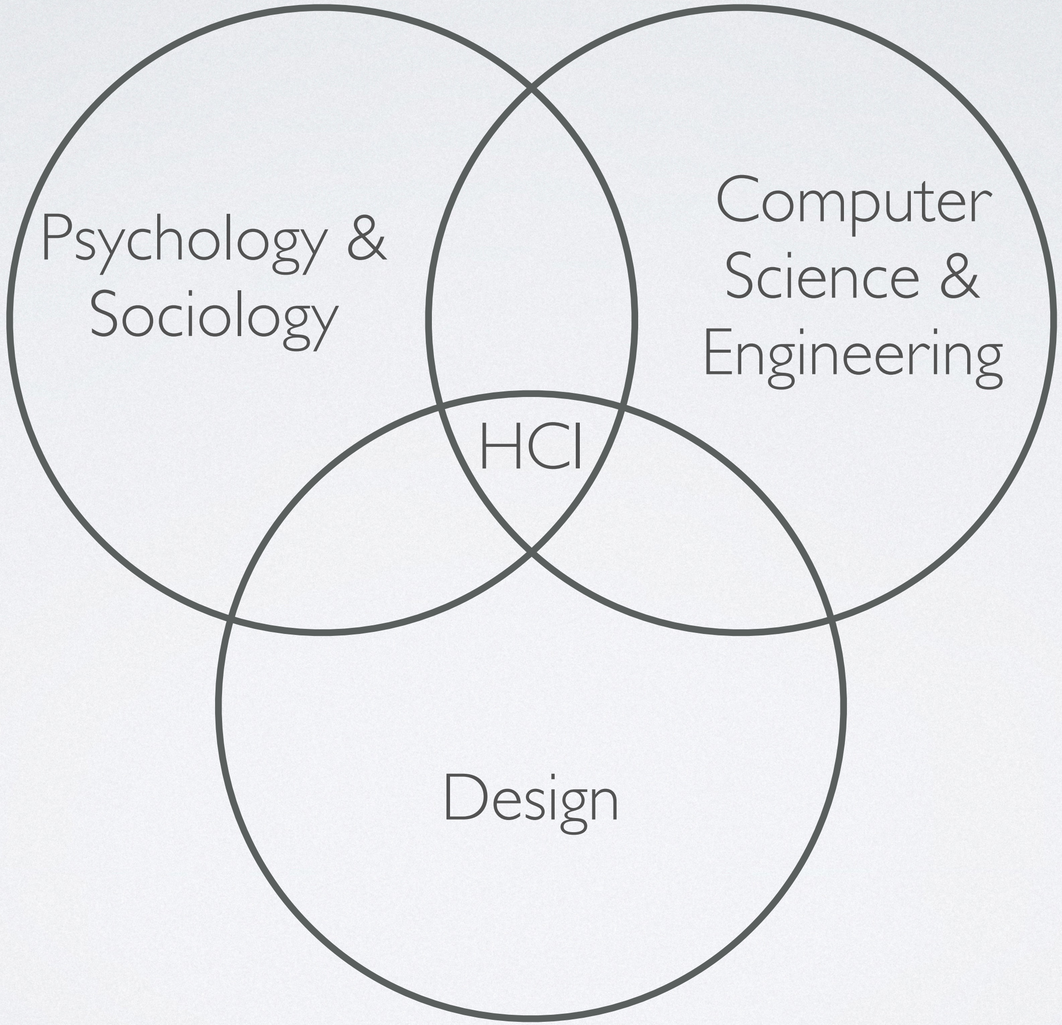
# A PRACTICAL GUIDE TO GENERATIVE MUSIC AI FOR DEVELOPERS

Session 4: Real-Life Collaborations between Artists and Engineers

# OUR SPECIAL GUEST: JORDAN RUDESS

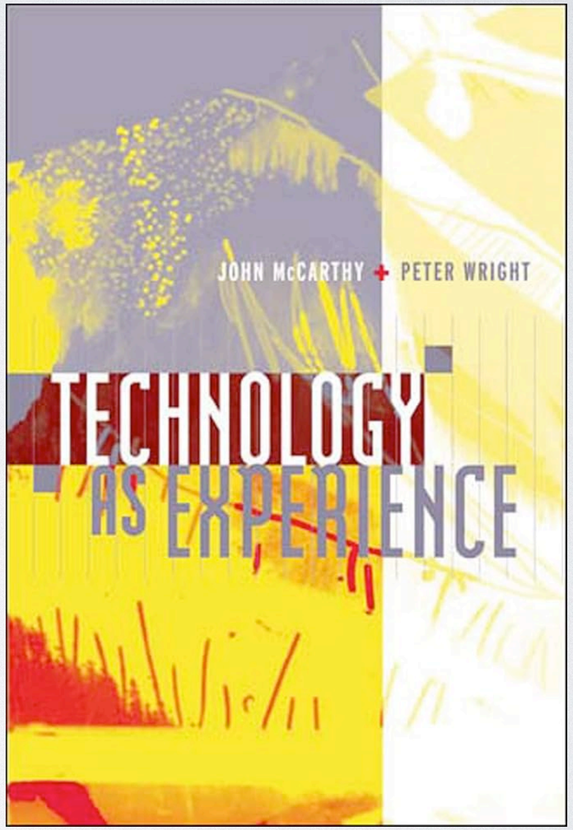
- GRAMMY-winning keyboardist, Dream Theater
- Software Developer & Technologist, Wizdom Music
- Distinguished Visiting Artist at MIT

# A LOOK INTO **HCI**



# A LOOK INTO **HCI**

"HCI is concerned with understanding the influence technology has on how people *think*, *value*, *feel*, and *relate* and using this understanding to inform technology design." *Wright & McCarthy (2008)*

