Prototyping ideas and game design

An overview about tools and techniques, and why good game mechanics matter.

Linnéuniversitetet

4ME308-VT21

Mobile Games and Entertainment

Today

- Why do good game mechanics matter?
- A brief game analysis: INTO THE BREACH
- Let's do a game jam!
- Prototyping (Game-) Design
- Overview about tools and techniques for game prototyping *(focus on free/open source)*
    - Paper prototyping, mockups and conceptual walkthroughs
    - Game engines
    - Creative coding frameworks

Why do good game mechanics matter?

Halo: Combat Evolved

"In Halo 1, there was maybe 30 seconds of fun that happened over and over and over and over again. And so, if you can get 30 seconds of fun, you can pretty much stretch that out to be an entire game."

Half-Minute Halo: An Interview with Jaime Griesemer

A brief game analysis: INTO THE BREACH

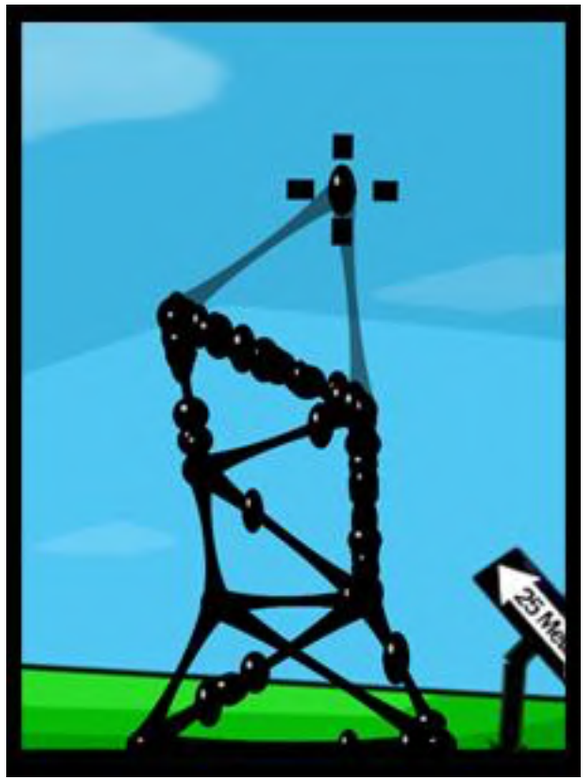
turn-based – strategy vs. puzzle – "telegraphed attacks" (i.e., all information are displayed, no surprises)

Website – Trailer – GDC Talk: Into the Breach Design Postmortem

Let’s do a game jam!

[Indie Train Jam]

[Nordic Game Jam]



[How to Prototype a Game in Under 7 Days]

Prototyping (Game-) Design

Wheel lifecycle for UX engineering

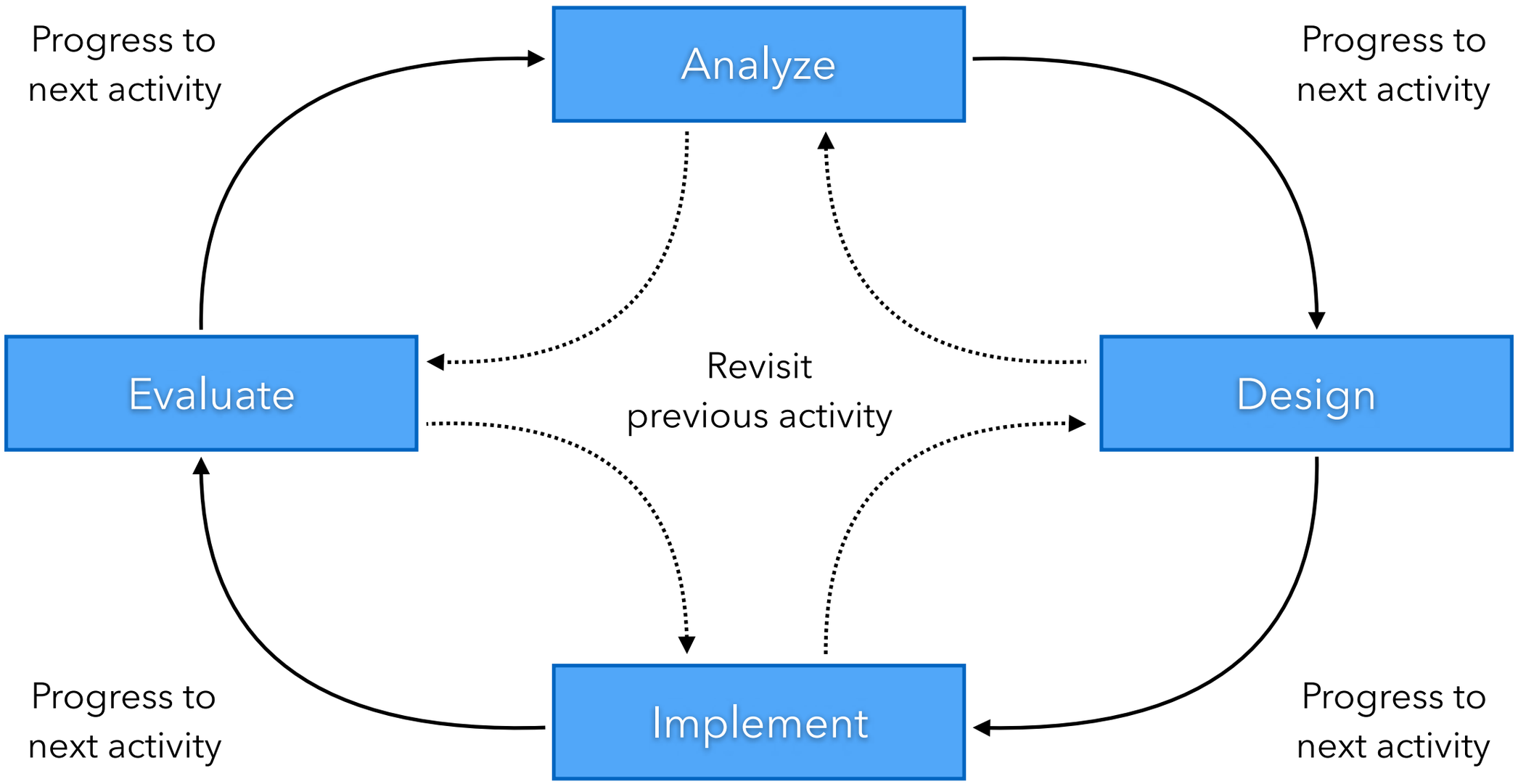


Illustration of the wheel lifecycle process and its opportunities for iteration (via 3D User Interfaces. 2nd edition. pp. 109-110)

