

ΠΑΝΕΠΙΣΤΗΜΙΟ ΜΑΚΕΔΟΝΙΑΣ
ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
ΤΜΗΜΑΤΟΣ ΕΦΑΡΜΟΣΜΕΝΗΣ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΜΕΘΟΔΟΙ ΚΑΙ ΕΡΓΑΛΕΙΑ ΑΝΑΠΤΥΞΗΣ ΠΑΙΧΝΙΔΙΩΝ HTML5: ΕΠΙΣΚΟΠΗΣΗ ΚΑΙ
ΣΥΓΚΡΙΤΙΚΗ ΑΝΑΛΥΣΗ

Διπλωματική Εργασία
της
Κλεφτογιάννη Αλεξάνδρας

Θεσσαλονίκη, 2018

ΜΕΘΟΔΟΙ ΚΑΙ ΕΡΓΑΛΕΙΑ ΑΝΑΠΤΥΞΗΣ ΠΑΙΧΝΙΔΙΩΝ HTML5: ΕΠΙΣΚΟΠΗΣΗ ΚΑΙ
ΣΥΓΚΡΙΤΙΚΗ ΑΝΑΛΥΣΗ

Κλεφτογιάννη Αλεξάνδρα

Πτυχίο Μηχανικού Πληροφορικής & Επικοινωνιών Τεχνολογικής Εκπαίδευσης, ΤΕΙ Κεντρικής
Μακεδονίας, 2014

Διπλωματική Εργασία

υποβαλλόμενη για τη μερική εκπλήρωση των απαιτήσεων του

ΜΕΤΑΠΤΥΧΙΑΚΟΥ ΤΙΤΛΟΥ ΣΠΟΥΔΩΝ ΣΤΗΝ ΕΦΑΡΜΟΣΜΕΝΗ ΠΛΗΡΟΦΟΡΙΚΗ

Επιβλέπων Καθηγητής
Κασκάλης Θεόδωρος

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την

Κασκάλης Θεόδωρος

Χατζηγεωργίου Αλέξανδρος

Ξυνόγαλος Στυλιανός

.....

.....

.....

Κλεφτογιάννη Αλεξάνδρα

.....

Περίληψη

Η παρούσα διπλωματική εργασία ερευνά και αναλύει τεχνολογίες και πρακτικές για την ανάπτυξη παιχνιδιών HTML5. Πραγματοποιήθηκε σύγκριση και αξιολόγηση των υφιστάμενων μηχανών ανάπτυξης παιχνιδιών καθώς επίσης και ανάλυση των αντίστοιχων frameworks και libraries. Τα πλεονεκτήματα και τα μειονεκτήματα της εκάστοτε μηχανής παιχνιδιών περιγράφονται αναλυτικά και επιχειρήθηκε συγκριτική ανάπτυξη μιας εφαρμογής-αναφοράς.

Μετά την αναζήτηση, ανάγνωση και καταχώρηση των σχετιζόμενων ιστοσελίδων, άρθρων και βιβλίων, ακολουθεί η καταγραφή και η σύγκριση των δυνατοτήτων και των χαρακτηριστικών των τεχνολογιών, των βιβλιοθηκών, των εργαλείων και των frameworks. Για την εξαγωγή και την αξιολόγηση των συμπερασμάτων, αναπτύχθηκε το ίδιο παιχνίδι για κάθε μηχανή, ώστε να διαπιστωθούν οι διαφορές που υπάρχουν μεταξύ των εξεταζόμενων μηχανών.

Λέξεις Κλειδιά: μηχανές παιχνιδιών, HTML5, CSS, JavaScript, framework, libraries

Abstract

This diploma thesis explores and analyzes technologies and practices for the development of HTML5 games. Comparison and evaluation of existing game development machines as well as analysis of corresponding frameworks and libraries were made. The advantages and disadvantages of each game machine are described in detail and a comparative development of a reference application has been attempted.

Following researching, reading and registering related webpages, articles and books, the comparison of the capabilities and features of technologies, libraries, tools and frameworks has been made. For the extraction and the evaluation of conclusions, the same game for each machine was developed to ascertain the differences between the machines under consideration.

Keywords: game engine, HTML5, CSS, JavaScript, framework, libraries

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου, κ. Κασκάλη Θεόδωρο για την υπομονή, την βοήθεια και την καθοδήγησή του καθ' όλη τη διάρκεια εκπόνησης της εργασίας αυτής και φυσικά την οικογένεια μου, κάθε ένα μέλος της ξεχωριστά, που στέκεται αρωγός σε κάθε προσπάθεια μου.

Περιεχόμενα

1. ΕΙΣΑΓΩΓΗ	1
1.1 Αντικείμενο της πτυχιακής εργασίας.....	1
1.2 Βασικές Έννοιες.....	2
1.2.1 Library.....	2
1.2.2 Framework	2
1.2.3 Game Engine.....	3
1.2.4 WebGL API	7
1.3 Διάρθρωση της μελέτης	8
2. Βιβλιογραφική Επισκόπηση- Θεωρητικό Υπόβαθρο	9
3. Ανάλυση Μηχανών Παιχνιδιών.....	15
3.1 Μηχανή PlayCanvas	15
3.1.1 Γενικά για την μηχανή PlayCanvas	15
3.1.2 Υποστηριζόμενοι Browsers	17
3.1.3 Κόστος μηχανής PlayCanvas	17
3.1.4 Τρόπος λειτουργίας της μηχανής PlayCanvas	18
3.1.4.1 Profile.....	19
3.1.4.2 Projects.....	20
3.1.4.3 Editor.....	21
3.2 Μηχανή Construct 3.....	27
3.2.1 Γενικά για την μηχανή Construct 3.....	27
3.2.2 Υποστηριζόμενοι Browsers Construct 3	28
3.2.3 Κόστος μηχανής Construct 3	29
3.2.4 Τρόπος λειτουργίας της μηχανής Construct 3	30
3.3 Μηχανή Wimi5	34
3.3.1 Γενικά για τη μηχανή Wimi5	34
3.3.2 Υποστηριζόμενοι Browsers Wimi5	35
3.3.3 Κόστος μηχανής Wimi5.....	36
3.3.4 Τρόπος λειτουργίας της μηχανής Wimi5	36
3.3.5 Scene Editor	37
3.3.6 Logic Editor	41
3.4 Μηχανή Ceilfire	45
3.4.1 Γενικά για τη μηχανή Ceilfire.....	45
3.4.2 Υποστηριζόμενοι Browsers Ceilfire	46
3.4.3 Κόστος μηχανής Ceilfire.....	46
3.4.4 Τρόπος λειτουργίας της μηχανής Ceilfire.....	46
3.5 Μηχανή Cocos Creator	51

3.5.1 Γενικά για τη μηχανή Cocos Creator	51
3.5.2 Υποστηριζόμενοι Browsers Cocos Creator	52
3.5.3 Κόστος μηχανής Cocos Creator	52
3.5.4 Τρόπος λειτουργίας της μηχανής Cocos Creator	53
4. Ανάπτυξη Παιχνιδιών	57
4.1 Ανάπτυξη Παιχνιδιού στην PlayCanvas	58
4.1.1. Δημιουργία Αντικειμένου	58
4.1.2. Παραμετροποίηση του αντικειμένου	59
4.2 Ανάπτυξη Παιχνιδιού στην Construct 3	64
4.2.1. Littlebird.....	65
4.2.2. Γρασίδι, σωλήνες και νομίσματα	67
4.2.3. Σκοπός του παιχνιδιού	69
4.3 Ανάπτυξη Παιχνιδιού στην Wimi5	70
4.3.1. Scene Editor	71
4.3.2. Logic Editor	73
4.4 Ανάπτυξη Παιχνιδιού στην Ceilfire	80
4.4.1. Littlebird.....	82
4.4.2. Σωλήνες.....	82
4.4.3 Γρασίδι	83
4.4.4. Συγκρούσεις	84
4.4.5. Χάρτης	85
4.4.6. Βαθμολογία	85
4.5 Ανάπτυξη Παιχνιδιού στην Cocos Creator	86
5. Συμπεράσματα	91
5.1. PlayCanvas.....	92
5.2. Construct 3	92
5.3. Wimi5.....	93
5.4. Ceilfire	93
5.5. Cocos Creator.....	93
Επίλογος.....	96
Βιβλιογραφία.....	97

Κατάλογος Εικόνων

Εικόνα 1- Framework & Library.....	3
Εικόνα 2- Game engine	5
Εικόνα 3- Digital Out of Home	16
Εικόνα 4- Sign up PlayCanvas	19
Εικόνα 5- Demo νέου χρήστη PlayCanvas	19
Εικόνα 6- Profile PlayCanvas 1	20
Εικόνα 7- Overview Header PlayCanvas	20
Εικόνα 8- PlayCanvas Interface	21
Εικόνα 9- PlayCanvas Menu.....	22
Εικόνα 10- PlayCanvas Toolbar.....	22
Εικόνα 11- PlayCanvas Hierarchy	23
Εικόνα 12- PlayCanvas Inspector	24
Εικόνα 13- PlayCanvas Gizmos.....	25
Εικόνα 15- PlayCanvas Assets (2)	26
Εικόνα 14- PlayCanvas Assets (1)	26
Εικόνα 16- Το περιβάλλον της Construct 3	30
Εικόνα 17- Επεξήγηση ιδιοτήτων	31
Εικόνα 18- Οι καρτέλες του παιχνιδιού	32
Εικόνα 19- Παράθυρο επιλογής αντικειμένου	32
Εικόνα 20- Διάγραμμα ενός Event.....	33
Εικόνα 21- Dashboard Wimi5.....	36
Εικόνα 22- Project Settings Wimi5	37
Εικόνα 23- Scene Editor Wimi5	38
Εικόνα 24- Τα πάνελ του Scene Editor	38
Εικόνα 25- Asset Manager Wimi5	39
Εικόνα 26- Resources Manager Wimi5	40
Εικόνα 27- Sprites Wimi5	41
Εικόνα 28- Logic Chart Wimi5.....	42
Εικόνα 29- Συνδεσμολογία αντικειμένων Wimi5.....	43
Εικόνα 30- Είσοδος/ Έξοδος σεναρίου Wimi5	44
Εικόνα 31- Logic Tree Wimi5	45
Εικόνα 32- Περιβάλλον Ceilfire	47
Εικόνα 33- Asset Browser Ceilfire.....	47
Εικόνα 34- Μενού ρυθμίσεων Ceilfire.....	48
Εικόνα 35- Ρυθμίσεις κάμερας Ceilfire	49
Εικόνα 36- Παράθυρο δημιουργίας event Ceilfire.....	49
Εικόνα 37- Επιλογή συνθήκης για την δημιουργία event Ceilfire.....	50
Εικόνα 38- Επιλογή δράσης για την δημιουργία event Ceilfire.....	51
Εικόνα 39- Dashboard Cocos Creator	53
Εικόνα 40- Editor Cocos Creator	55
Εικόνα 41- Node Tree Cocos Creator	56
Εικόνα 42- Βιβλιοθήκη στοιχείων Cocos Creator	56
Εικόνα 43- Οι εικόνες των παιχνιδιών	57
Εικόνα 44- Η ιεράρχηση των στοιχείων του παιχνιδιού	59
Εικόνα 45- New material	59
Εικόνα 46- Οι ρυθμίσεις ενός material στοιχείου	60
Εικόνα 47- Παραμετροποίηση της οντότητας Camera	61
Εικόνα 48- Δενδροδιάγραμμα οντοτήτων.....	61

Εικόνα 49- Η οντότητα Foreground.....	62
Εικόνα 50- Η οντότητα UI	63
Εικόνα 51- Η οντότητα Frame	63
Εικόνα 52- Το Viewport του παιχνιδιού	64
Εικόνα 53- Στιγμιότυπο του παιχνιδιού με τα πάνελ σε τυχαία θέση.....	64
Εικόνα 54- Ιδιότητες animation Construct3	65
Εικόνα 55- Επιλογή συμπεριφοράς.....	66
Εικόνα 56- Το event της κίνησης.....	66
Εικόνα 57- Παράθυρο επιλογής αντικειμένου για την δημιουργία event.....	67
Εικόνα 58- Προσθήκη δράσης στο event.....	67
Εικόνα 59- Event για την δημιουργία κίνησης στο γρασίδι, στους σωλήνες και στο νόμισμα	68
Εικόνα 60- Event για την επαναλαμβανόμενη κίνηση του γρασιδιού	68
Εικόνα 61- Το event με το οποίο κάθε δυο δευτερόλεπτα το σύστημα δημιουργεί με τυχαία σειρά τους σωλήνες και τα νομίσματα.....	69
Εικόνα 62- Τα events των συγκρούσεων	69
Εικόνα 63- Βαθμολογία και επανεκκίνηση παιχνιδιού.....	70
Εικόνα 64- Στιγμιότυπο του παιχνιδιού	70
Εικόνα 65- Layer Background	71
Εικόνα 66- Layer Action.....	71
Εικόνα 67- Layers Scoreboard & gameOver	72
Εικόνα 68- Στιγμιότυπο του Scene View	72
Εικόνα 69- Τα sprites του παιχνιδιού.....	73
Εικόνα 70- Πρώτο επίπεδο σεναρίων	73
Εικόνα 71- Το σενάριο pipe-top.....	75
Εικόνα 72- Το σενάριο movingGrass.....	75
Εικόνα 73- Το σενάριο bird	76
Εικόνα 74- Το σενάριο input.....	76
Εικόνα 75- Το σενάριο movement	77
Εικόνα 76- Το σενάριο bird-pipe-top.....	78
Εικόνα 77- Το σενάριο GameOver	78
Εικόνα 78- Το σενάριο Scoreboard.....	79
Εικόνα 79- Το σενάριο init.....	79
Εικόνα 80- Το υποσενάριο του init.....	80
Εικόνα 81- Τα layers του παιχνιδιού.....	80
Εικόνα 82- Τα αντικείμενα του παιχνιδιού	81
Εικόνα 83- Η σκηνή του παιχνιδιού.....	81
Εικόνα 84- Animation Editor Ceilfire.....	82
Εικόνα 85- Ο κανόνας τοποθέτησης του grass2 στην θέση x:1200.....	84
Εικόνα 86- Οι κανόνες σύγκρουσης του χαρακτήρα	84
Εικόνα 87- Η οθόνη στο τέλος του παιχνιδιού.	85
Εικόνα 88- Ορισμός μεταβλητής	85
Εικόνα 89- Οι δράσεις που ενεργοποιούνται όταν ένα ζεύγος σωληνώσεων εξαφανίζεται από την οθόνη του παιχνιδιού	86
Εικόνα 90- Οι ρυθμίσεις του Widget για το στοιχείο Background.....	87
Εικόνα 91- Οι ρυθμίσεις του Sprite για το στοιχείο bg.....	88
Εικόνα 92- Η στοιχίση των στοιχείων ground στην σκηνή του παιχνιδιού	88
Εικόνα 93- Χρήση του σεναρίου Scroller στο στοιχείο ground.	89
Εικόνα 94- Δημιουργία animation	90
Εικόνα 95- Ρυθμίσεις collider	90
Εικόνα 96- Διάγραμμα ροής για την επιλογή της κατάλληλης μηχανής	95

Κατάλογος Πινάκων

Πίνακας 1- Απαιτήσεις Προγραμμάτων Περιήγησης.....	17
Πίνακας 2- Συγκριτικό τιμολόγιο της μηχανής Play Canvas.....	18
Πίνακας 3- Απαιτήσεις Προγραμμάτων Περιήγησης.....	28
Πίνακας 4- Συγκριτικό τιμολόγιο της μηχανής Construct 3.....	29
Πίνακας 5- Απαιτήσεις Προγραμμάτων Περιήγησης Wimi5.....	35
Πίνακας 6- Συγκριτικό τιμολόγιο της μηχανής Wimi5.....	36
Πίνακας 7- Συγκριτικός Πίνακας Χαρακτηριστικών των Μηχανών Παιχνιδιών που εξετάστηκαν	91

1. ΕΙΣΑΓΩΓΗ

1.1 Αντικείμενο της πτυχιακής εργασίας

Η ανάπτυξη της τεχνολογίας έχει προσφέρει πληθώρα συσκευών (desktop, tablet, κινητά τηλέφωνα) στους χρήστες, το gaming -είτε για λόγους διασκέδασης είτε για λόγους εκμάθησης (εκπαιδευτικά παιχνίδια)- έχει εισχωρήσει στην βιομηχανία της πληροφορικής. Όπως συμβαίνει σχεδόν σε όλες τις περιπτώσεις, έτσι και για τη δημιουργία παιχνιδιών, έχουν αναπτυχθεί κάποια πρότυπα ή αλλιώς εργαλεία, στα οποία στηρίζεται ο επαγγελματίας προγραμματιστής αλλά και ο ερασιτέχνης, που επιθυμεί να αναπτύξει το δικό του παιχνίδι. Μια γενική ορολογία για αυτά τα εργαλεία είναι “μηχανή παιχνιδιών”.

“Μια μηχανή παιχνιδιού είναι ένα σύστημα λογισμικού σχεδιασμένο για τη δημιουργία και την ανάπτυξη βιντεοπαιχνιδιών. Υπάρχουν πολλές μηχανές παιχνιδιών οι οποίες είναι σχεδιασμένες να δουλεύουν σε κονσόλες βιντεοπαιχνιδιών και λειτουργικά συστήματα επιτραπέζιων υπολογιστών όπως τα Microsoft Windows, το Linux, και το Mac OS X. Η κεντρική λειτουργικότητα που παρέχεται τυπικά από μια μηχανή παιχνιδιού περιλαμβάνει μια μηχανή φωτοαπόδοσης ("renderer") για 2D ή 3D γραφικά, μια μηχανή φυσικής ή εντοπισμού συγκρούσεων (collision detection, καθώς και collision response), ήχο, scripting, animation, τεχνητή νοημοσύνη, δικτύωση, streaming, διαχείριση μνήμης, νήματα (threading), υποστήριξη τοπικοποίησης και ένα γράφο σκηνής(scene graph). Η διαδικασία της ανάπτυξης παιχνιδιού συχνά οικονομικοποιείται με το ότι σε μεγάλο μέρος η ίδια μηχανή παιχνιδιού επαναχρησιμοποιείται για να δημιουργηθούν διαφορετικά παιχνίδια.” (wikipedia, μηχανή παιχνιδιού, 2017)

Η παρούσα διπλωματική εργασία αναφέρεται στην έρευνα και τη σύγκριση των υφιστάμενων τεχνολογιών (frameworks, libraries, game engines) για την ανάπτυξη παιχνιδιών βασισμένα στην γλώσσα σήμανσης HTML5. Σκοπός της έρευνας είναι η ενημέρωση των χρηστών, που επιθυμούν να αναπτύξουν το δικό τους παιχνίδι σχετικά με τα χαρακτηριστικά, τις δυνατότητες και τις λειτουργίες της κάθε μηχανής, ώστε να καταλήξουν στην επιλογή της καταλληλότερης για αυτούς μηχανής, βάση του παιχνιδιού που θέλουν να αναπτύξουν. Πιο συγκεκριμένα η εργασία απευθύνεται στο κοινό που δεν ασχολείται επαγγελματικά με τον προγραμματισμό. Πόσο εύκολο είναι για έναν απλό χρήστη του διαδικτύου, με μηδενικές γνώσεις πάνω στην ανάπτυξη λογισμικού, να ερευνήσει, να αναπτύξει και στο τέλος να αποκομίσει κέρδος από ένα παιχνίδι βασισμένο στην HTML5. Η μελέτη επικεντρώνεται σε τέσσερις μηχανές παιχνιδιών που είναι cloud-based, καθώς δεν απαιτείται η εγκατάσταση λογισμικού και σε μια μηχανή για την οποία απαιτείται η εγκατάσταση λογισμικού.

1.2 Βασικές Έννοιες

Κατά τη διάρκεια της έρευνας πάνω στις μηχανές παιχνιδιών εντοπίστηκαν αρκετές ορολογίες, η οποία κάθε μια ξεχωριστά αποτελεί αντικείμενο μελέτης. Στο παρόν κεφάλαιο θα αναλυθούν εν τάχει οι ορολογίες:

- Library
- Framework
- Game Engine
- Online Game Engine
- WebGL API

Σκοπός αυτής της σύντομης ανάλυσης είναι να εισάγει τον αναγνώστη ομαλά στο γνωστικό πεδίο της έρευνας και κατά συνέπεια και της εργασίας. Ο όρος library και ο όρος framework αποτελούν τα θεμελιώδη στοιχεία, καθώς η χρήση τους αποσκοπεί στην ταχύτερη παραγωγή των παιχνιδιών, μειώνοντας τον χρόνο γραφής κώδικα. Στην συνέχεια δίνεται ο ορισμός της μηχανής παιχνιδιού, που είναι και το βασικό αντικείμενο μελέτης της εργασίας καθώς επίσης αναλύεται η υποκατηγορία της online μηχανής παιχνιδιού, που θα μας απασχολήσει στη συνέχεια. Τέλος αναλύεται η βιβλιοθήκη WebGL. Η επιλογή ανάλυσης αυτής της βιβλιοθήκης από τις εκατοντάδες που κυκλοφορούν δεν είναι τυχαία. Το WebGL μαζί με το DirectX και το OpenGL είναι APIs γραφικών που υποστηρίζονται από τις μηχανές παιχνιδιών. Σε αντίθεση όμως με το DirectX και το OpenGL που χρησιμοποιούνται σε διάφορες πλατφόρμες, το WebGL είναι διαθέσιμο σε κάθε πρόγραμμα περιήγησης χωρίς την χρήση plug-ins.

1.2.1 Library

Με τον όρο library, εννοείται η συλλογή από έτοιμα υπο-προγράμματα κώδικα και δεδομένων με σκοπό την επαναχρησιμοποίησή τους στην εκτέλεση εργασιών και ανάπτυξη λογισμικού. Κάθε βιβλιοθήκη καλείται από το πρόγραμμα για συγκεκριμένες λειτουργίες, όπως είναι για παράδειγμα η αναπαραγωγή ήχου ή η κίνηση των χαρακτήρων (animation). Χρησιμοποιώντας δηλαδή μια βιβλιοθήκη ο χρήστης, αποφεύγει να γράψει από την αρχή τον κώδικα που απαιτείται για την υλοποίηση των διαφόρων λειτουργιών, όπως τα παραδείγματα που αναφέρθηκαν παραπάνω (gamefromscratch 2015/ wikipedia).

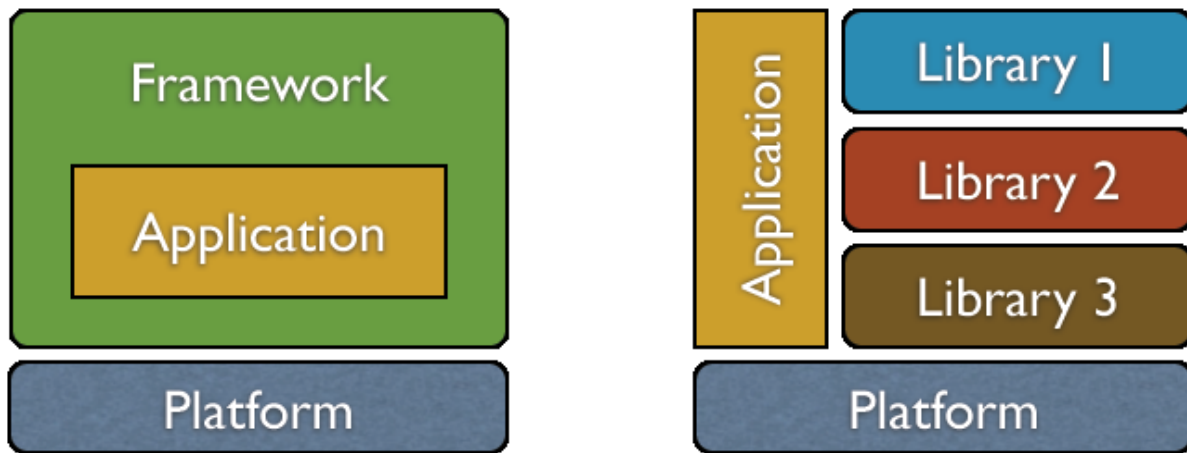
1.2.2 Framework

Με τον όρο framework εννοείται μια συλλογή από βιβλιοθήκες και εργαλεία, που σκοπό έχει την επίλυση ενός προβλήματος, στην περίπτωση της διπλωματικής αυτής, την δημιουργία ενός παιχνιδιού. Ο στόχος ενός framework είναι να βελτιώσει την αποτελεσματικότητα και την παραγωγικότητα των προγραμματιστών, καθώς επίσης να βελτιώσει την ποιότητα, την αξιοπιστία

και την ευρωστία του νέου λογισμικού. Η χρήση των frameworks επιτρέπει στους προγραμματιστές να ασχοληθούν στις μοναδικές απαιτήσεις της εφαρμογής τους αντί να ξοδεύουν χρόνο στην ανάπτυξη κώδικα για λειτουργίες για τις οποίες ήδη έχει αναπτυχθεί και ελεγχθεί κώδικας.

Πολλές φορές οι όροι library και framework εξισώνονται, καθώς και μια βιβλιοθήκη και ένα framework λειτουργούν με την επαναχρησιμοποίηση κώδικα. Ωστόσο υπάρχει μια σημαντική διαφορά μεταξύ τους και αυτή η διαφορά αποκαλείται inversion of control- αντιστροφή ελέγχου.

Κατά την χρήση μιας βιβλιοθήκης, τα αντικείμενα και οι μέθοδοι που εφαρμόζονται από την βιβλιοθήκη δημιουργούνται από τον κώδικα που αναπτύσσει ο προγραμματιστής. Αυτό σημαίνει ότι ο προγραμματιστής πρέπει να γνωρίζει ποια αντικείμενα ή μεθόδους πρέπει να καλέσει για να επιτύχει τον στόχο του. Σε αντίθεση με την χρήση ενός framework ο προγραμματιστής δεν καλεί τις μεθόδους του framework αλλά το framework καλεί τον κώδικα του προγραμματιστή. Με αυτόν τον τρόπο μπορούμε να πούμε ότι το framework ορίζει την ροή του ελέγχου για την εφαρμογή (Freeman & Robson, 2011).



Εικόνα 1- Framework & Library

Όταν δημιουργείται μια εφαρμογή χρησιμοποιώντας ένα συγκεκριμένο framework, η εφαρμογή “ζει” μέσα στο framework αυτό.

Εναλλακτικά, μια εφαρμογή μπορεί να αναπτυχθεί χρησιμοποιώντας βιβλιοθήκες. Η εφαρμογή υπάρχει μόνη της και χρησιμοποιεί τις βιβλιοθήκες για να κάνει κάποιο μέρος του έργου.

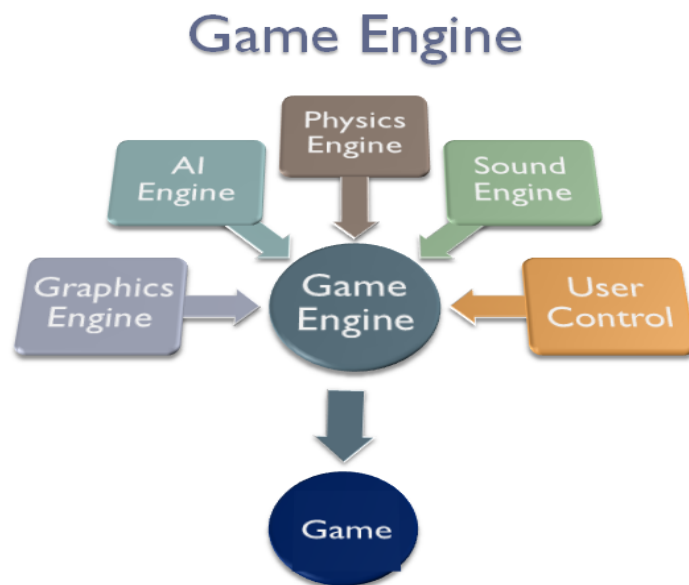
1.2.3 Game Engine

Ως Μηχανή Παιχνιδιού- Game Engine αναφέρουμε το μέρος του πηγαίου κώδικα του παιχνιδιού που παρέχει όλες τις λειτουργίες που προορίζονται για επαναχρησιμοποίηση σε περισσότερα παιχνίδια και επιτρέπουν την κωδικοποίηση και εκτέλεση του παιχνιδιού. Τα είδη διαφέρουν αναλόγως, το λειτουργικό σύστημα πάνω στο οποίο θα «τρέξουν» (Mac OS x, windows,

Linux), την πλατφόρμα για την οποία προορίζονται τα παιχνίδια, το είδος παιχνιδιού (*first person shooter, strategy, adventure*), τη γλώσσα που διαθέτουν, όπως και αν θέλουμε τα παιχνίδια να έχουν 3d ή 2d γραφικά. Ακόμα, μερικές από τις μηχανές, επιτρέπουν όχι μόνο την εισαγωγή και τη διαχείριση τρισδιάστατων αντικειμένων αλλά και την μοντελοποίηση τους (Πύλιος & Σοϊδης, 2011).

Πριν από τις μηχανές παιχνιδιών, τα παιχνίδια τυπικά γραφόντουσαν ως μοναδικές οντότητες. Ένα παιχνίδι για το Atari 2600, για παράδειγμα, έπρεπε να σχεδιαστεί από το μηδέν για να κάνει βέλτιστη χρήση του υλικού εμφάνισης – αυτή η κεντρική ρουτίνα εμφάνισης σήμερα καλείται πυρήνας από τους δημιουργούς Retro60 παιχνιδιών. Άλλες πλατφόρμες είχαν περισσότερα περιθώρια, αλλά ακόμα και όταν η εμφάνιση δεν ήταν μέριμνα, οι περιορισμοί μνήμης συνήθως σαμπόταραν προσπάθειες για τη δημιουργία της βαριάς σε δεδομένα σχεδίασης που μια μηχανή απαιτούσε. Ακόμα και σε πιο φιλικές πλατφόρμες, πολύ λίγα μπορούσαν να χρησιμοποιηθούν ανάμεσα σε παιχνίδια. Η γοργή ανάπτυξη του arcade – ο ακρογωνιαίος λίθος της αγοράς – σήμαινε ότι το μεγαλύτερο μέρος του κώδικα θα έπρεπε να πεταχτεί μετά, καθώς οι μεταγενέστερες γενιές παιχνιδιών θα χρησιμοποιούσαν τελείως διαφορετικές σχεδιάσεις παιχνιδιών, οι οποίες εκμεταλλευόντουσαν επιπλέον πόρους. Τοιουτοτρόπως οι περισσότερες σχεδιάσεις παιχνιδιών τη δεκαετία του 1980 σχεδιάζόντουσαν δια μέσω ενός σκληρά κωδικοποιημένου συνόλου κανόνων με μια μικρή ποσότητα δεδομένων επιπέδου και γραφικών (Γιακαλάρας, 2016).

Όπως βλέπει κανείς και στην Εικόνα 2, η μηχανή παιχνιδιών διαχειρίζεται και ελέγχει άλλες “μικρότερες” μηχανές, όπως την μηχανή γραφικών, την μηχανή τεχνητής νοημοσύνης, την μηχανή φυσικής, την μηχανή ήχου και ούτω καθεξής. Η μηχανή παιχνιδιών διαχειρίζεται όλα τα δεδομένα του παιχνιδιού και κάνει κάθε παιχνίδι να ζωντανεύει έχοντας όλες τις μηχανές συνδεδεμένες. Η βασική της λειτουργία είναι η εξής: τα δεδομένα του παιχνιδιού τα στέλνει στην μηχανή ΑΙ για τη λήψη αποφάσεων, στη συνέχεια στέλνει τις αποφάσεις στην μηχανή Ήχου για αναπαραγωγή, στη συνέχεια τις στέλνει στη Φυσική για πραγματική αλληλεπίδραση και τελικά τις στέλνει στα γραφικά για απόδοση και εμφάνιση στην οθόνη. Αυτή η διαδικασία πρέπει να επαναλαμβάνεται όσο το παιχνίδι λειτουργεί. Αυτός είναι ο τρόπος με τον οποίο μια μηχανή παιχνιδιού δημιουργεί ένα παιχνίδι.



Εικόνα 2- Game engine

Συνοπτικά, πλεονεκτήματα μια μηχανής παιχνιδιού αποτελούν τα εξής:

1. Ο χρήστης μπορεί να ξεκινήσει γρήγορα και άμεσα την δημιουργία του παιχνιδιού, καθώς δεν απαιτείται ιδιαίτερη γνώση προγραμματισμού,
2. Είναι σημαντικό ότι ο χρήστης έχει τη δυνατότητα, να εξαγει το παιχνίδι σε πολλές διαφορετικές πλατφόρμες.

Φυσικά, υπάρχουν και μειονεκτήματα. Δυο από τα πιο σημαντικά, είναι:

1. Ο χρήστης είναι εγκλωβισμένος στην αρχιτεκτονική της μηχανής και θα του είναι αδύνατο να διορθώσει τυχόν σφάλμα που θα προκύψει κατά την χρήση της, και
2. Ο προγραμματιστής δεν εξελίσσεται με τον ίδιο ρυθμό που θα είχε αν έγραφε μόνος του τον κώδικα για το παιχνίδι.

Τέλος, είναι σημαντικό να σημειωθεί ότι δεν υπάρχει ένας ολοκληρωμένος και γρήγορος ορισμός για τον κάθε όρο που αναλύθηκε παραπάνω, καθώς η σημασία του καθενός προδιαγράφεται ανάλογα με την χρήση που κάνει ο εκάστοτε προγραμματιστής και την εμπειρική προσέγγισή του.

1.2.3.1 Online Game Engine

Υπάρχουν δύο κατηγορίες μηχανών παιχνιδιών. Οι online μηχανές παιχνιδιών οι οποίες είναι cloud-based και οι μηχανές που απαιτούν την λήψη και εγκατάσταση συγκεκριμένων

εργαλείων και προγραμμάτων για την χρήση τους. Μία online μηχανή επιτρέπει στον χρήστη να εργάζεται σε οποιοδήποτε λειτουργικό σύστημα με τη χρήση απλώς ενός προγράμματος περιήγησης (Elliot, 2014).

Πλεονεκτήματα Online Μηχανών Παιχνιδιών:

1. Μόνιμη Αποθήκευση:

Μία online μηχανή αποθηκεύει αυτόματα όλες τις ενέργειες του χρήστη. Αυτό καθιστά ευκολότερη την εργασία του χρήστη, καθώς δεν χρειάζεται να αποθηκεύει ξανά και ξανά το έργο του για να αποφύγει απροσδόκητες απώλειες.

2. Ασφάλεια:

Οι online μηχανές χάρη στην cloud αποθήκευση που προσφέρουν προστατεύουν τον χρήστη από οποιαδήποτε ζημιά συμβεί στον υπολογιστή του (κλοπή, ιός, καταστροφή υλικού) και έτσι δεν επηρεάζεται η ανάπτυξη του παιχνιδιού. Όλα τα δεδομένα αποθηκεύονται στο σύννεφο και διατίθενται ανά πάσα στιγμή από οποιαδήποτε συσκευή.

3. Εργασία σε συνεργασία σε πραγματικό χρόνο:

Διάφοροι προγραμματιστές με διαφορετικό προφίλ μπορούν να εργαστούν στο ίδιο project. Επειδή η εργασία του κάθε προγραμματιστή πάνω στο Project είναι σε πραγματικό χρόνο, ο καθένας βλέπει τι αλλαγές κάνει ο άλλος. Στις περισσότερες μηχανές υπάρχει σύστημα διαχείρισης χρηστών με διαφορετικούς ρόλους έτσι ώστε να καλύπτονται όλες οι ανάγκες ενός project.

4. Άμεση Επικοινωνία:

Οι προγραμματιστές μπορούν να επικοινωνούν μέσω συστημάτων άμεσων μηνυμάτων που είναι ενσωματωμένα στις περισσότερες Online μηχανές.

5. Άμεσες ενημερώσεις και βελτιώσεις:

Δεδομένου ότι το λογισμικό της μηχανής το διαχειρίζεται απευθείας ο πάροχος υπηρεσιών, μπορεί να ενημερωθεί η τρέχουσα έκδοση ανά πάσα στιγμή με μια νέα, εισάγοντας βελτιώσεις και διορθώνοντας σφάλματα. Αυτό καθιστά πολύ πιο εύκολη την ενημέρωση σε νέες εκδόσεις της μηχανής του παιχνιδιού, καθώς ο προγραμματιστής

παιχνιδιών δεν χρειάζεται να κάνει λήψη νέων εκδόσεων ή να εγκαταστήσει ενημερώσεις κώδικα.

6. Ταχύτητα διανομής:

Η έκδοση του κάθε παιχνιδιού είναι μια απλή διαδικασία η οποία ολοκληρώνεται μέσα σε λίγα click. Οι περισσότερες online μηχανές φιλοξενούν το παιχνίδι στον server τους και ο χρήστης μπορεί να διανείμει το url του παιχνιδιού, με όποιο τρόπο επιθυμεί πχ, μέσω mail, social media κτλ.

7. Ενσωμάτωση με άλλες υπηρεσίες και εφαρμογές web:

Ένα άλλο πλεονέκτημα που προσφέρει μια online μηχανή ανάπτυξης είναι η δυνατότητα ενσωμάτωσης με άλλες υπηρεσίες και εφαρμογές ιστού. Μια online μηχανή μπορεί να ενσωματωθεί, για παράδειγμα, με τα API των υπηρεσιών όπως το Dropbox, το Google Drive ή το Box για την αποθήκευση. Με το Charbeat ή το New Relic για αναλύσεις. GitHub ή Bitbucket για έλεγχο έκδοσης. Trello ή Wunderlist για τη διαχείριση εργασιών, οποιοδήποτε κοινωνικό δίκτυο. κ.λπ., κλπ. Οι δυνατότητες είναι ατελείωτες.

Στον αντίποδα, ένα σημαντικό μειονέκτημα των online μηχανών είναι η πρόσβαση στο internet. Είναι απαραίτητη η πρόσβαση στο διαδίκτυο για να μπορέσει ο χρήστης να εκμεταλλευθεί και να αξιοποιήσει πλήρως όλες τις δυνατότητες που του δίνει μια online μηχανή.

(<http://wimi5.com/online-game-engine-discover-why-you-should-use-it/>)

1.2.4 WebGL API

To WebGL (Web Graphics Library) είναι ένα Javascript API που χρησιμοποιείται για αναπαράσταση και απόδοση διαδραστικών 3D και 2D γραφικών σε οποιοδήποτε συμβατό web browser χωρίς την χρήση plug-ins. Το WebGL έχει ενσωματωθεί πλήρως σε όλα τα web πρότυπα του προγράμματος περιήγησης, επιτρέποντας στη GPU να κάνει χρήση της φυσικής, της επεξεργασίας εικόνας και κάποιων εφέ ως μέρος της ιστοσελίδας. Τα στοιχεία του WebGL μπορούν να αναμειχθούν με άλλα στοιχεία HTML και με άλλα τμήματα της σελίδας ή του φόντου της σελίδας. Τα WebGL προγράμματα αποτελούνται από κώδικα γραμμένο σε Javascript και από κώδικα (shader) που εκτελείται στην μονάδα επεξεργασίας γραφικών του υπολογιστή (GPU) (wikipedia, WebGL, 2018). Με την χρήση του WebGL δίνεται η δυνατότητα στους προγραμματιστές να αξιοποιούν την πλήρη ισχύ του υλικού τους (κάρτα γραφικών) για 3D αποδόσεις γραφικών μέσα από το πρόγραμμα περιήγησης με την χρήση JavaScript. Πριν από την χρήση του WebGL οι

προγραμματιστές έπρεπε να βασίζονται σε plugins ή native εφαρμογές τις οποίες θα έπρεπε οι χρήστες να κατεβάσουν και να εγκαταστήσουν ώστε να βιώσουν την 3D εμπειρία. Με βάση την OpenGL ES 2.0 (το λογισμικό API που τρέχει σε κινητές συσκευές), το WebGL επιτρέπει σε έναν προγραμματιστή να καθορίσει τα αντικείμενα και τις λειτουργίες που εμπλέκονται στην παραγωγή υψηλής ποιότητας γραφικών και ειδικά έγχρωμες εικόνες των 3D αντικειμένων. Αναζητώντας κάποιος τον όρο WebGL θα συναντήσει συχνά την έκφραση “επιτάχυνση κάρτας γραφικών”. Αυτό πρακτικά σημαίνει πως η τεχνολογία WebGL αξιοποιώντας την κάρτα γραφικών πετυχαίνει να “ζωγραφίζονται” με μεγαλύτερη ταχύτητα και ποιότητα τα γραφικά στο web. Μία από τις διασημότερες βιβλιοθήκες του WebGL είναι η three.js και η πιο διαδεδομένη χρήση του είναι σε παιχνίδια βολής πρώτου προσώπου, που απαιτούν υψηλής ποιότητας γραφικά (Parisi, 2014).

Συνοπτικά:

- Πρόκειται για API σχεδίασης χαμηλού επιπέδου
- Βασίζεται στην τεχνολογία OpenGL ES 2.0
- Αλληλεπιδρά με περιεχόμενα του web
- Είναι cross-platform
- Δεν αποτελεί ετικέτα της HTML5
- Είναι ελεύθερο λογισμικό και δεν απαιτείται άδεια χρήσης.

1.3 Διάρθρωση της μελέτης

Η διάρθρωση της μελέτης έχει ως εξής:

Στο πρώτο κεφάλαιο αναφέρεται το αντικείμενο της μελέτης, ο σκοπός της παρούσας εργασίας και αναλύονται κάποιες βασικές τεχνολογίες όπως library, framework και game engine. Στο δεύτερο μέρος γίνεται μία ανασκόπηση της βιβλιογραφίας η οποία είναι σχετική με τη μελέτη. Στο τρίτο γίνεται η θεωρητική ανάλυση των πέντε μηχανών που επιλέχθηκαν να αναπτυχθούν και να εξεταστούν με το ίδιο παιχνίδι αναφοράς. Εξετάστηκαν οι τεχνολογίες που υποστηρίζονται καθώς και το κόστος και ο τρόπος λειτουργίας τους. Στο τέταρτο κεφάλαιο παρουσιάζεται η εκάστοτε ανάπτυξη του παιχνιδιού-αναφοράς. Τέλος, στο πέμπτο κεφάλαιο εξάγονται τα συμπεράσματα της συγκριτικής μελέτης και αναλύονται τα αποτελέσματα.

2. Βιβλιογραφική Επισκόπηση- Θεωρητικό Υπόβαθρο

Στην βιβλιογραφική έρευνα εντοπίστηκαν πολλές μελέτες, άρθρα και εργασίες με παρεμφερή θεματολογία με την παρούσα μελέτη. Αυτό οφείλεται στην συνεχή ανάγκη της αγοράς για ανάπτυξη νέων παιχνιδιών, τα οποία θα “κερδίσουν” του καταναλωτές. Σε αυτό το μέρος της εργασίας, θα παρουσιάσουμε το υλικό που εντοπίστηκε κατά την έρευνα, πώς δηλαδή διαρθρώνεται η κάθε εργασία, η θεματολογία της και τα χαρακτηριστικά που ανέλυσαν.

Η πρώτη εργασία που ξεχώρισε ήταν η διατριβή του Μ. Γιακαλάρα, (2016) «Εικονικοί κόσμοι». Το μοτίβο της εργασίας είναι αρκετά θεωρητικό, ωστόσο εισαγάγει τον αναγνώστη πάνω στις μηχανές παιχνιδιών και στα κύρια χαρακτηριστικά τους (Γιακαλάρας, 2016). Στη συνέχεια, ακολουθούν δυο εργασίες και ένα άρθρο, όπου το κοινό χαρακτηριστικό τους είναι ο διαχωρισμός των μηχανών. Ωστόσο σε κάθε κείμενο ο διαχωρισμός των μηχανών γίνεται με διαφορετικά κριτήρια. Στο άρθρο “A Brief Review of Game Engines for Educational and Serious Games Development”, οι συγγραφείς διαχωρίζουν τις μηχανές ως προς το περιβάλλον εκτέλεσής τους (Marin et al., 2017). Εάν δηλαδή οι μηχανές είναι HTML-Based ή αν είναι ιδιόκτητες, εάν δηλαδή απαιτείται η εγκατάσταση ειδικού λογισμικού για την λειτουργία τους ή όχι. Στην εργασία “Ανάπτυξη και αξιολόγηση ενός διαδραστικού ηλεκτρονικού παιχνιδιού με αφηγηματικό περιεχόμενο”, ο συγγραφέας για να προχωρήσει στην επιλογή της μηχανής για την ανάπτυξη του παιχνιδιού του, αρχικά μελετάει τις πλατφόρμες που υπάρχουν και στην συνέχεια αναλύει τις μηχανές που έχει στη διάθεσή του (Κούτρας, 2014). Μια άλλη οπτική γωνία στον διαχωρισμό των μηχανών, παρουσιάζεται στην εργασία “Η χρήση οπτικών εργαλείων ανάπτυξης διαδικτυακών παιχνιδιών(online games). Συγκριτική μελέτη δυνατοτήτων και παραδειγματική εφαρμογή χρήσης τους στην κατασκευή ολοκληρωμένου διαδικτυακού παιχνιδιού”. Ο συγγραφέας ταξινομεί τις μηχανές που αναλύει βάσει δυο κριτηρίων είτε σε οπτική μηχανή, δηλαδή μηχανή που βασίζεται σε σύστημα drag and drop είτε σε μηχανή που απαιτεί τη συγγραφή κώδικα για την ανάπτυξη του παιχνιδιού (Βακιρτζής, 2017). Τέλος στην εργασία “Investigation into the use of HTML5 game engines to create a responsive social educational game for children”, αναφέρεται η ανάπτυξη ενός παιχνιδιού με στοχευμένο κοινό. Για την ανάπτυξη του παιχνιδιού, αναλύθηκαν μερικές από τις δημοφιλέστερες μηχανές και συγκρίθηκαν τεχνικά χαρακτηριστικά, όπως για παράδειγμα η υποστήριξη 3D γραφικών (Kashyap, 2015). Από την μελέτη των εργασιών αυτών, διακρίνεται πόσο μεγάλο είναι το φάσμα έρευνας πάνω στις μηχανές παιχνιδιών και πώς οι απαιτήσεις του παιχνιδιού που θέλει κάποιος να αναπτύξει ή ακόμα και οι προσωπικές προτιμήσεις του προγραμματιστή καθορίζουν τα χαρακτηριστικά επιλογής μιας μηχανής.

