b08

The Essence of No-Code Platforms

Building without code, fast!

Logan, Mattie & Kalyana

What are no-code platforms?

Visual development environments that allow users to create applications or websites by dragging and dropping pre-built components, using a graphical user interface (GUI) instead of writing code.

Eliminates the need for extensive coding knowledge, making software development more accessible to a wider audience.

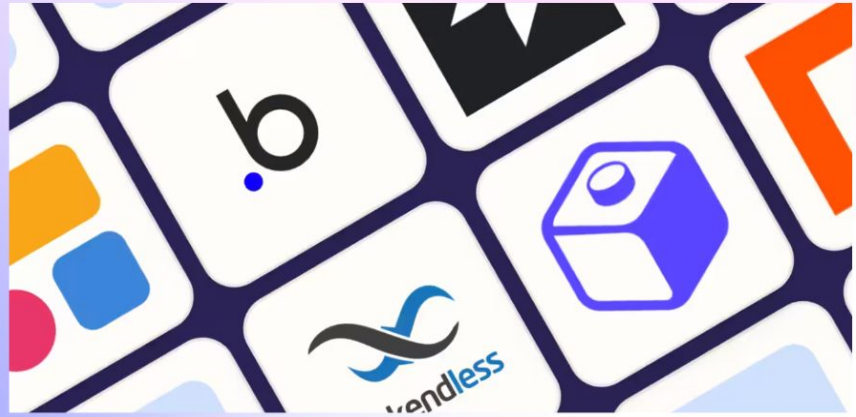
Types of No-Code Platforms

Website Builders

Squarespace, Wix, Framer, etc

App Development Tools

Bubble, Apian, etc



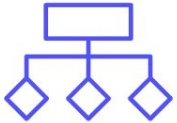
Game Development

Unreal engine

Core Features

Drag & Drop

Allows users to build websites & applications by simply dragging and dropping pre-built components onto a canvas.