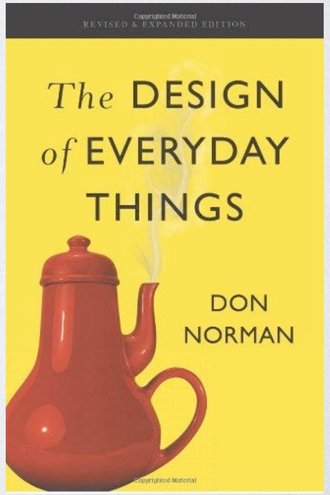


# A LOOK INTO **HCI**

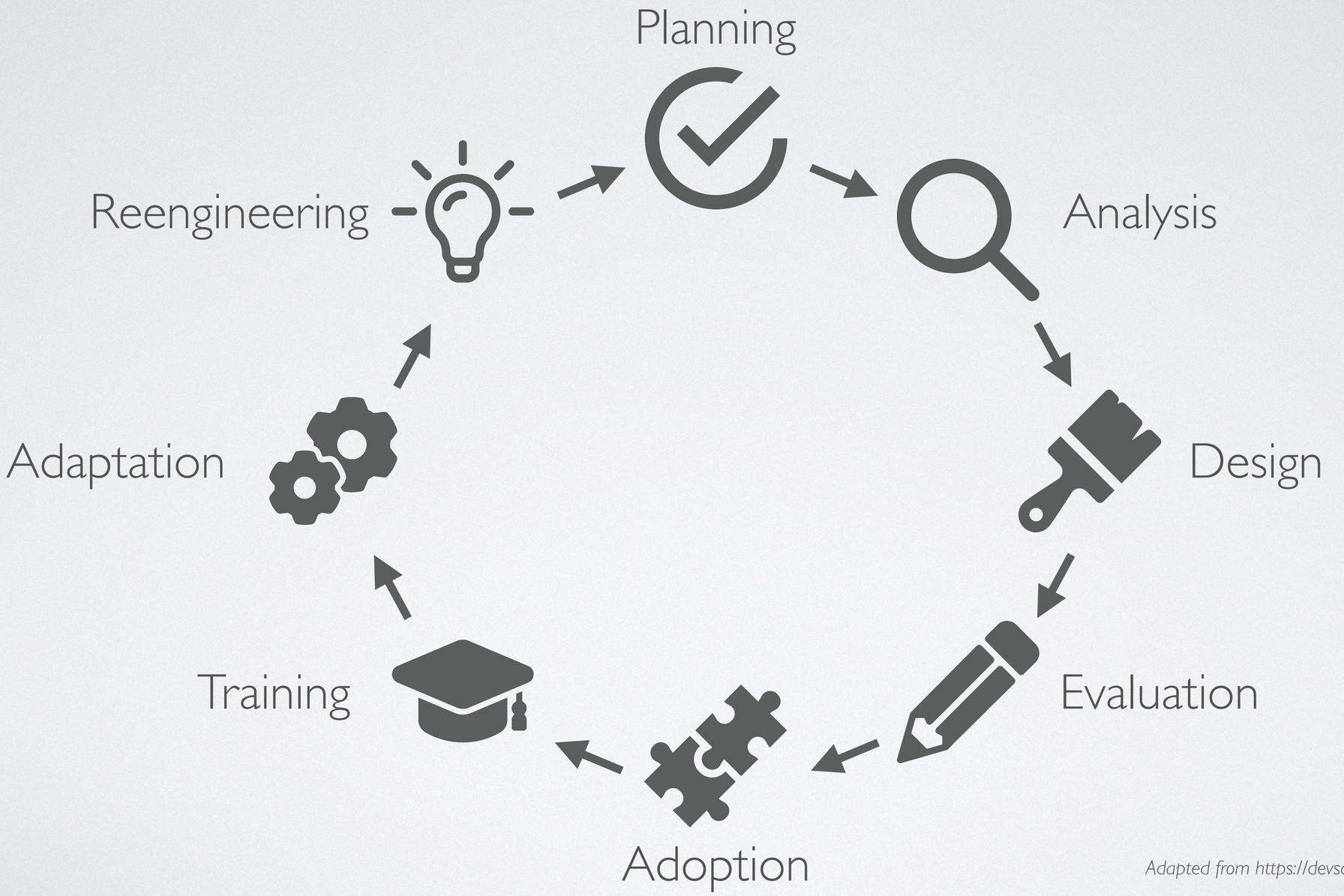
*https://99percentinvisible.org/article/norman-doors-dont-know-whether-push-pull-blame-design/*

"Norman Doors":