Στην διδακτορική διατριβή του Μ. Γιακαλάρα με τίτλο “Εικονικοί Κόσμοι: Διερεύνηση των ζητημάτων των σύγχρονων μηχανών παιχνιδιών ως προς το φωτορεαλισμό στον τομέα φωτισμού και του αντίκτυπου στον χρήστη ενός παιχνιδιού” (2016), αν και το κυρίως θέμα που διαπραγματεύεται είναι αποκλειστικά ο φωτορεαλισμός στα παιχνίδια και τι αντίκτυπο έχει στον χρήστη και στην δημοφιλία του παιχνιδιού, πραγματοποιεί μια ουσιαστική έρευνα πάνω στις μηχανές παιχνιδιών και τα κύρια χαρακτηριστικά τους. Για την υλοποίηση της έρευνας, οι μηχανές διαχωρίζονται σε δυο είδη, στις μηχανές παιχνιδιών και στις μηχανές γραφικών και χρησιμοποιεί τον όρο “game middle-ware”, ο οποίος είναι ο εμπορικός όρος για τις μηχανές παιχνιδιών καθώς παρέχουν μια ευέλικτη και επαναχρησιμοποιήσιμη πλατφόρμα λογισμικού, η οποία παρέχει όλη την κεντρική λειτουργικότητα που απαιτείται, για την ανάπτυξη μια εφαρμογής παιχνιδιού. Η μεθοδολογία που ακολουθήθηκε είναι η εξής: στο πρώτο κεφάλαιο διατυπώνεται το ερευνητικό ερώτημα της εργασίας, ο σκοπός, οι στόχοι και οι όροι που χρησιμοποιούνται, στο δεύτερο κεφάλαιο γίνεται η ανασκόπηση της βιβλιογραφίας και η θεωρητική προσέγγιση του θέματος, ενώ στο κεφάλαιο τρία γίνονται γνωστά τα αποτελέσματα του ερωτηματολογίου που έχει τεθεί στους gamers που έλαβαν μέρος στην έρευνα. Στο τέταρτο κεφάλαιο αναλύεται η χρήση του φωτός τόσο στην δημιουργία του παιχνιδιού όσο και στην ανέλιξή του. Στο πέμπτο κεφάλαιο γίνεται η ανάλυση συγκεκριμένου ηλεκτρονικού παιχνιδιού σε σχέση με τη χρήση φωτορεαλιστικών γραφικών και την δημιουργία του, το σενάριο, το λογισμικό ανάπτυξης. Τέλος, στο έκτο κεφάλαιο ο συγγραφέας δηλώνει πως με την χρήση σύγχρονων μηχανών παιχνιδιών και φωτορεαλιστικών γραφικών μπορούν να αναπαρασταθούν με ακρίβεια φυσικά μνημεία, ιστορικές μάχες και άλλες προσομοιώσεις που κάποτε θα ήταν αδύνατες. Εξίσου σημαντικός είναι ο συγκεντρωτικός πίνακας που παρουσιάζεται σε παράρτημα της εργασίας και εμφανίζει στοιχεία μηχανών γραφικών και παιχνιδιών. Τα στοιχεία αυτά είναι η γλώσσα προγραμματισμού με την οποία έχουν αναπτυχθεί, εάν είναι cross platform, τι φωτοαπόδοση έχουν, από ποιες πλατφόρμες υποστηρίζονται, γνωστά παιχνίδια που έχουν αναπτυχθεί με την εκάστοτε μηχανή και αν είναι ελεύθερα ή όχι (Γιακαλάρας, 2016).

Το άρθρο με τίτλο “A Brief Review of Game Engines for Educational and Serious Games Development”, των Humberto Marin-Vega, Giner Alor-Hernández, Ramon Zatarain-Cabada, Maria Lucia Barron-Estrada και Jorge Luis García-Alcaraz (2017) είναι και αυτό σχετικό με την παρούσα έρευνα. Θέμα του άρθρου αποτελεί ο σημαντικός ρόλος του παιχνιδιού στην εκπαίδευση και τον τρόπο ανάπτυξης των παιχνιδιών ως εκπαιδευτικά εργαλεία με τη χρήση των game engines. Αρχικά διαχωρίζει τα λογισμικά σε δυο είδη: α) σε HTML5-based και β) σε ιδιόκτητα game engines. Η διαφορά τους είναι πως τα πρώτα δεν έχουν ιδιαίτερες απαιτήσεις σε πλατφόρμες και λογισμικά, χρησιμοποιούν απλά τον browser, ενώ τα δεύτερα έχουν αναπτυχθεί σε συγκεκριμένη γλώσσα και για συγκεκριμένες πλατφόρμες (Viveros & Garcia, 2009). Τα βασικά χαρακτηριστικά για την

επιλογή των μηχανών που θα αναλύσουν είναι η υποστήριξη ήχου, βίντεο, 2D και 3D γραφικών, η άδεια χρήσης, η τελευταία έκδοση και εάν είναι crossplatform. Επιπλέον κύριο χαρακτηριστικό επιλογής των μηχανών που κατηγοριοποιούνται ως ιδιόκτητες, είναι η γλώσσα προγραμματισμού. Συνολικά, εξετάζονται δεκαοκτώ μηχανές, οι οποίες είναι οι: Construct2, ImpactJS, Quintus, Wade, pixi.js, EaselJS, melonJS, Three.js, Phaser, Playcanvas, Unity, Cryengine, UnrealEngine, Cocos2d, Blender, BigWorld, Leadwerks, HeroEngine. Οι συγγραφείς του άρθρου εισχώρησαν σε μια ποιοτική αξιολόγηση όλων των μηχανών για την δημιουργία εφαρμογών παιχνιδιών και παιχνιδιών σοβαρού σκοπού. Η μεθοδολογία της έρευνας βασιζόταν στον έλεγχο παραδειγμάτων τα οποία αντλούσαν από τις επίσημες ιστοσελίδες των μηχανών. Εάν τα παραδείγματα αυτά δεν αρκούσαν για την επιβεβαίωση των χαρακτηριστικών που εξετάζονταν, τότε προχωρούσαν στην ανάπτυξη δικών τους παραδειγμάτων. Τα αποτελέσματα της έρευνάς τους, παρουσιάζονται σε έναν συγκεντρωτικό πίνακα. Ο σκοπός του πίνακα αυτού και εν τέλει του άρθρου είναι όταν οι προγραμματιστές ξεκινούν την δημιουργία ενός νέου παιχνιδιού να μπορούν να προσδιορίζουν ποια χαρακτηριστικά θα υπάρχουν στο νέο παιχνίδι και ποια μηχανή παιχνιδιού είναι αυτή που θα τους προσφέρει την καλύτερη υποστήριξη για το συγκεκριμένο project. Τα γενικά χαρακτηριστικά που πρέπει να αναπτύσσονται με τις μηχανές είναι οι κανόνες, οι στόχοι, οι προκλήσεις, η συνεργασία και ο ανταγωνισμός μεταξύ των παικτών και τέλος η ανατροφοδότηση και η αξιολόγηση (Marin et al., 2017).

Η εργασία με τίτλο “Ανάπτυξη και αξιολόγηση ενός διαδραστικού ηλεκτρονικού παιχνιδιού με αφηγηματικό περιεχόμενο” του Θ. Κούτρα πραγματεύεται την ανάπτυξη και την αξιολόγηση ενός διαδραστικού παιχνιδιού περιπέτειας τρίτου προσώπου σε γραφικό περιβάλλον. Τα αρχικά ερωτήματα που έθεσε ο συγγραφέας ήταν δυο και αφορούσαν την επιλογή της πλατφόρμας ανάπτυξης του παιχνιδιού και την επιλογή της μηχανής για τον προγραμματισμό του παιχνιδιού. Η επιλογή της πλατφόρμας έχει να κάνει τόσο με πρακτικά κριτήρια, όπως ο αριθμός των χρηστών όσο και με θεωρητικά που αφορούν την μελλοντική ανάπτυξη της πλατφόρμας. Επιπλέον κριτήρια που εξετάζει για την επιλογή της πλατφόρμας είναι τα επαγγελματικά οφέλη που θα προσκομίσουν από την πλατφόρμα, τι κόστος υπάρχει για την απόκτηση των Software Development Kit (SDK) ή το τεστάρισμα, πόσο εύκολη είναι η διαδικασία εντοπισμού σφαλμάτων (debugging), εάν υπάρχουν ανάλογα είδη παιχνιδιών στην πλατφόρμα, ποιες είναι οι δυνατότητες marketing της πλατφόρμας, ακόμα και εάν υπάρχει η δυνατότητα αποφυγής εκδότη και επιτρέπεται το self-publishing. Λαμβάνοντας υπόψη του τα παραπάνω κριτήρια, ο συγγραφέας ανάμεσα σε οκτώ πλατφόρμες επιλέγει ως πλατφόρμα ανάπτυξης, τους προσωπικούς υπολογιστές (PC). Η επόμενη επικρατέστερη πλατφόρμα, η iOS, η οποία απορρίφθηκε κυρίως για οικονομικούς λόγους και διότι δεν παρείχε την δυνατότητα για cross platform. Η επιλογή της μηχανής βασίστηκε τόσο σε προγραμματιστικά κριτήρια, όσο και σε οικονομικά. Τα πιο σημαντικά από αυτά είναι η γλώσσα

προγραμματισμού, η εισαγωγή έτοιμων σεναρίων, η υπολογιστική ισχύς που απαιτείται, η υποστήριξη από την εταιρία παραγωγής, η ύπαρξη ενεργής κοινότητας χρηστών, το κόστος άδειας της μηχανής και της μελλοντικής αναβάθμισης, η ευκολία χρήσης, η ποιότητα και η ποσότητα ανάλογων παιχνιδιών που έχουν αναπτυχθεί με την μηχανή καθώς και εάν μπορεί να αντεπεξέλθει αποδοτικά σε οποιεσδήποτε μελλοντικές ευκαιρίες, όπως είναι η δυνατότητα ενσωμάτωσης νέων τεχνολογιών και η ευκολία αλλαγής ή πρόσθεσης κώδικα. Κατόπιν της έρευνάς του, ο συγγραφέας απορρίπτει της μηχανές Adventure Game Studio και Unity 3D λόγω “ηλικίας” και κόστους και καταλήγει στην Visionaire Studio, η οποία είναι εύχρηστη και διαθέτει ελεύθερο λογισμικό. Η μεθοδολογία η οποία ακολουθήθηκε για την ανάπτυξη του παιχνιδιού βασίστηκε στο τετράπτυχο σενάριο-διεπαφή-γραφικά-ήχος. Ο συγγραφέας έχει επιλέξει από την αρχή του παιχνιδιού πως θα είναι ένα point & click παιχνίδι περιπέτειας, άρα από την αρχή δίνεται βαρύτητα στο animation και στην φυσικότητα. Ο χειρισμός του avatar γίνεται αποκλειστικά με το ποντίκι, ωστόσο υπάρχουν και μερικές συντομεύσεις πλήκτρων με σκοπό την βελτίωση της εμπειρίας του παίκτη, όπως είναι η πρόσβαση στο μενού, η παράκαμψη διαλόγων ή ακόμα και η εισαγωγή δεδομένων με εικονικό πληκτρολόγιο, με προοπτική μια μελλοντική έκδοση για συσκευές με οθόνη αφής. Στην χρήση των γραφικών έχει επιλεγεί η μέγιστη δυνατή ανάλυση με σκοπό να διασφαλιστεί η επαναχρησιμοποίηση των γραφικών στοιχείων σε μελλοντικές εφαρμογές. Ο ήχος, τέλος, εξυπηρετεί στην ολοκλήρωση του gameplay. Έχουν χρησιμοποιηθεί αποκλειστικά αρχεία τύπου .ogg, καθώς είναι ελεύθερα και υποστηρίζονται από την Visionaire. Στην αξιολόγηση του παιχνιδιού συμμετείχαν 3 αξιολογητές (με παρόμοιο υπόβαθρο σε γνώσεις, εμπειρία, ηλικία) στους οποίους αρχικά δόθηκε το σενάριο του παιχνιδιού και ένα ερωτηματολόγιο. Κατά την αξιολόγηση του παιχνιδιού δεν διαπιστώθηκαν προβλήματα απόδοσης της εφαρμογής ή σοβαρά προβλήματα σχεδιασμού. Τα αποτελέσματα αυτά ήταν πολύ σημαντικά διότι διαφορετικά θα έπρεπε ο συγγραφέας να “επιστρέψει” ξανά πίσω στην φάση του σχεδιασμού και του προγραμματισμού (Κούτρας, 2014).

Η εργασία με τίτλο “Η χρήση οπτικών εργαλείων ανάπτυξης διαδικτυακών παιχνιδιών(online games). Συγκριτική μελέτη δυνατοτήτων και παραδειγματική εφαρμογή χρήσης τους στην κατασκευή ολοκληρωμένου διαδικτυακού παιχνιδιού.” του Βακίρτζη Στυλιανού εξετάζει ένα πλήθος μηχανών παιχνιδιών με σκοπό την ανάπτυξη ενός online παιχνιδιού. Ο συγγραφέας αρχικά εξετάζει την ιστορική διαδρομή των ηλεκτρονικών παιχνιδιών, τις κατηγορίες τους και πώς αυτές εξελίχθηκαν στον χρόνο. Στη συνέχεια ο συγγραφέας αναλύει τα χαρακτηριστικά 16 μηχανών παιχνιδιών. Η πρωταρχική ταξινόμηση αυτών των μηχανών είναι σε οπτικές μηχανές, όπου η ανάπτυξη του παιχνιδιού γίνεται με τη χρήση του ποντικιού και απλές εντολές και σε μη οπτικές μηχανές όπου η ανάπτυξη του παιχνιδιού γίνεται με την παραγωγή κώδικα. Στην πρώτη κατηγορία λοιπόν εξετάζονται οι μηχανές: Construct2, Wimi5, Microsoft XNA game studio, Visual

Studio, Game Maker και Play Canvas, ενώ στην δεύτερη κατηγορία εξετάζονται οι μηχανές ImpactJS, EaseJS, Pixi.js, Turbulenz, melonJS, Wade, Crafty, Enchant.js, Isogenic Engine και Kiwi. Στον συγκριτικό πίνακα οι μηχανές παρουσιάζονται ως προς το κόστος τους, την δημοτικότητά τους, εάν είναι ανοιχτού κώδικα και εάν είναι cross platform. Όσον αφορά την ανάπτυξη του παιχνιδιού στην συγκεκριμένη εργασία, προσωπική επιλογή του συγγραφέα είναι η μηχανή Construct καθ' ότι είναι εύκολη στην χρήση της, διαθέτει φιλικό περιβάλλον, εύχρηστο μενού, documentation και online βοήθεια στην επίσημη ιστοσελίδα της. Στο τελικό στάδιο της εργασίας παρουσιάζεται βήμα προς βήμα η ανάπτυξη του παιχνιδιού και ο σχεδιασμός των αντικειμένων που το απαρτίζουν. Τα συμπεράσματα της εργασίας παρουσιάζουν την αναγκαιότητα της συνεχούς ανάπτυξης των τεχνολογιών που περιβάλλουν τα ηλεκτρονικά παιχνίδια, καθώς αποτελούν μέρος την καθημερινότητας εκατομμυρίων ανθρώπων. Κάτι τέτοιο συνεπάγεται κέρδος και φανερώσει την ευκολία της διαδικασίας ανάπτυξης ενός παιχνιδιού ακόμα και για έναν άνθρωπο με βασικές γνώσεις προγραμματισμού, με τη χρήση των κατάλληλων εργαλείων (Βακιρτζής, 2017).

Τέλος η εργασία “Investigation into the use of HTML 5 game engines to create a responsive social educational game for children” του Pankaj Kashyap (2015) υλοποιεί την ανάπτυξη ενός εκπαιδευτικού παιχνιδιού που απευθύνεται σε μαθητές ηλικίας 5 έως 8 ετών με σκοπό την εκμάθηση και την εμπέδωση των βασικών μαθηματικών πράξεων. Βασικός παράγοντας για την επιλογή της πλατφόρμας ανάπτυξης ήταν το παιχνίδι να είναι λειτουργικό σε οποιαδήποτε ηλεκτρονική συσκευή διαθέτει πρόγραμμα περιήγησης ιστού. Η επιλογή της HTML5 ήταν “μονόδρομος” καθώς έχει αναπτυχθεί με σκοπό ο ίδιος κώδικας να τρέχει σε διάφορους browsers και σε διάφορες συσκευές. Ένα από τα πιο σημαντικά στοιχεία της HTML5 είναι η δυνατότητα των σελίδων της να προσαρμόζονται στα ποικίλα μεγέθη των οθονών. Στον αντίποδα της HTML5 ο συγγραφέας αναφέρει και συγκρίνει την τεχνολογία Flash. Μερικά από τα μειονεκτήματα που αναφέρονται είναι το κόστος ανάπτυξης, καθώς απαιτούνται εργαλεία της Adobe, τα οποία δεν διατίθενται δωρεάν, και η κατάταξη του Flash τελευταίο στις μηχανές αναζήτησης οι οποίες δίνουν έμφαση στο κείμενο, ενώ ο Flash φορτώνει περισσότερες και καθαρότερες εικόνες. Στη συνέχεια της εργασίας ακολουθεί η ανάλυση μερικών από των πιο διάσημων μηχανών ανάπτυξης παιχνιδιών. Οι μηχανές που αξιολογούνται είναι οι : Impact, PlayCanvas, Turbulenz, CoCos2dJS, Game Closure, Gamvas, KIWIJS, CraftyJS, LycheeJS, melonJS. Μέσα από έναν συγκεντρωτικό πίνακα του οποίου τα κριτήρια είναι η υποστήριξη 2D ή 3D γραφικών, η φυσική, οι συγκρούσεις, ο ήχος, το WebGL, οι πλατφόρμες εφαρμογής ,τα πλεονεκτήματα και τα μειονεκτήματα, καταλήγει στην επιλογή της μηχανής Impact για την ανάπτυξη του παιχνιδιού, παρόλο που είναι η μόνη μηχανή για την χρήση της οποίας απαιτείται η αγορά της με κόστος 99 USD. Ο λόγος που επιλέγει η μηχανή Impact είναι διότι συγκεντρώνει όλα τα κριτήρια που είναι απαραίτητα για την ανάπτυξη

του παιχνιδιού και αναφέρονται στον πίνακα καθώς επίσης υπάρχουν διαθέσιμα documentation, online βοήθεια, επιταχύνει την εκτέλεση HTML5 και διαθέτει μηχανισμό εντοπισμού σφαλμάτων (debugging). Η μεθοδολογία της ανάπτυξης του παιχνιδιού βασίζεται στην τεχνογνωσία, στην ανάλυση, τον σχεδιασμό, στην δοκιμή και τέλος στην αξιολόγηση του παιχνιδιού και των αποτελεσμάτων. Αξίζει να αναφερθεί πως η μέθοδος ανάπτυξης που επιλέχθηκε είναι αυτή της διαδικτυακής εφαρμογής που μπορεί να αποθηκευτεί σε απομακρυσμένο διακομιστή και να είναι προσβάσιμη από οποιαδήποτε συσκευή μέσω του περιηγητή ιστού. Τέλος παρουσιάζονται τα συμπεράσματα του συγγραφέα που αφορούν τεχνικά θέματα, όπως πώς να αποφύγει κανείς την “υπερχείλιση” των στοιχείων στις οθόνες, τη μορφή των εικόνων που θα πρέπει να χρησιμοποιηθούν, ή ποιος συνδυασμός CSS-Javascript αποφέρει την καλύτερη απόδοση (Kashyap, 2015).

3. Ανάλυση Μηχανών Παιχνιδιών

3.1 Μηχανή PlayCanvas

3.1.1 Γενικά για την μηχανή PlayCanvas

Η μηχανή ανάπτυξης παιχνιδιών PlayCanvas είναι μια πλατφόρμα οπτικής ανάπτυξης διαδραστικού περιεχομένου ιστού. Η εταιρεία PlayCanvas ιδρύθηκε στο Λονδίνο το 2011 και η πρώτη εμπορική άδεια της μηχανής δόθηκε το 2013. Το όραμα της εταιρείας είναι μια πλατφόρμα ανάπτυξης παιχνιδιών κοινωνικού περιεχομένου με έμφαση στη συνεργασία και την προσβασιμότητα από όλους. Η PlayCanvas είναι μια ισχυρή WebGL μηχανή παιχνιδιού με πολύ μικρό download size και advanced χαρακτηριστικά, η οποία εκτελείται σε όλες τις σύγχρονες desktop και mobile συσκευές που υποστηρίζουν την τεχνολογία WebGL. Με την τεχνολογία Flash να περιθωριοποιείται από το web, η PlayCanvas έχει στόχο να την αντικαταστήσει παρέχοντας ανεπτυγμένα browsers tools για τη δημιουργία interactive 3D περιεχομένου, στηριζόμενη σε σύγχρονες τεχνολογίες, όπως είναι η JavaScript (Delome, 2014).

Κύριο χαρακτηριστικό του ισχυρού PlayCanvas Editor είναι δυνατότητα συνεργασίας που προσφέρει. Αυτό σημαίνει πως τα μέλη μιας ομάδας, είναι ικανά να επεξεργάζονται την ίδια σκηνή σε πραγματικό χρόνο, χωρίς να υπάρχουν προβλήματα συγχώνευσης. Ένα ακόμα χαρακτηριστικό της PlayCanvas που ξεχωρίζει, είναι η cloud based αποθήκευση. Ο προγραμματιστής, έχει πρόσβαση στο project του από οποιαδήποτε συσκευή, σε οποιαδήποτε μέρος του κόσμου και μπορεί να το επεξεργαστεί και να το αποθηκεύσει. Επιπλέον η έκδοση του εκάστοτε project είναι μια εύκολη διαδικασία αφού πρόκειται απλά για ένα URL, το οποίο μπορεί να μοιραστεί στα social media, να ενσωματωθεί σε ήδη υπάρχουσες ιστοσελίδες και να έχουν πρόσβαση σε αυτό εκατομμύρια χρήστες.

Η μηχανή PlayCanvas πέρα από την ανάπτυξη παιχνιδιών, αποτελεί ιδανική λύση για χρήση και σε άλλους κλάδους της οικονομίας, όπως της εικονικής πραγματικότητας, των διαφημίσεων ή ακόμα και της αρχιτεκτονικής. Θεωρείται μια από τις κορυφαίες μηχανές γραφικών και συνεργάζεται άμεσα με όλους τους κύριους browsers της αγοράς για την ανάπτυξη προτύπων του WebVR. Στο διαφημιστικό πλαίσιο, σκοπός είναι η απόκτηση και η διατήρηση των πελατών-παικτών. Οι τεχνικές:

- Playable ads
- Brand Experience
- Digital Out of Home Server

εξυπηρετούν αυτόν τον σκοπό, καθώς προσφέρουν εμπειρίες μοναδικές στον χρήστη, όπως η διαμόρφωση του τελικού προϊόντος με ένα κλικ ή οι υπαίθριες ψηφιακές εμπειρίες (εικόνα 3).

Η PlayCanvas προσφέρει μια καθολική λύση στην ανάπτυξη του κάθε project. Ο σχεδιασμός της μηχανής είναι τέτοιος ώστε οι καλλιτέχνες, οι σχεδιαστές και οι προγραμματιστές να μπορούν να την χρησιμοποιούν με ευκολία και παράλληλα να συνεργάζονται άψογα.

Μερικά από τα χαρακτηριστικά της:

- Physically Based Rendering
- WebVR Support
- Texture Compression
- Real-Time 3D Physics
- 3D Audio
- Lightmapping
- JavaScript
- Animation



Εικόνα 3- Digital Out of Home

Στο Cannes Lions το DOOH.com χρησιμοποίησε την PlayCanvas για να εντυπωσιάσει τα πλήθη, με το μεγαλύτερο παιχνίδι Space Invaders που έγινε ποτέ. Χρησιμοποιώντας μόνο το κινητό τους τηλέφωνο οι παίκτες μπορούσαν να μπουν και να παίξουν.

3.1.2 Υποστηριζόμενοι Browsers

Η μηχανή PlayCanvas σχεδιάστηκε για να “τρέχει” σε κάθε browser. Είναι σχεδιασμένη σε Javascript και HTML5 και αυτό την καθιστά ιδιαίτερα σημαντική καθώς

1. Οι κινητές συσκευές (smartphone/tablet), δεν επιτρέπουν την εγκατάσταση plugins και υποστηρίζουν HTML5
2. Οι εφαρμογές της PlayCanvas φορτώνονται χωρίς την εγκατάσταση plugin
3. Οι εφαρμογές της PlayCanvas τρέχουν σε ένα standard DOM στοιχείο
4. Δίνεται η δυνατότητα σύνθεσης ή τροποποίησης της σελίδας μέσω CSS
5. Δεν υπάρχει εξάρτηση από κάποια τρίτη τεχνολογία, μόνο από τον browser

Πίνακας 1- Απαιτήσεις Προγραμμάτων Περιήγησης

Browser	Win	MacOS X	Linux	Android	iOS
Chrome	✓	✓	✓	✓	
Firefox	✓	✓	✓	✓	✓
IE	✓				
Edge	✓				
Safari		✓			✓

3.1.3 Κόστος μηχανής PlayCanvas

Η μηχανή PlayCanvas διατίθεται σε τρεις εκδόσεις. Η πρώτη έκδοση είναι δωρεάν και προσφέρει απεριόριστο αριθμό δημοσίων project, όλα τα λειτουργικά χαρακτηριστικά της μηχανής και δωρεάν φιλοξενία. Η δεύτερη έκδοση απευθύνεται σε μεμονωμένους χρήστες και προσφέρει απεριόριστο αριθμό ιδιωτικών project και εξαγωγή των project για self-hosting. Τέλος, η τρίτη

έκδοση απευθύνεται σε οργανισμούς και επιχειρήσεις και προσφέρει επιπλέον εργαλεία διαχείρισης ομάδας και τη δυνατότητα αφαίρεσης του λογότυπου PlayCanvas. Στον Πίνακα 2 που ακολουθεί αναφέρονται όλα τα βασικά χαρακτηριστικά της κάθε έκδοσης, καθώς επίσης και το κοστολόγιό τους.

Πίνακας 2- Συγκριτικό τιμολόγιο της μηχανής Play Canvas

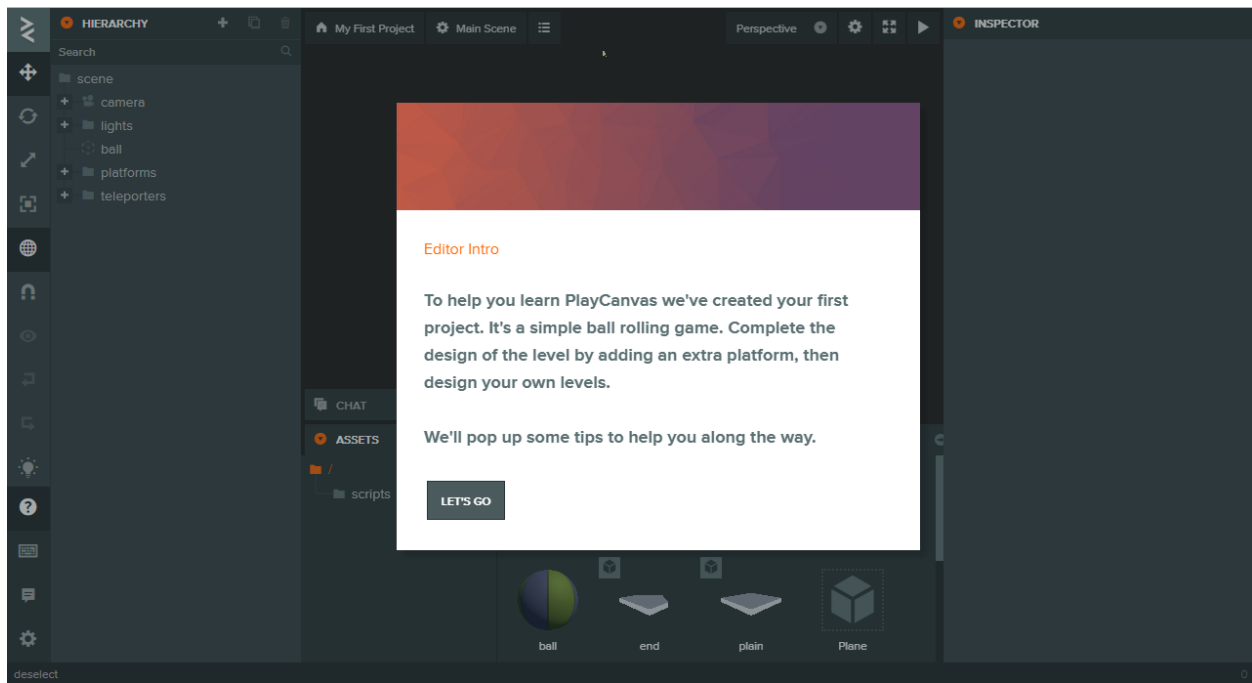
Χαρακτηριστικά	Δωρεάν Χρήση	Personal Χρήση	Επιχειρηματική Χρήση
Χώρος Αποθήκευσης	200 MB	1 GB	10 GB
Απεριόριστα Public Projects	NAI	NAI	NAI
Δωρεάν Φιλοξενία στο PlayCanvas	NAI	NAI	NAI
Απεριόριστα Private Projects	OXI	NAI	NAI
Εξαγωγή σε HTML για self-hosting	OXI	NAI	NAI
Αφαίρεση λογότυπου PlayCanvas	OXI	OXI	NAI
Offline Αποθήκευση & Επαναφορά	OXI	OXI	NAI
Δυνατότητα επεξεργασίας από ομάδες	OXI	OXI	NAI
Πρόσβαση στο REST API	OXI	OXI	NAI
Έκδοση Τιμολογίου	OXI	OXI	NAI
Κόστος ανά Μήνα	—	15\$	50\$

3.1.4 Τρόπος λειτουργίας της μηχανής PlayCanvas

Η επίσημη ιστοσελίδα της μηχανής PlayCanvas διαθέτει πολλά tutorials, user manuals και ένα ενεργό forum, με σκοπό την καλύτερη κατανόηση και χρήση της μηχανής, τόσο από τους επαγγελματίες προγραμματιστές, όσο και από ερασιτέχνες που θέλουν να ασχοληθούν με την ανάπτυξη ενός παιχνιδιού. Για να ξεκινήσει κανείς την χρήση της μηχανής PlayCanvas θα πρέπει αρχικά να δημιουργήσει λογαριασμό. Η διαδικασία αυτή είναι απλή και οι πληροφορίες που απαιτούνται είναι το email του χρήστη και ένας κωδικός (<https://playcanvas.com/>).

Εικόνα 4- Sign up PlayCanvas

Εάν ο χρήστης διαθέτει ήδη λογαριασμό επιλέγει το εικονίδιο LOGIN



Εικόνα 5- Demo νέου χρήστη PlayCanvas

Με την δημιουργία του λογαριασμού, αυτόματα δημιουργείται και ένα demo project για να δοθεί στον χρήστη μια πρώτη γεύση από τον PlayCanvas Editor. Το demo αυτό βρίσκεται στην σελίδα Profile του χρήστη.

3.1.4.1 Profile

Όταν ολοκληρωθεί η διαδικασία εγγραφής/ εισόδου, αυτόματα ο χρήστης μεταφέρεται στο περιβάλλον Profile. Το Profile λειτουργεί όπως οι περισσότερες παρεμφερείς σελίδες διατηρώντας τις βασικές πληροφορίες του χρήστη. Η πρώτη εικόνα που λαμβάνει ο χρήστης όταν επισκέπτεται το Profile είναι ένας πίνακας που αποτελείται από τρία στοιχεία: HOME-PROJECTS-ACCOUNT [Εικόνα 6]. Στο πεδίο Account βρίσκονται τα στοιχεία του λογαριασμού που ο χρήστης μπορεί να επεξεργαστεί. Στο πεδίο Projects φιλοξενούνται τα projects που δημιουργεί ο χρήστης και τέλος στο πεδίο Home βρίσκει κανείς τα πρόσφατα σχόλια που έχουν δημοσιευτεί στα tutorials της ιστοσελίδας και δίνετε η δυνατότητα να ελέγξει την πρόσφατη δραστηριότητά του. Τέλος, βρίσκονται links που οδηγούν στο forum, στο blog και στα documentations της επίσημης ιστοσελίδας.



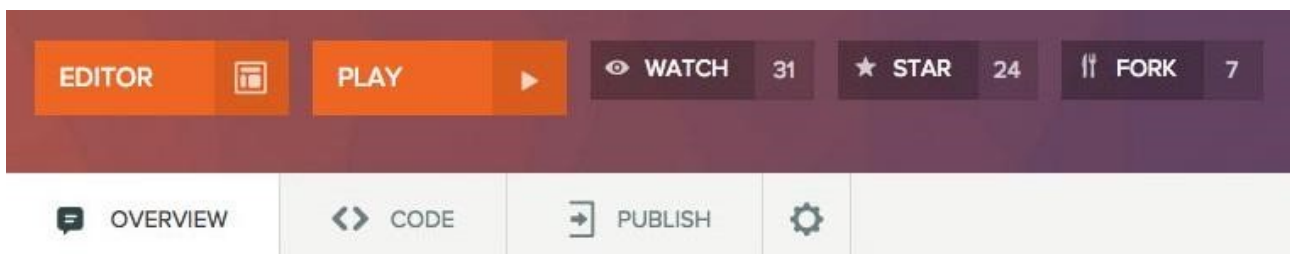
Εικόνα 6- Profile PlayCanvas 1

3.1.4.2 Projects

Όπως έχει αναφερθεί και στην προηγούμενη ενότητα, στο πεδίο Projects φιλοξενούνται τα projects που δημιουργεί ο χρήστης. Από αυτό το πεδίο ο χρήστης είναι σε θέση να επεξεργαστεί τα projects του και εφόσον το project είναι public να το δουν και άλλοι χρήστες της PlayCanvas. Πρόκειται ουσιαστικά για τον Πίνακα Ελέγχου από όπου δίνεται η δυνατότητα στον χρήστη να:

- Εισέλθει στον PlayCanvas Editor
- Εκτελέσει/ τρέξει το project
- Ελέγξει το αρχείο καταγραφής
- Ελέγξει τη δραστηριότητα του project
- Επεξεργαστεί τις ρυθμίσεις

Πιο αναλυτικά, στην κεφαλίδα του project [Εικόνα 7], παρέχονται οι εξής επιλογές, για την γρήγορη εκτέλεση των ενεργειών:



Εικόνα 7- Overview Header PlayCanvas

EDITOR: Άνοιγμα του PlayCanvas Editor.

PLAY: Ο επισκέπτης μπορεί να παίζει το Primary Build του έργου.

WATCH: Ο επισκέπτης μπορεί να “ακολουθήσει” το έργο και να λαμβάνει ενημερώσεις για νέες δημοσιεύσεις σχετικά με το έργο και ο χρήστης μπορεί να ελέγχει την επισκεψιμότητα του έργου

STAR: Ο επισκέπτης έχει την δυνατότητα να αξιολογήσει το έργο με αστεράκι. Η δυνατότητα παρακολούθησης της δημοφιλίας του έργου, παρέχει κίνητρα στον δημιουργό για να συνεχίσει να εξελίσει το παιχνίδι.

FORK: Ο επισκέπτης με αυτή την επιλογή μπορεί να δημιουργήσει έναν κλώνο του project στον λογαριασμό του.

Στις Ρυθμίσεις του project ο χρήστης έχει τη δυνατότητα να επεξεργαστεί το όνομα και την περιγραφή του έργου, να προσθέσει φωτογραφία 720x720 που θα αντιπροσωπεύει το έργο (η οποία είναι απαραίτητη εάν ο χρήστης επιθυμεί να εμφανίζεται το παιχνίδι στο playcanvas.com), να προσθέσει άλλους χρήστες στην ομάδα του και τέλος να διαγράψει το έργο (η ενέργεια διαγραφής δεν μπορεί να ανακληθεί).

3.1.4.3 Editor

Ο PlayCanvas Editor είναι ο επεξεργαστής στον οποίο ο χρήστης δημιουργεί και τροποποιεί τις σκηνές και τις οντότητες που αποτελούν το παιχνίδι του. Λειτουργεί στον browser χωρίς να απαιτείται κάποια επιπλέον εγκατάσταση και αυτό έχει σαν αποτέλεσμα να είναι διαθέσιμος όπου και αν βρίσκεται ο χρήστης. Διαθέτει ζωντανή επεξεργασία, που σημαίνει ότι ο χρήστης μπορεί να επέμβει στο παιχνίδι ενώ αυτό εκτελείται.

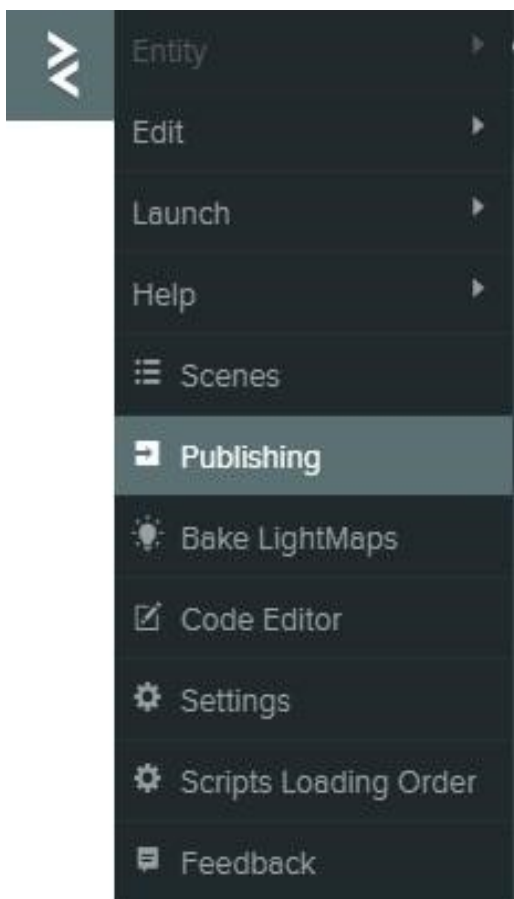


Εικόνα 8- PlayCanvas Interface

Με μία πρώτη ματιά στον PlayCanvas Editor βλέπει κανείς πως η οθόνη χωρίζεται σε 4 σημεία:

1. Menu- Toolbar- Hierarchy
2. Assets
3. Viewport
4. Inspector

Το Menu περιέχει μια πλήρη λίστα με όλες τις διαθέσιμες εντολές που μπορεί να εκτελέσει κανείς στην σκηνή του. Το μενού είναι διαθέσιμο κάνοντας κλικ στο εικονίδιο PLAYCANVAS. [Εικόνα 9]. Το Toolbar διαθέτει εν συντομία τις συνήθεις εντολές για εύκολη και γρήγορη πρόσβαση [Εικόνα 10].



Εικόνα 9- PlayCanvas Menu

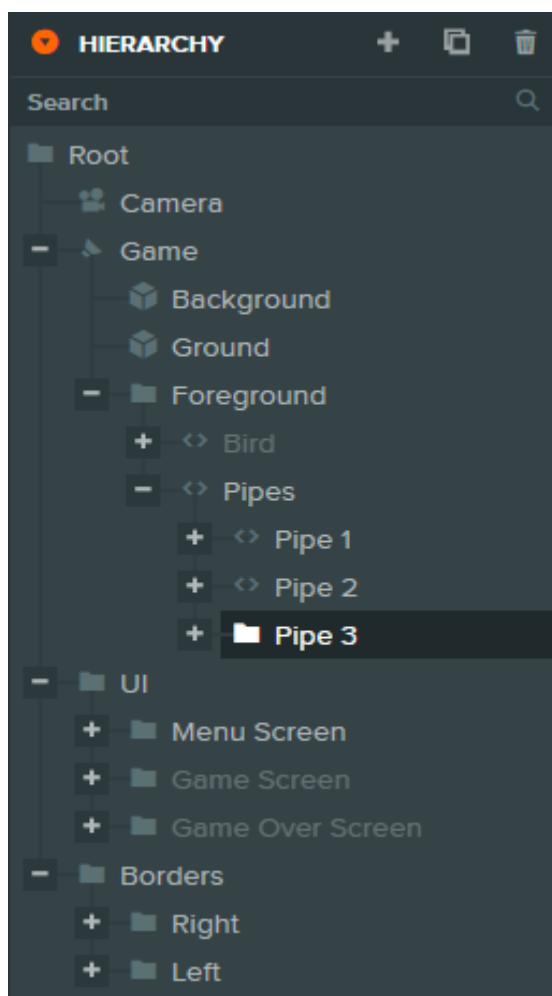


Εικόνα 10- PlayCanvas Toolbar

Ενώ το menu και το toolbar είναι σταθερής μορφής σε οποιοδήποτε project, η ενότητα του Hierarchy διαφέρει καθώς διαμορφώνεται κατά την εξέλιξη του εκάστοτε project [Εικόνα 11]. Σε αυτήν την ενότητα βρίσκονται όλες οι οντότητες που αποτελούν την τρέχουσα σκηνή,

ταξινομημένες ιεραρχικά σε δέντρο. Χάρη σε αυτόν τον ιεραρχημένο πίνακα, η δημιουργία, η εύρεση, η διαγραφή και η οργάνωση των οντοτήτων της σκηνής απλοποιείται και εξοικονομείται χρόνος. Οι οντότητες είναι ορατές συνεχώς και με ένα απλό κλικ γίνεται η επιλογή της κάθε μιας. Εφόσον ο χρήστης έχει επιλέξει μια οντότητα μπορεί πολύ εύκολα να δημιουργήσει μια νέα υποενότητα της ή απλά να την διαγράψει είτε χρησιμοποιώντας τον κάδο ανακύκλωσης είτε με δεξί κλικ.

Η οργάνωση σε ιεραρχικό δέντρο είναι πολύ σημαντική διότι διευκολύνεται η επεξεργασία των οντοτήτων αλλά έχει ιδιαίτερη σημασία η σειρά με την οποία ταξινομούνται, καθώς ισχύει η σχέση γονέας-παιδί. Μία οντότητα παιδί, κληρονομεί τα χαρακτηριστικά του γονέα. Εάν λοιπόν κατά τη διάρκεια της επεξεργασίας, οι ιδιότητες του γονέα αλλάξουν ή ο γονέας μετακινηθεί, ή ακόμα διαγραφεί, τότε τις ίδιες αλλαγές θα ισχύσουν και για όλες τις οντότητες-παιδιά. Για αυτόν τον λόγο η δομή του δέντρου πρέπει να διατηρείται σε όσο το δυνατό απλούστερη μορφή. Τέλος, οι επιλογές αντιγραφή, επικόλληση μπορεί να πραγματοποιηθούν και ανάμεσα σε οντότητες που ανήκουν σε διαφορετικές σκηνές ή ακόμα και σε διαφορετικά projects.

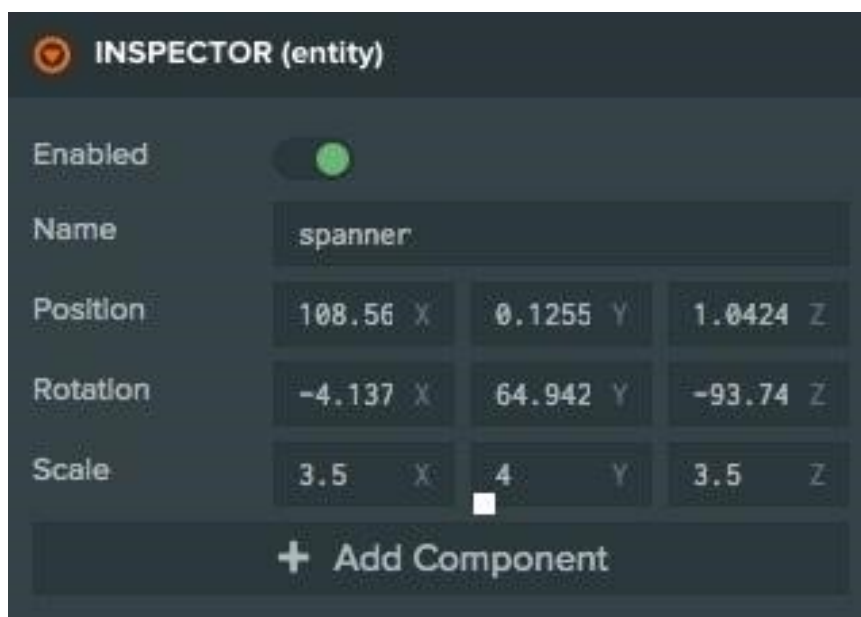


Εικόνα 11- PlayCanvas Hierarchy

Στην ενότητα Inspector εμφανίζονται οι τιμές των χαρακτηριστικών για το τρέχον επιλεγμένο στοιχείο. Ορισμένες από τις διαθέσιμες επιλογές είναι:

- Inspector οντοτήτων/ συστατικών στοιχείων
- Inspector υφής
- Inspector υλικών
- Inspector Cubemap

Με την τροποποίηση των τιμών στην ενότητα Inspector καθορίζεται η συμπεριφορά των οντοτήτων [Εικόνα 12].



Εικόνα 12- PlayCanvas Inspector

Στο κέντρο της οθόνης εμφανίζεται το παράθυρο προβολής της σκηνής ή αλλιώς το Viewport. Στο Viewport υπάρχει μια 3D απεικόνιση της σκηνής, πάνω στην οποία τοποθετούνται οι οντότητες. Μέσα σε αυτή την απεικόνιση ο χρήστης μετακινείται χάρη στην κάμερα του Editor. Ο χειρισμός της κάμερας ανάλογα με την γωνία εστίασης που επιθυμεί ο χρήστης, έχει τις εξής επιλογές:

- Perspective
- Top
- Bottom
- Front
- Back

- Left
- Right
- SplashCamera
- Camera

Gizmos

Κάθε χαρακτηριστικό κατά την επεξεργασία του διαθέτει τρεις άξονες. Οι τρεις χρωματισμένοι άξονες ονομάζονται Gizmo [Εικόνα 13]. Αυτοί οι άξονες χρησιμοποιούνται για τον χειρισμό των οντοτήτων εκ των έσω. Υπάρχουν 3 τύποι:

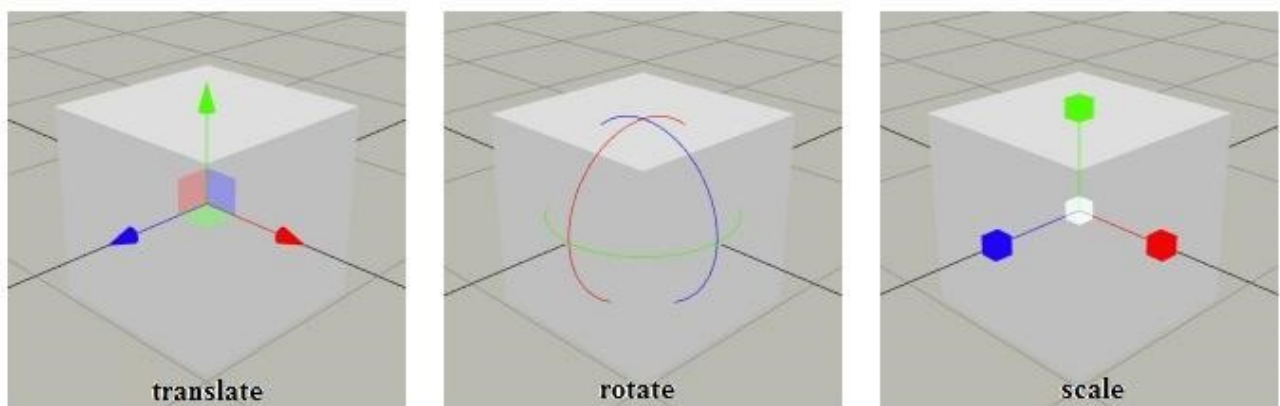
- Translate- με βέλη στα άκρα των αξόνων
- Rotate- δακτύλιοι
- Scale- με κύβους στα άκρα των αξόνων

Πρόκειται ουσιαστικά για τον έλεγχο που μπορεί να ασκηθεί με το ποντίκι για την επεξεργασία των χαρακτηριστικών μιας οντότητας. Αποτελείται όπως έχει αναφερθεί από τρεις άξονες που αντιστοιχούν στους άξονες του 3D χώρου.

