E.L.D (Educational Level Designer)

Game Design Document

Contents

[1.0 Overview 2](#_Toc445209280)

[1.1 Key Features 2](#_Toc445209281)

[1.2 Project Points 2](#_Toc445209282)

[2.0 UI Flow and Functionality 2](#_Toc445209283)

[2.1 Opening Logos 2](#_Toc445209284)

[2.2 Front-End Menus 2](#_Toc445209285)

[2.2.1 Main Menu 2](#_Toc445209286)

[2.3 In-Game Menus 2](#_Toc445209287)

[2.3.1 Pause Menu 3](#_Toc445209288)

[2.4 Build Mode UI 3](#_Toc445209289)

[2.5 Connect Mode UI 3](#_Toc445209290)

[2.6 Play Mode UI 3](#_Toc445209291)

[3.0 General Game Structure 3](#_Toc445209292)

[3.1 Build Mode 3](#_Toc445209293)

[3.2 Connect Mode 3](#_Toc445209294)

[3.3 Play Mode 3](#_Toc445209295)

[3.4 Share and Load Level 3](#_Toc445209296)

[4.0 Game Mode Descriptions 3](#_Toc445209297)

[4.1 Build Mode 3](#_Toc445209298)

[4.1.1 Functionality 3](#_Toc445209299)

[4.1.2 Objects 4](#_Toc445209300)

[4.1.2.1 Non Interactable Objects 4](#_Toc445209301)

[4.1.2.2 Interactable Objects 4](#_Toc445209302)

[4.1.3 Object Skins 4](#_Toc445209303)

[4.2 Connect Mode 4](#_Toc445209304)

[4.2.1 Functionality 4](#_Toc445209305)

[4.2.2 Objects 4](#_Toc445209306)

[4.3 Play Mode 5](#_Toc445209307)

[4.3.1 Functionality 5](#_Toc445209308)

[5.0 General Gameplay Elements 5](#_Toc445209309)

[5.1 Controls 5](#_Toc445209310)

[5.2 Editor Tools 6](#_Toc445209311)

[5.3 Saving and Loading 6](#_Toc445209312)

[6.0 Asset List 6](#_Toc445209313)

[6.1 Objects & Skins 6](#_Toc445209314)

[6.2 UI 6](#_Toc445209315)

[6.2 Animation List 6](#_Toc445209316)

[6.3 VFX 6](#_Toc445209317)

[6.4 Sound Effects 6](#_Toc445209318)

1.0 Overview

E.L.D is a tool focused on allowing users to create their own levels and games using systems taken from simple electrical engineering widgets. Users can also share their levels with friends or publicly online to play or recreate.

* 1. Key Features

1.2 Project Points

* Release Date: 5/13/2016
* Platform: PC
* Audience: 8-18 children and young adults who want to get into learning about level design or enjoy creating and sharing interactive experiences.

2.0 UI Flow and Functionality

2.1 Opening Logos

* When the application is run, the Columbia College Chicago Logo is displayed.
* This is followed by the Michael Pattengale Logo.
* The user can advance through these after a short time.

2.2 Front-End Menus

2.2.1 Main Menu

The Main Menu UI allows the user to select from several options:

* Create New
  + Opens a new Build Scene.
* Load & Edit
  + Opens a Load Select Screen.
* Load & Play
  + A Pop-up appears with a textbox allowing users to enter a play code.
  + Once entered the player can hit a “Play” button to open a Play Scene with the level opened.
  + The play can click outside of the pop-up to close it, or hit a “Return” button.
* Options
  + Opens the Options Scene.

2.3 In-Game Menus

2.3.1 Pause Menu

The pause menu can be activated by pressing “Escape.”

* Build Scene
  + Resume
    - This returns the user to the gameplay where they left off.
  + Quit
    - This ends the game and returns the user to the Main Menu.
    - A confirmation asks the player to confirm that they want to exit, losing all progress.
* Play Scene
  + Resume
    - This returns the user to the gameplay where they left off.
  + Quit
    - This ends the game and returns the user to the Main Menu.
    - A confirmation asks the player to confirm that they want to exit.

