

ΠΑΝΕΠΙΣΤΗΜΙΟ ΜΑΚΕΔΟΝΙΑΣ
ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
ΤΜΗΜΑΤΟΣ ΕΦΑΡΜΟΣΜΕΝΗΣ ΠΛΗΡΟΦΟΡΙΚΗΣ

Ανάπτυξη on-line παιχνιδιού με τη Blockly για την καλλιέργεια της
Υπολογιστικής Σκέψης σε μικρούς μαθητές

Διπλωματική Εργασία

του

Καρακάση Χριστόδουλου

Θεσσαλονίκη, Φεβρουάριος 2020

Ανάπτυξη on-line παιχνιδιού με τη Blockly για την καλλιέργεια της
Υπολογιστικής Σκέψης σε μικρούς μαθητές

Καρακάσης Χριστόδουλος

Πτυχίο Εφαρμοσμένης Πληροφορικής, Παν. Μακεδονίας, 2018

Διπλωματική Εργασία

υποβαλλόμενη για τη μερική εκπλήρωση των απαιτήσεων του

ΜΕΤΑΠΤΥΧΙΑΚΟΥ ΤΙΤΛΟΥ ΣΠΟΥΔΩΝ ΣΤΗΝ ΕΦΑΡΜΟΣΜΕΝΗ
ΠΛΗΡΟΦΟΡΙΚΗ

Επιβλέπων Καθηγητής
Ξυνόγαλος Στυλιανός

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την

ΞΥΝΟΓΑΛΟΣ ΣΤΕΛΙΟΣ

ΣΑΤΡΑΤΖΕΜΗ ΜΑΡΙΑ

ΚΑΣΚΑΛΗΣ ΘΕΩΔΩΡΟΣ

.....

.....

.....

Καρακάσης Χριστόδουλος

.....

Περίληψη

Σκοπός της παρούσας διπλωματικής εργασίας είναι η σχεδίαση και η ανάπτυξη ενός διαδικτυακού παιχνιδιού σοβαρού σκοπού για την καλλιέργεια της Υπολογιστικής Σκέψης σε μικρούς μαθητές, αλλά και σε όσους θέλουν να ψυχαγωγηθούν και να μάθουν παίζοντας το παιχνίδι. Το παιχνίδι ονομάζεται "BlocklyScript" και για την ανάπτυξή του χρησιμοποιήθηκαν δύο σημαντικά εργαλεία. Το πρώτο εργαλείο είναι το πλαίσιο ανάπτυξης δισδιάστατων παιχνιδιών Phaser, σχεδιασμένο από την ομάδα προγραμματιστών Photon Storm και με κατοχυρωμένη άδεια λογισμικού από το πανεπιστήμιο MIT. Το δεύτερο εργαλείο είναι η βιβλιοθήκη Blockly η οποία βασίζεται στην JavaScript και σχεδιάστηκε από την Google. Η βιβλιοθήκη Blockly συνεισφέρει στην δημιουργία γλωσσών οπτικού προγραμματισμού και συντακτών βασισμένων σε μπλοκ.

Για την ανάπτυξη του "BlocklyScript" αποτέλεσε απαραίτητη προϋπόθεση η μελέτη διαδικτυακών παιχνιδιών σοβαρού σκοπού που ως κύριο στόχο έχουν την καλλιέργεια της Υπολογιστικής Σκέψης. Τα κριτήρια επιλογής των παιχνιδιών προς μελέτη δεν περιορίστηκαν μόνο σε παιχνίδια που υποστηρίζουν τον οπτικό προγραμματισμό, αλλά επιλέχθηκαν και αναγνωρισμένα παιχνίδια, δημοφιλή στον ιστό, που εστιάζουν στον κλασικό προγραμματισμό με κειμενική γλώσσα.

Σχετικά με το "BlocklyScript" παρουσιάζονται μερικές από τις βασικές έννοιες που περιλαμβάνονται σε όλες σχεδόν τις γλώσσες προγραμματισμού. Ο χρήστης παίζοντας το παιχνίδι και χωρίς να είναι απαραίτητη η γνώση κάποιας γλώσσας προγραμματισμού μαθαίνει τις επαναλήψεις, τις δηλώσεις "if", την έννοια της μεθόδου και την έννοια της μεταβλητής. Οι παραπάνω έννοιες παρουσιάζονται μέσα στο παιχνίδι με την μορφή μπλοκ, τα οποία ο χρήστης μπορεί να σύρει και να τα συνδέσει μεταξύ τους με στόχο να νικήσει το αντίστοιχο επίπεδο. Επιπλέον το παιχνίδι υποστηρίζει την αποθήκευση της προόδου του χρήστη σε βάση δεδομένων χωρίς να τον περιορίζει να το τερματίσει με μια προσπάθεια.

Με την ολοκλήρωση της βιβλιογραφικής επισκόπησης αναλύονται τα επίπεδα του παιχνιδιού, το πλαίσιο σχεδίασης και ανάπτυξης του "BlocklyScript" και η αρχιτεκτονική του λογισμικού. Τέλος γίνεται μια αναφορά για τα αποτελέσματα και τις

εντυπώσεις των χρηστών που έπαιξαν το παιχνίδι καθώς και τα συμπεράσματα της έρευνας.

Λέξεις Κλειδιά: Υπολογιστική Σκέψη, παιχνίδια σοβαρού σκοπού, Phaser, Blockly, οπτικός προγραμματισμός

Abstract

The purpose of this master thesis is to design and develop a serious online game aimed at cultivating Computational Thinking for young students, as well as for those who want to have fun and learn by playing the game. The game is called "BlocklyScript" and two important tools were used to develop it. The first tool is the Phaser, a 2D game framework designed by the Photon Storm developers team and licensed by MIT university. The second tool is the Blockly JavaScript library. The Blockly library contributes to the development of block-based visual programming languages and editors.

A prerequisite for the development of "BlocklyScript" was the study of other serious online games that are primarily aimed at the development of Computational Thinking. The selection criteria for the games to be studied were not only limited to games that support visual programming, but also recognized web-based games that focus on classic scripting.

"BlocklyScript" presents some of the basic concepts that almost all programming languages have. The user learns the loops, the "If" statements, the methods and the variables by playing the game and without having to know any programming language. The above concepts are presented within the game in the form of blocks, which the user can draw and connect with each other in order to beat the corresponding level. In addition, the game stores the user's progress in a database.

Upon the completion of the online game research, the game levels are analyzed, BlocklyScript's design and development framework is presented and then the software's architecture takes place.

Finally a report is made about the results and impressions of the users who played the game with conclusions about the research.

Keywords: Computational Thinking, serious games, Phaser, Blockly, visual programming

Ευχαριστίες

Η εκπόνηση της παρούσας διπλωματικής εργασίας έγινε στα πλαίσια του μεταπτυχιακού προγράμματος σπουδών του τμήματος Εφαρμοσμένης Πληροφορικής του Πανεπιστημίου Μακεδονίας. Προσωπικά, θα ήθελα να ευχαριστήσω τον επιβλέπον καθηγητή μου κ. Στέλιο Ξυνόγαλο για την αναλυτική καθοδήγησή του κατά την διάρκεια της σχεδίασης και της υλοποίησης της εργασίας. Τέλος θα ήθελα να ευχαριστήσω και την οικογένεια μου που μου έδωσαν την δυνατότητα να συνεχίσω τις σπουδές μου σε μεταπτυχιακό επίπεδο.

Περιεχόμενα

1	Εισαγωγή	1
1.1	Πρόβλημα – Σημαντικότητα του θέματος	1
1.2	Σκοπός – Στόχοι	1
1.3	Συνεισφορά	1
1.4	Διάρθρωση της μελέτης	2
2	Βιβλιογραφική Επισκόπηση – Θεωρητικό Υπόβαθρο	3
2.1	Πώς ορίζεται ένα παιχνίδι;	3
2.2	Πώς ορίζεται ένα παιχνίδι "σοβαρού σκοπού";	4
2.3	Η έννοια της Υπολογιστικής Σκέψης	6
2.4	Οπτικός Προγραμματισμός	8
2.4.1	Οπτικός προγραμματισμός στην εκπαίδευση	8
2.4.2	Γλώσσες Οπτικού Προγραμματισμού στην εκπαίδευση	10
2.4.3	Ανάπτυξη γλωσσών οπτικού προγραμματισμού με το Blockly	18
2.5	Παιχνίδια σοβαρού σκοπού για την ανάπτυξη της Υπολογιστικής Σκέψης	24
2.5.1	Code Combat	24
2.5.2	Hour of Code: Minecraft	28
2.5.3	Rapid Router	30
2.5.4	Run Marco!	31
2.6	Συγκριτική ανάλυση παιχνιδιών σοβαρού σκοπού	33
2.6.1	Συμπεράσματα ανάλυσης	36
3	Περιγραφή παιχνιδιού	39
3.1	Εισαγωγή	39
3.2	Είσοδος και εγγραφή χρήστη στο παιχνίδι	40
3.3	Αρχική οθόνη χρήστη	41
3.4	Επίπεδα παιχνιδιού - διδακτικοί στόχοι	45
3.4.1	Γενικά χαρακτηριστικά	45
3.4.2	Πρώτο επίπεδο παιχνιδιού - Movement	46
3.4.3	Δεύτερο επίπεδο παιχνιδιού - Jump	53
3.4.4	Τρίτο επίπεδο παιχνιδιού - Repeat 'n' times	57
3.4.5	Τέταρτο επίπεδο παιχνιδιού - Repeat While	61
3.4.6	Πέμπτο επίπεδο παιχνιδιού - Repeat Until	69

3.4.7 Έκτο επίπεδο παιχνιδιού - Simple if	73
3.4.8 Έβδομο επίπεδο παιχνιδιού - If-else	77
3.4.9 Ογδοο επίπεδο παιχνιδιού - Nested if	81
3.4.10 Ένατο επίπεδο παιχνιδιού - Method	86
3.4.11 Δέκατο επίπεδο παιχνιδιού - Limited Blocks	91
3.4.12 Εντέκατο επίπεδο παιχνιδιού - Variable	93
3.4.13 Δωδέκατο επίπεδο παιχνιδιού - Restore the stars	97
4 Σχεδίαση και ανάπτυξη του παιχνιδιού	99
4.1 Πλαίσιο σχεδίασης	99
4.2 Πλαίσιο ανάπτυξης	105
4.3 Γραφικά και ήχοι παιχνιδιού	106
4.4 Αρχιτεκτονική Λογισμικού	107
4.4.1 Είσοδος - Εγγραφή στο παιχνίδι	107
4.4.2 Οργάνωση βάσης δεδομένων του παιχνιδιού	110
4.4.3 Λήψη και ενημέρωση των δεδομένων της βάσης	112
4.4.4 Σχεδίαση επιπέδων παιχνιδιού	116
5 Αξιολόγηση παιχνιδιού	132
5.1 Μεθοδολογία	132
5.2 Συμμετέχοντες	133
5.3 Αποτελέσματα αξιολόγησης	133
5.3.1 Δημογραφικά δεδομένα	133
5.3.2 Χρηστικότητα	135
5.3.3 Εμπειρία παίκτη	136
5.3.4 Ερωτήσεις ανοιχτού τύπου	139
5.4 Συμπεράσματα αξιολόγησης	142
6 Επίλογος	144
6.1 Σύνοψη και συμπεράσματα	144
6.2 Όρια και περιορισμοί έρευνας	145
6.3 Μελλοντικές επεκτάσεις	146
Βιβλιογραφία	147
7 Παράρτημα Α	149
7.1 Δημοσίευση ιστοσελίδας στο Διαδίκτυο	149

Κατάλογος Εικόνων

Εικόνα 1: Ορισμός των παιχνιδιών σοβαρού σκοπού (Laamarti, Eid & El Saddik, 2014)	5
Εικόνα 2: Υπολογιστική Σκέψη (Harimuti et al.(2019)).....	7
Εικόνα 3: Προγραμματισμός σε μορφή κειμένου	9
Εικόνα 4: Οπτικός προγραμματισμός με μπλοκ.....	9
Εικόνα 5: Το λογότυπο της γλώσσας Scratch	11
Εικόνα 6: Η γλώσσα Scratch.....	13
Εικόνα 7: Λογότυπο της γλώσσας Kodu.....	15
Εικόνα 8: Περιβάλλον σχεδίασης της Kodu	16
Εικόνα 9: Δημιουργία χαρακτήρα ή αντικειμένου στη Kodu	16
Εικόνα 10: Προγραμματισμός χαρακτήρων στην Kodu	17
Εικόνα 11: Λογότυπο της βιβλιοθήκης Blockly	18
Εικόνα 12: Οργανισμοί που χρησιμοποιούν την βιβλιοθήκη Blockly	19
Εικόνα 13: Οδηγός εκμάθησης της βιβλιοθήκης Blockly.....	20
Εικόνα 14: Block Factory.....	22
Εικόνα 15: Workspace Factory	23
Εικόνα 16: Λογότυπο Code Combat	24
Εικόνα 17: Επιλογή χαρακτήρα και γλώσσας στο Code Combat.....	26
Εικόνα 18: Επίπεδο παιχνιδιού Code Combat	27
Εικόνα 19: Επιλογή χαρακτήρα στο Minecraft Hour of Code.....	29
Εικόνα 20: Επίπεδο παιχνιδιού Minecraft Hour of Code.....	29
Εικόνα 21: Επίπεδα παιχνιδιού Rapid Router	30
Εικόνα 22: Επίπεδο παιχνιδιού Rapid Router	31
Εικόνα 23: Επιλογή χαρακτήρα στο παιχνίδι Run Marco!	32
Εικόνα 24: Επίπεδο παιχνιδιού Run Marco!	33
Εικόνα 25: Οθόνη εγγραφής BlocklyScript	40
Εικόνα 26: Οθόνη εισόδου με εσφαλμένα στοιχεία.....	41
Εικόνα 27: Αρχική οθόνη - επιλογή χαρακτήρα	41
Εικόνα 28: Οθόνη καλωσορίσματος - έναρξης παιχνιδιού.....	42
Εικόνα 29: Σενάριο BlocklyScript	43
Εικόνα 30: Επίπεδα του BlocklyScript με ξεκλειδωμένο το πρώτο επίπεδο	44
Εικόνα 31: Πίνακας με ονόματα των πέντε υψηλότερων σκορ.....	44

Εικόνα 32: Οθόνη πρώτου επιπέδου BlocklyScript	46
Εικόνα 33: Οδηγίες πρώτου επιπέδου	46
Εικόνα 34: Κουμπί βοήθειας πρώτου επιπέδου	49
Εικόνα 35: Βέλτιστη λύση πρώτου επιπέδου	50
Εικόνα 36: Παράδειγμα βέλτιστης λύσης πρώτου επιπέδου.....	51
Εικόνα 37: Παράδειγμα απλής λύσης πρώτου επιπέδου.....	51
Εικόνα 38: Παράδειγμα αποτυχημένης προσπάθειας πρώτου επιπέδου.....	52
Εικόνα 39: Οθόνη δεύτερου επιπέδου.....	53
Εικόνα 40: Οδηγίες δεύτερου επιπέδου	53
Εικόνα 41: Βέλτιστη λύση δεύτερου επιπέδου	55
Εικόνα 42: Περίπτωση πληγωμένου ήρωα.	56
Εικόνα 43: Οθόνη τρίτου επιπέδου	57
Εικόνα 44: Οδηγίες τρίτου επιπέδου	57
Εικόνα 45: Βέλτιστη λύση τρίτου επιπέδου	60
Εικόνα 46: Οθόνη τέταρτου επιπέδου.....	61
Εικόνα 47: Οδηγίες τέταρτου επιπέδου.....	62
Εικόνα 48: Βέλτιστη λύση τετάρτου επιπέδου.....	64
Εικόνα 49: Έλεγχοι βέλτιστης λύσης τέταρτου επιπέδου (μέρος α).....	66
Εικόνα 50: Έλεγχοι βέλτιστης λύσης τέταρτου επιπέδου (μέρος β).....	67
Εικόνα 51: Έλεγχοι βέλτιστης λύσης τέταρτου επιπέδου (μέρος γ).....	68
Εικόνα 52: Οθόνη πέμπτου επιπέδου	69
Εικόνα 53: Οθόνη οδηγιών πέμπτου επιπέδου.....	69
Εικόνα 54: Βέλτιστη λύση πέμπτου επιπέδου.....	71
Εικόνα 55: Οθόνη έκτου επιπέδου	73
Εικόνα 56: Οδηγίες έκτου επιπέδου	73
Εικόνα 57: Βέλτιστη λύση έκτου επιπέδου.....	75
Εικόνα 58: Οθόνη έβδομου επιπέδου.....	77
Εικόνα 59: Οδηγίες έβδομου επιπέδου	77
Εικόνα 60: Βέλτιστη λύση έβδομου επιπέδου	80
Εικόνα 61: Οθόνη όγδοου επιπέδου.....	81
Εικόνα 62: Οδηγίες όγδοου επιπέδου.....	82
Εικόνα 63: Βέλτιστη λύση όγδοου επιπέδου.....	83
Εικόνα 64: Οθόνη ένατου επιπέδου	86

Εικόνα 65: Οδηγίες ένατου επιπέδου	86
Εικόνα 66: Βέλτιστη λύση ένατου επιπέδου	87
Εικόνα 67: Οθόνη δέκατου επιπέδου	91
Εικόνα 68: Οδηγίες δέκατου επιπέδου	91
Εικόνα 69: Βέλτιστη λύση δέκατου επιπέδου	92
Εικόνα 70: Οθόνη εντέκατου επιπέδου	93
Εικόνα 71: Οδηγίες εντέκατου επιπέδου	94
Εικόνα 72: Βέλτιστη λύση εντέκατου επιπέδου	96
Εικόνα 73: Οθόνη δωδέκατου επιπέδου	97
Εικόνα 74: Τα αστέρια επέστρεψαν στην θέση του στο διάστημα	97
Εικόνα 75: Οθόνη αποτελεσμάτων παιχνιδιού	98
Εικόνα 76: Βεβαίωση ολοκλήρωσης του BlocklyScript	98
Εικόνα 77: Πλαίσιο σχεδίασης εκπαιδευτικού παιχνιδιού (Ibrahim & Jaafar, 2009)....	100
Εικόνα 78: Λογότυπο Phaser	105
Εικόνα 79: Δημογραφικά δεδομένα αξιολόγησης	134
Εικόνα 80: Η σχεδίαση του παιχνιδιού	136
Εικόνα 81: Αποτελέσματα αξιολόγησης της εμπειρίας των χρηστών	138
Εικόνα 82: Δημιουργία νέας ιστοσελίδας	149
Εικόνα 83: Διαχειριστής αρχείων ιστοσελίδας	150
Εικόνα 84: Κουμπί δημιουργίας βάσης δεδομένων	150

Κατάλογος Εικόνων Κώδικα

Κώδικας 1: Εγγραφή χρήστη στην βάση δεδομένων (register.php)	108
Κώδικας 2: Έλεγχοι δεδομένων εγγραφής χρήστη στην βάση δεδομένων (server.php)	109
Κώδικας 3: Έλεγχοι δεδομένων εισόδου χρήστη στην βάση δεδομένων (server.php)..	110
Κώδικας 4: Λήψη δεδομένων συνδεδεμένου χρήστη (server.php).....	112
Κώδικας 5: Αποθήκευση δεδομένων χρήστη στην τοπική μνήμη του περιηγητή (index.php)	113
Κώδικας 6: Αρχικοποίηση δεδομένων χρήστη (Level1/phaserScript.js).....	114
Κώδικας 7: Ανανέωση δεδομένων χρήστη κατά την δημιουργία του παιχνιδιού (Level1/phaserScript.js)	114
Κώδικας 8: Ενημέρωση δεδομένων προόδου χρήστη στην τοπική μνήμη και αποστολή στον server (level1Update.php) για ενημέρωση της βάσης (Level1/phaserScript.js)	115
Κώδικας 9: Ενημέρωση δεδομένων των πεδίων της βάσης (Level1/level1Update.php)	116
Κώδικας 10: Σύνδεση Phaser και Blockly στην αρχική σελίδα index.html του πρώτου επιπέδου.....	117
Κώδικας 11: Σύνδεση αρχείων blocklyScript.js και phaserScript.js με το αρχείο index.html του πρώτου επιπέδου.....	118
Κώδικας 12: Περιοχή παιχνιδιού και συντάκτη Blockly πρώτου επιπέδου (index.html)	118
Κώδικας 13: Αρχικοποίηση διαθέσιμων μπλοκ εντολών της εργαλειοθήκης του συντάκτη Blockly του πρώτου επιπέδου (index.html).....	118
Κώδικας 14: Ορισμός μπλοκ πρώτου επιπέδου (block_defs.js)	119
Κώδικας 15: Κώδικας του μπλοκ move_left (blocks_stub.js)	120
Κώδικας 16: API του μπλοκ move_left (blocks_stub.js).....	120
Κώδικας 17: Ενσωμάτωση συντάκτη Blockly στην αρχική σελίδα (index.html) του πρώτου επιπέδου (blocklyScript.js).....	122
Κώδικας 18: Ακροατής αλλαγών της περιοχής εργασίας του συντάκτη Blockly (blocklyScript.js)	125
Κώδικας 19: Ρυθμίσεις οθόνης παιχνιδιού και σύνδεση με την σελίδα εντέκατου επιπέδου (Level11/phaserScript.js)	126

Κατάλογος Πινάκων

Πίνακας 1: Χαρακτηριστικά ενός παιχνιδιού.....	3
Πίνακας 2: Κατηγοριοποίηση εντολών Scratch.....	14
Πίνακας 3: Τμήματα του Blockly Developer Tools.....	22
Πίνακας 4: Προγραμματιστικές έννοιες παιχνιδιών.....	34
Πίνακας 5: Χαρακτηριστικά παιχνιδιών	35
Πίνακας 6: Στόχοι, έννοιες και μπλοκ πρώτου επιπέδου	47
Πίνακας 7: Έλεγχοι περιοχής Program Output πρώτου επιπέδου.....	50
Πίνακας 8: Στόχοι, έννοιες και μπλοκ δεύτερου επιπέδου	54
Πίνακας 9: Έλεγχοι περιοχής Program Output δεύτερου επιπέδου	55
Πίνακας 10: Στόχοι, έννοιες και μπλοκ τρίτου επιπέδου	58
Πίνακας 11: Έλεγχοι περιοχής Program Output τρίτου επιπέδου.....	60
Πίνακας 12: Στόχοι, έννοιες και μπλοκ τέταρτου επιπέδου.....	63
Πίνακας 13: Έλεγχοι περιοχής Program Output τέταρτου επιπέδου	64
Πίνακας 14: Στόχοι, έννοιες και μπλοκ πέμπτου επιπέδου.....	70
Πίνακας 15: Έλεγχοι περιοχής Program Output πέμπτου επιπέδου	71
Πίνακας 16: Στόχοι, έννοιες και μπλοκ έκτου επιπέδου	74
Πίνακας 17: Έλεγχοι περιοχής Program Output έκτου επιπέδου.....	76
Πίνακας 18: Στόχοι, έννοιες και μπλοκ έβδομου επιπέδου.....	78
Πίνακας 19: Έλεγχοι περιοχής Program Output έβδομου επιπέδου	80
Πίνακας 20: Στόχοι, έννοιες και μπλοκ όγδοου επιπέδου.....	83
Πίνακας 21: Έλεγχοι περιοχής Program Output έβδομου επιπέδου	85
Πίνακας 22: Στόχοι, έννοιες και μπλοκ ένατου επιπέδου	87
Πίνακας 23: Έλεγχοι περιοχής Program Output ένατου επιπέδου.....	89
Πίνακας 24: Στόχοι, έννοιες και μπλοκ δέκατου επιπέδου	92
Πίνακας 25: Έλεγχοι περιοχής Program Output δέκατου επιπέδου.....	93
Πίνακας 26: Στόχοι, έννοιες και μπλοκ δέκατου επιπέδου	94
Πίνακας 27: Έλεγχοι περιοχής Program Output εντέκατου επιπέδου	96
Πίνακας 28: Χαρακτηριστικά σχεδίασης του BlocklyScript (Game design).....	101
Πίνακας 29: Παιδαγωγικά θέματα του BlocklyScript (Pedagogy)	103
Πίνακας 30: Μοντελοποίηση μαθησιακού περιεχομένου του BlocklyScript (Learning Content Modeling)	104
Πίνακας 31: Στήλες βάσης δεδομένων για την πρόοδο του χρήστη στο παιχνίδι	111

Πίνακας 32: Σημαντικότερες καθολικές μεταβλητές στο αρχείο blocklyScript.js	123
Πίνακας 33: Βασικές συναρτήσεις στο αρχείο blocklyScript.js	124
Πίνακας 34: Σημαντικότερες καθολικές μεταβλητές στο αρχείο phaserScript.js.....	127
Πίνακας 35: Συναρτήσεις στο αρχείο Level11/phaserScript.js.....	128
Πίνακας 36: Ποια είναι η γνώμη σας για το παιχνίδι; Τι σας άρεσε σε αυτό;	141
Πίνακας 37: Ποια είναι η γνώμη σας για το παιχνίδι; Τι δεν σας άρεσε σε αυτό;.....	141

1 Εισαγωγή

1.1 Πρόβλημα – Σημαντικότητα του θέματος

Η ραγδαία ανάπτυξη της τεχνολογίας κατά τις τελευταίες δεκαετίες καθιστά ως απαραίτητη ανάγκη την εξοικείωση του ανθρώπου με τον υπολογιστή. Για τον εξής λόγο η προσοχή των επιστημόνων αλλά και των κυβερνήσεων πολλών κρατών έχει στραφεί στην καλλιέργεια και στην ανάπτυξη της αλγοριθμικής και υπολογιστικής σκέψης των μαθητών της πρωτοβάθμιας και δευτεροβάθμιας εκπαίδευσης.

Για την επίτευξη αυτού το στόχου, έχουν αναπτυχθεί αρκετά διαδικτυακά παιχνίδια σοβαρού σκοπού. Ο όρος 'παιχνίδι σοβαρού σκοπού' συναντάται συχνά και ως 'edutainment' που είναι μια συνένωση των λέξεων 'education' (εκπαίδευση) και 'entertainment' (ψυχαγωγία) και αναφέρεται στα παιχνίδια που ως κύριο στόχο έχουν την ανάπτυξη των δεξιοτήτων και των γνώσεων του χρήστη πάνω σε έναν συγκεκριμένο επιστημονικό κλάδο.

Η προσέγγιση των ερευνητών για την ανάπτυξη παιχνιδιών σοβαρού σκοπού προήλθε έπειτα από πιλοτικές μελέτες και εμπειρικές έρευνες πάνω στην απόδοση των παιχνιδιών σοβαρού σκοπού στον τομέα της εκπαίδευσης. Σύμφωνα με την De Gloria et al (2014), η οποία συγκέντρωσε και μελέτησε έρευνες που αφορούσαν την επίδραση των παιχνιδιών σοβαρού σκοπού στην εκπαίδευση, απέδειξε ότι σε παιχνίδια με ελκυστικό περιβάλλον και περιεχόμενο οι μαθητές μπορούν να αποκτήσουν με αποδοτικό και αποτελεσματικό τρόπο νέες δεξιότητες και γνώσεις.

1.2 Σκοπός – Στόχοι

Σκοπός αυτής της εργασίας είναι η σχεδίαση και η ανάπτυξη ενός διαδικτυακού παιχνιδιού σοβαρού σκοπού για την καλλιέργεια της Υπολογιστικής Σκέψης σε μικρούς μαθητές με τη βοήθεια της Blockly. Η Blockly είναι μια βιβλιοθήκη της JavaScript ανεπτυγμένη από την Google και προσθέτει στις εφαρμογές έναν συντάκτη οπτικού προγραμματισμού στον οποίο παρουσιάζεται ο κώδικας ως αλληλοσυνδεδεμένα μπλοκ.

1.3 Συνεισφορά

Με τη σχεδίαση και την ανάπτυξη του σοβαρού παιχνιδιού 'BlocklyScirpt' παρέχεται η δυνατότητα σε μικρούς μαθητές να έρθουν σε επαφή και να εξοικειωθούν με τον επιστημονικό τομέα της πληροφορικής με τρόπο διασκεδαστικό και ευχάριστο.

Παίζοντας το παιχνίδι ο μαθητής θα μπορεί να συνδυάσει την διασκέδαση με την μάθηση ξεφεύγοντας με αυτόν τον τρόπο τους μονότονους ρυθμούς της παραδοσιακής διδασκαλίας.

Επιπλέον στην συγκεκριμένη εργασία παρουσιάζεται μια συγκριτική μελέτη μεταξύ παιχνιδιών σοβαρού σκοπού που είναι ελεύθερα διαθέσιμα στο διαδίκτυο με στόχο να δείξει τα δυνατά και αδύναμα σημεία τους.

Τέλος οι αναφορές για τις τεχνολογίες που χρησιμοποιήθηκαν για την ανάπτυξη του παιχνιδιού, καθώς και η αναλυτική περιγραφή του μοντέλου σχεδίασης του παιχνιδιού, δίνουν χρήσιμες πληροφορίες στον αναγνώστη που επιθυμεί να αναπτύξει το δικό του παιχνίδι σοβαρού σκοπού.

1.4 Διάρθρωση της μελέτης

Στο πρώτο κεφάλαιο γίνεται μια σύντομη εισαγωγή για την σημαντικότητα του θέματος της εργασίας, ο σκοπός της και η συνεισφορά της στους αναγνώστες.

Στο δεύτερο κεφάλαιο γίνεται η βιβλιογραφική επισκόπηση που αφορά τα παιχνίδια σοβαρού σκοπού, την υπολογιστική σκέψη, τις γλώσσες οπτικού προγραμματισμού και γίνεται μια συγκριτική ανάλυση για παιχνίδια σοβαρού σκοπού που συμβάλουν στην ανάπτυξη της υπολογιστικής σκέψης.

Στο τρίτο κεφάλαιο περιγράφονται τα επίπεδα του BlocklyScript.

Στο τέταρτο κεφάλαιο αναλύονται τα πλαίσια σχεδίασης και ανάπτυξης του παιχνιδιού και στην συνέχεια η αρχιτεκτονική του.

Στο πέμπτο κεφάλαιο παρέχεται η αξιολόγηση του παιχνιδιού από εκπαιδευτικούς πληροφορικής της πρωτοβάθμιας και δευτεροβάθμιας εκπαίδευσης.

Στο έκτο κεφάλαιο καταγράφονται τα συμπεράσματα για το παιχνίδι και προτείνονται μελλοντικές επεκτάσεις πάνω σε αυτό.

Στα τέλος παρέχονται η βιβλιογραφία και ένα παράρτημα που αφορά την διαδικασία δημοσίευσης της ιστοσελίδας στο διαδίκτυο.