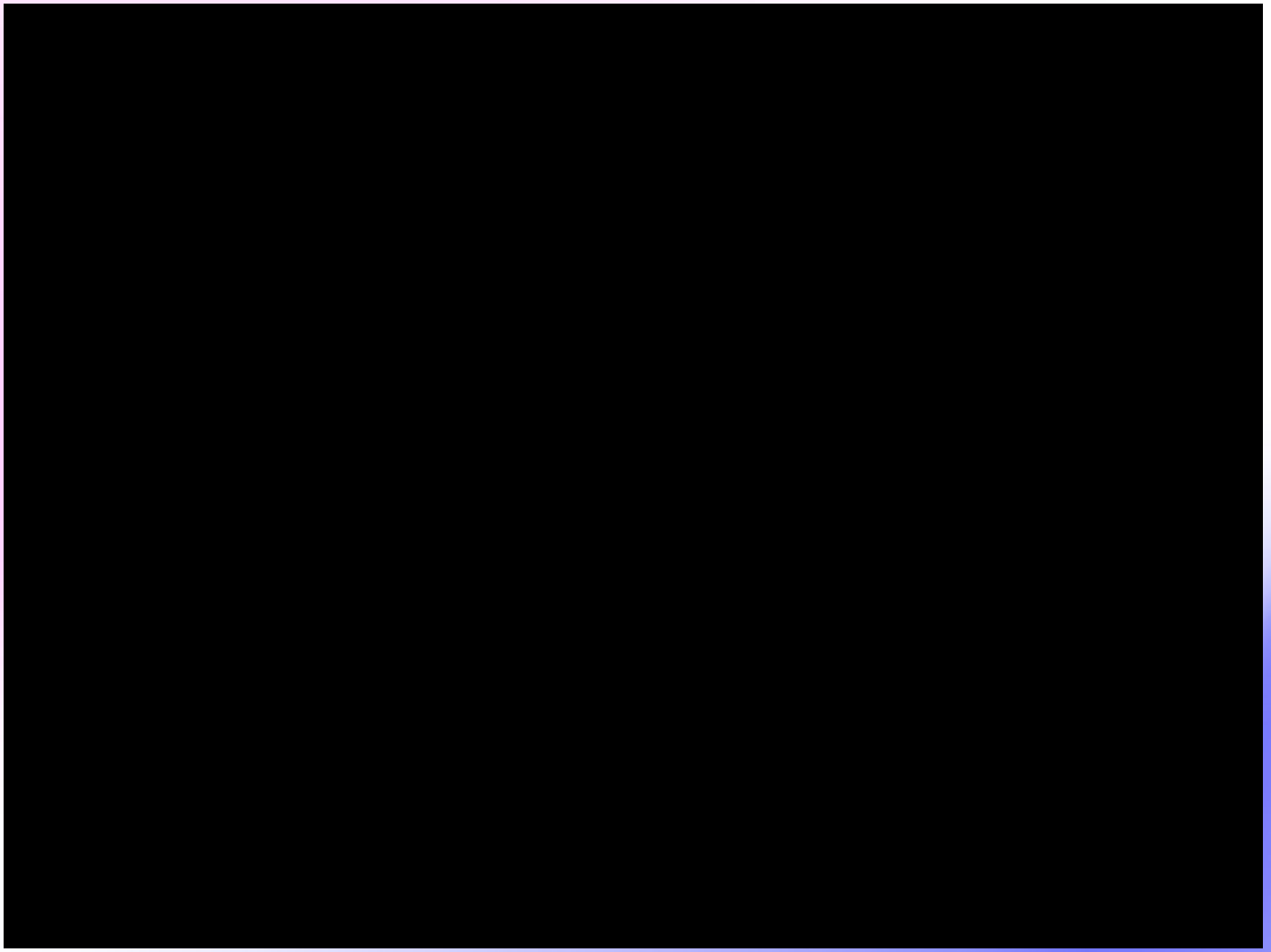


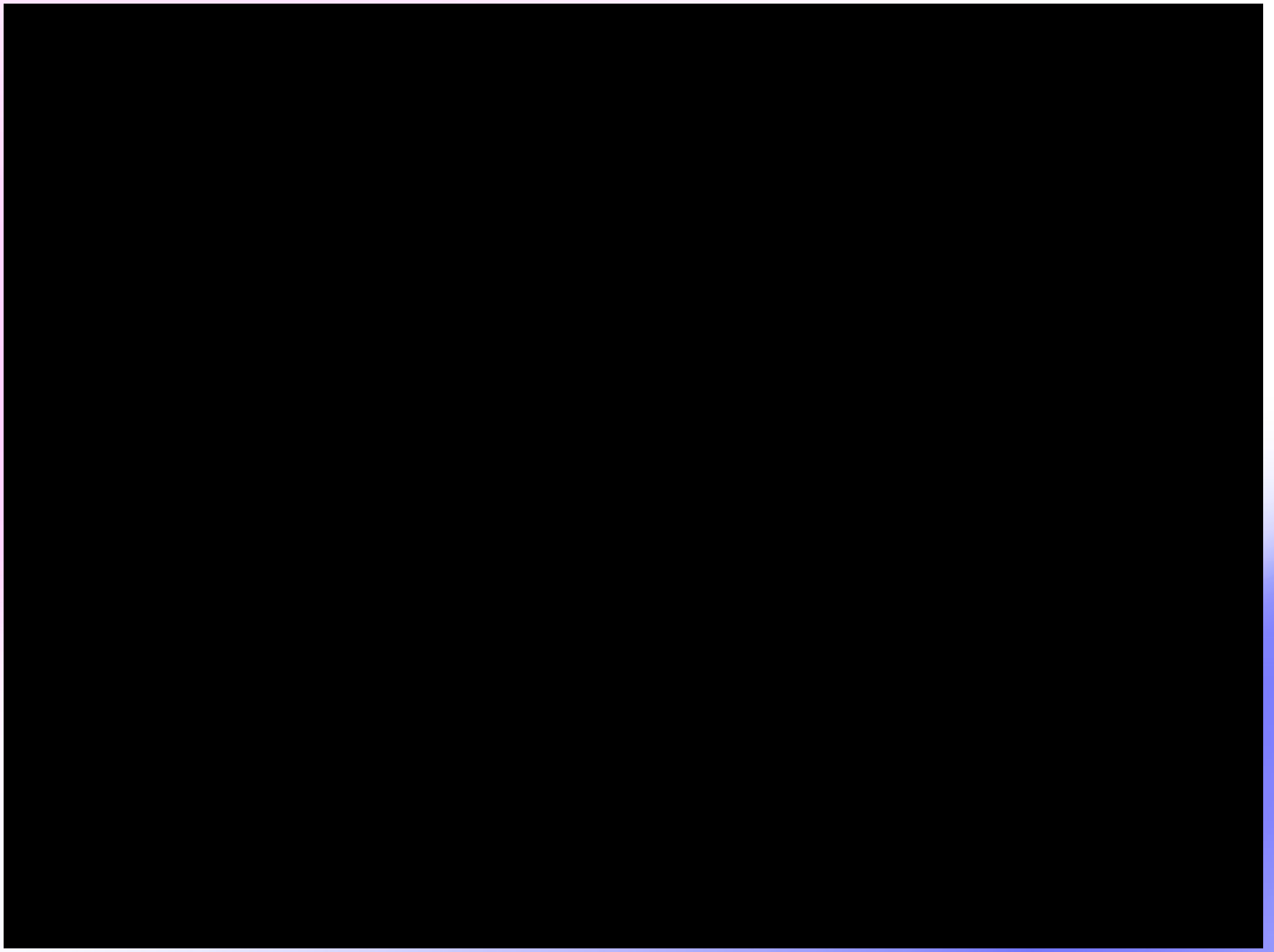
Pre-built Templates

Provides a wide-range of customizable templates for various types of applications or websites

Visual Programming

Enables users to create automated workflows and automate processes using a visual and intuitive representation of code.





b08

No-Code Functionality

How can I handle functionality without code?

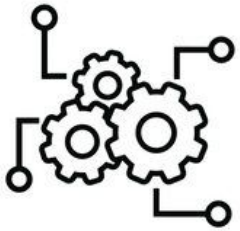
Code is usually used to add functionality to applications

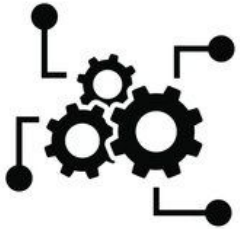
How can no-code apps implement functionality?

- Pre-built blocks
  - Blocks contain built in functionality
  - Purely drag and drop

- Visual programming languages
  - Nocode?
  - Simple or complex depending on the platform