Άξονας X: κόκκινο

Άξονας Y: πράσινο

Άξονας Z: μπλε

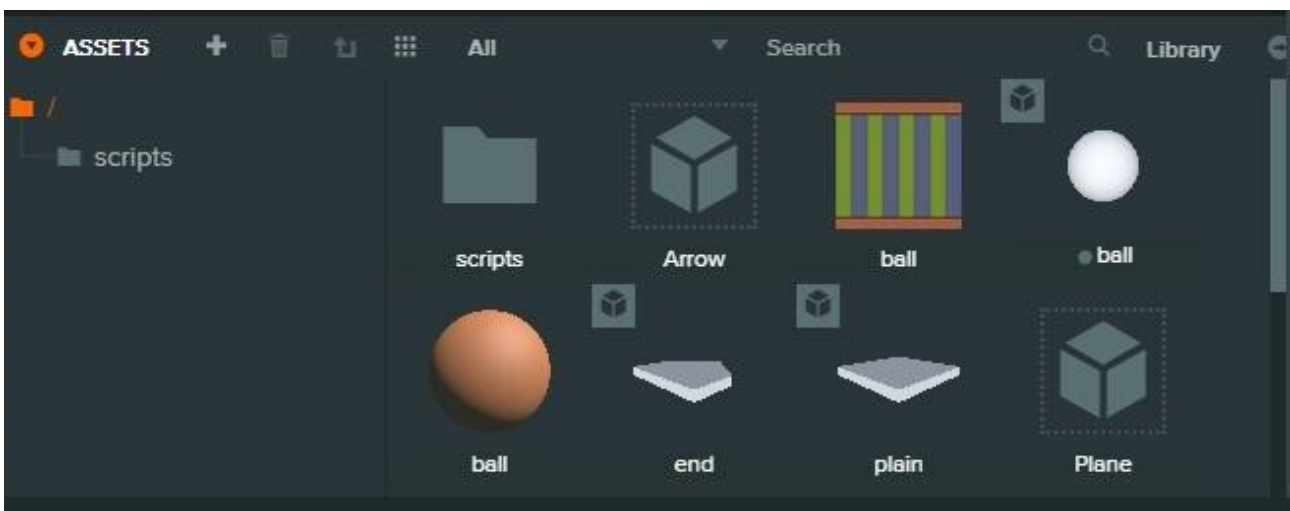


Εικόνα 13- PlayCanvas Gizmos

Με την επιλογή translate επιτρέπεται ο χειρισμός δυο διαστάσεων ταυτόχρονα, στα επίπεδα X και Y, Y και Z, Z και X.

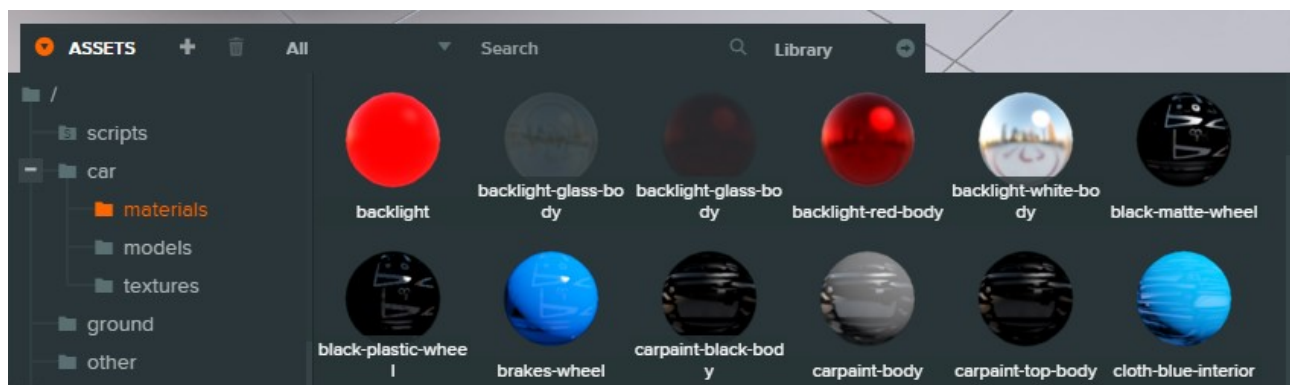
Στην ενότητα Assets βρίσκει κανείς όλες τις οντότητες, όλα τα στοιχεία που υπάρχουν στο project. Ο χρήστης είναι ικανός να οργανώσει τα στοιχεία σε ένα δενδροδιάγραμμα και διατίθενται όλες τις απαραίτητες λειτουργίες για την συντήρησή του, όπως είναι η δημιουργία φακέλου, η μετονομασία, η μετακίνηση και η διαγραφή. Μια σημαντική λειτουργία της PlayCanvas, μέσα από την ενότητα Assets είναι η δημιουργία των στοιχείων με δυο τρόπους. Ο πρώτος τρόπος είναι μέσα

από τον Editor. Ο δεύτερος τρόπος είναι με την μεταφορά αρχείων από τον υπολογιστή στην ενότητα Assets, όπου ο επεξεργαστής φορτώνει και εισάγει τα δεδομένα που έχει ζητήσει ο χρήστης. Κάθε οντότητα του παιχνιδιού διαθέτει ένα μοναδικό αναγνωριστικό ID, το οποίο μπορεί ο χρήστης να το χρησιμοποιήσει στην αναζήτησή του. Η αναζήτηση επίσης είναι δυνατή και με την χρήση κανονικών εκφράσεων, με το σύμβολο *. Εάν ο χρήστης σύρει ένα στοιχείο από την ενότητα Assets στο Viewport, τότε αυτόματα θα δημιουργηθεί μια νέα οντότητα με τα χαρακτηριστικά του στοιχείου. Μια επιπλέον λειτουργία της ενότητας Assets είναι η αναφορά των στοιχείων ώστε να γνωρίζει ο χρήστης που χρησιμοποιούνται. Εάν ο επεξεργαστής δεν εντοπίσει αναφορά για κάποιο στοιχείο, τότε θα εμφανιστεί μια μικρή κουκίδα στην μικρογραφία του. Αξίζει να σημειωθεί πως ο Editor δεν μπορεί να εντοπίσει αναφορές στοιχείων που έχουν δημιουργηθεί μέσω κώδικα. Τέλος, ο χρήστης μπορεί να κατεβάσει δωρεάν 3D μοντέλα από το PlayCanvas Store.



Εικόνα 14- PlayCanvas Assets (1)

Η εικόνα της ενότητας Assets στο Demo project



Εικόνα 15- PlayCanvas Assets (2)

Η εικόνα της ενότητας Assets σε ολοκληρωμένο project.

3.2 Μηχανή Construct 3

3.2.1 Γενικά για την μηχανή Construct 3

Η μηχανή ανάπτυξης παιχνιδιών Construct απευθύνεται κυρίως σε απλούς χρήστες, μη προγραμματιστές, επιτρέποντας τη γρήγορη δημιουργία παιχνιδιών με το σύστημα drag and drop στο περιβάλλον του χρήστη. Η πρώτη έκδοση της Construct Classic πραγματοποιήθηκε στις 27 Οκτωβρίου του 2007, ενώ η έκδοση της Construct 2 στις 20 Απριλίου του 2013. Στις 27 Ιανουαρίου του 2015 στο επίσημο blog της εταιρείας, ανακοινώθηκε η πρώτη έκδοση της Construct 3. Οι νέες λειτουργίες της μηχανής περιλαμβάνουν την χρήση σε συστήματα Mac και Linux, την πολυγλωσσική υποστήριξη και την χρήση της σε κινητές συσκευές Android και iOS. Για την λειτουργία της μηχανής δεν χρειάζεται η εγκατάσταση κάποιου λογισμικού. Εκτελείται στο πρόγραμμα περιήγησης και ενημερώνεται αυτόματα στην πιο πρόσφατη έκδοσή της. Έτσι εξοικονομούνται υπολογιστικοί πόροι στο σύστημα του κάθε χρήστη, καθώς και ο χρόνος που απαιτείται τόσο για την εγκατάσταση του λογισμικού όσο και την εγκατάσταση των διάφορων ενημερώσεων. Η μηχανή Construct 3 δίνει στον χρήστη την επιλογή να αποθηκεύσει το project του είτε τοπικά είτε απευθείας σε cloud, με την χρήση του Dropbox, του Google Drive ή του OneDrive. Η επιλογή Cloud Save της μηχανής είναι εξαιρετικά χρήσιμη καθώς επιτρέπει στον χρήστη να αποθηκεύει και να διαχειρίζεται το project του απευθείας σε κάποιον λογαριασμό Dropbox, Google Drive και One Drive και έτσι καθιστά την εργασία του διαθέσιμη σε αυτόν σε οποιοδήποτε μέρος και δεδομένης της υποστήριξης της μηχανής σε πολλαπλές πλατφόρμες βοηθά επιπλέον να επεξεργάζεται το έργο του σε οποιαδήποτε συσκευή.

Ενδιαφέρον παρουσιάζει ο τρόπος παρουσίασης και προώθησης της μηχανής Construct 3 μέσω της επίσημης ιστοσελίδας της, καθώς απευθύνεται κυρίως στην εκπαιδευτική κοινότητα. Χρησιμοποιώντας την δημιουργία παιχνιδιών ως εργαλείο για την εκπαίδευση επιτυγχάνεται ο στόχος, οι μαθητές να αποκτήσουν ένα ευρύ φάσμα δεξιοτήτων και ικανοτήτων όπως η:

- Εκμάθηση βασικών αρχών προγραμματισμού
- Συνεργασία σε ομάδες
- Πρωτότυπες ιδέες
- Ενασχόληση με το Marketing

Η μηχανή Construct 3 θεωρείται ιδανική τόσο για παιδιά ηλικίας ακόμα και Δημοτικού όσο και για την εκπαίδευση ενηλίκων.

Χαρακτηριστικό της Construct 3, όπως είναι και της Construct 2, είναι πως η ροή του project γίνεται σε φύλλα γεγονότων (event sheets). Αυτό το χαρακτηριστικό επιτρέπει την εκμάθηση του προγραμματισμού με έναν κατανοητό τρόπο για όλους. Η μηχανή Construct 3 δίνει την επιλογή σε πιο προχωρημένους και έμπειρους χρήστες να ξεφύγουν από τα όρια της και να

επεκτείνουν το project τους γράφοντας τα δικά τους plugins με την χρήση της Javascript. Εφόσον η γλώσσα προγραμματισμού Javascript είναι μια ευρέως διαδεδομένη και εμπορική γλώσσα, καθιστά την Construct 3 προσιτή σε εκατομμύρια χρήστες.

Το πιο σημαντικό χαρακτηριστικό της μηχανής είναι το γεγονός ότι ο χρήστης είναι ικανός να σχεδιάσει διδιάστατα παιχνίδια με περισσότερη έμφαση στα platform παιχνίδια και στα arcade παιχνίδια.

Η εξέλιξη της μηχανής Construct 3 μέσα σε ένα χρόνο από την έκδοσή της είναι κάτι παραπάνω από αλματώδης και αυτό φαίνεται από διάφορα στοιχεία:

- Χρήση σε 210 χώρες
- Διατίθεται σε 5 γλώσσες
- Βρίσκονται σε εξέλιξη οι μεταφράσεις σε άλλες 12 γλώσσες
- Υπάρχουν πάνω από 400 ενημερώσεις χωρίς να περιλαμβάνονται σε αυτό τον αριθμό οι διορθώσεις σφαλμάτων
- Ενεργή κοινότητα χρηστών (Forum και Blog)
- Tutorials
- Manuals & Documentation

Όλα αυτά καθιστούν την μηχανή Construct 3 μια δυναμική παρουσία στον χώρο των game engines με πολλές προοπτικές εξέλιξης και εδραίωσης στον χώρο.

3.2.2 Υποστηριζόμενοι Browsers Construct 3

Οι απαιτήσεις συστήματος για την λειτουργία της Construct 3 αναφέρονται στον παρακάτω πίνακα:

Πίνακας 3- Απαιτήσεις Προγραμμάτων Περιήγησης

Browser	Win7 ή νεότερο	MacOS X	Linux	Android \geq 1 GB RAM	IOS 11
Chrome	✓	✓	✓	✓	
Firefox	✓	✓	✓	✓	✓
IE					
Edge	✓				
Safari		✓			✓

Μια σημαντική απαίτηση ακόμα, είναι το πρόγραμμα περιήγησης να υποστηρίζει την τεχνολογία WebGL. Ένα επιπλέον χαρακτηριστικό της Construct 3 είναι η χρήση της χωρίς την

σύνδεση στο διαδίκτυο, με την προϋπόθεση ότι έχει προηγηθεί έστω μια φορά η ενασχόληση με την μηχανή όταν ο χρήστης ήταν συνδεδεμένος στο διαδίκτυο.

3.2.3 Κόστος μηχανής Construct 3

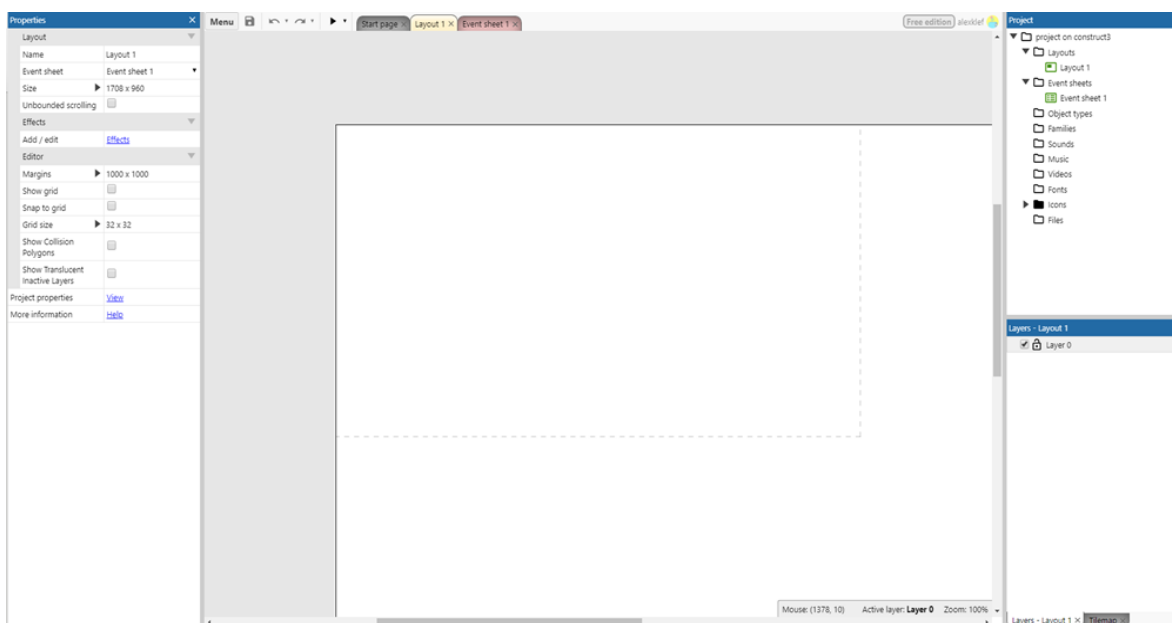
Η μηχανή Construct 3 για να καλύψει την εμπειρία του κάθε χρήστη που θέλει να ασχοληθεί με την ανάπτυξη παιχνιδιών, διαθέτει τέσσερα πακέτα πληρωμής. Το πρώτο και δωρεάν πακέτο, με αρκετά περιορισμένες ιδιότητες απευθύνεται σε περιστασιακούς χρήστες. Η επόμενη έκδοση της Construct 3, η Personal, απευθύνεται σε άτομα που ασχολούνται με την δημιουργία παιχνιδιών ως χόμπι. Η έκδοση Business απευθύνεται σε επιχειρήσεις και οργανισμούς. Η συγκεκριμένη έκδοση διαθέτει δυο άδειες χρήσης, την StartUp και την Standard. Για να αγοράσει μια επιχείρηση την StartUp άδεια θα πρέπει το ετήσιο εισόδημα να είναι μικρότερο των 50.000\$ και τα καθαρά περιουσιακά της στοιχεία λιγότερα των 100.000\$. Εάν οποιοδήποτε από αυτά τα όρια ξεπεραστεί τότε η επιχείρηση θα πρέπει να προχωρήσει στην αγορά της Standard άδειας χρήσης. Η έκδοση Educational απευθύνεται αποκλειστικά και μόνο σε εκπαιδευτικούς οργανισμούς και δίνεται η δυνατότητα η άδεια να αγοραστεί είτε με το μήνα είτε με το χρόνο. Τέλος σημειώνεται πώς τα κόστη που αναφέρονται στο συγκριτικό τιμολόγιο που ακολουθεί αφορούν μια θέση και πως υπάρχει η επιλογή μέχρι εκατό θέσεων.

Πίνακας 4- Συγκριτικό τιμολόγιο της μηχανής Construct 3

Χαρακτηριστικά	Δωρεάν	Personal	Επιχειρηματική	Εκπαιδευτική
Μέγιστος αριθμός Events	40	Χωρίς Όριο	Χωρίς Όριο	Χωρίς Όριο
Μέγιστος αριθμός Επιπέδων	2	Χωρίς Όριο	Χωρίς Όριο	Χωρίς Όριο
Χρήση Ειδικών Εφέ	2	Χωρίς Όριο	Χωρίς Όριο	Χωρίς Όριο
Παιχνίδια Multiplayer	OXI	NAI	NAI	NAI
Debugger	OXI	NAI	NAI	NAI
Αποθήκευση Cloud	NAI	NAI	NAI	NAI
Έκδοση στο Web	NAI	NAI	NAI	NAI
Έκδοση σε άλλες Πλατφόρμες	OXI	NAI	NAI	NAI
Κόστος ανά Έτος	—	99\$	149\$-399\$	12,99\$-39,99\$

3.2.4 Τρόπος λειτουργίας της μηχανής Construct 3

Ακολουθώντας την λογική και των υπόλοιπων online μηχανών παιχνιδιών ο χρήστης μπορεί να ξεκινήσει να χρησιμοποιεί την Construct 3 αφού ολοκληρώσει την εγγραφή του και την σύνδεσή του στο σύστημα. Ωστόσο υπάρχει η δυνατότητα χρήσης της μηχανής, χωρίς την εγγραφή, απλά σαν guest. Αυτό όμως περιορίζει τις ήδη μειωμένες επιλογές της free έκδοσης, όπως για παράδειγμα ο αριθμός των events που μπορούν να χρησιμοποιηθούν. Ο εγγεγραμμένος χρήστης μπορεί να χρησιμοποιήσει μέχρι 50 events, ενώ ο guest μόλις 25.



Εικόνα 16-Το περιβάλλον της Construct 3

Με την επιλογή new project, εμφανίζεται η παραπάνω οθόνη. Η Construct 3 παρουσιάζει ένα απλό σχετικά User Interface, χωρίς κάτι το εξεζητημένο. Για έναν χρήστη ο οποίος έχει ήδη ασχοληθεί με μερικές μηχανές παιχνιδιών, θα αντικρίσει ένα οικείο και φιλικό περιβάλλον. Το περιβάλλον αυτό χωρίζεται αρχικά σε τέσσερα τμήματα.

1) Properties

Στο τμήμα Properties εμφανίζονται σε λίστα όλες οι ιδιότητες του επιλεγμένου στοιχείου κάθε φορά και από αυτή τη λίστα ο χρήστης μπορεί να ρυθμίσει τις όποιες αλλαγές επιθυμεί. Υπάρχουν πολυάριθμες ιδιότητες στην μηχανή Construct 3 ανάλογα με το στοιχείο που είναι επιλεγμένο. Για την διευκόλυνση του χρήστη, όλες οι ιδιότητες έχουν μια περιγραφή που παρέχει τις απαραίτητες πληροφορίες και χρήσιμες συμβουλές για το πώς χρησιμοποιείται η κάθε ιδιότητα. Αυτή η περιγραφή εμφανίζεται αφού ο χρήστης έχει επιλέξει την ιδιότητα που τον ενδιαφέρει, στην κάτω αριστερά γωνία της οθόνης.

Show Collision Polygons	<input type="checkbox"/>
Show Translucent Inactive Layers	<input type="checkbox"/>
Project properties	View
More information	Help

Show Collision Polygons: Show the outlines of collision polygons in the layout view.

Εικόνα 17- Επεξήγηση ιδιοτήτων

2) Projects

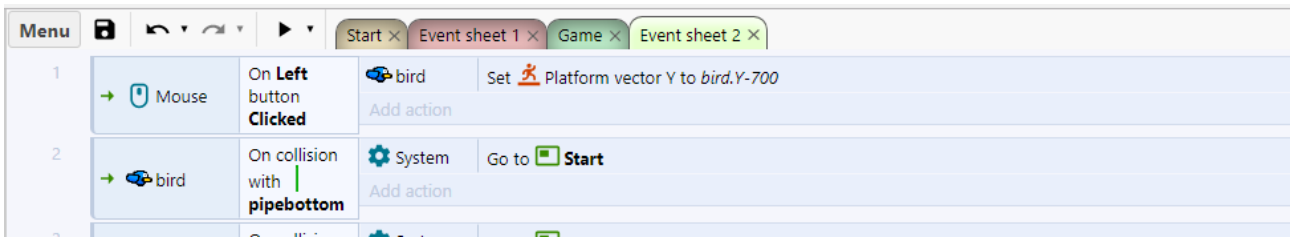
Το τμήμα Projects περιλαμβάνει όλα τα στοιχεία και αντικείμενα του παιχνιδιού και δίνει μια συνολική εικόνα του έργου, δίνοντας τη δυνατότητα στον χρήστη να περιηγηθεί μέσα σε αυτό. Εκτός από την συνοπτική παρουσίαση του έργου, ο χρήστης μπορεί να οργανώσει τα στοιχεία του παιχνιδιού σε φακέλους και υποφακέλους για να τα διαχειρίζεται ομαδικά, με βασική προϋπόθεση τα στοιχεία του κάθε φακέλου να είναι του ίδιου τύπου, πχ αρχεία εικόνων. Σημαντικό επίσης είναι πως εάν ο χρήστης θέλει να διαγράψει ένα στοιχείο οριστικά από το παιχνίδι του, μπορεί να το κάνει μόνο μέσα από την ενότητα Projects.

3) Layers- Layout

Το τμήμα αυτό χρησιμοποιείται για την προσθήκη, την επεξεργασία και την διαγραφή των “επιπέδων” που βρίσκονται μέσα σε μια σειρά. Ένα επίπεδο επιτρέπει την εύκολη διάταξη των αντικειμένων που προβάλλονται μπροστά από άλλα αντικείμενα, όπως για παράδειγμα φαίνονται τα αντικείμενα που βρίσκονται στο προσκήνιο μπροστά από την εικόνα του background. Η σειρά μέσα στην λίστα έχει ιδιαίτερη σημασία, καθώς τα επίπεδα που βρίσκονται στο επάνω μέρος της λίστα εμφανίζονται μπροστά, ενώ τα επίπεδα που βρίσκονται στο κάτω μέρος της λίστας, εμφανίζονται στο πίσω μέρος, όπως το background. Η επιλογή ενός επιπέδου της λίστας αυτής από τον χρήστη, το ενεργοποιεί ως προς επεξεργασία και εισαγωγή νέων αντικειμένων.

4) Main View

Το κεντρικό και μεγαλύτερο μέρος της οθόνης καταλαμβάνει το τμήμα στο οποίο σχεδιάζεται το παιχνίδι. Με την χρήση καρτελών ο χρήστης επεξεργάζεται παράλληλα και τα επίπεδα του παιχνιδιού αλλά και τα συμβάντα.

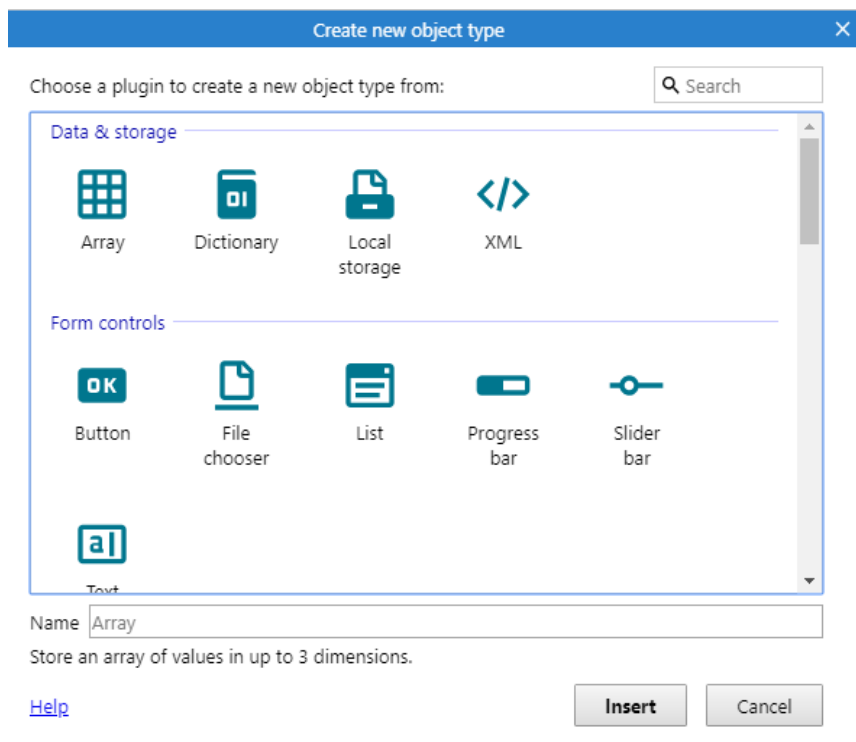


Εικόνα 18- Οι καρτέλες του παιχνιδιού

Επεξήγηση καρτελών του Main View:

- **Layout**

Η προβολή διάταξης είναι η καρτέλα στην οποία ο χρήστης σχεδιάζει τα αντικείμενα του παιχνιδιού και ρυθμίζει την διάταξή τους. Στην καρτέλα αυτή, ο χρήστης θα δημιουργήσει τα επίπεδα του παιχνιδιού ή το μενού του παιχνιδιού. Δεν είναι όμως όλη η επιφάνεια διαθέσιμη για την σχεδίαση της εικόνας που θα βλέπει ο παίκτης. Το διακεκομμένο ορθογώνιο στο πάνω αριστερά μέρος της περιοχής υποδεικνύει το μέγεθος της προβολής δηλαδή το viewport. Για επιπλέον βοήθεια προς τον χρήστη, στην κάτω αριστερή γωνία της καρτέλας εμφανίζεται το τρέχον σημείο του ποντικιού, το ποσοστό της μεγέθυνσης και το όνομα του επιπέδου που είναι ενεργό και επεξεργάζεται εκείνη τη στιγμή. Η εναλλαγή των επιπέδων γίνεται από το τμήμα Layers- Layout. Στη συνέχεια, η δημιουργία νέων αντικειμένων γίνεται πολύ εύκολα, είτε με αριστερό διπλό κλικ είτε με δεξί κλικ, όπου θα εμφανιστεί το αντίστοιχο παράθυρο διαλόγου για να επιλέξει ο χρήστης το αντικείμενο που επιθυμεί.

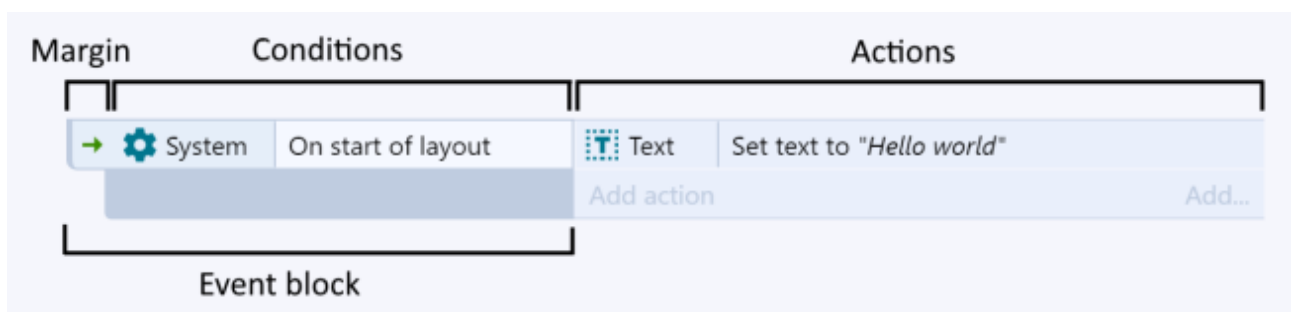


Εικόνα 19- Παράθυρο επιλογής αντικειμένου

Δημιουργώντας λοιπόν ένα νέο αντικείμενο και επιλέγοντάς το, αυτόματα στο τμήμα Properties θα εμφανιστούν οι ιδιότητες που αντιστοιχούν σε αυτό. Επιπρόσθετα, κάνοντας δεξί κλικ κατευθείαν πάνω στο αντικείμενο, εμφανίζεται ένα νέο μενού με κύριες επιλογές, την αντιγραφή, την διαγραφή, την προσθήκη μεταβλητών ή συμπεριφορών.

- **Event Sheet**

Στην καρτέλα των συμβάντων βρίσκεται η ουσιαστική διαφορά της φιλοσοφίας της Construct έναντι των άλλων μηχανών που υποστηρίζουν την ανάπτυξη παιχνιδιών με την χρήση προγραμματισμού. Τα συμβάντα είναι το κύριο χαρακτηριστικό της Construct που αντί για περίπλοκες γλώσσες προγραμματισμού, scripting και σφάλματα, ο χρήστης ορίζει την λειτουργία του παιχνιδιού χρησιμοποιώντας ένα απλούστερο σύστημα από block events.



Εικόνα 20- Διάγραμμα ενός Event

Τα events αποτελούνται από τρία κύρια μέρη:

1. Event Block
2. Conditions
3. Actions

Η βασική ιδέα των συμβάντων είναι ότι οι συνθήκες φιλτράρουν τις περιπτώσεις που πληρούν την προϋπόθεση και οι ενέργειες εκτελούνται μόνο για αυτές. Το Event Block περιέχει τις συνθήκες (Conditions), και την επιλογή ο χρήστης να επιλέξει ολόκληρο το event. Οι συνθήκες οι οποίες εμφανίζονται μέσα στο block συμβάντων ενώ οι ενέργειες (Actions) παρατίθενται δεξιά του block.

Αν και η μηχανή Construct δεν διαθέτει Editor για την συγγραφή κώδικα, καθώς το σύστημα των Events είναι αρκετά ισχυρό για να αντικαταστήσει το scripting τελείως, υπάρχει η επιλογή για τους προγραμματιστές και για αυτούς που επιθυμούν να εντρυφήσουν στην Construct να χρησιμοποιήσουν τον Addon SDK.

3.3 Μηχανή Wimi5

3.3.1 Γενικά για τη μηχανή Wimi5

Η μηχανή ανάπτυξης παιχνιδιών Wimi5, παρουσιάστηκε πρώτη φορά στο κοινό τον Σεπτέμβριο του 2013 στην Ιαπωνία και συγκεκριμένα στο Tokyo Game Show 2013 (*Tokyo Game Show- γνωστό ως TGS, είναι μια έκθεση βιντεοπαιχνιδιών που πραγματοποιείται κάθε χρόνο τον Σεπτέμβριο στο Makuhari Messe, στο Chiba της Ιαπωνίας. Παρουσιάζεται από το Computer Entertainment Supplier's Association (CESA) και την Nikkei Business Publications, Inc. Ο κύριος σκοπός της έκθεσης είναι η παρουσίαση των παιχνιδιών που αναπτύσσονται από Ιάπωνες προγραμματιστές, αλλά ορισμένοι διεθνείς προγραμματιστές παιχνιδιών το χρησιμοποιούν για να προβάλλουν επερχόμενες εκδόσεις / σχετικό υλικό. Όπως και το gamescom της Γερμανίας, το Tokyo Game Show επιτρέπει στο ευρύ κοινό να παρακολουθήσει τις τελευταίες δύο ημέρες– Wikipedia 2018*). Το όραμα των δημιουργών της μηχανής Wimi5 είναι να «ξεκλειδώσουν» τις δυνατότητες της HTML5 για όλους και να δημιουργήσουν ένα τυποποιημένο εργαλείο που να επιτρέπει σε οποιονδήποτε να δημιουργήσει, να δημοσιεύσει και να κερδίσει χρήματα από την ανάπτυξη παιχνιδιών. Έτσι καταλήγει στο παρακάτω τρίπτυχο που συνοψίζει την φιλοσοφία της μηχανής και των δημιουργών της:

- Δημιουργία παιχνιδιών
- Έκδοση παιχνιδιών
- Κέρδος από τα παιχνίδια

Η λειτουργία της μηχανής Wimi5 βασίζεται στις διαδικασίες drag and drop και αυτό ακριβώς είναι που την κάνει προσιτή στο ευρύ κοινό, χωρίς να απαιτούνται ιδιαίτερες γνώσεις προγραμματισμού. Η ευκολία στην χρήση της επιτρέπει τόσο σε αρχάριους όσο και σε έμπειρους προγραμματιστές να κατασκευάσουν παιχνίδια σε λιγότερο χρόνο από ότι με την χρήση κάποιας γλώσσας προγραμματισμού. Η Wimi5 παρέχει στους χρήστες της την δυνατότητα να επιλέξουν δικά τους αρχεία ήχου, γραφικών και backgrounds είτε να επιλέξουν κάποιο από τα διαθέσιμα templates. Οι χρήστες της μηχανής μπορούν να διαχειριστούν όλα τα στοιχεία του παιχνιδιού τους, όπως οι σκηνές ή οι χαρακτήρες μέσω του scene editor. Στη συνέχεια, χρησιμοποιώντας τον Logic editor μπορούν να ρυθμίσουν τη ροή των σκηνών και τα υπόλοιπα συμβάντα που συνθέτουν το παιχνίδι. Ένα ακόμα στοιχείο που βοηθά την Wimi5 να χαρακτηριστεί ως μια φιλική μηχανή προς τον απλό χρήστη, είναι τα οπτικά μενού.

Η έκδοση των παιχνιδιών, όπως και στις περισσότερες online game engines βασίζεται στην διαδικασία one click. Έτσι, πολύ εύκολα ο δημιουργός του παιχνιδιού μπορεί να διανείμει το url του παιχνιδιού του σε οποιαδήποτε ιστοσελίδα, σε οποιοδήποτε μέσω κοινωνικής δικτύωσης είτε σε online καταστήματα, όπως το Google Chrome Web Store ή το Firefox Marketplace.

Η μηχανή Wimi5 προβάλλεται ως η καταλληλότερη επιλογή ανάμεσα στο πλήθος των μηχανών παιχνιδιών στηριζόμενη στους τομείς της διαφήμισης, της online κοινότητας, και φυσικά στα παιχνίδια που απευθύνονται σε παιδιά και στα επώνυμα παιχνίδια. Η αναπαραγωγή των διαφημίσεων έχει σχεδιαστεί έτσι ώστε να ανταποκρίνονται αποτελεσματικά στο περιβάλλον που προβάλλονται, ανεξάρτητα της συσκευής και του λειτουργικού συστήματος που χρησιμοποιεί ο καταναλωτής. Η online κοινότητα της Wimi5 διαφέρει από το συνηθισμένο forum που υποστηρίζουν σχεδόν όλες οι μηχανές παιχνιδιών. Η Wimi5 παρακινεί τους χρήστες να επικοινωνήσουν απευθείας μέσω της επίσημης ιστοσελίδα της είτε για βοήθεια είτε ακόμα και για χρήσιμες πληροφορίες ή ιδέες. Το αίτημα του χρήστη αφού σταλεί στην εταιρεία, θα προωθηθεί σε κάποιον από τους πιστοποιημένους προγραμματιστές της Wimi5 και θα απαντηθεί. Όσο αφορά τα παιχνίδια, διατίθεται μια εξαιρετικά μεγάλη λίστα με παιχνίδια με ποικίλο περιεχόμενο, τα οποία μπορούν να προσαρμοστούν ανάλογα με τις ανάγκες και τις επιθυμίες του χρήστη.

Στόχος της μηχανής Wimi5 είναι ο καθένας, χωρίς απαραίτητα να είναι προγραμματιστής, να μπορέσει με τη μηχανή να φτιάξει ένα παιχνίδι, το οποίο θα είναι λειτουργικό σε οποιαδήποτε πλατφόρμα και συσκευή, θα μπορεί να διαμοιραστεί άμεσα στα κοινωνικά δίκτυα, να προσελκύσει διαφημιστές, να δημιουργήσει μια θετική εμπειρία στον παίκτη, η οποία θα οδηγήσει σε περισσότερες επισκέψεις και τέλος να αποφέρει κέρδη στον δημιουργό του.

3.3.2 Υποστηριζόμενοι Browsers Wimi5

Η μηχανή Wimi5 βασίζεται εξ' ολοκλήρου στην HTML5, το οποίο σημαίνει ότι τα παιχνίδια τα οποία δημιουργούνται με την μηχανή Wimi5 μπορούν να αναπαραχθούν σε οποιαδήποτε συσκευή η οποία διαθέτει ένα σύγχρονο πρόγραμμα περιήγησης. Δεν ισχύει κανένας περιορισμός από την μεριά της μηχανής, παρά μόνο οι απαιτήσεις και οι ιδιαιτερότητες που έχει η κάθε πλατφόρμα. Ωστόσο στην επίσημη ιστοσελίδα της μηχανής, προτείνεται ο Chrome.

Πίνακας 5- Απαιτήσεις Προγραμμάτων Περιήγησης Wimi5

Browser	Windows	MacOS	Linux	Android	IOS
Chrome	✓	✓	✓	✓	
Firefox	✓	✓	✓	✓	✓
IE	✓		✓	✓	
Edge	✓		✓	✓	
Safari	✓	✓	✓	✓	✓

3.3.3 Κόστος μηχανής Wimi5

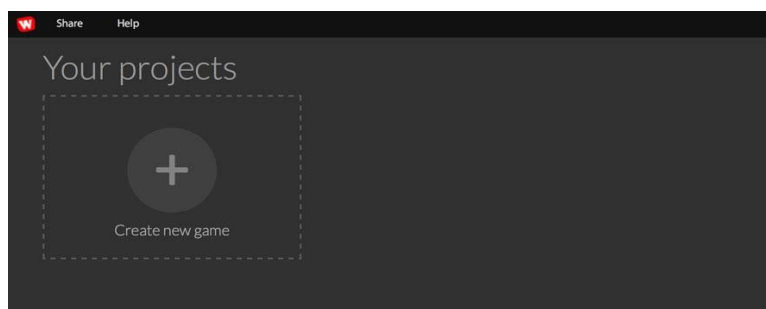
Η μηχανή Wimi5 διατίθεται με πέντε άδειες κυκλοφορίας και με δωρεάν την φιλοξενία του παιχνιδιού στην πλατφόρμα της Wimi5. Για τους χρήστες της μηχανής που στόχος τους είναι το κέρδος, υπάρχει το πρόγραμμα Free2Play. Το Free2Play πρόκειται ουσιαστικά για ένα επιχειρηματικό μοντέλο ή αλλιώς ένα κανάλι εσόδων. Όταν ένας παίκτης που απολαμβάνει ένα παιχνίδι της Wimi5 έχει την επιλογή να αγοράσει ένα στοιχείο, όπως είναι μια ζωή, ένα budget, ή μια βοήθεια μέσα στο παιχνίδι, τότε ο δημιουργός του παιχνιδιού λαμβάνει το 70% των εσόδων.

Πίνακας 6- Συγκριτικό τιμολόγιο της μηχανής Wimi5

Χαρακτηριστικά	Personal	Starter	Profesional	Επιχειρηματική	Εκπαιδευτική
Διαφημίσεις στο παιχνίδι	NAI	OXI	OXI	OXI	OXI
Χρήση Logo	NAI	NAI	OXI	OXI	OXI
Αριθμός Project	5	10	40	Χωρίς Όριο	Χωρίς Όριο
Χώρος αποθήκευσης	100MB	500MB	2GB	5GB	5GB
Αποθήκευση Cloud	NAI	NAI	NAI	NAI	NAI
Εκδοση στο Web	NAI	NAI	NAI	NAI	NAI
Εκδοση σε άλλες Πλατφόρμες	OXI	NAI	NAI	NAI	NAI
Κόστος ανά Μήνα	—	15€	50€	300€	100€

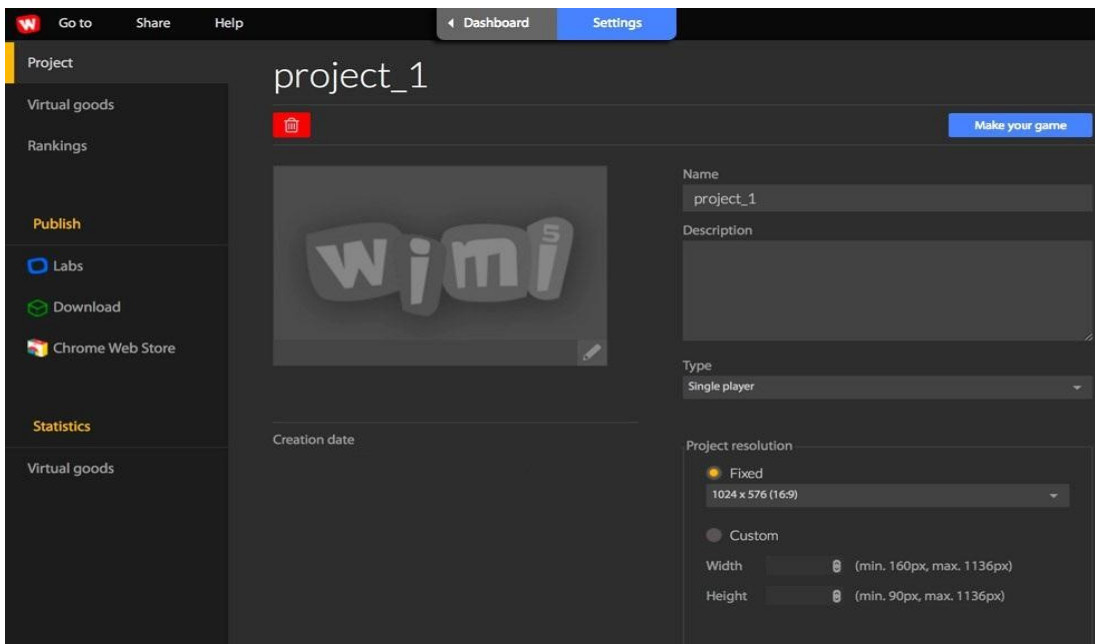
3.3.4 Τρόπος λειτουργίας της μηχανής Wimi5

Για την χρήση της μηχανής Wimi5 δεν απαιτείται τίποτα περισσότερο από μια εγγραφή και ένα login στην επίσημη ιστοσελίδα της μηχανής. Μέσω της επίσημης ιστοσελίδας ο χρήστης έχει πρόσβαση σε αρκετά tutorials, video tutorials, documentation, templates και παραδείγματα ώστε να είναι σε θέση να προχωρήσει στην δημιουργία του παιχνιδιού του. Με την είσοδό του στο σύστημα την πρώτη φορά, εμφανίζεται στον χρήστη η παρακάτω εικόνα:



Εικόνα 21- Dashboard Wimi5

Με την δημιουργία νέου παιχνιδιού, δίνονται στον χρήστη δύο επιλογές. Η πρώτη είναι να ξεκινήσει ένα νέο παιχνίδι από την αρχή και η δεύτερη να κλωνοποιήσει ένα υπάρχον παιχνίδι και να δουλέψει πάνω σε αυτό. Στη συνέχεια της δημιουργίας του project εμφανίζεται η εξής οθόνη, η οποία περιέχει γενικά στοιχεία όπως το όνομα του project, η περιγραφή, στατιστικά στοιχεία και άλλες πληροφορίες.