Prototype

- early representation of the design built to model, evaluate, and iterate on the design of a product

Chapter 4.4.4: Prototyping the Design
(via 3D User Interfaces. 2nd edition. pp. 118-119)

Benefits

- provide concrete representation of a design that can be communicated to others
- allow "test-driving" and evaluation
- provide visibility
- help in transition from an old system to the new one

Drawbacks

- stakeholder may associate limited functionality with poor design
- stakeholders may assume "magic"

Chapter 4.4.4: Prototyping the Design
(via 3D User Interfaces. 2nd edition. pp. 118-119)

Breadth

- of a prototype concerns how many features are implemented

Depth

- of a prototype represents how much functionality the features provide

Chapter 4.4.4: Prototyping the Design
(via 3D User Interfaces. 2nd edition. pp. 118-119)

Horizontal prototype

- prototype very broad in features but with less depth in functionality
- great for evaluating how users will navigate a design

Vertical prototype

- prototype contains as much depth of functionality as possible for one feature
- beneficial for exploring the design of a particular feature in detail

Chapter 4.4.4: Prototyping the Design
(via 3D User Interfaces. 2nd edition. pp. 118-119)

T prototype

- realizes much of the design at a shallow level but covers one or a few features in depth

Local prototype

- limited in breadth and depth that is focused on a particular isolated feature of the design
- used to evaluate design alternatives for specific portions of the UI

Chapter 4.4.4: Prototyping the Design
(via 3D User Interfaces. 2nd edition. pp. 118-119)

Prototype fidelity

- how completely and closely a prototype represents the intended design

Chapter 4.4.4: Prototyping the Design
(via 3D User Interfaces. 2nd edition. pp. 118-119)

low-fidelity
- impressions of the intended design with little to no functionality

medium-fidelity
- look and feel of the indented design with rudimentary functionality

high-fidelity
- closely resembles the final product
- aesthetics nearly identical to the final product
- should have most, if not all, features fleshed out with full functionality

Chapter 4.4.4: Prototyping the Design
(via 3D User Interfaces. 2nd edition. pp. 118-119)

Prototype interactivity

- degree to which interactions are realized

4 common levels of interactivity

- animated prototypes
- scripted prototypes
- fully programmed prototype
- Wizard of Oz prototype

Wizard of Oz prototype

- deceptively high levels of interactivity with little functionality actually implemented
- a hidden UX team member observes user's actions and then causes the interface to respond appropriately

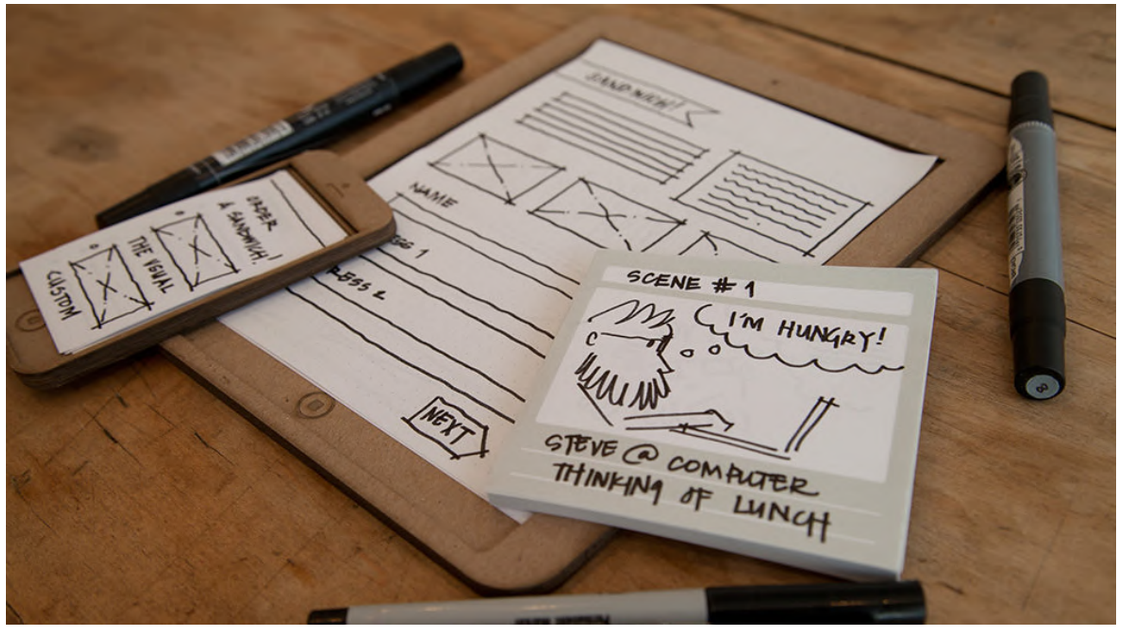
Chapter 4.4.4: Prototyping the Design
(via 3D User Interfaces. 2nd edition. pp. 118-119)

Tools and techniques:

Paper prototyping, mockups and conceptual walkthroughs

Paper prototyping

- think creative / imaginative
- feel close to what you are creating
- easy to change / rearrange things
- often a collaborative / cooperative process with multiple participants
- positive feedback loop by building upon each other's ideas



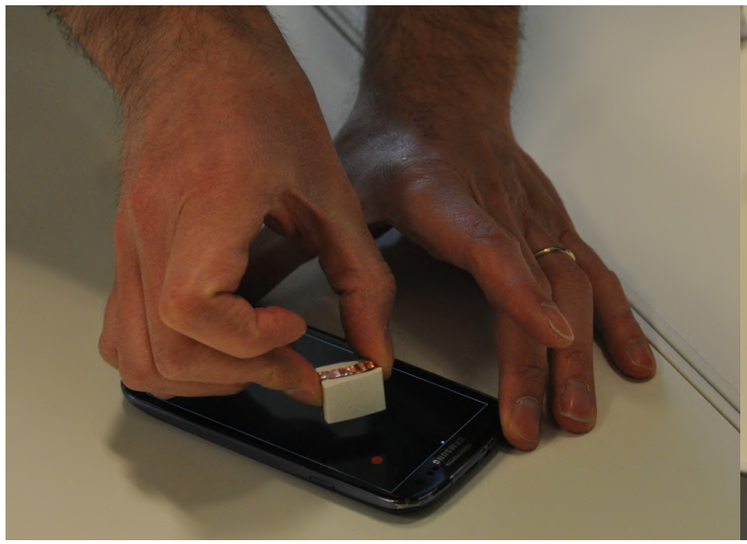
The Skeptic's Guide To Low-Fidelity Prototyping