2 Βιβλιογραφική Επισκόπηση – Θεωρητικό Υπόβαθρο

2.1 Πώς ορίζεται ένα παιχνίδι;

Τα παιχνίδια είναι σημαντικά στη ζωή του ανθρώπου και ιδιαίτερα για τα παιδιά. Κάθε άνθρωπος έχει παίξει τουλάχιστον μια φορά στη ζωή του και γνωρίζει τα ευχάριστα συναισθήματα που μπορεί να του προσφέρει το παιχνίδι. Η έννοια του παιχνιδιού υπήρχε από τα αρχαία χρόνια και κατά την πάροδο του χρόνου κάποια παιχνίδια χάθηκαν, ορισμένα παιχνίδια διασώθηκαν και πολλά παιχνίδια εφευρέθηκαν αξιοποιώντας τις τεχνολογίες της αντίστοιχης εποχής. Σήμερα, στην εποχή του 21ου αιώνα, όπου η τεχνολογία γνωρίζει ραγδαία ανάπτυξη και συγκεκριμένα στον επιστημονικό τομέα της πληροφορικής το παιχνίδι και η διασκέδαση έχει συνδυαστεί σε μεγάλο βαθμό με τα βιντεοπαιχνίδια σε κινητές συσκευές, σε ηλεκτρονικές συσκευές και γενικότερα σε μηχανές παιχνιδιών (π.χ. κονσόλες).

Τι θεωρείται όμως παιχνίδι; Κατά τους αιώνες έχουν δοθεί αρκετοί ορισμοί για την έννοια 'παιχνίδι' και συμφωνούν στο ότι είναι ένας τρόπος διασκέδασης και ψυχαγωγίας. Σύμφωνα με τον Huizinga (1949) η λέξη 'παίζω', που συνδέεται άμεσα με την λέξη 'παιχνίδι', μπορεί να οριστεί εφόσον πληροί τα χαρακτηριστικά του Πίνακα 1.

Πίνακας 1: Χαρακτηριστικά ενός παιχνιδιού

Χαρακτηριστικά των όρων 'παίζω' και 'παιχνίδι	
Είναι μια εθελοντική δραστηριότητα	Το να παίζεις για να διατάξεις δεν είναι πλέον παιχνίδι. Το παιχνίδι είναι μια ελευθερία που ξεφεύγει από την πορεία την φυσικής διαδικασίας.
Δεν είναι μια "συνηθισμένη" ή "πραγματική" ζωή	Είναι ένας τρόπος να ξεφεύγεις από την "πραγματική" ζωή, δημιουργώντας μια σφαίρα ή έναν κόσμο δραστηριότητας που έχει τη δική της/του διάθεση.
Διαχωρίζεται τοπικά και χρονικά από την "συνηθισμένη ζωή"	Ένα παιχνίδι εξελίσσεται κατά την διάρκεια ενός συγκεκριμένου χρονικού διαστήματος και χώρου

Ένα παιχνίδι δημιουργεί κανόνες, είναι ένας κανόνας	Σε ένα παιχνίδι που δεν υπάρχουν κανόνες ή οι παίκτες ξεφεύγουν από αυτούς τότε καταστρέφεται το παιχνίδι και η διασκέδαση που εκείνο προσφέρει.
Απορροφά την προσοχή του παίκτη	Κατά τη διάρκεια ενός παιχνιδιού ο παίκτης εφιστά την προσοχή του στην δραστηριότητα του παιχνιδιού.
Η επιθυμία για τη νίκη	Είναι η επιθυμία να υπερτερήσει των άλλων παικτών, έτσι ώστε να κατακτήσει την πρώτη θέση και να εκτιμηθεί η προσπάθεια του από τους υπόλοιπους

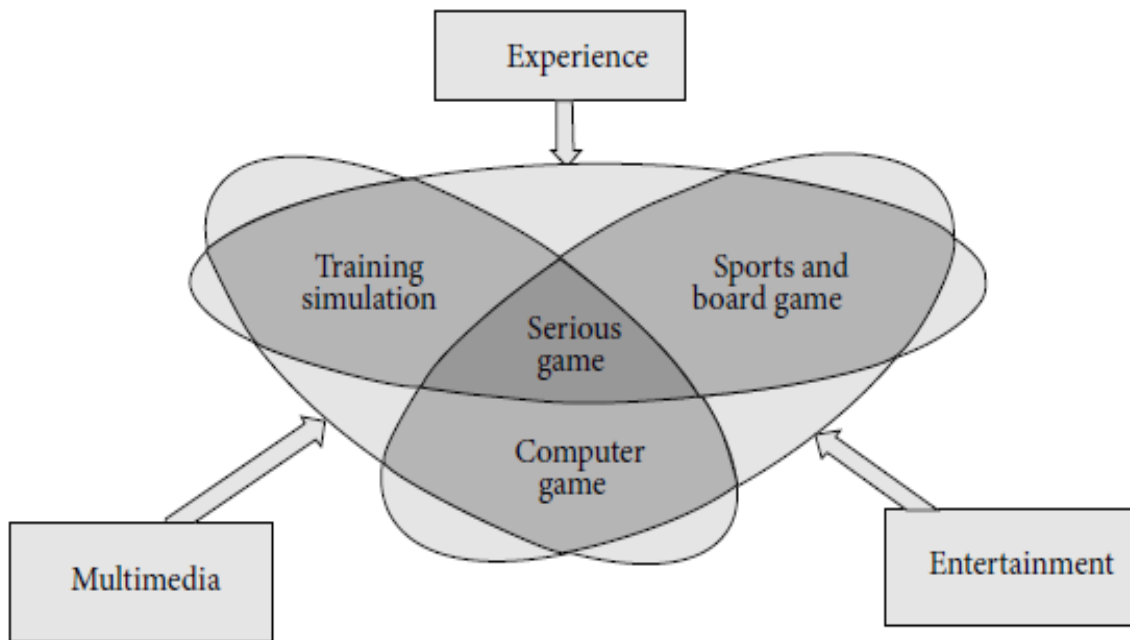
Με βάση τον Πίνακα 1, αυτά πρέπει να είναι τα κύρια χαρακτηριστικά ενός παιχνιδιού για να μπορεί να θεωρείται παιχνίδι. Βέβαια υπάρχουν και άλλα χαρακτηριστικά που εξαρτώνται κυρίως από τον τύπο του παιχνιδιού, όπως για παράδειγμα αν το παιχνίδι χρειάζεται έναν ή περισσότερους παίκτες για να ξεκινήσει. Στην δεύτερη περίπτωση προβάλλεται η συνεργασία ή/και ο ανταγωνισμός μεταξύ των παικτών. Επομένως ένα παιχνίδι ορίζεται εφόσον πληρεί τα παραπάνω κύρια χαρακτηριστικά, χωρίς ωστόσο να περιορίζεται η προσθήκη νέων χαρακτηριστικών στους διάφορους τύπους παιχνιδιών.

2.2 Πώς ορίζεται ένα παιχνίδι "σοβαρού σκοπού";

Ο όρος παιχνίδι "σοβαρού σκοπού" έγινε γνωστός στο ευρύ κοινό από τον Clark C.Abt με την δημοσίευση του βιβλίου του "Serious Games" το 1970. Στο βιβλίου του αναφέρει: *"Τα παιχνίδια σοβαρού σκοπού έχουν ένα ρητό και προσεκτικά σχεδιασμένο εκπαιδευτικό σκοπό και δεν προορίζονται κυρίως για διασκέδαση. Αυτό δεν σημαίνει ότι τα παιχνίδια σοβαρού σκοπού δεν είναι, ή δεν πρέπει να είναι διασκεδαστικά"*. (Abt, 1970, σελ.9)

Από το 1970 μέχρι σήμερα έχουν δοθεί αρκετοί ορισμοί για την έννοια "παιχνίδι σοβαρού σκοπού" και η αμφιβολία έγκειται στο εάν τα παιχνίδια αυτού του τύπου θα πρέπει ή όχι να περιέχουν το στοιχείο της διασκέδασης. Σε μια σύγχρονη μελέτη που έγινε για την επιρροή των παιχνιδιών "σοβαρού σκοπού" στον τομέα της εκπαίδευσης υποστηρίζεται ότι τα παιχνίδια αυτού του τύπου θα πρέπει να συνδυάζουν τα στοιχεία

της διασκέδασης και της μάθησης, διότι τα καθιστούν πιο ελκυστικά. Συγκεκριμένα το πόρισμα της έρευνας παρουσιάζονται στην Εικόνα 1.



Εικόνα 1: Ορισμός των παιχνιδιών σοβαρού σκοπού (Laamarti, Eid & El Saddik, 2014)

Όπως δείχνει η Εικόνα 1 ένα παιχνίδι πρέπει να περιέχει τρία βασικά δομικά στοιχεία:

- Εμπειρία
- Πολυμέσα
- Ψυχαγωγία

- Εμπειρία. Ο όρος "σοβαρό" στα παιχνίδια σοβαρού σκοπού αποδίδεται στον στόχο του παιχνιδιού να μεταδώσει ένα μήνυμα ή κάποιο ερέθισμα στο παίκτη. Ανάλογα με το είδος του παιχνιδιού ο παίκτης θα έχει τη δυνατότητα να αναπτύξει τις εγκυκλοπαιδικές του γνώσεις ή τις ικανότητες του. Ο στόχος επομένως του παιχνιδιού βασίζεται στην τεχνογνωσία και στην εμπειρία του παίκτη από παρόμοια πεδία όπως για παράδειγμα η εκπαίδευση, η υγεία, οι τέχνες κ.τ.λ.

-Πολυμέσα. Τα παιχνίδια σοβαρού σκοπού συνδυάζουν αρκετές τεχνικές για κερδίσουν την προσοχή του παίκτη. Για παράδειγμα κάποια παιχνίδια συνδυάζουν ήχους, στατικές εικόνες, κινούμενες εικόνες, βίντεο, κείμενο, χρώματα κ.τ.λ.

- Ψυχαγωγία. Είναι αρκετά σημαντικό και ιδιαίτερα για μαθητές μικρής ηλικίας να συνδυάσουν τη μάθηση με την διασκέδαση. Ένα παιχνίδι που δεν είναι μονότονο και παρέχει ένα ενδιαφέρον σενάριο κεντρίζει το ενδιαφέρον του μαθητή. Με αυτόν τον τρόπο ο μαθητής δεν συνδυάζει τη μάθηση με την υποχρέωση και την κούραση, αλλά με την ευχαρίστηση.

Επίσης στον Εικόνα 1 παρέχονται και οι ορολογίες παιχνίδι υπολογιστή, προσομοίωση εκπαίδευσης, αθλήματα και επιτραπέζιο παιχνίδι με σκοπό να εντοπιστούν οι διαφορές τους με ένα παιχνίδι σοβαρού σκοπού. Οι τρεις αυτές έννοιες περιέχουν μόνο δύο από τα τρία δομικά στοιχεία που υπάρχουν σε ένα παιχνίδι σοβαρού σκοπού και κάθε μια από αυτές συνδυάζει τα δομικά στοιχεία με διαφορετικό τρόπο. Έτσι ένα παιχνίδι και πιο συγκεκριμένα ένα ηλεκτρονικό παιχνίδι μπορεί να θεωρηθεί σοβαρού σκοπού εάν ενσωματώνει τις παραπάνω έννοιες.

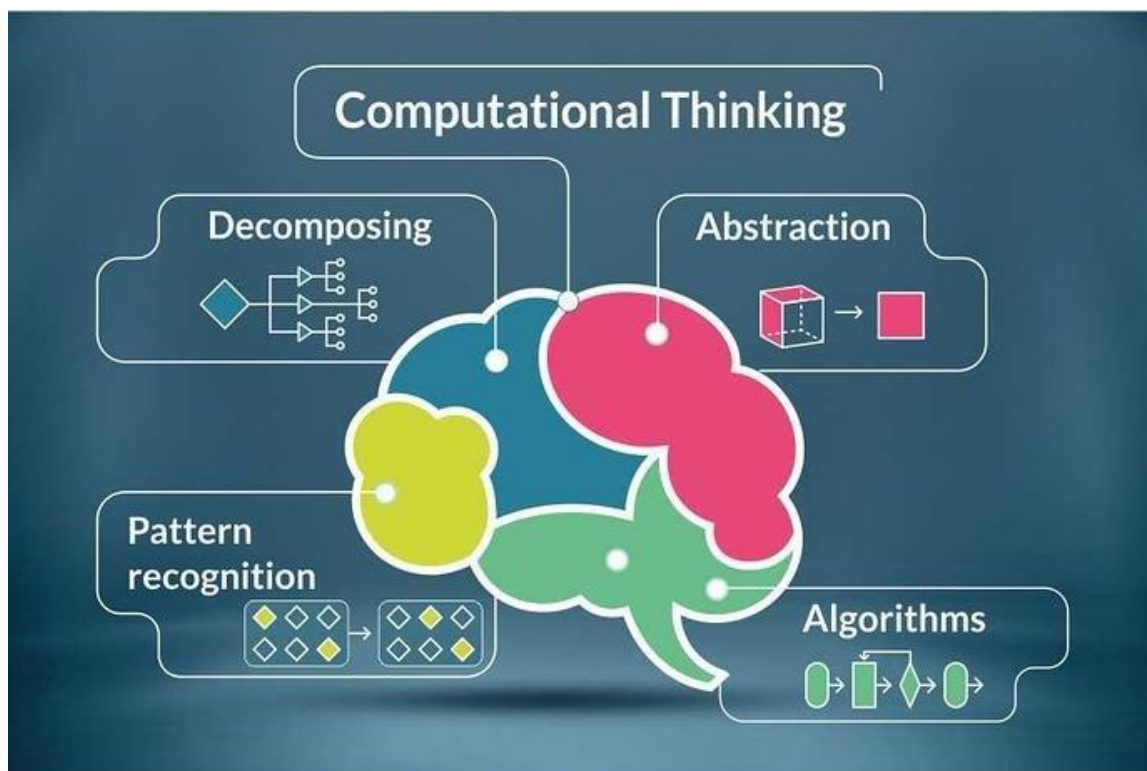
2.3 Η έννοια της Υπολογιστικής Σκέψης

Ο όρος "Υπολογιστική Σκέψη" έγινε δημοφιλής με την εμφάνιση των ηλεκτρονικών υπολογιστών και σχετίζεται με τον τρόπο που ένας υπολογιστής "σκέφτεται" για την επίλυση του εκάστοτε προβλήματος. Με την εξέλιξη του επιστημονικού τομέα της Πληροφορικής και γενικότερα της τεχνολογίας η "Υπολογιστική Σκέψη" αποκτάει ολοένα και περισσότερη βαρύτητα για τα προσόντα που πρέπει να διαθέτει και να αναπτύσσει ένας άνθρωπος.

Ωστόσο, πέρα από τους δύο αυτούς τομείς η Υπολογιστική Σκέψη είναι απαραίτητη και σε άλλους τομείς που αποτελούν ένα κομμάτι της βασικής εκπαίδευσης του κάθε ανθρώπου όπως είναι για παράδειγμα τα Μαθηματικά, η Φυσική και η Χημεία. *"Το να σκέφτεσαι όπως ένας επιστήμονας της Επιστήμης των υπολογιστών σημαίνει κάτι περισσότερο από το να μπορείς απλά να προγραμματίζεις έναν υπολογιστή. Απαιτεί σκέψη σε πολλαπλά επίπεδα αφαίρεσης."*(Wing, 2006). Συμπεριλαμβάνοντας και τους επιστημονικούς τομείς πέρα της Πληροφορικής η "Υπολογιστική Σκέψη" επαναπροσδιορίζεται ως ο τρόπος σκέψης που πρέπει να κάνει κάποιος για να κατανοήσει ένα περίπλοκο πρόβλημα και να το σπάσει σε μικρότερα πιο απλά

προβλήματα με σκοπό να βρει τη λύση. Τα μικρότερα αυτά προβλήματα θα πρέπει να είναι κατανοητά από τον άνθρωπο ή/και τον υπολογιστή ανάλογα την περίπτωση.

Οι ερευνητές στην προσπάθεια τους να προσδιορίσουν τον όρο της "Υπολογιστικής Σκέψης" διατύπωσαν τέσσερις βασικούς πυλώνες που την χαρακτηρίζουν (David & Kenny(2014), Harimuti et al.(2019), Krauss et al.(2016)).



Εικόνα 2: Υπολογιστική Σκέψη (Harimuti et al.(2019))

- ❖ Αποσύνθεση (Decomposing): Ένα πρόβλημα μπορεί να είναι σύνθετο και περίπλοκο και η λύση του να μην είναι εμφανής εκ πρώτης όψεως. Για τον λόγο αυτό πρέπει το αρχικό πρόβλημα να σπάσει σε μικρότερα κομμάτια που θα είναι πιο εύκολα διαχειρίσιμα και πιο εύκολα να απαντηθούν. Από τα νέα αυτά κομμάτια εάν κάποιο εξακολουθεί να είναι περίπλοκο τότε θα πρέπει να διασπαστεί πάλι σε μικρότερα κομμάτια κ.ο.κ.
- ❖ Αφαίρεση (Abstraction): Η διαδικασία που θα πρέπει κάποιος να ακολουθήσει για να βρει τη λύση θα πρέπει να μην περιλαμβάνει λεπτομέρειες, αλλά να

περιέχει όσο το δυνατόν γενικότερες εντολές. Με τον τρόπο αυτό θα μπορεί να επιλύσει και προβλήματα που μοιάζουν στο αρχικό του πρόβλημα.

- ❖ **Αναγνώριση προτύπου (Patter recognition):** Ένα πρόβλημα ενώ αρχικά ίσως να φαίνεται μεγάλο και δύσκολο, αν μελετηθεί προσεκτικά τότε μπορεί σε πολλά σημεία οι λύσεις του να μοιάζουν. Για τον λόγο αυτό η αναγνώριση του προτύπου επηρεάζεται άμεσα από τις αφαιρετικές εντολές που θα δώσει ο λύτης του προβλήματος. Έτσι δεν θα χρειάζεται να υπολογίζει επανειλημμένα παρόμοια κομμάτια του προβλήματος.
- ❖ **Αλγόριθμοι (Algorithms):** Ο λύτης ενός προβλήματος θα πρέπει να σκέφτεται αλγοριθμικά. Αυτό σημαίνει ότι θα πρέπει να δημιουργεί μια σειρά πεπερασμένων, καλά ορισμένων βημάτων που θα τον οδηγήσουν στην λύση του συγκεκριμένου προβλήματος. Παρόμοια λογική με του αλγορίθμους ακολουθούν και οι συνταγές μαγειρικής στις οποίες ο μάγειρας πρέπει να ακολουθήσει τα βήματα της συνταγής για φτάσει στο επιθυμητό αποτέλεσμα. Η διαφορά έγκειται στο ότι πολλά βήματα από τις συνταγές δεν είναι καλά ορισμένα σε αντίθεση με τα βήματα του αλγορίθμου.

2.4 Οπτικός Προγραμματισμός

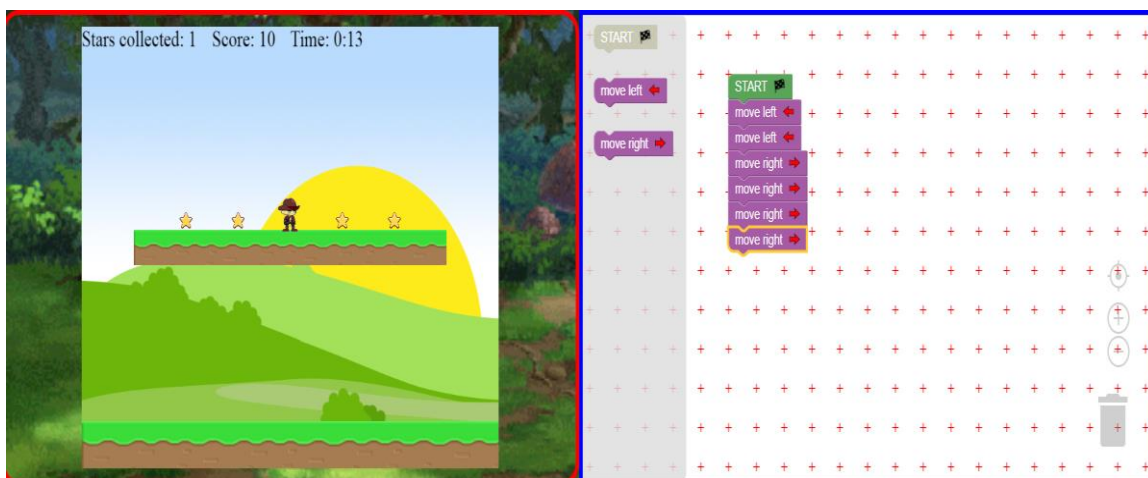
2.4.1 Οπτικός προγραμματισμός στην εκπαίδευση

Οι γλώσσες προγραμματισμού τα τελευταία χρόνια εμφανίζουν κάποιες αλλαγές ως προς τη μορφή τους. Με βάση τα σημερινά δεδομένα δύο είναι οι κύριες μορφές που μπορεί να έχει μια γλώσσα προγραμματισμού. Η πρώτη κατηγορία είναι οι κλασσικές γλώσσες προγραμματισμού με μορφή κειμένου. Για να μπορέσει ένας προγραμματιστής να δημιουργήσει ένα πρόγραμμα στον υπολογιστή θα πρέπει να γνωρίζει μια γλώσσα προγραμματισμού και με τη γλώσσα αυτή να γράψει σε μορφή κειμένου ένα σύνολο από γραμμές τις οποίες θα μεταφράσει ο υπολογιστής και θα τις εκτελέσει. Στην Εικόνα 3 παρουσιάζεται μια γλώσσα προγραμματισμού με μορφή κειμένου.



Εικόνα 3: Προγραμματισμός σε μορφή κειμένου

Η δεύτερη κατηγορία είναι ο οπτικός προγραμματισμός. Η κατηγορία αυτή περιλαμβάνει τις γλώσσες προγραμματισμούς που συνδυάζουν τις εντολές της με γραφικά. Έτσι ο προγραμματιστής θα μπορεί με πιο εύκολο τρόπο να καταλάβει τη λειτουργικότητα των εντολών, σε αντίθεση με την πρώτη κατηγορία όπου ο χρήστης βλέποντας μόνο τα ονόματα των εντολών αρκετές φορές δεν κατανοεί τη λειτουργικότητα τους ή τις ξεχνάει και πρέπει να ανατρέξει στα τεκμήρια της αντίστοιχης γλώσσας.



Εικόνα 4: Οπτικός προγραμματισμός με μπλοκ

Η επιρροή που μπορεί να ασκήσει κάθε μια από τις δύο κατηγορίες των γλωσσών προγραμματισμού σε μαθητές μικρής ηλικίας είναι διαφορετική. Σε μια έρευνα που έγινε με στόχο την σύγκριση των δύο παραπάνω κατηγοριών και την επιρροή τους σε άτομα που έρχονται για πρώτη φορά σε επαφή με ένα γνωστικό αντικείμενο τα αποτελέσματα έδειξαν ότι ο οπτικός προγραμματισμός είναι καλύτερη προσέγγιση (Saito et al. (2017)), Αρχικά οι γλώσσες κειμένου τείνουν να ζητούν την είσοδο μεγαλύτερου όγκου δεδομένων από τον χρήστη, γεγονός που επιδράει αρνητικά στη συμπεριφορά του χρήστη. Επίσης η έρευνα έδειξε ότι ο οπτικός προγραμματισμός επιδρά πιο θετικά στην συμπεριφορά των χρηστών σε σχέση με τον προγραμματισμό με κείμενο. Συγκεκριμένα η μεταβολή στην συμπεριφορά που παρατηρήθηκε από τους χρήστες πριν διδαχτούν τις έννοιες με μια από τις δύο γλώσσες χωρίστηκε σε πέντε άξονες. Το ενδιαφέρον των χρηστών αυξήθηκε περισσότερο μαθαίνοντας τις έννοιες με τον οπτικό προγραμματισμό και η εντύπωση τους σχετικά με το πόσο δύσκολο είναι κάποιος αρχάριος να μάθει προγραμματισμό βελτιώθηκε.

Επιπλέον στους χρήστες που έμαθαν τις νέες έννοιες με τον οπτικό προγραμματισμό βελτιώθηκε η γνώμη τους σχετικά με την σημαντικότητα του προγραμματισμού στην ζωή τους. Από την άλλη οι χρήστες που παρακολούθησαν την διάλεξη με την βοήθεια του κειμενογράφου αποθαρρύνθηκαν με αποτέλεσμα να υποστηρίξουν λιγότερα άτομα την σημαντικότητα του προγραμματισμού στην ζωή τους . Επιπροσθέτως η ομάδα του οπτικού προγραμματισμού διασκέδασε μαθαίνοντας να προγραμματίζει με γραφικά, ενώ υπήρξε μια μείωση στην ψυχαγωγία της άλλης ομάδας. Τέλος η ομάδα του οπτικού προγραμματισμού έδειξε μεγαλύτερο ενδιαφέρον στην εκμάθηση μιας γλώσσας προγραμματισμού σε σύγκριση με την δεύτερη ομάδα.

2.4.2 Γλώσσες Οπτικού Προγραμματισμού στην εκπαίδευση

Οι γλώσσες του οπτικού προγραμματισμού έχουν γίνει αρκετά δημοφιλείς στην ανάπτυξη εφαρμογών. Οι εφαρμογές που αναπτύσσονται δεν περιορίζονται μόνο στην εκπαίδευση αλλά και σε άλλους τομείς. Για παράδειγμα ένας γνωστός τομέας που έχει επηρεαστεί σε μεγάλο βαθμό από τις γλώσσες οπτικού προγραμματισμού είναι τα πολυμέσα. Στον τομέα αυτό έχουν αναπτυχθεί δεκάδες εφαρμογές για την επεξεργασία ήχου (π.χ. AudioMuich), για την επεξεργασία τρισδιάστατων γραφικών (π.χ. Blender), για την επεξεργασία βίντεο (π.χ. vvvv - a multipurpose toolkit) κ.α. Παρόμοιες

εφαρμογές έχουν αναπτυχθεί σε τομείς όπως η παραγωγή παιχνιδιών (π.χ. Unity), η προσομοίωση (π.χ. LabVIEW), η αυτοματοποίηση (π.χ. CiMPLE), η ανάλυση δεδομένων (π.χ. Alteryx Designer) κ.α.

Γενικά οι γλώσσες οπτικού προγραμματισμού ενσωματώνονται ολοένα και σε περισσότερους τομείς καθώς η τεχνολογία εξελίσσεται. Με τη συνεχή ανάπτυξη της τεχνολογίας οι έννοιες που εμφανίζονται είναι δύσκολες να κατανοηθούν από το ευρύ κοινό και προκαλούν σύγχυση. Για τον λόγο αυτό οι γλώσσες οπτικού προγραμματισμού μέσω εικόνων και γραφικών προσπαθούν να απλοποιήσουν αυτές τις έννοιες έτσι ώστε να είναι πιο εύκολα κατανοητές για εκείνους που δεν τις γνωρίζουν και έρχονται σε μια πρώτη επαφή με αυτές.

Στην ενότητα αυτή η προσοχή θα δοθεί στην παρουσίαση κάποιων γλωσσών οπτικού προγραμματισμού που στόχο έχουν την εξοικείωση των μικρών μαθητών με την Υπολογιστική Σκέψη.

2.4.2.1 Scratch

<https://scratch.mit.edu/>



Εικόνα 5: Το λογότυπο της γλώσσας Scratch

Μια από τις πιο δημοφιλείς γλώσσες οπτικού προγραμματισμού είναι η Scratch, ανεπτυγμένη από το MIT Media Lab. Η γλώσσα Scratch παρέχεται δωρεάν στο διαδίκτυο και στόχο έχει την ανάπτυξη της δημιουργικότητας, της μεθοδικότητας και συνεργατικού πνεύματος σε μαθητές ηλικίας από 8 έως 16 χρονών αλλά και οποιονδήποτε θέλει να ασχοληθεί μαζί της. Η γλώσσα Scratch, όπως υποστηρίζεται στην

ιστοσελίδα της, έχει μεταφραστεί σε περισσότερες από 40 γλώσσες παγκοσμίως και εκατομμύρια άνθρωποι τη χρησιμοποιούν στα σχολεία, στα σπίτια, στα μουσεία, στις βιβλιοθήκες και στα κοινοτικά κέντρα.

Με τη βοήθεια της γλώσσας Scratch οι μαθητές δημοτικού έως λυκείου έχουν τη δυνατότητα να αναπτύξουν τις γνώσεις τους σε διάφορους επιστημονικούς τομείς όπως είναι τα μαθηματικά, οι κοινωνικές επιστήμες, οι γλώσσες, η τέχνη και η επιστήμη των υπολογιστών. Επιπλέον οι εκπαιδευτικοί μπορούν μέσω της κοινότητας του Scratch να ανταλλάξουν απόψεις με άλλους εκπαιδευτικούς που μπορεί να βρίσκονται στην άλλη άκρη του κόσμου, να μοιραστούν τα έργα τους και να απαντούν σε ερωτήσεις μαθητών.

