding among students causes declining attendance to computing classes (Allwood, 1986; Winslow, 1996; Robins et al., 2003; Ring et al., 2008; Yardi & Bruckman, 2007).
- The lack of inadequate computer literacy education at earlier ages (Guzdial, 2009; Yardi & Bruckman, 2007)
- The wrong choice of programming language and out-of-date course materials (Brown & Wilson, 2018; Robins et al., 2003; Guzdial, 2009; Hansen, 2019).
- Contextualizing Programming Fundamentals with Art Increases Student Engagement (Liao & Pope, 2008; Guzdial, 2009).
- But How?

Methodology

Theoretical Foundations
Actor-Network Theory: A Birds-eye View of The Research
Research Design

Theoretical Foundations

- Paradigmatic stance between Pragmatism and Constructionism
- Ontology: Interpretation/negotiation of reality.
- Epistemology: Any relevant tool can suit to solve a problem.
- Constructionist Approach → Educational methods should be improved and backed-up within technology (Papert's Turtle/Helper libraries)
- Learners should be involved in the process. Instructing vs. Constructing Knowledge…
- Both defends the experience of the individual.
- Practical application of knowledge ~ Self construction of knowledge

Actor-Network Theory: A birds-eye View of The Research

ANT provides an overall umbrella for the purpose of this research to analyze and develop methods and tools for the study.

- ANT ontologically aligns well with Constructionist and Pragmatist paradigms, pointing out the existence of multiple realities while analyzing the social world.
- The current state of the research problem (engagement of design students in programming classes) represents all the subjects above as macro-actors.
- "Sunsual Objects" from Harman's OOO (student prejudices).

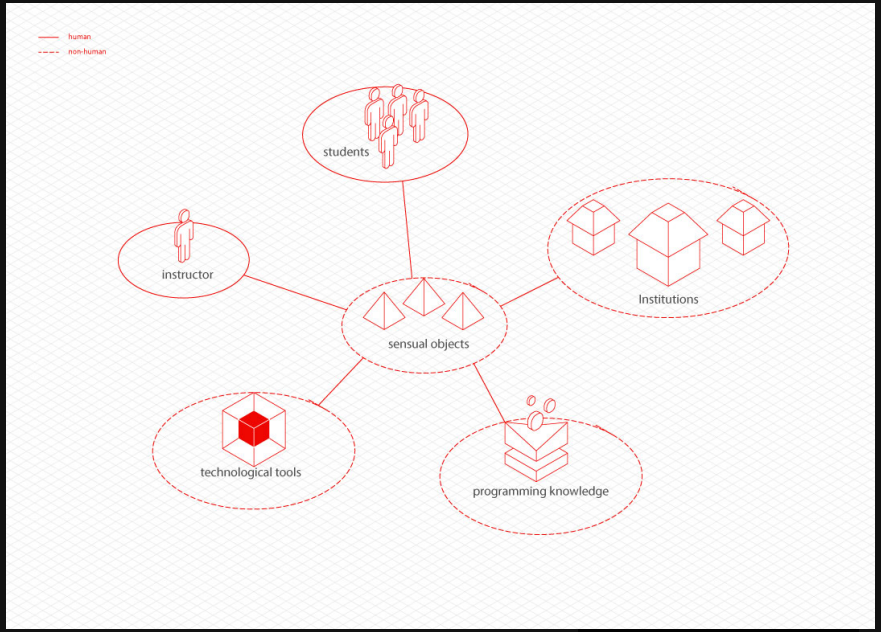
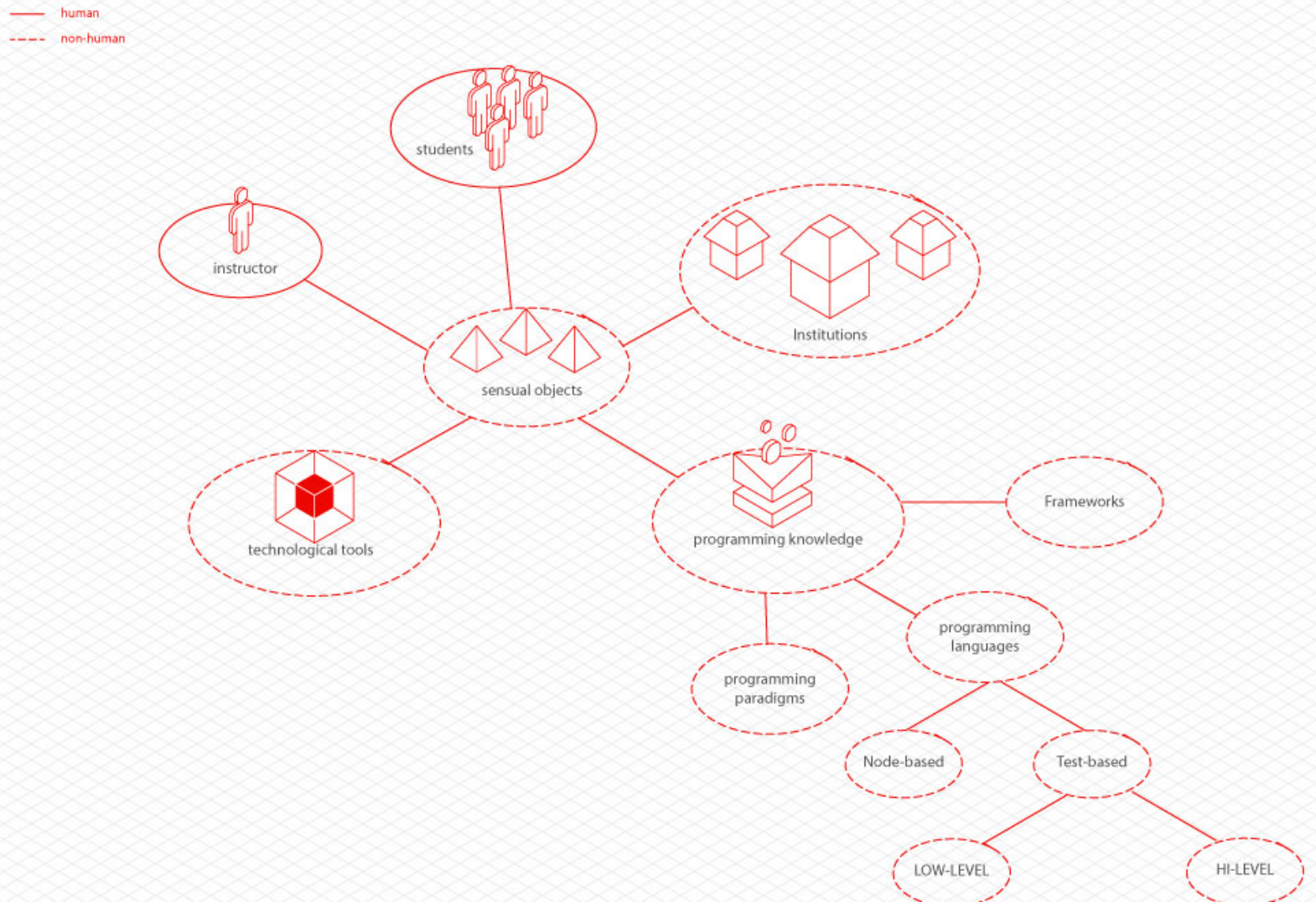