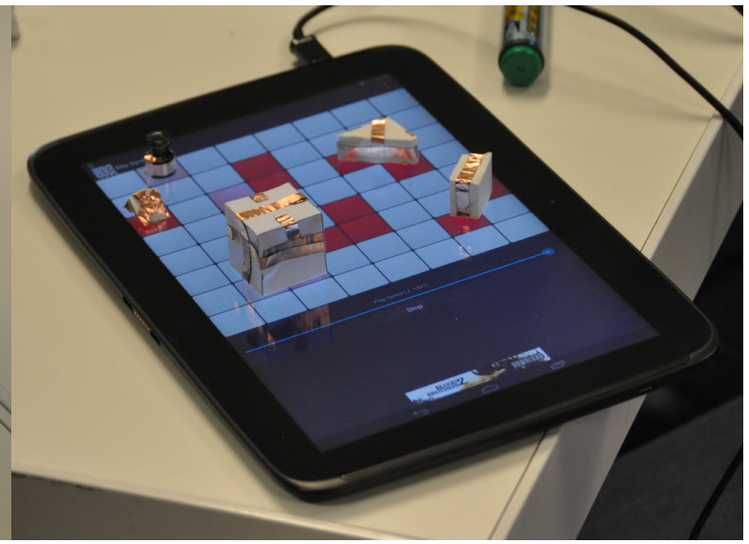
Basic prototyping tools:
- pens, paper (white and colored), cardboard boxes, scissors, sticky notes, tape, glue

Advanced prototyping tools:
- conductive ink / tape, 3D printer, clear/transparent film

Utilising existing artefacts:
- board games (figures, dices, items), card games

Paper prototyping





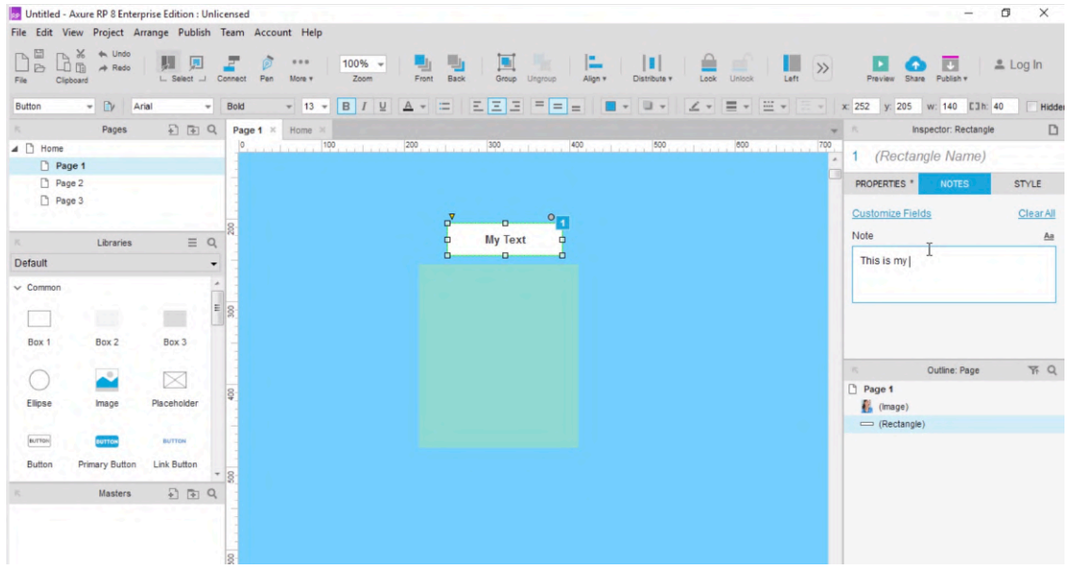
[Prototyping TUI: Conductive tape/ink.]

[Business orignami.]

[A Browser Made For A Gaming Console.]

[Game Design Prototype Creation.]

Axure



Getting started with Axure RP.

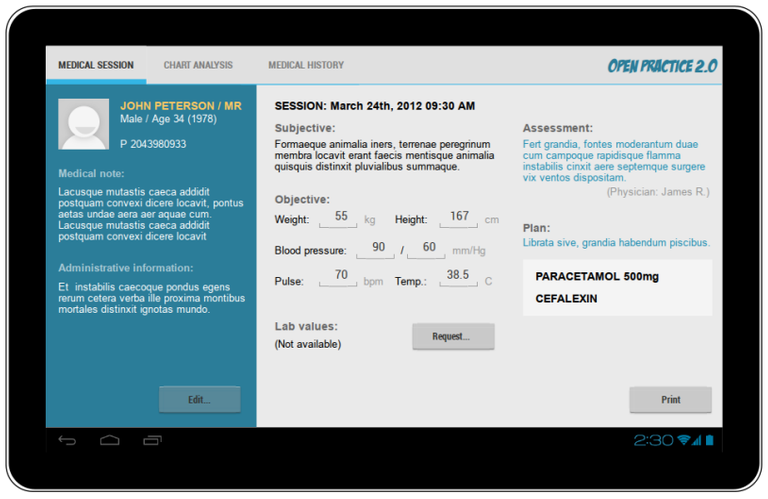
Infobox

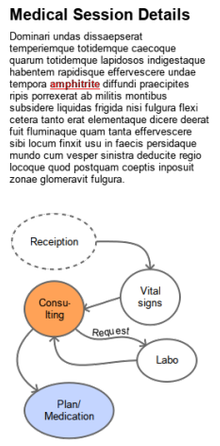
Name:	Axure
URL:	axure.com
Docs:	axure.com/learn
Language:	none (visual programming)
Level:	beginner
Platforms:	Windows, OS X
Availability:	free for students (edu licence)

What is Axure?