Σε μια έρευνα που έγινε από την ομάδα της Scratch σε συνεργασία με άλλους ερευνητές εντοπίστηκαν κάποιοι παράγοντες που αποτελούσαν εμπόδια στην εκπαίδευση μικρών μαθητών. Πριν την ανάπτυξη των γλωσσών οπτικού προγραμματισμού τα εμπόδια που εμφανίζονταν ήταν τα εξής (Resnick et al., (2009)):

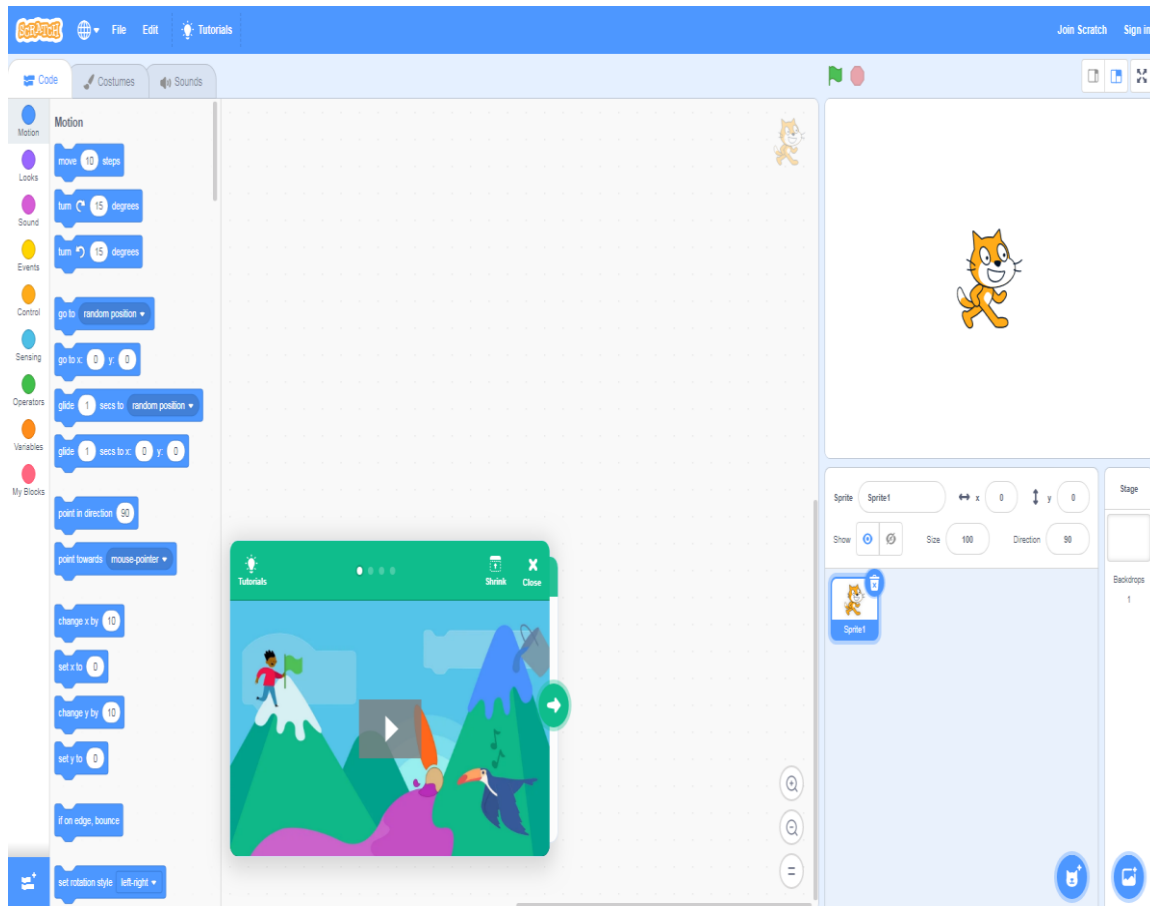
- Οι πρώιμες γλώσσες προγραμματισμού με κείμενο ήταν δύσκολο να χρησιμοποιηθούν και οι μαθητές μικρής ηλικίας ήταν δύσκολο να κατανοήσουν τη σύνταξη της γλώσσας
- Οι έννοιες του προγραμματισμού παρουσιάζονταν στους μαθητές με απλές δραστηριότητες, όπως η δημιουργία λιστών που περιέχει πρώτους αριθμούς και η σχεδίαση ευθειών γραμμών, που δεν κέρδιζαν το ενδιαφέρον τους.
- Οι προγραμματιστικές έννοιες παρουσιάζονταν σε περιβάλλοντα στα οποία κάνεις δεν μπορούσε να προσφέρει καθοδήγηση στους μαθητές όταν εμφανίζονταν σφάλματα και δεν μπορούσαν να ενθαρρύνουν τους μαθητές σε περίπτωση που τα αποτελέσματα ήταν σωστά.

Βασισμένη σε αυτές τις αδυναμίες η ομάδα σχεδίασης και ανάπτυξης του Scratch ανέπτυξε τη γλώσσα διατηρώντας τρεις βασικούς πυλώνες:

1. Πιο εύχρηστη: Η ομάδα σχεδίασης συνεργάστηκε με την εταιρία παιχνιδιών Lego και ενσωμάτωσε στη γλώσσα της την λογική των αλληλοσυνδεδεμένων μπλοκ.
2. Με περισσότερο νόημα: Τα έργα που μπορεί να αναπτύξει κανείς στην Scratch πρέπει να έχουν νόημα για τον δημιουργό τους. Μέσω ανάπτυξης ιστοριών, παιχνιδιών, κινούμενων σχεδίων και προσομοιώσεων το ενδιαφέρον ενός μικρού μαθητή παραμένει ακάθεκτο.

3. Πιο κοινωνική: Μια γλώσσα θα πρέπει να συνδέει όσους την μιλάνε. Για τον λόγο αυτό η Scratch επιτρέπει στον χρήστη να δημοσιεύει τα έργα του και να επικοινωνεί εύκολα με άλλους χρήστες της γλώσσας.

Με βάση αυτά η ομάδα της γλώσσας Scratch σχεδίασε την ιστοσελίδα που φαίνεται στην Εικόνα 6.



Εικόνα 6: Η γλώσσα Scratch

Στην Εικόνα 6 φαίνονται δύο βασικές περιοχές. Αριστερά είναι η περιοχή που ο χρήστης ενσωματώνει τον κώδικά με τη μορφή μπλοκ και καταλαμβάνει το μεγαλύτερο τμήμα της οθόνης. Δεξιά είναι η οθόνη παρουσίασης του αποτελέσματος που θα εμφανιστεί στον χρήστη μόλις τρέξει τις εντολές που τοποθέτησε στην περιοχή που βρίσκεται στο κέντρο της οθόνης. Επίσης στο κέντρο και προς τα κάτω της οθόνης βρίσκεται ένα βίντεο το οποίο μαθαίνει στον χρήστη τις βασικές λειτουργίες της γλώσσας Scratch.

Πιο συγκεκριμένα η αριστερή περιοχή απαρτίζεται από τις κατηγορίες κώδικας, σχεδίαση κουστουμιών και επεξεργασία ήχου. Οι κατηγορίες αυτές εναλλάσσονται ανάλογα με εκείνη που έχει επιλεγεί από τον χρήστη και καταλαμβάνουν όλη την αριστερή περιοχή της οθόνης. Οι δύο τελευταίες είναι σχετικά απλές για να κατανοηθούν από τον μαθητή. Η κατηγορία της σχεδίασης κουστουμιών αναλαμβάνει την επεξεργασία του χαρακτήρα ενώ η κατηγορία επεξεργασίας ήχου αναλαμβάνει τις ρυθμίσεις των αρχείων ήχου. Από την άλλη η κατηγορία του κώδικα που είναι και η πιο σημαντική απαιτεί περισσότερη προσοχή. Η ομάδα σχεδίασης της γλώσσας με σκοπό να διευκολύνουν τους μαθητές ομαδοποίησαν τις διαθέσιμες εντολές. Οι ομάδες εντολών παρουσιάζονται στον Πίνακα 2.

Πίνακας 2: Κατηγοριοποίηση εντολών Scratch

Κατηγορία	Λειτουργικότητα
Κίνηση	Αναλαμβάνει την κίνηση των χαρακτήρων, αλλάζει τις μοίρες και τη θέση τους στον χώρο.
Εμφάνιση	Αναλαμβάνει την εμφάνιση μηνυμάτων, την αλλαγή κουστουμιών και του μεγέθους του χαρακτήρα.
Ήχος	Ελέγχει πότε θα ξεκινήσει ή θα σταματήσει ο ήχος και ρυθμίζει την ένταση
Γεγονότα	Ακούει την πιθανή είσοδο του χρήστη από κάποιο περιφερικό μέσο π.χ. το πληκτρολόγιο και μόλις ενεργοποιηθεί ξεκινάει μια ενέργεια.
Έλεγχος	Περιέχει έννοιες του προγραμματισμού που βοηθάν στου ελέγχους όπως για παράδειγμα ο βρόχος "περίμενε μέχρι"
Αίσθηση	Ο χαρακτήρας μπορεί να επικοινωνήσει με το περιβάλλον στο οποίο βρίσκεται
Τελεστές	Υπάρχουν οι μαθηματικοί και οι λογικοί τελεστές

Μεταβλητές	Επιτρέπει τη δημιουργία μεταβλητών και λιστών
Δικά μου μπλοκ	Ο χρήστης μπορεί να δημιουργήσει δικά του μπλοκ προσαρμοσμένα στις ανάγκες του προβλήματος του.

Τέλος η ιστοσελίδα περιέχει πλήθος από παραδείγματα τα οποία είτε περιέχουν μικρά βίντεο, είτε απλά βήματα με εντολές, είτε τον συνδυασμό και των δύο. Έτσι ο χρήστης χωρίς να έχει προαπαιτούμενες γνώσεις μπορεί να μάθει εύκολα και ευχάριστα τη γλώσσα Scratch.

2.4.2.2 Kodu Game Lab

<https://www.kodugamelab.com/>



Εικόνα 7: Λογότυπο της γλώσσας Kodu

Μια δεύτερη γλώσσα οπτικού προγραμματισμού είναι η Kodu Game Lab (KGL), ανεπτυγμένη από την ομάδα Microsoft's FUSE Labs. Η γλώσσα οπτικού προγραμματισμού Kodu προορίζεται για μαθητές μικρής ηλικίας και στόχο έχει να τους διδάξει τη δημιουργικότητα, την επίλυση προβλημάτων, την αφήγηση ιστοριών και τον προγραμματισμό. Βέβαια η γλώσσα αυτή δεν αποθαρρύνει και οποιονδήποτε ενήλικα θέλει να μάθει και να αναπτύξει τις γνώσεις του διασκεδάζοντας παράλληλα. Η κύρια λειτουργικότητα της KGL είναι η ανάπτυξη τρισδιάστατων παιχνιδιών μέσα από ένα λιτό και εύχρηστο για τον χρήστη περιβάλλον.



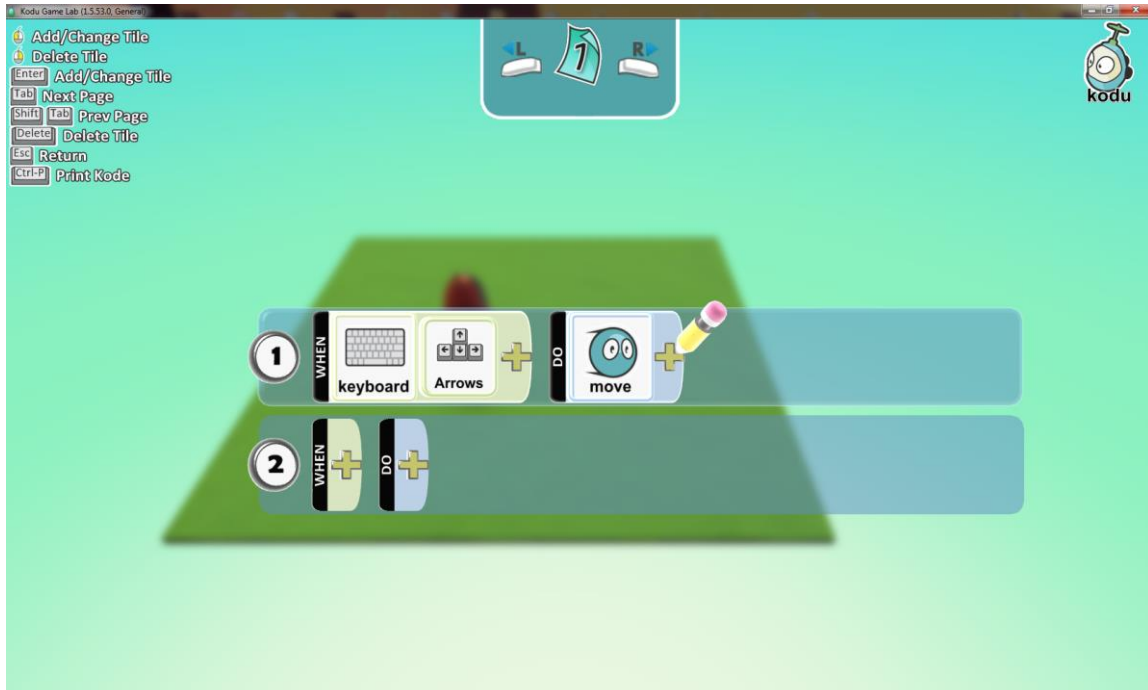
Εικόνα 8: Περιβάλλον σχεδίασης της Kodu

Όπως φαίνεται και στην Εικόνα 8 το περιβάλλον σχεδίασης των παιχνιδιών είναι λιτό. Στο κάτω μέρος και κεντρικά της οθόνης είναι το μενού με τα διαθέσιμα εργαλεία που διαθέτει ο χρήστης. Ξεκινώντας από αριστερά τα τρία πρώτα εικονίδια είναι υπεύθυνα για την επιστροφή στο αρχικό μενού, την έναρξη του παιχνιδιού και την μετακίνηση της κάμερας κατά την σχεδίαση αντίστοιχα. Η τέταρτη εικόνα με το ρομπότ είναι υπεύθυνο για δημιουργία χαρακτήρων ή αντικειμένων στο παιχνίδι. Πατώντας στην εικόνα αυτή και έπειτα κάνοντας αριστερό κλικ στην πράσινη περιοχή (έδαφος) εμφανίζεται η παρακάτω εικόνα:



Εικόνα 9: Δημιουργία χαρακτήρα ή αντικειμένου στη Kodu

Ο βασικός χαρακτήρας δεν είναι άλλος από το ρομποτάκι Kodu, το οποίο ο χρήστης μπορεί να προγραμματίσει για να εκτελεί τις εντολές του. Πέρα από το μικρό ρομπότ ο χρήστης μπορεί να δημιουργήσει σωλίνες, δέντρα, πέτρες, μήλα και διάφορα άλλα αντικείμενα.



Εικόνα 10: Προγραμματισμός χαρακτήρων στην Kodu

Ο προγραμματισμός των χαρακτήρων είναι μια αρκετά εύκολη διαδικασία και απαρτίζεται από δύο καταστάσεις, την "ΟΤΑΝ" (WHEN) και την "ΚΑΝΕ" (DO). Στο παράδειγμα τις Εικόνας 10 φαίνεται ότι ο χαρακτήρας "ΟΤΑΝ" ακούσει τον χρήστη να πατάει ένα από τα βελάκια στο πληκτρολόγιο τότε ("ΚΑΝΕ") θα κινηθεί ανάλογα στον χώρο. Πέρα από αυτήν την απλή εντολή ο χρήστης μπορεί να προσθέσει και άλλες σε επόμενες γραμμές και να δημιουργήσει πιο πολύπλοκες δραστηριότητες. Επιπλέον η εφαρμογή Kodu Game Lab επιτρέπει στον χρήστη να ρυθμίσει λειτουργίες όπως η ταχύτητα του χαρακτήρα, το μέγεθος, την ταχύτητα περιστροφής κ.α.

Γυρνώντας πάλι στο μενού σχεδίασης της Εικόνας 8, οι επόμενες δύο εικόνες είναι υπεύθυνες για τη δημιουργία μονοπατιών (δρόμων) και προσθήκης νέου εδάφους. Εν συνεχεία οι επόμενες επιλογές δίνουν την επιλογή στον χρήστη να δημιουργήσει βουνά και ποτάμια και να τα επεξεργαστεί. Τέλος οι δύο επιλογές που απομένουν

αναλαμβάνουν τη διαγραφή χαρακτήρων από το παιχνίδι και την αλλαγή των ρυθμίσεων του κόσμου.

Πέρα από το λιτό περιβάλλον η Kodu Game Lab παρέχει αρκετά μαθήματα για την εξοικείωση ενός αρχάριου με το περιβάλλον. Τα μαθήματα αυτά ξεκινάνε παρουσιάζοντας απλές λειτουργίες της Kodu και συνεχίζουν εντάσσοντας ολοένα και περισσότερα χαρακτηριστικά στο παιχνίδι. Επιπλέον η KGL υποστηρίζει την δημοσίευση των έργων των χρηστών στην ιστοσελίδα της και επιτρέπει το κατέβασμα αυτών από άλλους. Με τον τρόπο αυτό ενισχύει την δημιουργικότητα των χρηστών και αναπτύσσει την κοινότητα της.

2.4.3 Ανάπτυξη γλωσσών οπτικού προγραμματισμού με το Blockly

<https://developers.google.com/blockly>



Εικόνα 11: Λογότυπο της βιβλιοθήκης Blockly

Όπως παρουσιάστηκε στην προηγούμενη ενότητα η Scratch είναι μια γλώσσα οπτικού προγραμματισμού για την εξοικείωση μικρών μαθητών με την Επιστήμη των Υπολογιστών. Για την ανάπτυξη της γλώσσας Scratch καθοριστικό ρόλο έπαιξε η βιβλιοθήκη Blockly της JavaScript, σχεδιασμένη από την Google. Η Blockly δίνει τη δυνατότητα στον προγραμματιστή εφαρμογών να εντάξει έναν συντάκτη στην εφαρμογή, ο οποίος παρουσιάζει τις προγραμματιστικές έννοιες με μορφή αλληλοσυνδεδεμένων μπλοκ. Ο συντάκτης έπειτα διαβάζοντας τα μπλοκ παράγει κώδικα συντακτικά σωστά δομημένο. Ο παραγόμενος κώδικας υποστηρίζεται σε γλώσσες όπως η JavaScript, η PHP, η Python, η Lua και η Dart.

Σύμφωνα με την επίσημη ιστοσελίδα της Blockly εκατοντάδες έργα παγκοσμίως χρησιμοποιούν τη βιβλιοθήκη στην εκπαίδευση. Μερικά από τα έργα έχουν δημιουργηθεί από τους παρακάτω αναγνωρισμένους οργανισμούς:



Εικόνα 12: Οργανισμοί που χρησιμοποιούν την βιβλιοθήκη Blockly

Σε υψηλό επίπεδο ανάλυσης, τα βήματα για την δημιουργία μια εφαρμογής Blockly είναι τρία:

1. Η ενσωμάτωση του συντάκτη της βιβλιοθήκης. Ο συντάκτης Blockly είναι ένα απλό περιβάλλον και χωρίζεται σε δύο βασικά τμήματα. Το πρώτο τμήμα είναι η εργαλειοθήκη (toolbox) στην οποία αποθηκεύονται τα διαθέσιμα μπλοκ που έχει δημιουργήσει ο προγραμματιστής. Το δεύτερο τμήμα είναι η περιοχή εργασίας (workspace) στην οποία ο χρήστης μπορεί να σέρνει και να αφήνει (drag and drop) τα μπλοκ που βρίσκονται στην εργαλειοθήκη.
2. Δημιουργία μπλοκ. Εφόσον γίνει η ενσωμάτωση του συντάκτη στην εφαρμογή, ο προγραμματιστής θα πρέπει να δημιουργήσει τα δικά του μπλοκ ή να χρησιμοποιήσει κάποια έτοιμα μπλοκ που του διαθέτει η βιβλιοθήκη.
3. Η ανάπτυξη της υπόλοιπης εφαρμογής. Το Blockly αναλαμβάνει μόνο την παραγωγή κώδικα συντακτικά σωστά δομημένου. Δεν καθορίζει την λειτουργικότητα των μπλοκ. Ο προγραμματιστής θα πρέπει να καθορίσει τη λειτουργικότητα τους.

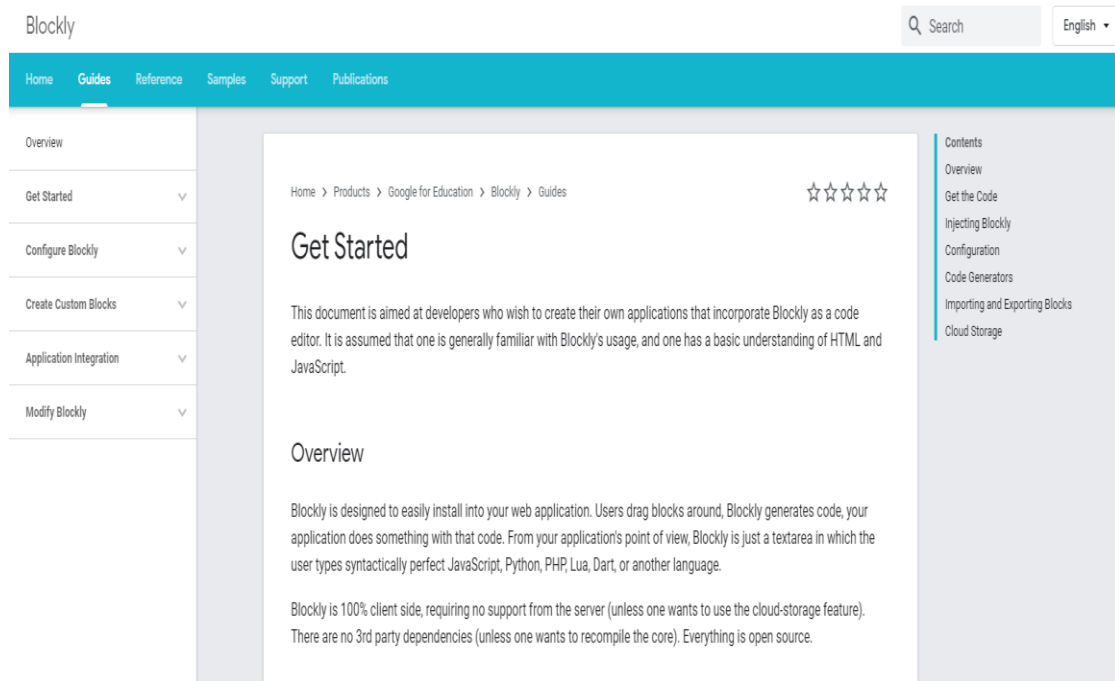
Ο λόγος που κάποιος θα επιλέξει τη βιβλιοθήκη Blockly για την ανάπτυξη της εφαρμογής του, είναι τα θετικά χαρακτηριστικά που αυτή θα του προσφέρει. Τα χαρακτηριστικά αυτά είναι:

- **Εξαγωγίσιμος κώδικας (Exportable code).** Ο χρήστης μπορεί να εξαγάγει τα μπλοκ που βρίσκονται στην περιοχή εργασίας και να τα τροποποιήσει σε μορφή

κειμένου της γλώσσας προγραμματισμού που έχει επιλέξει και υποστηρίζεται από το Blockly.

- **Ανοιχτός κώδικας (Open source).** Ο χρήστης έχει πρόσβαση σε κάθε κομμάτι κώδικα της βιβλιοθήκης. Έτσι έχει την δυνατότητα να τροποποιήσει οτιδήποτε επιθυμεί για να ικανοποιήσει τις ανάγκες της εφαρμογής του.
- **Επεκτάσιμο (Extensible).** Ο προγραμματιστής μπορεί να δημιουργήσει τα δικά του μπλοκ, να τροποποιήσει ήδη υπάρχοντα και να αφαιρέσει όσα δεν τον ενδιαφέρουν.
- **Υψηλές δυνατότητες (Highly capable).** Τα μπλοκ μπορούν να αναλάβουν τον υπολογισμό πολύπλοκων εντολών.
- **Διεθνές (International).** Το Blockly υποστηρίζει την μετάφραση των μπλοκ σε πάνω από 40 γλώσσες.

Επιπροσθέτως στην ιστοσελίδα του Blockly υπάρχει ένας αναλυτικός οδηγός εκμάθησης της βιβλιοθήκης για τους προγραμματιστές εφαρμογών καθώς και αναφορές στα τεκμήρια της.



Εικόνα 13: Οδηγός εκμάθησης της βιβλιοθήκης Blockly

Μέσα από τον οδηγό αυτό ο προγραμματιστής θα κατεβάσει την βιβλιοθήκη, θα μάθει πώς να ενσωματώσει τον συντάκτη Blockly στην εφαρμογή του, πώς να τοποθετήσει τα μπλοκ στον συντάκτη και πώς να δημιουργήσει τα δικά του μπλοκ. Στο τέλος παρέχονται κάποιες σύντομες αναφορές για καλές πρακτικές σχεδίασης των μπλοκ καθώς και ένας σύντομος οδηγός για την ενσωμάτωση ενός διερμηνευτή της JavaScript (JS Interpreter). Αν ο προγραμματιστής θέλει τα μπλοκ του χρήστη να λειτουργούν καλά τότε πρέπει να συνδυάσει το Blockly με τον διερμηνευτή της JavaScript. Έτσι θα πετύχει:

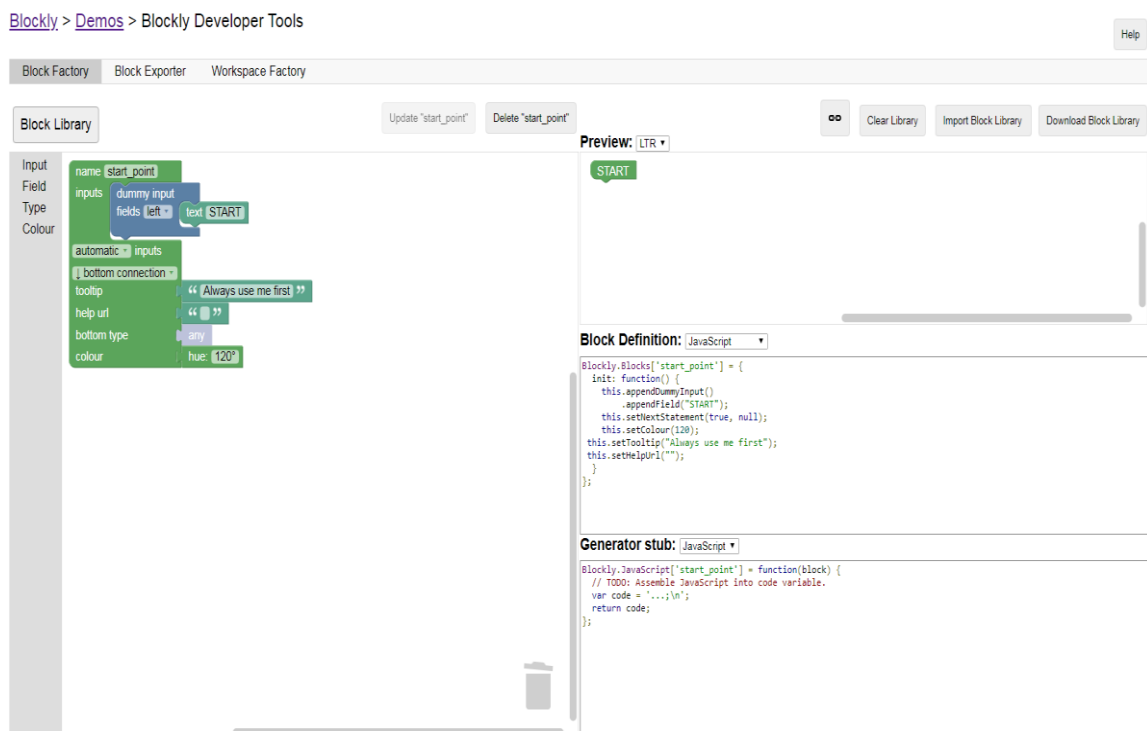
- Την εκτέλεση του κώδικα σε οποιαδήποτε ταχύτητα εκείνος επιλέξει.
- Τη δυνατότητα παύσης (pause), τη συνέχιση (resume) και την εκτέλεση ανά βήμα (step-through) του κώδικα των μπλοκ.
- Την επισκίαση του μπλοκ που εκτελείται εκείνη την στιγμή.
- Την απομόνωση της εκτέλεσης των μπλοκ από την JavaScript του περιηγητή (browser).

Επειδή όμως ένας οδηγός δεν είναι αρκετός για έναν αρχάριο να μάθει να χρησιμοποιεί την εκάστοτε γλώσσα ή βιβλιοθήκη, στην ιστοσελίδα παρέχονται και κάποια παραδείγματα τα οποία εφαρμόζουν την θεωρία που εμφανίζεται στον οδηγό. Έτσι ο προγραμματιστής μπορεί να πειραματιστεί πάνω σε αυτά και να κατανοήσει πλήρως την θεωρία. Μέσα στα παραδείγματα αυτά περιέχεται και ένα με τίτλο "Blockly Developer Tools" το οποίο διαδραματίζει καθοριστικό παράγοντα στην σχεδίαση των εξατομικευμένων μπλοκ.

Η δημιουργία των εξατομικευμένων μπλοκ γίνεται μια εύκολη διαδικασία μέσω του γραφικού περιβάλλοντος του "Blockly Developer Tools". Το συγκεκριμένο εργαλείο χωρίζεται σε τρία τμήματα όπως φαίνεται στον Πίνακας 3.

Πίνακας 3: Τμήματα του Blockly Developer Tools

Blockly Developer Tools	
Εργοστάσιο των μπλοκ (Block Factory)	Είναι υπεύθυνο για τη δημιουργία των μπλοκ.
Εξαγωγέας μπλοκ (Block Exporter)	Αναλαμβάνει την εξαγωγή των επιλεγμένων μπλοκ σε κώδικα της επιλεγμένης γλώσσας προγραμματισμού.
Εργοστάσιο της περιοχής εργασίας (Workspace Factory)	Είναι υπεύθυνο για τη διαμόρφωση της εργαλειοθήκης και της περιοχής εργασίας του συντάκτη της Blockly



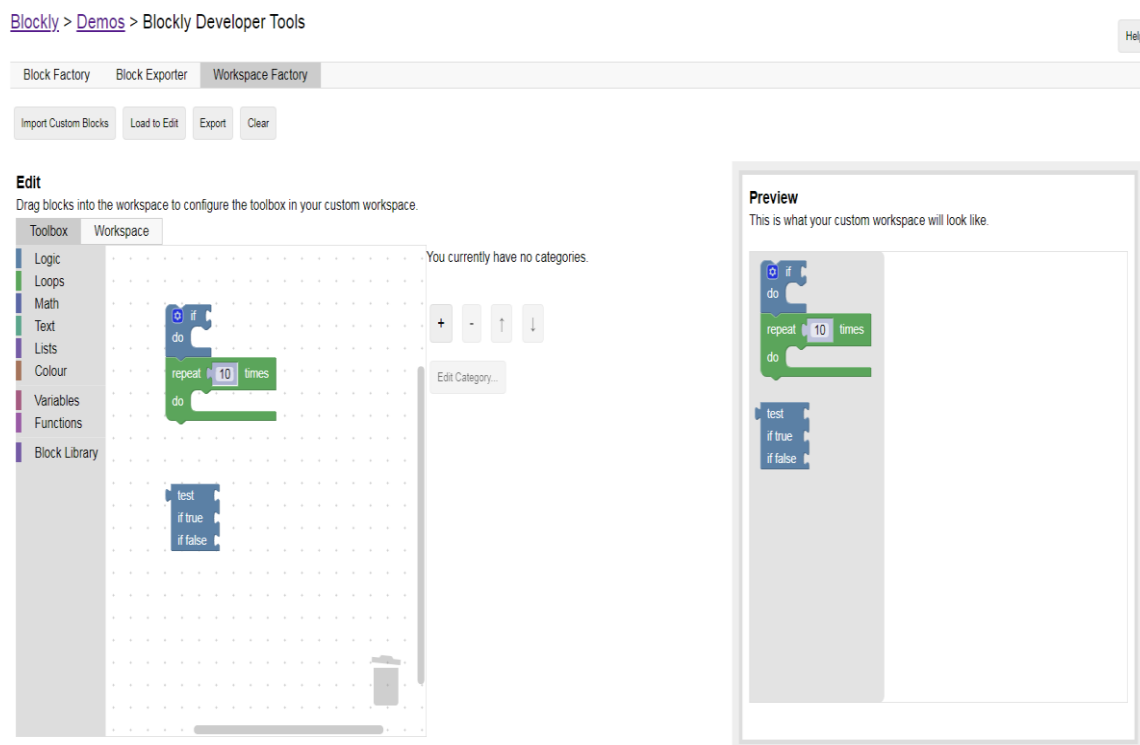
Εικόνα 14: Block Factory

Στην Εικόνα 14 παρουσιάζεται το εργοστάσιο των μπλοκ. Σε αυτό το εργοστάσιο ο προγραμματιστής μπορεί μέσω του γραφικού περιβάλλοντος να δημιουργήσει νέα μπλοκ. Καθώς δημιουργεί τα μπλοκ έχει τη δυνατότητα να ελέγχει στα δεξιά την τελική εμφάνιση του μπλοκ προς κατασκευή. Ακριβώς κάτω από την προεπισκόπηση του μπλοκ υπάρχουν ο ορισμός του μπλοκ (Block Definition) και το στέλεχος παραγωγής

(Generator stub). Ο ορισμός του μπλοκ αφορά τον σχεδιασμό του, όπως για παράδειγμα είναι το χρώμα του, ο τρόπος σύνδεσης του με τα υπόλοιπα μπλοκ, ο τύπος των μπλοκ με τα οποία επιτρέπεται να συνδεθεί κ.α. Το στέλεχος παραγωγής δημιουργεί τον βασικό σκελετό που θα έχει το αντίστοιχο μπλοκ στην αντίστοιχη γλώσσα προγραμματισμού. Στον σκελετό αυτό ο προγραμματιστής αργότερα θα αναπτύξει κώδικα που θα καθορίσει την λειτουργικότητα του αντίστοιχου μπλοκ. Τέλος στο τμήματα του εργοστάσιο παραγωγής μπλοκ, ο προγραμματιστής έχει την δυνατότητα να αποθηκεύσει διαδικτυακά τα μπλοκ που δημιουργεί σε μια δική του βιβλιοθήκη και να τα επαναχρησιμοποιήσει όποτε ο ίδιος επιθυμεί.