Developers program blocks for designers to use





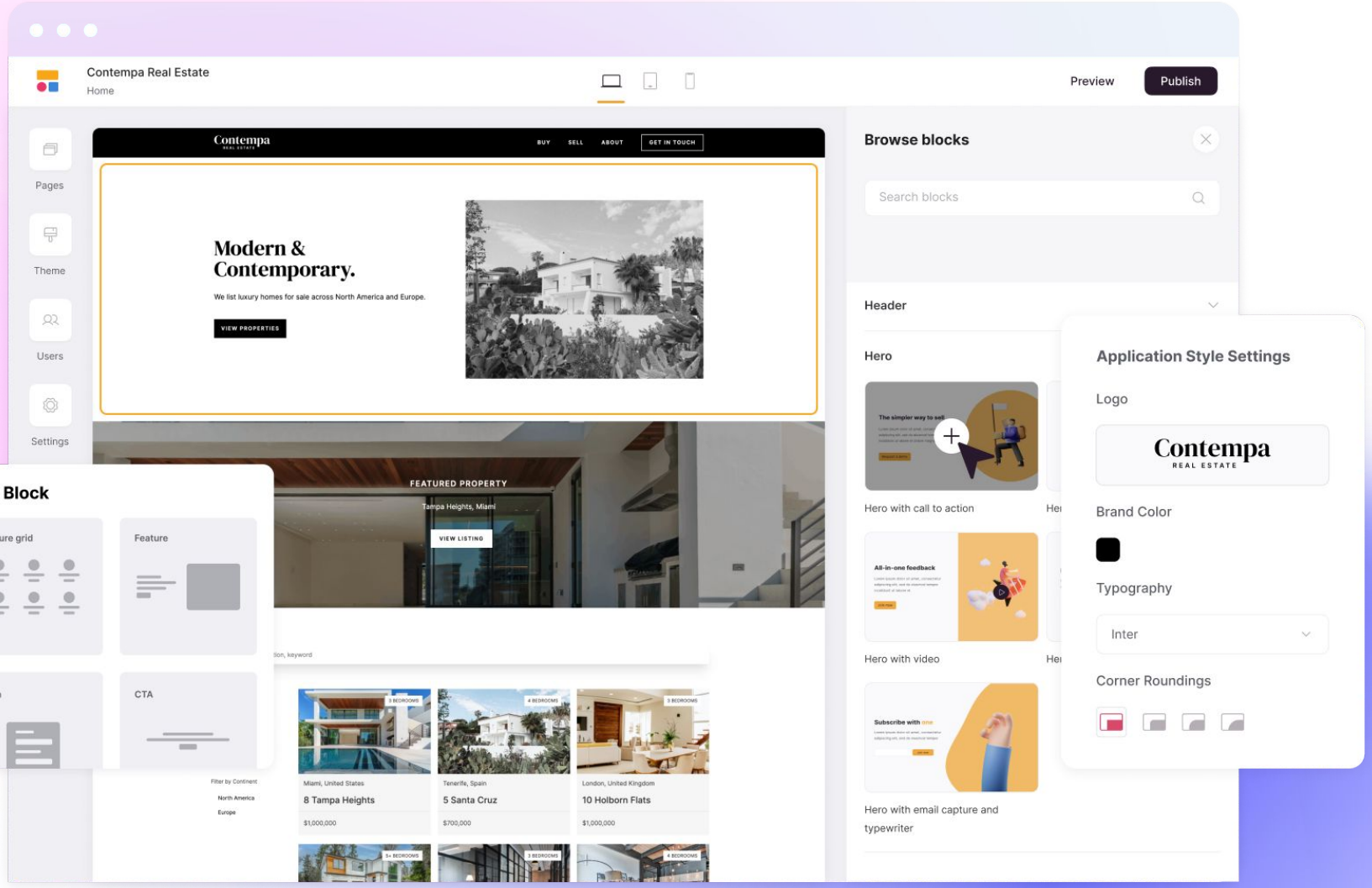
Prebuilt Components

Many no-code platforms simply come with pre-built components with functionality

- Example: Softr

- Often sacrifices flexibility for simplicity

- Sometimes requires improvisation


Contempa Real Estate
Home
Preview
Publish
Pages
Theme
Users
Settings
Contempa
BUY
SELL
ABOUT
GET IN TOUCH
Modern & Contemporary.
We list luxury homes for sale across North America and Europe.
VIEW PROPERTIES
FEATURED PROPERTY
Tampa Heights, Miami
VIEW LISTING
Add Block
Feature grid
Feature
Form
CTA
Filter by Continent
North America
Europe
Miami, United States
8 Tampa Heights
$1,000,000
Tenerife, Spain
5 Santa Cruz
$700,000
London, United Kingdom
10 Holborn Flats
$1,000,000
Browse blocks
Search blocks
Header
Hero
Hero with call to action
Hero with video
Hero with email capture and typewriter
Application Style Settings
Logo
Contempa
REAL ESTATE
Brand Color
Typography
Inter
Corner Roundings

Simple visual programming languages

- Example: Bubble

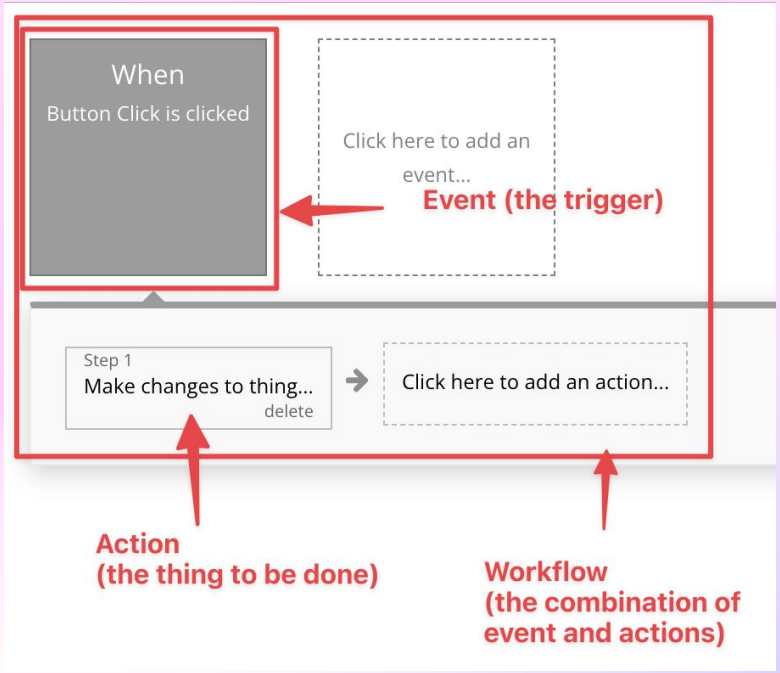
- Workflows: Event->actions

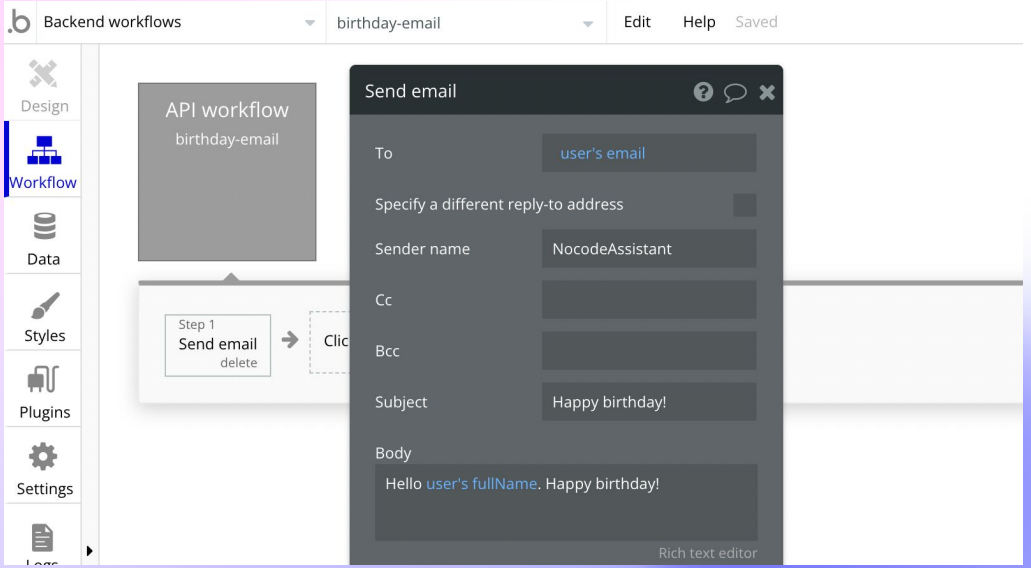
- More flexibility, but not suitable for more complex apps like games