- prototyping websites and app interfaces without coding
- great tool for creating interface design mockups
- advanced features: not just static, but also interactive (design logic and links, let your mockup appear as it might be a real product)
- the (interactive) mockup can be exported in HTML format, which makes it easy to deploy over the Internet and accessible to others

Pencil





Easy GUI prototyping with Pencil.

Infobox

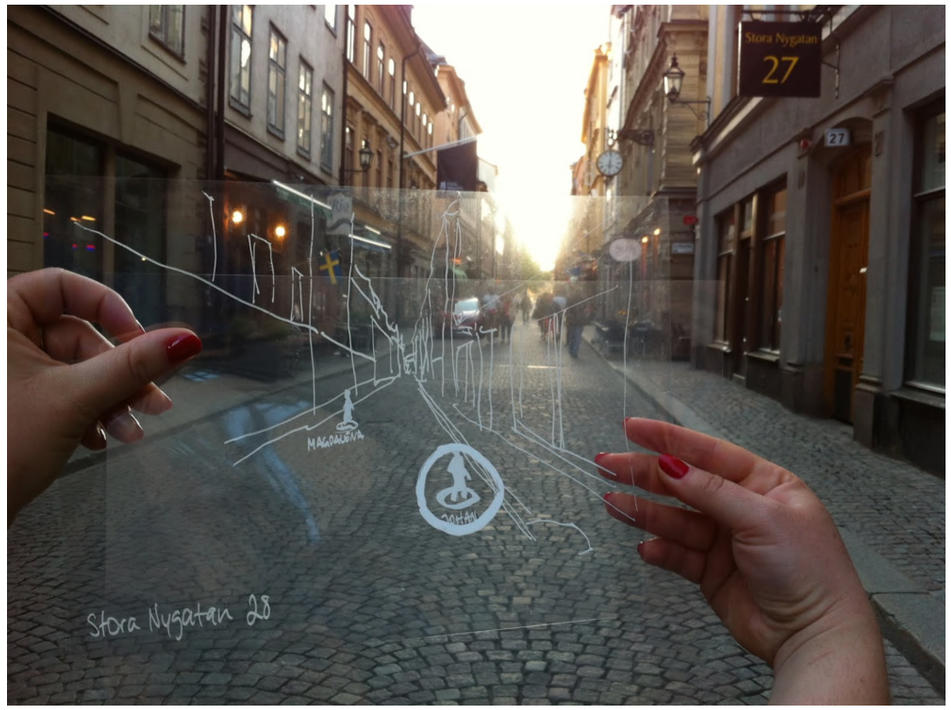
Name:	Pencil
URL:	pencil.evolus.vn
Docs:	pencil.evolus.vn/WikiIndex.html
Language:	none (visual programming)
Level:	beginner
Platforms:	Windows, OS X, Linux
Availability:	open source

What is Pencil?

- prototyping Graphical User Interface (GUI) mockups for desktop and mobile applications
- provides a rich built-in shape collection, plus OpenClipart integration
- multiple pages in the mockup can be linked
- export to HTML, as well as PNG, PDF, SVG, ODT
- bonus: diagram drawing support

Conceptual Walkthrough

- also refered to as *cognitive walkthrough*
- prepare material, that you use to present to and walk participants through the concept of your game design
- material can be sketches, paper prototypes, video, audio, …
- walkthrough should represent a “typical” session based on your game design idea
- consider interaction and choices:
  structure your walkthrough in a way that the participant can decide between multiple options in certain situations
- ask questions / interviews = immediate feedback



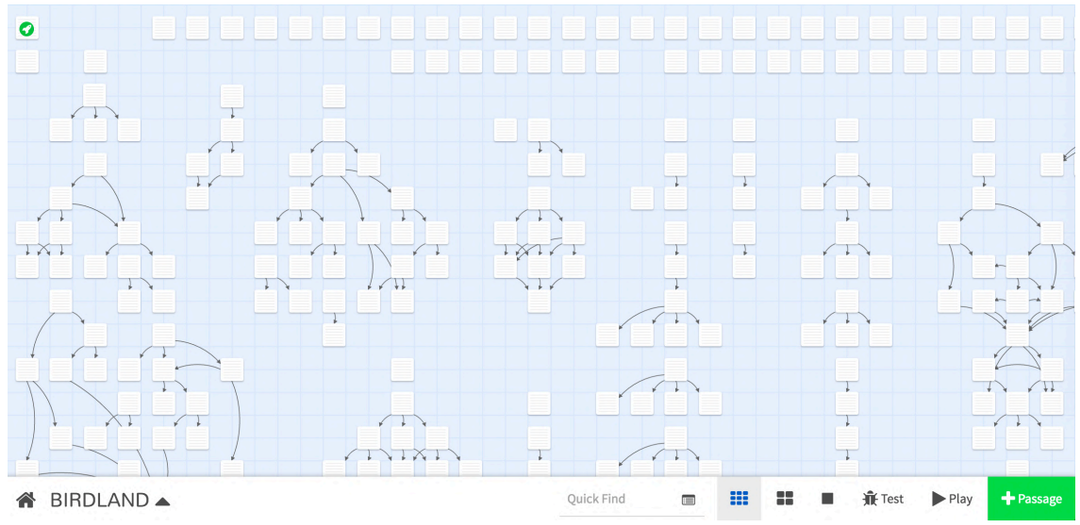
History Explorer (4ME108-VT14)

Chase 'n' Race (4ME108-VT14)

Tools and techniques:

Game engines

Twine



Editing a story in Twine 2.0.