Το δεύτερο τμήμα του "Blockly Developer Tools" είναι ο εξαγωγέας των μπλοκ, ο οποίος αναλαμβάνει την εξαγωγή του κώδικα των "Block Definition" και "Generator Stub" σε αρχεία κειμένου.

Το τρίτο και τελευταίο τμήμα είναι το εργοστάσιο επεξεργασίας της περιοχής εργασίας. Σε αυτό το εργοστάσιο ο προγραμματιστής διαμορφώνει τον συντάκτη Blockly που θα ενσωματώσει αργότερα στην εφαρμογή του. Συγκεκριμένα ο προγραμματιστής έχει τη δυνατότητα να διαμορφώσει την τελική εμφάνιση του συντάκτη και να εισάγει τα μπλοκ που θέλει να εμφανίζονται στην εργαλειοθήκη και στην επιφάνεια εργασίας του.



Εικόνα 15: Workspace Factory

Συμπερασματικά η βιβλιοθήκη Blockly είναι ένα εργαλείο παραγωγής γλωσσών οπτικού προγραμματισμού. Η αξιοποίηση της Blockly αποσκοπεί στην ανάπτυξη εφαρμογών και παιχνιδιών που προορίζονται για την εκπαίδευση. Αυτό βέβαια δεν περιορίζει του προγραμματιστές που θέλουν να τη χρησιμοποιήσουν και για άλλο σκοπό πέρα της εκπαίδευσης. Επιπλέον ο οδηγός που παρέχει η ιστοσελίδα βοηθάει έναν προγραμματιστή να μάθει πώς να την χειρίζεται και μέσω της δημιουργίας ενός χώρου συναντήσεων (forum) οι χρήστες μπορούν να λύσουν τυχόν απορίες μεταξύ τους. Με τη βοήθεια αυτών ο προγραμματιστής μπορεί να σχεδιάσει τις δικές του εντολές σε μορφή μπλοκ και να τις ενσωματώσει στην εφαρμογή του δημιουργώντας ένα περιβάλλον που υποστηρίζεται από μια γλώσσα οπτικού προγραμματισμού.

2.5 Παιχνίδια σοβαρού σκοπού για την ανάπτυξη της Υπολογιστικής Σκέψης

Για τη σχεδίαση και την ανάπτυξη του παιχνιδιού σοβαρού σκοπού (BlocklyScript) της εργασίας επιτακτική ήταν η ανάγκη της μελέτης διάφορων παιχνιδιών σοβαρού σκοπού στο διαδίκτυο που στοχεύουν στην ανάπτυξη της Υπολογιστικής Σκέψης. Τα παιχνίδια που μελετήθηκαν δεν περιορίστηκαν μόνο σε εκείνα που υποστηρίζουν τον οπτικό προγραμματισμό αλλά και σε εκείνα που περιέχουν συντάκτη κειμένου. Παρακάτω παρουσιάζονται 5 από τα παιχνίδια.

2.5.1 Code Combat

<https://codecombat.com/home>



Εικόνα 16: Λογότυπο Code Combat

Το Code Combat είναι ένα διαδικτυακό εκπαιδευτικό παιχνίδι ρόλων (Role Playing Game - RPG) που στοχεύει στην εξοικείωση των μαθητών με τον προγραμματισμό. Για την ανάπτυξη του παιχνιδιού συμμετείχαν πάνω από 450 άτομα συνεισφέροντας με κώδικα είτε με ιδέες. Το Code Combat υποστηρίζει τη μετάφραση του σε πάνω από 50 γλώσσες και 190 χώρες παγκοσμίως. Η κοινότητα του απαρτίζεται από περισσότερα από πέντε εκατομμύρια παίκτες, οι οποίοι έχουν ξεπεράσει τον αριθμό τους ενός δισεκατομμυρίου σε γραμμές κώδικα. Επιπλέον το παιχνίδι διατίθεται δωρεάν στους παίκτες εκτός από ορισμένα επίπεδα όπου απαιτείται η συνδρομή. Πληρώνοντας κάποιος τη μηνιαία συνδρομή, η οποία αντιστοιχεί στα 9.99\$, μπορεί να έχει πρόσβαση σε πάνω από 300 επίπεδα, σε κλειδωμένους ήρωες και κατοικίδια και να μάθει πώς να φτιάχνει τα δικά του παιχνίδια και επίπεδα. Σε περίπτωση που ο χρήστης επιθυμεί τη συνδρομή τότε πρέπει να εγγραφεί στο παιχνίδι για να αποθηκευτεί η επιλογή του αλλά και η πρόοδος του στο παιχνίδι.

Με την έναρξη του παιχνιδιού ο παίκτης καλείται να επιλέξει έναν χαρακτήρα και μια γλώσσα προγραμματισμού της επιλογής του. Κάθε χαρακτήρας έχει τη δική του ιστορία και τα δικά του στατιστικά. Επιπροσθέτως οι χαρακτήρες έχουν τον δικό τους προσωπικό χώρο αποθήκευσης για αντικείμενα (π.χ. όπλο, εξοπλισμός κ.τ.λ.) που μπορούν να βρουν κατά τη διάρκεια του παιχνιδιού ως ανταμοιβή για τις επιτυχημένες τους προσπάθειες. Ο εξοπλισμός που κουβαλάει ο χαρακτήρας επηρεάζει τα στατιστικά του τα οποία θα του χρησιμεύουν αργότερα σε πιο απαιτητικά επίπεδα. Πέρα από τον εξοπλισμό ο παίκτης μπορεί να μαζέψει κρυστάλλους και να τους ανταλλάξει αργότερα για εξοπλισμό.



Εικόνα 17: Επιλογή χαρακτήρα και γλώσσας στο Code Combat

Όπως φαίνεται και στην Εικόνα 17 ο χρήστης έχει την δυνατότητα να επιλέξει μεταξύ των γλωσσών προγραμματισμού Python(default), JavaScript και Coffee Script(experimental). Εφόσον ο χρήστης επιλέξει ήρωα και γλώσσα είναι έτοιμος να ξεκινήσει το παιχνίδι.



Εικόνα 18: Επίπεδο παιχνιδιού Code Combat

Στην Εικόνα 18 παρουσιάζεται ένα επίπεδο του παιχνιδιού Code Combat. Με την έναρξη του επιπέδου ο παίκτης μαθαίνει του στόχους του παιχνιδιού και τις διαθέσιμες εντολές με τις οποίες μπορεί να ελέγξει τον ήρωα του. Στο αριστερό μέρος της οθόνης είναι η περιοχή του παιχνιδιού στην οποία ζει ο ήρωας και προσπαθεί να βρει τον σωστό δρόμο για τη λύση. Στο κέντρο της οθόνης βρίσκονται οι διαθέσιμες εντολές για τον έλεγχο του ήρωα και δεξιά από αυτό είναι ο συντάκτης κειμένου στον οποίο ο παίκτης θα γράψει τις κατάλληλες εντολές για να λύσει το πρόβλημα. Επιπλέον σε περίπτωση που ο χρήστης δυσκολευτεί να βρει τη λύση, το παιχνίδι του παρέχει βοήθεια για να τον καθοδηγήσει.

Καθώς ο παίκτης προχωράει σε νέα επίπεδα αντιμετωπίζει νέα εμπόδια και αντιπάλους και μαθαίνει νέες έννοιες. Οι έννοιες αφορούν τη σωστή σύνταξη κώδικα, μεθόδους, παραμέτρους, βρόχους, μεταβλητές, αλφαριθμητικά, δηλώσεις if/else κ.α.. Γενικά το παιχνίδι προορίζεται για μαθητές μικρής ηλικίας που δεν έχουν κάποιες ιδιαίτερες γνώσεις προγραμματισμού και προσπαθεί να τους εντάξει ομαλά στην Επιστήμη των Υπολογιστών.

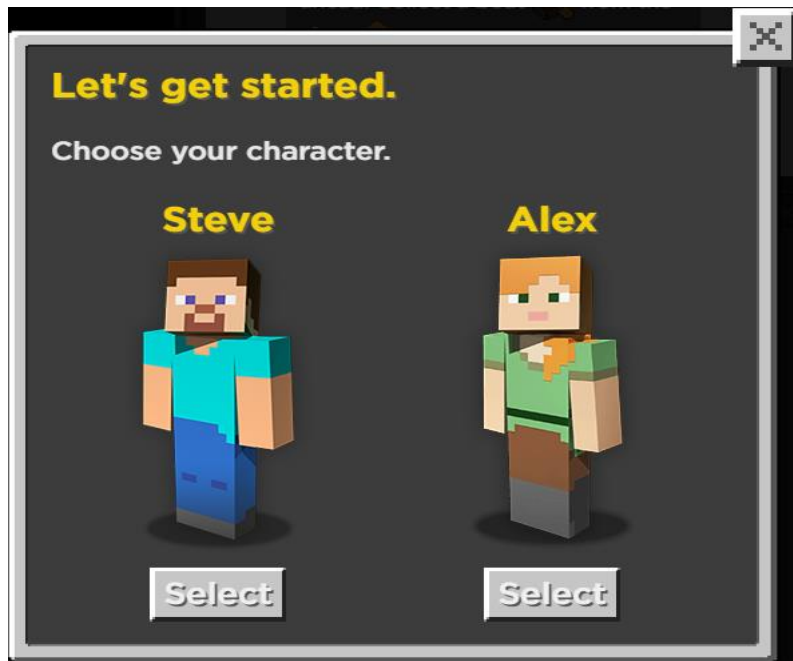
2.5.2 Hour of Code: Minecraft

<https://code.org/minecraft>

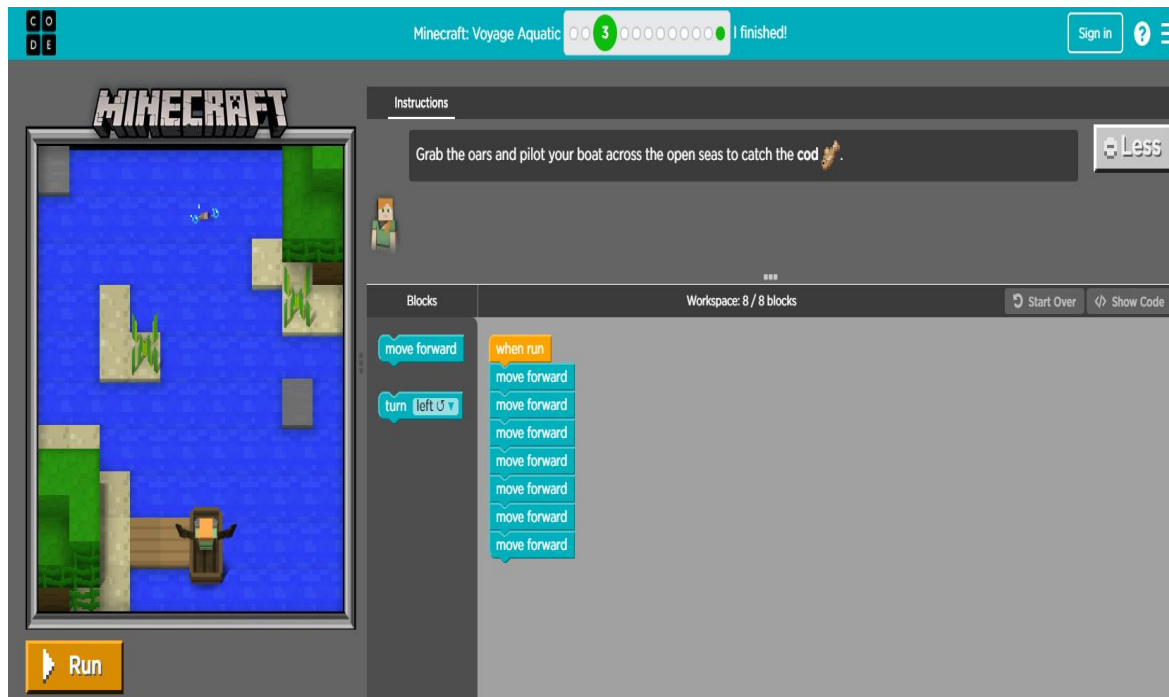
Το παιχνίδι Minecraft Hour of Code δημιουργήθηκε από τον μη κερδοσκοπικό οργανισμό Code.org σε συνεργασία με την Mojang AB και τη Microsoft. Στόχος αυτού του οργανισμού είναι να αυξήσει τη συμμετοχή των γυναικών και των εκπροσωπούμενων μειονοτήτων στην επιστήμη των υπολογιστών. Για τον λόγο αυτό ανέπτυξαν το συγκεκριμένο παιχνίδι που αποσκοπεί στην εισαγωγή των παικτών στις βασικές έννοιες της επιστήμης των υπολογιστών. Στην ανάπτυξη του παιχνιδιού καθοριστικό παράγοντα έπαιξε η βιβλιοθήκη Blockly που παρουσιάστηκε σε προηγούμενη ενότητα, η οποία υποστηρίζει τον οπτικό προγραμματισμό.

Γενικά το παιχνίδι Minecraft Hour of Code αποτελείται από μικρά μαθήματα που διαρκούν κατά μέσο όρο μια ώρα. Προς το παρόν υπάρχουν στην ιστοσελίδα 4 σενάρια Minecraft που έχουν 12-14 επίπεδα το καθένα. Το πρώτο σενάριο αναδεικνύει τον όμορφο κόσμο του βυθού της θάλασσας, το δεύτερο παρουσιάζει το μαγευτικό ταξίδι ενός ήρωα, το τρίτο αφορά τους εξερευνητές και το τελευταίο τους σχεδιαστές. Ακόμη παρέχεται και το Minecraft για τους δασκάλους το οποίο τους δίνει τη δυνατότητα να φτιάξουν τον δικό τους κόσμο Minecraft και τα δικά τους μαθήματα.

Για τα τέσσερα πρώτα σενάρια ο παίκτης μπορεί να επιλέξει μεταξύ μιας γυναίκας και ενός άντρα. Με την έναρξη των επιπέδων παρουσιάζονται στον παίκτη οι στόχοι του εκάστοτε επιπέδου, η περιοχή του παιχνιδιού και η περιοχή με τα διαθέσιμα μπλοκ εντολών. Συνδέοντας τα μπλοκ ο παίκτης ελέγχει τον χαρακτήρα του παιχνιδιού και τον καθοδηγεί στον στόχο του. Σε περίπτωση που ο παίκτης δυσκολεύεται να βρει τη λύση παρέχεται βοήθεια για το χρήστη είτε με τη μορφή κειμένου ή βίντεο.



Εικόνα 19: Επιλογή χαρακτήρα στο Minecraft Hour of Code



Εικόνα 20: Επίπεδο παιχνιδιού Minecraft Hour of Code

2.5.3 Rapid Router

<https://www.codeforlife.education/rapidrouter/>

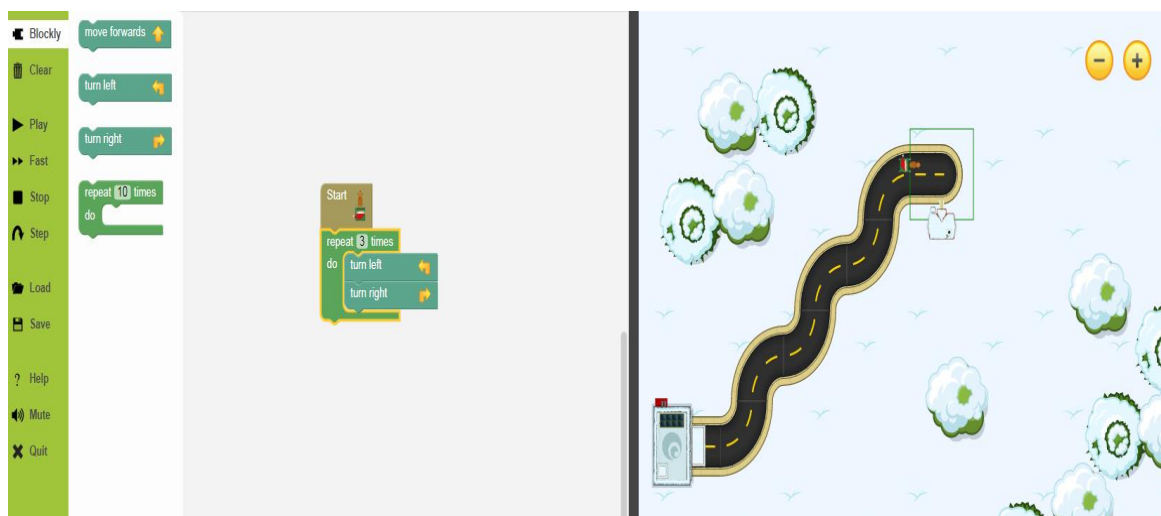
Το Rapid Router είναι ένα εκπαιδευτικό παιχνίδι που έχει αναπτυχθεί από τον μη κερδοσκοπικό οργανισμό Code for Life και στόχο έχει να εντάξει τους μαθητές μικρής ηλικίας στην επιστήμη των Υπολογιστών. Πρόκειται για ένα παιχνίδι που ενσωματώνει τον οπτικό προγραμματισμό με τη βοήθεια της βιβλιοθήκης Blockly, αλλά περιέχει και συντάκτη κειμένου για τα τελευταία του επίπεδα που αφορούν τη γλώσσα προγραμματισμού Python. Ο παίκτης μπορεί να μάθει παίζοντας μέσα από 109 επίπεδα, καθοδηγώντας ένα όχημα να φτάσει στον προορισμό του. Σέρνοντας και αφήνοντας μπλοκ εντολών το ένα κάτω από το άλλο ο χρήστης προσανατολίζει την πορεία που θα ακολουθήσει το όχημα, στρίβοντας το δεξιά και αριστερά ή οδηγώντας ευθεία μπροστά. Στα επίπεδα του παιχνιδιού παρουσιάζονται βασικές έννοιες προγραμματισμού όπως οι εντολές if, οι βρόχοι και οι επαναλήψεις, αλλά υπάρχουν και επίπεδα που προκαλούν τον παίκτη να βρει τη συντομότερη διαδρομή ή να χρησιμοποιήσει περιορισμένο αριθμό από μπλοκ. Σε κάθε περίπτωση ο παίκτης βαθμολογείται για την απάντηση του και επιβραβεύεται με άριστα ή βέλτιστη λύση.

Levels

▶ Getting Started	Levels 1-12
▶ Shortest Route	Levels 13-18
▶ Loops and Repetitions	Levels 19-28
▶ Loops with Conditions	Levels 29-32
▶ If... Only	Levels 33-43
▶ Traffic Lights	Levels 44-50
▶ Limited Blocks	Levels 51-60
▶ Procedures	Levels 61-67
▶ Blockly Brain Teasers	Levels 68-79
▶ Introduction to Python	Levels 80-91
▶ Python	Levels 92-109

Εικόνα 21: Επίπεδα παιχνιδιού Rapid Router

Πριν ξεκινήσει το επίπεδο γίνεται στον παίκτη μια εισαγωγή για τα νέα μπλοκ που υπάρχουν καθώς και οι στόχοι του. Σε περίπτωση που ο παίκτης δυσκολευτεί μπορεί να ζητήσει βοήθεια πατώντας το κουμπί "? Help" ή στο τέλος του επιπέδου να εμφανιστεί μια βοήθεια για την λύση του προβλήματος. Επιπλέον το παιχνίδι βοηθάει τον παίκτη να καταλάβει ποια εντολή οδήγησε το αυτοκίνητο εκτός δρόμου και ποια πιθανά μπλοκ δεν συνδέθηκαν μεταξύ τους ενώ περίμεναν να συνδεθούν με κάποιο άλλο μπλοκ (π.χ. το μπλοκ while).



Εικόνα 22: Επίπεδο παιχνιδιού Rapid Router

Τέλος το Rapid Router δίνει τη δυνατότητα στους εκπαιδευτικούς να δημιουργήσουν τα δικά τους επίπεδα και μαθήματα παρέχοντας σε αυτούς τα κατάλληλα εργαλεία. Για τα μαθήματα τους οι εκπαιδευτικοί μπορούν να δημιουργήσουν τμήματα με μαθητές και να ελέγχουν την πορεία τους στο παιχνίδι. Έτσι μπορεί να καταλάβει σε ποιο σημείο ίσως να δυσκολεύονται περισσότερο οι μαθητές και να τους βοηθήσει να κατανοήσουν τις έννοιες αυτές.

2.5.4 Run Marco!

<https://www.allcancode.com/hourofcode>

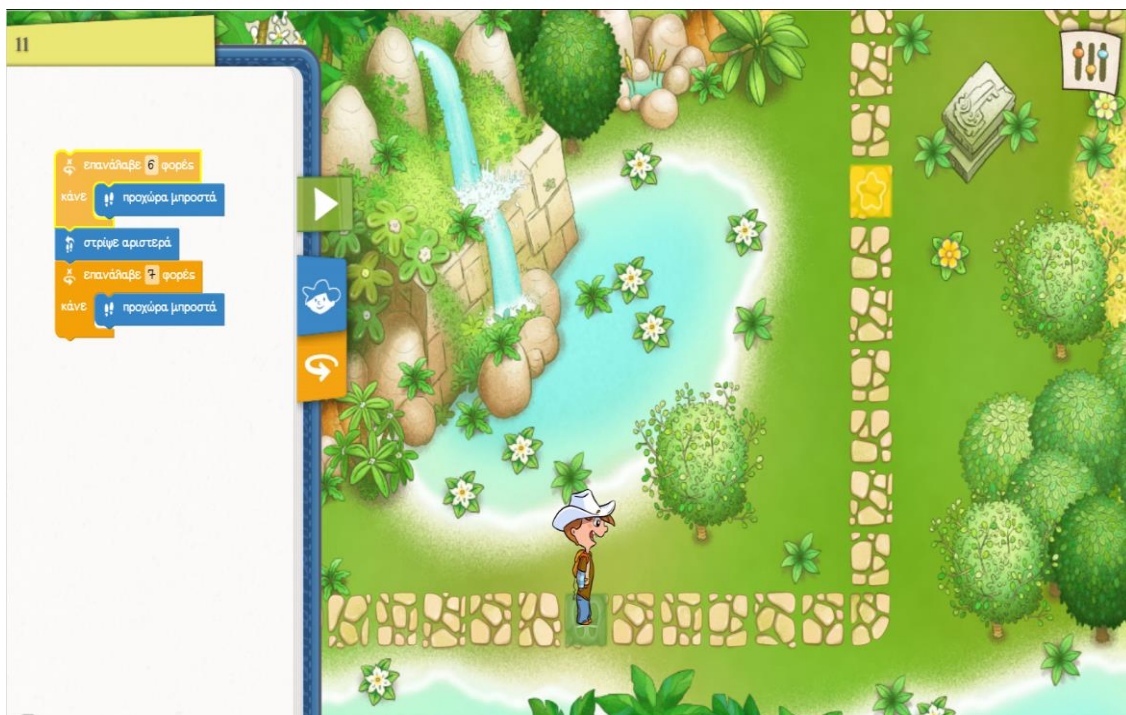
Το παιχνίδι Run Marco! είναι ένα παιχνίδι σοβαρού σκοπού ανεπτυγμένο από την ομάδα προγραμματιστών Allcancode και στόχο έχει την ανάπτυξη της Υπολογιστικής Σκέψης σε μικρούς μαθητές. Το παιχνίδι έχει σχεδιαστεί με τη βοήθεια της βιβλιοθήκης Blockly

και ενσωματώνει τον οπτικό προγραμματισμό. Επιπλέον το παιχνίδι περιλαμβάνει αρκετά επίπεδα τα οποία καλείται να επιλύσει ο μαθητής χωρίς να έχει ιδιαίτερες γνώσεις προγραμματισμού. Μέσω των επιπέδων παρουσιάζονται νέες έννοιες και εντολές όπως είναι η αλληλουχία των εντολών, απλές εντολές If, βρόχοι κ.α. οι οποίες θα τον βοηθήσουν να φτάσει στη λύση του επιπέδου.

Αρχικά με την έναρξη του παιχνιδιού ο χρήστης καλείται να επιλέξει μεταξύ ενός κοριτσιού ή ενός αγοριού εξερευνητή. Αφού επιλέξει τον χαρακτήρα του επιλέγει το επίπεδο που έχει ξεκλειδώσει και επιθυμεί να παίξει. Στόχος του παίκτη είναι να καθοδηγήσει τον εξερευνητή μέσω του μονοπατιού στη ζούγκλα για να φτάσει στον προορισμό του και να βρει τους φίλους του. Σε περίπτωση που ο παίκτης δυσκολευτεί να βρει την λύση μπορεί να ζητήσει τη βοήθεια του παιχνιδιού, η οποία του δείχνει με ποια σειρά θα πρέπει να συνδεθούν τα μπλοκ μέσω κινούμενων εικόνων. Κάθε φορά που ο παίκτης κερδίζει το επίπεδο επιβραβεύεται για τη λύση του.



Εικόνα 23: Επιλογή χαρακτήρα στο παιχνίδι Run Marco!



Εικόνα 24: Επίπεδο παιχνιδιού Run Marco!

2.6 Συγκριτική ανάλυση παιχνιδιών σοβαρού σκοπού

Στην προηγούμενη ενότητα παρουσιάστηκαν τέσσερα παιχνίδια σοβαρού σκοπού που ως στόχο έχουν την εισαγωγή μικρών μαθητών στην επιστήμη των Υπολογιστών. Κάθε ένα από τα παιχνίδια αυτά ενσωματώνει κάποιες από τις βασικές έννοιες του προγραμματισμού, οι οποίες μπορούν να βοηθήσουν στην ανάπτυξη της Υπολογιστικής Σκέψης του μαθητή. Στον Πίνακα 4 καταγράφονται οι προγραμματιστικές έννοιες που εμφανίζονται στα παιχνίδια που παρουσιάστηκαν.

Πίνακας 4: Προγραμματιστικές έννοιες παιχνιδιών

Προγραμματιστικές έννοιες	Παιχνίδια			
	Code Combat	Minecraft Hour of Code	Rapid Router	Run Marco!
Σειριακές εντολές	✓	✓	✓	✓
Απλή εντολή if	✓	✓	✓	✓
Εντολή if...else	✓	✓	✓	✓
Εμφωλευμένη εντολή if	✓	✗	✓	✓
Βρόχος for	✓	✓	✓	✓
Βρόχος Do while or Repeat until	✓	✓	✓	✓
Συναρτήσεις	✓	✓	✓	✗
Αντικειμενοστρεφής προγραμματισμός	✓	✗	✓	✗
Μεταβλητές	✓	✗	✓	✗

Επιπλέον μελετήθηκαν και κάποια βασικά χαρακτηριστικά που πρέπει να πληρεί ένα παιχνίδι σοβαρού σκοπού. Τα χαρακτηριστικά αυτά μπορούν να χωριστούν σε τέσσερις κατηγορίες. Η πρώτη κατηγορία αφορά το σενάριο του παιχνιδιού, η δεύτερη τους μηχανισμούς του, η τρίτη την εκτέλεση του προγράμματος καθώς και την υποστήριξη ελέγχων και αποσφαλμάτωσης και η τέταρτη κατηγορία επισημαίνει την διαθεσιμότητα του παιχνιδιού στις πλατφόρμες iOS, Android και διαδίκτυο (Giannakoulas & Xinogalos (2018)). Ακόμη πέρα από τις τέσσερις κατηγορίες, στη μελέτη συμπεριλήφθηκε και το

προαπαιτούμενο επίπεδο γνώσεων των παικτών. Στον Πίνακα 5 καταγράφονται οι κατηγορίες που αναφέρθηκαν.