- Data structures, data manipulation

- Actions can also be JavaScript code

- Allows advanced users to have more flexibility using code


When
Button Click is clicked
Click here to add an event...
Event (the trigger)
Step 1
Make changes to thing...
delete
Click here to add an action...
Action
(the thing to be done)
Workflow
(the combination of event and actions)


Backend workflows
birthday-email
Edit
Help
Saved
Design
Workflow
Data
Styles
Plugins
Settings
API workflow
birthday-email
Step 1
Send email
delete
Send email
To
user's email
Specify a different reply-to address
Sender name
NocodeAssistant
Cc
Bcc
Subject
Happy birthday!
Body
Hello user's fullName. Happy birthday!
Rich text editor

Complete programming languages

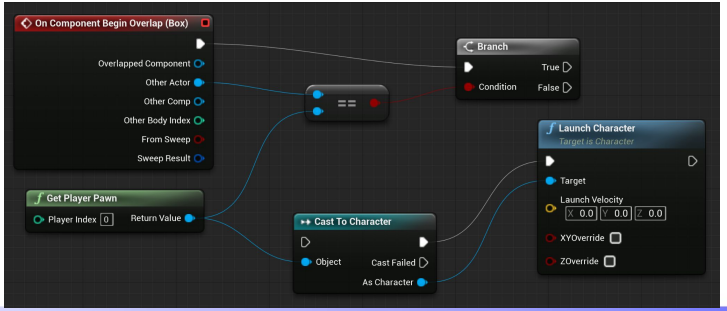
Example: Unreal Engine

Visual scripting language called Blueprints that acts as visual code
Object oriented

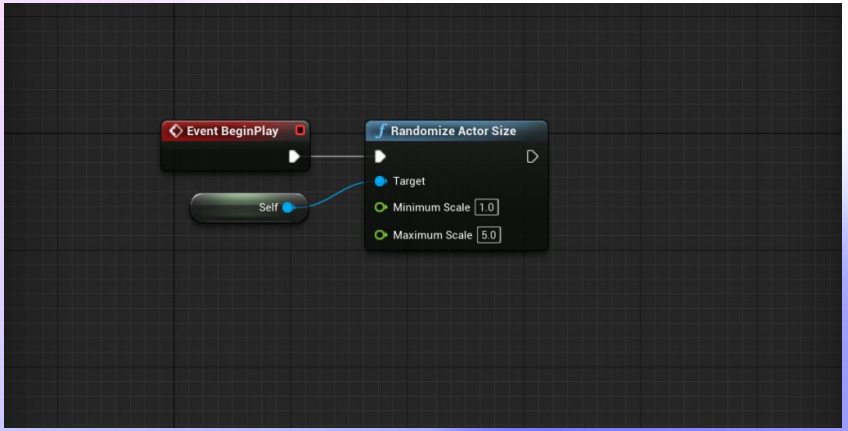
More nodes can be scripted using Blueprints, or using C++

Oftentimes, developers script nodes in C++ for designers to use in Blueprints

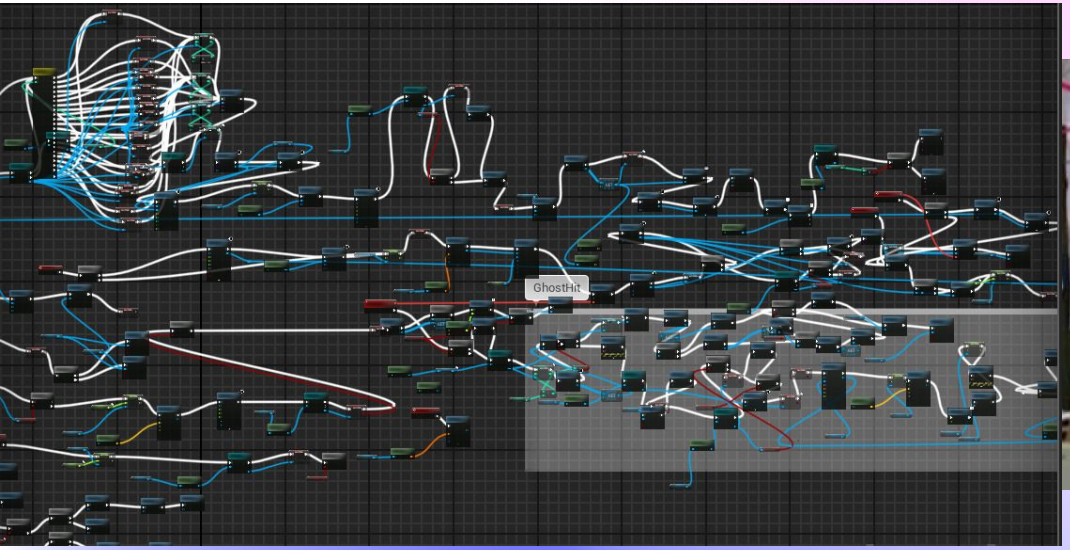
- This workflow is very common for game development
- Designers don't have to know how to program, or only require very little knowledge