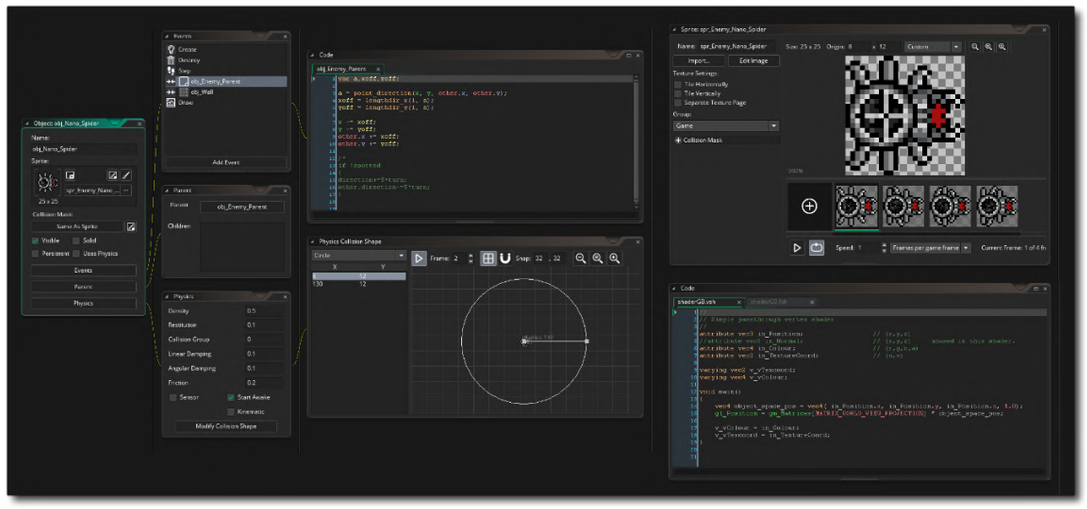
Infobox

Name:	Twine
URL:	twinery.org
Docs:	twinery.org/wiki/
Language:	none (visual programming)
Level:	beginner
Platforms:	Windows, OS X, Linux, web
Availability:	open source

What is Twine?

- tool for non-linear, interactive story telling
- stories are made out of *Passages*
- Passages are linked by using special syntax directly in the text (similar to e.g. Markdown or HTML)
- more features: images, styling through CSS, variables, expressions
- a story can be exported in HTML format, which makes it easy to deploy over the Internet and accessible to others
- possibility to integrate with other web technologies (e.g., p5.js, D3.js, …)

GameMaker Studio 2



GameMaker Studio 2 Quick Start.

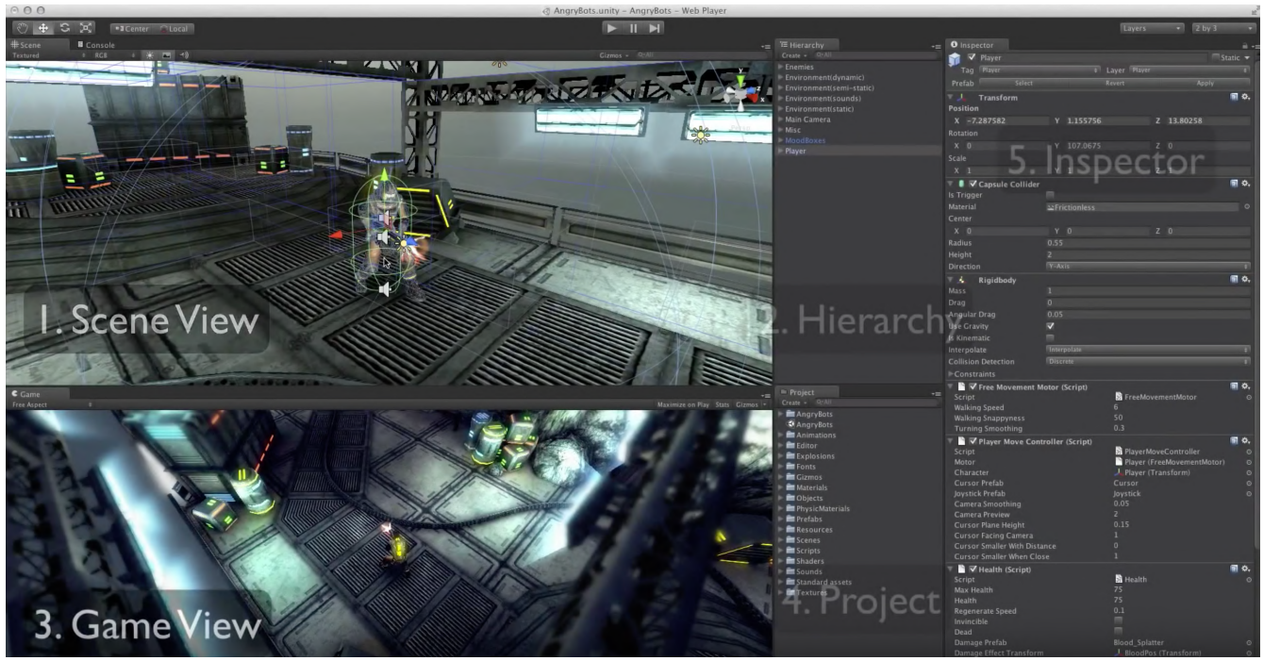
Infobox

Name:	GameMaker: Studio
URL:	yoyogames.com/gamemaker
Docs:	docs2.yoyogames.com
Language:	visual programming + GML
Level:	intermediate
Platforms:	Windows, OS X
Availability:	trial (Licence details)

What is GameMaker: Studio?

- very visual (GUI) game creation approach, with setting parameters
- advanced: create own scripts based on GameMaker Language (GML)
- levels consist of *rooms* and *objects* placed in these rooms
- primarily meant for making 2D and isometric games,
  although some 3D functionalities are available
- paid licences available for additional features, such as export to Windows, OS X, Linux, iOS, Android, HTML5 and more

Unity



Infobox

Name:	Unity
URL:	unity.com
Docs:	docs.unity3d.com
Language:	C#, JavaScript
Level:	intermediate
Platforms:	Windows, OSX
Availability:	free (personal edition)

Unity Learn platform.