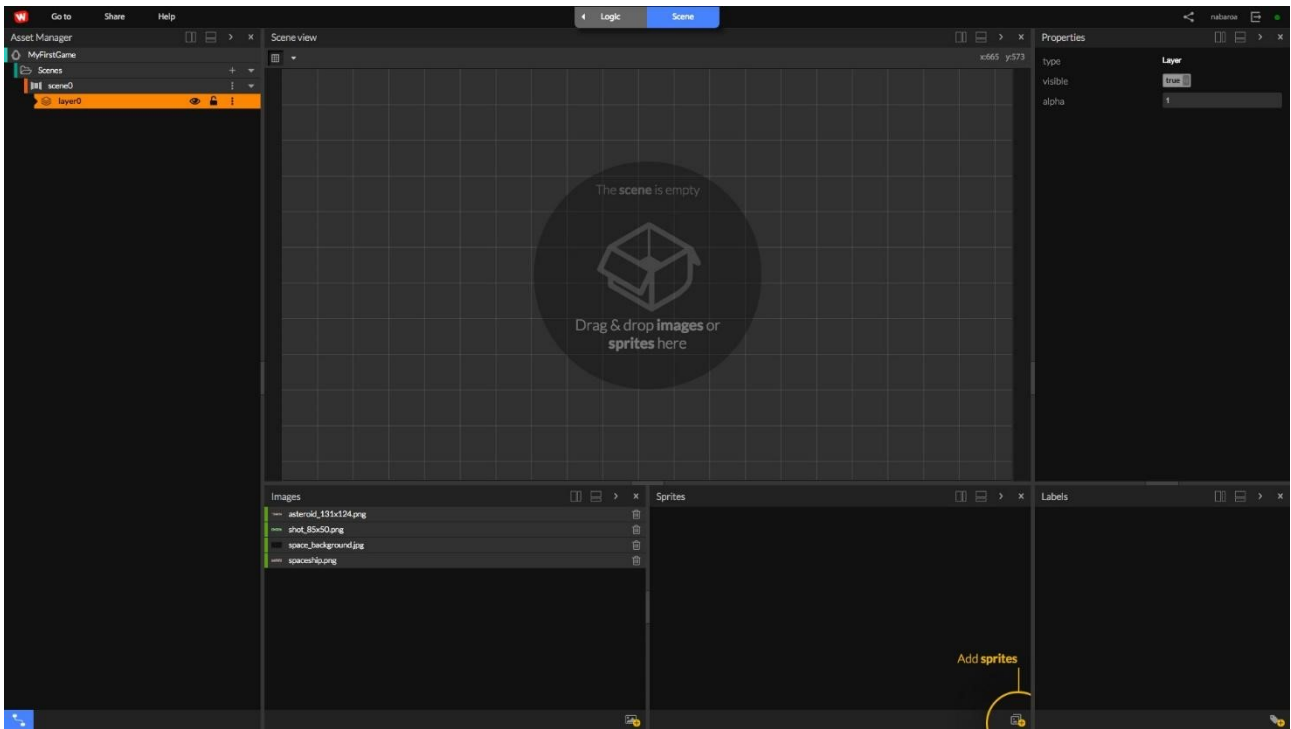


Εικόνα 22- Project Settings Wimi5

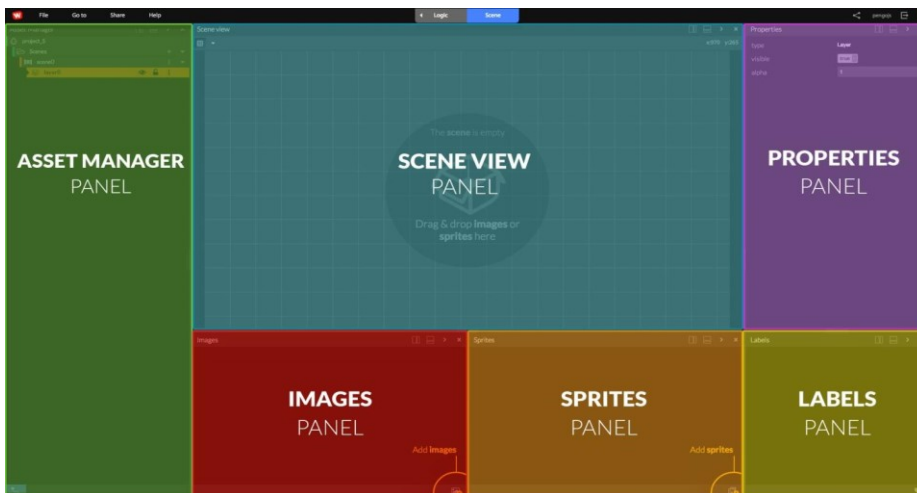
Ένα σημαντικό στοιχείο της μηχανής Wimi5 είναι ο διαχωρισμός της σε δυο διαφορετικά πεδία ή αλλιώς σε δυο διαφορετικές επιφάνειες εργασίας. Η μια επιφάνεια εργασίας χρησιμοποιείται για την δημιουργία των σκηνών και των οντοτήτων του παιχνιδιού και η άλλη για την δημιουργία της λογικής του παιχνιδιού.

3.3.5 Scene Editor

Ο Scene Editor της μηχανής Wimi5 αποτελεί τον χώρο στον οποίο ο χρήστης δημιουργεί το παιχνίδι του οπτικά. Μέσω του Scene Editor ο χρήστης διαχειρίζεται όλες τις λειτουργίες που σχετίζονται με την οπτική δημιουργία των σκηνών του παιχνιδιού, όπως για παράδειγμα το ανέβασμα μιας εικόνας ή το αρχείο ενός ήχου. Ο Scene Editor αποτελείται αρχικά από τα πάνελ που αφορούν την προβολή σκηνών, τις εικόνες, τα Sprites, τις ιδιότητες, τις ετικέτες και το Asset Manager.



Εικόνα 23- Scene Editor Wimi5

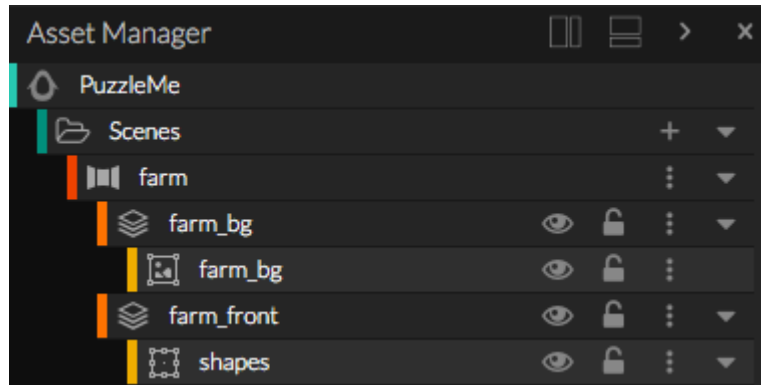


Εικόνα 24- Τα πάνελ του Scene Editor

Asset Manager

Στο πεδίο του Asset Manager ο χρήστης μπορεί να δημιουργήσει και να διαχειριστεί τις σκηνές και τα επίπεδα που απαιτούνται για το παιχνίδι. Η δημιουργία ενός αντικειμένου γίνεται πολύ απλά σέρνοντας την εικόνα που χρειάζεται, από το πάνελ Images μέσα στην προβολή σκηνής (Scene View). Αυτόματα τότε δημιουργείται στο Asset Manager η οντότητα του νέου αντικειμένου. Στο Asset Manager ο χρήστης μπορεί να επεξεργαστεί κάθε αντικείμενο ξεχωριστά ή και ομαδικά. Μπορεί να το διαγράψει, να το κλωνοποιήσει, να δημιουργήσει επιπλέον οντότητες που θα είναι

«παιδιά» σε αυτό, να το μπλοκάρει σε ορισμένα επίπεδα, να το μετακινήσει βαθμίδες αλλά και θέση μέσα στην σκηνή.



Εικόνα 25- Asset Manager Wimi5

Scene View

Μέσα στον πίνακα προβολής ο χρήστης είναι ικανός να μεταφέρει τα γραφικά στοιχεία για δημιουργήσει την σκηνή του παιχνιδιού. Με την χρήση του ποντικιού μπορεί να αλλάξει το μέγεθος ενός στοιχείου, να το περιστρέψει, να το διαγράψει ή και να το κλωνοποιήσει. Για περισσότερη ακρίβεια στις κινήσεις του υπάρχει ένα επιπλέον εργαλείο το πλέγμα. Ακόμα καθ' όλη την διάρκεια όπου ο δείκτης του ποντικιού βρίσκεται μέσα στο Scene View στην πάνω δεξιά γωνία εμφανίζονται οι συντεταγμένες του.

Images

Μέσω του πάνελ Images πραγματοποιείται η μεταφόρτωση αρχείων εικόνας και ήχου στο διαθέσιμο χώρο του cloud που προσφέρει η Wimi5. Η διαδικασία είναι πάρα πολύ απλή καθώς γίνεται με την χρήση του ειδικού κουμπιού που βρίσκεται στην κάτω δεξιά γωνία του πάνελ, είτε απλά σέρνοντας το αρχείο από το σύστημα του υπολογιστή μέσα στην Wimi5. Και στις δυο περιπτώσεις θα μας εμφανιστεί το ίδιο παράθυρο διαλόγου, στο οποίο παρουσιάζονται όλα τα αρχεία εικόνας και ήχου τα οποία έχει ανεβάσει ο χρήστης συνολικά σε κάθε project. Από την μεριά της μηχανής Wimi5, δεν υπάρχει κάποιος περιορισμός στον τύπο των αρχείων που μπορεί να ανεβάσει ο χρήστης. Αυτό που θα πρέπει να λάβει υπόψη του όμως ο χρήστης για το ποιο τύπο αρχείου θα χρησιμοποιήσει, είναι εάν υπάρχει κάποιος περιορισμός από τον browser, καθώς ορισμένοι browser υποστηρίζουν συγκεκριμένες επεκτάσεις αρχείων. Τέλος για καλύτερη οργάνωση και διαχείριση, υπάρχει η δυνατότητα δημιουργίας φακέλων και υποφακέλων.



Εικόνα 26- Resources Manager Wimi5

Sprites

Η δημιουργία των sprites στην μηχανή Wimi5 υλοποιείται μέσω του πάνελ Sprites. Η λειτουργία αυτού του πάνελ συνδέεται με το πάνελ Images καθώς μέσα από το συγκεκριμένο πάνελ φορτώνονται και οι εικόνες που θα χρησιμοποιηθούν ως sprites και στη συνέχεια μεταφέρονται στο πάνελ Sprites. Και πάλι ως βασικές ιδιότητες ο χρήστης μπορεί να κλωνοποιήσει ή να διαγράψει το sprite και φυσικά να ρυθμίσει την κίνηση του και να προσθέσει όσες κινήσεις χρειάζονται.



Εικόνα 27- Sprites Wimi5

Properties

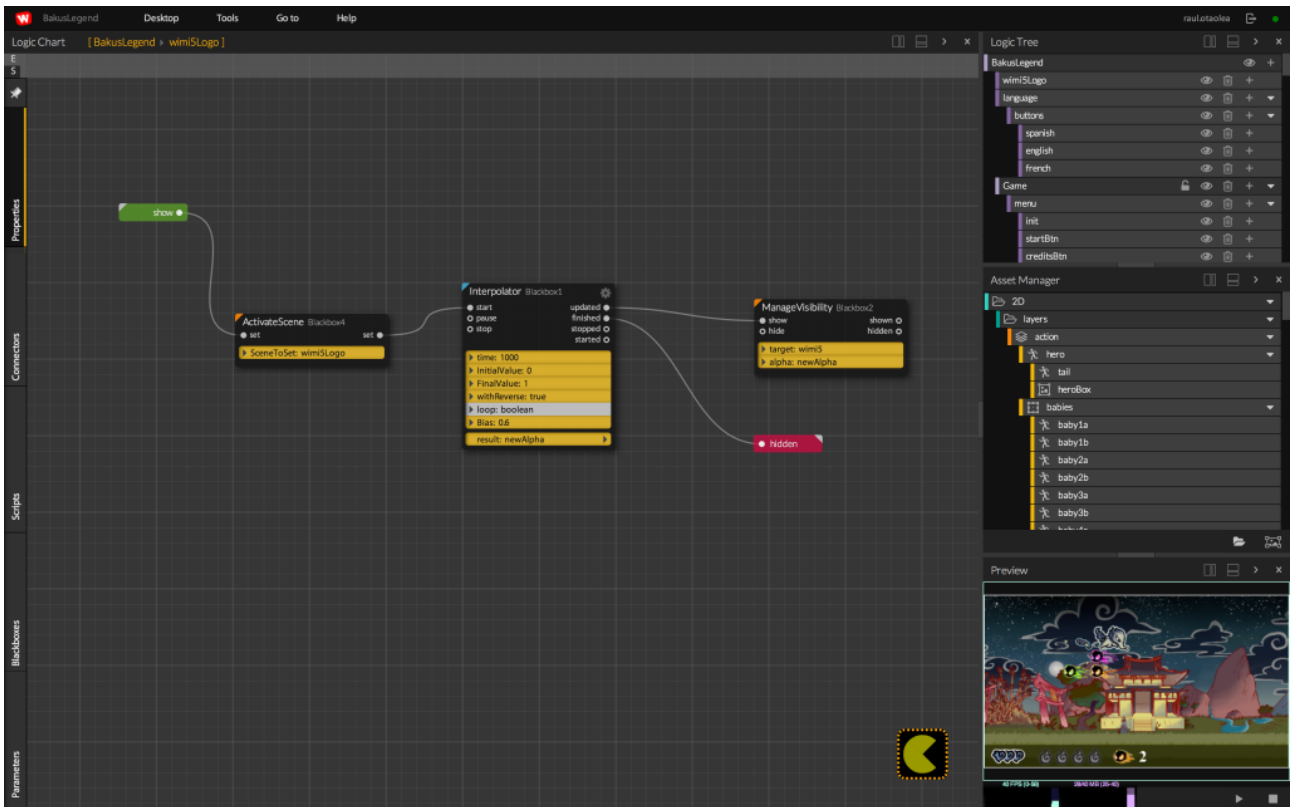
Από το πάνελ Properties ο χρήστης μπορεί να ρυθμίσει όλες τις ιδιότητες που διαθέτουν τα γραφικά στοιχεία που βρίσκονται στο Scene View. Μπορεί να ρυθμίσει τις συντεταγμένες, την κλίμακα, αν ένα στοιχείο θα είναι ορατό ή όχι, αν θα είναι ορατό ή εάν θα επαναλαμβάνεται. Ανάλογα με το στοιχείο, εάν είναι εικόνα ή sprite για παράδειγμα, το πάνελ παρουσιάζει και άλλες ρυθμίσεις όπως είναι η ρύθμιση για το frame per second.

Labels

Στο πάνελ Labels ο χρήστης μπορεί να δημιουργήσει και να διαχειρίζεται ετικέτες. Οι ετικέτες δίνουν την δυνατότητα να ομαδοποιεί ένα σύνολο στοιχείων και αντίστοιχα να το χρησιμοποιεί όπως αυτός επιθυμεί. Έτσι μειώνεται ο χρόνος επεξεργασίας στοιχείων και η ανάπτυξη του παιχνιδιού υλοποιείται με περισσότερη οργάνωση και ευκολία.

3.3.6 Logic Editor

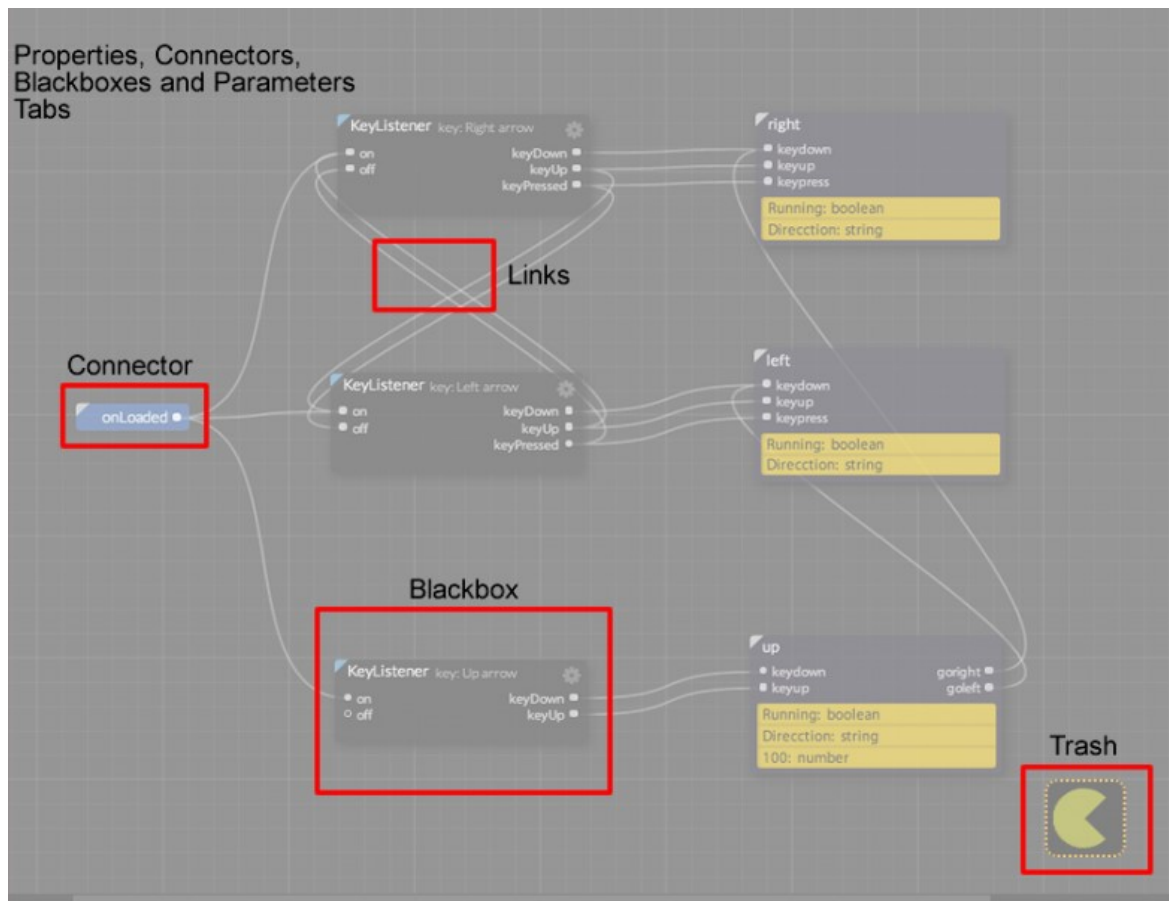
Ο χώρος του Logic Editor είναι ο χώρος στον οποίο μέσα ο χρήστης δημιουργεί τη λογική του παιχνιδιού. Πρόκειται ουσιαστικά για τον πυρήνα της HTML5 όπου αναπτύσσονται όλες οι ενέργειες και οι συμπεριφορές των οντοτήτων του παιχνιδιού.



Εικόνα 28- Logic Chart Wimi5

Logic Chart

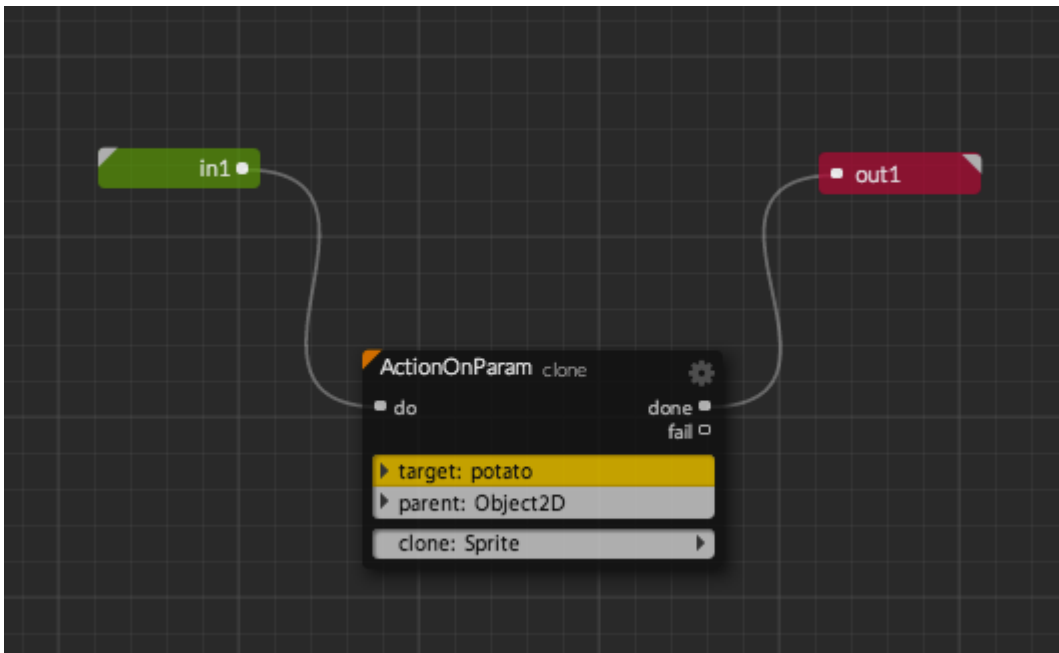
Στον κεντρικό χώρο του Logic Editor ο χρήστης δημιουργεί όλα τα λογικά σενάρια για το παιχνίδι του. Σε αυτόν τον χώρο με την απλή διαδικασία drag and drop κάνει χρήση των Blackboxes που περιέχουν όλες τις ενέργειες, τους κανόνες και τις συμπεριφορές του παιχνιδιού. Ακόμα ο χρήστης μπορεί να περιηγηθεί μέσα στα σενάρια των οντοτήτων, να μεταφέρει συνδέσμους, παραμέτρους και ιδιότητες ώστε να δημιουργήσει το σενάριο που χρειάζεται για το παιχνίδι.



Εικόνα 29- Συνδεσμολογία αντικειμένων Wimi5

Connectors Tab

Από τα πιο σημαντικά εργαλεία που παρέχει ο Logic Editor είναι οι Connectors Tab. Οι Connectors είναι τα σημεία εισόδου και εξόδου των ενεργειών που λειτουργούν τα σενάρια του παιχνιδιού. Καθορίζουν ουσιαστικά, τη ροή εκτέλεσης του παιχνιδιού. Μια απλουστευμένη περιγραφή που μπορεί να δοθεί στους Connectors Tab είναι αυτή του διακόπτη. Αυτός ο διακόπτης ενεργοποιεί τη λειτουργία του σεναρίου, του BlackBox κτλ. Ο χρήστης συνδέοντας τον Connector με οποιοδήποτε άλλο πάνελ, επιλέγει εάν θα είναι ενεργό ή ανενεργό. Έχει στην κατοχή του λοιπόν, έναν διακόπτη με τον οποίο ελέγχει την είσοδο και την έξοδο των πάνελ και το πότε θα λειτουργήσει το συμβάν που είναι συνδεδεμένος, ακριβώς όπως ένας διακόπτης.



Εικόνα 30- Είσοδος/Έξοδος σεναρίου Wimi5

BlackBoxes Tab

Η καρτέλα του BlackBox είναι το βασικό στοιχείο σύνθεσης της λογικής του παιχνιδιού που δημιουργεί ο χρήστης, το οποίο εμφανίζεται οπτικά ως ένα απλό κουτί. Αυτό το κουτί αντιπροσωπεύει μια διαδικασία η οποία μπορεί να λειτουργήσει κατά βούληση και σε συνδυασμό φυσικά με άλλα στοιχεία του Logic Editor για την κατασκευή πιο σύνθετων διαδικασιών. Τα Blackboxes είναι στοιχεία που ενεργοποιούνται και βάση αυτών ενεργοποιούνται και εκτελούνται νέες ενέργειες, σύμφωνα πάντα με την σχεδίαση του παιχνιδιού.

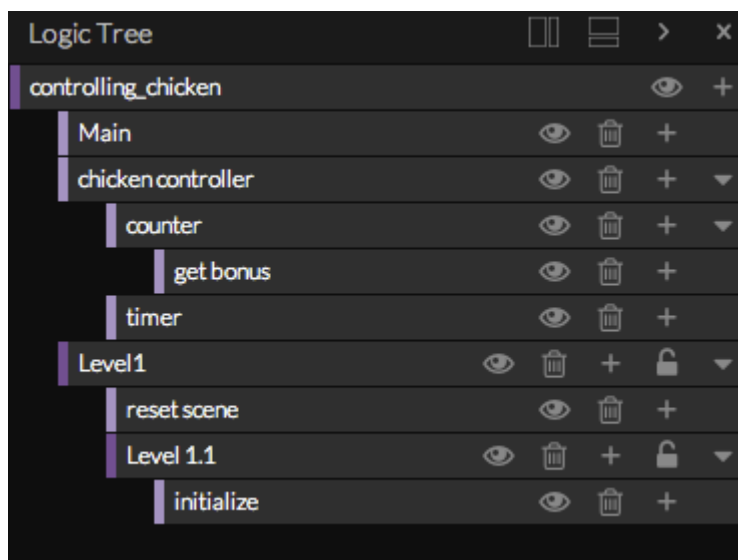
Parameters Tab

Το πάνελ των παραμέτρων διαθέτει στον χρήστη όλες τις επιλογές με τις οποίες μπορεί να συνδέσει τις εισόδους και τις εξόδους των Blackboxes. Το σύστημα της μηχανής Wimi5 παρέχει έναν αυτοματοποιημένο έλεγχο πάνω στις συνδέσεις αυτές. Εάν ο τύπος της παραμέτρου δεν είναι ο κατάλληλος για την είσοδο ή την έξοδο του Blackbox που θέλει ο χρήστης, τότε δεν θα του επιτραπεί να πραγματοποιήσει τη σύνδεση, εμφανίζοντάς του ένα εικονίδιο απαγόρευσης.

Logic Tree

Πρωταγωνιστικό ρόλο στην οπτικοποίηση της δομής του παιχνιδιού έχει το πάνελ Logic Tree. Σε γενικά πλαίσια ένα δενδροδιάγραμμα έχει ιδιαίτερη σημασία στον προγραμματισμό, καθώς σε αυτό οργανώνονται ιεραρχικά όλες οι δομές και οι λειτουργίες ενός προγράμματος. Έτσι λοιπόν στην Wimi5 και στο Logic Tree ο χρήστης έχει τη δυνατότητα να δημιουργήσει τη δομή,

πάνω στην οποία θα οργανώνει οπτικά τα σενάρια του παιχνιδιού. Τα σενάρια οργανώνονται σε στρώματα και αυτό επιτρέπει στον χρήστη να τα διαχειρίζεται πιο εύκολα. Εκτός από τα σενάρια του παιχνιδιού, στο Logic Tree ο χρήστης μπορεί να διαχειρίζεται και τις σκηνές αλλά και τα επίπεδα του παιχνιδιού, καθώς επίσης και να διαχειριστεί τα στοιχεία (εικόνες κτλ) που υπάρχουν σε κάθε επίπεδο του παιχνιδιού.



Εικόνα 31- Logic Tree Wimi5

Asset Manager

Μέσα στο πάνελ του Asset Manager απεικονίζεται η δομή των αντικειμένων που έχουν δημιουργηθεί στον Scene Editor. Μέσα από τον Asset Manager ο χρήστης μπορεί να μεταφέρει αυτά τα αντικείμενα στις παραμέτρους εισόδου και εξόδου των Blackboxes και να ορίσει έτσι την λειτουργία τους (<http://wimi5.com/roadmap/>).

3.4 Μηχανή Ceilfire

3.4.1 Γενικά για τη μηχανή Ceilfire

Η μηχανή Ceilfire είναι μια online μηχανή ανάπτυξης παιχνιδιών HTML5. Η μηχανή αυτή βασίζεται στην τεχνολογία του cloud. Όπως έχουμε αναφέρει ήδη, αυτό σημαίνει ότι για την λειτουργία της, δεν χρειάζεται η εγκατάσταση κανενός λογισμικού ή κάποιου plugin διότι η ανάπτυξη των παιχνιδιών γίνεται απευθείας στην επίσημη ιστοσελίδα της μηχανής. Επίσης για τον χειρισμό της μηχανής δεν απαιτείται η εκμάθηση κάποιας περίπλοκης γλώσσας προγραμματισμού από τον χρήστη, καθώς ο Editor της μηχανής Ceilfire βασίζεται στο σύστημα drag and drop. Η έκδοση του παιχνιδιού που δημιουργεί ο χρήστης πραγματοποιείται απλά με ένα κλικ στην πλατφόρμα της Ceilfire και ο σύνδεσμος του παιχνιδιού, μπορεί να κοινοποιηθεί σε οποιοδήποτε site, blog και μέσο κοινωνικής δικτύωσης. Η πρώτη έκδοση της μηχανής Ceilfire έγινε το 2014 και

στηρίζεται στο Phaser.js, με στόχο η ανάπτυξη παιχνιδιών για browsers και κινητές συσκευές να γίνει όσο πιο απλή και προσιτή για όλους όσους θέλουν να ασχοληθούν με τον τομέα της ανάπτυξης παιχνιδιών. Σκοπός των δημιουργών της μηχανής Ceilfire είναι η ανάπτυξη μιας μηχανής από προγραμματιστές για τους προγραμματιστές. Στα πρώτα στάδια της μηχανής υπήρχε μια διαρκής ενημέρωση των χρηστών της μηχανής μέσα από forums για κάθε διόρθωση ή νέα λειτουργία που πρόσθεταν. Οι δημιουργοί της μηχανής βασίζονταν στο feedback των χρηστών για να διορθώσουν τυχόν σφάλματα που δεν είχαν γίνει μέχρι τότε αντιληπτά ή για να προσθέσουν κάποια λειτουργία την οποία πρότειναν οι χρήστες. Το ενδιαφέρον στην μηχανή Ceilfire είναι το User Interface της. Με μια πρώτη ματιά δεν διαφέρει από τις άλλες μηχανές που έχουν αναλυθεί μέχρι τώρα, είναι και αυτή χωρισμένη σε διάφορα τμήματα, με κεντρικό τον scene editor. Ωστόσο η συνολική της μορφή είναι πολύ πιο απλοποιημένη και εύχρηστη.

3.4.2 Υποστηριζόμενοι Browsers Ceilfire

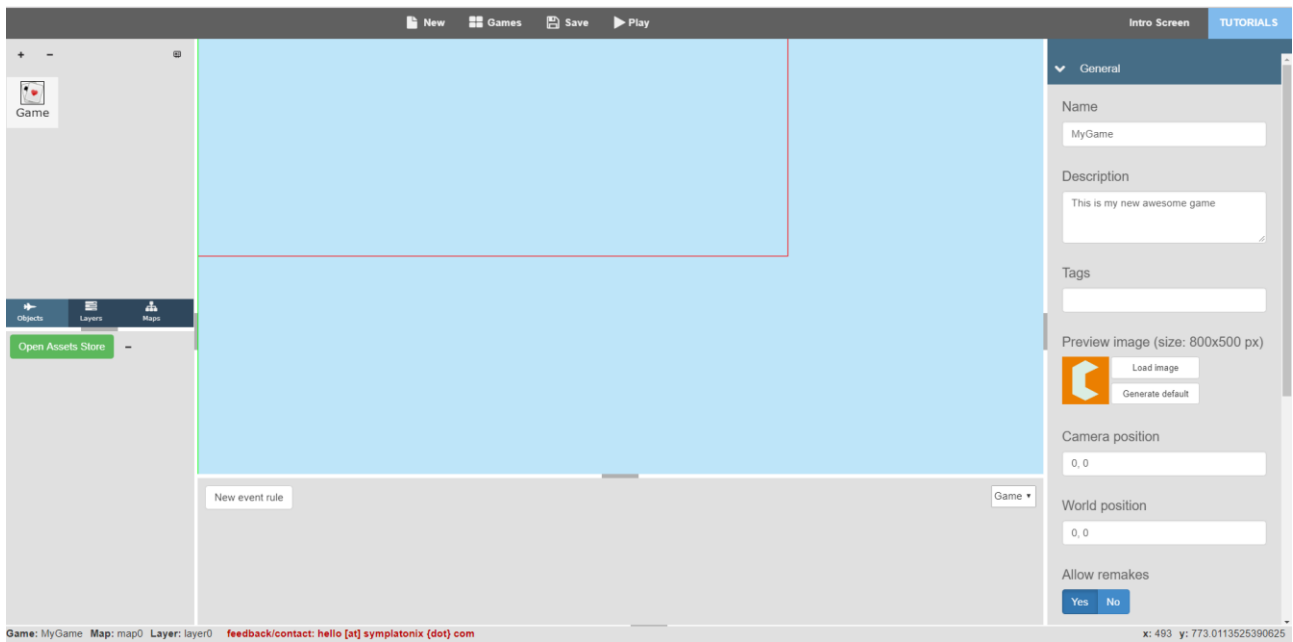
Η μηχανή Ceilfire υποστηρίζεται από όλους τους browsers, χωρίς κάποιο περιορισμό. Η δημοσίευση των παιχνιδιών, πραγματοποιείται επίσης χωρίς κάποιον περιορισμό. Ωστόσο σε browsers όπου υπάρχει προεγκατεστημένο κάποιο πρόγραμμα για την μη αυτόματη αναπαραγωγή διαφημίσεων, παρατηρείται μια μικρή καθυστέρηση στην έναρξη των παιχνιδιών, μερικών δευτερολέπτων [Πίνακας 1].

3.4.3 Κόστος μηχανής Ceilfire

Η μηχανή Ceilfire διατίθενται εντελώς δωρεάν προς χρήση. Δωρεάν είναι και η δημοσίευση των παιχνιδιών αλλά και η φιλοξενία τους στην πλατφόρμα της Ceilfire. Η Ceilfire δεν διαθέτει εξειδικευμένα πακέτα έναντι κάποιου ποσού για την χρήση από εταιρείες, πανεπιστημιακά ιδρύματα ή από άτομα που επιθυμούν να ασχοληθούν σε μεγαλύτερο βαθμό με την μηχανή και την ανάπτυξη παιχνιδιών.

3.4.4 Τρόπος λειτουργίας της μηχανής Ceilfire

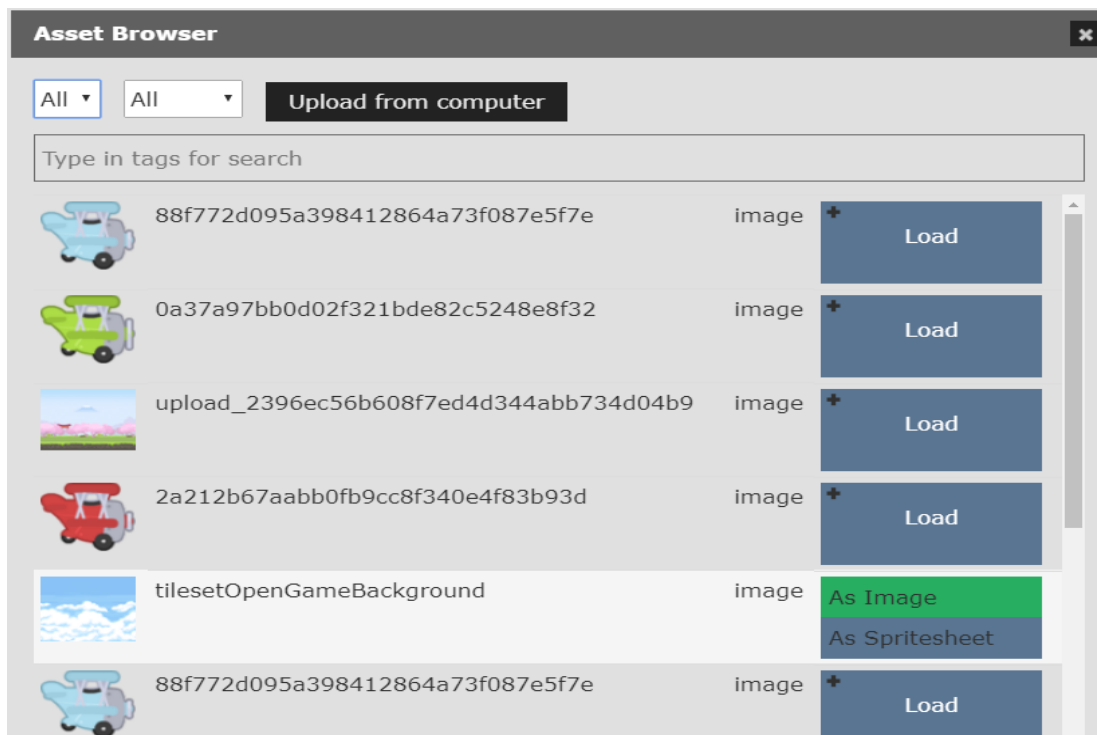
Ο τρόπος λειτουργίας της μηχανής Ceilfire είναι αρκετά απλός και κατανοητός. Όπως έχει ήδη αναφερθεί η μηχανή είναι web-based και cloud-based και έτσι ο χρήστης έχει πρόσβαση στην μηχανή, μόνο μέσω της επίσημης ιστοσελίδας της. Για να ξεκινήσει την χρήση της μηχανής, ο χρήστης θα πρέπει να δημιουργήσει λογαριασμό και να συνδεθεί. Το περιβάλλον της μηχανής είναι όμορφα σχεδιασμένο και εύχρηστο.



Εικόνα 32- Περιβάλλον Ceilfire

Η γραμμή εργαλείων της μηχανής δίνει στον χρήστη την δυνατότητα να ανοίξει ένα νέο project, να περιηγηθεί στα παιχνίδια του, να κάνει αποθήκευση και τέλος να κάνει μια προεπισκόπηση του παιχνιδιού.

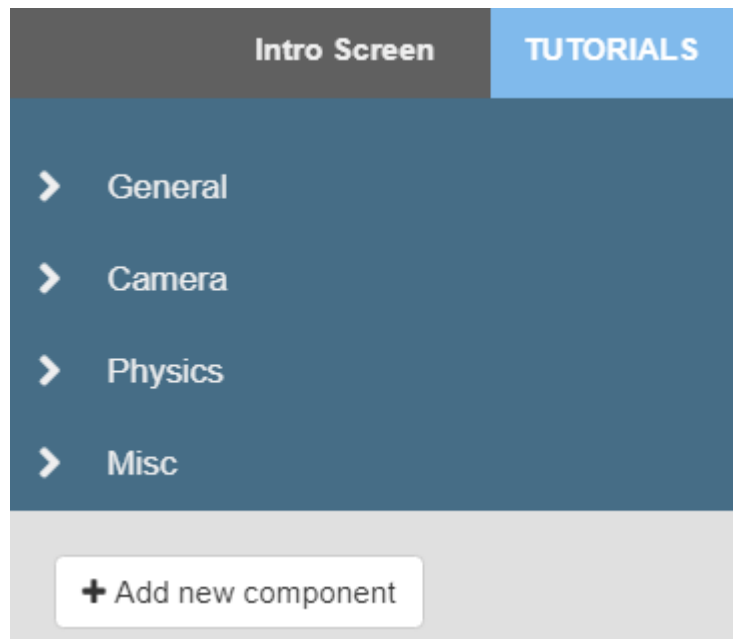
Η μηχανή Ceilfire μέσω της επίσημης ιστοσελίδας, προσφέρει στους χρήστες της πληθώρα στοιχείων, έτοιμων προς χρήση. Ο χρήστης επισκέπτεται την ιστοσελίδα, «κατεβάζει» τα στοιχεία που επιθυμεί και τα φορτώνει στην μηχανή.



Εικόνα 33- Asset Browser Ceilfire

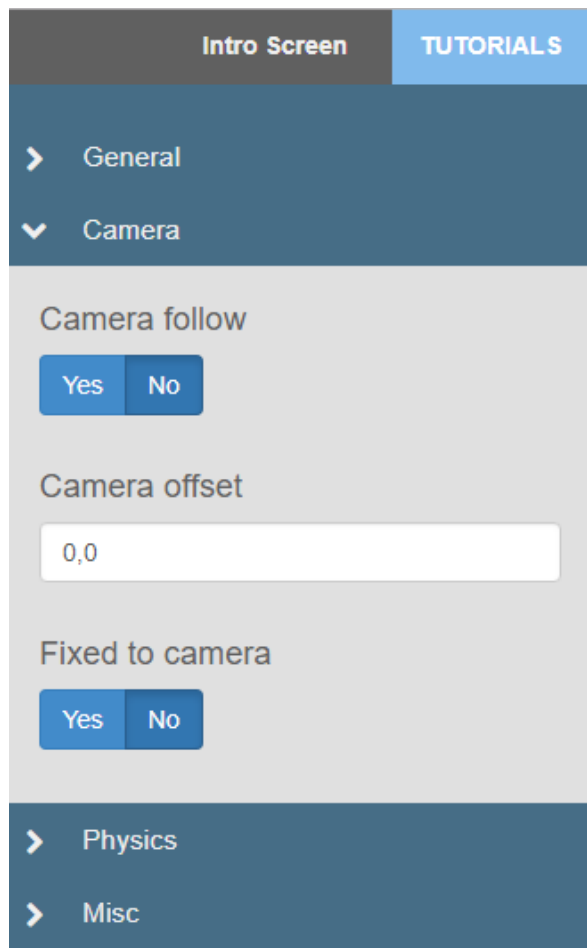
Τα στοιχεία ανεβαίνουν στην μηχανή, είτε σαν εικόνες, είτε σαν sprites, όπου ο χρήστης μπορεί να ρυθμίσει τις στήλες και τις γραμμές της εικόνας, όπως αυτός θέλει.

Εάν τοποθετηθεί ένα στοιχείο μέσα στην κύρια οθόνη της μηχανής, τότε δημιουργείται ένα αντικείμενο και εμφανίζονται στα δεξιά της οθόνης οι ρυθμίσεις που υπάρχουν για το κάθε στοιχείο, αντικείμενο ή επίπεδο.



Εικόνα 34- Μενού ρυθμίσεων Ceilfire

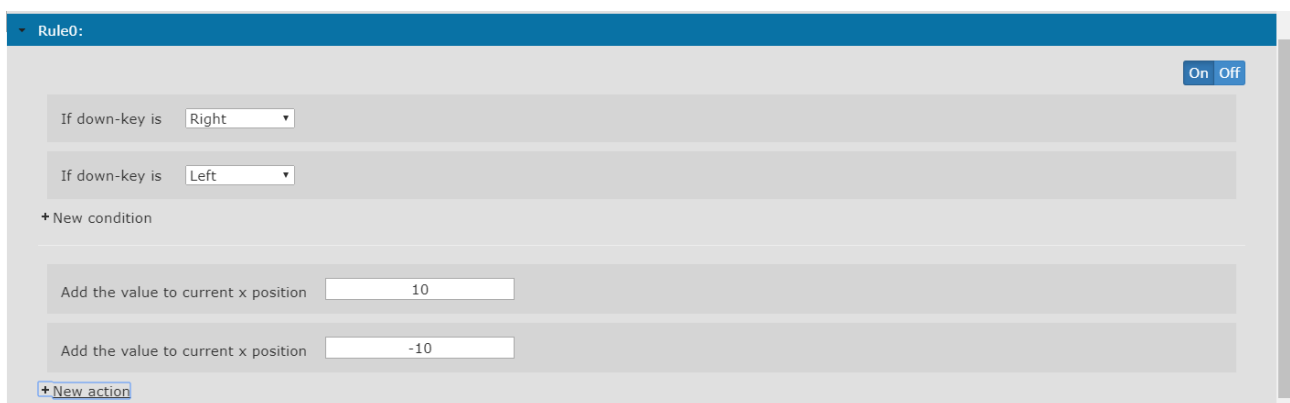
Οι γενικές ρυθμίσεις έχουν να κάνουν με το όνομα του στοιχείου, με το επίπεδο στο οποίο ανήκει, εάν θα είναι ορατό, με το transparency, την θέση του μέσα στην εικόνα και το μέγεθός του, καθώς επίσης και με το χρώμα του, την κλίση του και με το αν σχετίζεται με διάφορες μεταβλητές. Οι ρυθμίσεις της κάμερας δεν έχουν πολλές επιλογές, εξάλλου τα παιχνίδια που υποστηρίζει η Ceilfire είναι μόνο δυο διαστάσεων.



Εικόνα 35- Ρυθμίσεις κάμερας Ceilfire

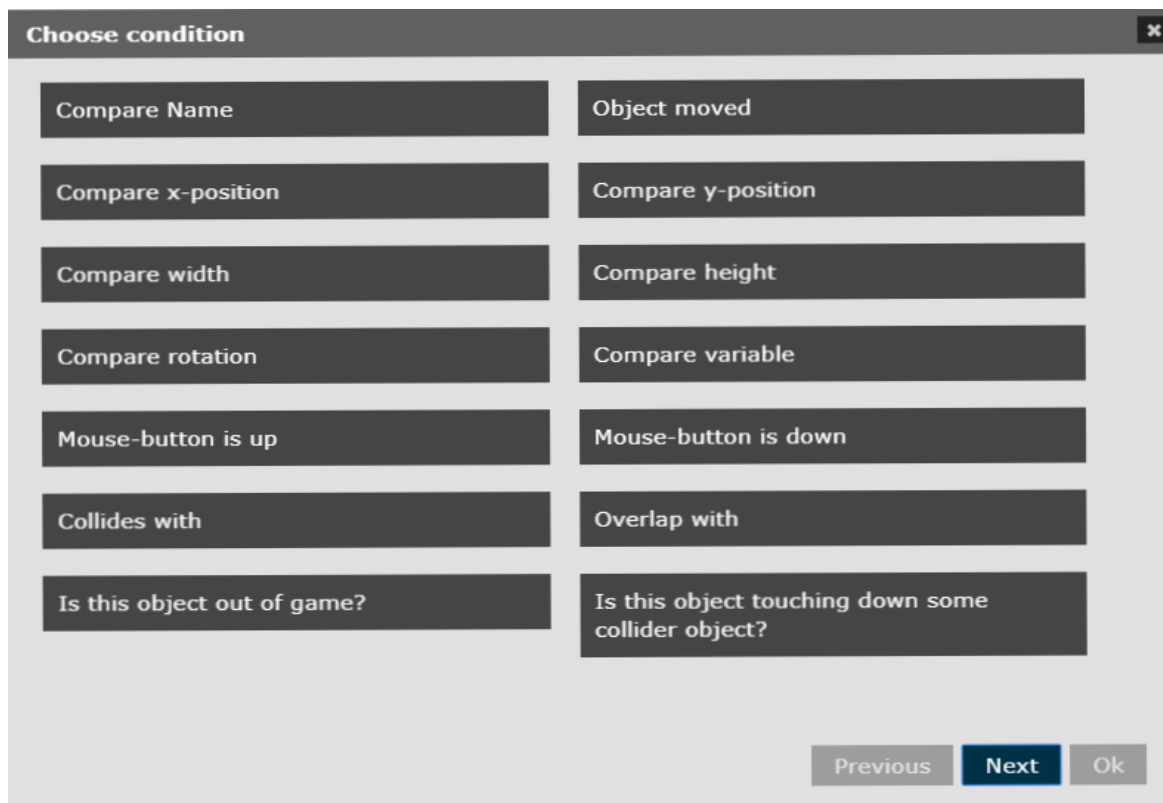
Οι ρυθμίσεις που υπάρχουν στο μενού Physics είναι αυτές που ρυθμίζουν το πώς θα στέκεται ένα στοιχείο μέσα στο παιχνίδι, εάν θα είναι ορατό, εάν θα είναι συμπαγές, την ταχύτητά του και την μάζα του. Μερικές ακόμα ρυθμίσεις είναι αυτή της επιτάχυνσης και του ρυθμού της επιτάχυνσης.

Με την μηχανή Ceilfire η δημιουργία ενός event υλοποιείται μέσα σε λίγα λεπτά, καθώς δεν απαιτείται η ανάπτυξη κώδικα.