Doors with bad design

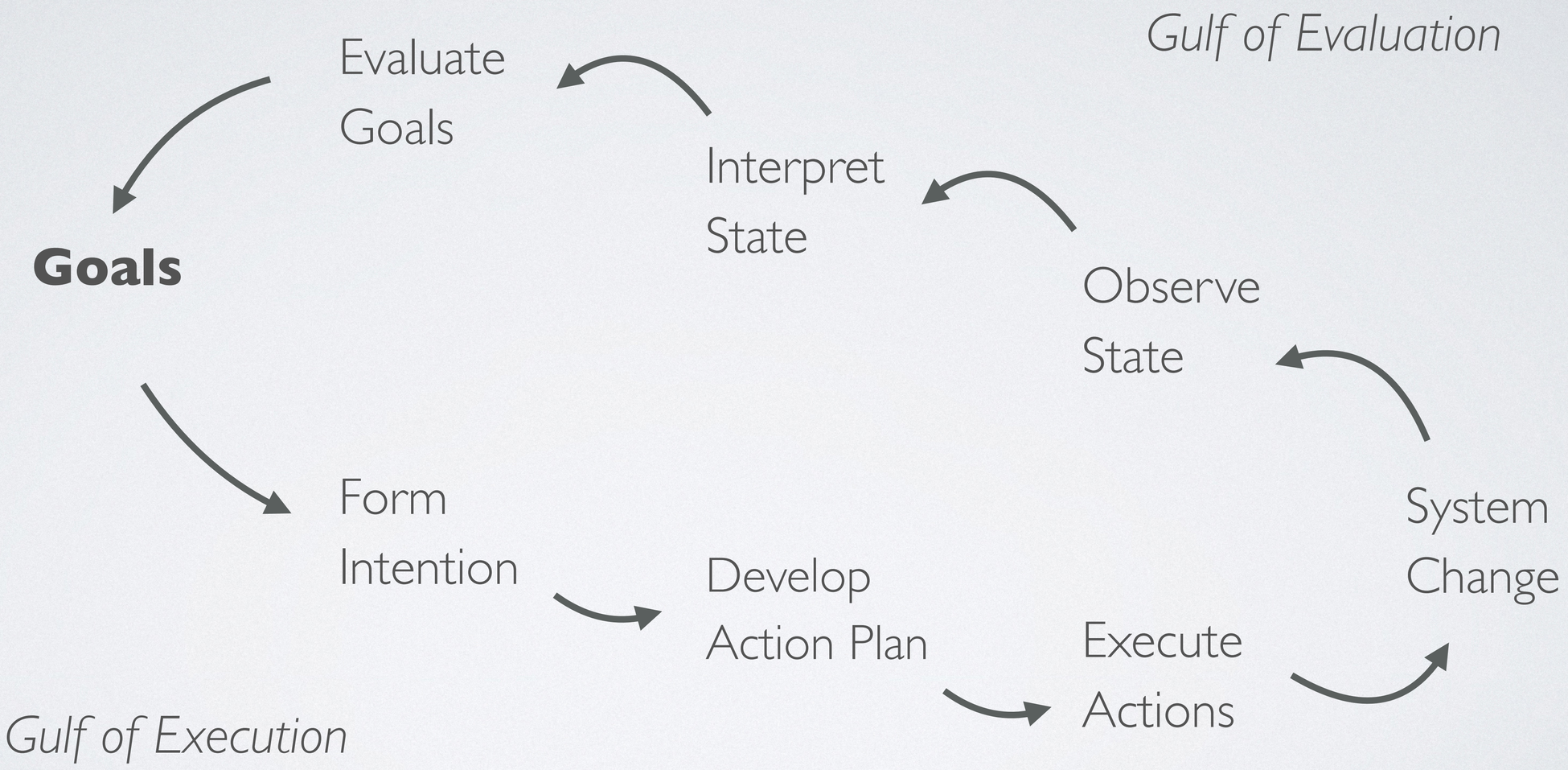


# USER-CENTERED DESIGN (UCD)




*Adapted from https://devsquad.com/blog/user-centered-design-process*

# NORMAN'S EXECUTION-EVALUATION CYCLE




*Adapted from UW CSE440*

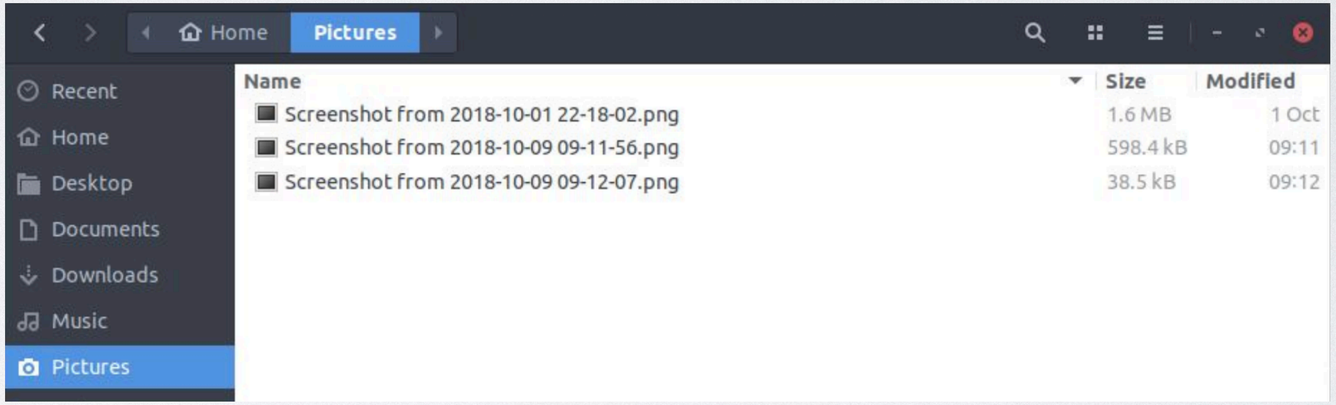
# NORMAN'S EXECUTION-EVALUATION CYCLE

*Gulf of Execution*

*Adapted from UW CSE440*

# NORMAN'S EXECUTION-EVALUATION CYCLE

*Gulf of Evaluation*




*Adapted from UW CSE440*

# USER-CENTERED DESIGN IN MUSIC

*Logic Pro ES2*

*Xfer Records Serum*

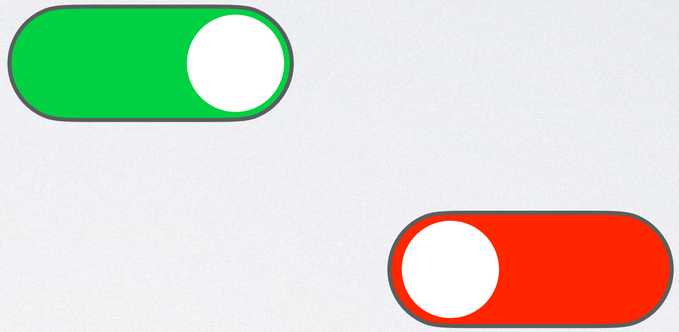
# DESIGN PRINCIPLES

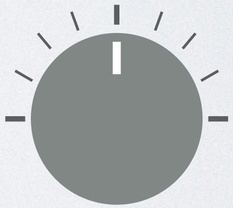
- Affordances
- Constraints
- Feedback
- Consistency
- Metaphors
- Mappings
- Visibility