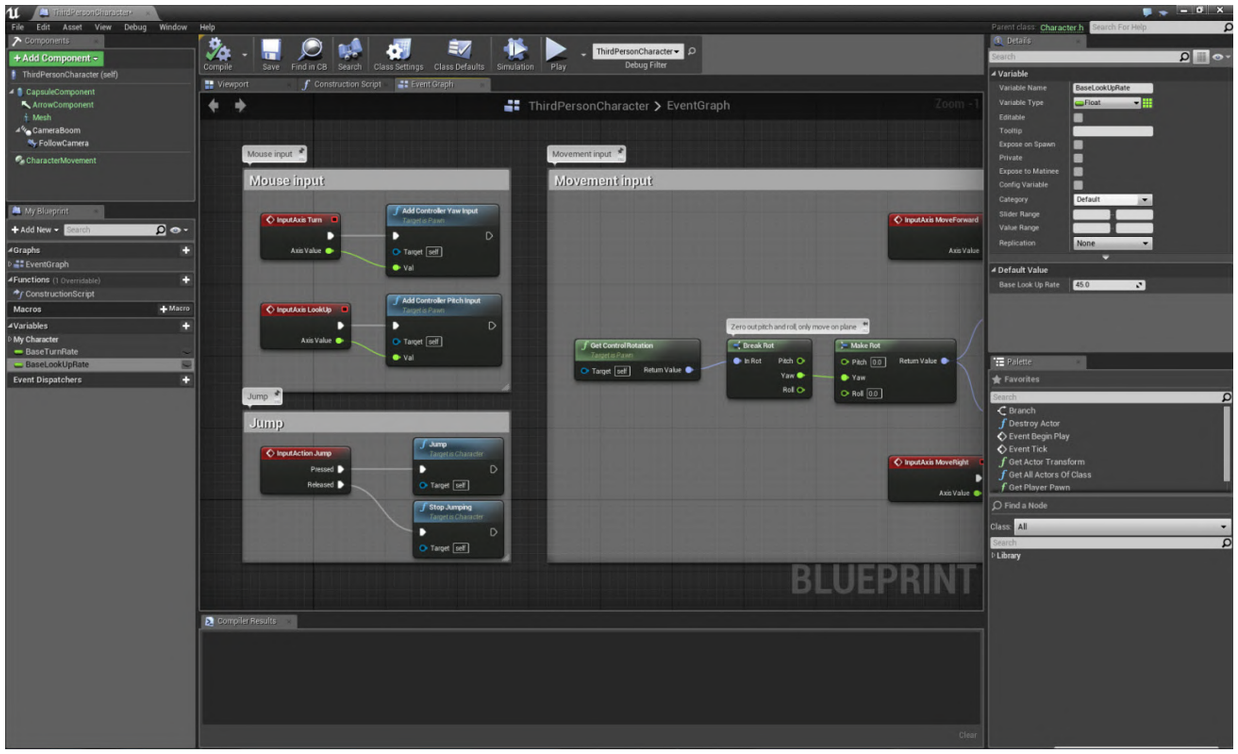
What is Unity?

- powerful game engine, featuring a mixture of Graphical User Interface (GUI) and coding scripts for game development
- optional: user is knowledgable about basic programming concepts
- levels are structured in *Scenes*, containing *GameObjects* with various *Components* attached, such as *Scripts*
- cross-platform game export to Windows, OS X, Linux, iOS, Android, WebGL and more

Unreal Engine

Infobox

Name:	Unreal Engine
URL:	unrealengine.com
Docs:	docs.unrealengine.com
Language:	C++, Visual Scripting
Level:	intermediate
Platforms:	Windows, OS X
Availability:	free

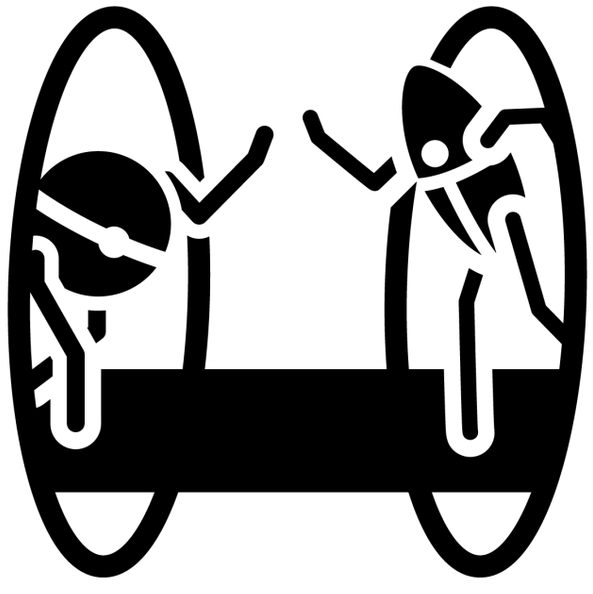


Introduction to Blueprints.

What is Unity?

- powerful game engine, featuring a mixture of Graphical User Interface (GUI) and coding scripts for game development
- special Blueprint Visual Scripting enables the user to rapidly prototype and build games, simulations and visualizations, no programming required
- cross-platform game export to Windows, OS X, Linux, iOS, Android, WebGL, PlayStation, Xbox and more

Tools and techniques:
Creative coding frameworks



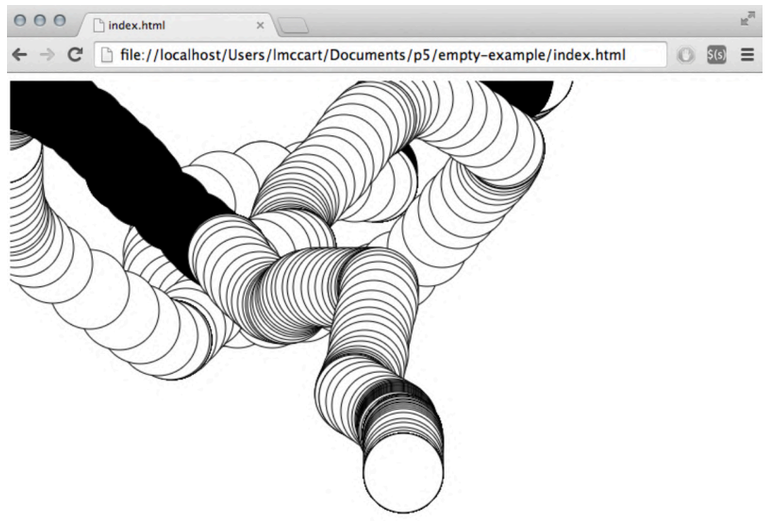
p5.js + p5.play

Infobox

Name:	p5.js
URL:	p5js.org
Docs:	p5js.org/reference/ and p5play.molleindustria.org
Language:	JavaScript
Level:	beginner
Platforms:	web
Availability:	open source

```
function setup() {
  createCanvas(640, 480);
}

function draw() {
  if (mouseIsPressed) {
    fill(0);
  } else {
    fill(255);
  }
  ellipse(mouseX, mouseY, 80, 80);
}
```



Get started with p5.js.