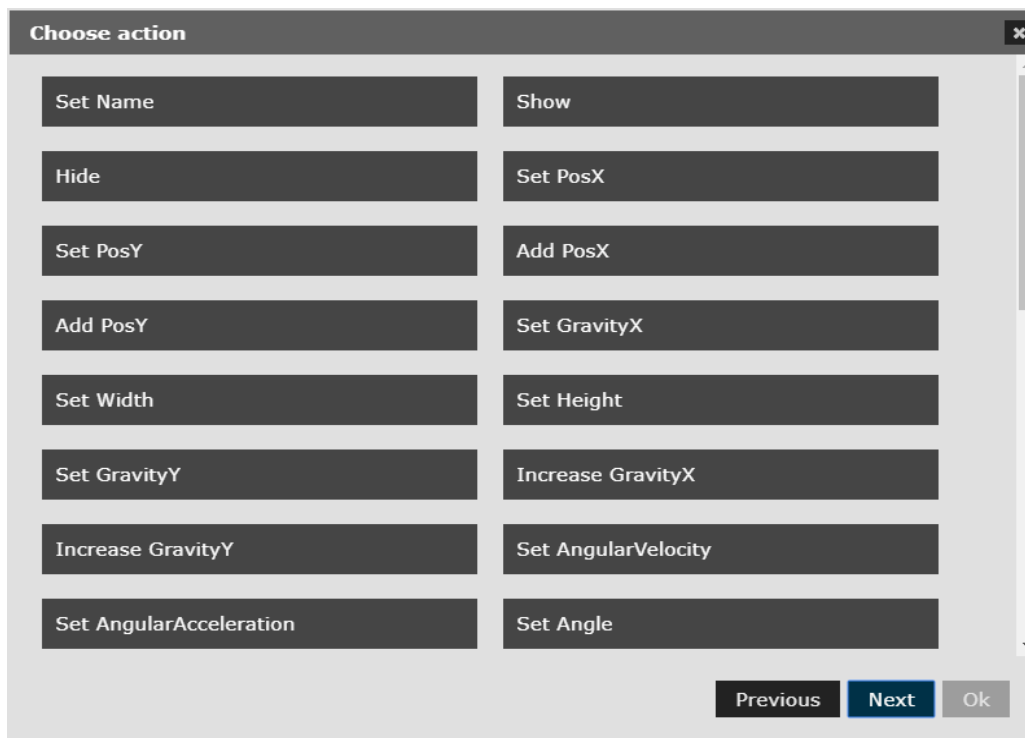


Εικόνα 36- Παράθυρο δημιουργίας event Ceilfire

Η διαδικασία, έχει ως σκοπό την δημιουργία ενός κανόνα που θα ισχύει ανάμεσα σε δυο αντικείμενα, όπως πχ μπορεί να είναι ο χαρακτήρας του παιχνιδιού με το έδαφος. Αυτός ο κανόνας αποτελείται από δυο μέρη, την συνθήκη που πρέπει να ισχύει και την δράση που θα ενεργοποιείται με την ισχύ της συνθήκης. Η μηχανή Ceilfire για να απλοποιήσει ακόμα περισσότερο την ανάπτυξη του παιχνιδιού και να μειώσει τον χρόνο αυτής, προσφέρει δυο βιβλιοθήκες που περιέχουν τις συνηθέστερες λειτουργίες, όπως είναι για παράδειγμα το πάτημα του ποντικιού, το πάτημα ενός πλήκτρου και τα αποτελέσματα, όπως είναι η μετακίνηση στους άξονες των x,y.



Εικόνα 37- Επιλογή συνθήκης για την δημιουργία event Ceilfire



Εικόνα 38- Επιλογή δράσης για την δημιουργία event Ceilfire

3.5 Μηχανή Cocos Creator

3.5.1 Γενικά για τη μηχανή Cocos Creator

Η μηχανή Cocos Creator είναι μια ολοκληρωμένη λύση για την ανάπτυξη παιχνιδιών, η οποία περιλαμβάνει την εφαρμογή Javascript που διαθέτει η μηχανή Cocos2d-x καθώς και διάφορα επιπλέον εργαλεία, που στόχο έχουν την ανάπτυξη νέων παιχνιδιών σε μικρότερο χρόνο. Η μηχανή Cocos είναι ικανή να υποστηρίξει την διαδικασία δημιουργίας ενός παιχνιδιού από την αρχή μέχρι το τέλος. Ο σχεδιασμός, η ανάπτυξη, η προεπισκόπηση του παιχνιδιού, ο εντοπισμός και η διόρθωση σφαλμάτων και τέλος η δημοσίευση του παιχνιδιού σε διάφορες πλατφόρμες, αφού πρόκειται για μια cross-platform μηχανή, πραγματοποιούνται μέσα από τον editor της μηχανής. Ο editor της μηχανής παρέχει, όχι μόνο ένα ισχυρό και πλήρες εργαλείο αλλά και ένα open plugin framework. Αυτό δίνει την δυνατότητα στους χρήστες να διευρύνουν τη λειτουργικότητα του editor που τους προσφέρει εξατομικευμένες λύσεις στην ανάπτυξη των παιχνιδιών, με τη χρήση HTML5 και Javascript.

Τα κυριότερα χαρακτηριστικά της μηχανής Cocos Creator είναι τα εξής:

1. Οι ιδιότητες των δεδομένων μπορούν να προσαρμοστούν στον επεξεργαστή ανά πάσα στιγμή και να δηλωθούν στον κώδικα. Έτσι οι ρυθμίσεις των παραμέτρων μπορούν να ολοκληρωθούν ανεξάρτητα από το σχεδιαστικό τμήμα του παιχνιδιού.

2. Το User Interface υποστηρίζει την έξυπνη προσαρμογή του στοιχείου canvas, αυτό οδηγεί τα στοιχεία του παιχνιδιού να ταξινομούνται ομοιόμορφα σε όλες τις οθόνες συσκευών, οποιαδήποτε ανάλυσης.
3. Το σύστημα animation είναι ειδικά σχεδιασμένο για την ανάπτυξη 2D παιχνιδιών, καθώς υποστηρίζει την προβολή κινούμενης εικόνας σε τροχιά και την επεξεργασία σύνθετων καμπυλών.
4. Υποστηρίζει τη λειτουργία πολλαπλών πλατφορμών
5. Υποστηρίζει τη δημιουργία παιχνιδιών για τις πλατφόρμες Facebook Instant Games και QQ Play
6. Υποστηρίζει την τεχνολογία WebGL

Για την μηχανή Cocos Creator η δημιουργία και η επεξεργασία της σκηνής έχει γίνει ο πυρήνας της ανάπτυξης παιχνιδιών και αυτό χάρη στην ανάλυση δεδομένων, που καθοδηγούν την διαδικασία της ανάπτυξης. Αυτό οδηγεί στην ταυτόχρονη ανάπτυξη του σχεδιαστικού τμήματος και του λειτουργικού τμήματος του παιχνιδιού, χωρίς να δημιουργούνται προβλήματα και έτσι δεν έχει σημασία ποιο τμήμα του παιχνιδιού θα αναπτυχθεί πρώτο.

3.5.2 Υποστηριζόμενοι Browsers Cocos Creator

Στόχος της μηχανής Cocos Creator είναι το παιχνίδι να αναπτυχθεί μια φορά και μετά να δημοσιευτεί σε όλες τις πλατφόρμες. Έτσι λοιπόν η Cocos Creator δίνει την δυνατότητα στους χρήστες της να δημοσιεύουν τα παιχνίδια τους σε περιβάλλοντα Android, iOS, Windows, Mac και σε Instant Games. Η Cocos Creator υποστηρίζει όλους τους web browsers που υποστηρίζει η εκάστοτε πλατφόρμα και λειτουργικό σύστημα [Πίνακας 1].

3.5.3 Κόστος μηχανής Cocos Creator

Η μηχανή Cocos Creator διανέμεται εντελώς δωρεάν και δεν υπάρχουν ειδικές εκδόσεις για επιχειρήσεις και εκπαιδευτικούς οργανισμούς. Και οι δυο αυτές μηχανές, όπως και πολλές άλλες, στηρίζονται στην κοινότητα τους, τόσο για την ανάπτυξή τους, όσο και για την διάδοσή τους. Η Cocos Creator, όπως και η Cocos2d-x από την οποία προέρχεται, διανέμονται με την άδεια MIT.

Η MIT License είναι μία από τις πιο φιλελεύθερες άδειες που υπάρχουν. Δημιουργήθηκε από το Τεχνολογικό Ινστιτούτο της Μασαχουσέτης (MIT) και είναι συμβατή με το GNU General Public License. Δηλαδή η GPL επιτρέπει τη χρήση και αναδιανομή με λογισμικό που χρησιμοποιεί MIT License.

Με την χρήση της, μπορεί κάποιος να αντιγράψει, να τροποποιήσει, να συγχωνεύσει, να εκδώσει, να διανέμει, να εκμισθώσει, να πουλήσει μέρος του λογισμικού, να επιτρέψει το ίδιο και σε αυτούς που έχουν το λογισμικό αυτό, χωρίς να φέρει οποιαδήποτε ευθύνη για οτιδήποτε. Υπάρχουν πολλές εκδόσεις της (wikipedia 2018).

3.5.4 Τρόπος λειτουργίας της μηχανής Cocos Creator

Για την λειτουργία της μηχανής Cocos Creator είναι απαραίτητη η εγκατάσταση του λογισμικού στον υπολογιστή. Η λήψη του λογισμικού της μηχανής γίνεται μέσα από την επίσημη ιστοσελίδα της μηχανής και η εγκατάστασή της ολοκληρώνεται εύκολα και σε λίγα βήματα. Ωστόσο για την χρήση της μηχανής ο χρήστης οφείλει να δημιουργήσει λογαριασμό χρήστη στην επίσημη ιστοσελίδα και με τον κωδικό που έχει ορίσει, να ξεκινήσει να εργάζεται με την μηχανή. Η όλη διαδικασία, από την λήψη της μηχανής, μέχρι την εγκατάσταση και την λειτουργία της, είναι απλοποιημένη και ολοκληρώνεται σε πολύ λίγο χρόνο και φυσικά υπάρχουν οδηγίες, που καθοδηγούν τον χρήστη βήμα-βήμα.

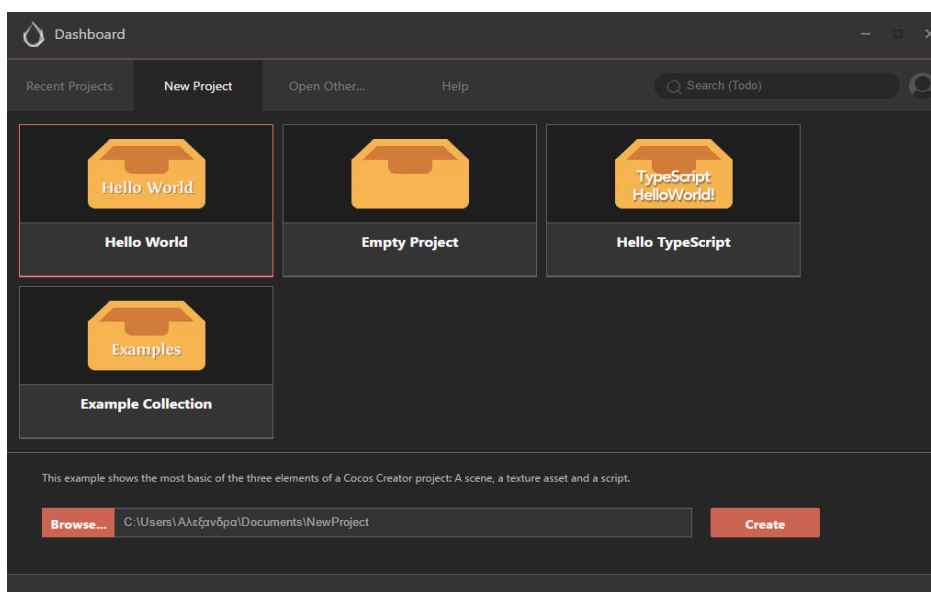
Dashboard

Με την έναρξη της μηχανής ο χρήστης έχει τις εξής επιλογές:

1. Recent Projects: Η επιλογή αυτή, εμφανίζει τα πρόσφατα project με τα οποία έχει ασχοληθεί ο χρήστης
2. New Project: Το μενού από το οποίο ο χρήστης θα ξεκινήσει την δημιουργία του νέου του παιχνιδιού.
3. Open Other: Εάν κάποιο παιχνίδι που θέλει να ανοίξει ο χρήστης, δεν βρίσκεται στα Recent Projects, τότε θα επιλέξει αυτό το μενού.
4. Help: Το μενού βοήθειας, που περιλαμβάνει, tutorials, παραδείγματα και λοιπές πληροφορίες.

New Project

Η μηχανή Cocos Creator δίνει την επιλογή κάποιων προτύπων. Αυτά τα πρότυπα περιλαμβάνουν διάφορα είδη βασικών frameworks, καθώς επίσης περιλαμβάνουν και διάφορα παραδείγματα για να βοηθηθεί ο χρήστης και να ολοκληρώσει το παιχνίδι του πιο γρήγορα.



Εικόνα 39- Dashboard Cocos Creator

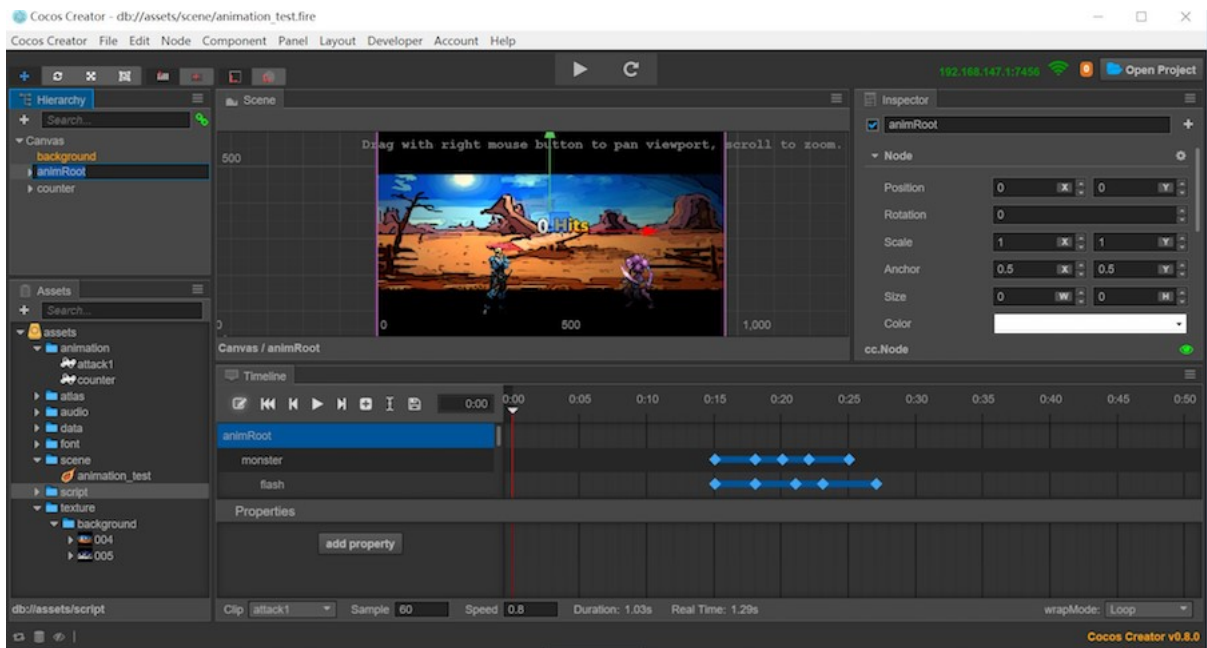
Στο κάτω μέρος του πίνακα, εμφανίζεται το μονοπάτι και το όνομα με το οποίο θα αποθηκευτεί το παιχνίδι. Εφόσον ο χρήστης ρυθμίσει το όνομα και την τοποθεσία που επιθυμεί για το παιχνίδι του, επιλέγει το κουμπί Create. Αυτό οδηγεί στον τερματισμό του Dashboard και στο άνοιγμα του κυρίου παραθύρου της Cocos Creator. Έχουν σημειωθεί αρκετές αναφορές σφάλματος σε αυτό το σημείο της μηχανής. Ο τερματισμός και η εκτέλεση της μηχανής ως Διαχειριστής διορθώνει το σφάλμα. Τέλος πρέπει να σημειωθεί ότι με την δημιουργία του νέου project, δεν εμφανίζεται απλά ένα αρχείο στο φάκελο προορισμού που έχει επιλέξει ο χρήστης. Δημιουργείται ένας νέος φάκελος με το όνομα που έχει επιλέξει ο χρήστης. Μέσα στον φάκελο αυτόν υπάρχουν συγκεκριμένοι υποφάκελοι, ίδιοι σε κάθε project:

- **Assets:** Στον φάκελο αυτόν ο χρήστης φορτώνει όλα τα στοιχεία που επιθυμεί να υπάρχουν στο παιχνίδι
- **Library:** Ο φάκελος αυτός δημιουργείται μόλις το project ανοιχθεί για πρώτη φορά. Σε αυτό τον φάκελο όλα τα στοιχεία μετονομάζονται με ένα μοναδικό κωδικό.
- **Local:** Στον φάκελο αυτόν περιέχονται οι ρυθμίσεις του editor, όπως για παράδειγμα το μέγεθος και τη θέση του παραθύρου. Δεν πρέπει να τροποποιείται το περιεχόμενο αυτού του φακέλου.
- **Settings:** Εδώ περιέχονται οι ρυθμίσεις που αφορούν το κάθε project και συγκεκριμένα τις ρυθμίσεις του Build Panel.
- **project.json:** Το αρχείο αυτό περιέχει τον τρέχοντα αναγνωριστικό αριθμό της μηχανής και τον τοπικό φάκελο που βρίσκεται ο Editor.

Ο σκοπός των υποφακέλων είναι ίδιος για κάθε project και ο χρήστης πρέπει να είναι πολύ προσεκτικός στην διαχείρισή τους, διότι η σωστή χρήση των φακέλων, συνεπάγεται και την σωστή λειτουργία του παιχνιδιού.

Cocos Creator Editor

Ο editor της μηχανής Cocos Creator αποτελείται από πάνελ, μενού και κουμπιά, τα οποία μπορούν να μετακινηθούν και να συνδυαστούν σύμφωνα με την βούληση του κάθε χρήστη και με στόχο πάντα να καλύψουν τις διαφορετικές ανάγκες του κάθε χρήστη και του κάθε project.



Εικόνα 40- Editor Cocos Creator

Asset Manager

Στο τμήμα αυτό εμφανίζονται όλα τα στοιχεία που βρίσκονται στον υποφάκελο assets του παιχνιδιού. Η παρουσίαση τους είναι ως δενδροδιάγραμμα και με την εισαγωγή ενός νέου στοιχείου στον φάκελο, αυτόματα ενημερώνεται και το δενδροδιάγραμμα. Σε αυτό το τμήμα ο χρήστης έχει την επιλογή διαφόρων λειτουργιών, όπως προσθήκη, αντιγραφή, διαγραφή, να επιλέξει πολλά αντικείμενα και να τα μετακινήσει. Ακόμα μπορεί να κάνει αναζήτηση των στοιχείων, των κωδικών τους ή να φιλτράρει τα στοιχεία βάση του τύπου τους και να τα εμφανίσει ως λίστα.

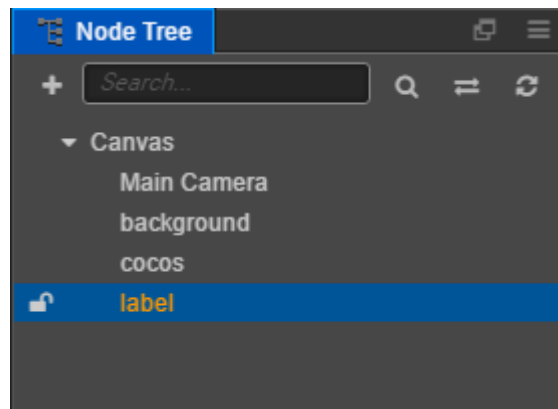
Scene Editor

Ο Scene Editor, όπως και στις προηγούμενες μηχανές παιχνιδιών, χρησιμοποιείται για την εμφάνιση και την επεξεργασία της ορατής στον παίκτη, σκηνής. Πρόκειται για τον βασικό χώρο εργασίας του χρήστη. Σε αυτόν τον χώρο τοποθετούνται προς επεξεργασία οι εικόνες, οι χαρακτήρες, τα ειδικά εφέ, το UI του παιχνιδιού και όλα τα υπόλοιπα στοιχεία που αποτελούν το παιχνίδι.

Node Tree

Στο τμήμα του Node Tree εμφανίζονται όλοι οι κόμβοι που υπάρχουν στην τρέχουσα ανοιχτή σκηνή, ταξινομημένοι ιεραρχικά και ανεξάρτητα αν ο κόμβος περιλαμβάνει ορατές εικόνες. Το περιεχόμενο του Scene Editor μαζί με το περιεχόμενο του Node Tree κατά την επεξεργασία της σκηνής συγχρονίζεται στην οθόνη και έτσι δημιουργείται η σκηνή του παιχνιδιού. Και σε αυτό το τμήμα ισχύουν οι λειτουργίες της δημιουργίας, της αντιγραφής, της μετακίνησης, της μετονομασίας, της διαγραφής. Το πιο σημαντικό στοιχείο του Node Tree όμως είναι πως

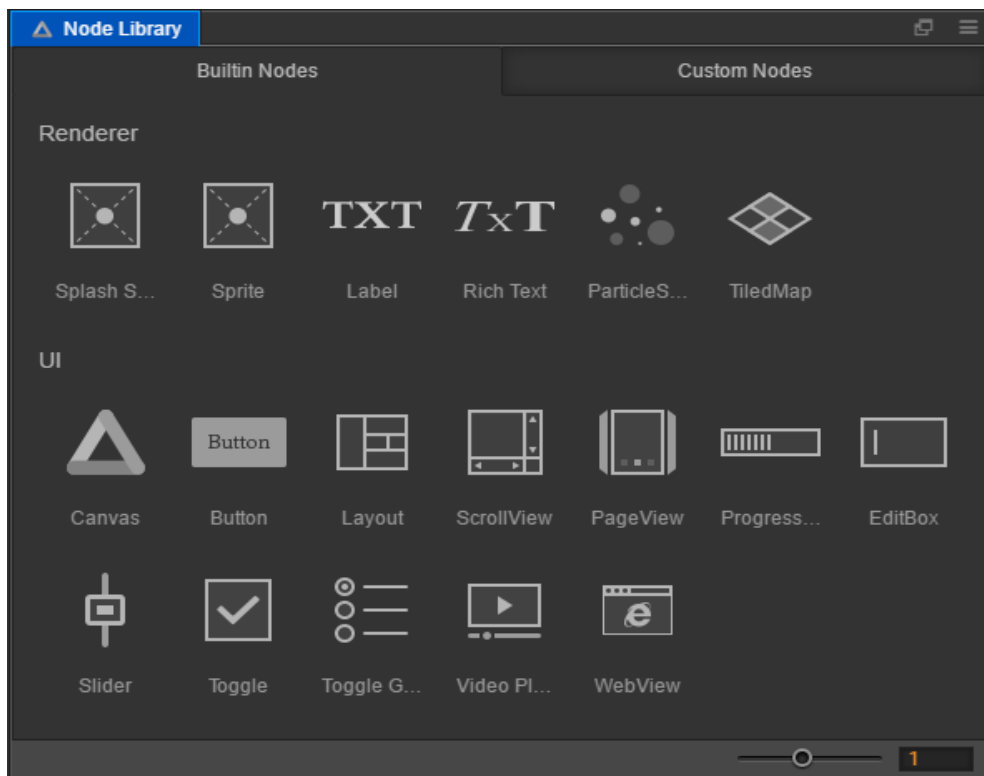
ανάμεσα στους κόμβους ισχύει η σχέση γονέα-παιδιού. Χρειάζεται προσοχή λοιπόν στην επεξεργασία των κόμβων, καθώς ο κόμβος παιδί κληρονομεί τις ιδιότητες του κόμβου γονέα.



Εικόνα 41- Node Tree Cocos Creator

Node Library

Η Node Library είναι μια βιβλιοθήκη έτοιμων κόμβων. Αυτό σημαίνει ότι ο χρήστης πολύ απλά, άμεσα και οπτικά, μπορεί να μεταφέρει μέσα στην σκηνή οποιοδήποτε κόμβο χρειάζεται και να ολοκληρώσει τάχιστα την διαδικασία δημιουργίας κόμβου.



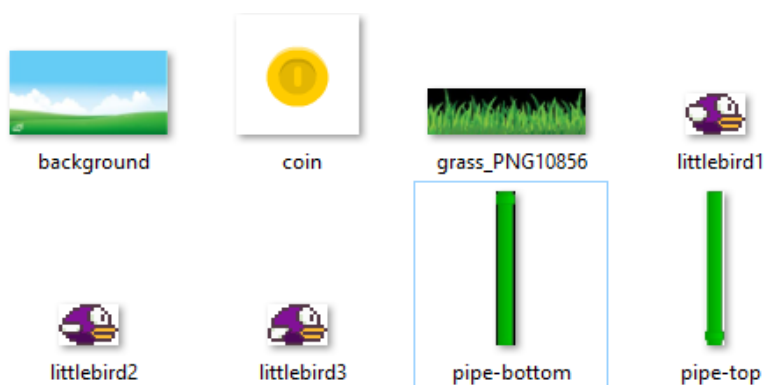
Εικόνα 42- Βιβλιοθήκη στοιχείων Cocos Creator

Η βιβλιοθήκη περιέχει δυο ενότητες, την Builtin Nodes και την Custom nodes. Στην πρώτη, οι κόμβοι που προσφέρονται προς χρήση είναι έτοιμοι και επιλεγμένοι από την μηχανή. Στην δεύτερη ο χρήστης είναι αυτός που δημιουργεί την δική του βιβλιοθήκη με τα στοιχεία-κόμβους που επιθυμεί.

4. Ανάπτυξη Παιχνιδιών

Σε αυτό το κεφάλαιο παρουσιάζεται η ανάπτυξη του ίδιου παιχνιδιού ξεχωριστά και στις πέντε μηχανές που παρουσιάστηκαν στο κεφάλαιο 3. Το παιχνίδι που επιλέχθηκε βασίζεται στο γνωστό 2D παιχνίδι Flappy Bird. Το παιχνίδι αυτό έχει έναν χαρακτήρα ο οποίος κινείται με το ποντίκι και ο σκοπός του είναι να περνάει μέσα από εμπόδια υπό τη μορφή σωλήνων και να μαζεύει πόντους. Η δημιουργία των παιχνιδιών σε κάθε μηχανή θα υλοποιηθεί σύμφωνα με τα tutorials που υπάρχουν στις επίσημες ιστοσελίδες των μηχανών. Το παιχνίδι Flappy Bird είναι αρκετά δημοφιλές και κυκλοφορούν πολλές εκδοχές και παραλλαγές του. Σκοπός είναι να γίνει μια σύγκριση ανάμεσα στις μηχανές, όχι μόνο για τα τεχνικά χαρακτηριστικά τους αλλά και για την ευκολία χρήσης τους, τι δυνατότητες προσφέρουν στον χρήστη και γενικά την συνολική τους απόδοση.

Είναι συνετό, πριν την έναρξη των εργασιών, ο χρήστης να έχει βρει τις εικόνες που θέλει να χρησιμοποιήσει στο παιχνίδι του. Για τις ανάγκες του παιχνιδιού θα χρησιμοποιηθούν οι παρακάτω εικόνες, οι οποίες αντλήθηκαν από το διαδίκτυο.



Εικόνα 43- Οι εικόνες των παιχνιδιών

4.1 Ανάπτυξη Παιχνιδιού στην PlayCanvas

Για την ανάπτυξη του παιχνιδιού Flappy Bird στην μηχανή PlayCanvas, θα βασιστούμε στα γενικά tutorials που υπάρχουν και στο ήδη υπάρχον παιχνίδι που δημιουργήθηκε με την PlayCanvas (<https://playcanvas.com/project/375389/overview/flappy-bird>). Βασιζόμενοι στα στοιχεία του Project θα κατασκευάσουμε εκ νέου το παιχνίδι, ενώ τα σενάρια του χρησιμοποιήθηκαν αυτούσια. Έχοντας ολοκληρώσει την διαδικασία εγγραφής και σύνδεσης προχωράμε στην δημιουργία νέου Project. Η δημιουργία ενός παιχνιδιού στην PlayCanvas χωρίζεται σε τρία κύρια μέρη:

- I. Δημιουργία αντικειμένου
- II. Παραμετροποίηση του αντικειμένου
- III. Δημιουργία σεναρίου, εάν αυτό είναι απαραίτητο.

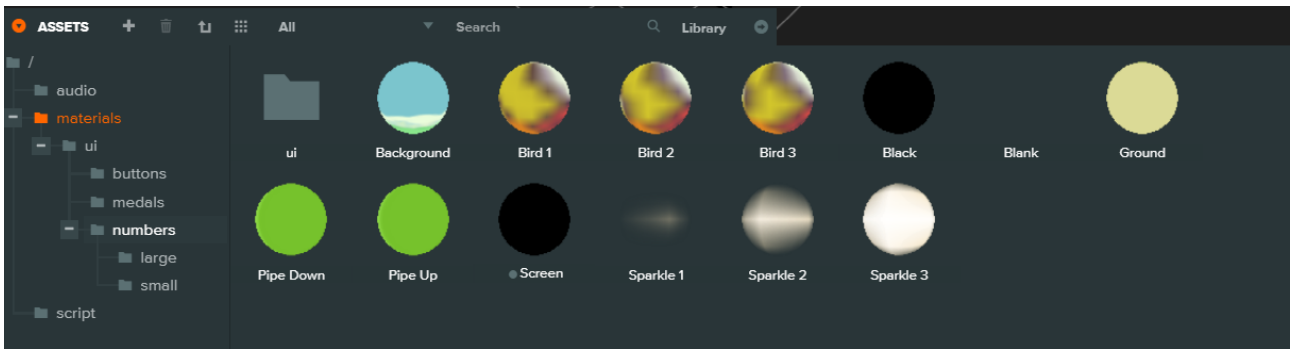
Αυτά τα τρία βήματα επαναλαμβάνονται μέχρι να ολοκληρωθεί η δημιουργία όλων των απαραίτητων αντικειμένων.

4.1.1. Δημιουργία Αντικειμένου

Μέσω του πάνελ Hierarchy έχουμε τη δυνατότητα να δημιουργήσουμε τις οντότητες ή διαφορετικά, τα αντικείμενα του παιχνιδιού. Ένα αντικείμενο είναι η κάμερα, ενώ ένα άλλο είναι ο σωλήνας που περνάει ο χαρακτήρας του παιχνιδιού. Με την δημιουργία μιας νέας οντότητας μπορούμε να επιλέξουμε τι σχήμα θα έχει, εάν θα πρόκειται για ήχο, σενάριο ή αντικείμενο φωτισμού. Προτού προχωρήσουμε στις λειτουργικές ρυθμίσεις που θα έχει ένα αντικείμενο μέσα στο παιχνίδι, θα πρέπει να ρυθμίσουμε την εμφάνιση που θα έχει. Αυτό επιτυγχάνεται στο πεδίο Assets με την δημιουργία των Materials.

i. Assets

Στο πεδίο Assets βρίσκονται όλα τα αρχεία που υπάρχουν στο παιχνίδι, όπως οι εικόνες και τα αρχεία ήχου. Πρόκειται ουσιαστικά για την βιβλιοθήκη των στοιχείων μας και είναι απαραίτητο να την οργανώσουμε με τον καλύτερο δυνατό τρόπο ώστε να είναι λειτουργική και εύκολα διαχειρίσιμη όπως φαίνεται στο παράδειγμα της εικόνας 15. Τα στοιχεία που θα αποτελούν το παιχνίδι είναι οι ήχοι και ένα spritesheet. Μια εικόνα δηλαδή που αποτελείται από μικρότερες εικόνες και βάση των ρυθμίσεων θα εμφανίζεται κάθε φορά ένα διαφορετικό στιγμιότυπο. Σύμφωνα με το παράδειγμα, ακολουθήθηκε η παρακάτω ιεράρχηση των στοιχείων:



Εικόνα 44- Η ιεράρχηση των στοιχείων του παιχνιδιού

Μέσα στον αρχικό φάκελο (/) βρίσκονται οι υποφάκελοι και το spritesheet. Οι υποφάκελοι αποτελούνται από τον φάκελο audio, που περιέχει όλα τα αρχεία ήχου, από τον φάκελο materials και τους υποφακέλους του όπου βρίσκονται όλα τα στοιχεία που αποτελούν το παιχνίδι, όπως είναι το background και τα νούμερα. Τέλος υπάρχει ο φάκελος script όπου είναι αρχειοθετημένα όλα τα σενάρια του παιχνιδιού.

4.1.2. Παραμετροποίηση του αντικειμένου

ii. Materials

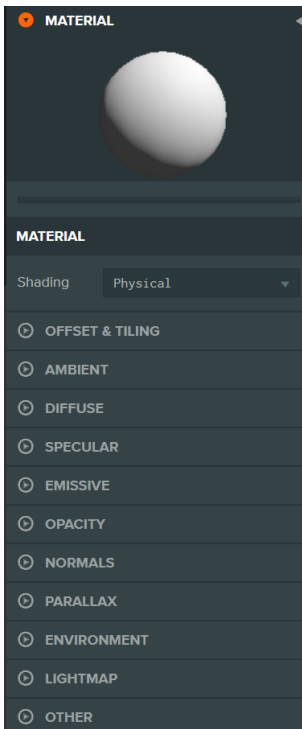
Για την δημιουργία ενός Material στο πεδίο Assets ακολουθείται η εξής διαδικασία:

Δεξί κλικ → New Asset → Material. Αυτές οι κινήσεις οδήγησαν στην εμφάνιση του στοιχείου που φαίνεται στην εικόνα:



Εικόνα 45- New material

Εάν επιλέξουμε αυτό το στοιχείο, στο δεξί μέρος της οθόνης θα εμφανιστούν όλες οι πληροφορίες όπως το όνομα και το ID, καθώς και όλες οι ρυθμίσεις που μπορούμε να τροποποιήσουμε. Το ID του κάθε στοιχείου είναι μοναδικό και δεν μπορούμε να παρέμβουμε.



Εικόνα 46- Οι ρυθμίσεις ενός material στοιχείου

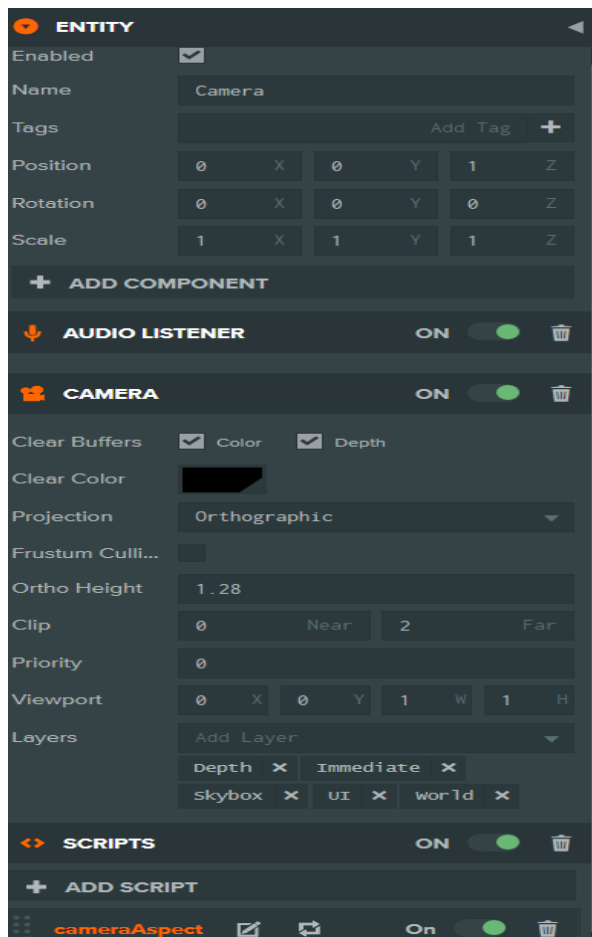
Για την ρύθμιση του στοιχείου background θα παραμετροποιήσουμε τις ρυθμίσεις Offset & Tiling και Emissive. Η πρώτη ρύθμιση αφορά τις τιμές αντιστάθμισης, οριοθέτησης και μετατόπισης πάνω στο spritesheet. Με αυτόν τον τρόπο ρυθμίζουμε ποιο σημείο του spritesheet θα εμφανίζει το στοιχείο. Η δεύτερη ρύθμιση αφορά τον τρόπο με τον οποίο το υλικό θα εκπέμπει φως. Αυτές είναι και οι ρυθμίσεις που θα γίνουν σε όλα τα materials εκτός ορισμένων στοιχείων όπου θα παραμετροποιήσουμε και το Opacity, το οποίο έχει να κάνει με το επίπεδο διαφάνειας. Ένα material μπορεί να χρησιμοποιηθεί σε περισσότερα από ένα αντικείμενα.

iii. Hierarchy

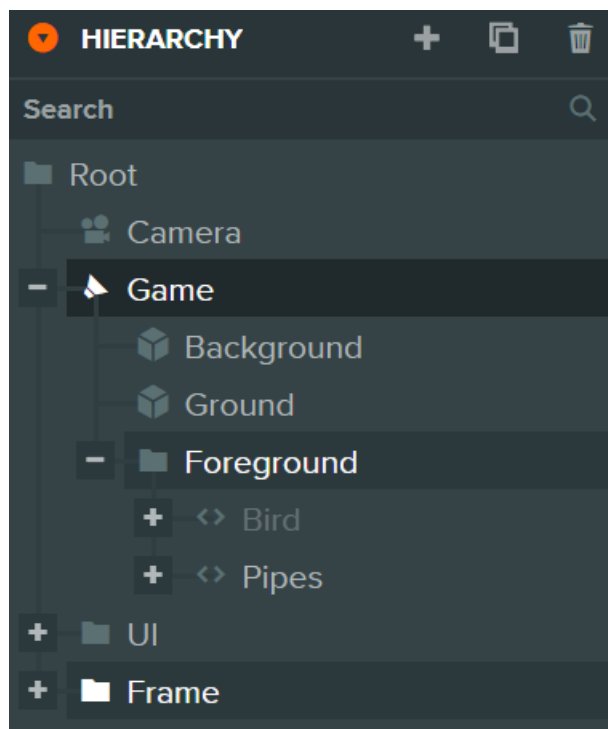
Ολοκληρώνοντας την διαδικασία ρυθμίσεων της εμφάνισης της οντότητας, επιστρέφουμε στο πεδίο Hierarchy ώστε αυτή τη φορά να ρυθμίσουμε την λειτουργική υπόσταση του αντικειμένου μέσα στο παιχνίδι. Η δομή του δένδροδιαγράμματος στο πεδίο Hierarchy αποτελείται από τέσσερις οντότητες:

- Camera
- Game
- UI
- Frame

Στην οντότητα της Camera, έχουμε ρυθμίσει την θέση που θα έχει καθ' όλη την διάρκεια του παιχνιδιού και έχουμε ενεργοποιήσει τον ήχο και το σενάριο που θα εκτελούνται. Οι υπόλοιπες οντότητες, είναι κύριες οντότητες, έχουν δηλαδή το ρόλο του γονέα.

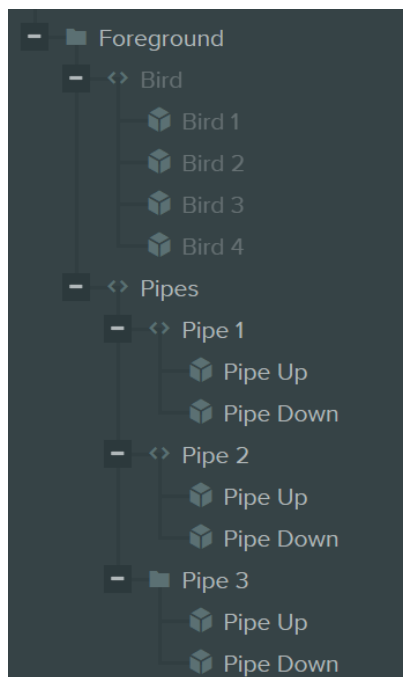


Εικόνα 47- Παραμετροποίηση της οντότητας Camera



Εικόνα 48- Δενδροδιάγραμμα οντοτήτων

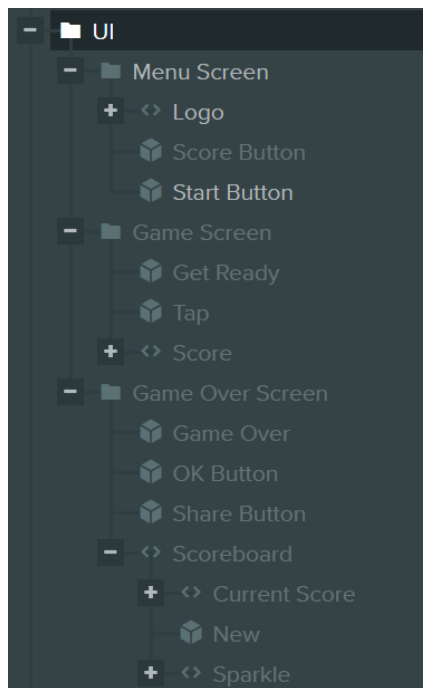
Η οντότητα Game περιλαμβάνει τα μοντέλα Background, Ground και Foreground, η οποία με τη σειρά της έχει τα «παιδιά» Bird και Pipes. Αντίστοιχα η οντότητα Bird είναι γονέας των οντοτήτων Bird1, Bird2, Bird3, Bird4, όπου χάρη σε αυτά δίνεται η αίσθηση του πετάγματος, ενώ η οντότητα Pipes είναι γονέας των παιδιών Pipe1, Pipe2 και Pipe3, που με τη σειρά τους περιλαμβάνουν της οντότητες PipeUp και PipeDown



Εικόνα 49- Η οντότητα Foreground

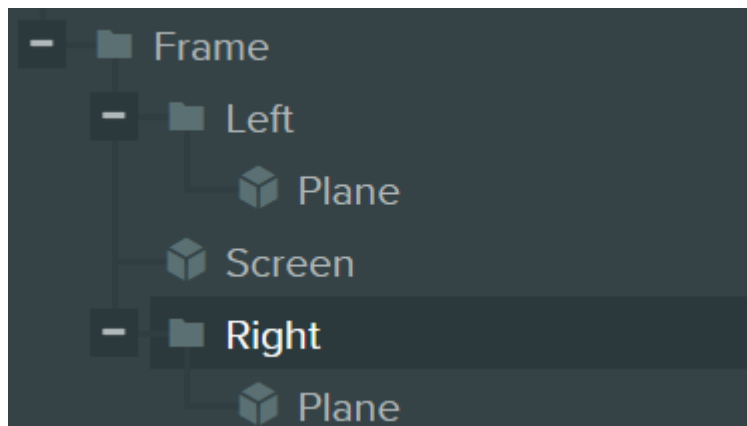
Εκτός από την επιλογή του material , δηλαδή την εμφάνιση της κάθε οντότητας, ρυθμίζεται η θέση της μέσα στην σκηνή και ο τύπος του μοντέλου (όλες οι οντότητες είναι τύπου Plane). Ορισμένες οντότητες φέρουν και σενάρια. Τα σενάρια αυτά, είναι υπεύθυνα για την επαναλαμβανόμενη κίνηση των σωλήνων και του χαρακτήρα.

Η οντότητα UI περιλαμβάνει όσες οντότητες έχουν σχέση με τις κινήσεις που έχει τη δυνατότητα να πραγματοποιήσει ο παίκτης, όπως η κίνηση του χαρακτήρα ή επιλογή να ξεκινήσει το παιχνίδι.

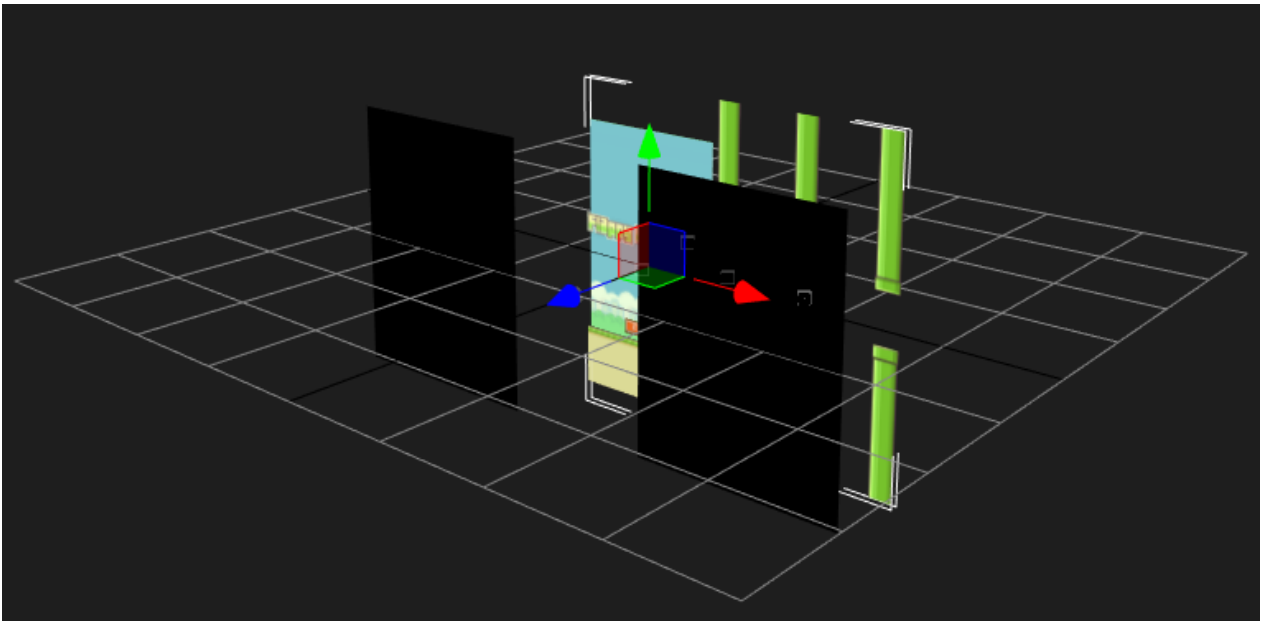


Εικόνα 50- Η οντότητα UI

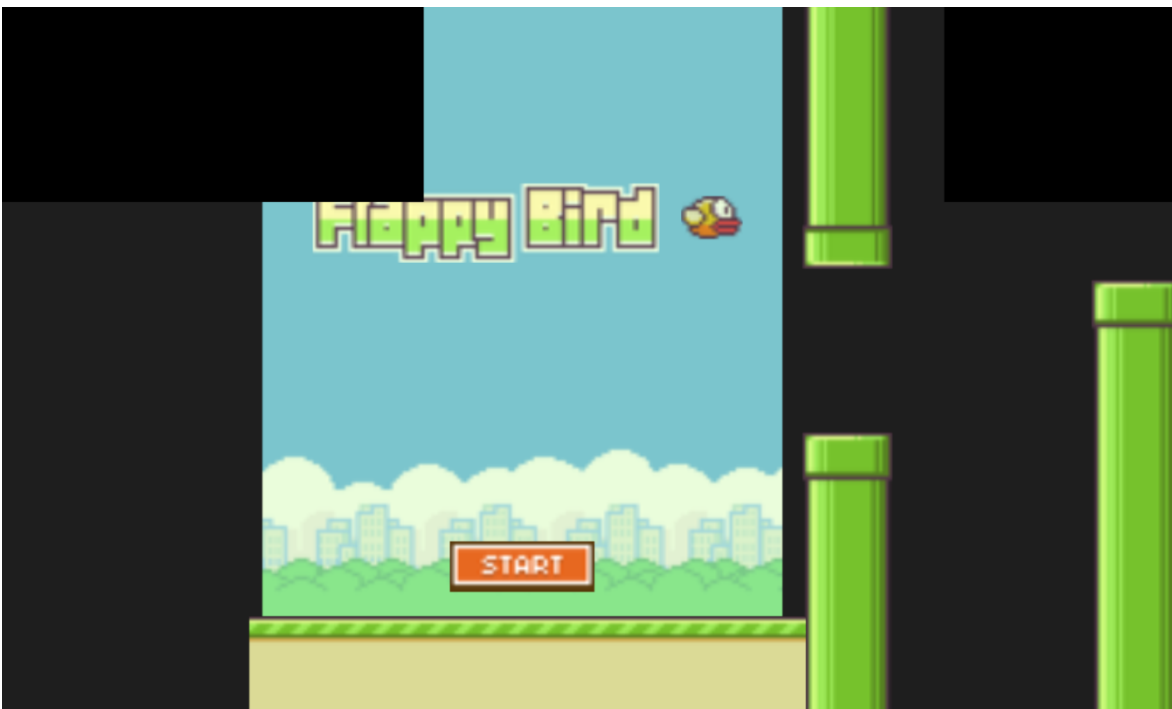
Η οντότητα Frame περιλαμβάνει τις οντότητες που σχετίζονται με την οριοθέτηση της οθόνης που παρουσιάζεται στον χρήστη. Με την χρήση των δυο μαύρων πάνελ, όπως φαίνονται στην εικόνα 52, επιτυγχάνεται ο χρήστης να βλέπει την κίνηση του εδάφους και των σωλήνων, χωρίς να βλέπει την αρχή και το τέλος των κινήσεων αυτών. Εάν δεν είχαν τοποθετηθεί τα πάνελ αυτά η εικόνα που θα αντίκριζε θα ήταν αυτή της εικόνας 53.