Figure: Research Network Actors


human
non-human
students
instructor
Institutions
sensual objects
technological tools
programming knowledge
Frameworks
programming languages
programming paradigms
Node-based
Test-based
LOW-LEVEL
HI-LEVEL

Research Design

The mixed-method research approach will be used to conduct the empirical (action) research approach study at ÖZÜ in COD 207/208 design students.

The research will be conducted in three phases:

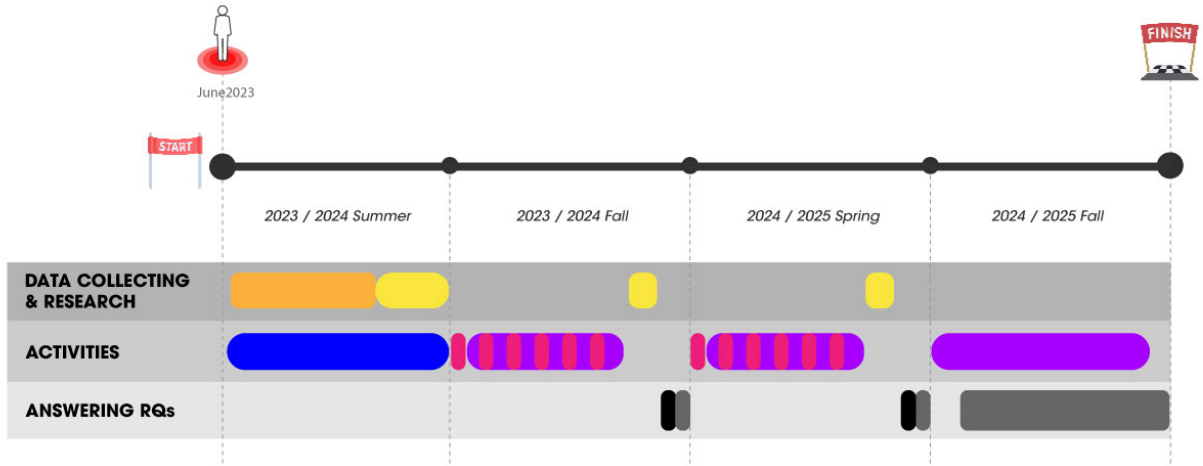
1. Literature Review (CT, AA, GA)
2. Pre- and Post-Survey Design
3. Qualitative Analysis (Interviews and Surveys)

Surveys & Interviews

A Linkert survey will be applied as a pre-test and post-test for students to understand their attitudes toward programming, their beliefs about the usefulness of programming, and their behaviors related to programming.

- Pre-test at the begining of the semester.
- Post-test at the end of the semester.
- Interviews will be carried out during the study.

Timeline



DATA COLLECTING & RESEARCH

- Algorithmic Art Survey (1960-2022)
- Course Evaluation

ACTIVITIES

- p5.Utils Library Revision
- Developing AA~Prog. methods
- Teaching Programming

ANSWERING RQ & SRQ

- Findings
- Anlyze & report the data

The chart displays the order of the planned work flow during the research. The color palette does not signify any priority for the relevant title.

Current Network of Research

Goto network mindmap