Πίνακας 5: Χαρακτηριστικά παιχνιδιών

Κατηγορίες	Χαρακτηριστικά	Παιχνίδια			
		Code Combat	Minecraft Hour of Code	Rapid Router	Run Marco!
	Προαπαιτούμενες γνώσεις	Όχι	Όχι	Όχι	Όχι
	Εκπαιδευτικό επίπεδο παικτών	Δημοτικού	Δημοτικού	Δημοτικού	Δημοτικού
Σενάριο παιχνιδιού	Ποικιλία δραστηριοτήτων	Προσδιορισμός της κίνησης του χαρακτήρα μέσω αλγορίθμων	Προσδιορισμός της κίνησης του χαρακτήρα μέσω αλγορίθμων	Προσδιορισμός της κίνησης του χαρακτήρα μέσω αλγορίθμων	Προσδιορισμός της κίνησης του χαρακτήρα μέσω αλγορίθμων
	Δυνατότητα επιλογής χαρακτήρα	✓	✓	✗	✓
	Εχθροί - αντίπαλοι	✓	✓	✗	✗
	Υπάρχουσες δραστηριότητες	Σχεδιασμός αλγορίθμου και ανάθεση τιμών σε μεταβλητές	Σχεδιασμός αλγορίθμου	Σχεδιασμός αλγορίθμου	Σχεδιασμός αλγορίθμου
Μηχανισμοί Παιχνιδιού	Σκελετός προγράμματος / Υποστήριξη	Οδηγίες ανά επίπεδο σε μορφή κειμένου	Οδηγίες ανά επίπεδο σε μορφή κειμένου, εισαγωγικά βίντεο	Οδηγίες ανά επίπεδο σε μορφή κειμένου	Οδηγίες ανά επίπεδο σε μορφή κειμένου
	Σκορ	✓	✗	✓	✗
	Επιβράβευση με την νίκη κάθε επιπέδου	✓	✓	✓	✓
	Επιβράβευση καλύτερη λύσης	✗	✓	✓	✓
	Επιτεύγματα	✓	✗	✗	✗

	Γλώσσα προγραμματισμού	Python, JavaScript, Coffee Script	Οπτικός προγραμ/μός με Blockly	Οπτικός προγραμ/μός με Blockly	Οπτικός προγραμ/μός με Blockly
Εκτέλεση προγράμματος- Έλεγχος και αποσφαλμάτωση.	Επισήμανση της εντολής προς εκτέλεση	✓	✗	✓	✓
	Εκτέλεση του προγράμματος σε διαφορετικές ταχύτητες	✓	✗	✓	✗
	Ανατροφοδότηση σε περίπτωση λάθους	✓	✗	✓	✗
	Έλεγχος και αποσφαλμάτωση	Εκτελώντας το πρόγραμμα	Εκτελώντας το πρόγραμμα	Εκτελώντας το πρόγραμμα ή ανά εντολή	Εκτελώντας το πρόγραμμα
	Διακοπή εκτέλεσης προγράμματος	✓	✗	✓	✗
	Διαδικτυακές τάξεις	✓	✓	✓	✗ (Δεν υποστηρίζεται πλέον)
	Δυνατότητα δημιουργίας επιπέδων	✓	✓	✓	✗
Διαθέσιμες πλατφόρμες	Διαδίκτυο	✓	✓	✓	✓
	Android	✗	✗	✗	✓
	iOS	✗	✗	✗	✓

2.6.1 Συμπεράσματα ανάλυσης

Παρατηρώντας κανείς τον πίνακα 4 της προηγούμενης ενότητας μπορεί να διαπιστώσει ότι τα τέσσερα παιχνίδια που μελετήθηκαν ενσωματώνουν τις περισσότερες από τις βασικές έννοιες μιας γλώσσας προγραμματισμού. Ενδιαφέρον αποτελεί η προσπάθεια των παιχνιδιών Code Combat και Rapid Router, τα οποία συμπεριλαμβάνουν όλες τις

έννοιες παρέχοντας έτσι στον χρήστη μια σφαιρική εικόνα για τον προγραμματισμό. Επιπλέον αξίζει να αναφερθεί η αποχή εννοιών αντικειμενοστρεφούς προγραμματισμού και μεταβλητών από τα παιχνίδια Minecraft Hour of Code και Run Marco!. Συγκεκριμένα η έννοια του αντικειμενοστρεφή προγραμματισμού απουσιάζει σε όλα τα παιχνίδια που χρησιμοποιούν γλώσσα οπτικού προγραμματισμού. Ακόμη και στο παιχνίδι Rapid Router η έννοια αυτή εμφανίζεται στα τελευταία επίπεδα του παιχνιδιού που υποστηρίζουν τη σύνταξη των εντολών με κείμενο.

Επιπροσθέτως, με βάση τα δεδομένα του πίνακα 5, όλα τα παιχνίδια προορίζονται για μαθητές μικρής ηλικίας ξεκινώντας από τις τάξεις του δημοτικού. Για τον εξής λόγο τα παιχνίδια δεν προϋποθέτουν συγκεκριμένες γνώσεις από τους παίκτες. Παίζοντας ο μαθητής θα έρθει σε μια πρώτη επαφή με τις νέες έννοιες του προγραμματισμού και θα έχει την δυνατότητα να αναπτύξει την Υπολογιστική του Σκέψη.

Επιπλέον στην κατηγορία του σεναρίου σε όλα τα παιχνίδια ο χρήστης μπορεί να ελέγξει την κίνηση του παίκτη μέσω της σχεδίασης αλγορίθμων. Μάλιστα στα παιχνίδια εκτός του Rapid Router ο χρήστης έχει τη δυνατότητα να επιλέξει τον χαρακτήρα που θα ελέγχει αργότερα με σκοπό να ταυτιστεί με την επιλογή του. Πέρα από την κίνηση του χαρακτήρα όμως, το Code Combat και το Minecraft περιέχουν και εχθρούς - αντιπάλους στο σενάριο με σκοπό να εντείνουν το στοιχείο της δράσης μέσα στο παιχνίδι και να προσελκύσουν την προσοχή του παίκτη.

Η επόμενη κατηγορία προς μελέτη αφορά τους μηχανισμούς των παιχνιδιών. Στην κατηγορία αυτή όλα τα παιχνίδια δίνουν βοήθεια στους παίκτες με την μορφή μηνυμάτων σε περίπτωση που δυσκολευτούν να βρουν τη λύση. Το Minecraft επιπλέον παρέχει σε ορισμένα επίπεδα και εισαγωγικά βίντεο για να καλωσορίσει τους παίκτες και να τους εξηγήσει κάποιες βασικές έννοιες που θα μάθουν. Γενικότερα, όμως, σε αυτήν την κατηγορία υπερτερούν τα παιχνίδια Code Combat και Rapid Router έναντι των άλλων μιας και ενσωματώνουν περισσότερα χαρακτηριστικά.

Επίσης τα δύο συγκεκριμένα παιχνίδια διατηρούν την υπεροχή τους στην κατηγορία των ελέγχων και της αποσφαλμάτωσης. Είναι σημαντικό για τον χρήστη να γνωρίζει ποια εντολή εκτελείται τη στιγμή που εκτελείται το πρόγραμμα. Εφόσον γνωρίζει σε ποιο σημείο βρίσκεται είναι πιο εύκολο για τον ίδιο να κατανοήσει τη λειτουργικότητα των εντολών αλλά και να εντοπίσει τυχόν σφάλματα στον αλγόριθμό του. Έπειτα εντοπίζοντας τα σφάλματα και διαβάζοντας τη βοήθεια που του προσφέρει το παιχνίδι

μπορεί να επανασχεδιάσει τον αλγόριθμό του διορθώνοντας κάποια βήματα που τυχόν δεν πρόσεξε.

Εκτός των κατηγοριών του πίνακα 5 δύο πολύ σημαντικά χαρακτηριστικά είναι η γλώσσα προγραμματισμού και η δυνατότητα δημιουργίας διαδικτυακών τάξεων. Αρχικά, όσον αφορά τη γλώσσα προγραμματισμού τα τρία παιχνίδια εκτός του Code Combat χρησιμοποιούν γλώσσα οπτικού προγραμματισμού που ανέπτυξαν με την βοήθεια της Blockly. Οι εντολές των παιχνιδιών παρουσιάζονται με τη μορφή αλληλοσυνδεδεμένων μπλοκ γεγονός που διευκολύνει τον χρήστη να κατανοήσει τις έννοιες του. Το Code Combat, από την άλλη, προσφέρει έναν συντάκτη κειμένου για τις εντολές αλλά παρέχει και τη δυνατότητα επιλογής μιας από τις τρεις διαθέσιμες γλώσσες του λαμβάνοντας έτσι υπόψη τις προτιμήσεις του εκάστοτε παίκτη.

Όσο αφορά τη δημιουργία διαδικτυακών τάξεων, το Run Marco! είναι το μόνο παιχνίδι που δεν υποστηρίζει αυτή τη δυνατότητα. Γενικά η δημιουργία διαδικτυακών τάξεων είναι ένα πολύ χρήσιμο εργαλείο για τους δασκάλους, μιας και μέσω αυτού μπορούν να ελέγχουν την πρόοδο των μαθητών τους αλλά και να δημιουργούν δικά τους μαθήματα προσαρμοσμένα στις ανάγκες τους. Επίσης έχουν την δυνατότητα να αναλάβουν την εκπαίδευση μαθητών από όλο τον κόσμο χωρίς να υπάρχουν γεωγραφικά σύνορα. Έτσι ο δάσκαλος θα έχει τη δυνατότητα να απαντάει σε απορίες των μαθητών του που δεν βρίσκονται εκείνη τη στιγμή μπροστά του και να τους καθοδηγεί μέσω μηνυμάτων.

Κλείνοντας μπορεί εύκολα κανείς να δει ότι τα παιχνίδια βρίσκονται διαθέσιμα στο διαδίκτυο με εξαίρεση το Run Marco!, το οποίο παρέχεται και στις πλατφόρμες Android και iOS. Τα παιχνίδια αυτά διατίθενται δωρεάν εκτός από το Code Combat που απαιτείται η συνδρομή του χρήστη για να ξεκλειδωθούν τα περισσότερα επίπεδα του.

3 Περιγραφή παιχνιδιού

3.1 Εισαγωγή

Το παιχνίδι που αναπτύχθηκε στα πλαίσια της διπλωματικής εργασίας ονομάζεται BlocklyScript. Πρόκειται για ένα παιχνίδι ρόλων που συνδυάζει τα στοιχεία της περιπέτειας και της δράσης. Το σενάριο του παιχνιδιού εξελίσσεται στο φανταστικό κόσμο της Blocklyland, η οποία αντιμετωπίζει ένα σοβαρό πρόβλημα. Ξαφνικά, πριν από μερικές εβδομάδες, άρχισαν να πέφτουν από τον ουρανό τα αστέρια και να προσγειώνονται στην Blocklyland. Το γεγονός αυτό άρχισε να απασχολεί τους πολίτες της χώρας διότι ανησυχούσαν ότι στο τέλος θα πέσουν όλα τα αστέρια με αποτέλεσμα να ζουν μονίμως στο σκοτάδι. Για τον λόγο αυτό εμφανίστηκαν τολμηροί εξερευνητές που ως βασικό τους στόχο έχουν να ταξιδέψουν σε όλη την χώρα και να βρουν τα πεφταστέρια με σκοπό να τα ξαναστείλουν πίσω στην αρχική τους θέση στο διάστημα.

Στόχος του χρήστη είναι να επιλέξει μεταξύ δύο διαθέσιμων εξερευνητών εκείνον που του ταιριάζει και να τον καθοδηγήσει με τη βοήθεια μπλοκ εντολών να μαζέψει όλα τα αστέρια και να σώσει την Blocklyland. Παίζοντας ο χρήστης θα έρθει σε μια πρώτη επαφή με προγραμματιστικές έννοιες και δομές όπως η εντολή if-else, ο βρόχος repeat while, οι μεταβλητές κ.α. Επίσης μέσω του τρόπου σκέψης για την επίλυση του προβλήματος θα έχει τη δυνατότητα να αναπτύξει την Υπολογιστική του Σκέψη αλλά και να διασκεδάσει.

Ακόμη το παιχνίδι έχει διαμορφωθεί με τέτοιο τρόπο ώστε να υποστηρίζει τον παίκτη σε κάθε επίπεδο, δίνοντας του ξεκάθαρους στόχους και γενικότερες οδηγίες για τα εμπόδια του κάθε επιπέδου αλλά και για τα μπλοκ των εντολών. Σε περίπτωση που δεν κατανοήσει πλήρως τις οδηγίες και καταλήξει να μην μπορεί να κερδίσει το επίπεδο το παιχνίδι μέσω της περιοχής Program Output προσφέρει επιπλέον βοήθεια και προτείνει διορθώσεις που οδηγούν στη βέλτιστη λύση του προβλήματος.

Πέρα από το σενάριο, τους εκπαιδευτικούς στόχους και την υποστήριξη του παίκτη το παιχνίδι υποστηρίζεται από μια βάση δεδομένων. Η βάση δεδομένων διατηρεί την πρόοδο του χρήστη μαζί με κάποια χρήσιμα στατιστικά από τις λύσεις. Έτσι ο παίκτης έχει την δυνατότητα να ασχοληθεί και να μάθει με το BlocklyScript όποτε διαθέτει ελεύθερο χρόνο και έχει την όρεξη για να ασχοληθεί.

3.2 Είσοδος και εγγραφή χρήστη στο παιχνίδι

Για να μπορέσει κάποιος να παίξει το παιχνίδι και να αποθηκευτεί η πρόοδος του σε αυτό απαραίτητη προϋπόθεση αποτελεί η εγγραφή του στο σύστημα. Η εγγραφή μπορεί να γίνει εδώ: <https://desperance.000webhostapp.com/register.php> και είναι μια εύκολη διαδικασία. Δεν χρειάζεται η ηλεκτρονική διεύθυνση του χρήστη παρά μόνο ένα μοναδικό ψευδώνυμο και ένα συνθηματικό. Εφόσον γίνει η επιτυχημένη εγγραφή στο σύστημα ο χρήστης κατευθύνεται στην αρχική οθόνη του παιχνιδιού και είναι έτοιμος να ξεκινήσει.



Εικόνα 25: Οθόνη εγγραφής BlocklyScript

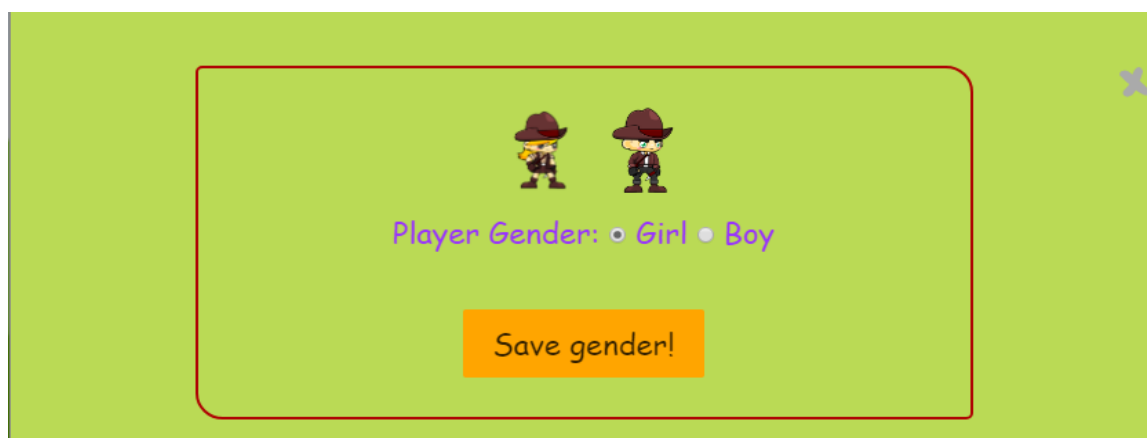
Σε περίπτωση που ο χρήστης έχει κάνει ήδη εγγραφή στο παιχνίδι τότε μπορεί να κατευθυνθεί στην οθόνη εισόδου και να πληκτρολογήσει το ψευδώνυμο σε συνδυασμό με το συνθηματικό του. Εφόσον τα πληκτρολογήσει σωστά θα οδηγηθεί στην αρχική οθόνη του παιχνιδιού, διαφορετικά θα εμφανιστούν κατάλληλα μηνύματα για λάθος ψευδώνυμο και συνθηματικό.



Εικόνα 26: Οθόνη εισόδου με εσφαλμένα στοιχεία.

3.3 Αρχική οθόνη χρήστη

Με την επιτυχή είσοδο του χρήστη στο σύστημα εμφανίζεται η αρχική οθόνη του χρήστη. Η πρώτη ενέργεια που καλείται να κάνει ο χρήστης είναι η επιλογή του εξερευνητή στο παιχνίδι. Οι επιλογές του είναι δύο και καθορίζουν αν το φύλλο του χαρακτήρα θα είναι αγόρι ή κορίτσι. Η επιλογή που θα κάνει ο χρήστης αποθηκεύεται αργότερα με την ολοκλήρωση του πρώτου επιπέδου του παιχνιδιού και δεν θα ξαναζητηθεί. Η παρακάτω εικόνα δείχνει τις διαθέσιμες επιλογές του παίκτη.



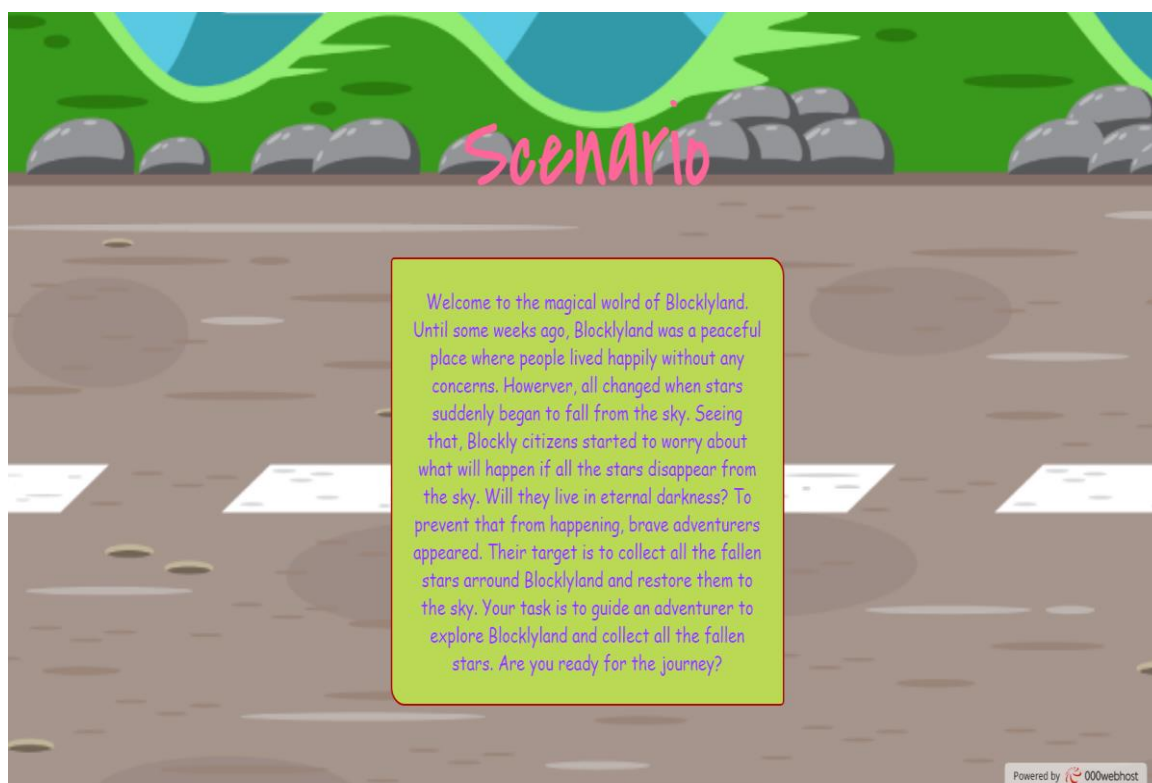
Εικόνα 27: Αρχική οθόνη - επιλογή χαρακτήρα

Εφόσον γίνει η επιλογή του χαρακτήρα και αποθηκευτεί το γένος ο χρήστης μπορεί να περιπλανηθεί ελεύθερα στην αρχική οθόνη. Η θεματολογία της αρχικής οθόνης διαχωρίζεται σε τέσσερις βασικές κατηγορίες. Η πρώτη κατηγορία είναι το σπίτι (Home) και είναι το τμήμα καλωσορίσματος του χρήστη στο παιχνίδι. Αρχικά δείχνει στον χρήστη ότι είναι συνδεδεμένος στο σύστημα και έπειτα τον καλωσορίζει. Επιπλέον δίνει στον χρήστη τους βασικούς στόχους ανάπτυξης του παιχνιδιού και ακριβώς κάτω από το κείμενο υπάρχει ένας σύνδεσμος για την έξοδο (logout) του χρήστη από το σύστημα καθώς και ένα κουμπί έναρξης του παιχνιδιού. Με το που πατήσει το κουμπί έναρξης ο χρήστης θα οδηγηθεί σε εκείνο το επίπεδο που έμεινε την τελευταία φορά και δεν κατάφερε ακόμα να κερδίσει.



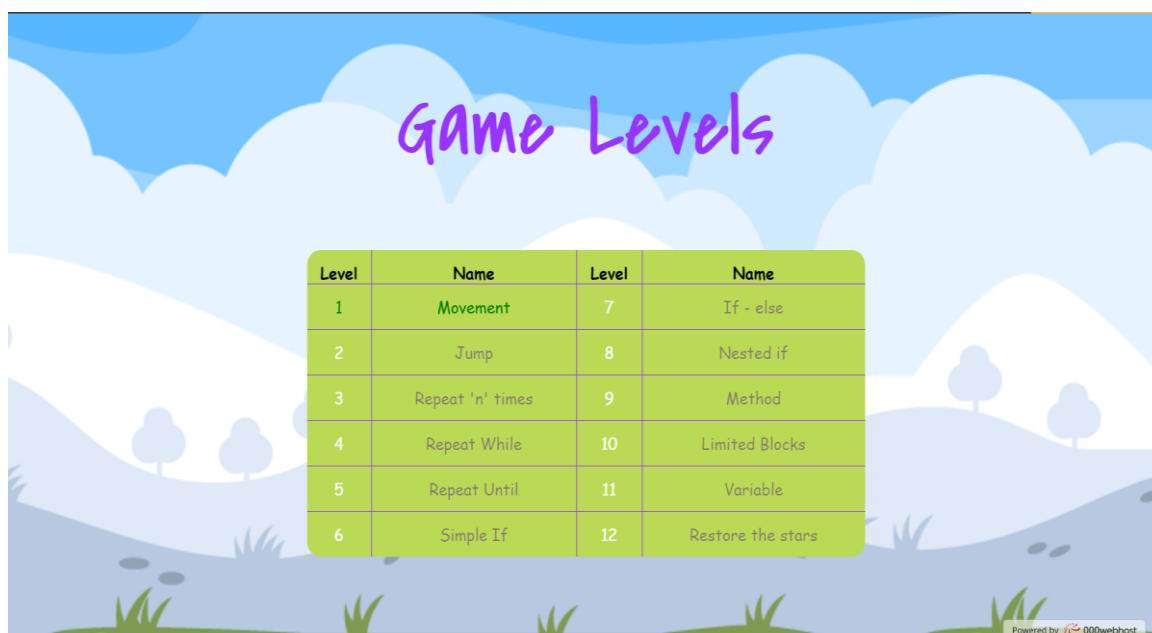
Εικόνα 28: Οθόνη καλωσορίσματος - έναρξης παιχνιδιού.

Η δεύτερη κατηγορία περιλαμβάνει το σενάριο του παιχνιδιού. Στόχος του σεναρίου είναι να τραβήξει το ενδιαφέρον του χρήστη και να του δημιουργήσει ξεκάθαρους στόχους όταν το παίζει. Έτσι μέσα από το σενάριο ο στόχος του παίκτη είναι να βοηθήσει τον χαρακτήρα που επέλεξε αρχικά και να τον καθοδηγήσει για να μαζέψει όλα τα πεφταστέρια και να τα επαναφέρει στην αρχική τους θέση στο διάστημα.



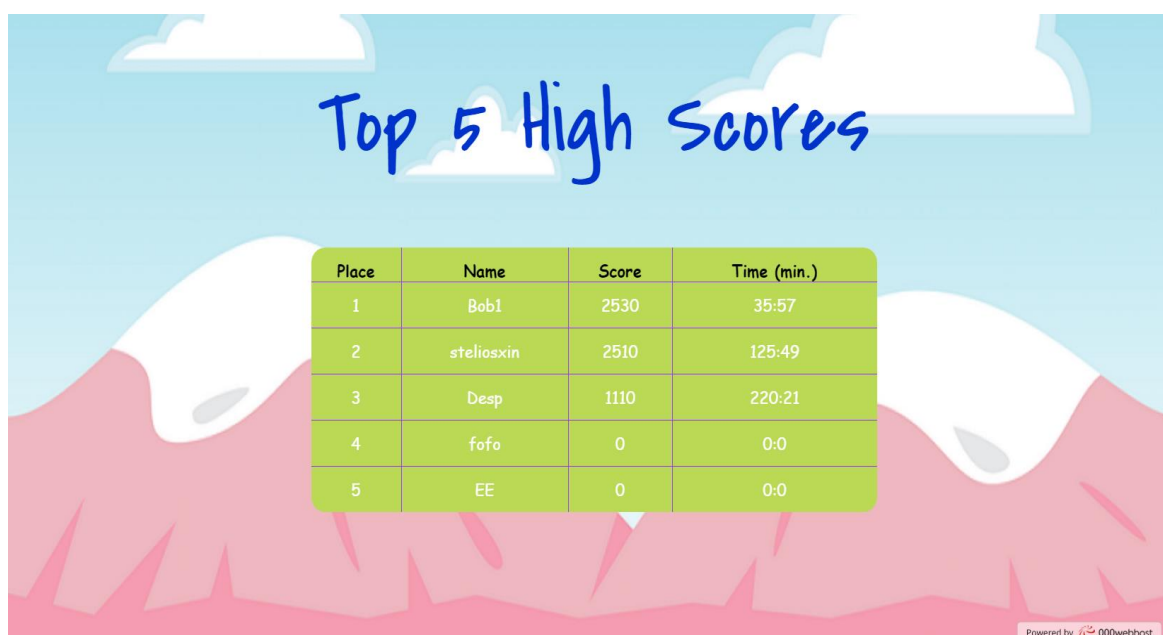
Εικόνα 29: Σενάριο BlocklyScript

Στην τέταρτη κατηγορία της αρχικής οθόνης εμφανίζεται ένας πίνακας με όλα τα επίπεδα του παιχνιδιού και τους διδακτικούς τους στόχους. Όπως φαίνεται και από την Εικόνα 30 τα επίπεδα του παιχνιδιού είναι 12 και περιλαμβάνουν διάφορες προγραμματιστικές έννοιες. Στην αρχή τα στοιχεία με τα επίπεδα του πίνακα είναι γκρι και δεν επιτρέπεται στον χρήστη να κατευθυνθεί στα αντίστοιχα επίπεδα. Μόλις όμως ο χρήστης κερδίσει κάποια επίπεδα τότε τα αντίστοιχα στοιχεία του πίνακα γίνονται πράσινα και ουσιαστικά ξεκλειδώνονται. Έτσι ο χρήστης μπορεί να ξαναπαίξει όσες φορές επιθυμεί τα επίπεδα που έχει ήδη ξεκλειδώσει.



Εικόνα 30: Επίπεδα του BlocklyScript με ξεκλειδωμένο το πρώτο επίπεδο

Τέλος, η κατηγορία που απομένει περιέχει έναν πίνακα με τα 5 υψηλότερα σκορ των παικτών. Για την εμφάνιση ενός παίκτη στον συγκεκριμένο πίνακα θα πρέπει πρώτα να έχει συγκεντρώσει του περισσότερους πόντους και να βρίσκεται στην αρχική πεντάδα με τα υψηλότερα σκορ και δεύτερον να έχει τερματίσει και τα 12 επίπεδα του παιχνιδιού. Σε περίπτωση που δύο άτομα ισοψηφούν στο σκορ τότε η διαφορά τους καλύπτεται με βάση τον χρόνο που έκαναν.



Εικόνα 31: Πίνακας με ονόματα των πέντε υψηλότερων σκορ.

3.4 Επίπεδα παιχνιδιού - διδακτικοί στόχοι

3.4.1 Γενικά χαρακτηριστικά

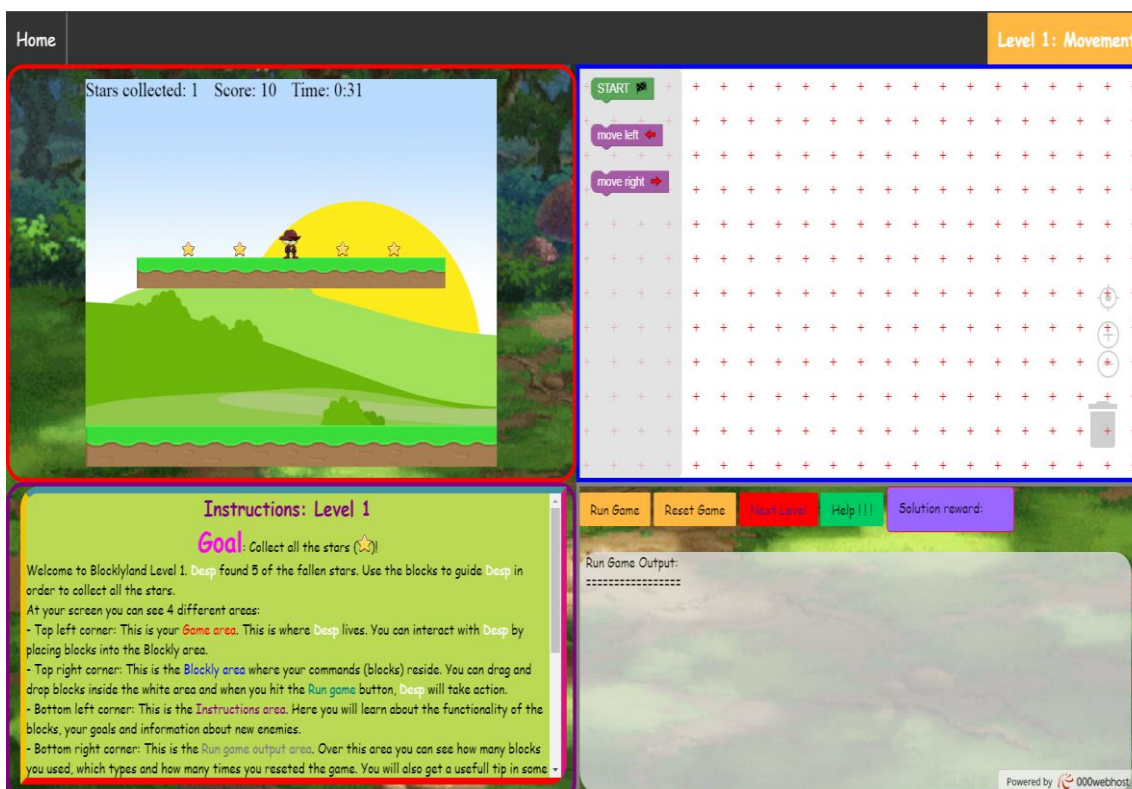
Το παιχνίδι αποτελείται από 12 επίπεδα συνολικά. Για να κερδίσει ο παίκτης το αντίστοιχο επίπεδο θα πρέπει να μαζέψει όλα τα διαθέσιμα αστέρια που εμφανίζονται σε εκείνο. Σε μερικά επίπεδα υπάρχουν και πολύχρωμα αιωρούμενα μπαλόνια τα οποία επίσης πρέπει να συλλέξει ο χρήστης για να κερδίσει το επίπεδο. Επιπλέον υπάρχουν και δευτερεύοντες στόχοι στο παιχνίδι όπως είναι η αποφυγή εμποδίων και αντιπάλων τους οποίους πρέπει ο χρήστης να αντιμετωπίσει για να συλλέξει τα αστέρια ή/και διαθέσιμα μπαλόνια.