*Adapted from UW CSE440*

# DESIGN PRINCIPLES

- **Affordances**
- Constraints
- Feedback
- Consistency
- Metaphors
- Mappings
- Visibility





*Adapted from UW CSE440*

# DESIGN PRINCIPLES

- Affordances
- **Constraints**
- Feedback
- Consistency
- Metaphors
- Mappings
- Visibility

*Adapted from UW CSE440*

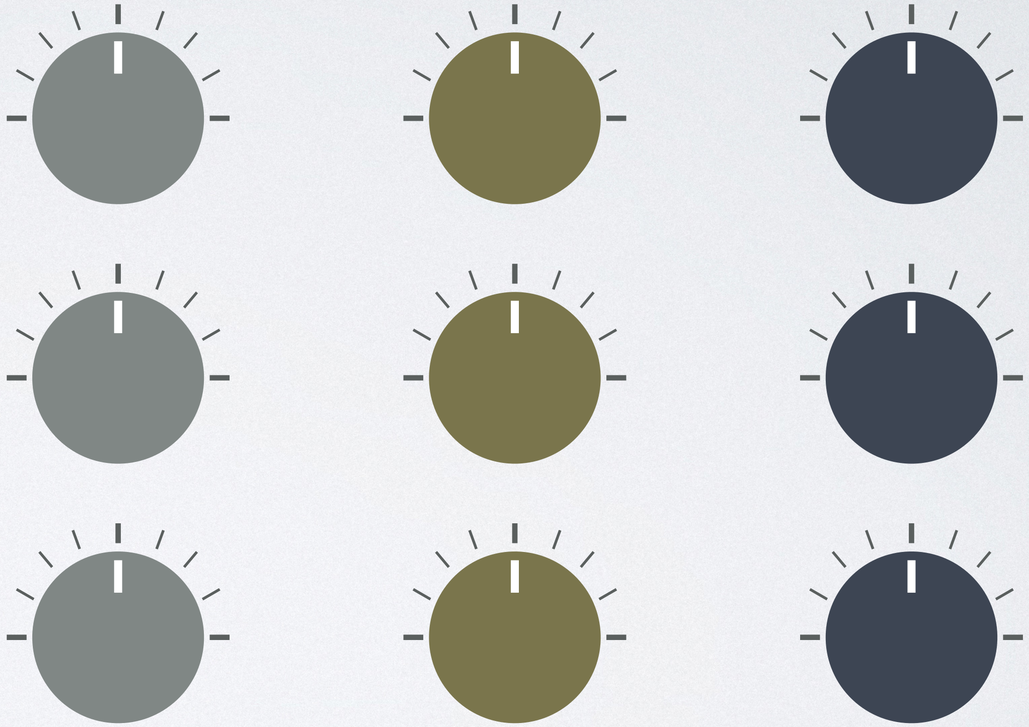
# DESIGN PRINCIPLES

- Affordances
- Constraints
- **Feedback**
- Consistency
- Metaphors
- Mappings
- Visibility

*Adapted from UW CSE440*

# DESIGN PRINCIPLES

- Affordances
- Constraints
- Feedback
- **Consistency**
- Metaphors
- Mappings
- Visibility



*Adapted from UW CSE440*

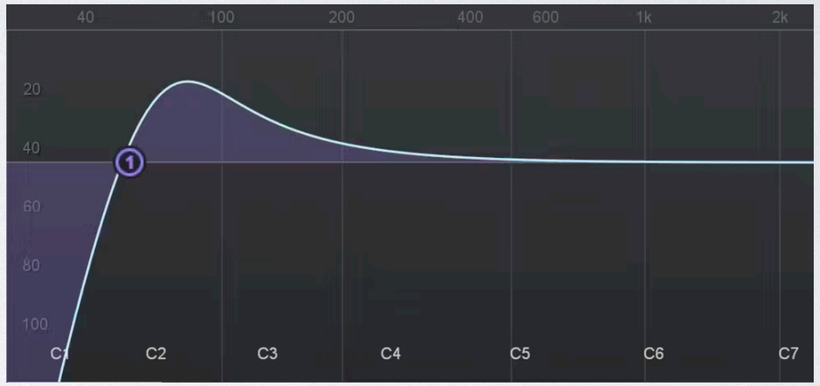
# DESIGN PRINCIPLES

- Affordances
- Constraints
- Feedback
- Consistency
- **Metaphors**
- Mappings
- Visibility

*Adapted from UW CSE440*

# DESIGN PRINCIPLES

- Affordances
- Constraints
- Feedback
- Consistency
- Metaphors
- **Mappings**
- Visibility