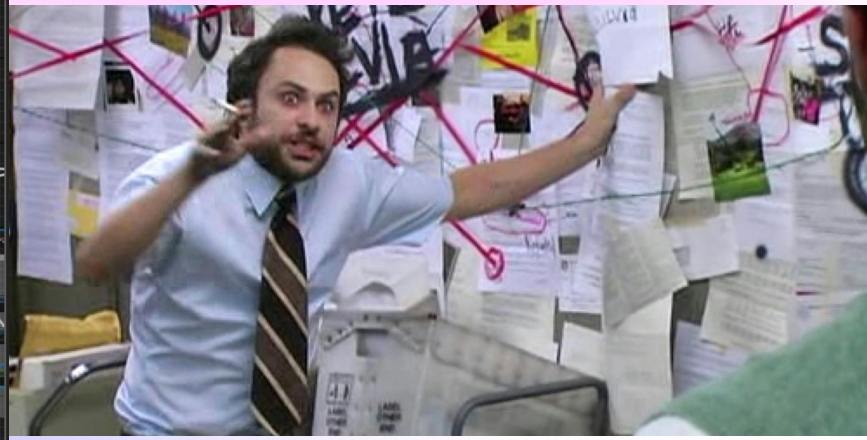


```cpp
void AActorWithCustomNodes::RandomizeActorSize(AActor* Target, const float MinimumScale, const float MaximumScale)
{
    // If the target is not valid, stop.
    if(!IsValid(Target)) return;
    // Get the target's mesh components.
    TArray<UStaticMeshComponent*> MeshPointers;
    Target->GetComponents<UStaticMeshComponent>(MeshPointers, true);
    // If the target actor has no mesh components, stop.
    if(!MeshPointers.Num()) return;
    // Get the first mesh component that is initialized and not marked for kill.
    UStaticMeshComponent* ValidMeshComponent = nullptr;
    for(int32 i = 0; i < MeshPointers.Num(); i++)
        if(IsValid(MeshPointers[i]))
        {
            ValidMeshComponent = MeshPointers[i];
            break;
        }
    // If there isn't one, stop.
    if(!ValidMeshComponent) return;
    // Set the size of the mesh component.
    const float Scale = MinimumScale + FGenericPlatformMath::FRand() * (MaximumScale - MinimumScale);
    ValidMeshComponent->SetWorldScale3D(FVector(Scale, Scale, Scale));
}
```



Why not JUST use visual programming languages?





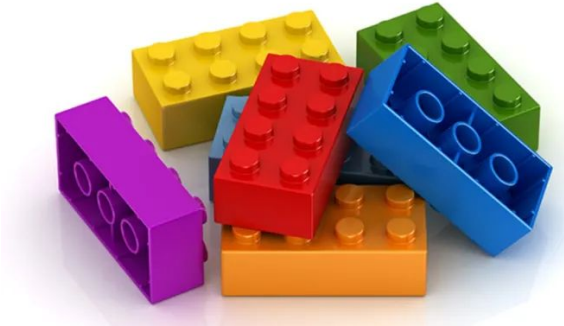
What using these no-code platforms can teach you about programming

Good code is modular (like legos)

Low coupling, high cohesion is extremely important

Programming your apps in specific and independent modules allows for a codebase that is much easier (and more fun) to work in

Hide the complexity of your modules from people who use them
- Example: Softr blocks

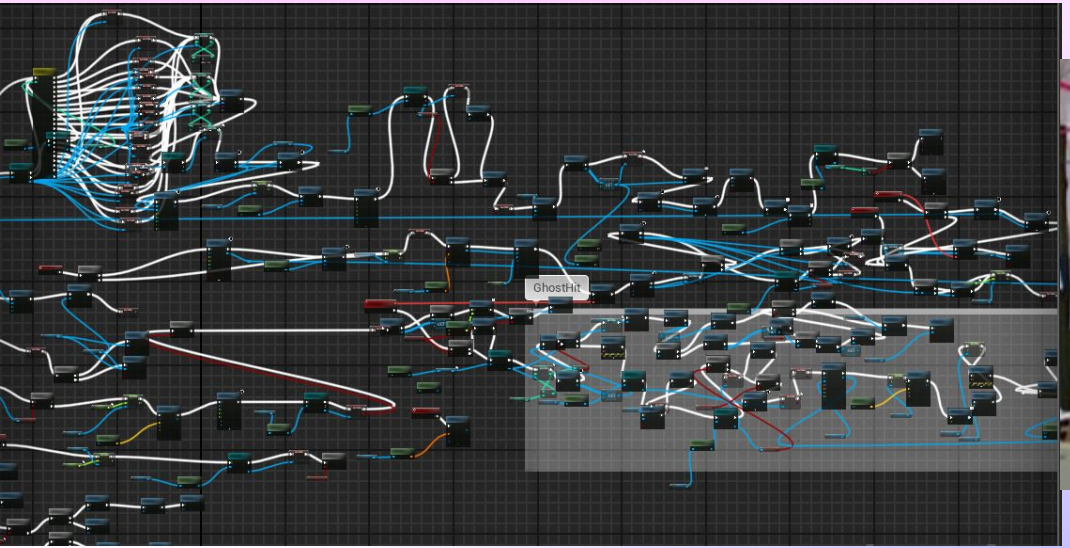


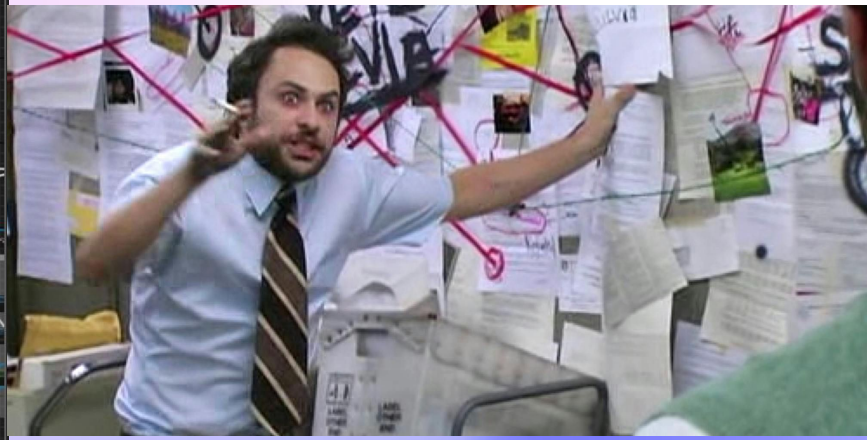
Scratch is the best programming language

Navigating the challenges & looking ahead!