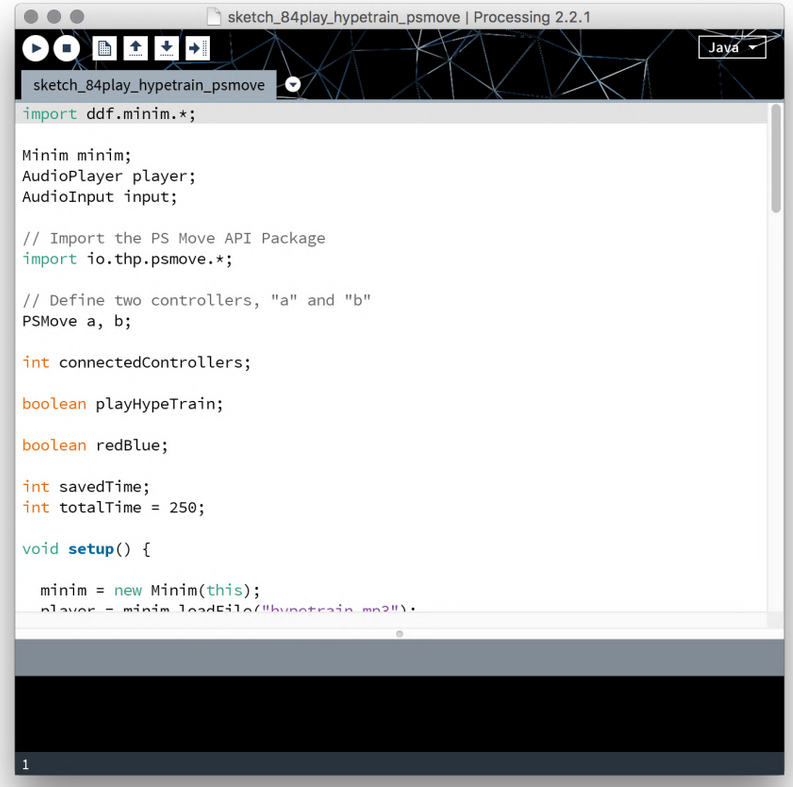
What is p5.js?

- creative coding toolkit originated from the Processing project
- easy and quick to program interactive visual output
- can be mixed with other JavaScript libraries / web technologies
- extending library p5.play further supports the creation of game-like sketches
- sketches are deployed in HTML format over the web, which makes it easy to deploy and accessible to others

Processing

Infobox

Name:	Processing
URL:	processing.org
Docs:	processing.org/reference/
Language:	Java
Level:	beginner
Platforms:	Windows, OSX, Linux
Availability:	open source



Processing: Examples.

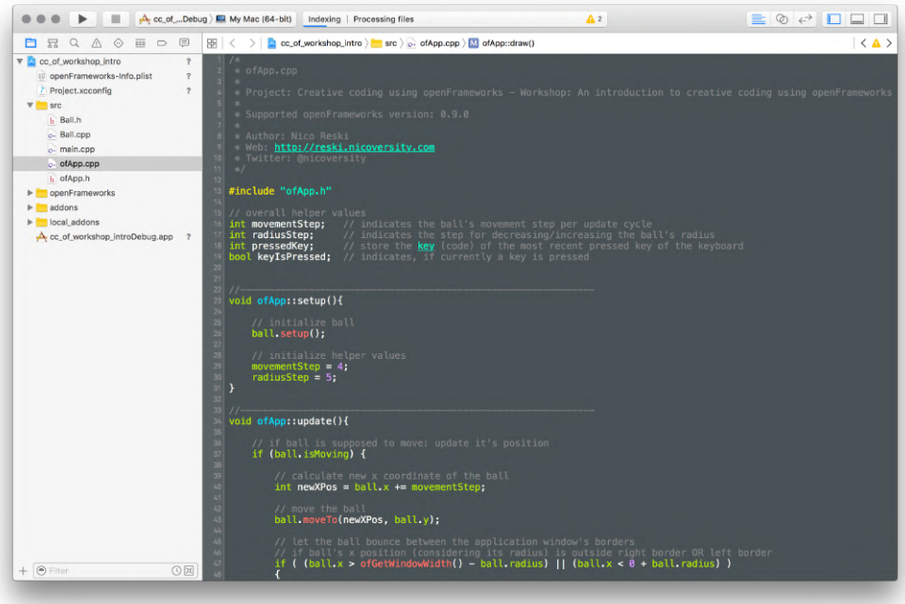
What is Processing?

- Java based software sketchbook with the aim to make learning how to code / program easy and accessible
- very good at programming visual and audio output
- capable of both 2D and 3D output (OpenGL)
- strong community, lots of additional libraries and plugins
- some functionalities to work with additional hardware

openFrameworks

Infobox

Name:	openFrameworks
URL:	openframeworks.cc
Docs:	openframeworks.cc/documentation/
Language:	C++
Level:	intermediate
Platforms:	Windows, OSX, Linux, iOS, Android
Availability:	open source



Editing OF code in Xcode (OS X).

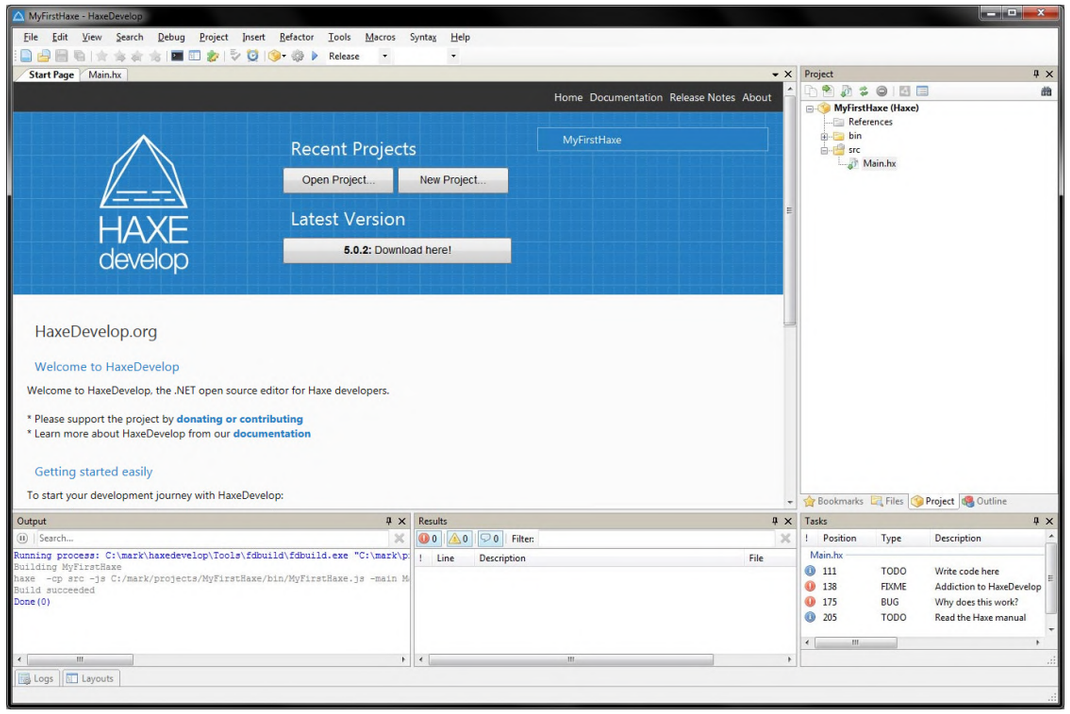
What is openFrameworks?