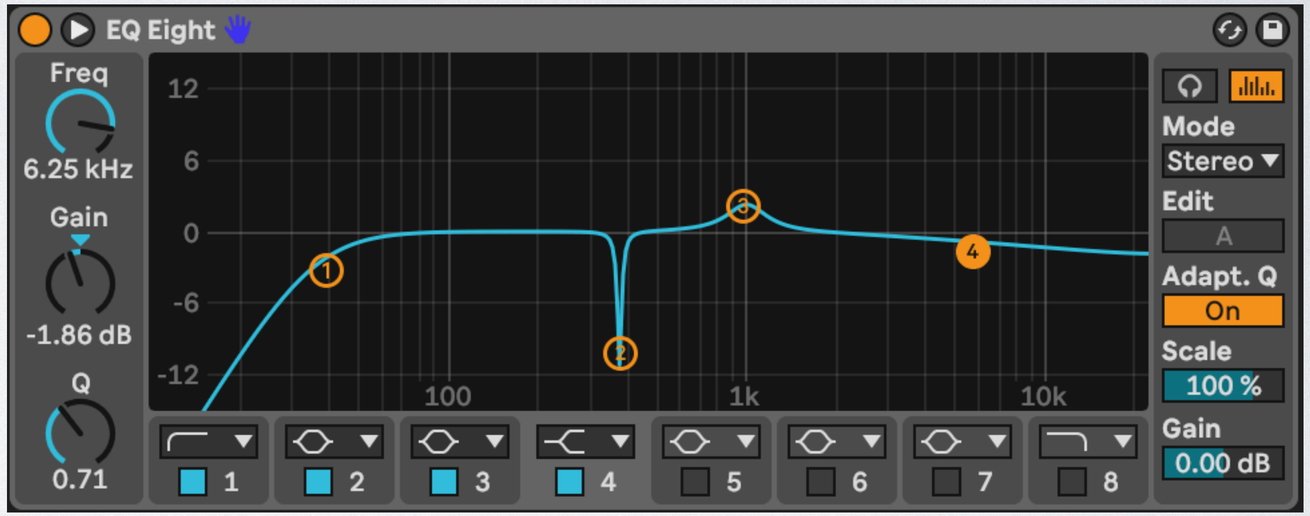

```
void FrequalizerAudioProcessor::createFrequencyPlot (juce::Path& p, const std::vector<double>& mags,
{
    p.startNewSubPath (float (bounds.getX()), mags [0] > 0 ? float (bounds.getCentreY() - pixelsPerDo
    const auto xFactor = static_cast<double> (bounds.getWidth()) / frequencies.size();
    for (size_t i=1; i < frequencies.size(); ++i)
    {
        p.lineTo (float (bounds.getX() + i * xFactor),
                  float (mags [i] > 0 ? bounds.getCentreY() - pixelsPerDouble * std::log (mags [i]) /
    }
}

void FrequalizerAudioProcessor::createAnalyserPlot (juce::Path& p, const juce::Rectangle<int> bounds,
{
    if (input)
        inputAnalyser.createPath (p, bounds.toFloat(), minFreq);
    else
        outputAnalyser.createPath (p, bounds.toFloat(), minFreq);
}
```

*Adapted from UW CSE440*

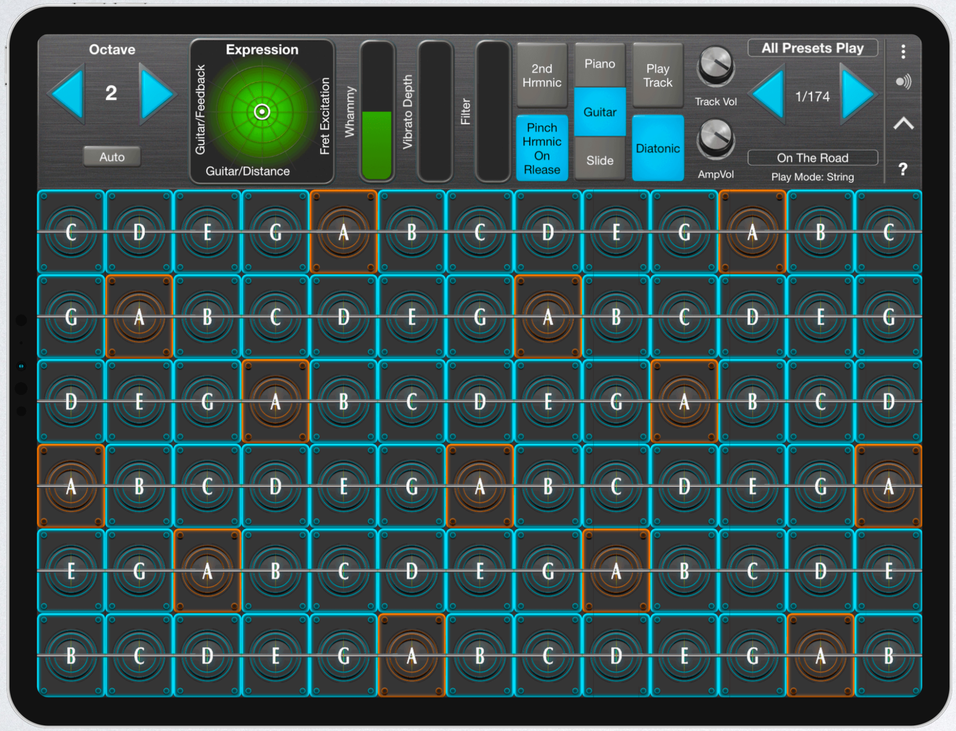
# DESIGN PRINCIPLES

- Affordances
- Constraints
- Feedback
- Consistency
- Metaphors
- Mapping
- **Visibility**