First, the challenges

- Scalability is bad
- Introduces security issues
- Usually more rudimentary and not as capable as textual languages
- Few practical applications outside of teaching
- Significantly bulkier than textual languages


GhostHit



First, the challenges

- Scalability is bad
- Introduces security issues
- Usually more rudimentary and not as capable as textual languages
- Few practical applications outside of teaching
- Significantly bulkier than textual languages

Looking ahead!

- Game creation, workflow management, and website building

- Has a future as a great way to assist game builders who are unfamiliar, uncomfortable, or new to coding

- Some of its future is in the past (LabVIEW)

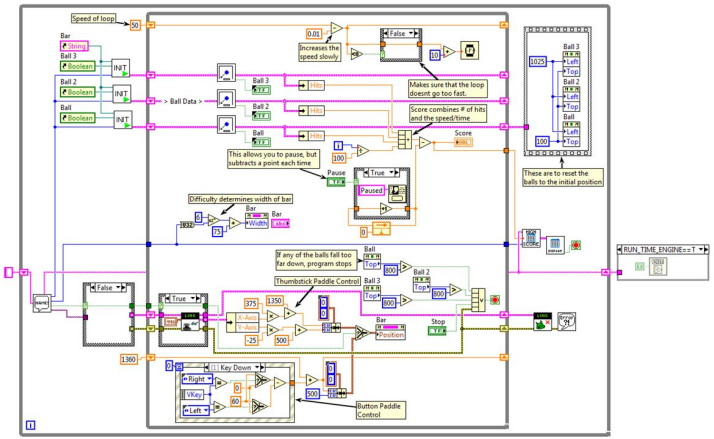
- AI no-code (Pico)

- Could have potential uses when paired with virtual reality

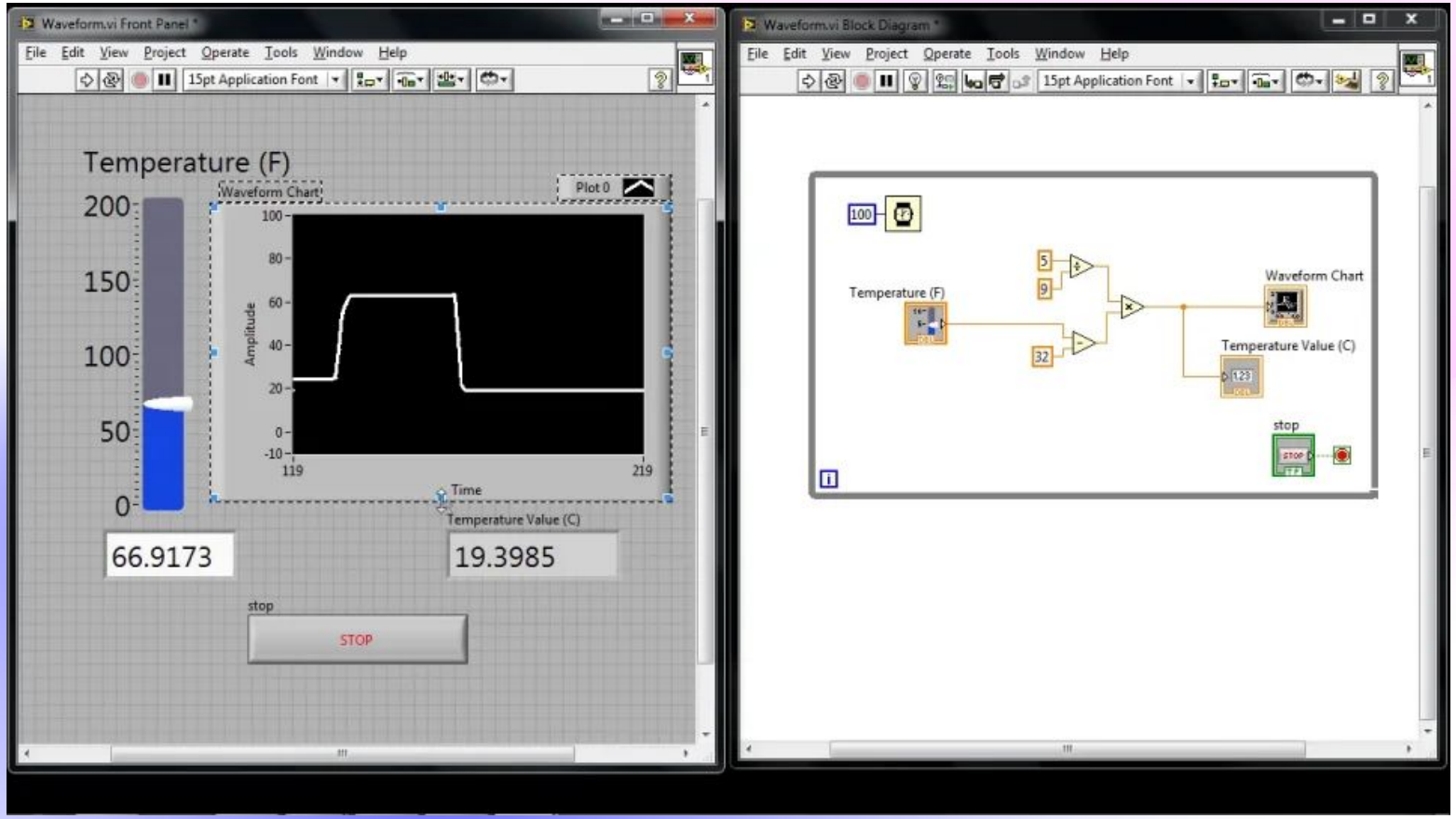
The most promising ones...

LabVIEW

- Developed in 1986 by National Instruments
- Built on C
- Used extensively in Engineering fields
- Visualization of how data flows through a program diagrammatically



What am I looking at?



PICO

- Able to create programs from a text prompt.