Για κάθε αστέρι/μπαλόνι που συλλέγει ο εξερευνητής το σκορ αυξάνεται κατά 10 πόντους. Κάθε φορά που ο χρήστης επανεκκινεί το επίπεδο χάνει 10 πόντους. Οι πόντοι που μπορεί να χάσει ο παίκτης σε ένα επίπεδο ισοδυναμούν με του συνολικούς πόντους που μπορεί να κερδίσει στο συγκεκριμένο επίπεδο. Για παράδειγμα εάν τα αστέρια είναι 6 τότε ο παίκτης μπορεί να χάσει μέχρι και 60 πόντους στο συγκεκριμένο επίπεδο. Εξαίρεση στο σύστημα αφαίρεσης πόντων αποτελεί το πρώτο επίπεδο το οποίο είναι επίπεδο εξοικείωσης του χρήστη με το περιβάλλον.

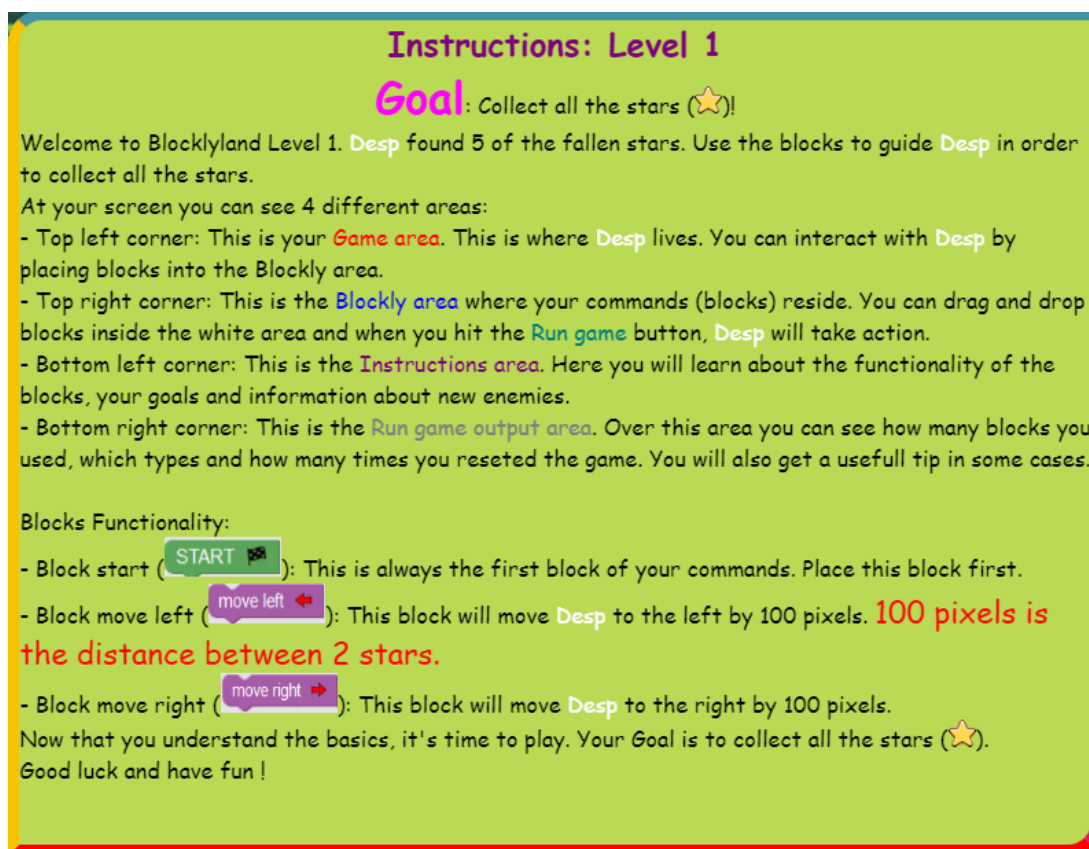
Επιπλέον το σκορ που κερδίζει ο παίκτης ανά επίπεδο συνεχίζει και στο επόμενο επίπεδο. Το σκορ που κερδίζει ανά επίπεδο αποθηκεύεται στη βάση δεδομένων μετά τη εκτέλεση (Run Game) του επιπέδου. Επομένως το σκορ που θα αποθηκευτεί στη βάση δεδομένων είναι το αποτέλεσμα της τελευταίας εκτέλεσης του χρήστη. Ο λόγος που γίνεται αυτό είναι επειδή το παιχνίδι επιτρέπει στον χρήστη σε περίπτωση που δεν έχει βρει τη βέλτιστη λύση να ξαναπροσπαθήσει με σκοπό να κερδίσει περισσότερους πόντους.

Τελευταίο σημαντικό χαρακτηριστικό είναι ο χρόνος λύσης του επιπέδου. Η βαθμολόγηση των χρηστών στον πίνακα των καλύτερων σκόρερ υπολογίζεται με βάση το τελικό σκορ τους, ωστόσο σε περίπτωση ισοψηφίας η διαφορά μεταξύ των δύο ή παραπάνω παικτών καθορίζεται με βάση τον συνολικό χρόνο επίλυσης όλων των επιπέδων του καθενός.

3.4.2 Πρώτο επίπεδο παιχνιδιού - Movement



Εικόνα 32: Οθόνη πρώτου επιπέδου BlocklyScript




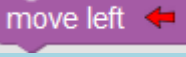
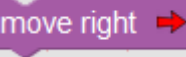
Εικόνα 33: Οδηγίες πρώτου επιπέδου

Το πρώτο επίπεδο είναι αφοσιωμένο στην εξοικείωση του παίκτη με το περιβάλλον. Γενικά το πρώτο βήμα που θα πρέπει να κάνει ο παίκτης είναι να διαβάσει την περιοχή με τις οδηγίες που βρίσκεται στο κάτω αριστερό μέρος της οθόνης. Στη συγκεκριμένη περιοχή παρέχονται οι στόχοι του επιπέδου, ένα μήνυμα καλωσορίσματος στο αντίστοιχο επίπεδο, μια περιγραφή για τους νέους αντιπάλους, εμπόδια και γενικότερες έννοιες και τέλος μια περιγραφή για τη λειτουργικότητα των μπλοκ μαζί με παραδείγματα όπου κρίνεται αναγκαίο. Για την ευκολότερη κατανόηση των εννοιών χρησιμοποιούνται ανάλογες εικόνες και χρώματα καθώς επίσης και το όνομα του χρήστη το οποίο αναφέρεται στον εξερευνητή του παιχνιδιού.

Με βάση αυτά οι πληροφορίες που εμφανίζονται στην περιοχή των οδηγιών του πρώτου επιπέδου περιγράφονται στον Πίνακα 6.

Πίνακας 6: Στόχοι, έννοιες και μπλοκ πρώτου επιπέδου

Στόχοι παιχνιδιού	
Στόχοι	<ul style="list-style-type: none"> - Γίνεται μια σύντομη ξενάγηση στις τέσσερις περιοχές που υπάρχουν σε κάθε επίπεδο του BlocklyScript - Ο χρήστης μαθαίνει πώς να μετακινεί τον παίκτη δεξιά και αριστερά με τη βοήθεια των μπλοκ καθώς και πόση απόσταση διανύει ο χαρακτήρας ανά μπλοκ μετακίνησης δεξιά ή αριστερά
Διδακτικός στόχος	Κατανόηση της ακολουθιακής εκτέλεσης των εντολών σε μορφή μπλοκ
Έννοιες	
Περιβάλλον	<p>Επισημαίνονται με 4 διαφορετικά χρώματα οι 4 σημαντικές περιοχές στο παράθυρο του παιχνιδιού. Αυτές είναι οι έξι:</p> <ul style="list-style-type: none"> - Πάνω αριστερή γωνία: Αυτή είναι η περιοχή του παιχνιδιού - Πάνω δεξιά γωνία: Αυτή είναι η περιοχή της Blockly η οποία περιέχει τις διαθέσιμες εντολές σε μορφή μπλοκ. - Κάτω αριστερή γωνία: Αυτή είναι η περιοχή των οδηγιών.

	- Κάτω δεξιά γωνία: Αυτή είναι η περιοχή εμφάνισης των αποτελεσμάτων εκτέλεσης του προγράμματος
Μπλοκ	
 Start Block	Αυτό είναι πάντα το πρώτο μπλοκ εντολών. Ο χρήστης πρέπει να τοποθετήσει το συγκεκριμένο μπλοκ πρώτο στην περιοχή εργασίας της Blockly.
 Move Left Block	Αυτό το μπλοκ μετακινεί τον εξερευνητή κατά 100 pixel αριστερά. 100 pixel είναι η απόσταση μεταξύ δύο αστεριών.
 Move Right Block	Αυτό το μπλοκ μετακινεί τον εξερευνητή κατά 100 pixel δεξιά.

Στην περιοχή των οδηγιών αναλύονται σε μεγαλύτερο βάθος οι έννοιες του πίνακα με στόχο να εξηγήσουν όσο τον δυνατόν πιο εύκολα τη λειτουργία των περιοχών. Αρκετό ενδιαφέρον, ωστόσο, παρουσιάζουν η περιοχή Blockly και η περιοχή εκτέλεσης και αποσφαλμάτωσης.

Όπως αναφέρθηκε στον πίνακα η περιοχή Blockly περιλαμβάνει τα διαθέσιμα μπλοκ εντολών. Στο συγκεκριμένο σημείο υπάρχουν δύο περιοχές, η εργαλειοθήκη του συντάκτη Blockly και η περιοχή εργασίας αυτού. Ο χρήστης για να δημιουργήσει τον δικό του αλγόριθμο θα πρέπει να σύρει, να αφήσει και να συνδέσει κατάλληλα τα μπλοκ από την εργαλειοθήκη στην επιφάνεια εργασίας. Μόλις ο αλγόριθμός του είναι έτοιμος τότε μπορεί να πατήσει το κουμπί Run Game που βρίσκεται στην περιοχή της εκτέλεσης. Με το που πατήσει το συγκεκριμένο κουμπί θα δει τον χαρακτήρα του να ανταποκρίνεται στις εντολές του αλγορίθμου του με τη σειρά που ο ίδιος τις τοποθέτησε. Επιπλέον, η περιοχή εκτέλεσης και αποσφαλμάτωσης περιέχει το κουμπί της επανεκκίνησης του επιπέδου, το κουμπί για το επόμενο επίπεδο, το κουμπί για βοήθεια, ένα πλαίσιο ανταμοιβής για τη λύση του χρήστη και τέλος το πλαίσιο Program Output που περιέχει χρήσιμες συμβουλές μετά την εκτέλεση του αλγορίθμου. Το κουμπί της επανεκκίνησης αναλαμβάνει την επανεκτέλεση του επιπέδου στην περιοχή του παιχνιδιού επαναφέροντας το στην αρχική του κατάσταση. Το κουμπί για το επόμενο επίπεδο παραμένει απενεργοποιημένο μέχρι ο χρήστης να λύσει για πρώτη φορά το επίπεδο. Στη συνέχεια το κουμπί για τη βοήθεια επιτρέπει στον χρήστη να διαβάσει τις

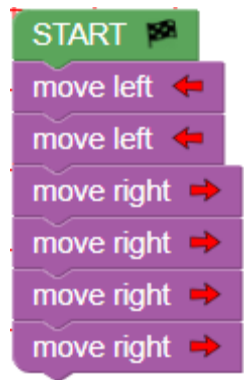
έννοιες και τη λειτουργικότητα των μπλοκ μέχρι και το τρέχων επίπεδο που βρίσκεται. Ακόμη πέρα από το σκορ του παίκτη το παιχνίδι προσφέρει και ένα πλαίσιο ανταμοιβής. Στο πλαίσιο αυτό ανάλογα με τη λύση του χρήστη για το αντίστοιχο επίπεδο εμφανίζεται η ανάλογη κονκάρδα (badge) ανταμοιβής. Οι κονκάρδες ανταμοιβής χωρίζονται σε τρεις κατηγορίες, χρυσή, ασημένια και χάλκινη και αντιστοιχούν στην βέλτιστη, στην απλή και στην αποτυχημένη λύση αντίστοιχα.



Εικόνα 34: Κουμπί βοήθειας πρώτου επιπέδου

Το τελευταίο πλαίσιο είναι το Program Output το οποίο εμφανίζει τους ελέγχους μετά την εκτέλεση του αλγορίθμου. Οι έλεγχοι κατηγοριοποιούνται σε 2 ομάδες. Αν ο παίκτης κατάφερε να μαζέψει όλα τα αστέρια/μπαλόνια τότε έχει νικήσει, διαφορετικά έχει χάσει το επίπεδο και πρέπει να ξαναπροσπαθήσει.

Σε κάθε επίπεδο και σε κάθε ομάδα εμφανίζεται μετά από κάθε "Run Game" πόσα μπλοκ χρησιμοποίησε ο χρήστης από κάθε είδος που είχε διαθέσιμο στο τρέχον επίπεδο. Επίσης εμφανίζεται πόσες φορές έχει κάνει επανεκκίνηση το επίπεδο αλλά και πόσους πόντους θα χάσει από τις επαναλήψεις του. Έπειτα ανάλογα με το σενάριο που έτρεξε γίνονται οι έλεγχοι που παρουσιάζονται στον Πίνακα 7.



Εικόνα 35: Βέλτιστη λύση πρώτου επιπέδου

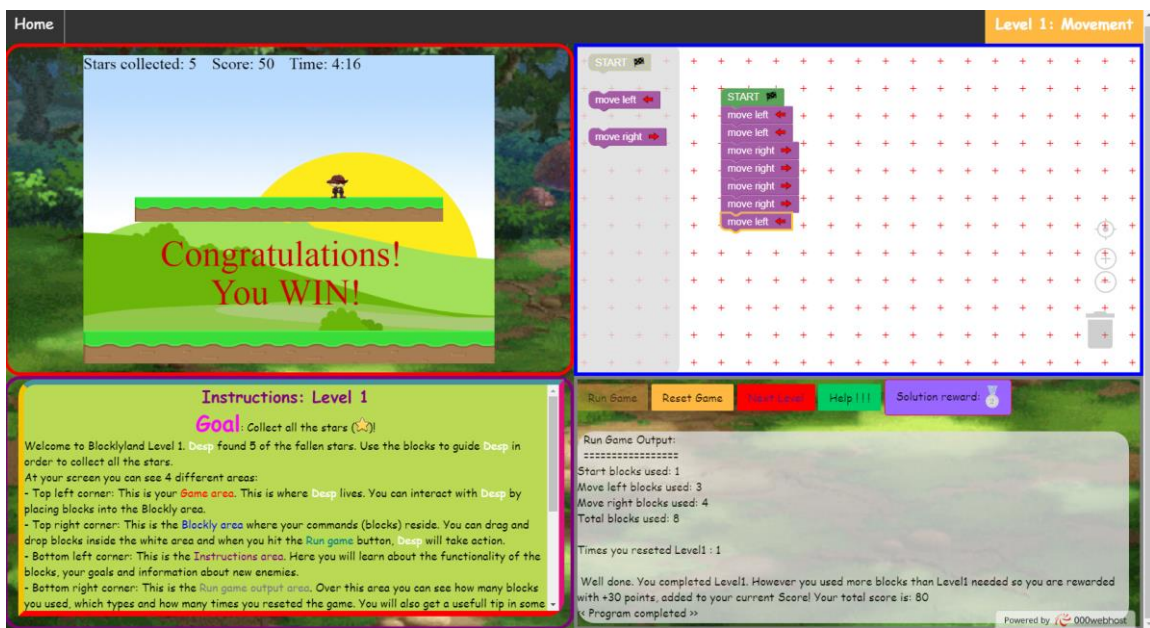
Σημείωση: Η μεταβλητή `totalBlocksUsed` μετράει το πλήθος των μπλοκ που χρησιμοποιήθηκε ο χρήστης στην αντίστοιχη προσπάθεια του.

Πίνακας 7: Έλεγχοι περιοχής Program Output πρώτου επιπέδου

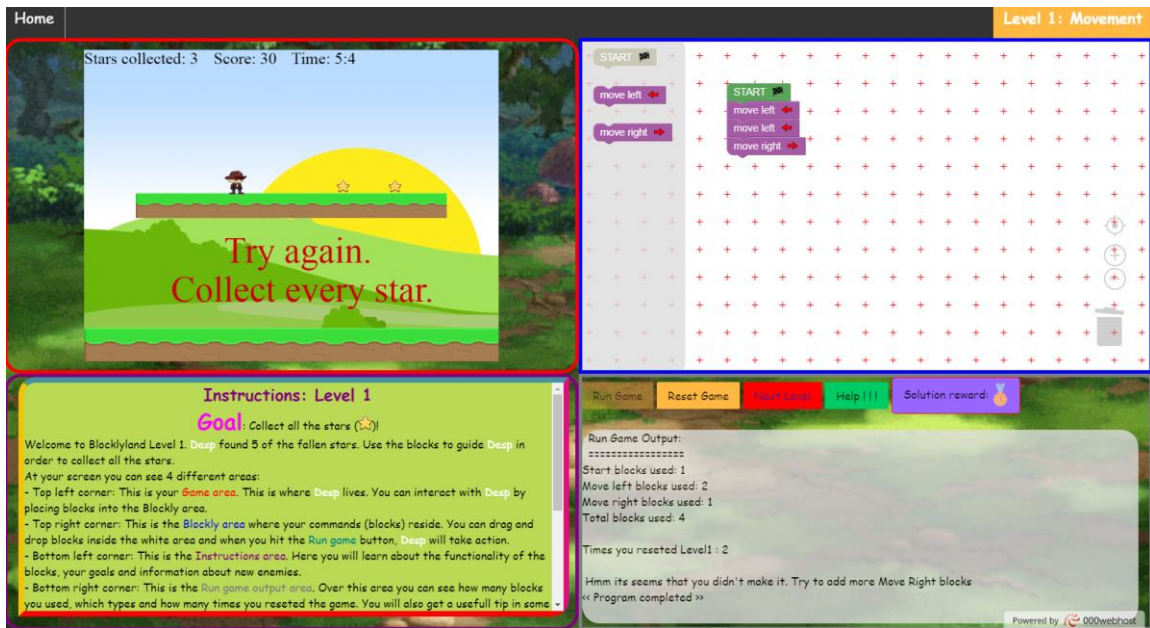
Νίκη	
<u>Βέλτιστη λύση:</u> if(totalBlocksUsed == 7)	Congratulations. You used the minimum number of blocks in order to complete Level 1. You are rewarded with +50 points, added to your current Score! Your total score is: 100
else if(totalBlocksUsed > 7)	Well done. You completed Level1. However you used more blocks than Level1 needed so you are rewarded with +30 points, added to your current Score! Your total score is: 80
Ήττα	
if(moveRight < 2)	Hmm it seems that you didn't make it. Try to add more Move Right blocks
else if(moveLeft < 2)	Hmm it seems that you didn't make it. Try to add more Move Left blocks
else	Hmm it seems that you didn't make it. Try again, you are almost there



Εικόνα 36: Παράδειγμα βέλτιστης λύσης πρώτου επιπέδου

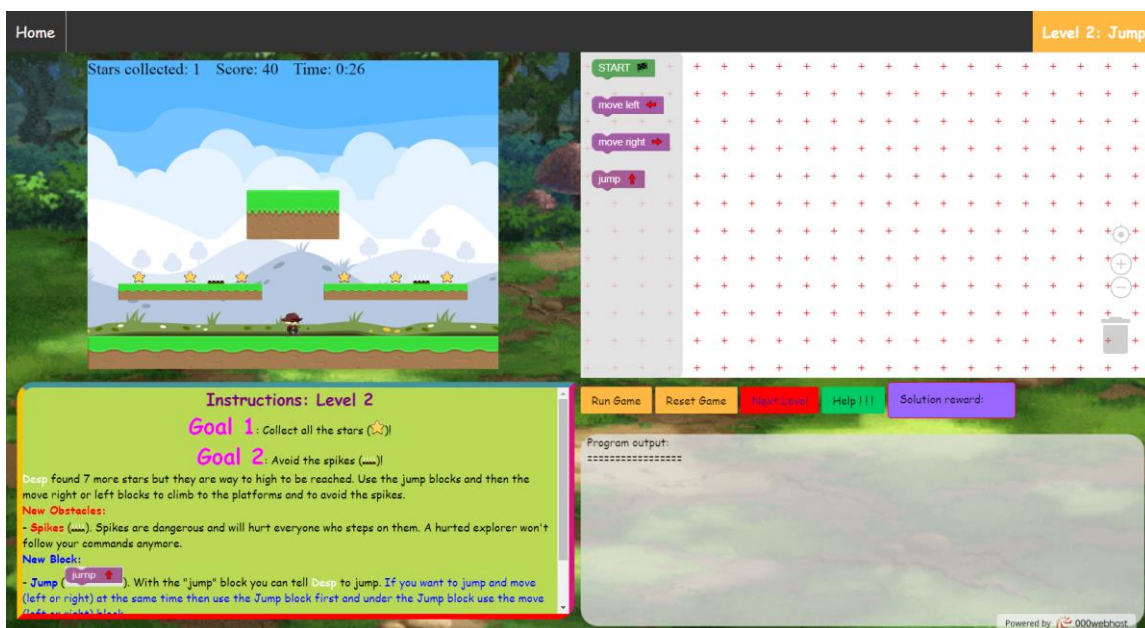


Εικόνα 37: Παράδειγμα απλής λύσης πρώτου επιπέδου

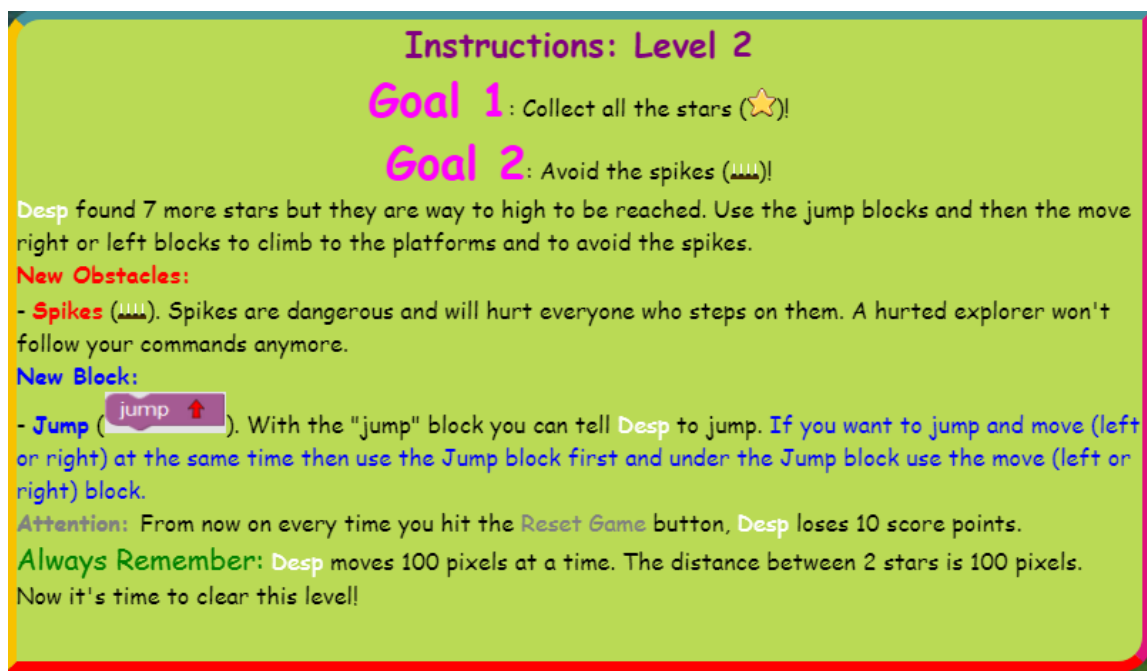


Εικόνα 38: Παράδειγμα αποτυχημένης προσπάθειας πρώτου επιπέδου

3.4.3 Δεύτερο επίπεδο παιχνιδιού - Jump



Εικόνα 39: Οθόνη δεύτερου επιπέδου

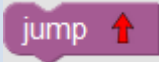


Εικόνα 40: Οδηγίες δεύτερου επιπέδου

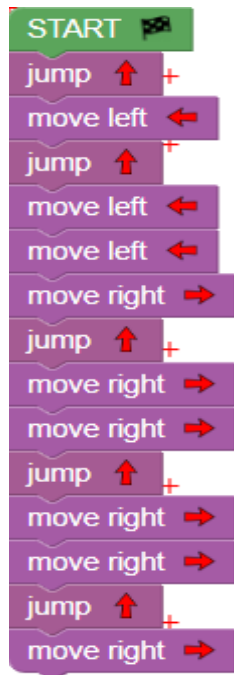
Στο δεύτερο επίπεδο του παιχνιδιού ο παίκτης μαθαίνει την εντολή του άλματος. Η εντολή αυτή είναι αρκετά χρήσιμη για τον χαρακτήρα διότι θα τον βοηθάει από το δεύτερο επίπεδο και έπειτα να ξεφεύγει εμπόδια και να πηδάει στο έδαφος που βρίσκεται σε ανώτερο επίπεδο. Για να μπορέσει να συνδυάσει την εντολή του άλματος με την κίνηση του χαρακτήρα δεξιά η αριστερά ο χρήστης θα πρέπει να τοποθετήσει πρώτα το

μπλοκ του άλματος και έπειτα την κίνηση δεξιά ή αριστερά. Επίσης σε περίπτωση που ο χρήστης τοποθετήσει την μια εντολή "Jump" κάτω από την άλλη τότε θα εκτελεστεί μόνο η πρώτη για όση ώρα βρίσκεται ο χαρακτήρας στον αέρα.

Πίνακας 8: Στόχοι, έννοιες και μπλοκ δεύτερου επιπέδου

Στόχοι παιχνιδιού	
Στόχος	Ο χρήστης μαθαίνει πώς να διατάζει τον χαρακτήρα του να κάνει άλμα έτσι ώστε να ξεφεύγει εμπόδια και να ανεβαίνει σε πλατφόρμες
Διδακτικοί Στόχοι	Κατανόηση της ακολουθιακής εκτέλεσης των εντολών σε μορφή μπλοκ. Κατάλληλος συνδυασμός μπλοκ για την επίτευξη πιο σύνθετης λειτουργίας
Έννοιες	
 Spikes	Spikes are dangerous and will hurt everyone who steps on them. A hurted explorer won't follow your commands anymore.
Μπλοκ	
 Jump block	With the “jump” block you can tell the explorer to jump. If you want to jump and move (left or right) at the same time then use the Jump block first and under the Jump block use the move (left or right) block.

Επιπλέον από το δεύτερο επίπεδο αλλά και για τα επόμενα επίπεδα κάθε φορά που ο παίκτης εκτελεί την επανεκκίνηση του επιπέδου χάνει 10 πόντους. Οι συνολικοί πόντοι που θα χάσει ο χρήστης από το σκορ του εμφανίζονται στην περιοχή Program Output.

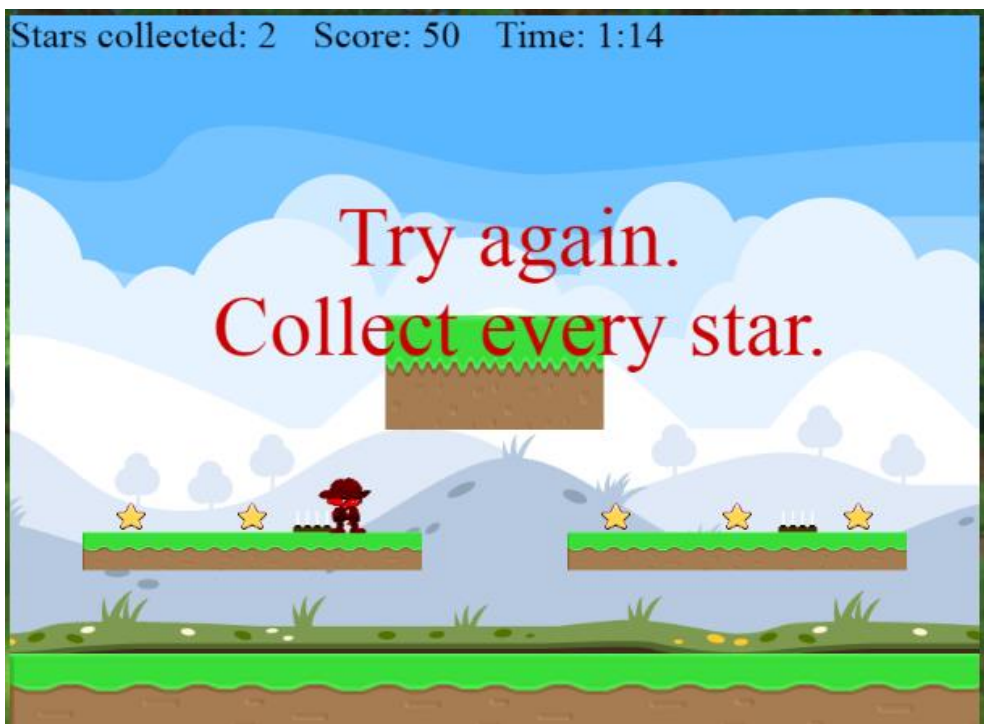


Εικόνα 41: Βέλτιστη λύση δεύτερου επιπέδου

Πίνακας 9: Έλεγχοι περιοχής Program Output δεύτερου επιπέδου

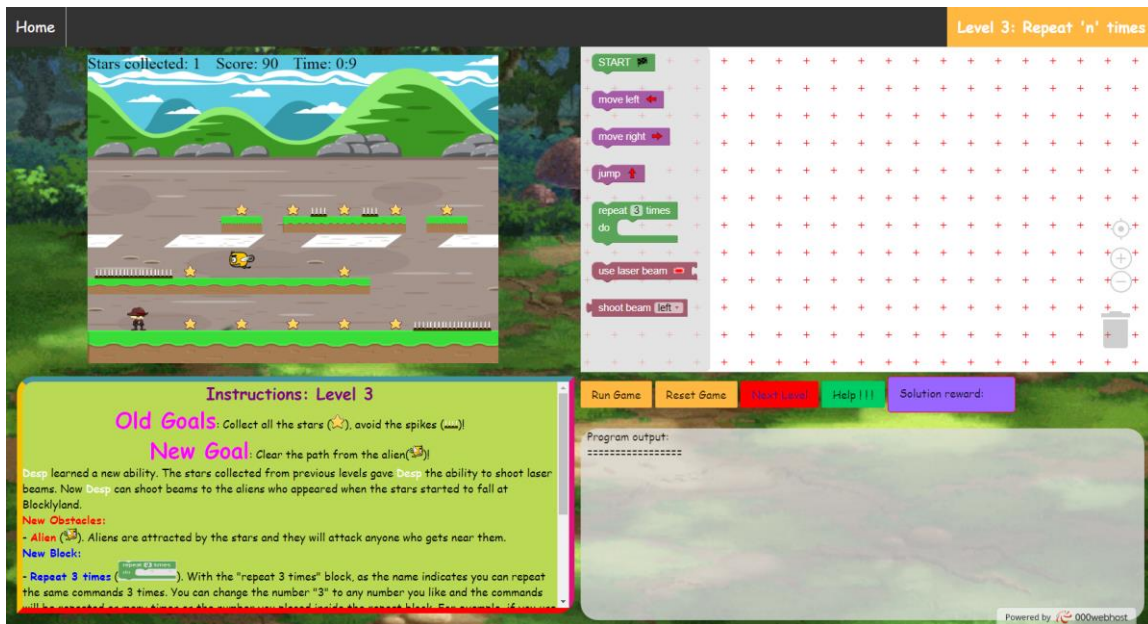
Νίκη	
<p>if(totalBlocksUsed === 15) Υπολογισμός του νέου σκορ: ((blocklyPoints + totalScore + levelScore) - Number(losePoints()))</p>	<p>Congratulations. You used the minimum number of blocks in order to complete Level 2. You are rewarded with +50 points, added to your current Score! Your total score is: '+((blocklyPoints + totalScore + levelScore) - Number(losePoints()))</p>
<p>else if(totalBlocksUsed > 15)</p>	<p>Well done. You completed Level2. However you used more blocks than Level2 needed so you are rewarded with +30 points, added to your current Score! Your total score is: '+((blocklyPoints + totalScore + levelScore) - Number(losePoints()))</p>
Ήττα	
<p>Αν ηττηθεί (γενικά μηνύματα)</p>	<p>-Remember to use the Jump block before you try to Move</p>

	<p>Left or Right</p> <p>-Don't stack two or more jump blocks under each other. The adventurer will jump only once very fast and then continue executing the rest of the commands.</p> <p>-Tip: Try to let the adventurer fall from the 1st floor ground and then use the jump block.</p>
if(moveRight < 3 moveLeft < 3)	You have to add at least 3 blocks Move Left and Move Right. Try to add more Move blocks
else if(jumpBlock < 5)	You have to add at least 5 jump blocks. Try to add more jump blocks
else	Hmm its seems that you didn't make it. Try again, you are almost there



Εικόνα 42: Περίπτωση πληγωμένου ήρωα.