2.4 Build Mode UI

Object UI

Always is displayed on the screen, shows the available objects that the player can place in the level.

Object Skins UI

After selecting an object from the Object UI, this slides out from behind the Object UI and displays the options for the different skins available for that object. When a skin is selected, this menu returns to hiding behind the Object UI.

2.5 Connect Mode UI

2.6 Play Mode UI

3.0 General Game Structure

3.1 Build Mode

3.2 Connect Mode

3.3 Play Mode

3.4 Share and Load Level

4.0 Game Mode Descriptions

4.1 Build Mode

4.1.1 Functionality

In this mode, the user builds the level that the player can see and interact with.

4.1.2 Objects

4.1.2.1 Non Interactable Objects

Objects that the player can interact with without pressing the “Interact” Button, or Objects that cannot be interacted with.

**Walls**

Stops the player from moving through them.

**Floor Buttons**

When the player steps on the switch, if it is on, it is set to off, if it is off, it is set to on

**Powered Doors**

Opens when powered, closes when unpowered.

**Pushable Objects**

Objects that will move when the player or other objects collide with them.

**Exposed Wire**

When covered by a pushable object, will block power from passing through the node above it.

**Powered Trap Door**

Closes when powered, opens when unpowered. Kills the player if they come in contact with it while it is open.

4.1.2.2 Interactable Objects

Objects that the player must press the “Interact” Button to use.

**Switches**

Stops the player from moving through them.

When the player interacts with the switch, if it is on, it is set to off, if it is off, it is set to on.

**Buttons**

When the player interacts with the Wall Button, the switch is turned on for a short period of time before it is automatically turned off.

**Doors**

Must be manually opened and closed by players.

4.1.3 Object Skins

Displays the various “Skins” for objects. These use different sprites and are purely aesthetic.

4.2 Connect Mode

4.2.1 Functionality

In this Mode the user creates the logic behind the objects within the Build Mode using power currents similar to an electrical engineer’s breadboard.

4.2.2 Objects

Power Line

Connects Objects in this mode and moves power from Powered Objects to other objects

**Battery**

Powers objects with a set amount of power.

**Connector Switch**

If connected to a switch, powered, and the switch is set to on, this will Power objects, however if it is not connected to a switch, or the switch is not on, it cuts off power.

**Gates**

**AND Gate**

Tests two inputs, if both are powered, power the output.

**OR Gate**

Tests two inputs, if one or both are powered, power the output.

**XOR Gate**

Tests two inputs, if only one is powered, power the output.

**NOR Gate**

Tests two+ inputs, only if none are powered, power the output

**NAND Gate**

Test two+ inputs, only if both are not powered, power the output

**NOT Gate**

Test input(s), it no input(s) are powered, power the output (nor but only needs 1 input)

**XNOR Gate**

Tests two+ inputs, only if all are powered or all are not powered, power the output

**Cross Wire**

Only allows power coming from the north/south to flow north/south and only allows power coming from the east/west to flow east/west

**Clock**

Sends out power at various frequencies set by the user.  
 **Resistor**

Decreases the power by an amount set by the user

**Comparator**

Compares the power from this object’s inputs. If the designated input has a greater input, it outputs power

**Power Delay**

When powered, delays power output for a set amount of time.

4.3 Play Mode

4.3.1 Functionality

Allows the user to test-play the level they are working on.

5.0 General Gameplay Elements

5.1 Controls

Left click to use current tool

Right click to erase

Hold Left to move object mouse is over

Middle click to pan

Scroll to scroll in and out

5.2 Editor Tools

Select

Move

Erase

Help

5.3 Saving and Loading

“S” to Save.

“L” to Load.

6.0 Asset List

6.1 Objects & Skins

6.2 UI

Button Sprites

Play Mode Button

Return Button

Erase Tool Button

Hand Tool Button

Help Tool Button

Line Tool Button

Edit Mode Button

Connect Mode Button

Button Text Font

Tutorial Text Font

Dot Tile

Connection Line Sprite

Object List UI Background

6.2 Animation List

6.3 VFX

6.4 Sound Effects