- Powered by GPT-4

-Not very powerful, but an interesting proof of concept

- Basically a glorified interface editor

- Strong use of CSS, HTML, and JavaScript

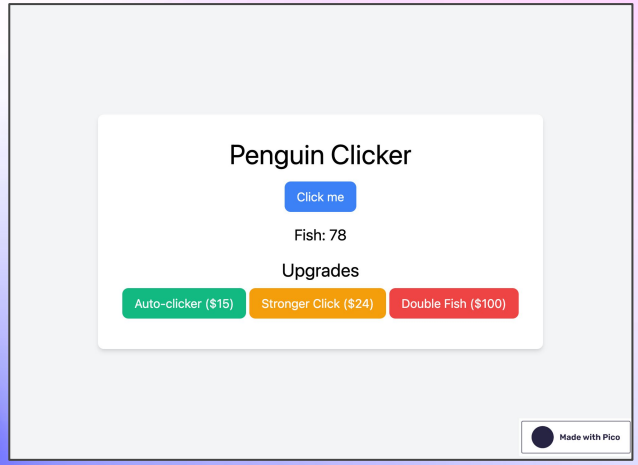
- Also allows for visual editing

- Visual editing seems to be very barebones and not really capable at all

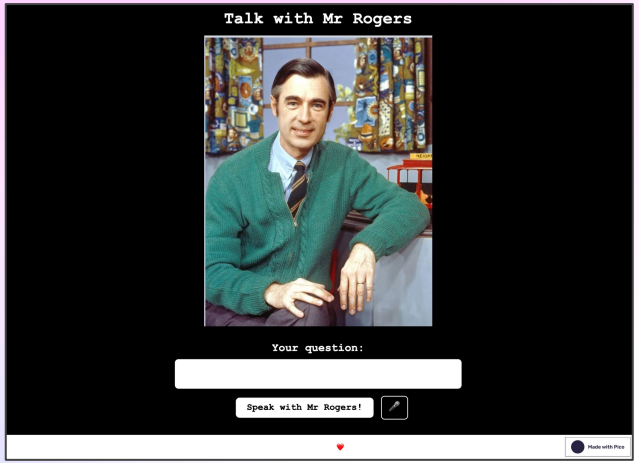
Program showcase

Tweet Generator

Penguin Clicker Game



Mr. Rogers Chatbot

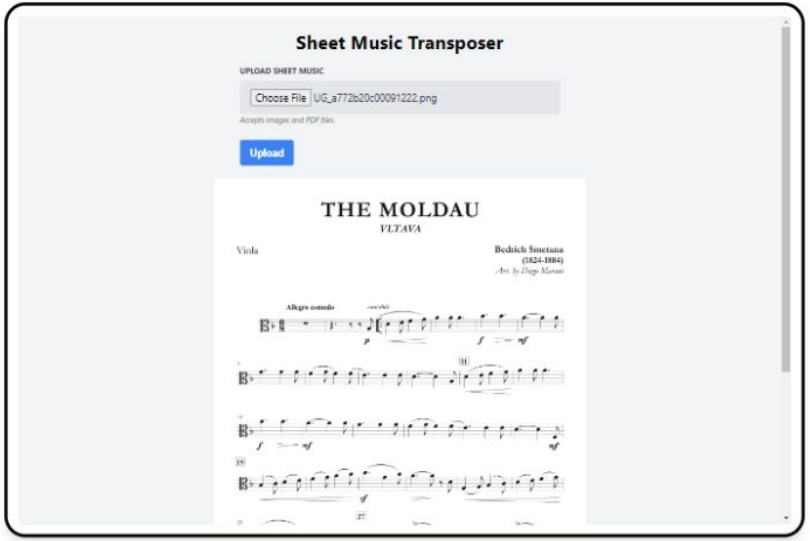


My experience

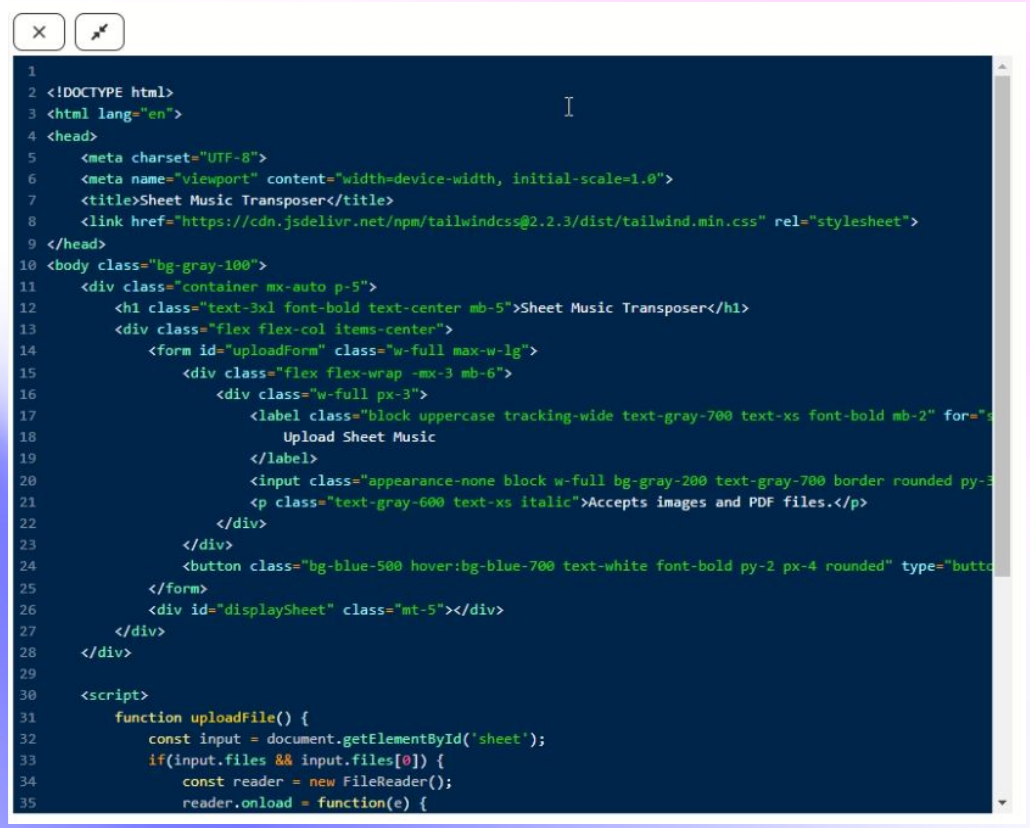
Build me a...