3.4.4 Τρίτο επίπεδο παιχνιδιού - Repeat 'n' times



Εικόνα 43: Οθόνη τρίτου επιπέδου




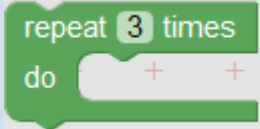
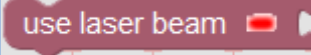
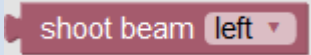
Εικόνα 44: Οδηγίες τρίτου επιπέδου

Στο τρίτο επίπεδο ο χρήστης μαθαίνει τον βρόχο επανάληψης 'for' που αποτελεί μία από τις βασικότερες έννοιες στον προγραμματισμό. Με την βοήθεια του 'for' ο χρήστης μπορεί να επαναλαμβάνει τις εντολές που επιθυμεί όσες φορές έχει δηλώσει στην αρχή του βρόχου. Στο παιχνίδι η λειτουργικότητα του βρόχου 'for' ενσωματώνεται στο μπλοκ "Repeat 'n' times". Όσα μπλοκ υπάρχουν στο σώμα του "Repeat 'n' times" θα επαναληφθούν n φορές. Έτσι ο χρήστης έχει τη δυνατότητα να μετακινεί τον εξερευνητή δεξιά ή αριστερά σε συνδυασμό με ή χωρίς άλμα χρησιμοποιώντας λιγότερα μπλοκ εντολών. Αυτό θα έχει ως αποτέλεσμα την απλοποίηση του αλγορίθμου του.

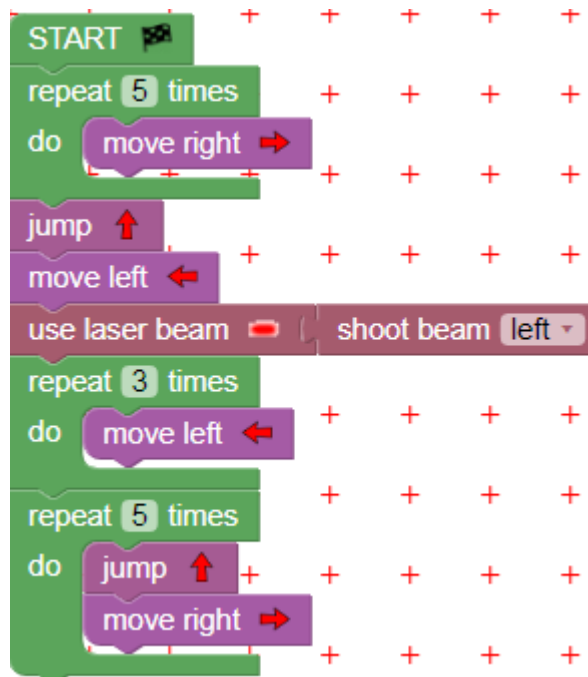
Επιπλέον στο συγκεκριμένο επίπεδο ο χρήστης αντιμετωπίζει ένα νέο αντίπαλο. Ο αντίπαλος του είναι ένας ιπτάμενος εξωγήινος ο οποίος εκλύεται από τα αστέρια και προσπαθεί να τα κρατήσει για τον ίδιο. Ο εξωγήινος είναι σε θέση να πληγώσει όποιον τον πλησιάζει και προσπαθεί να μαζέψει τα αστέρια. Για τον λόγο αυτό ο εξερευνητής θα πρέπει να χρησιμοποιήσει την ακτίνα φωτός που έμαθε μαζεύοντας τα αστέρια από τα δύο πρώτα επίπεδα για να νικήσει τον εξωγήινο και να μαζέψει όλα τα αστέρια.

Πίνακας 10: Στόχοι, έννοιες και μπλοκ τρίτου επιπέδου

Στόχοι παιχνιδιού	
Στόχοι	Ο χρήστης μαθαίνει πώς να χρησιμοποιεί τα μπλοκ "repeat 'n' times", "use laser beam" και "shoot beam 'direction'" για να κερδίσει τον εξωγήινο, να μαζέψει όλα τα αστέρια και να βρει τη βέλτιστη λύση.
Διδακτικός στόχος	Ο παίκτης έρχεται σε μια πρώτη επαφή με την προγραμματιστική έννοια του βρόχου επανάληψης "for"
Έννοιες	

 <p>Alien</p>	<p>Aliens are attracted by the stars and they will attack anyone who gets near them.</p>
<p>Μπλοκ</p>	
 <p>Μπλοκ "Repeat 'n' times"</p>	<p>With the “repeat 3 times” block, as the name indicates you can repeat the same commands 3 times. You can change the number "3" to any number you like and the commands will be repeated as many times as the number you placed inside the repeat block. For example, if you use repeat "2" times and inside the repeat block you place the “move right” block, then the explorer will move 2 stars to the right (200 pixels).</p>
 <p>Μπλοκ "use laser beam"</p>	<p>With this block you prepare the explorer to shoot a laser beam.</p>
 <p>Μπλοκ "shoot beam 'direction'"</p>	<p>With this block you can guide the explorer at which direction to shoot the laser beam.</p>

Έπειτα στην περιοχή Program Output γίνονται οι έλεγχοι του Πίνακα 11.



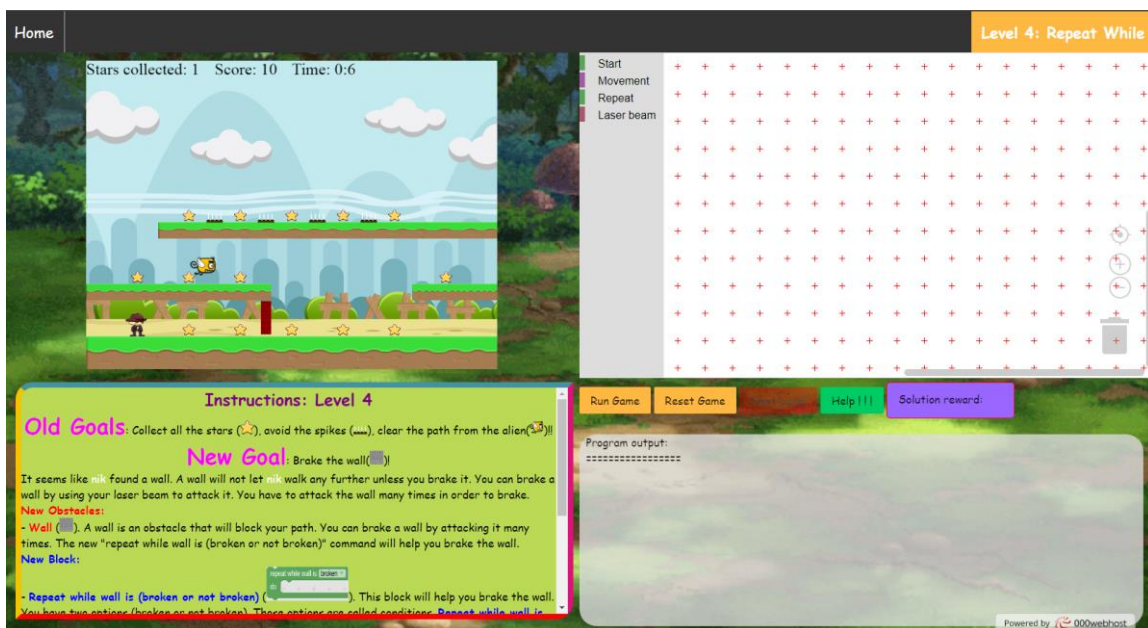
Εικόνα 45: Βέλτιστη λύση τρίτου επιπέδου

Πίνακας 11: Έλεγχοι περιοχής Program Output τρίτου επιπέδου

Νίκη	
if(totalBlocksUsed == 12)	<p>Congratulations. You used the minimum number of blocks in order to complete Level 3. You are rewarded with +100 points, added to your current Score! Your total score is: '+((blocklyPoints + totalScore + levelScore) - Number(losePoints()))</p>
else if(totalBlocksUsed > 12)	<p>-Tip: If you used two or more blocks of the same type under each other, then you should use a repeat N times block to earn more points</p> <p>-Well done. You completed Level 3. However you used more blocks than Level 3 needed so you are rewarded with +40 points, added to your current Score! Your total score is: '+((blocklyPoints + totalScore + levelScore) - Number(losePoints()))</p>
Ήττα	
Αν ηττηθεί (γενικά)	<p>Tip: If you used two or more blocks of the same type under each other, then you should place that type of block 1 time inside a</p>

μηνύματα)	repeat N times block
if(repeatNTimes < 3)	You may have stacked blocks of the same type under each other. Use that type of block 1 time inside repeat N times block
else if(useLaserBeam == 0 shootBeamDir== 0)	In order to clear the path from the alien you have to use your laser beam and aim it at the right direction
else if(jumpBlock < 3)	You have to use the jump block at least 3 times in order to win this Level
else	Hmm its seems that you didn't make it. Try again, you are almost there

3.4.5 Τέταρτο επίπεδο παιχνιδιού - Repeat While



Εικόνα 46: Οθόνη τέταρτου επιπέδου

Instructions: Level 4

Old Goals: Collect all the stars (★), avoid the spikes (🚫), clear the path from the alien (👾)!!

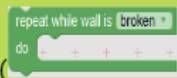
New Goal: Brake the wall(🧱)!

It seems like **nik** found a wall. A wall will not let **nik** walk any further unless you brake it. You can brake a wall by using your laser beam to attack it. You have to attack the wall many times in order to brake.

New Obstacles:

- **Wall** (🧱). A wall is an obstacle that will block your path. You can brake a wall by attacking it many times. The new "repeat while wall is (broken or not broken)" command will help you brake the wall.

New Block:



- **Repeat while wall is (broken or not broken)** (🔁). This block will help you brake the wall. You have two options (broken or not broken). Those options are called conditions. **Repeat while wall is (broken or not broken)** will keep repeating the blocks you put inside it as many times as your condition is true. For example when you start this level the wall inside the game is "not broken". If you keep attacking the wall then it will eventually brake. So you can use the "**repeat while wall is not broken**" block and inside it put the "use laser beam" block to attack the wall many times. When the wall is broken your "**repeat while wall is not broken**" block will stop repeating the blocks you used inside it .

Attention 1: Every time you hit the **Reset Game** button, **nik** loses 10 score points.

Attention 2: If you don't place any blocks inside your **Repeat while wall is (broken or not broken)** and your condition is true, then **nik** will keep going left and right with out to know what to do.

Always Remember: **nik** moves 100 pixels at a time. The distance between 2 stars is 100 pixels.

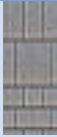
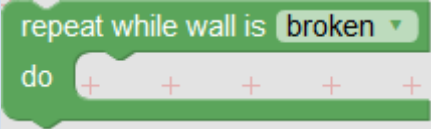
Now it's time to clear this level!

Εικόνα 47: Οδηγίες τέταρτου επιπέδου

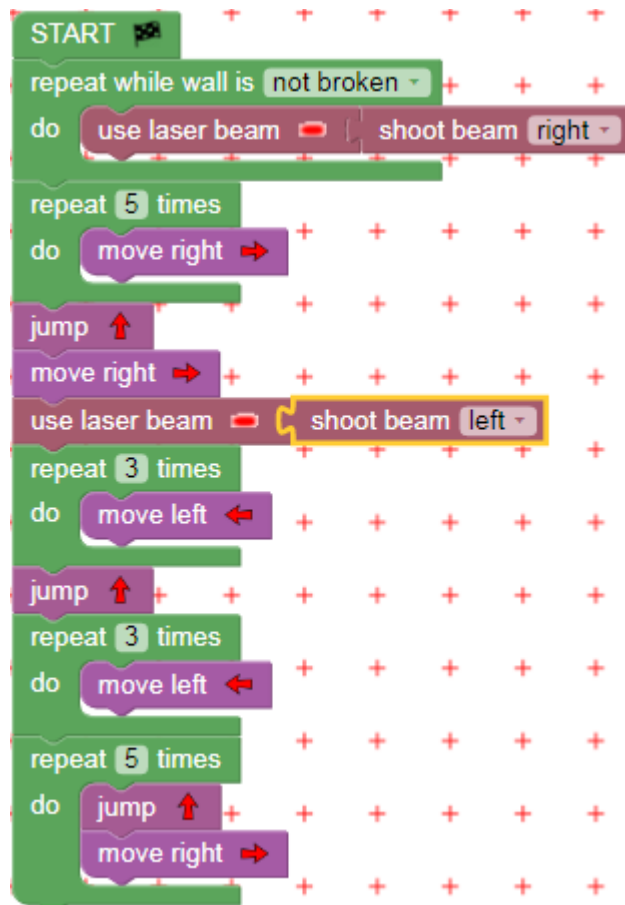
Μια ακόμη πολύ σημαντική έννοια στον προγραμματισμό είναι ο βρόχος επανάληψης "repeat while". Ο βρόχος "repeat while" χρησιμοποιείται σε περιπτώσεις που ο χρήστης δεν γνωρίζει τον συγκεκριμένο αριθμό των απαιτούμενων επαναλήψεων όπως συμβαίνει με την "for", αλλά ελέγχει μια συνθήκη. Πιο συγκεκριμένα κάθε φορά που εκτελείται το σώμα της "repeat while", στην αρχή κάθε επανάληψης ελέγχεται μια συνθήκη. Όσο η συνθήκη είναι αληθής το σώμα του βρόχου επαναλαμβάνεται. Με το που αλλάξει η κατάσταση της συνθήκης και γίνει ψευδής σταματάει αυτόματα και η εκτέλεση του βρόχου. Έπειτα το πρόγραμμα συνεχίζει από το σημείο που σταμάτησε ο βρόχος.

Στο επίπεδο η έννοια "Repeat while" υπάρχει στο μπλοκ "repeat while wall is (broken or not broken)". Η συνθήκη στον βρόχο είναι η κατάσταση του τοίχου, δηλαδή εάν έχει σπάσει ή όχι. Επομένως με τη βοήθεια αυτού του μπλοκ ο χρήστης μπορεί να σπάσει τον τοίχο που εμποδίζει το μονοπάτι του, εφόσον δεν γνωρίζει πόσες ακτίνες φωτός χρειάζονται για να σπάσει.

Πίνακας 12: Στόχοι, έννοιες και μπλοκ τέταρτου επιπέδου

Στόχοι παιχνιδιού	
Στόχος	Ο παίκτης πρέπει με τη χρήση του βρόχου επανάληψης "repeat while" να σπάσει τον τοίχο για να μαζέψει όλα τα αστέρια
Διδακτικός στόχος	Ο παίκτης μαθαίνει να χρησιμοποιεί τον βρόχο επανάληψης "repeat while"
Έννοιες	
 Wall	A wall is an obstacle that will block your path. You can brake a wall by attacking it many times. The new "repeat while wall is (broken or not broken)" command will help you brake the wall.
Μπλοκ	
 Μπλοκ "repeat while wall is '(broken or not broken)'"	This block will help you brake the wall. You have two options (broken or not broken). Those options are called conditions. Repeat while wall is (broken or not broken) will keep repeating the blocks you put inside it as many times as your condition is true. For example when you start this level the wall inside the game is "not broken". If you keep attacking the wall then it will eventually brake. So you can use the "repeat while wall is not broken" block and inside it put the "use laser beam" block to attack the wall many times. When the wall is broken your "repeat while wall is not broken" block will stop repeating the blocks you used inside it .

Στον Πίνακα 13 παρουσιάζονται οι έλεγχοι που εμφανίζονται στην περιοχή Program Output.



Εικόνα 48: Βέλτιστη λύση τετάρτου επιπέδου

Πίνακας 13: Έλεγχοι περιοχής Program Output τέταρτου επιπέδου

Νίκη	
if(totalBlocksUsed === 18)	<p>Congratulations. You used the minimum number of blocks in order to complete Level 4. You are rewarded with +100 points, added to your current Score! Your total score is: '+((blocklyPoints + totalScore + levelScore) - Number(losePoints()))</p>
else if(totalBlocksUsed > 18)	<p>-Tip: If you used two or more blocks of the same type under each other, then you should use a repeat N times block to earn more points</p> <p>-Well done. You completed Level 4. However you used more blocks than Level 4 needed so you are rewarded with +40 points, added to your current Score! Your total score is: '+((blocklyPoints + totalScore + levelScore) -</p>

	Number(losePoints()))
Ήττα	
Αν ηττηθεί (γενικά μηνύματα)	Tip: The "Repeat while wall is" block will repeat all the block commands you used inside it if the condition (broken or not broken) is true.

Επειδή από το τέταρτο επίπεδο και μετά τα παιχνίδια γίνονται πιο περίπλοκα, το ίδιο συμβαίνει και στους ελέγχους. Για τον λόγο αυτό ο έλεγχος της κάθε λύσης που δίνει ο χρήστης γίνεται μπλοκ προς μπλοκ ξεκινώντας από το πρώτο μπλοκ και κατεβαίνοντας προς τα κάτω μέχρι να τελειώσουν τα μπλοκ που έχει δώσει. Στην Εικόνα 59 η μεταβλητή blocks είναι μια λίστα στην JavaScript με τα μπλοκ που έδωσε ο χρήστης και με την σειρά που τα έδωσε. Οι έλεγχοι που γίνονται καθοδηγούν τον χρήστη προς την βέλτιστη λύση.

Με κάθε εκτέλεση του παιχνιδιού οι έλεγχοι λαμβάνουν τη λίστα με τα μπλοκ και την σειρά του κάθε μπλοκ στον αλγόριθμο. Έπειτα ξεκινώντας από το πρώτο μπλοκ και συνεχίζοντας προς τα κάτω ελέγχονται με τη σειρά τα μπλοκ που έδωσε ο χρήστης και οι τύποι των μπλοκ σε σχέση με του τύπους των μπλοκ της βέλτιστης λύσης. Εάν παρουσιαστεί ένα μπλοκ που ο τύπος του δεν είναι ίδιος με εκείνο της βέλτιστης λύσης τότε εμφανίζεται στην περιοχή Program Output το αντίστοιχο μήνυμα. Για παράδειγμα εάν ο χρήστης στο συγκεκριμένο επίπεδο σαν δεύτερο μπλοκ έβαζε οποιοδήποτε άλλο μπλοκ πέρα του "repeat while wall is (broken or not broken)" τότε θα εμφανιζόταν το μήνυμα για τη διόρθωση του δεύτερου μπλοκ. Το μήνυμα στη συγκεκριμένη περίπτωση σύμφωνα με την Εικόνα 59 θα ήταν "Block 2: Try to use a 'repeat while wall is' with the right condition." Σε περίπτωση που δεν βάλει αυτό το μπλοκ στη θέση δύο τότε θα αναγκαστεί αργότερα να χρησιμοποιήσει περισσότερα μπλοκ με αποτέλεσμα να ξεφύγει από τη βέλτιστη λύση.


```

for( var i=0; i < blocks.length; i++) {
  if( i === 0){
    if(blocks[i].type !== "start_point") {
      outputArea.value += '\nStart by using the
      start block \n';
      break;
    }
  }else if(i === 1) {
    if(blocks[i].type !== "repeat_while") {
      outputArea.value += '\nBlock '+(i + 1)+' : Try
      to use a "repeat while wall is" block with the
      right condition. \n';
      break;
    }
  }else if(i === 2) {
    if(blocks[i].type !== "laser_beam") {
      outputArea.value += '\nBlock '+(i + 1)+' : Try
      to use a "use laser beam" block inside you
      repeat while block. \n';
      break;
    }
  }else if(i === 3) {
    if(blocks[i].type !== "laser_beamdirection") {
      outputArea.value += '\nBlock '+(i + 1)+' : Did
      you give directions for the laser beam?. \n';
      break;
    }
  }else if(i === 4) {
    if(blocks[i].type !== "controls_repeat") {
      outputArea.value += '\nBlock '+(i + 1)+' : Try
      to use a repeat N times block here. This block
      should be outside of the reapeat while block \n';
      break;
    }
  }else if(i === 5) {
    if(blocks[i].type !== "move_right") {
      outputArea.value += '\nBlock '+(i + 1)+' : Try
      to use a move right block here. This block
      should be inside of the reapeat N times block\n';
      break;
    }
  }else if(i === 6) {
    if(blocks[i].type !== "move_up") {
      outputArea.value += '\nBlock '+(i + 1)+' : Try
      to use a jump block here. This block should be
      outside of the reapeat N times block\n';
      break;
    }
  }
}

```

Εικόνα 49: Έλεγχοι βέλτιστης λύσης τέταρτου επιπέδου (μέρος α)

```

}else if(i == 7) {
    if(blocks[i].type != "move_right") {
        outputArea.value += '\nBlock '+(i + 1)+' : Try
        to use a jump block here.\n';
        break;
    }
}else if(i == 8) {
    if(blocks[i].type != "laser_beam") {
        outputArea.value += '\nBlock '+(i + 1)+' : Try
        to use a use laser beam block here.\n';
        break;
    }
}
else if(i == 9) {
    if(blocks[i].type != "laser_beamdirection") {
        outputArea.value += '\nBlock '+(i + 1)+' : Did
        you give directions for the laser beam?\n';
        break;
    }
}else if(i == 10) {
    if(blocks[i].type != "controls_repeat") {
        outputArea.value += '\nBlock '+(i + 1)+' : Try
        to use a repeat N times block here.\n';
        break;
    }
}else if(i == 11) {
    if(blocks[i].type != "move_left") {
        outputArea.value += '\nBlock '+(i + 1)+' : Try
        to use a move left block inside the repeat
        block.\n';
        break;
    }
}else if(i == 12) {
    if(blocks[i].type != "move_up") {
        outputArea.value += '\nBlock '+(i + 1)+' : Try
        to use a move jump block outside the repeat
        block.\n';
        break;
    }
}else if(i == 13) {
    if(blocks[i].type != "controls_repeat") {
        outputArea.value += '\nBlock '+(i + 1)+' : Try
        to use a repeat N times block.\n';
        break;
    }
}

```

Εικόνα 50: Έλεγχος βέλτιστης λύσης τέταρτου επιπέδου (μέρος β)

```

}else if(i === 14) {
    if(blocks[i].type !== "move_left") {
        outputArea.value += '\nBlock '+(i + 1)+' : Try
        to use a move left block inside the repeat
        block.\n';
        break;
    }
}
}else if(i === 15) {
    if(blocks[i].type !== "controls_repeat") {
        outputArea.value += '\nBlock '+(i + 1)+' : Try
        to use a repeat N times block here.\n';
        break;
    }
}
}else if(i === 16) {
    if(blocks[i].type !== "move_up") {
        outputArea.value += '\nBlock '+(i + 1)+' : Try
        to use a jump block inside the repeat block.\n';
        break;
    }
}
}else if(i === 17) {
    if(blocks[i].type !== "move_right") {
        outputArea.value += '\nBlock '+(i + 1)+' : Try
        to use a move right block inside the repeat
        block.\n';
        break;
    }
}
}

}

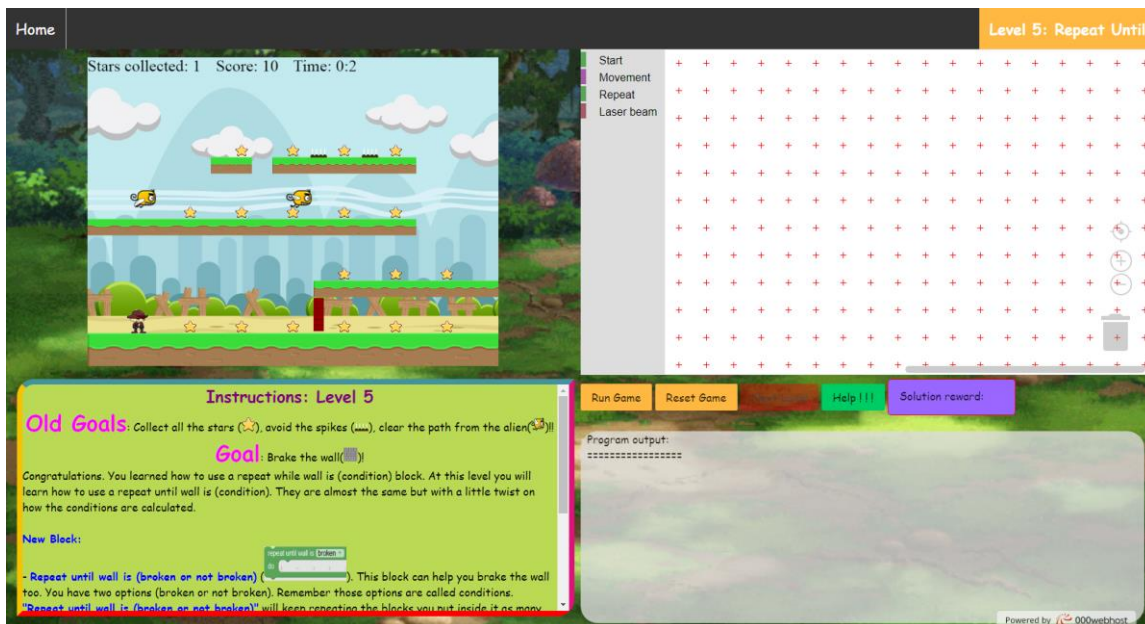
if(blocks.length < 18){
    outputArea.value += '\nIt seems like you didn t made
    it. Add more blocks in order to complete this level.\n';
}
}

```

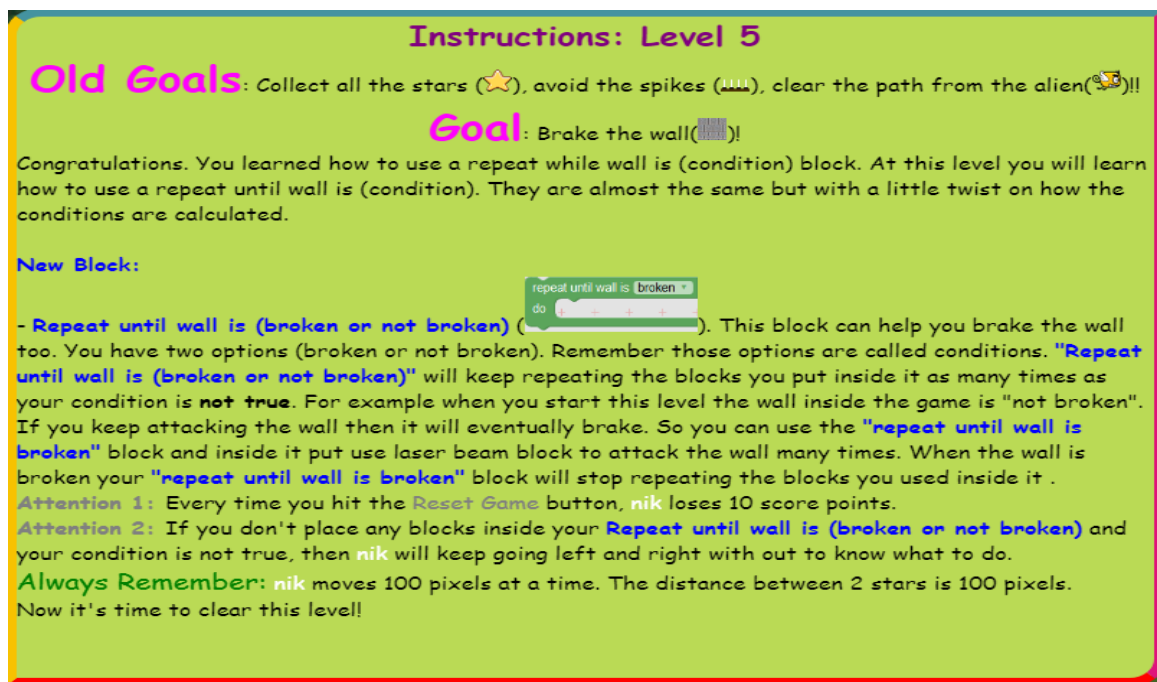
Εικόνα 51: Έλεγχος βέλτιστης λύσης τέταρτου επιπέδου (μέρος γ)

Ο τελευταίος έλεγχος if υπολογίζει την περίπτωση που ο χρήστης έδωσε μικρότερο πλήθος μπλοκ από ότι χρειάζεται για να λύσει το επίπεδο.