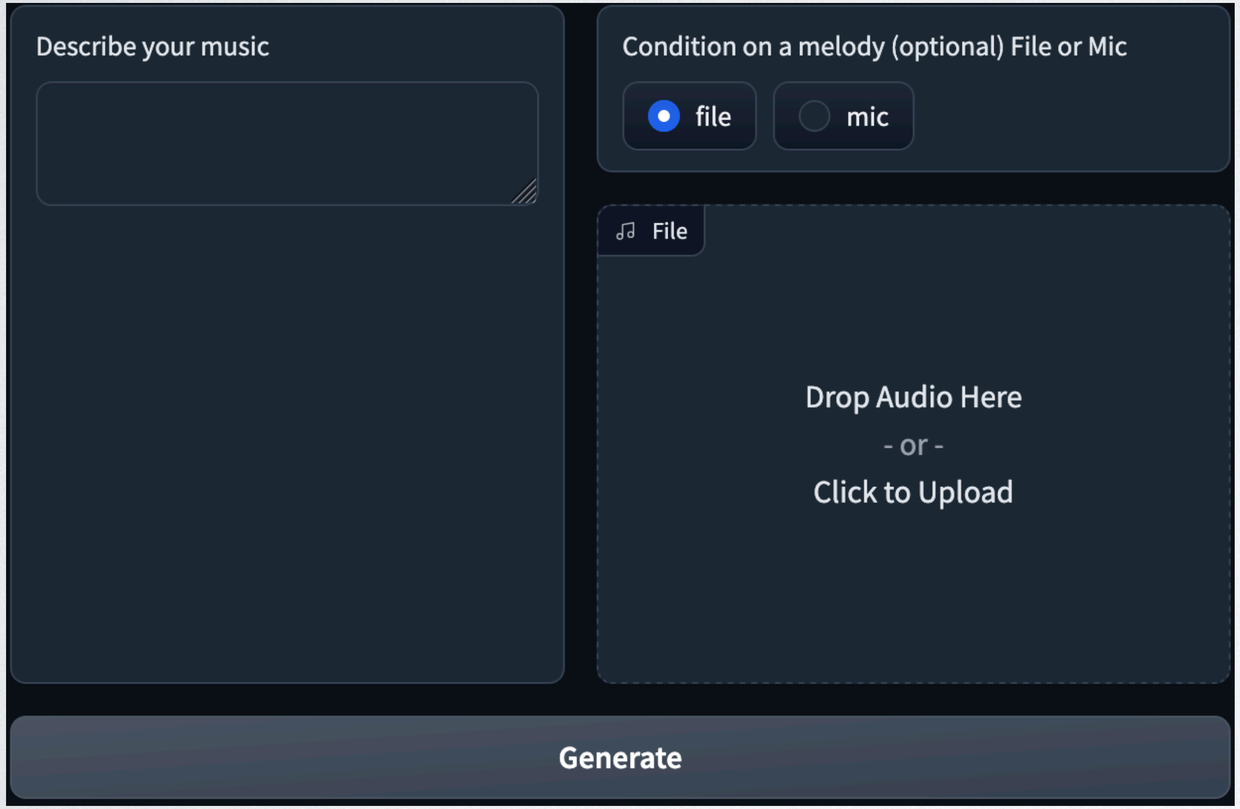
*Adapted from UW CSE440*

# DESIGN IN MUSIC SOFTWARE




GeoShred

# INTERACTION & DESIGN IN AI MUSIC




"Writing about music is like dancing about architecture."

Using concepts that are traditionally used to describe existing music:

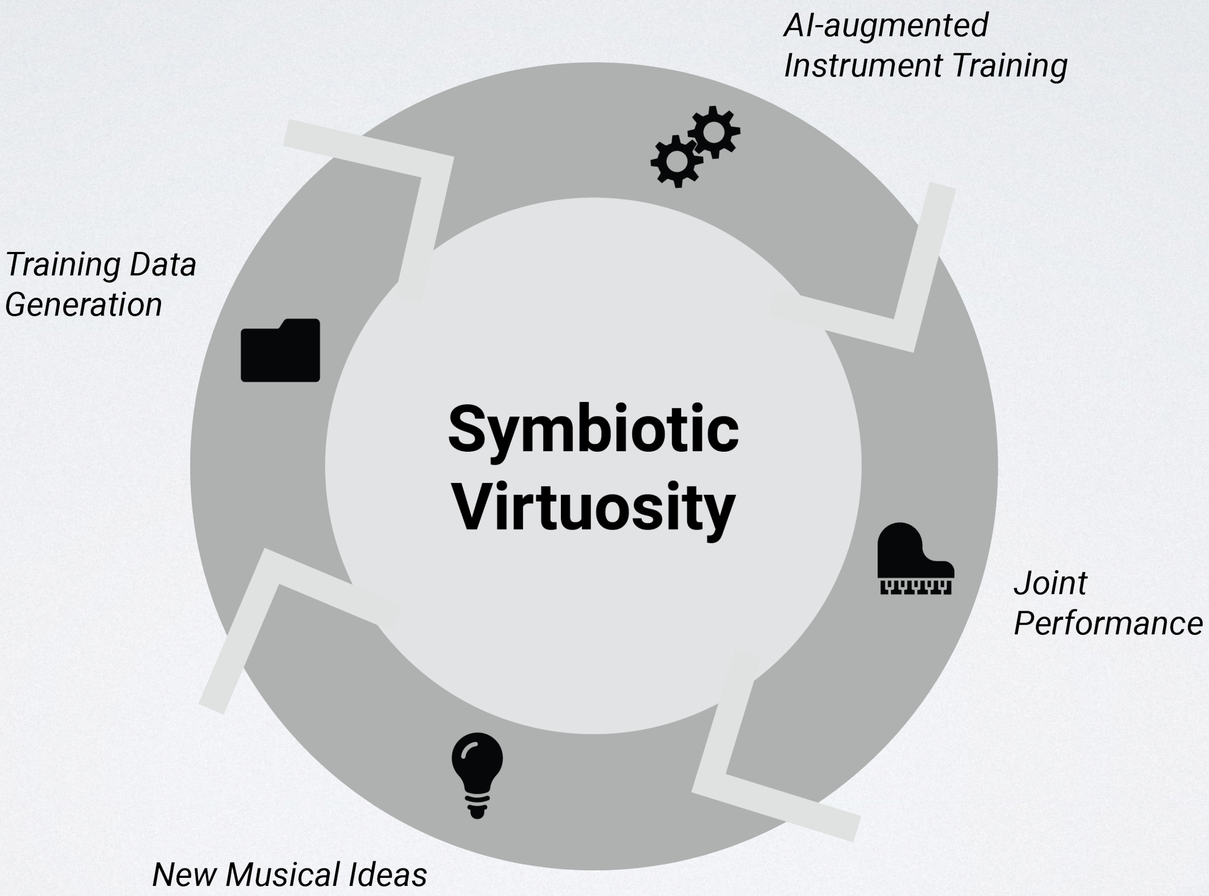
Genre? Tempo? *Emotion?*

# COLLABORATIONS BETWEEN ARTISTS AND ENGINEERS

ARTS
CENTER FOR
ART, SCIENCE &
TECHNOLOGY
AT MIT

CCRMA
Center for Computer Research in Music
and Acoustics - Stanford University

# SYMBIOTIC VIRTUOSITY

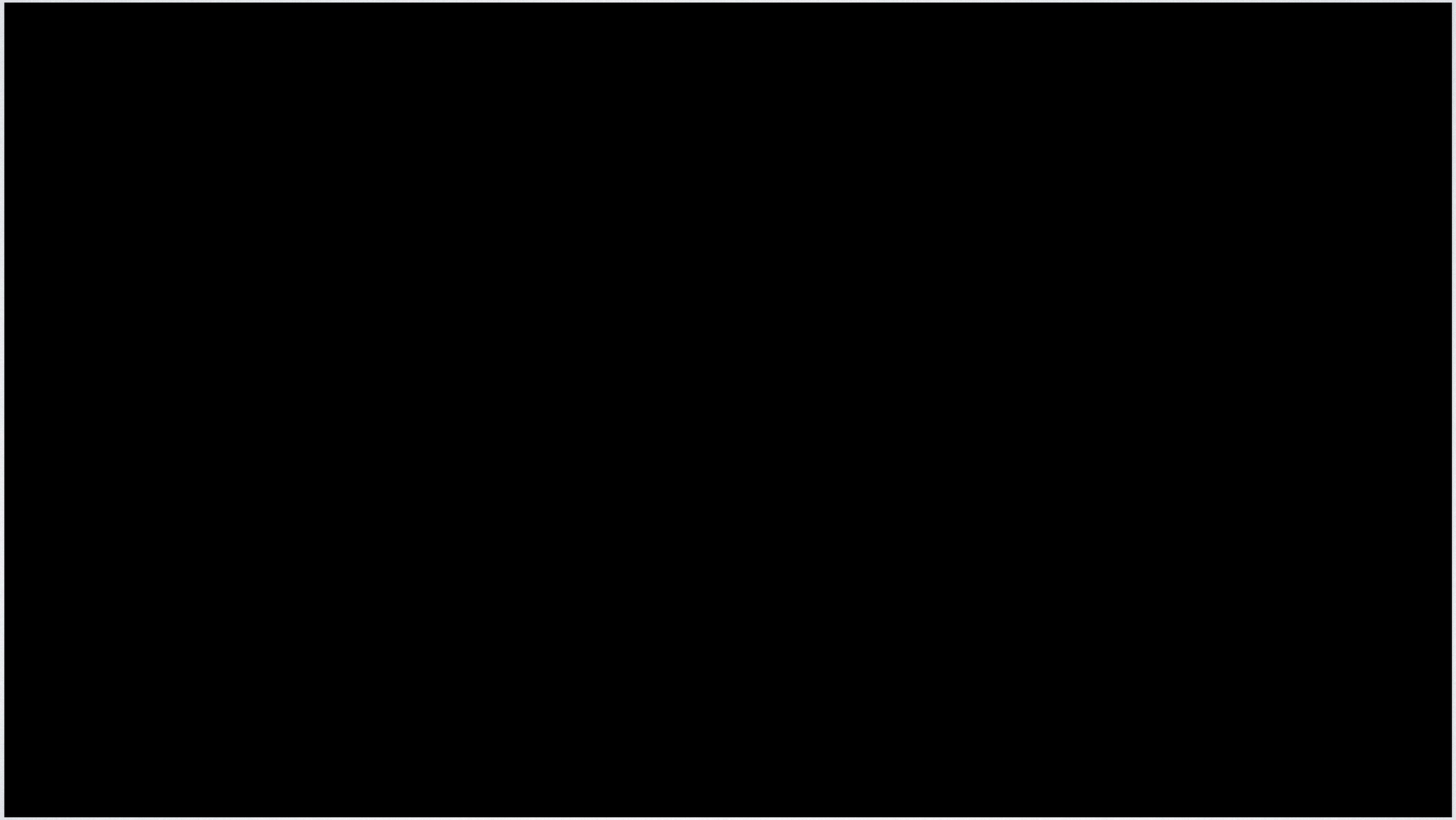


# THE GENESIS OF THE *JAM_BOT*

**Set of requirements:**

- Controllability
- Improvisation
- Capability to generate multiple instruments
- Capability to generate in multiple styles