Build me an app which can transpose sheet music which is given to the application in the form of a screenshot or pdf. The app should be able to transpose music between alto clef, bass clef, and treble clef. It also needs to be able to transpose music into another key signature. The app should be able to do this with any key signature. The output needs to keep all other notations within the sheet music in tact, such as accents, crescendos and decrescendos, and anything else that is necessary for playing the song. The interface of this app needs to have an easy way of inputting the given sheet music. The interface should also display the transposed music on the page itself, and also provide a download button to easily download the transposed music in pdf format. In addition to this, the display showing the transposed music should allow the user to move notes up or down on the staff easily by dragging and dropping

Create



My experience



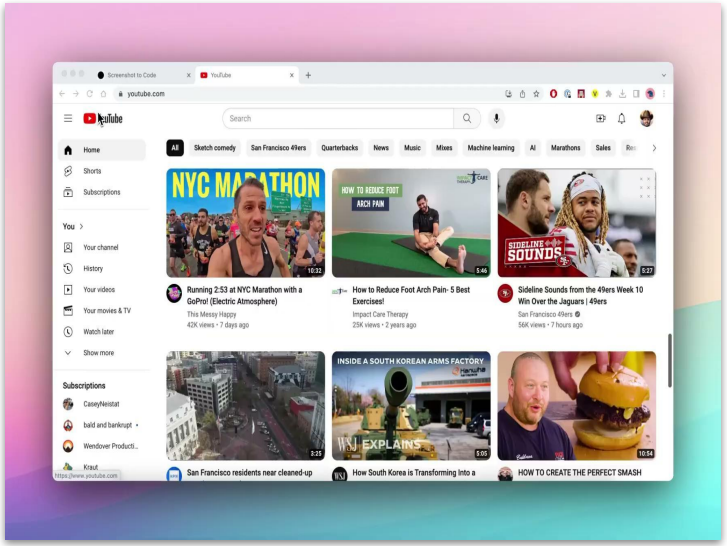
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Sheet Music Transposer</title>
    <link href="https://cdn.jsdelivr.net/npm/tailwindcss@2.2.3/dist/tailwind.min.css" rel="stylesheet">
</head>
<body class="bg-gray-100">
    <div class="container mx-auto p-5">
        <h1 class="text-3xl font-bold text-center mb-5">Sheet Music Transposer</h1>
        <div class="flex flex-col items-center">
            <form id="uploadForm" class="w-full max-w-lg">
                <div class="flex flex-wrap -mx-3 mb-6">
                    <div class="w-full px-3">
                        <label class="block uppercase tracking-wide text-gray-700 text-xs font-bold mb-2" for="g
                            Upload Sheet Music
                        </label>
                        <input class="appearance-none block w-full bg-gray-200 text-gray-700 border rounded py-3
                        <p class="text-gray-600 text-xs italic">Accepts images and PDF files.</p>
                    </div>
                </div>
                <button class="bg-blue-500 hover:bg-blue-700 text-white font-bold py-2 px-4 rounded" type="butto
            </form>
            <div id="displaySheet" class="mt-5"></div>
        </div>
    </div>

    <script>
        function uploadFile() {
            const input = document.getElementById('sheet');
            if(input.files && input.files[0]) {
                const reader = new FileReader();
                reader.onload = function(e) {
```

Ethical concerns of PICO

- Pico creations are hosted on Pico website

- General ethical concerns of generative AI

- Screenshot to Code



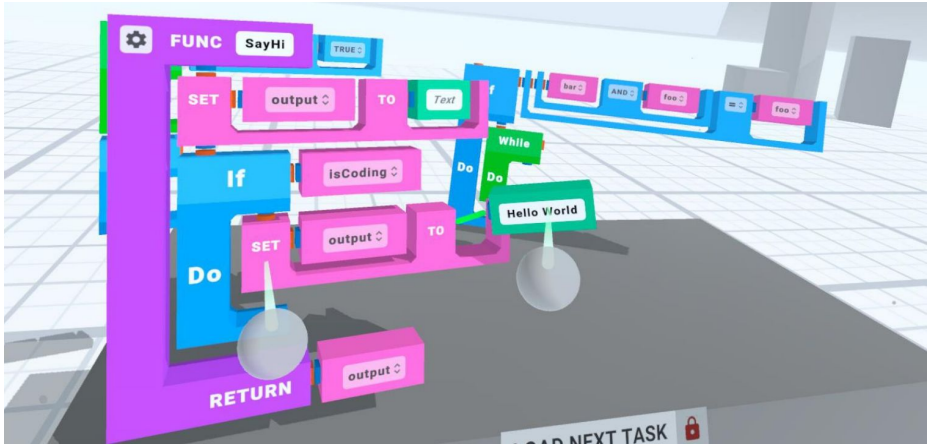
Virtual reality & no-code

- Study conducted by students at the KTH Royal Institute of Technology

- Tested usability of a VR VPL based on Blockly

- Possible to help reduce sedentary behavior

- Not the only test of VR VPLs



KTH Royal Institute of Technology, 2023

Conclusion

- **No-code/low-code platforms allow people to develop applications with little to no code**
- **Functionality in no-code apps is implemented through pre-built blocks which can also be built by traditional programmers, or some kind of visual programming language**
- **No-code/low-code platforms can be useful for learning**
- **The future of no-code could be through AI assistance.**
- **AI still doesn't have unique design sensibilities or understand proper design principles** (design systems, grid systems, typography, psychology of colors, etc)**, designers are still needed for custom solutions**

References

https://kth.diva-portal.org/smash/get/diva2:1794924/FULLTEXT01.pdf
https://docs.makewithpico.com/showcase?ref=NavBar
https://github.com/abi/screenshot-to-code
https://picoapps.xyz/
https://learn.ni.com/learn/article/labview-tutorial
https://labviewwiki.org
https://dl.acm.org/doi/10.1145/3139131.3141785
https://dl.acm.org/doi/10.1145/3379337.3415824
https://docs.google.com/presentation/d/1MD-CgzODFWzdpnYXr8bEgysfDmb8PDV6iCAjH5JIval/edit#slide=id.g1da0625f1b_0_56
https://medium.com/@CodyEngel/good-code-is-like-lego-d9a51dc03ab0
https://bubble.io/
https://www.softr.io/
https://www.unrealengine.com/en-US
https://embrox.com/blog/visual-programming
https://quixy.com/blog/all-about-no-code-development
https://www. squarespace.com
https://www.framer.com

Thanks, questions?

b08