Εικόνα 51- Η οντότητα Frame



Εικόνα 52- Το Viewport του παιχνιδιού



Εικόνα 53- Στιγμιότυπο του παιχνιδιού με τα πάνελ σε τυχαία θέση

Το τελικό παιχνίδι:

<https://playcanv.as/p/Rd7GdZRw/>

4.2 Ανάπτυξη Παιχνιδιού στην Construct 3

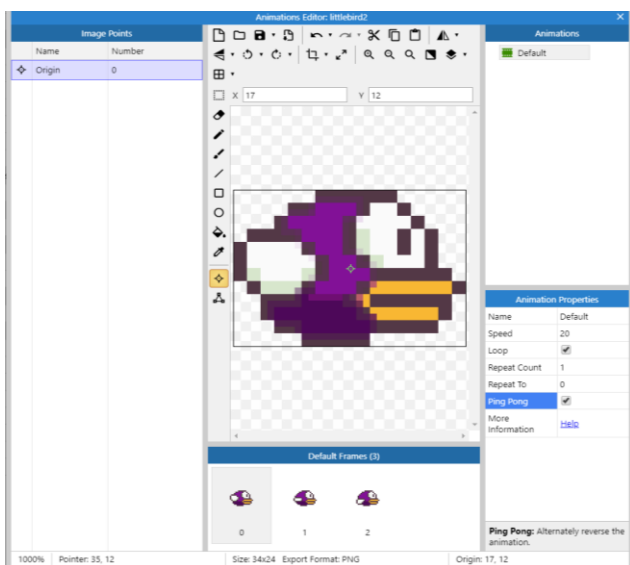
Για να ξεκινήσει κάποιος την δημιουργία ενός παιχνιδιού στην μηχανή Construct 3 πρέπει αρχικά να δημιουργήσει έναν λογαριασμό, για να έχει περισσότερα δικαιώματα πάνω στην μηχανή, όπως έχει ήδη αναφερθεί. Εφόσον ολοκληρώσει την διαδικασία σύνδεσης, ο χρήστης μπορεί να

εισέλθει στον Editor της μηχανής (<https://editor.construct.net/>). Με την επιλογή New Project και τον καθορισμό του ονόματος του παιχνιδιού, ο χρήστης οδηγείται στο περιβάλλον όπου θα αναπτύξει το παιχνίδι του, όπως αυτό φαίνεται και στην εικόνα 16.

Το πρώτο βήμα είναι να ορίσουμε τις διαστάσεις που θα έχει το παιχνίδι. Για να εμφανίζεται η εικόνα του παιχνιδιού σε όλο το εύρος που θέλουμε, πρέπει να ορίσουμε τις ίδιες διαστάσεις σε δυο πεδία ιδιοτήτων. Στο πεδίο Size του στοιχείου Layout 1 και στο πεδίο Viewport size του στοιχείου Project. Εάν επιλέξουμε διαφορετικές τιμές, τότε μέσα στο πεδίο δράσης του παιχνιδιού θα εμφανίζονται κάποια κενά σημεία που δεν θα καλύπτονται από το background. Οι τιμές που επιλέγουμε είναι 800x480. Στη συνέχεια, τοποθετούμε την εικόνα που επιλέξαμε ως background και μετά επιλέγουμε ταυτόχρονα τις τρεις εικόνες littlebird για να μπορέσουμε μετά να τις επεξεργαστούμε ως animation. Εάν μαζί με τις εικόνες littlebird επιλέγαμε και την εικόνα του background, τότε θα δημιουργούνταν ένα και μόνο animation. Η σειρά με την οποία γίνεται το ανέβασμα των εικόνων έχει σημασία, καθώς σύμφωνα με αυτή τη σειρά εμφανίζονται και στον editor. Εάν δηλαδή ανεβάζαμε πρώτα τις εικόνες little bird και μετά το background, αυτό θα τις υπερκάλυπτε και δεν θα φαινόντουσαν. Αυτό βέβαια είναι κάτι που διορθώνεται κάνοντας δεξί κλικ πάνω στην εικόνα που θέλουμε να εμφανίζεται μπροστά και επιλέγοντας Z order → Send to top of layer. Η διαδικασία είναι αρκετά εύκολη όταν οι εικόνες είναι μόνο δυο όπως στο παράδειγμα (οι τρεις εικόνες, εφόσον επιλέχθηκαν μαζί, είναι πλέον ένα animation), ωστόσο με περισσότερα στοιχεία είναι πιθανό να γίνει χρονοβόρα.

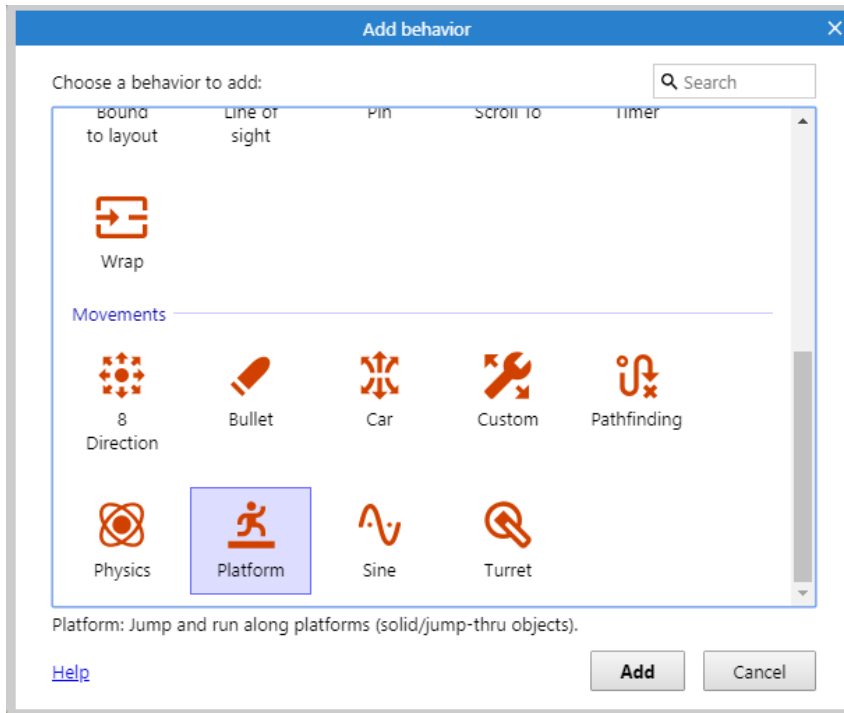
4.2.1. Littlebird

Εν συνεχεία θα δώσουμε την κίνηση στα φτερά του animation κάνοντας δεξί κλικ → Edit. Με αυτό τον τρόπο θα εμφανιστεί ένα νέο παράθυρο από το οποίο μπορούμε να ρυθμίσουμε την ταχύτητα των frames και την επαναλαμβανόμενη κίνηση του animation.



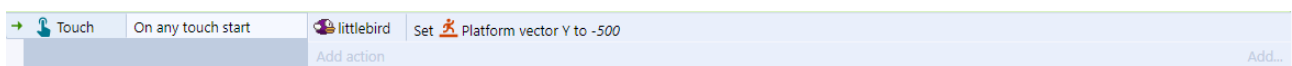
Εικόνα 54- Ιδιότητες animation Construct3

Για να μπορεί το στοιχείο του animation να έχει κίνηση στον άξονα των x,y θα πρέπει να του προσθέσουμε συμπεριφορά κάνοντας πάλι δεξί κλικ → Add → Behavior και να επιλέξουμε την συμπεριφορά Platform που έχει ως ιδιότητα την κίνηση του αντικειμένου με άλματα και ταχύτητα.

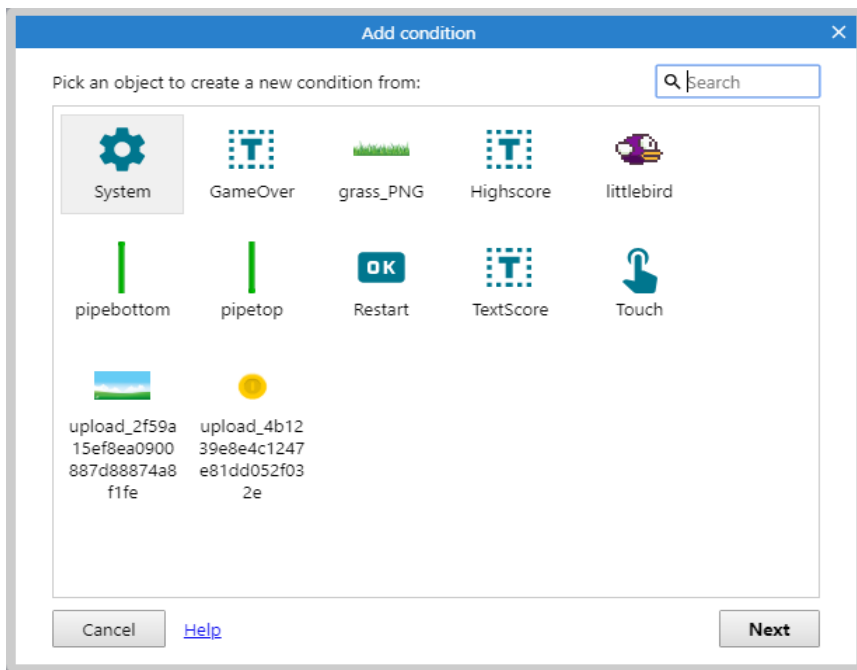


Εικόνα 55- Επιλογή συμπεριφοράς

Για να ολοκληρώσουμε την κίνηση, από το πάνελ Properties ρυθμίζουμε το πεδίο Gravity (η επιτάχυνση που θα έχει το αντικείμενο καθώς θα πέφτει) και απενεργοποιούμε το πεδίο Default Controls, ώστε να μην μπορούν να ενεργοποιήσουν την κίνηση τα πλήκτρα με τα βέλη. Το παιχνίδι παίζεται αποκλειστικά με την χρήση του ποντικιού. Για να συνδέσουμε το ποντίκι λοιπόν με το παιχνίδι θα κάνουμε δεξί κλικ οπουδήποτε μέσα στο layout και θα επιλέξουμε Insert New Object. Από το παράθυρο που θα μας εμφανιστεί [εικόνα 19] θα επιλέξουμε το αντικείμενο Touch. Ο λόγος που δεν επιλέγουμε το αντικείμενο Mouse είναι διότι με το αντικείμενο Touch, αυτόματα χωρίς καμία άλλη ρύθμιση το παιχνίδι μας θα παίζεται και σε συσκευές με οθόνη αφής. Στη συνέχεια ολοκληρώνουμε την διαδικασία με την δημιουργία ενός event όπου θα ενεργοποιεί την κίνηση με κάθε κλικ του ποντικιού.



Εικόνα 56- Το event της κίνησης



Εικόνα 57- Παράθυρο επιλογής αντικειμένου για την δημιουργία event

Σε κάθε κλικ του ποντικιού λοιπόν, το littlebird θα μετακινείται στον άξονα y κατά 500 pixels ανά δευτερόλεπτο. Πιο αναλυτικά, στην καρτέλα Event Sheet [Εικόνα 18] επιλέξαμε το Add Event, από την καρτέλα που εμφανίστηκε [Εικόνα 57], επιλέξαμε το αντικείμενο Touch. Στο επόμενο παράθυρο που εμφανίστηκε επιλέξαμε ποια είναι η συνθήκη που θα ενεργοποιεί το event και αυτή δεν είναι άλλη από την «On any touch start». Τη δράση που θα πυροδοτείται από το κάθε κλικ του ποντικιού θα την επιλέξουμε κάνοντας κλικ πάνω στην επιλογή Add Action [Εικόνα 58]. Στο παράθυρο που θα εμφανιστεί, θα παρουσιάζονται πάλι όλα τα στοιχεία που έχουμε βάλει μέχρι στιγμής στο παιχνίδι (η διαφορά με την εικόνα , είναι πως στον τίτλο θα γράφει Add Action) ώστε να επιλέξουμε ποιο στοιχείο θα εκτελεί την δράση, στην προκειμένη περίπτωση το littlebird. Τέλος επιλέγουμε την δράση η οποία είναι από την ενότητα Platform η Set Vector Y. Επιλέγοντάς την, μας εμφανίζει το επόμενο παράθυρο στο οποίο θα ρυθμίσουμε τα pixels και έτσι θα ολοκληρωθεί το event. Όσο πιο μικρός ο αριθμός των pixels τόσο πιο μικρό άλμα θα κάνει το littlebird.



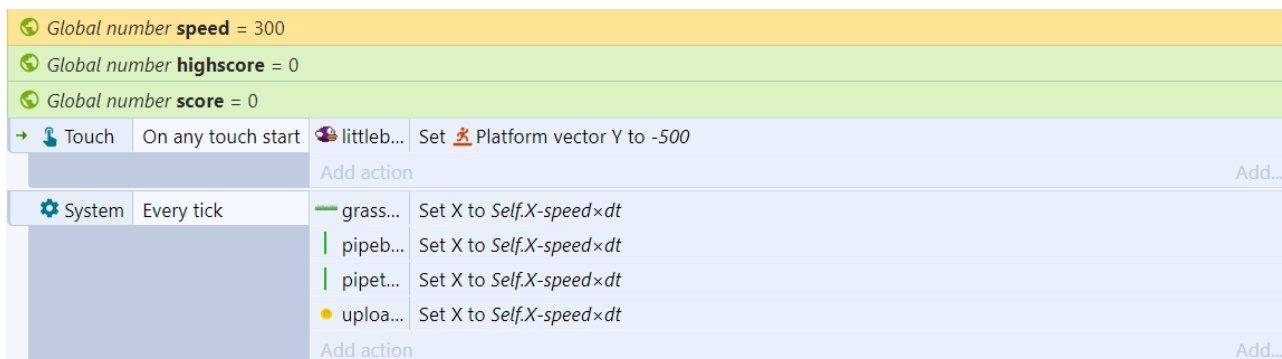
Εικόνα 58- Προσθήκη δράσης στο event

4.2.2. Γρασίδι, σωλήνες και νομίσματα

Για να ολοκληρωθεί το σκηνικό του παιχνιδιού πρέπει να προσθέσουμε το γρασίδι, τους σωλήνες και τα νομίσματα. Ο ρόλος του γρασιδιού είναι πολύ σημαντικός και αυτό διότι το γρασίδι δεν θα είναι στάσιμο και θα δίνει την αίσθηση της κίνησης, καθώς επίσης θα είναι ένα από τα αντικείμενα όπου θα συγκρούεται το στοιχείο littlebird. Για να δοθεί η κίνηση λοιπόν, τοποθετούμε

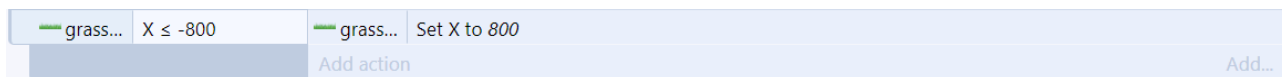
την εικόνα του γρασιδιού μέσα στον editor και την μεγαλώνουμε ή μικραίνουμε τόσο ώστε να επεκτείνεται σε όλο τον άξονα x. Με την διαδικασία δεξί κλικ→ Edit, ανοίγεται το παράθυρο διαλόγου για τις ρυθμίσεις του Animation. Από το πάνελ Image Point και κάνοντας δεξί κλικ, επιλέγουμε Quick Assign→Left. Αυτή η ρύθμιση μας επιτρέπει να ορίσουμε τις συντεταγμένες x,y του αντικειμένου και καθορίσουμε το σημείο εκκίνησης. Αντιγράφουμε το στοιχείο του γρασιδιού και το επικολλούμε το ένα δίπλα στο άλλο. Με αυτόν τον τρόπο, το πρώτο γρασίδι ξεκινάει από την αριστερή πλευρά της οθόνης και χάνεται προς τα δεξιά και το δεύτερο γρασίδι ξεκινάει από την δεξιά πλευρά της οθόνης και ακολουθεί το πρώτο προς τα αριστερά. Ωστόσο για να επιτύχουμε μια επαναλαμβανόμενη κίνηση πρέπει να δημιουργήσουμε το κατάλληλο event.

Πριν προχωρήσουμε στην δημιουργία του event, θα δημιουργήσουμε μια μεταβλητή global, με το όνομα speed και θα ορίσουμε την ταχύτητα των εικόνων σε 300 pixels ανά second, καθώς και δυο global μεταβλητές με τα ονόματα score και highscore που θα τις αρχικοποιήσουμε με 0. Για το event θα χρησιμοποιήσουμε σαν αντικείμενο το System και θα ορίσουμε σαν συνθήκη την Every Tick, που σημαίνει πάντα. Για την δράση του event θα χρησιμοποιήσουμε το αντικείμενο γρασίδι και η δράση θα είναι από την ενότητα Size and Position, η Set X και όπου ορίζουμε τις νέες συντεταγμένες του x σε pixels, με την έκφραση $Self.X-speed*dt$. Εφόσον χρειαζόμαστε ακριβώς την ίδια λειτουργία και για τους σωλήνες αλλά και για το νόμισμα, θα επαναλάβουμε την δράση μέσα στο ίδιο event.



Εικόνα 59- Event για την δημιουργία κίνησης στο γρασίδι, στους σωλήνες και στο νόμισμα

Με το παραπάνω event έχουμε πετύχει την κίνηση των στοιχείων αλλά μόνο μια φορά. Για αυτό θα δημιουργήσουμε ένα επιπλέον event ώστε η κίνηση να επαναλαμβάνεται.



Εικόνα 60- Event για την επαναλαμβανόμενη κίνηση του γρασιδιού

Ωστόσο η κίνηση των σωλήνων και του νομίσματος δεν θέλουμε να είναι συνέχεια επαναλαμβανόμενη, αλλά να πραγματοποιείται τυχαία και επιπλέον το νόμισμα να εμφανίζεται 2

φορές για να έχει μεγαλύτερο ενδιαφέρον το παιχνίδι. Για αυτόν τον σκοπό χρησιμοποιούμε το παρακάτω event με τις πολλαπλές δράσεις:

System	Every 2 seconds	System	Create object pipetop on layer 1 at (800, random(50,-250))
		System	Create object pipebottom on layer 1 at (800, pipetop.Y+700)
		System	Create object upload_4b1239e8e4c1247e81dd052f032e on layer 1 at (1100, random(200,300))
		System	Create object upload_4b1239e8e4c1247e81dd052f032e on layer 1 at (1200, random(200,300))
		Add action Add...	

Εικόνα 61- Το event με το οποίο κάθε δυο δευτερόλεπτα το σύστημα δημιουργεί με τυχαία σειρά τους σωλήνες και τα νομίσματα

4.2.3. Σκοπός του παιχνιδιού

Όπως είναι γνωστό σκοπός του παιχνιδιού είναι να βρίσκεται συνεχώς στον αέρα το στοιχείο littlebird και να περνάει ανάμεσα από τους σωλήνες χωρίς να τους ακουμπάει ή να ακουμπάει το έδαφος, διότι θα χάσει. Επιπλέον έχουμε προσθέσει στο παιχνίδι τα νομίσματα που είναι αυτά που δίνουν τη βαθμολογία του παίκτη. Αυτό μας οδηγεί στην δημιουργία events τα οποία θα ενεργοποιούνται με την σύγκρουση του στοιχείου littlebird με τα στοιχεία γρασίδι, σωλήνες και νομίσματα.

littlebird...	On collision with upload_4b1239e8e4c1247e81dd052f032e	<ul style="list-style-type: none"> upload... Destroy System Add 1 to score TextS... Set text to "Score: "&score 	Add...
littlebird...	On collision with pipebottom	<ul style="list-style-type: none"> System Set speed to 0 Restart Set Visible Highs... Set Visible Gam... Set Visible pipeb... Stop animation 	Add...
littlebird...	On collision with grass_PNG	<ul style="list-style-type: none"> System Set speed to 0 Restart Set Visible Highs... Set Visible Gam... Set Visible pipeb... Set Invisible 	Add...
littlebird...	On collision with pipetop	<ul style="list-style-type: none"> System Set speed to 0 Restart Set Visible Highs... Set Visible Gam... Set Visible pipet... Set Invisible 	Add...

Εικόνα 62- Τα events των συγκρούσεων

Σύγκρουση με νόμισμα:

Όταν το στοιχείο littlebird συγκρουστεί με κάποιο νόμισμα τότε το στοιχείο αυτό θα καταστραφεί και θα ενημερωθεί το στοιχείο του score προσθέτοντας συν ένα πόντο. Για την

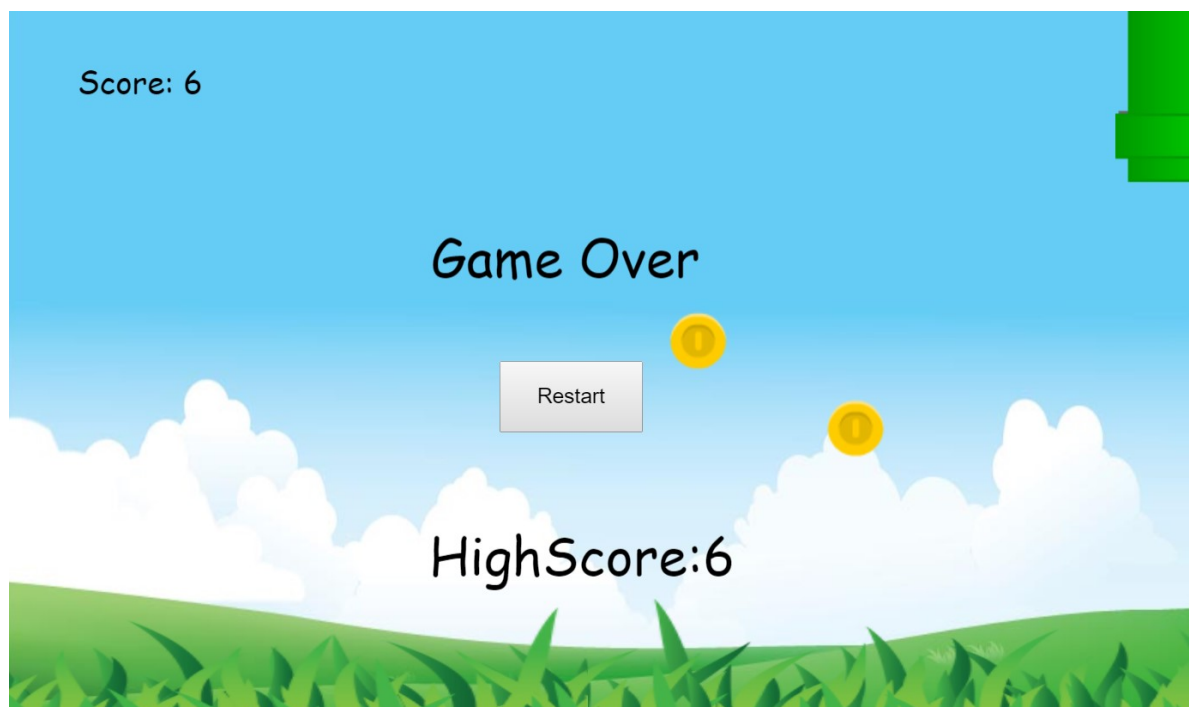
δημιουργία αυτής δράσης χρησιμοποιήθηκε η global μεταβλητή score. Στη συνέχεια θα ενημερωθεί και το text score που βρίσκεται στην οθόνη του παιχνιδιού.

Σύγκρουση με σωλήνα ή με το έδαφος:

Όταν το στοιχείο littlebird συγκρουστεί με ένα από τα παραπάνω στοιχεία τότε η ταχύτητα του littlebird θα μηδενιστεί και θα εμφανιστούν τα στοιχεία Restart, HighScore και GameOver. Το στοιχείο highscore εμφανίζεται μόνο αν ο παίκτης έχει πετύχει καλύτερη βαθμολογία. Το στοιχείο Restart επανεκκινεί το παιχνίδι.

System	score > highscore	System	Set highscore to score
		Highs...	Set text to "HighScore:"&highscore
		Add action	
Restart	On clicked	System	Set score to 0
		System	Set speed to 300
		System	Restart layout
		Add action	

Εικόνα 63- Βαθμολογία και επανεκκίνηση παιχνιδιού



Εικόνα 64- Στιγμιότυπο του παιχνιδιού

Το τελικό παιχνίδι:

<https://www.scirra.com/arcade/other-games/construct3game-30659>

4.3 Ανάπτυξη Παιχνιδιού στην Wimi5

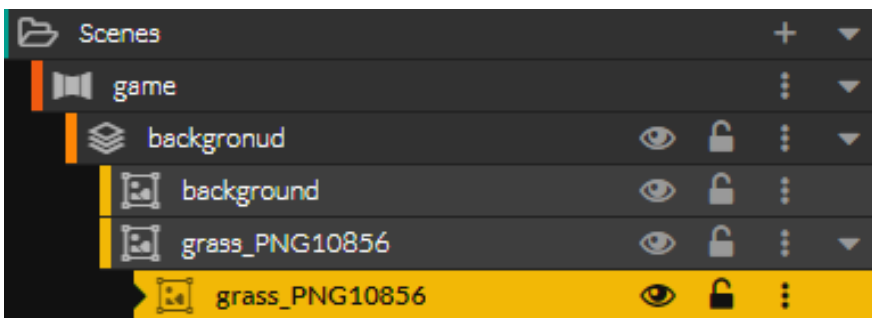
Η ανάπτυξη ενός παιχνιδιού στην μηχανή Wimi5 ξεκινάει με την είσοδο του χρήστη στο σύστημα. Με την ολοκλήρωση της διαδικασίας login, ο χρήστης μεταφέρεται στην οθόνη Dashboard όπου εκεί εμφανίζονται όλα τα projects που έχει υλοποιήσει και υπάρχει η επιλογή της δημιουργίας νέου project. Οι πρώτες επιλογές που έχει να κάνει ο χρήστης μετά την δημιουργία

ενός νέου project, είναι αυτή του ονόματος και των διαστάσεων του παιχνιδιού. Για το συγκεκριμένο project επιλέξαμε τις διαστάσεις 640x960 ώστε η εμφάνιση του παιχνιδιού μας να συμβαδίζει με την συνήθη μορφή του παιχνιδιού Flappy Bird.

4.3.1. Scene Editor

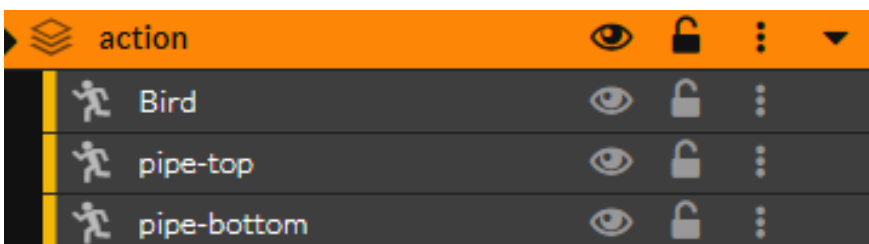
Η επόμενη οθόνη που εμφανίζεται είναι αυτή του Scene Editor όπου θα σχεδιάσουμε το παιχνίδι. Πριν προχωρήσουμε στην διαδικασία του σχεδιασμού όμως, ανεβάζουμε πρώτα τις εικόνες που θα χρησιμοποιήσουμε, μέσω του πάνελ Images. Στη συνέχεια τοποθετούμε την εικόνα που θα χρησιμοποιήσουμε ως background και την εικόνα του γρασιδιού δυο φορές. Ο λόγος που το κάνουμε αυτό είναι για να πετύχουμε και πάλι την αίσθηση της κίνησης, καθώς το γρασίδι θα μετατοπίζεται προς το αριστερό μέρος της οθόνης. Για την καλύτερη διαχείριση των αντικειμένων του παιχνιδιού, θα δημιουργήσουμε μέσα στην σκηνή που έχουμε μετονομάσει σε game, τέσσερα layers:

- i. Background: Μέσα σε αυτό το επίπεδο θα τοποθετήσουμε τις εικόνες του background και του γρασιδιού.



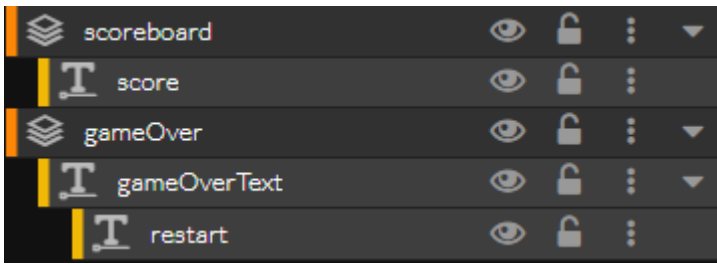
Εικόνα 65- Layer Background

- ii. Action: Σε αυτό το επίπεδο τοποθετούμε τις εικόνες που ορίζουμε ως sprite, δηλαδή, τις εικόνες Bird, pipe-top και pipe-bottom



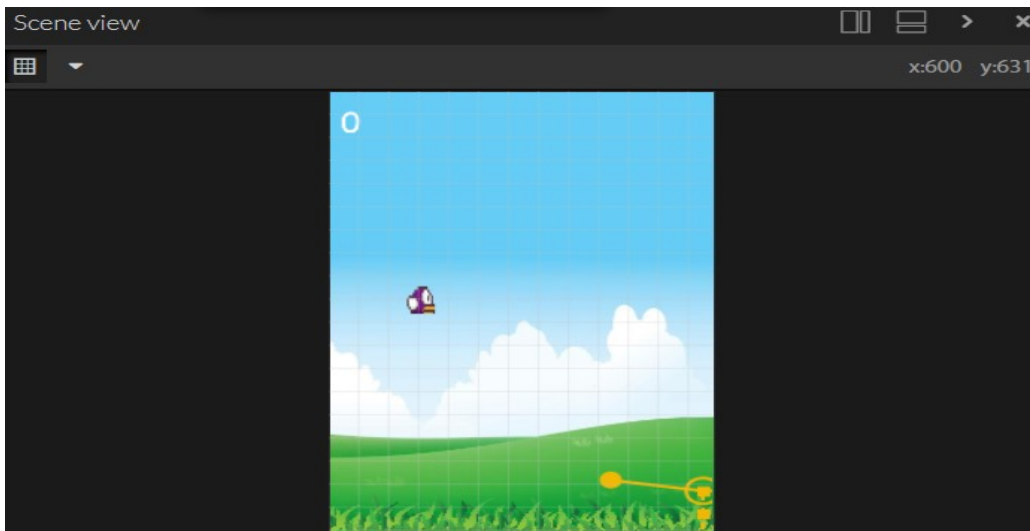
Εικόνα 66- Layer Action

- iii. Scoreboard: Στο επίπεδο αυτό τοποθετούμε το αντικείμενο, τύπου Text, όπου θα αναγράφεται η βαθμολογία του παίκτη
- iv. GameOver: Στο επίπεδο αυτό τοποθετούμε τα αντικείμενα τύπου Text όπου θα εμφανίζονται με το τέλος του παιχνιδιού. Το αντικείμενο restart είναι “παιδί” του αντικειμένου gameOverText, έτσι ώστε να εμφανίζεται αμέσως μετά από αυτό.



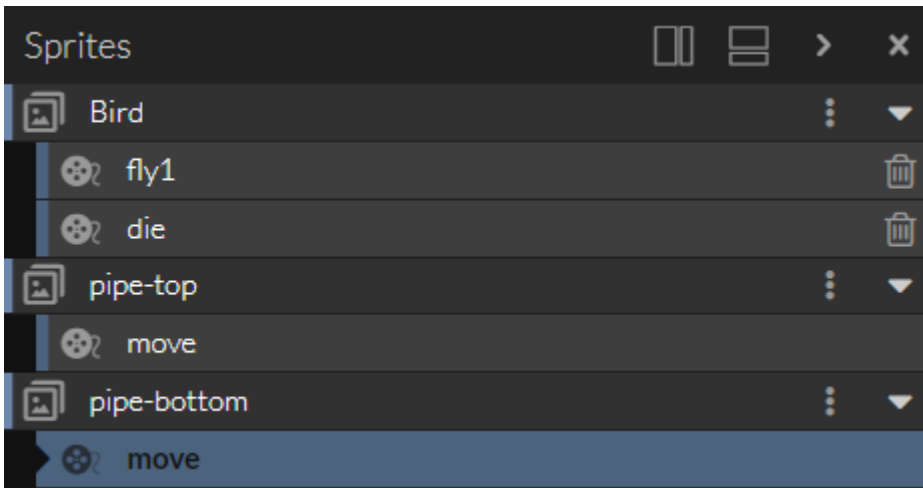
Εικόνα 67- Layers Scoreboard & gameOver

Για κάθε αντικείμενο που τοποθετούμε ορίζουμε και τις κατάλληλες ρυθμίσεις στο πάνελ Properties που αφορούν τη θέση του μέσα στην σκηνή του παιχνιδιού, το μέγεθός του και το αν θα είναι ορατό καθ' όλη τη διάρκεια του παιχνιδιού ή θα εμφανίζεται σε συγκεκριμένα σημεία.



Εικόνα 68- Στιγμιότυπο του Scene View

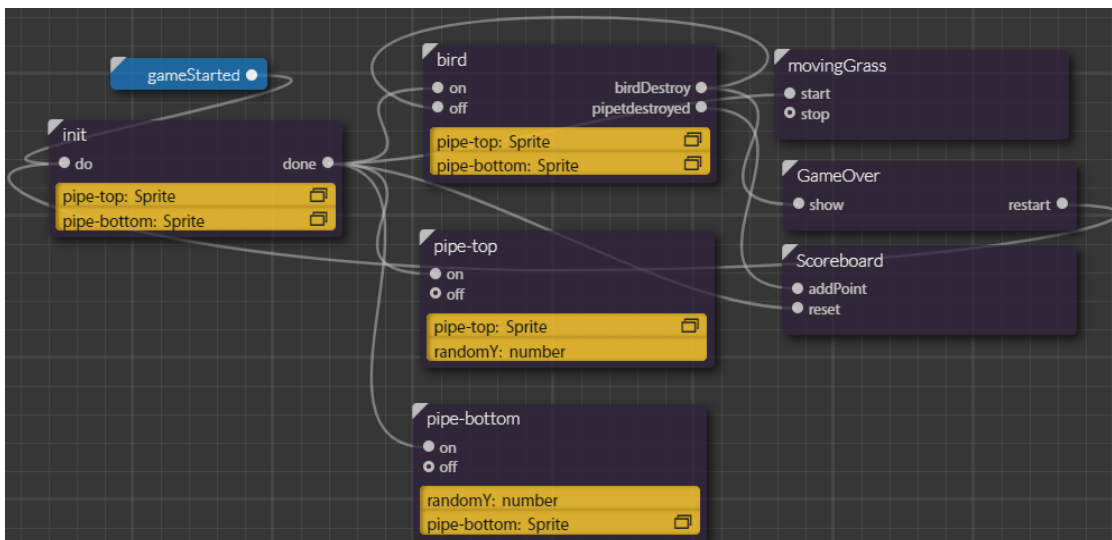
- v. Sprites: Για τις ανάγκες του παιχνιδιού δημιουργήθηκαν τρία sprites, το bird, το pipe-top και το pipe-bottom. Σε κάθε sprite, δημιουργήθηκαν ξεχωριστά τα απαιτούμενα animations. Στο sprite Bird χρειαζόμασταν 2 animations. Το πρώτο ήταν αυτό του πετάγματος και το δεύτερο αυτό της σύγκρουσης. Τα sprites pipe-top και pipe-bottom, χρειάστηκαν από ένα animation, αυτό της κίνησης. Όπως και στην τοποθέτηση των αντικειμένων στην σκηνή, έτσι και εδώ σε κάθε sprite και animation, μέσω του πάνελ properties ρυθμίστηκαν τα κατάλληλα πεδία που αφορούσαν τα frames, την ταχύτητα, την επανάληψη των κινήσεων και το μέγεθος.



Εικόνα 69- Τα sprites του παιχνιδιού

4.3.2. Logic Editor

Στην ενότητα του Logic Editor θα αναπτύξουμε τα σενάρια που θα ορίσουν την ροή του παιχνιδιού μας. Υπάρχουν δύο επιλογές εμφάνισης των σεναρίων, είτε σε διάγραμμα (Logic Chart), είτε σε δενδροδιάγραμμα (logic Tree). Στο πρώτο επίπεδο των σεναρίων υπάρχουν επτά σενάρια, που ενώνονται μεταξύ τους και όλα μαζί ενώνονται με τον διακόπτη έναρξης του παιχνιδιού.

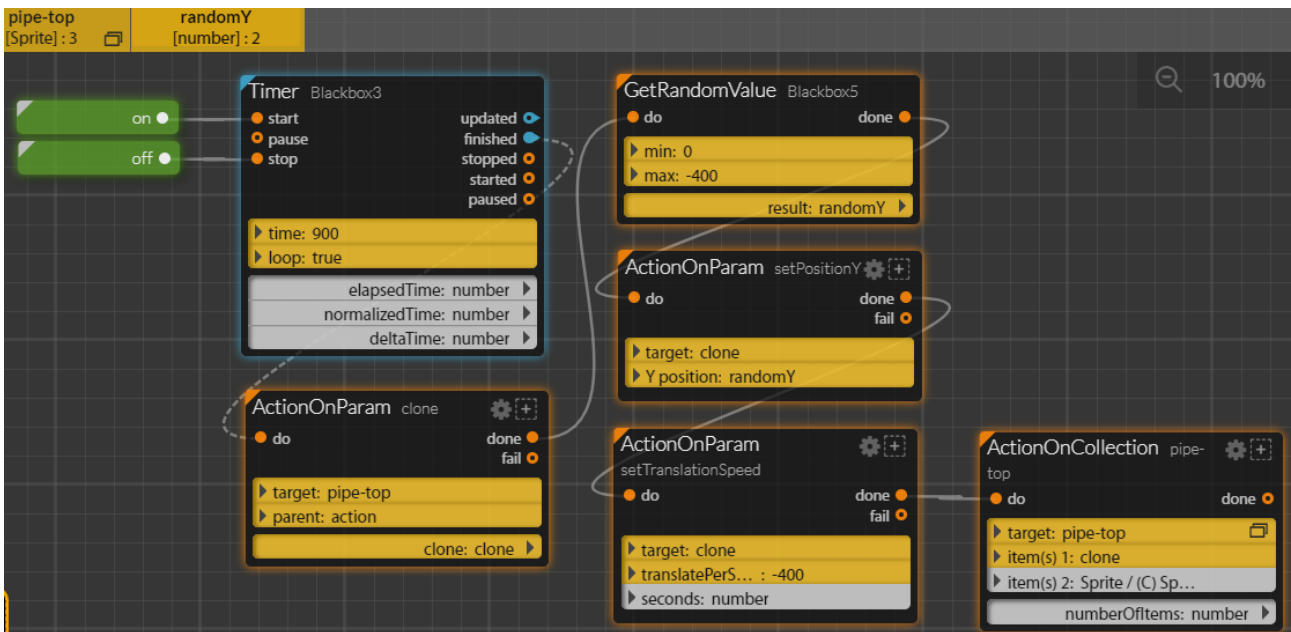


Εικόνα 70- Πρώτο επίπεδο σεναρίων

Τα σενάρια bird, pipe-top και pipe-bottom αφορούν την κίνηση των sprites, το σενάριο movingGrass αφορά την κίνηση του γρασιδιού, ενώ τα σενάρια GameOver και Scoreboard αφορούν την εμφάνιση των texts τερματισμού και την εμφάνιση της βαθμολογίας αντίστοιχα. Τέλος το σενάριο init, αν και ήταν το τελευταίο σενάριο που αναπτύχθηκε, τοποθετήθηκε πρώτο, καθώς αφορά την αρχικοποίηση του παιχνιδιού.

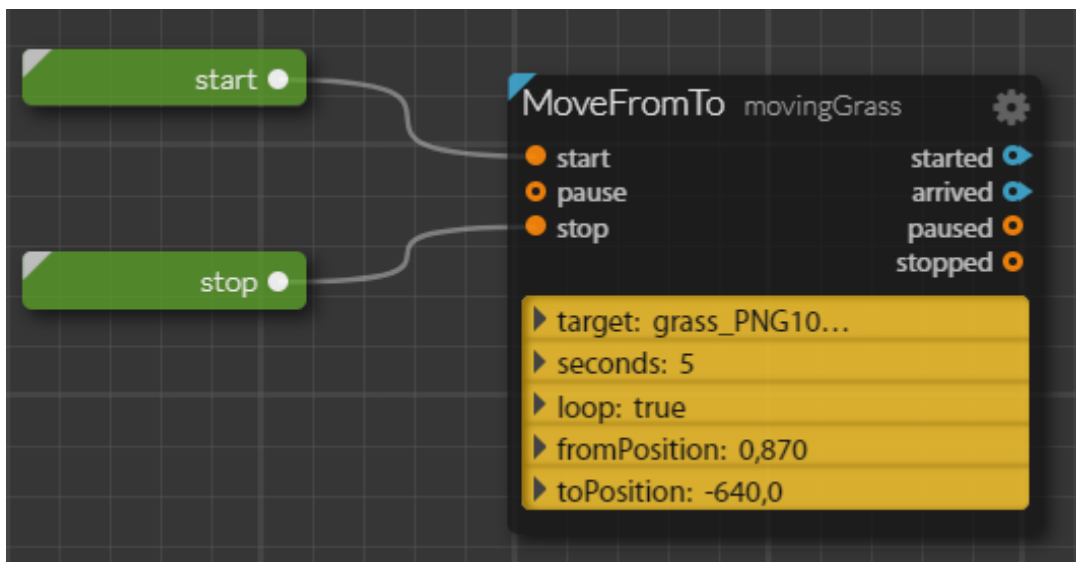
- i. Pipe-top & Pipe-bottom: Τα δυο αυτά σενάρια έχουν ακριβώς την ίδια δομή, αφού σκοπός τους είναι η επαναλαμβανόμενη κίνηση και εμφάνιση των σωλήνων στο παιχνίδι. Με την δημιουργία του κάθε σεναρίου, αρχικά τοποθετούμε τους δυο διακόπτες on & off. Το πρώτο

blackbox που τοποθετούμε είναι το Timer με το οποίο θα ρυθμίσουμε την συχνότητα με την οποία θα εμφανίζονται οι σωλήνες. Οι παράμετροι που ορίζουμε, είναι ο χρόνος στα 900 milliseconds και ο βρόγχος να επαναλαμβάνεται επ' άοριστον. Στη συνέχεια πρέπει να ορίσουμε το αντικείμενο το οποίο θα συνδέσουμε στον Timer. Για αυτόν τον σκοπό χρησιμοποιούμε το Blackbox ActionOnParam. Εκτός από το αντικείμενο που ορίζουμε, συγκεκριμένα το pipe-top, είναι σημαντικό να ορίσουμε και από ποιο layer προέρχεται αυτό το αντικείμενο, καθώς σε άλλη περίπτωση είναι δυνατόν το ίδιο αντικείμενο να βρίσκεται σε πάνω από ένα layer. Με το Blackbox GetRandomValue, πετυχαίνουμε το αντικείμενο μας να εμφανίζεται σε τυχαία θέση στην σκηνή με συγκεκριμένα όρια από το 0px έως το -400px. Για να ορίσουμε ότι τα όρια αυτά αφορούν τον άξονα Y, χρησιμοποιούμε ακόμα ένα blackbox ActionOnParam, ορίζοντας την δράση του σε setPositionY. Το επόμενο βήμα είναι να ορίσουμε την ταχύτητα με την οποία θα εμφανίζονται οι σωλήνες και την οποία ορίζουμε σε -400px ανά δευτερόλεπτο. Το τελευταίο blackbox που χρησιμοποιούμε στο σενάριο αυτό, είναι το ActionOnCollection. Η λειτουργία του είναι παρόμοια με το ActionOnParam, με την διαφορά όμως ότι χρησιμοποιείται για συλλογές αντικειμένων. Οι συλλογές αυτές μας βοηθούν να οργανώσουμε τα στοιχεία που πρέπει να επεξεργαστούμε από κοινού, όπως είναι το σύνολο των σωλήνων. Για να δημιουργήσουμε μια συλλογή αντικειμένων ανατρέχουμε στην καρτέλα Parameters και στο μενού 2D και από εκεί επιλέγουμε και τοποθετούμε μέσα στον editor την επιλογή sprite. Η δράση με την οποία δημιουργήθηκε το blackbox είναι η addItem. Δεν το αλλάζουμε, ωστόσο το μετονομάζουμε σε pipe-top και ορίζουμε ότι το sprite για στο οποίο αναφέρεται η συλλογή είναι το pipe-top. Τέλος μεταφέρουμε την συλλογή στην κοινή μπάρα, ώστε να είναι διαθέσιμη και για τα άλλα σενάρια στα οποία θα την χρησιμοποιήσουμε. Κάθε φορά λοιπόν που ένας σωλήνας θα δημιουργείται, θα τοποθετείται σε αυτή την συλλογή και θα φέρνει τις ίδιες λειτουργίες. Ακριβώς η ίδια λογική χρησιμοποιήθηκε και στο σενάριο pipe-bottom, με την διαφορά ότι τα όρια εμφάνισης στον άξονα Y, ορίστηκαν σε min=700px και max=800px.