- creative coding toolkit to provide a simple and intuitive framework for experimentation
- great at linking / combining lots of different technologies and libraries (graphics, audio, video, computer vision, 3D)
- generally good at drawing and producing visual output
- strong community with lots of add ons (ofxAddons)

Haxe

Infobox

Name:	Haxe
URL:	haxe.org
Docs:	api.haxe.org
Language:	Haxe programming language
Level:	intermediate
Platforms:	Windows, OSX, Linux
Availability:	open source



HaxeDevelop IDE.

What is Haxe?

- toolkit to build cross-platform tools and frameworks
- target your tools to iOS, Android, HTML5, Node.js, Python, Java and more
- great at accessing native features of different platforms
- one code basis, while deploying to lots of different platforms
- can be used for development of games, web and mobile apps, desktop applications, command line tools or cross-platform APIs

Cinder

Infobox

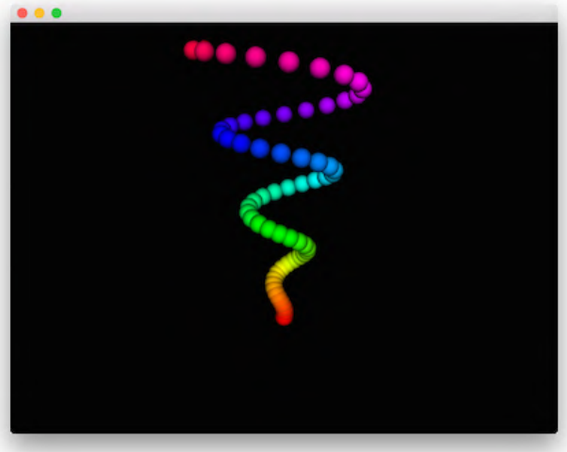
Name:	Cinder
URL:	libcinder.org
Docs:	libcinder.org/docs/
Language:	C++
Level:	advanced
Platforms:	Windows, OSX, iOS
Availability:	open source

```
void BasicApp::draw()
{
    gl::clear();
    gl::enableDepthRead();
    gl::enableDepthWrite();

    CameraPersp cam;
    cam.lookAt( vec3( 5, 2, 5 ), vec3( 0, 1, 0 ) );
    gl::setMatrices( cam );

    auto lambert = gl::ShaderDef().lambert().color();
    auto shader = gl::getStockShader( lambert );
    shader->bind();

    int numSpheres = 64;
    float maxAngle = M_PI * 7;
```



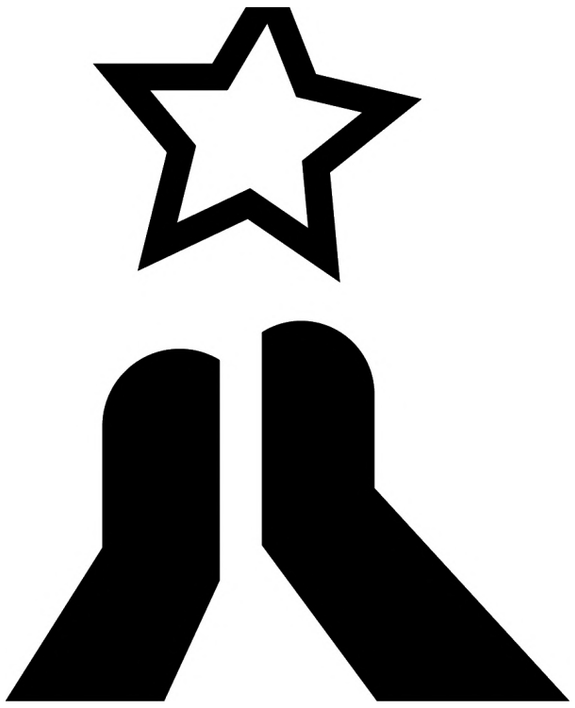
Cinder guide: The Third Dimension.

What is Cinder?

- open source library for professional-quality creative coding in C++
- capable of 2D, but strong focus on 3D with OpenGL
- low-level hardware / sensor access
- advanced library, best be used with existing knowledge about the concepts of computer graphics and OpenGL
- powerful once mastered

4ME308-VT21
Prototyping ideas and game design

Congratulations!



- You learned why good game design matters.
- You learned about characteristics and aspects of prototypes.
- You are aware of different tools and techniques to start prototyping / developing games.

- Game Maker's Toolkit (GMTK)
  by Mark Brown (via YouTube)
- GDC Vault
  by Game Developers Conference (via YouTube)
- The Coding Train
  by Daniel Shiffman (via YouTube)
- Creative coding using openFrameworks - Workshop: An introduction to creative coding using openFrameworks
  by Nico Reski
- Creative coding using p5.js - Workshop: An introduction to creative coding using p5.js
  by Nico Reski
- Creative coding using Processing - Workshop: An introduction to creative coding using Processing.
  by Nico Reski (via GitHub; Processing language intro)
- Augmented Reality using Unity and Vuforia (Part 1 / Part 2)
  by Nico Reski (via GitHub)

Contact

Nico Reski

reski.nicoversity.com

@nicoversity

github.com/nicoversity

nico.reski@lnu.se

(PGP Key ID: B061D75B,
PGP Fingerprint: E826 C9FF 1701 0BAC
CA98 308C 6772 4499 B061 D75B)

Office: HUS D 2269 A

VRxAR Labs

Department of Computer Science
and Media Technology (CM)

Faculty of Technology

Linnaeus University, Växjö

Portal icons in the presentation available via

bit.ly/portaliconpack