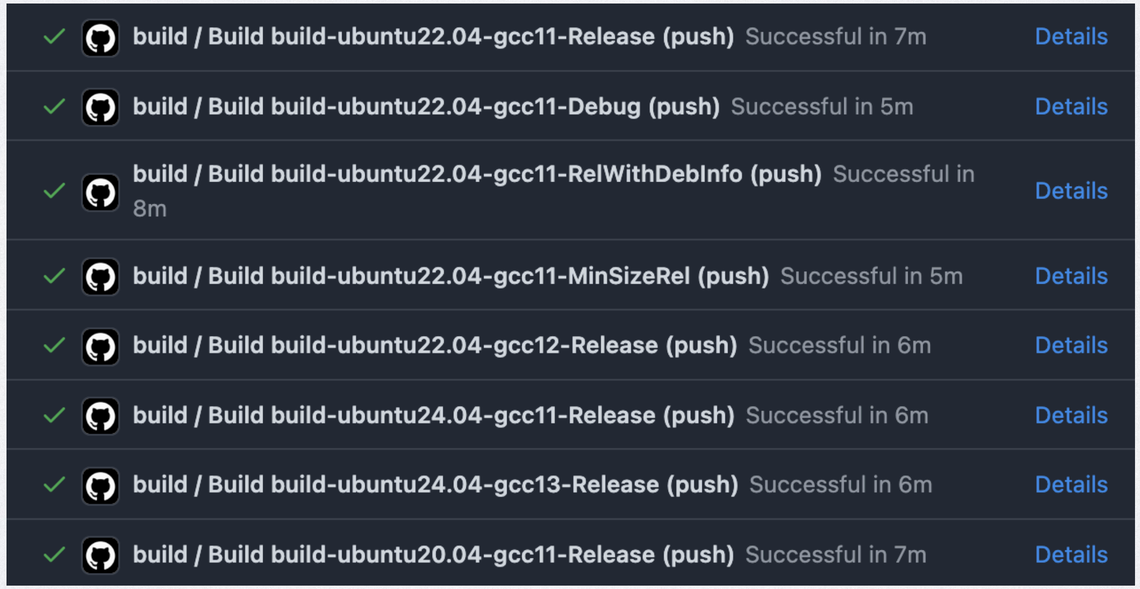
# A QUICK SNIPPET



# COLLECTING ARTIST FEEDBACK

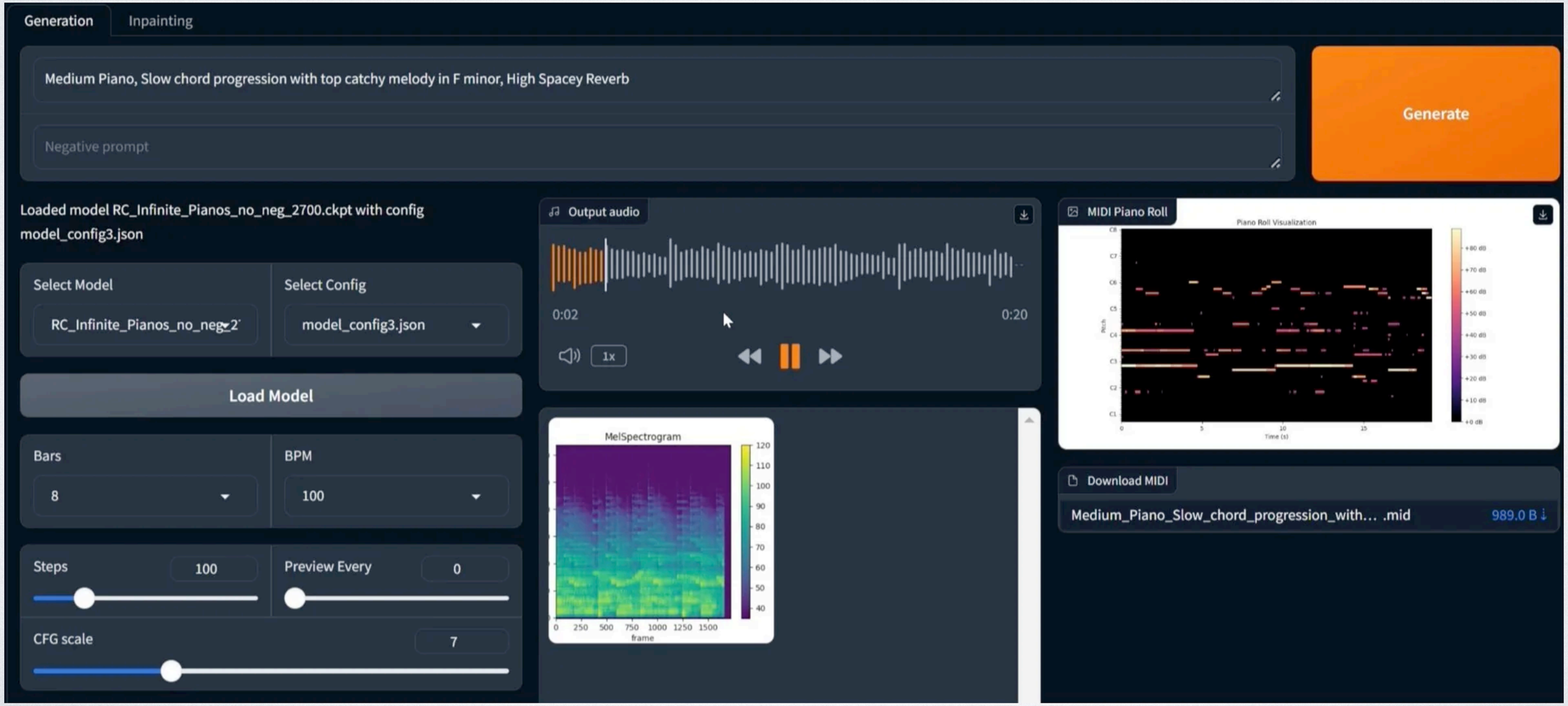
# CONTINUOUS DEPLOYMENT

- Replicable deployments with *Docker and GitHub Actions*
- *Creating artefacts with easily deployable models/binaries (e.g. TorchScript)*



# ITERATIVE DESIGN WITH *GRADIO*

- Enables easy prototyping of interfaces for any Python code



# DISCUSSION