Εικόνα 71- Το σενάριο pipe-top

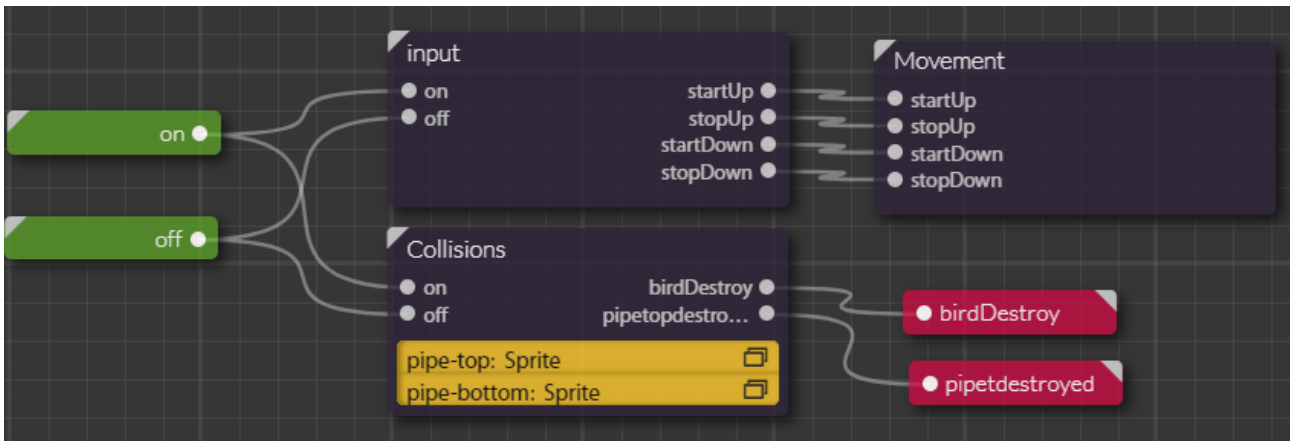
- ii. movingGrass: Στο σενάριο που αφορά την μετακίνηση των γρασιδιών χρησιμοποιήθηκε απλά το blackbox με το όνομα MoveFromTo. Οι παράμετροι που ορίσαμε, ήταν αυτοί του χρόνου, των συντεταγμένων και του βρόγχου. Η διαδικασία αυτή έγινε μόνο για το πρώτο αντικείμενο grass και αυτό διότι το δεύτερο αντικείμενο grass είναι παιδί του πρώτου και το ακολουθεί.



Εικόνα 72- Το σενάριο movingGrass

- iii. bird: Το σενάριο bird περιέχει τρία υποσενάρια και δυο διακόπτες εξόδου. Τα σενάρια input και movement αφορούν την κίνηση και την ταχύτητα του αντικειμένου bird. Για αυτόν τον λόγο οι εξοδοι του σεναρίου input είναι οι εισοδοι του σεναρίου movement. Για αυτό το

παιχνίδι, επιλέξαμε η κίνηση του bird να μην γίνεται με το ποντίκι, αλλά με τα πλήκτρα των βελών πάνω και κάτω.

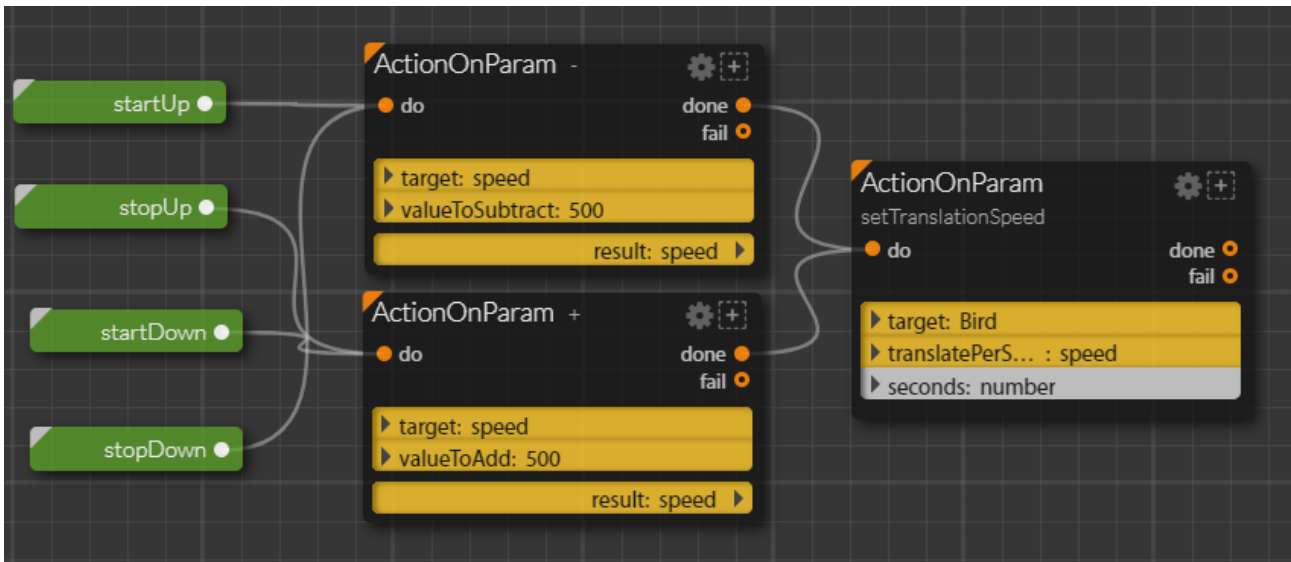


Εικόνα 73- Το σενάριο bird

Στο σενάριο input χρησιμοποιούνται δύο ίδια blackbox με την ονομασία KeyListener. Για το κάθε ένα επιλέγουμε το πλήκτρο που επιθυμούμε και στη συνέχεια τα συνδέουμε την κίνηση που είναι πατημένο το κάθε πλήκτρο και την κίνηση που δεν είναι πατημένο με την αντίστοιχη έξοδο. Στο σενάριο movement, με τη χρήση τριών blackboxes ActionOnParam καθορίζουμε την ταχύτητα ανά pixel με την οποία θα κινείται το bird στον άξονα Y.



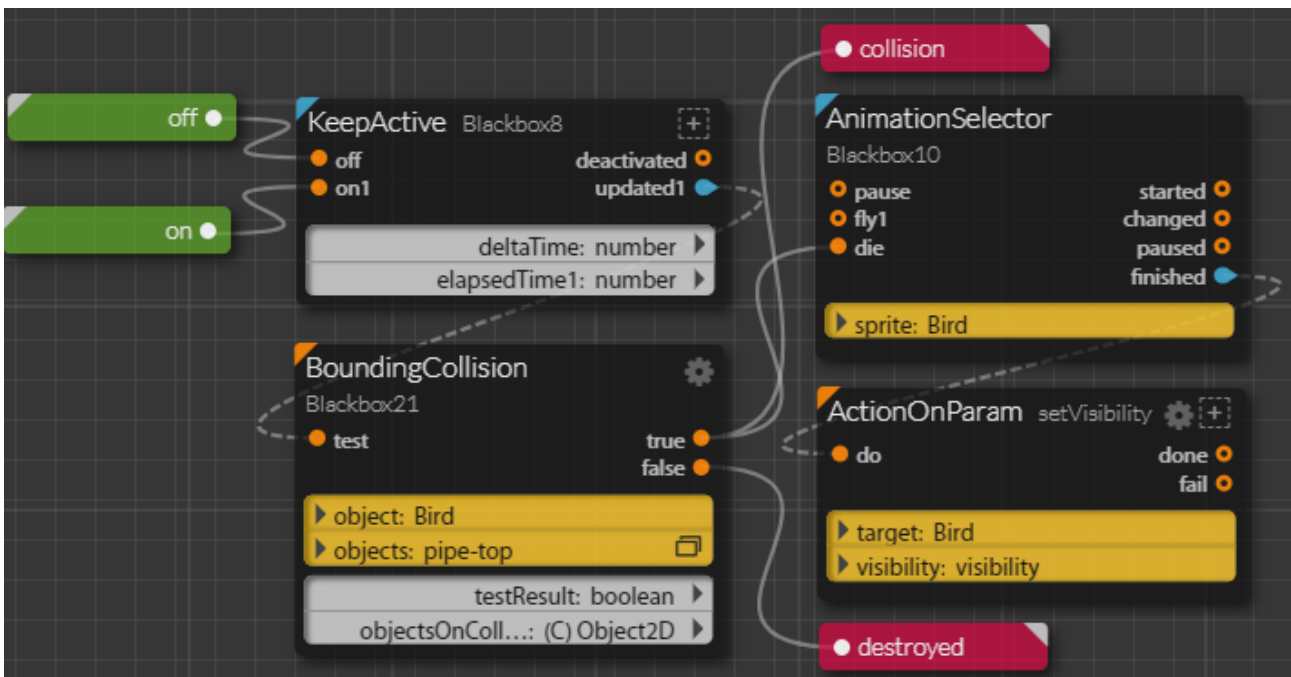
Εικόνα 74- Το σενάριο input



Εικόνα 75- Το σενάριο movement

Το τρίτο υποσενάριο με το όνομα Collisions αφορά τις συγκρούσεις ανάμεσα στο bird και στους σωλήνες. Μέσα σε αυτό το σενάριο υπάρχουν άλλα τρία σενάρια με τα ονόματα bird-riper-top, bird-riper-bottom & bird-bounds. Στο τελευταίο σενάριο ορίσαμε τα όρια μέσα στα οποία μπορεί να κινηθεί το bird μέσα στην σκηνή του παιχνιδιού, ώστε να μην χάνεται μέσα από το πεδίο δράσης. Τα άλλα δύο σενάρια που αφορούν το αντικείμενο bird και τους σωλήνες, όπως και στα σενάρια που αφορούν την κίνηση των σωλήνων, είναι πανομοιότυπα με μια μόνο μικρή διαφορά στο bird-riper-top που έχει σχέση με τη βαθμολογία.

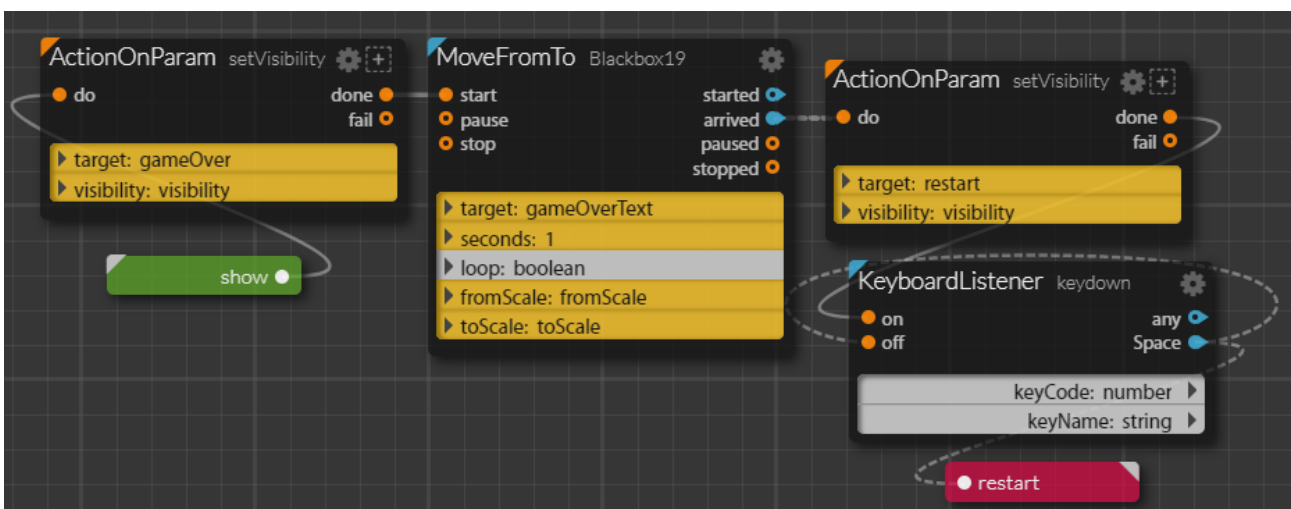
Για την υλοποίηση των συγκρούσεων μεταξύ του αντικειμένου bird και των σωλήνων χρησιμοποιήσαμε αρχικά το blackbox KeepActive ώστε η διαδικασία να είναι ενεργοποιημένη σε κάθε στιγμιότυπο του παιχνιδιού. Εν συνεχεία το συνδέσαμε με το blackbox BoundingCollision, το οποίο είναι υπεύθυνο για την ανίχνευση της σύγκρουσης μεταξύ των αντικειμένων. Εάν δεν υπάρχει σύγκρουση, η ροή του παιχνιδιού οδηγείται στην έξοδο destroyed η οποία συμβάλλει στην βαθμολογία του παιχνιδιού και συνδέεται με την έξοδο riperdestroyed, που με τη σειρά της συνδέεται στην ίδια έξοδο που υπάρχει στο επίπεδο του σεναρίου bird και τέλος συνδέεται με το σενάριο Scoreboard. Εάν υπάρχει όμως σύγκρουση, η ροή του παιχνιδιού οδηγείται στην έξοδο collision και στα blackboxes AnimationSelector & ActionOnParam. Στο πρώτο επιλέγουμε με την σύγκρουση να αλλάξει το animation του αντικειμένου bird, από fly σε die και με το δεύτερο το κάνουμε αόρατο.



Εικόνα 76- Το σενάριο bird-pipe-top

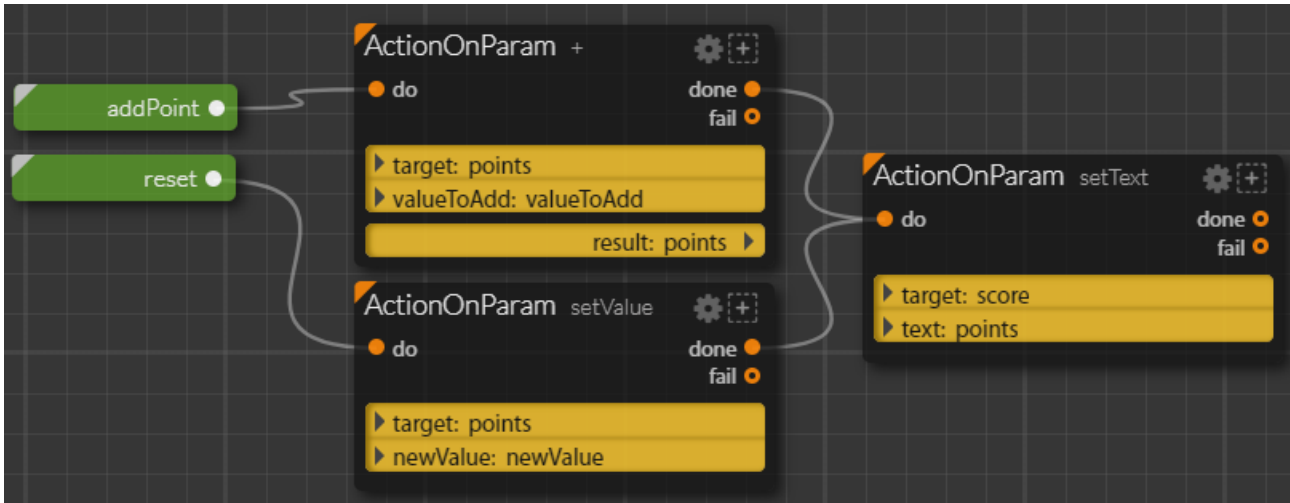
Στο σενάριο bird-pipe-bottom δεν υπάρχει η έξοδος destroyed. Ελέγχουμε μόνο αν δεν υπάρχει σύγκρουση με τον άνω σωλήνα για να αυξηθεί η βαθμολογία, καθώς οι δυο σωλήνες στην σκηνή του παιχνιδιού έχουν τοποθετηθεί ο ένας κάτω από τον άλλον. Η έξοδος collision των δύο σεναρίων bird-pipe-bottom και bird-pipe-top συνδέεται με την έξοδο birdDestroy, η οποία συνδέεται με το σενάριο GameOver.

- iv. GameOver: Το σενάριο αυτό ενεργοποιείται μόνο όταν υπάρχει κάποια σύγκρουση. Σκοπός του σεναρίου είναι να εμφανίζει τα δυο αντικείμενα text που ορίσαμε για την λήξη του παιχνιδιού και να δίνεται η επιλογή για την επανεκκίνησή του. Εφόσον όπως έχει αναφερθεί το δεύτερο text είναι παιδί του πρώτου, η διαδικασία θα γίνει μια φορά.



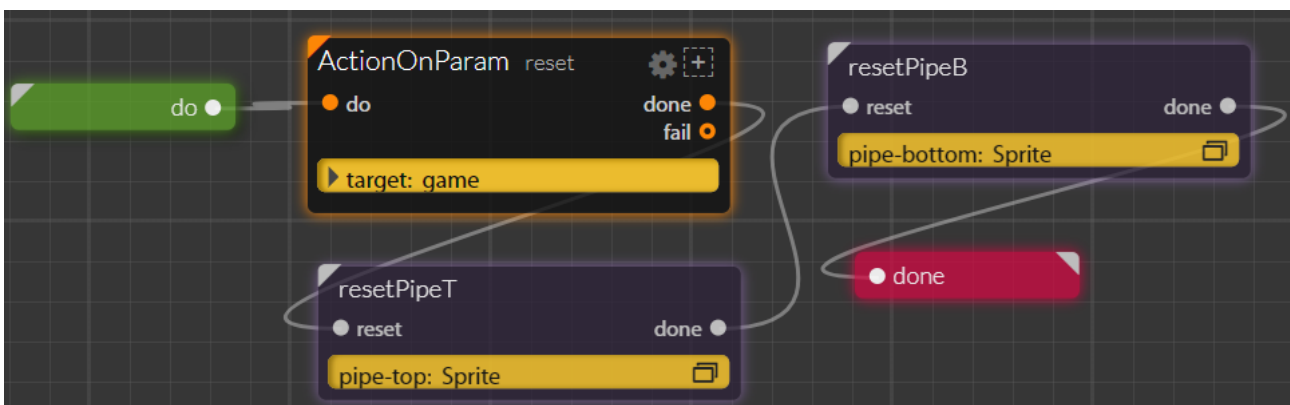
Εικόνα 77- Το σενάριο GameOver

- v. Scoreboard: Το σενάριο αυτό υλοποιείται με την χρήση τριών ActionOnParam blackboxes και διακοπών έναρξης με τα ονόματα addPoint και reset. Το πρώτο ActionOnParam είναι συνδεδεμένο με τον διακόπτη addPoint προσθέτει έναν πόντο κάθε φορά που καλείται, ενώ το δεύτερο είναι συνδεδεμένο με το reset και θέτει την βαθμολογία ξανά σε 0. Είτε κληθεί το πρώτο είτε το δεύτερο blackbox, η ροή του παιχνιδιού θα καταλήξει στο ActionOnParam με δράση setText, όπου θα ορίσει το κείμενο της βαθμολογίας με το ανάλογο text.



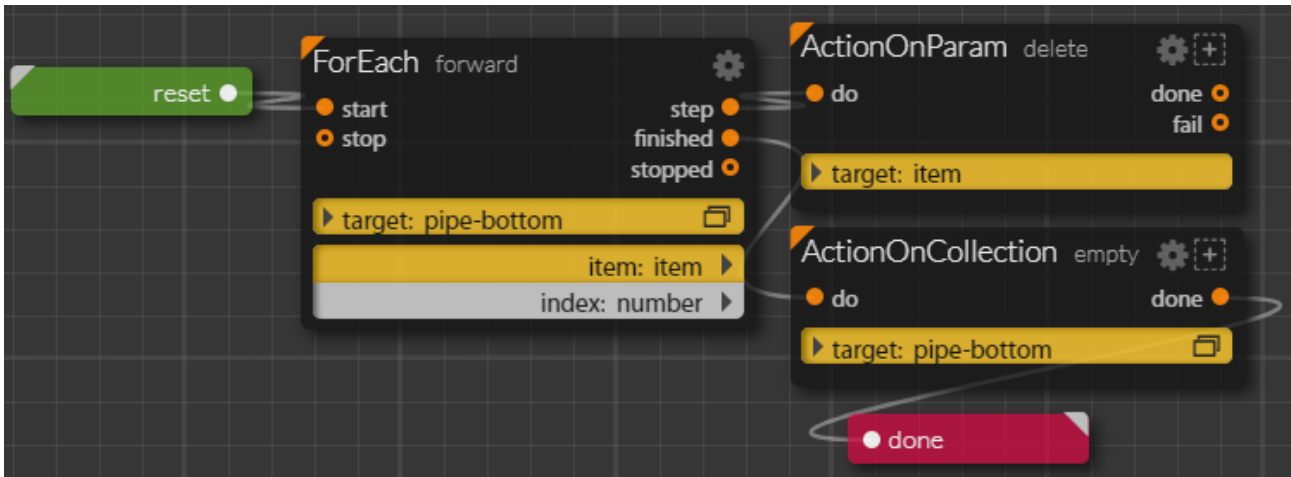
Εικόνα 78- Το σενάριο Scoreboard

- vi. init: Το σενάριο init όπως έχει ήδη αναφερθεί είναι αυτό που με αρχικοποιεί το παιχνίδι μετά την λήξη του. Για να ολοκληρωθεί η διαδικασία αυτή χρησιμοποιείται ένα ActionOnParam με επιλεγμένη δράση reset, δυο σενάρια που αφορούν τους σωλήνες και μια έξοδο.



Εικόνα 79- Το σενάριο init

Το σενάριο που αφορά την αρχικοποίηση του κάτω σωλήνα, χρησιμοποιεί ένα blackbox ForEach ώστε να υπάρχει πρόσβαση σε κάθε αντικείμενο της συλλογής, ένα ActionOnParam το οποίο διαγράφει το κάθε αντικείμενο σωλήνα που εμφανίζεται στην οθόνη και ένα ActionOnCollection το οποίο αδειάζει τη συλλογή αντικειμένων των σωλήνων. Το ίδιο σενάριο υπάρχει και για τον σωλήνα Pipe-top.



Εικόνα 80- Το υποσενάριο του init

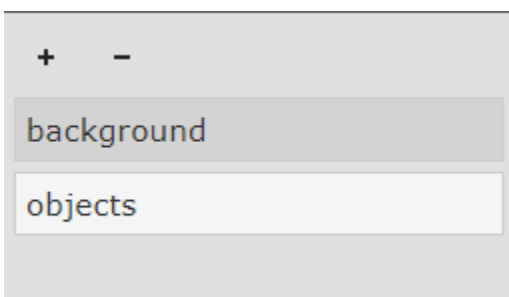
Η ορθή λειτουργία όλων των σεναρίων ολοκληρώνεται με την σωστή συνδεσμολογία, όπως αυτή φαίνεται στις εικόνες.

Το τελικό παιχνίδι:

https://games.wimi5.com/games/dokimastiko-195b1946aba61720e63f0084c1/0_0_2/

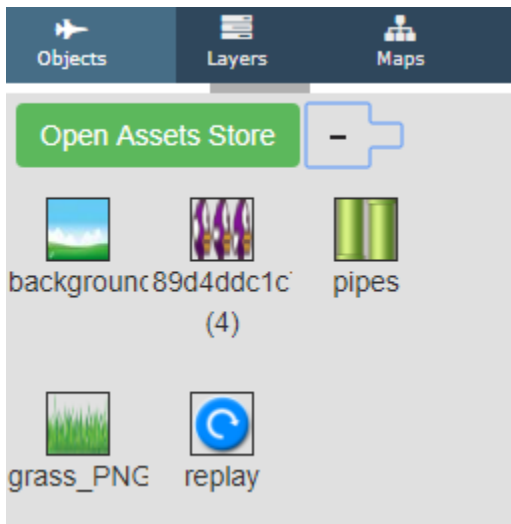
4.4 Ανάπτυξη Παιχνιδιού στην Ceilfire

Για την ανάπτυξη του παιχνιδιού στην μηχανή Ceilfire, είναι απαραίτητο να συνδεθούμε με τον λογαριασμό που έχουμε ήδη δημιουργήσει, όπως συνέβη και με τις προηγούμενες μηχανές. Η ανάπτυξη του παιχνιδιού θα ξεκινήσει με την δημιουργία δυο layers με τα ονόματα background και objects. Στο πρώτο layer θα έχουμε την εικόνα που θα χρησιμοποιήσουμε ως φόντο και στο δεύτερο θα έχουμε όλα τα αντικείμενα του παιχνιδιού όπως τον χαρακτήρα. Κοινό χαρακτηριστικό και των δυο επιπέδων είναι το αντικείμενο Game. Από οποιοδήποτε layer και αν δουλεύουμε το αντικείμενο Game, οι αλλαγές εμφανίζονται και στα δυο, καθώς πρόκειται για τον πυρήνα του παιχνιδιού.



Εικόνα 81- Τα layers του παιχνιδιού

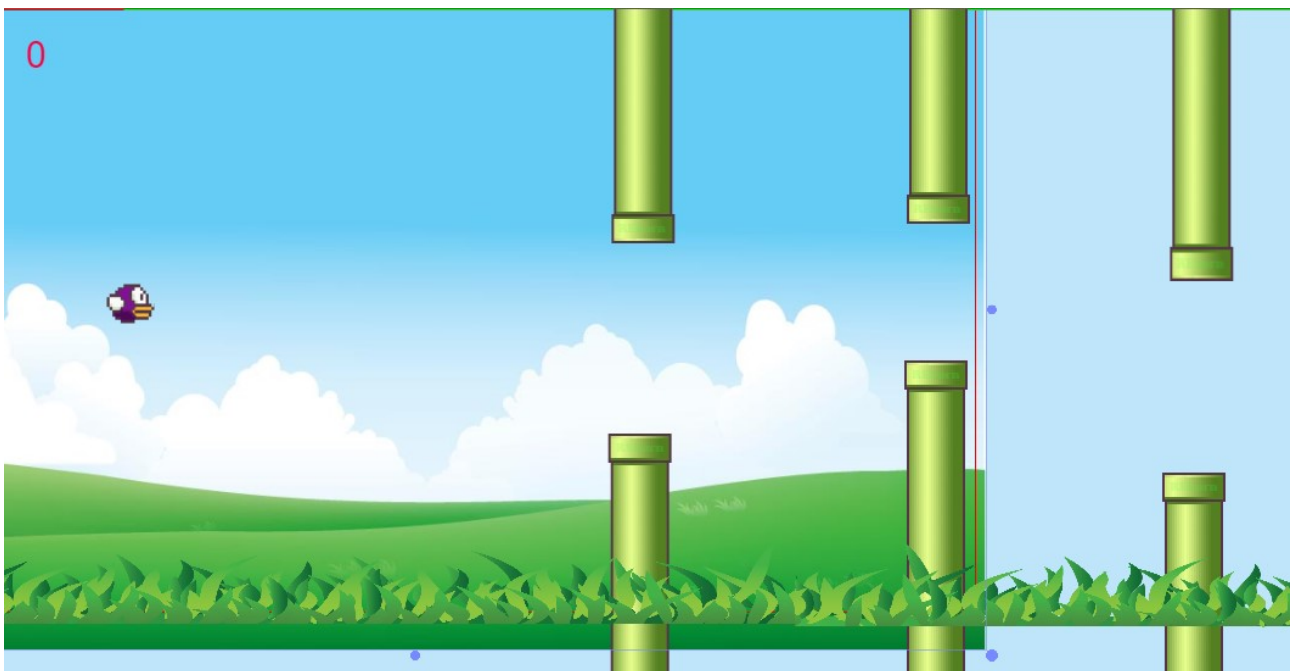
Η μεταφόρτωση των αντικειμένων γίνεται μέσω του Asset Browser (εικόνα 33). Το σύνολο των αντικειμένων που θα χρησιμοποιήσουμε παρουσιάζεται στην εικόνα :



Εικόνα 82- Τα αντικείμενα του παιχνιδιού

Τα αντικείμενα του χαρακτήρα, των σωλήνων και του γρασιδιού όταν τα επιλέξαμε μέσα από τον asset browser, τα ανεβάσαμε στο παιχνίδι ως Spritesheet. Η επιλογή αυτή δίνει στα αντικείμενα επιπλέον ιδιότητες που βοηθούν στην κίνηση του χαρακτήρα. Κατά την φόρτωση ενός αντικειμένου ως spritesheet ο χρήστης μπορεί να ρυθμίσει από πόσες στήλες και γραμμές θα αποτελείται το αντικείμενο του. Για τον χαρακτήρα επιλέξαμε μια γραμμή και τρεις στήλες, ώστε να απομονώνονται τα τρία στιγμιότυπα του χαρακτήρα, για τις σωλήνες επιλέξαμε μια γραμμή και δύο στήλες αντίστοιχα και στο γρασίδι επιλέξαμε μια γραμμή και μια στήλη.

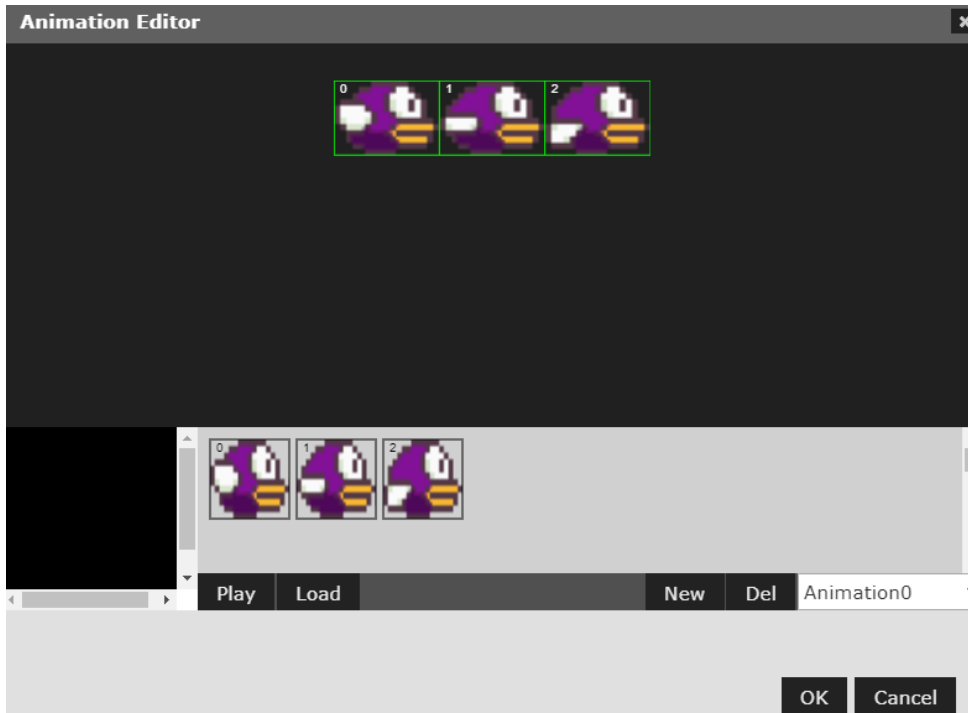
Με την επιλογή ενός αντικειμένου την φορά, θα στήσουμε την σκηνή του παιχνιδιού μέσα στο viewport:



Εικόνα 83- Η σκηνή του παιχνιδιού

4.4.1. Littlebird

Από την επιλογή του χαρακτήρα ως spritesheet, η εικόνα του αντικειμένου είναι χωρισμένη σε τρία στιγμιότυπα.



Εικόνα 84- Animation Editor Ceilfire

Αφού τοποθετήσουμε τον χαρακτήρα μέσα στην σκηνή, επιλέγουμε το κουμπί Animations, το οποίο βρίσκεται στην πάνω δεξιά γωνία του πεδίο Objects. Η επιλογή αυτή θα μας εμφανίσει ένα νέο παράθυρο, όπως φαίνεται στην εικόνα . Με το κουμπί New εμφανίζεται ένα νέο πεδίο στο οποίο μπορούμε να ορίσουμε την ταχύτητα του sprite και με το κουμπί Play, βλέπουμε την προεπισκόπηση της κίνησης του χαρακτήρα. Για να δώσουμε στον χαρακτήρα την κίνηση που επιθυμούμε, επιλέγουμε το αντικείμενο Game→ New Event Rule. Θα δημιουργήσουμε δυο κανόνες. Ο πρώτος κανόνας αφορά την έναρξη της κίνησης με την έναρξη του παιχνιδιού. Σαν συνθήκη θα επιλέξουμε Choose Condition→ Game now Started και σαν δράση New Action→ Επιλογή του χαρακτήρα→ Play Animation. Ο δεύτερος κανόνας αφορά την κίνηση του χαρακτήρα σε συνδυασμό με το πάτημα του ποντικιού. Η συνθήκη που θα επιλέξουμε αυτή τη φορά είναι Mouse-button is down, ενώ η δράση που θα επιλέξουμε και πάλι για τον χαρακτήρα, είναι η Set Velocity Y όπου και ρυθμίζουμε την ταχύτητα που θα έχει στον άξονα Y σε -400 pixels. Την ταχύτητα της βαρύτητας θα την ορίσουμε από τις ιδιότητες Physics και συγκεκριμένα στο πεδίο Gravity in y Direction, σε 800 pixels.

4.4.2. Σωλήνες

Η τοποθέτηση των εμποδίων θα πρέπει να γίνει για κάθε εμπόδιο ξεχωριστά. Για να αλλάξουμε την φορά των εμποδίων που θα βρίσκονται στο έδαφος, από επιλέγουμε από τις

ιδιότητες, Misc→ Current Frame→1. Στις ρυθμίσεις Physics, για όλα τα εμπόδια θα επιλέξουμε Velocity(-200,0) για να πετύχουμε την κίνηση των εμποδίων προς τα αριστερά. Για όλα τα αντικείμενα των εμποδίων θα δημιουργήσουμε από έναν κανόνα, με μια συνθήκη και τέσσερις δράσεις. Για να μειώσουμε τις διαδικασίες, θα ομαδοποιήσουμε τους σωλήνες ανά δύο. Για την κίνηση των σωλήνων pipe-t1 και pipe-b1, επιλέγουμε τον pipe-t1 και ορίζουμε τον εξής κανόνα:

New condition→ Is this object out of game?

New action→ pipe-t1→set the PosX of object to 800

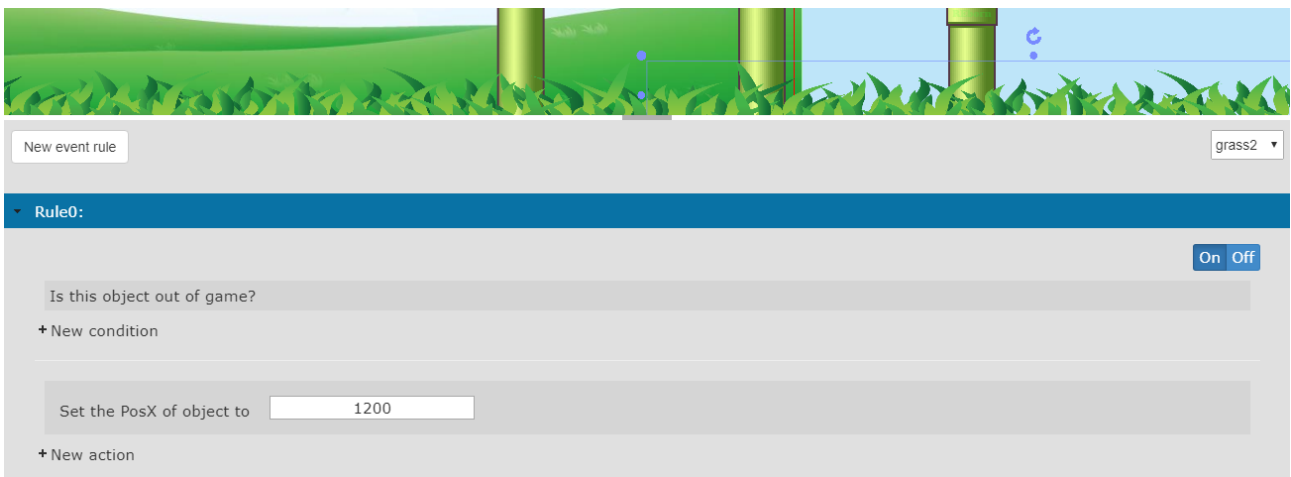
New action→ pipe-b1→set the PosX of object to 800

Με αυτόν τον τρόπο, κάθε φορά που ο σωλήνας pipe-t1 θα φεύγει από το αριστερό μέρος της σκηνής, θα εμφανίζεται εκ νέου στα δεξιά. Ο λόγος που δεν χρειάζεται να δημιουργήσουμε ξεχωριστό κανόνα μέσα στον pipe-b1 είναι διότι στην σκηνή τους έχουμε τοποθετήσει τον ένα κάτω από τον άλλον, έτσι ώστε να κινούνται μαζί. Οι ίδιοι ακριβώς κανόνες χρησιμοποιούνται και για τα ζεύγη pipe-t2/pipe-b2 και pipe-t3/pipe-b3.

Οι επόμενες δύο δράσεις που βρίσκονται και αυτές στην συνθήκη Is this object out of game αφορούν την βαθμολογία και θα αναλυθούν παρακάτω.

4.4.3 Γρασίδι

Με την κίνηση του γρασιδιού, εκτός από τον όμορφο σχεδιασμό της σκηνής προσφέρεται η αίσθηση της κίνησης και επιπλέον χρησιμοποιείται ως ένα ακόμα αντικείμενο με το οποίο μπορεί να συγκρουστεί ο χαρακτήρας. Χρησιμοποιώντας δύο φορές το αντικείμενο του γρασιδιού μέσα στην σκηνή με τις ίδιες διαστάσεις και τους ίδιους κανόνες πετυχαίνουμε μια ομαλή και συνεχόμενη κίνηση, καθώς όταν ολοκληρώνεται η κίνηση του grass1, ξεκινάει η κίνηση του grass2. Στις ιδιότητες Physics ρυθμίζουμε και πάλι το πεδίο Velocity ως (-200,0) ώστε να κινείται με την ίδια ταχύτητα και των υπολοίπων αντικειμένων. Ως κανόνα, χρησιμοποιούμε την ίδια λογική που χρησιμοποιήθηκε και με τους σωλήνες. Αυτό σημαίνει πως ως συνθήκη χρησιμοποιήθηκε η Is this Object out of game και ως δράση η Set the PosX of object to 1200.

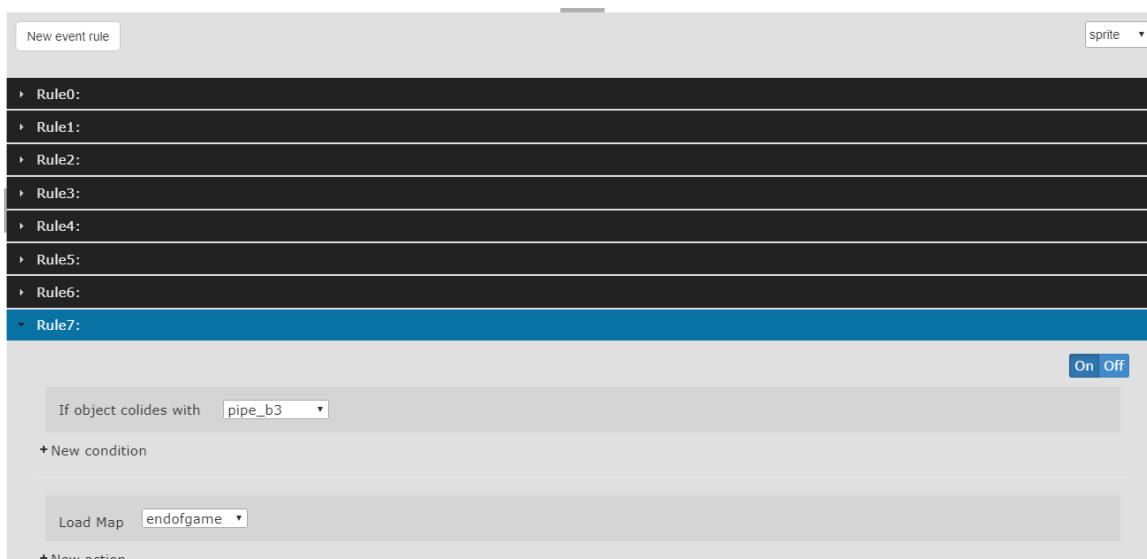


Εικόνα 85- Ο κανόνας τοποθέτησης του grass2 στην θέση x:1200

4.4.4. Συγκρούσεις

Οι συγκρούσεις σε αυτό το παιχνίδι καθορίζουν την συνέχεια του. Εάν ο χαρακτήρας συγκρουστεί με κάποιο αντικείμενο τότε το παιχνίδι σταματά και εμφανίζεται η επόμενη οθόνη που παρακινεί τον παίκτη να παίξει ξανά. Τα αντικείμενα με τα οποία ο χαρακτήρας μπορεί να συγκρουστεί είναι: 1) grass1, 2) grass2, 3)pipe-t1, 4)pipe-b1, 5)pipe-t2, 6)pipe-b2, 7)pipe-t3, 8)pipe-b3. Έτσι λοιπόν, θα δημιουργήσουμε 8 ξεχωριστούς κανόνες με τις ίδιες συνθήκες και δράσεις, ανάμεσα στον χαρακτήρα και στα αντικείμενα όπου θα διαμορφώνουν τη συνέχεια του παιχνιδιού μετά από μια σύγκρουση. Η διαδικασία όπου ακολουθείται είναι η εξής:

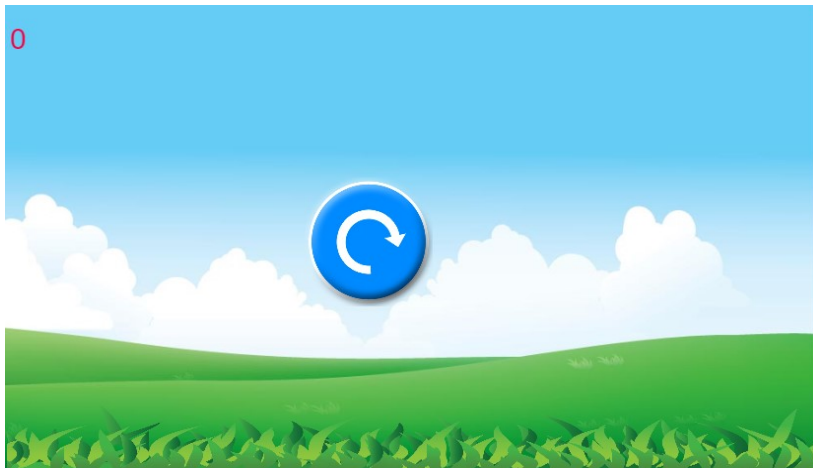
Επιλογή του χαρακτήρα → New Rule → New condition → Collides with → Επιλογή εμποδίου → Ok
 New action → Game → Load Map → Επιλογή χάρτη



Εικόνα 86- Οι κανόνες σύγκρουσης του χαρακτήρα

4.4.5. Χάρτης

Ο όρος χάρτης στην μηχανή Ceilfire περιγράφει τις οθόνες που μπορεί να έχει ένα παιχνίδι, για παράδειγμα μια οθόνη είναι το μενού. Στην περίπτωση αυτού το παιχνιδιού, είναι η οθόνη που θα εμφανίζεται στο τέλος του παιχνιδιού, μετά από μια σύγκρουση και την οποία ονομάσαμε endofgame.

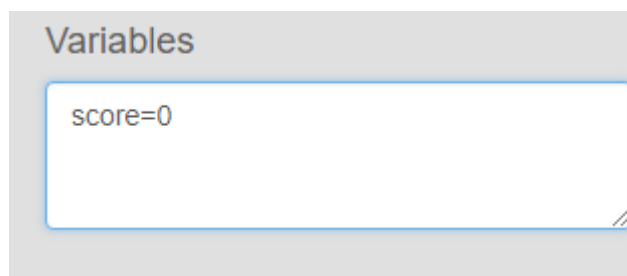


Εικόνα 87- Η οθόνη στο τέλος του παιχνιδιού.

Στην ιστοσελίδα που φιλοξενείται το παιχνίδι μετά την δημοσίευσή του, υπάρχει η επιλογή της επανάληψης, για αυτόν τον λόγο δεν ορίσαμε κάποια λειτουργία στον χάρτη endofgame.

4.4.6. Βαθμολογία

Η βαθμολογία του χρήστη αυξάνεται όταν ο χαρακτήρας περνάει ένα ζεύγος σωλήνων. Εφόσον ο χαρακτήρας περάσει χωρίς να συγκρουστεί με κάποιον από τους σωλήνες, οι σωλήνες συνεχίζουν να κινούνται προς τα αριστερά. Για αυτόν τον λόγο ορίσαμε η βαθμολογία του χρήστη να αυξάνεται όταν ένας σωλήνας από το ζεύγος εξαφανιστεί από την οθόνη. Όπως σε κάθε περίπτωση βαθμολογίας, πρέπει να ορίσουμε και να αρχικοποιήσουμε με 0 μια μεταβλητή. Αυτή η μεταβλητή είναι η score και την ορίσαμε στις γενικές ρυθμίσεις του αντικειμένου Game.



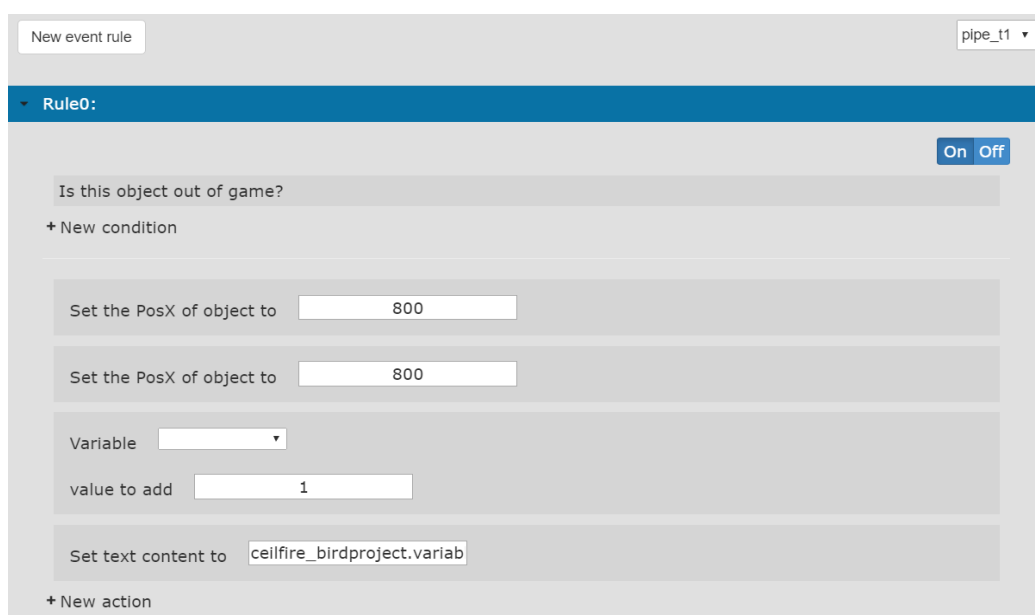
Εικόνα 88- Ορισμός μεταβλητής

Στη συνέχεια επιστρέφουμε στα αντικείμενα σωλήνες. Όπως αναφέρθηκε παραπάνω, ο κανόνας για την κίνηση των σωληνώσεων ανά ζεύγη, με την συνθήκη Is this object out of game περιέχει άλλες δύο δράσεις που αφορούν την βαθμολογία. Η μια δράση αφορά την αύξηση της

βαθμολογίας και η άλλη αφορά την εμφάνιση της στην οθόνη του χρήστη. Για την πρώτη δράση ακολουθούμε τα εξής βήματα: New action→ Game→ Increase Variable→ Επιλογή μεταβλητής και ρυθμού αύξησης. Για την εμφάνιση της βαθμολογίας δημιουργούμε ένα αντικείμενο text, μέσα στο layer objects πατώντας στο εικονίδιο για νέα δημιουργία στην αριστερή γωνία του πάνελ επιλέγουμε text και το ονομάζουμε score. Μετά από αυτήν την ενέργεια, επιστρέφουμε στην συνθήκη και προσθέτουμε και την τέταρτη δράση ως εξής:

New action→ score→ Set Text → ceilfire_birdproject.variables.score.

Όπως και για τις δύο πρώτες δράσεις που αναφέρθηκαν παραπάνω, έτσι και αυτές οι δράσεις επαναλαμβάνονται για όλα τα ζεύγη των σωληνώσεων.



Εικόνα 89- Οι δράσεις που ενεργοποιούνται όταν ένα ζεύγος σωληνώσεων εξαφανίζεται από την οθόνη του παιχνιδιού

Το τελικό παιχνίδι:

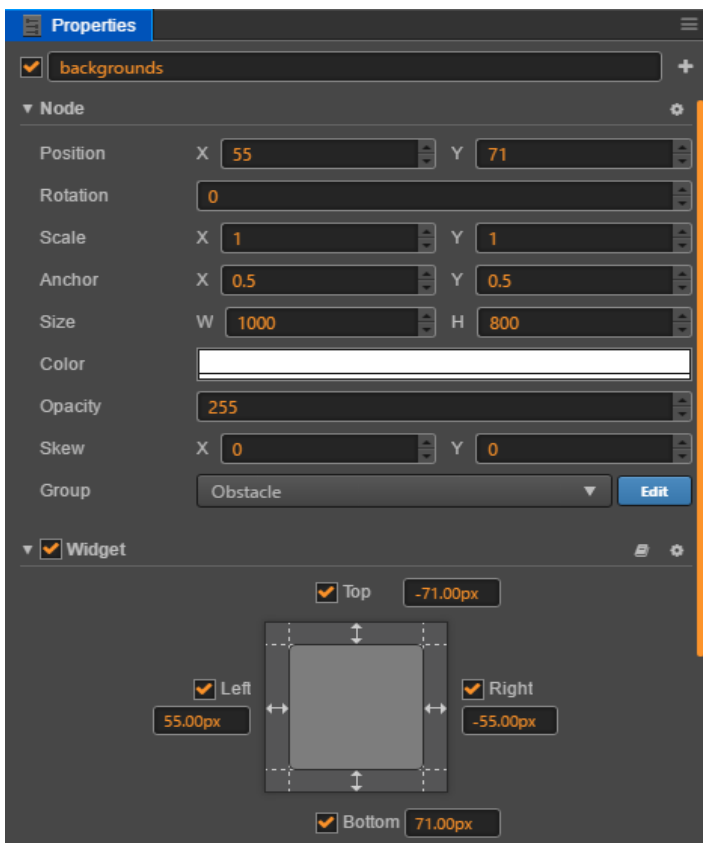
<https://ceilfire.com/p/5b167a128a3d9da54bf5bb1b>

4.5 Ανάπτυξη Παιχνιδιού στην Cocos Creator

Η δημιουργία ενός παιχνιδιού στην μηχανή Cocos Creator ξεκινά με την εγκατάσταση του λογισμικού της μηχανής. Ωστόσο για την πλήρη αξιοποίηση των δυνατοτήτων της μηχανής είναι απαραίτητη και η εγκατάσταση ενός Javascript editor. Στα tutorials της επίσημης ιστοσελίδας, προτείνεται το πρόγραμμα Visual Studio Code της Microsoft. Η ανάπτυξη του συγκεκριμένου παιχνιδιού βασίστηκε στα tutorials και στα παραδείγματα που βρίσκονται στην επίσημη ιστοσελίδα της μηχανής, ενώ τα σενάρια που χρησιμοποιούνται έχουν παρθεί από το παράδειγμα: <https://github.com/cocos-creator/tutorial-duang-sheep>. Με την έναρξη της μηχανής και την δημιουργία νέου project, όπως έχει ήδη αναφερθεί, στον φάκελο προέλευσης δημιουργούνται οι

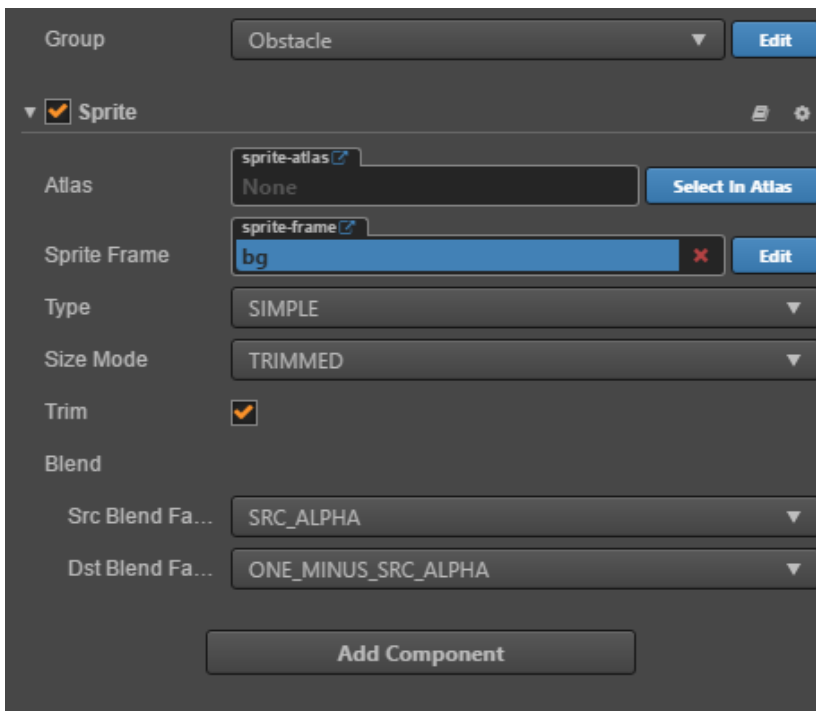
φάκελοι του παιχνιδιού. Στον φάκελο assets φορτώνουμε όλα τα αρχεία που θα χρειαστούμε, στους κατάλληλους υποφακέλους, για να υπάρχει σωστή οργάνωση στο παιχνίδι μας.

Η ανάπτυξη του παιχνιδιού, ξεκινάει με την δημιουργία της σκηνής, η οποία υλοποιείται από το πάνελ Assets→ Create→ Scene, την οποία μετονομάζουμε σε Game. Έπειτα προχωράμε στην σχεδίαση της σκηνής του παιχνιδιού. Στο πάνελ Node Tree, κάνοντας δεξί κλικ στο Canvas επιλέγουμε Create→ Create Empty Node και δημιουργούμε το στοιχείο backgrounds. Για την σταθερότητα του στοιχείου αυτού, στο πάνελ Properties επιλέγουμε Add Component→ Add UI Component→ Widget, ώστε να ρυθμίσουμε το μέγεθος και η θέση του στοιχείου, να προσαρμόζονται αυτόματα με τη θέση του “παιδιού”.



Εικόνα 90- Οι ρυθμίσεις του Widget για το στοιχείο Background

Στη συνέχεια σέρνοντας την εικόνα bg από το πάνελ Assets πάνω στο στοιχείο backgrounds, δημιουργούμε το “παιδί” bg και ρυθμίζουμε κατάλληλα τη θέση του. Επίσης με την διαδικασία Add Component→ Add Renderer Component→ Sprite, έχουμε την δυνατότητα να προβάλουμε την εικόνα. Κάθε φορά που θα χρησιμοποιούμε το component Sprite θα έχουμε μετά την επιλογή να εμφανίζουμε εικόνες από το πάνελ Assets μέσα στην σκηνή.



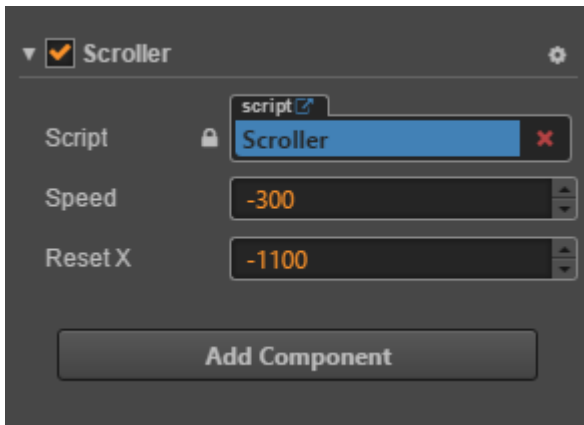
Εικόνα 91- Οι ρυθμίσεις του Sprite για το στοιχείο bg

Το επόμενο βήμα στην δημιουργία της σκηνής μας είναι να προσθέσουμε το γρασίδι. Με τον ίδιο τρόπο όπως και με το background, δημιουργούμε το στοιχείο ground και ως παιδιά δημιουργούμε τα ground 1, ground 2 και ground -1.



Εικόνα 92- Η στοίχιση των στοιχείων ground στην σκηνή του παιχνιδιού

Όπως και στα προηγούμενα παιχνίδια θα ρυθμίσουμε το γρασίδι ώστε να μετακινείται προς το αριστερό τμήμα της οθόνης για να δίνει την αίσθηση στον παίκτη ότι το bird είναι αυτό που πετάει. Για να το πετύχουμε αυτό, εκτός από το component sprite, στο στοιχείο ground που είναι και ο γονέας, με την διαδικασία Add Component → Add Custom Component θα προσθέσουμε το σενάριο Scroller το οποίο έχει δημιουργηθεί για αυτόν τον σκοπό.

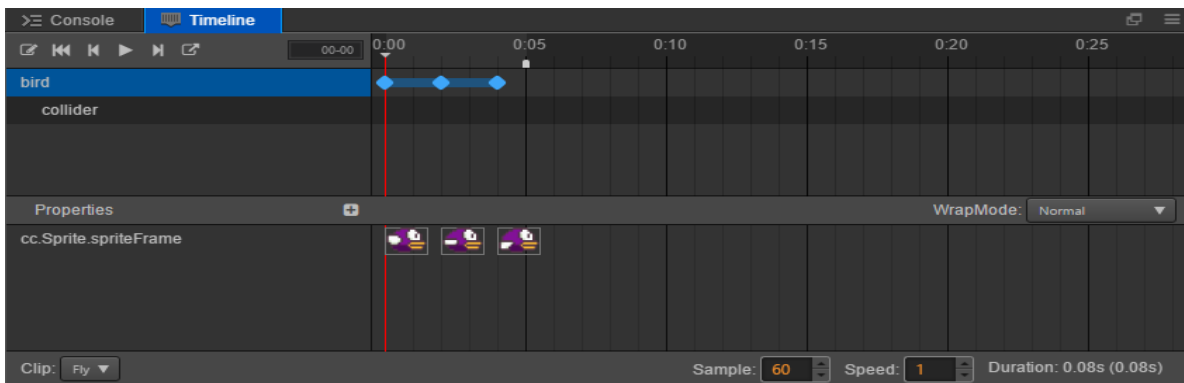


Εικόνα 93- Χρήση του σεναρίου Scroller στο στοιχείο ground.

Όταν αναπτύσσεται ένα σενάριο που προορίζεται για χρήση στην Cocos Creator, οι μεταβλητές που τυχόν θα χρησιμοποιηθούν δηλώνονται στο σενάριο, αλλά οι τιμές τους, ορίζονται από το πάνελ Properties. Έτσι για την αλλαγή μιας ταχύτητας για παράδειγμα, ο χρήστης δεν χρειάζεται να παρέμβει στον κώδικα.

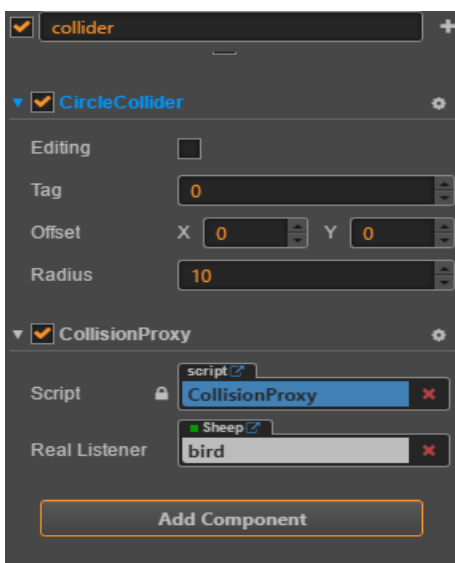
Για την συνεχή αναπαραγωγή των σωλήνων, θα επεξεργαστούμε το στοιχείο pipe, ως Prefab. Με τον όρο prefab εννοείται ένα στοιχείο που δημιουργείται στο πάνελ Node Tree, μεταφέρεται στο πάνελ Assets και έτσι έχουμε ένα στοιχείο το οποίο αναπαράγεται συνεχώς μέσα στο παιχνίδι μας. Αφού δημιουργήσουμε το prefab με το όνομα pipeLayer, προσθέτουμε σε αυτό τα σενάρια PipeGroupManager και PipeGroup, με τα οποία ρυθμίζεται ο χρόνος δημιουργίας των σωλήνων, η ταχύτητα και η απόσταση που θα έχουν μεταξύ τους στον άξονα Y οι σωλήνες, καθώς επίσης υπολογίζεται και η βαθμολογία του παίκτη.

Για την δημιουργία του χαρακτήρα του παιχνιδιού, δημιουργούμε και πάλι ένα νέο στοιχείο στο πάνελ Node Tree και στο πάνελ Properties του προσθέτουμε το component sprite. Στη συνέχεια στο πάνελ Assets, δημιουργούμε έναν νέο φάκελο με το όνομα Animations και μέσα σε αυτόν το φάκελο δημιουργούμε ένα Animation Clip με το όνομα fly. Αφού δημιουργήσουμε το clip αυτό, τότε επιλέγουμε ξανά το στοιχείο bird από το πάνελ Node Tree και στο πάνελ Properties του προσθέτουμε το component animation, ακολουθώντας την εξής διαδικασία: Add Component→ Add Other Component→ Animation. Στα νέα πεδία ρυθμίσεων που εμφανίζονται, στο πεδίο Default Clip, επιλέγουμε το clip fly. Για να ολοκληρωθεί η διαδικασία του animation, συνεχίζουμε στο πάνελ Timeline. Επιλέγουμε το στοιχείο bird και προσθέτουμε την ιδιότητα cc.Sprite.spriteFrame. Μετέπειτα επιλέγουμε τις εικόνες που αποτελούν το στοιχείο bird και τις τοποθετούμε μέσα στο Timeline. Τέλος στο στοιχείο bird, προσθέτουμε το component script και το συνδέουμε με το κατάλληλο σενάριο, το οποίο ρυθμίζει την συμπεριφορά του bird.



Εικόνα 94- Δημιουργία animation

Για την δημιουργία συγκρούσεων ανάμεσα στους σωλήνες και στο bird, θα δημιουργήσουμε ένα στοιχείο παιδί στο bird και θα το ονομάσουμε collider. Σε αυτό το στοιχείο θα προσθέσουμε τα components widget, script και CircleCollider. Στο component script θα συνδέσουμε το σενάριο το οποίο είναι υπεύθυνο για τις συγκρούσεις. Το component δημιουργεί έναν κύκλο τον οποίο τοποθετούμε στην σκηνή μας πάνω στο στοιχείο bird. Αυτός ο κύκλος που ακολουθεί το bird, βάση της μεταβλητής του σεναρίου, συγκρούεται με τους σωλήνες. Με το CircleCollider, ρυθμίζουμε την ακτίνα σύγκρουσης του στοιχείου collider.



Εικόνα 95- Ρυθμίσεις collider

Επόμενο βήμα στην ανάπτυξη του παιχνιδιού μετά την δημιουργία των συγκρούσεων είναι η εμφάνιση ενός μενού το οποίο θα εμφανίζει την τελική βαθμολογία και θα φέρει την επιλογή restart. Για αυτό τον σκοπό δημιουργούμε το στοιχείο gameOver στο Node Tree, που αποτελείται από τα εξής παιδιά: button_play, gameover, scoreTitle, score. Στο στοιχείο gameover, συνδέουμε το σενάριο GameOverMenu το οποίο καλεί το button_play και επανεκκινεί το παιχνίδι. Από το πάνελ Properties και την επιλογή Add Component → Add UI Component → Button δίνουμε στο παιδί button_play, την ιδιότητα του κουμπιού. Τα υπόλοιπα στοιχεία φέρουν απλά την ιδιότητα Label.

5. Συμπεράσματα

Σκοπός της ανάπτυξης αυτών των πέντε παιχνιδιών ήταν να μελετηθούν σε πρακτικό επίπεδο οι μηχανές παιχνιδιών που αναλύθηκαν στο θεωρητικό μέρος της εργασίας αυτής. Τα κύρια χαρακτηριστικά που ερευνήθηκαν ήταν η δημοφιλία, η χρηστικότητα, η εύκολη εκμάθηση, να είναι cross platform και γενικά οι δυνατότητες και η εμπειρία που προσφέρει η κάθε μηχανή στους χρήστες της. Βασικό γνώρισμα των τεσσάρων μηχανών είναι ο online editor που διαθέτουν και η cloud based τεχνολογία τους.

Πίνακας 7- Συγκριτικός Πίνακας Χαρακτηριστικών των Μηχανών Παιχνιδιών που εξετάστηκαν

<u>Χαρακτηριστικό</u>	<u>PlayCanvas</u>	<u>Construct 3</u>	<u>Wimi5</u>	<u>Ceilfire</u>	<u>Cocos Creator</u>
<u>Πρώτη Έκδοση</u>	09/05/2011	28/3/2017	17/01/2014	2015	17/12/2015
<u>Τελευταία Έκδοση</u>	21/05/2018	19/06/2018	22/12/2014	2016	15/03/2018
<u>Σύνολο Εκδόσεων</u>	132	115	3	2	10
<u>Κόστος</u>	Δωρεάν	Δωρεάν	Δωρεάν	Δωρεάν	Δωρεάν
<u>Ανάπτυξη Κώδικα</u>	Ναι	Όχι	Όχι	Όχι	Ναι
<u>Σύνδεση στο Internet για την χρήση της μηχανής</u>	Ναι	Όχι	Ναι	Ναι	Όχι
<u>Απαιτήσεις συστήματος</u>	115KB Engine Footprint	1MB Initial Download	---	---	Έκδοση 64-bit
<u>2D</u>	Ναι	Ναι	Ναι	Ναι	Ναι
<u>3D</u>	Ναι	Όχι	Όχι	Όχι	Ναι
<u>Cross Platform</u>	Ναι	Ναι	Ναι	Ναι	Ναι

<u>Instant Games</u>	Ναι	Ναι	Όχι	Όχι	Ναι
<u>Pageviews/μέρα</u>	17.690	15.160	1.910	348	16.700*
<u>Αξία Επίσημης Ιστοσελίδας</u>	38.740\$	33.200\$	4.190\$	760\$	36.570\$

* Η cocos creator φιλοξενείται στο επίσημο site της cocos2d-x. Το νούμερο αυτό των επισκέψεων, δεν αφορά αποκλειστικά και μόνο την Cocos Creator.

5.1. PlayCanvas

Η μηχανή PlayCanvas αποτελεί μια ολοκληρωμένη πρόταση για την ανάπτυξη ενός παιχνιδιού. Οι δυνατότητες που προσφέρει η συγκεκριμένη μηχανή είναι πολύ μεγαλύτερες από αυτές που χρειάζεται ένας χρήστης ο οποίος έχει λίγη εμπειρία και ασχολείται ερασιτεχνικά με την ανάπτυξη παιχνιδιών. Αυτό όμως δεν αποτελεί κριτήριο για να απορρίψει κάποιος την μηχανή. Το UI της μηχανής είναι πολύ φιλικό προς τον χρήστη και εύχρηστα δομημένο. Σε κάθε πεδίο αναφέρονται οι απαραίτητες πληροφορίες καθώς επίσης υπάρχουν παραπομπές που οδηγούν στα επίσημα tutorials και manuals της μηχανής. Ακόμα, χάρη στη φιλοσοφία του Open Source υπάρχουν πάρα πολλά έτοιμα παιχνίδια, τα οποία ο χρήστης μπορεί να τα επεξεργαστεί και να τα μελετήσει. Επίσης, η online κοινότητα της μηχανής είναι ενεργή με πάνω από 3.500 θέματα και απαντήσεις. Με τη σωστή μελέτη των στοιχείων της μηχανής, των tutorials και των manuals, η μηχανή μπορεί να χρησιμοποιηθεί από οποιοδήποτε που θέλει να ασχοληθεί με την ανάπτυξη παιχνιδιών.

5.2. Construct 3

Η εκμάθηση και η χρήση της Construct 3 είναι μια ευχάριστη και εύκολη διαδικασία. Το περιβάλλον της μηχανής είναι φιλικό και το πιο σημαντικό είναι πως σε κάθε βήμα, σε κάθε επιλογή, υπάρχει πληροφόρηση για την ταυτότητα και τη χρήση της διαδικασίας που έχει επιλέξει ο χρήστης. Η γνώση προγραμματισμού είναι επιθυμητή ωστόσο δεν είναι απαραίτητη. Κάποιος που έχει μηδαμινές γνώσεις πάνω στον προγραμματισμό, έχει τις ίδιες πιθανότητες να δημιουργήσει ένα πολύ καλό παιχνίδι με κάποιον που γνωρίζει προγραμματισμό, με την διαφορά ότι θα επενδύσει κάποιες ώρες παραπάνω να διαβάσει τις δυνατότητες των λειτουργιών που προσφέρει η Construct 3. Πολύ σημαντικό πλεονέκτημα σε αυτό, διαδραματίζουν τα tutorials και τα manuals που βρίσκονται στην επίσημη ιστοσελίδα. Αλλά και ακόμα εάν ο χρήστης δεν βρει αυτό που χρειάζεται για να βοηθηθεί, στο επίσημο blog της μηχανής βρίσκονται αμέτρητα άρθρα που μπορούν να τον

καθοδηγήσουν ενώ στο επίσημο forum της Construct 3 έχουν αναπτυχθεί πάνω από 103.000 θέματα με δεκάδες απαντήσεις. Η μηχανή Construct 3, είναι μια νέα μηχανή, αυτό που την ξεχωρίζει στο πεδίο των μηχανών είναι πως προέρχεται από την Construct 2, μια πολύ γνωστή και αξιόλογη μηχανή παιχνιδιών. Το γεγονός αυτό από μόνο του της δίνει το πλεονέκτημα της “αγοραστικής” δύναμης του κοινού. Η συνεχής ανακοίνωση νέων εκδόσεων, η υποστήριξη που παρέχει στους χρήστες και η δυναμική της κοινότητας των ενεργών χρηστών αλλά και των προγραμματιστών της την καθιστούν μια σημαντική επιλογή ανάμεσα στο πλήθος των μηχανών παιχνιδιών.

5.3. Wimi5

Η ανάπτυξη ενός παιχνιδιού στην μηχανή Wimi5 δεν απαιτεί την συγγραφή κώδικα. Ωστόσο είναι θεμιτό ο χρήστης να έχει γνώσεις πάνω στον προγραμματισμό. Αν και η μηχανή είναι πλήρως οπτικοποιημένη, εντούτοις είναι πιο περίπλοκη από άλλες μηχανές που κυκλοφορούν στο εμπόριο και κάποιος που δεν έχει ασχοληθεί στο παρελθόν με το αντικείμενο, θα δυσκολευτεί αρκετά στην δημιουργία του παιχνιδιού. Επιπλέον και για αυτή τη μηχανή οι πληροφορίες που διαθέτει είναι αρκετά περιορισμένες, ενώ δεν υπάρχει ενεργή κοινότητα χρηστών σε forum. Είναι μια καλή επιλογή για κάποιον που έχει ασχοληθεί με τον προγραμματισμό και ενδιαφέρεται να αποκτήσει νέες εμπειρίες, αλλά ένας χρήστης που δεν έχει ασχοληθεί ποτέ του με τον προγραμματισμό, θα αντιμετωπίσει πολλές δυσκολίες.

5.4. Ceilfire

Η φιλοσοφία της μηχανής είναι πολύ απλοποιημένη και αυτό περιορίζει αρκετά τις δυνατότητες που θα μπορούσε να έχει πάνω στην ανάπτυξη παιχνιδιών. Αν και η μηχανή βρίσκεται ενεργή, η ανάπτυξη της έχει σταματήσει σχεδόν από τον δεύτερο χρόνο λειτουργίας της, δεν υπάρχουν νέες εκδόσεις και τα tutorials της παρέχουν συγκεκριμένες και περιορισμένες πληροφορίες. Η μηχανή Ceilfire είναι κατάλληλη για κάποιον που δεν έχει ασχοληθεί ποτέ με τον προγραμματισμό και την ανάπτυξη παιχνιδιών και θέλει να εξοικειωθεί με το περιβάλλον μιας μηχανής και την λογική σκέψη πίσω από ένα παιχνίδι. Δεν συνίσταται ωστόσο για την έκδοση παιχνιδιών στην πλατφόρμα καθώς πλέον αντιμετωπίζει θέματα λειτουργίας.

5.5. Cocos Creator

Η μηχανή Cocos Creator είναι η μόνη από τις μηχανές που αναλύθηκαν η οποία δεν διαθέτει online editor και απαιτούσε την εγκατάσταση ειδικού λογισμικού. Αυτό από μόνο του είναι χρονοβόρο και αυξάνει τις απαιτήσεις του συστήματος. Ένα μειονέκτημα είναι ότι η μηχανή

δεν διαθέτει editor και απαιτείται η εγκατάσταση ενός επιπλέον προγράμματος επεξεργασίας και ανάπτυξης κώδικα, που αυτό συνεπάγεται επιπλέον χρόνο και ακόμα μεγαλύτερες απαιτήσεις από το σύστημα του χρήστη. Το UI της μηχανής δεν διαφέρει ιδιαίτερα από αυτά των υπολοίπων μηχανών. Ωστόσο η μηχανή δεν είναι το ίδιο εύχρηστη όσο οι άλλες μηχανές που αναλύθηκαν. Αρχικά τα tutorials που υπάρχουν στην επίσημη ιστοσελίδα της μηχανής καλύπτουν τα στοιχεία της μηχανής, αλλά η διαρρύθμιση της στοίχισης τους, είναι αρκετά χαοτική καθώς ένα θέμα μπορεί να αναπτυχθεί σε παραπάνω από ένα σημεία. Η μηχανή δεν διαθέτει το δικό της forum. Τα θέματα που αφορούν την μηχανή Cocos Creator φιλοξενούνται στο forum της Cocos2d-x. Τα θέματα που απασχολούν την Cocos Creator ανέρχονται στον αριθμό 1828. Σε γενικά πλαίσια είναι μια μηχανή που πρέπει να γνωρίζει κάποιος προγραμματισμό και να έχει ασχοληθεί με διάφορα άλλα προγράμματα, διότι λειτουργεί με πολύ αυστηρή σειρά. Σε ορισμένα δε σημεία, δεν ισχύει η ιδιότητα της αναίρεσης. Αυτό έχει ως αποτέλεσμα εάν ο χρήστης επιθυμεί να επιστρέψει σε μια προηγούμενη κατάσταση να πρέπει να επαναλάβει τα βήματα που είχε ακολουθήσει εκ νέου. Ένα σοβαρό επίσης πρόβλημα προέκυψε κατά την διάρκεια δημοσίευσης του παιχνιδιού. Το σύστημα συνεχώς αποτύγχανε να ολοκληρώσει την διαδικασία δημοσίευσης.

5.6. Κατάταξη βάσει κριτηρίων

Από τις πέντε αυτές μηχανές, η Construct 3, η Ceilfire και η Wimi5 δεν απαιτούν την δημιουργία κώδικα για την ανάπτυξη των παιχνιδιών. Σε επίπεδο ευκολίας οι μηχανές Construct 3 και Ceilfire θα μπορούσαν να βαθμολογηθούν το ίδιο και ίσως για κάποιους χρήστες η Ceilfire να είναι λίγο πιο εύκολη. Αντίθετα, για την ανάπτυξη παιχνιδιών με τις μηχανές PlayCanvas και Cocos Creator, βασική προϋπόθεση είναι η γνώση της γλώσσας Javascript.

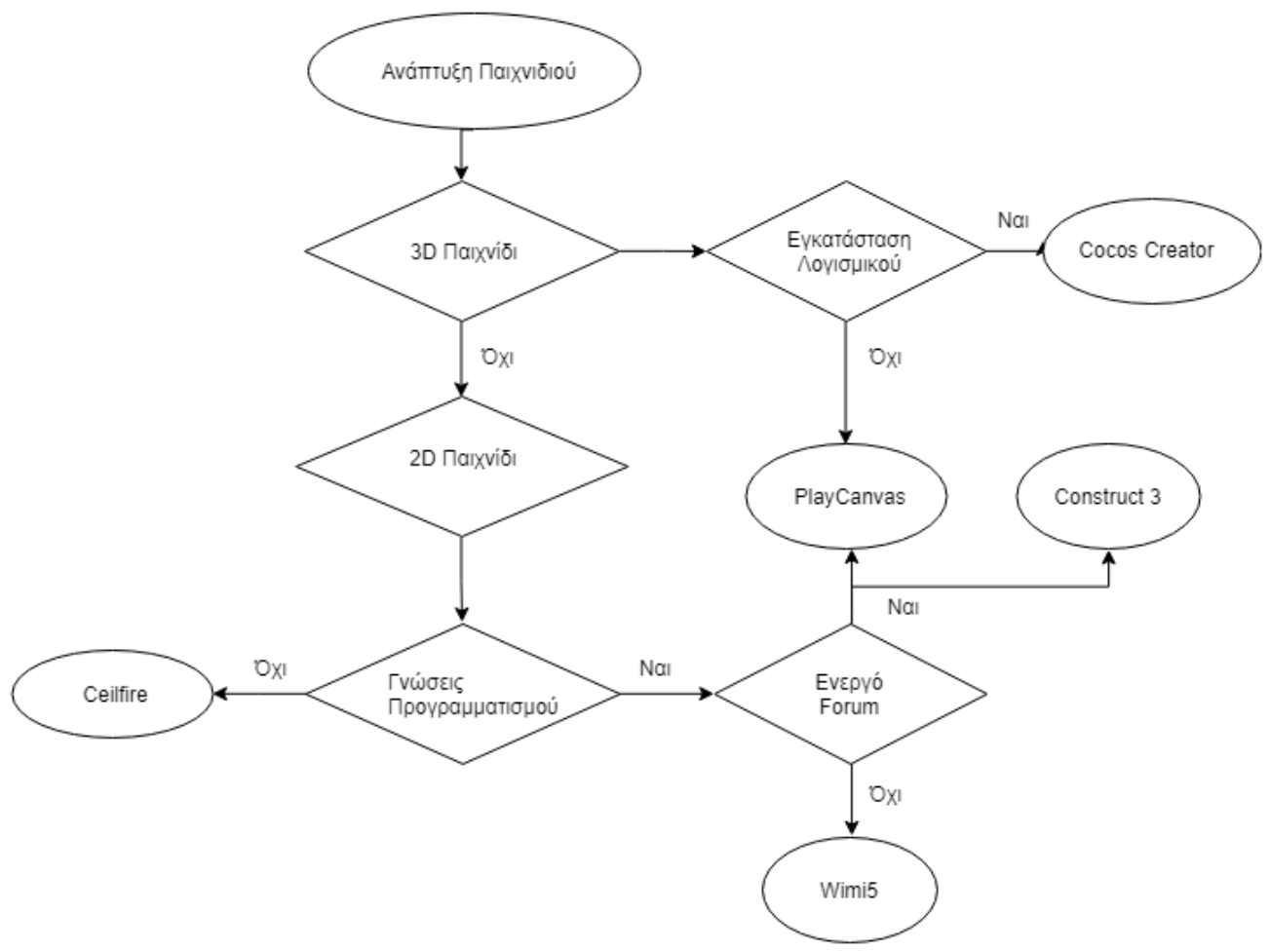
Από την θεωρητική μελέτη των μηχανών και την ενασχόληση για την ανάπτυξη των παιχνιδιών κατατάσσουμε τις μηχανές από την πιο εύκολη ως την πιο δύσκολη ως εξής:

1. Ceilfire
2. Construct 3
3. Wimi5
4. PlayCanvas
5. Cocos Creator

Η κατάταξη των μηχανών ως προς την ευκολία χρήσης τους σε συνάρτηση με την βοήθεια που παρέχεται στον χρήστη και την αξιοποίηση του παιχνιδιού είναι η εξής:

1. Construct 3
2. PlayCanvas

- 3. Wimi5
- 4. Cocos Creator
- 5. Ceilfire



Εικόνα 96- Διάγραμμα ροής για την επιλογή της κατάλληλης μηχανής

Επίλογος

Η ανάπτυξη ενός παιχνιδιού από το 0 είναι μια πολύ ενδιαφέρουσα διαδικασία η οποία μπορεί να αποσκοπεί σε εμπορικούς, διαφημιστικούς, δημιουργικούς ή ακόμα και σε εκπαιδευτικούς σκοπούς. Η επιλογή της κατάλληλης μηχανής, εκτός από την έρευνα που είναι απαραίτητη να γίνει πάνω σε συγκεκριμένα χαρακτηριστικά της μηχανής, βασίζεται και στην προσωπική επιλογή του χρήστη, καθώς θα επενδύσει πάρα πολλές ώρες εργαζόμενος πάνω σε αυτήν. Βάση λοιπόν της θεωρητικής μελέτης που πραγματοποιήσαμε για την κάθε μηχανή αλλά του παιχνιδιού που αναπτύξαμε στην κάθε μια, καταλήγουμε ότι από τις πέντε μηχανές οι καταλληλότερες για την ανάπτυξη παιχνιδιών είναι οι Construct 3 και η PlayCanvas. Το κυριότερο μειονέκτημα των υπολοίπων μηχανών ήταν η έλλειψη υλικού, υποστήριξης καθώς και το δύσχρηστο UI όταν περνάμε στο πρακτικό μέρος της ανάπτυξης του παιχνιδιού. Σε αντίθεση, το φιλικό UI, ο μεγάλος όγκος του υλικού εκμάθησης, η συνεχής πορεία στο χώρο, η μεγάλη και ενεργή κοινότητα χρηστών, το πλήθος των παραδειγμάτων, η δυνατότητα αποθήκευσης και έκδοσης του παιχνιδιού, είναι μερικά από τα πλεονεκτήματα που διαθέτουν οι μηχανές PlayCanvas και Construct 3. Ανάμεσα σε αυτές τις δυο, ο αναγνώστης μπορεί να επιλέξει σύμφωνα με δυο κριτήρια:

1. Τις γνώσεις που έχει πάνω στην γλώσσα Javascript ή
2. Εάν επιθυμεί να αναπτύξει 3D παιχνίδι,

καθώς η μηχανή Construct 3 δεν υποστηρίζει μέχρι στιγμής αυτή τη δυνατότητα.

Βιβλιογραφία

Ελληνική

Βακιρτζής Στυλιανός, 2017: Η χρήση οπτικών εργαλείων ανάπτυξης διαδικτυακών παιχνιδιών (online games). Συγκριτική μελέτη δυνατοτήτων και παραδειγματική εφαρμογή χρήσης τους στην κατασκευή ολοκληρωμένου διαδικτυακού παιχνιδιού, ΤΕΙ ΚΡΗΤΗΣ

Γιακαλάρας Μάριος, Απρίλιος 2016 Εικονικοί Κόσμοι: Διερεύνηση των ζητημάτων των σύγχρονων μηχανών παιχνιδιών ως προς το φωτορεαλισμό στον τομέα φωτισμού και του αντίκτυπου στο χρήστη ενός παιχνιδιού

Κούτρας Θεόδωρος, Μάρτιος 2014: Ανάπτυξη και αξιολόγηση ενός διαδραστικού ηλεκτρονικού παιχνιδιού με αφηγηματικό περιεχόμενο, Πάτρα

Πύλιος Θώμας και Σοϊδής Ιωάννης, Νοέμβριος 2011: Τρισδιάστατες Γραφικές Αναπαραστάσεις. Μια εφαρμογή στο κτήριο του Τμήματος Διαχείρισης Πληροφοριών του ΤΕΙ Καβάλας, με χρήση αρχιτεκτονικού προγράμματος και Πλατφόρμας ανάπτυξης παιχνιδιών

Ξένη

Delorme Rick, 2014: Programming in HTML5 with JavaScript and CSS3 (MCSD) 1st Edition

Elliot Eric, 2014: Programming JavaScript Applications

Freeman & Robson, 2011: Head First HTML5 Programming: Building Web Apps with JavaScript 1st Edition

Kashyap Pankaj, Μάιος 2015: Investigation into the use of HTML 5 game engines to create a responsive social educational game for children, Charles Darwin University

Marin Humberto, Alor-Hernández Giner, Zatarain Ramon, Lucia Barron Maria, Jorge Luis García-Alcaraz, Journal of Information Technology Research Volume 10, Issue 4 Οκτώβριος 2017: A Brief Review of Game Engines for Educational and Serious Games Development

Parisi Tony, 2014: Programming 3D Applications with HTML5 and WebGL: 3D Animation and Visualization for Web Pages

Viveros, M. C., & Garcva, D., 2009. Elaboracion de una guva para el desarrollo de aplicaciones en extjs. Instituto Tecnologico de Orizaba.

Ιστοσελίδες

wikipedia, μηχανή παιχνιδιού, 2017

<https://el.wikipedia.org/wiki/%CE%9C%CE%B7%CF%87%CE%B1%CE%BD%CE%AE%CF%80%CE%B1%CE%B9%CF%87%CE%BD%CE%B9%CE%B4%CE%B9%CE%BF%CF%8D> [07/09/2017]

Gamedev Glossary: Library vs Framework vs Engine

<http://www.gamefromscratch.com/post/2015/06/13/GameDev-Glossary-Library-Vs-Framework-Vs-Engine.aspx> [10/09/2017]

wikipedia, library, 2017

[https://el.wikipedia.org/wiki/%CE%92%CE%B9%CE%B2%CE%BB%CE%B9%CE%BF%CE%B8%CE%AE%CE%BA%CE%B7_\(%CF%85%CF%80%CE%BF%CE%BB%CE%BF%CE%B3%CE%B9%CF%83%CF%84%CE%AD%CF%82\)](https://el.wikipedia.org/wiki/%CE%92%CE%B9%CE%B2%CE%BB%CE%B9%CE%BF%CE%B8%CE%AE%CE%BA%CE%B7_(%CF%85%CF%80%CE%BF%CE%BB%CE%BF%CE%B3%CE%B9%CF%83%CF%84%CE%AD%CF%82))

framework

<https://info.cimatrix.com/blog/bid/22339/What-is-a-Software-Framework-And-why-should-you-like-em>

<http://tom.lokhorst.eu/2010/09/why-libraries-are-better-than-frameworks>

<https://stackoverflow.com/questions/3057526/framework-vs-toolkit-vs-library>

game engine

<https://gamedev.stackexchange.com/questions/31772/what-is-the-difference-between-a-game-framework-and-a-game-engine>

<https://gamyguru.wordpress.com/2012/07/16/inside-the-game-general-game-structure/>

<https://onclick786.blogspot.gr/2016/09/new.html>

<http://www.html5gamedevs.com/topic/14947-frameworks-vs-game-engines/>

<https://www.giantbomb.com/profile/michaelenger/blog/game-engines-how-do-they-work/101529/>

WebGL 2018

<https://el.wikipedia.org/wiki/WebGL>

<https://technewsingreek.blogspot.gr/2013/02/webgl.html>

<https://github.com/dionyziz/webgl-seminar/blob/04b85dbb666371bc7393d4d3203b9990d51302ec/slides/slides-ntua.pptx>

<https://technewsingreek.blogspot.com/2013/02/webgl.html#ixzz56E1Mcsr6>

Programming 3D Applications with HTML5 and WebGL: 3D Animation and Visualization for Web Pages, Tony Parisi 2014

<https://www.youtube.com/watch?v=axKIN0-Vm2g>

Lists of Game Engines

<https://html5gameengine.com/>

<http://www.gamepixmap.com/blog/the-big-list-of-html5-3d-games-engines/>

<http://www.gamepixmap.com/blog/the-big-list-of-2d-html5-games-engines/>

<https://www.htmlgoodies.com/html5/javascript/the-top-html5-canvas-game-engines.html>

http://www.designyourway.net/blog/resources/javascript-html5-game-engines-libraries-51-examples/?utm_source=the-webdesigner.co&utm_medium=newsletter

<http://ntt.cc/2011/01/31/66-open-source-javascript-game-engine-for-serious-developers.html>

<https://www.mevuk.com/development/the-top-16-game-engines-for-2014>

<https://colorlib.com/wp/javascript-engines-for-building-games/>

https://en.wikipedia.org/wiki/List_of_game_engines

<https://alternativeto.net/software/playcanvas/>

<https://www.capterra.com/game-development-software/>

<https://blog.sagi.pl.com/game-development-tools/>

Construct 3 επίσημη ιστοσελίδα 2018

<https://www.construct.net>

Construct 3 Community

<https://www.construct.net/gb/community>

Construct 3 Pricing

<https://www.construct.net/gb/make-games/buy-construct-3>

Construct 3 Tutorials

<https://www.construct.net/gb/learn>

Construct 3 User Manuals

<https://www.construct.net/gb/make-games/manuals/construct-3/getting-started/get-construct-3>

Tappy Plane #PART 1 - Construct 2 Tutorial

<https://www.youtube.com/watch?v=w1xE3IXdL9g>

Tappy Plane #PART 2 - Construct 2 Tutorial

https://www.youtube.com/watch?v=R_h86vryYSs&t=2s

<https://www.scirra.com/tutorials/857/flappy-birds-clone-in-10-minutes/page-1#tuttop>

PlayCanvas επίσημη ιστοσελίδα 2018

<https://playcanvas.com/explore>

PlayCanvas forum

<https://forum.playcanvas.com/>

PlayCanvas Tutorials

<https://developer.playcanvas.com/en/tutorials/>

PlayCanvas User Manuals

<https://developer.playcanvas.com/en/user-manual/>

PlayCanvas Pricing

<https://playcanvas.com/plans>

<http://mail.gameworld.gr/forum/42/152930-playcanvas-engine>

<https://github.com/playcanvas/engine>

<https://en.wikipedia.org/wiki/PlayCanvas>

PlayCanvas Tutorial 1 - Getting started with PlayCanvas

<https://www.youtube.com/watch?v=fXc-JjH2nNo>

PlayCanvas Tutorial 2 - Entities and materials

<https://www.youtube.com/watch?v=sk6eUNERs4E&t=107s>

PlayCanvas Tutorial 3 – Lighting

<https://www.youtube.com/watch?v=FSi13KQYCDI&t=21s>

PlayCanvas Tutorial 4 - Skyboxes in PlayCanvas

<https://www.youtube.com/watch?v=eKjTachH36Qg>

PlayCanvas Tutorial 5 - Particle systems in PlayCanvas

<https://www.youtube.com/watch?v=ys3wac2vaSE>

PlayCanvas Tutorial 6 - Overview of scripts in PlayCanvas

<https://www.youtube.com/watch?v=ING7ndSUK9o&t=364s>

PlayCanvas Tutorial 7 - Make a 3D object move with code

<https://www.youtube.com/watch?v=xgJKga1huBE&t=17s>

PlayCanvas Tutorial 8 - Moving objects with keyboard input

https://www.youtube.com/watch?v=VpJxc_iRmeM&t=9s

PlayCanvas Tutorial 9 - Rotating objects with keyboard input

<https://www.youtube.com/watch?v=amHq5qRBWP4>

PlayCanvas Tutorial 10 - Rigid Bodies and Collision in PlayCanvas

<https://www.youtube.com/watch?v=FbMwqd9MF3c&t=4s>

Wimi5 επίσημη ιστοσελίδα 2018

<http://wimi5.com/>

Wimi5 Pricing

<http://wimi5.com/pricing/>

Wimi5 Forum

<http://wimi5.com/forums/>

Wimi5 Tutorials

<http://wimi5.com/learn/tutorials/>

Wimi5 documentation

<http://developer.wimi5.com/doc/latest/>

Wimi5 Example

<http://wimi5.com/my-first-game-with-wimi5/>

Statistics

<http://www.websiteoutlook.com/>

Cocos Creator επίσημη ιστοσελίδα 2018

<http://www.cocos2d-x.org/products>

Cocos Creator documentation 2018

<http://www.cocos2d-x.org/docs/creator/manual/en/>

Cocos Creator animation 2018

<https://www.youtube.com/watch?v=KgFXVneSb7k>

Cocos Creator information 2018

<http://cocos.sonarlearning.co.uk/docs/what-is-cocos-studio>

Cocos Creator duang sheep 2018

<https://github.com/cocos-creator/tutorial-duang-sheep>

Ceilfire επίσημη ιστοσελίδα 2018

<https://ceilfire.com/>

Ceilfire tutorials 2018

<https://ceilfire.com/news/ceilfire/>

Ceilfire free assets 2018

<https://ceilfire.com/market>