3.4.6 Πέμπτο επίπεδο παιχνιδιού - Repeat Until



Εικόνα 52: Οθόνη πέμπτου επιπέδου



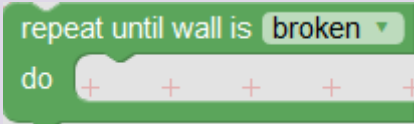
Εικόνα 53: Οθόνη οδηγιών πέμπτου επιπέδου

Στο πέμπτο επίπεδο παρουσιάζεται η προγραμματιστική έννοια "Repeat Until". Ο βρόχος επανάληψης "Repeat Until" είναι παρόμοιος με τον "Repeat While" που παρουσιάστηκε στην ενότητα 3.4.5. Όπως ο βρόχος "Repeat While" έτσι και ο βρόχος

"Repeat Until" χρησιμοποιείται για την εκτέλεση των εντολών που περιέχονται στο σώμα του και δεν είναι γνωστό το πλήθος των επαναλήψεων. Η διαφορά έγκειται στον τρόπο υπολογισμού της συνθήκης. Ο βρόχος "Repeat Until" εκτελεί το σώμα των επαναλήψεων μέχρι η συνθήκη του να γίνει αληθής ή με άλλα λόγια όσο η συνθήκη του είναι ψευδής.

Στο παιχνίδι η έννοια "Repeat until" υπάρχει στο μπλοκ "repeat until wall is (broken or not broken)". Σε αυτό το επίπεδο ο χρήστης για να κερδίσει το επίπεδο θα πρέπει να σκεφτεί ανάποδα σε σχέση με τον τρόπο σκέψης στο τέταρτο επίπεδο. Οι οδηγίες του πέμπτου επιπέδου θα τον βοηθήσουν να καταλάβει την διαφορά.

Πίνακας 14: Στόχοι, έννοιες και μπλοκ πέμπτου επιπέδου

Στόχοι επιπέδου	
Στόχος	Ο χρήστης θα πρέπει να σπάσει τον τοίχο χρησιμοποιώντας το μπλοκ "repeat until wall is (broken or not broken)" για να μαζέψει όλα τα αστέρια
Διδακτικός στόχος	Ο χρήστης θα μάθει την προγραμματιστική έννοια "Repeat Until" και τη διαφορά της με την "Repeat While"
Έννοιες	
-	
Μπλοκ	
 <p>Μπλοκ "repeat until wall is (broken or not broken)"</p>	<p>This block can help you brake the wall too. You have two options (broken or not broken). Remember those options are called conditions. "Repeat until wall is (broken or not broken)" will keep repeating the blocks you put inside it as many times as your condition is not true. For example when you start this level the wall inside the game is "not broken". If you keep attacking the wall then it will eventually brake. So you can use the "repeat until wall is broken" block and inside it put use laser beam block to attack the</p>

wall many times. When the wall is broken your “repeat until wall is broken” block will stop repeating the blocks you used inside it .

Στον Πίνακα 15 καταγράφονται οι έλεγχοι της περιοχής Program Output.



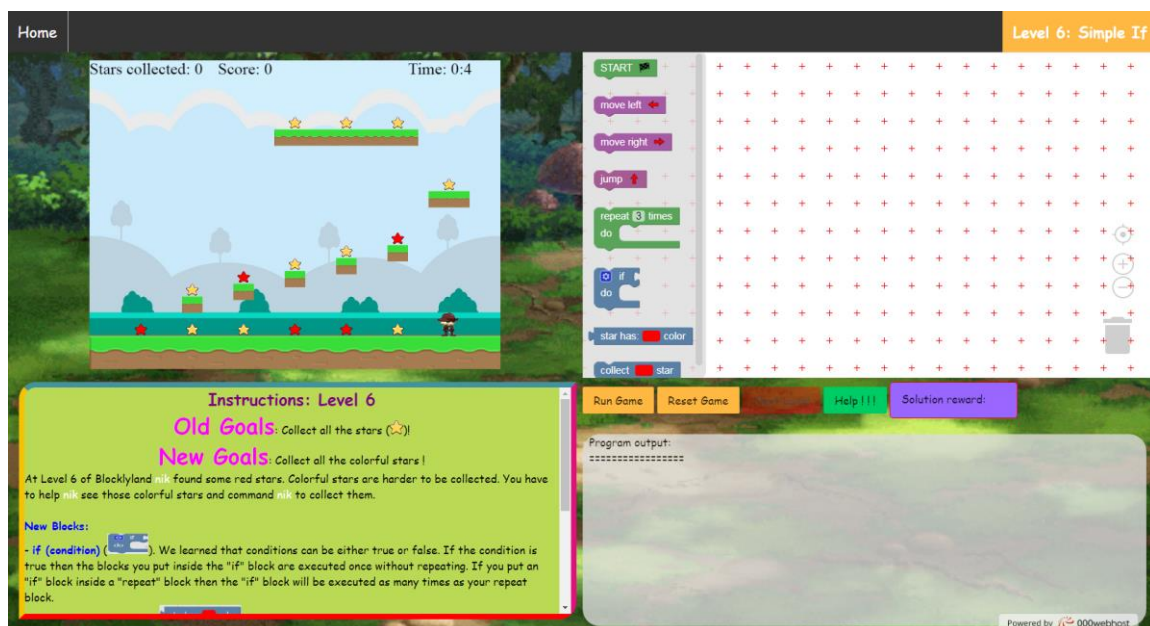
Εικόνα 54: Βέλτιστη λύση πέμπτου επιπέδου

Πίνακας 15: Έλεγχοι περιοχής Program Output πέμπτου επιπέδου

Νίκη	
if(totalBlocksUsed === 21)	Congratulations. You used the minimum number of blocks in order to complete Level 5. You are rewarded with +100 points, added to your current Score! Your total score is: '+((blocklyPoints + totalScore + levelScore) - Number(losePoints()))
else if(totalBlocksUsed	-Tip: If you used two or more blocks of the same type under each

<p>> 21)</p>	<p>other, then you should use a repeat N times block to earn more points</p> <p>-Well done. You completed Level 5. However you used more blocks than Level 5 needed so you are rewarded with +40 points, added to your current Score! Your total score is: '</p> <p>+((blocklyPoints + totalScore + levelScore) - Number(losePoints()))</p>
<p>Ήττα</p>	
<p>Αν ηττηθεί (γενικά μηνύματα)</p>	<p>Tip: The "repeat until wall is" block will repeat all the block commands you used inside it if the condition (broken or not broken) is not true.</p>
<p>Έλεγχοι βέλτιστης λύσης</p>	<p>Λόγω της μεγάλης έκτασης που θα καταλάμβαναν οι έλεγχοι για τη συγγραφή της παρούσας διπλωματικής εργασίας παραλήφθηκαν. Ωστόσο η μεθοδολογία για τον έλεγχο της βέλτιστης λύσης παρουσιάστηκε αναλυτικά στην ενότητα 3.4.5 που αφορά το τέταρτο επίπεδο. Με βάση την μεθοδολογία αυτή υπολογίζονται και οι έλεγχοι στα υπόλοιπα επίπεδα του παιχνιδιού.</p>

3.4.7 Έκτο επίπεδο παιχνιδιού - Simple if



Εικόνα 55: Οθόνη έκτου επιπέδου

Instructions: Level 6

Old Goals: Collect all the stars (★)!

New Goals: Collect all the colorful stars !

At Level 6 of Blocklyland nik found some red stars. Colorful stars are harder to be collected. You have to help nik see those colorful stars and command nik to collect them.

New Blocks:

- **if (condition)** (). We learned that conditions can be either true or false. If the condition is true then the blocks you put inside the "if" block are executed once without repeating. If you put an "if" block inside a "repeat" block then the "if" block will be executed as many times as your repeat block.
- **star has:"option" color** (). This is the **condition** of your "if" block. With this block you can check if nik steps on a red (or green or blue) star. If nik steps on the color you selected then the condition of the "if" block becomes true and the blocks you have putted inside it are executed.
- **collect "option" star** (). With this block you can tell nik to collect the color you selected. nik must be on top of the colorful star in order to be collected. Collecting colorful stars is hard and that's why you tell the adventurer to collect the colorful star at the right time only. Otherwise you will waste a large amount of time.

Attention: Every time you hit the **Reset Game** button, nik loses 10 score points.


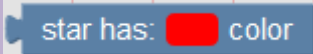
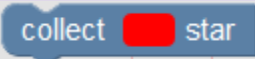
Always Remember: nik moves 100 pixels at a time. The distance between 2 stars is 100 pixels. Now it's time to clear this level!

Εικόνα 56: Οδηγίες έκτου επιπέδου

Η θεωρία που εμφανίζεται στο έκτο επίπεδο αλλάζει τη θεματολογία στις προγραμματιστικές έννοιες και από τους βρόχους περνάει στις εντολές "if". Ο χρήστης θα ξεκινήσει μαθαίνοντας τις απλές εντολές "if" για να ελέγχει τα χρώματα των αστεριών πίσω από τα οποία βρίσκεται ο εξερευνητής. Στο έκτο επίπεδο πέρα των απλών κίτρινων αστεριών υπάρχουν και τα κόκκινα αστερία, τα οποία απαιτούν περισσότερο χρόνο για να συλλεχτούν από τον εξερευνητή. Για να μπορέσει ο εξερευνητής να λύσει το επίπεδο όσον τον δυνατόν πιο γρήγορα θα πρέπει να ελέγχει το χρώμα του αστεριού και εάν το αστερί είναι κόκκινο να ζητήσει από τον εξερευνητή να το συλλέξει.

Πίνακας 16: Στόχοι, έννοιες και μπλοκ έκτου επιπέδου

Στόχοι επιπέδου	
Στόχος	Ο χρήστης θα πρέπει να κάνει τους κατάλληλους ελέγχους για να συλλέξει όλα τα αστερία όσο πιο σύντομα γίνεται
Διδακτικός στόχος	Ο χρήστης παίζοντας θα μάθει τη λειτουργικότητα της προγραμματιστικής έννοιας "Simple if statement"
Έννοιες	
Colorful stars	Colorful stars are harder to be collected. You have to help the explorer see those colorful stars and command the explorer to collect them.
Μπλοκ	

 <p>Μπλοκ "if"</p>	<p>If (condition). We learned that conditions can be either true or false. If the condition is true then the blocks you put inside the "if" block are executed once without repeating. If you put an "if" block inside a "repeat" block then the "if" block will be executed as many times as your repeat block.</p>
 <p>Μπλοκ "star has 'option' color"</p>	<p>star has:"option" color. This is the condition of your "if" block. With this block you can check if the explorer steps on a red (or green or blue) star. If the explorer steps on the color you selected then the condition of the "if" block becomes true and the blocks you have putted inside it are executed.</p>
 <p>Μπλοκ "collect 'option' star"</p>	<p>collect "option" star. With this block you can tell the explorer to collect the color you selected. The explorer must be on top of the colorful star in order to be collected. Collecting colorful stars is hard and that's why you tell the adventurer to collect the colorful star at the right time only. Otherwise you will waste a large amount of time.</p>

Στον Πίνακα 17 παρουσιάζονται οι έλεγχοι της περιοχής Program Output.

```

START
repeat 6 times
do
  move left
  if star has: color
  do collect star

repeat 6 times
do
  jump
  move right
  if star has: color
  do collect star

jump
repeat 3 times
do
  move left

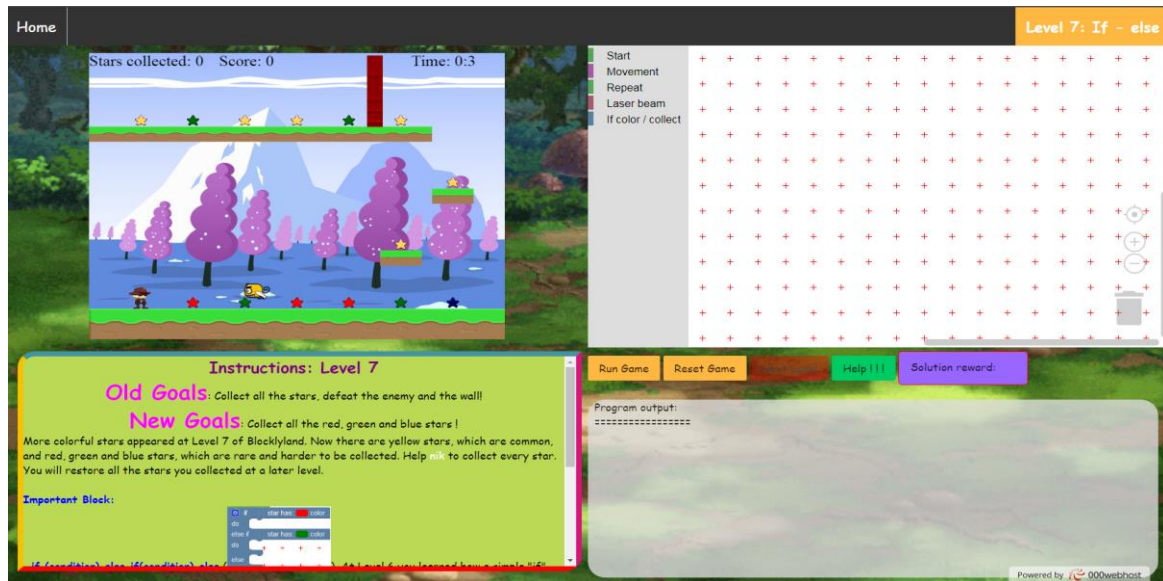
```

Εικόνα 57: Βέλτιστη λύση έκτου επιπέδου

Πίνακας 17: Έλεγχοι περιοχής Program Output έκτου επιπέδου

Νίκη	
if(totalBlocksUsed === 15)	<p>Congratulations. You used the minimum number of blocks in order to complete Level 6. You are rewarded with +100 points, added to your current Score! Your total score is: '+((blocklyPoints + totalScore + levelScore) - Number(losePoints()))</p>
else if(totalBlocksUsed > 15)	<p>-Tip: If you used two or more blocks of the same type under each other, then you should use a repeat N times block to earn more points'</p> <p>-Well done. You completed Level 6. However you used more blocks than Level 6 needed so you are rewarded with +40 points, added to your current Score! Your total score is: '+((blocklyPoints + totalScore + levelScore) - Number(losePoints()))</p>
else if(totalBlocksUsed < 15) Περίπτωση που ο χρήστης δεν έκανε τους απαιτούμενους ελέγχους	<p>Well done. You completed Level 6. You probably though didn't use the "if (condition)" block. You should try to use them in order to get more points. You are rewarded with +40, added to your current Score.'" '+((blocklyPoints + totalScore + levelScore) - Number(losePoints()))</p>
Ήττα	
Αν ηττηθεί (γενικά μηνύματα)	<p>-Tip: If (condition) block will execute all the blocks you put inside it once, if your condition is true. If your condition is false then it will skip all the blocks you put inside it.</p> <p>-Remember: The adventurer must step on the colorful star in order to collect it with the collect "option" star block</p>

3.4.8 Έβδομο επίπεδο παιχνιδιού - If-else



Εικόνα 58: Οθόνη έβδομου επιπέδου



Εικόνα 59: Οδηγίες έβδομου επιπέδου


Συνεχίζοντας με τη θεωρία στο έβδομο επίπεδο ο παίκτης μαθαίνει πώς να χρησιμοποιεί την εντολή "if-else". Συγκεκριμένα ο παίκτης πρέπει να κάνει τους κατάλληλους ελέγχους στις συνθήκες των δηλώσεων "if-else" με στόχο να μαζέψει τα πολύχρωμα αστέρια μέσα σε ένα σύντομο χρονικό διάστημα. Πλέον στην κατηγορία των πολύχρωμων αστεριών συμπεριλαμβάνονται τα κόκκινα, τα πράσινα και τα μπλε. Για να μαζέψει ο εξερευνητής ένα πολύχρωμο αστέρι πρέπει ο χρήστης με την κατάλληλη εντολή να τον βοηθήσει για να το συλλέξει. Διαφορετικά ο εξερευνητής θα το αγνοήσει και θα συνεχίσει να εκτελεί τις οδηγίες του χρήστη. Σε περίπτωση που ο χρήστης αγνοεί τους ελέγχους και μαζεύει τα αστέρια απερίσκεπτα τότε θα χάσει πολύτιμο χρόνο που θα τον βοηθούσε να κατακτήσει μια καλύτερη θέση στον πίνακα των υψηλότερων σκόρερ και θα ανταμειφτεί με λιγότερους πόντους εφόσον απέχει από τη βέλτιστη λύση.

Πίνακας 18: Στόχοι, έννοιες και μπλοκ έβδομου επιπέδου

Στόχοι επιπέδου	
Στόχος	Ο χρήστης πρέπει να κάνει του κατάλληλους ελέγχους με σκοπό να μαζέψει όλα τα αστέρια (πολύχρωμα και κανονικά)
Διδακτικός στόχος	Ο χρήστης εμβαθύνει τις γνώσεις του πάνω στη έννοια "if" μαθαίνοντας τις εντολές "if-else"
Έννοιες	
-	
Μπλοκ	

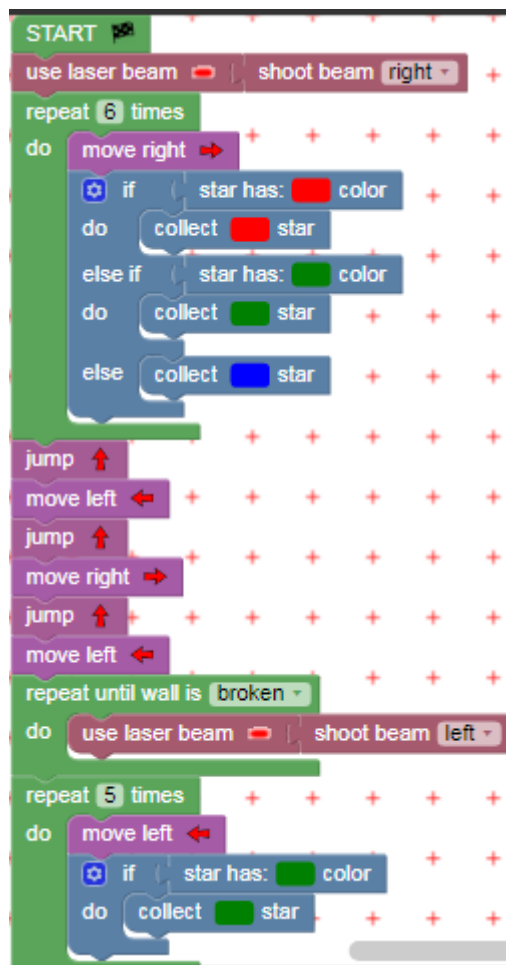


Μπλοκ "if-else"

if (condition) else if(condition) else. At Level 6 you learned how a simple "if" block works. Now it is time to create an "else if" control. If you check your "if" block you will see a little button  on it.

This button will help you expand your simple if block and check for more conditions. In our game there are 3 rare colorful types of stars (red, green, blue). Whenever the adventurer steps on a colorful rare star you have to check what color it is inside your "if" or "else if" condition and if the condition is true you will have to tell the explorer to collect the right colorful star. The explorer will start by checking the first condition of your "if-else" block. If your first condition is true then the explorer will execute the command blocks you putted inside it and ignore the rest of the "else if" conditions. On the other hand if your first condition is false, then the explorer will look at the second "else if" condition and so on.

Στον Πίνακα 19 παρουσιάζονται οι έλεγχοι της περιοχής Program Output.



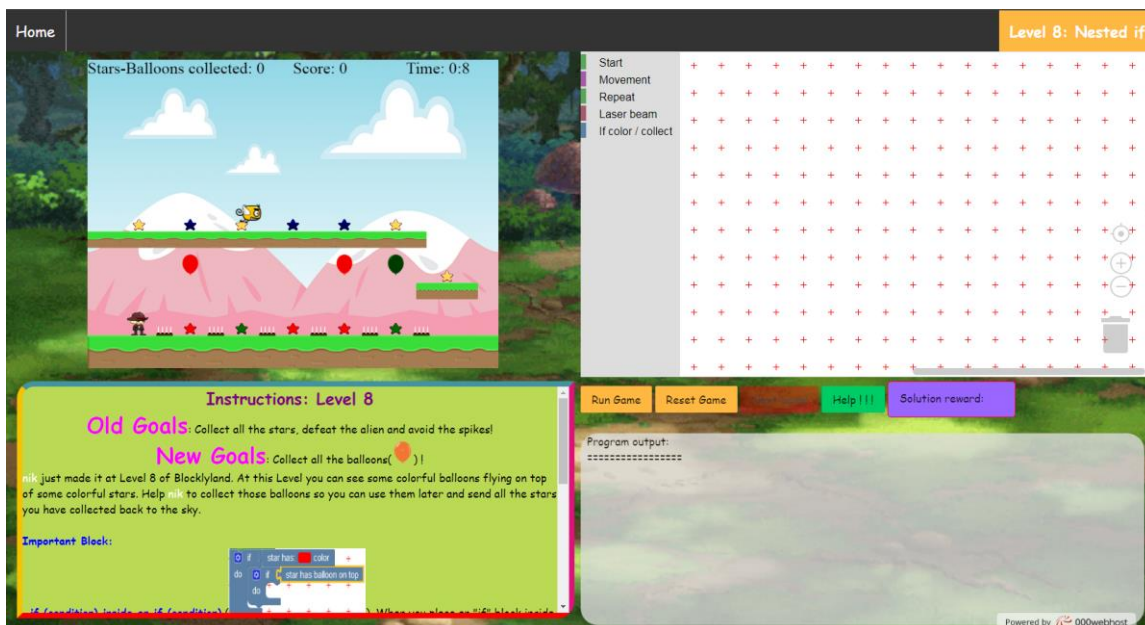
Εικόνα 60: Βέλτιστη λύση έβδομου επιπέδου

Πίνακας 19: Έλεγχοι περιοχής Program Output έβδομου επιπέδου

Νίκη	
if(totalBlocksUsed == 25)	Congratulations. You used the minimum number of blocks in order to complete Level 7. You are rewarded with +100 points, added to your current Score! Your total score is: '+((blocklyPoints + totalScore + levelScore) - Number(losePoints()))
else if(totalBlocksUsed > 25)	-Tip: If you used two or more blocks of the same type under each other, then you should use a repeat N times block to earn more points -Well done. You completed Level 7. However you used more blocks than Level 7 needed so you are rewarded with +40 points, added to your current Score! Your total score is: '

	$+((\text{blocklyPoints} + \text{totalScore} + \text{levelScore}) - \text{Number}(\text{losePoints}()))$
else if(totalBlocksUsed < 25)	<p>Well done. You completed Level 7. You probably though didn't use the "if (condition)" block, You should try to use them in order to get more points. You are rewarded with +40, added to your current Score." $+((\text{blocklyPoints} + \text{totalScore} + \text{levelScore}) - \text{Number}(\text{losePoints}()))$</p>
Ηττα	
Αν ηττηθεί (γενικά μηνύματα)	<p>-Tip: If (condition) block will execute all the blocks you put inside it once, if your condition is true. If your condition is false then it will skip all the blocks you put inside it. You can also expand your if block and drag and drop an else if and an else under your first if. That way you can check if the adventurer steps on a red, green or blue star.</p> <p>-Remember: The adventurer must step on the colorful star in order to collect it with the collect "option" star block</p>

3.4.9 Όγδοο επίπεδο παιχνιδιού - Nested if



Εικόνα 61: Οθόνη όγδοου επιπέδου

Instructions: Level 8

Old Goals: Collect all the stars, defeat the alien and avoid the spikes!

New Goals: Collect all the balloons()!

nik just made it at Level 8 of Blocklyland. At this Level you can see some colorful balloons flying on top of some colorful stars. Help nik to collect those balloons so you can use them later and send all the stars you have collected back to the sky.

Important Block:



- **if (condition) inside an if (condition)** (). When you place an "if" block inside an "if" block as the previous image shows, then the second "if" block is called **Nested if**. When you use 2 "if" blocks inside each other then you can check two conditions and if those conditions are true you can execute the blocks you put inside the do area. In our image example the first condition checks if nik is on top of a red star. If nik stands on top of a red star then the condition is true and then checks for the "Nested if" condition, which sees if the colorful star that the adventurer stands on has a balloon. The balloons have the same color as the stars beneath them.

New Blocks:

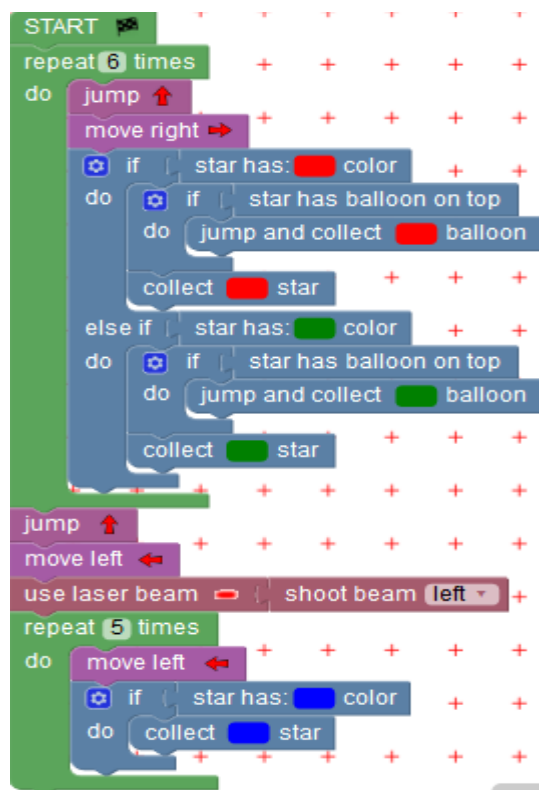
- **star has balloon on top** (). With this block you can check if the star nik stands on has a balloon on top of it. **Important: You should always collect the balloon first and then collect the star. Otherwise if you collect the star first and then check if the star has a balloon on top of it, nik will get confused and forget that there was a star there.**
- **jump and collect option balloon** (). With this block nik can jump and collect the colorful balloon of your choice

Attention: Every time you hit the **Reset Game** button, nik loses 10 score points.

Always Remember: nik moves 100 pixels at a time. The distance between 2 stars is 100 pixels.
Now it's time to clear this level!


Εικόνα 62: Οδηγίες όγδοου επιπέδου

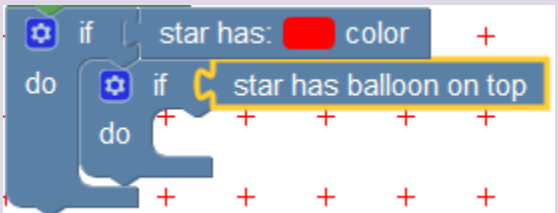
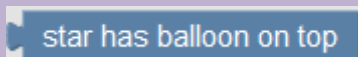

Στο όγδοο επίπεδο παρουσιάζεται η έννοια της εμφωλευμένης εντολής if. Στην περιοχή του παιχνιδιού ο χρήστης καλείται ,πέρα από τους ελέγχους για τα πολύχρωμα αστέρια, να ελέγχει και την ύπαρξη αιωρούμενων μπαλονιών πάνω από τα πολύχρωμα αστέρια. Τα μπαλόνια έχουν το χρώμα του αντίστοιχου αστεριού πάνω από το οποίο βρίσκονται. Μόλις ο εξερευνητής εντοπίσει ένα μπαλόνι πρέπει και να το συλλέξει μέσω της καθοδήγησης του χρήστη. Για να κερδίσει το επίπεδο ο χρήστης θα πρέπει να μαζέψει όλα τα αστέρια και τα μπαλόνια.



Εικόνα 63: Βέλτιστη λύση όγδοου επιπέδου

Πίνακας 20: Στόχοι, έννοιες και μπλοκ όγδοου επιπέδου

Στόχοι επιπέδου	
Στόχος	Ο χρήστης μέσω των κατάλληλων ελέγχων καλείται να συλλέξει όλα τα διαθέσιμα αστέρια και μπαλόνια
Διδακτικός στόχος	Ο χρήστης έρχεται σε μια πρώτη επαφή με τις εμφωλευμένες εντολές "if"
Έννοιες	
 Balloon	At this Level you can see some colorful balloons flying on top of some colorful stars. Help the explorer to collect those balloons so you can use them later and send all the stars you have collected back to the sky.
Μπλοκ	

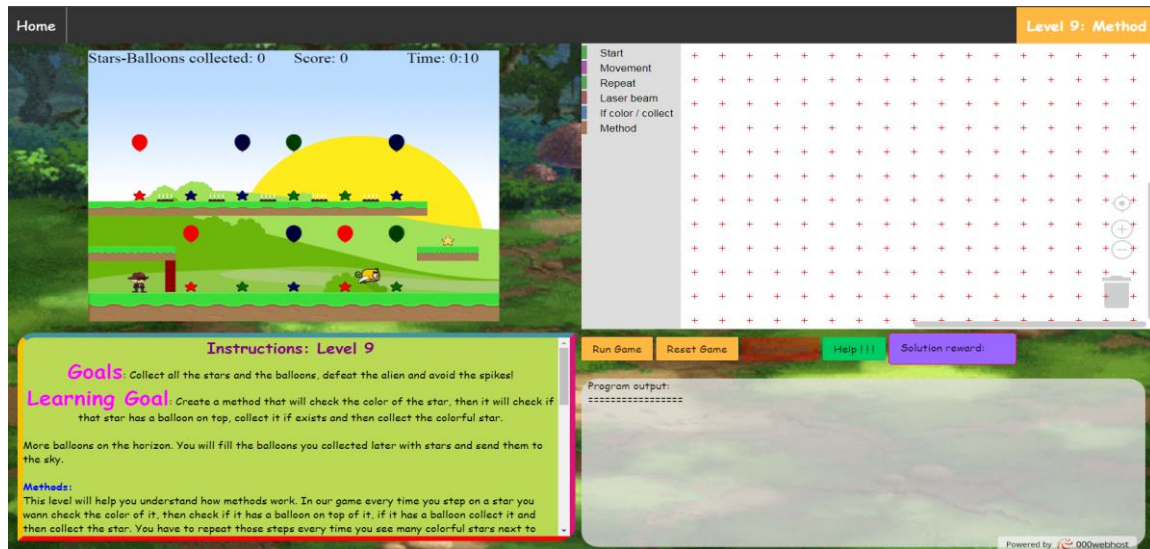
 <p>Μλοκ "Nested if"</p>	<p>if (condition) inside an if (condition). When you place an "if" block inside an "if" block as the previous image shows, then the second "if" block is called Nested if. When you use 2 "if" blocks inside each other then you can check two conditions and if those conditions are true you can execute the blocks you put inside the do area. In our image example the first condition checks if the explorer is on top of a red star. If the explorer stands on top of a red star then the condition is true and then checks for the "Nested if" condition, which sees if the colorful star that the adventurer stands on has a balloon. The balloons have the same color as the stars beneath them.</p>
 <p>Μλοκ "star has ballon on top"</p>	<p>With this block you can check if the star the explorer stands on has a balloon on top of it. Important: You should always collect the balloon first and then collect the star. Otherwise if you collect the star first and then check if the star has a balloon on top of it, the explorer will get confused and forget that there was a star there.</p>
 <p>Μλοκ "jump and collect 'option' balloon"</p>	<p>With this block the explorer can jump and collect the colorful balloon of your choice</p>

Στον Πίνακα 21 παρουσιάζονται οι έλεγχοι της περιοχής Program Output.

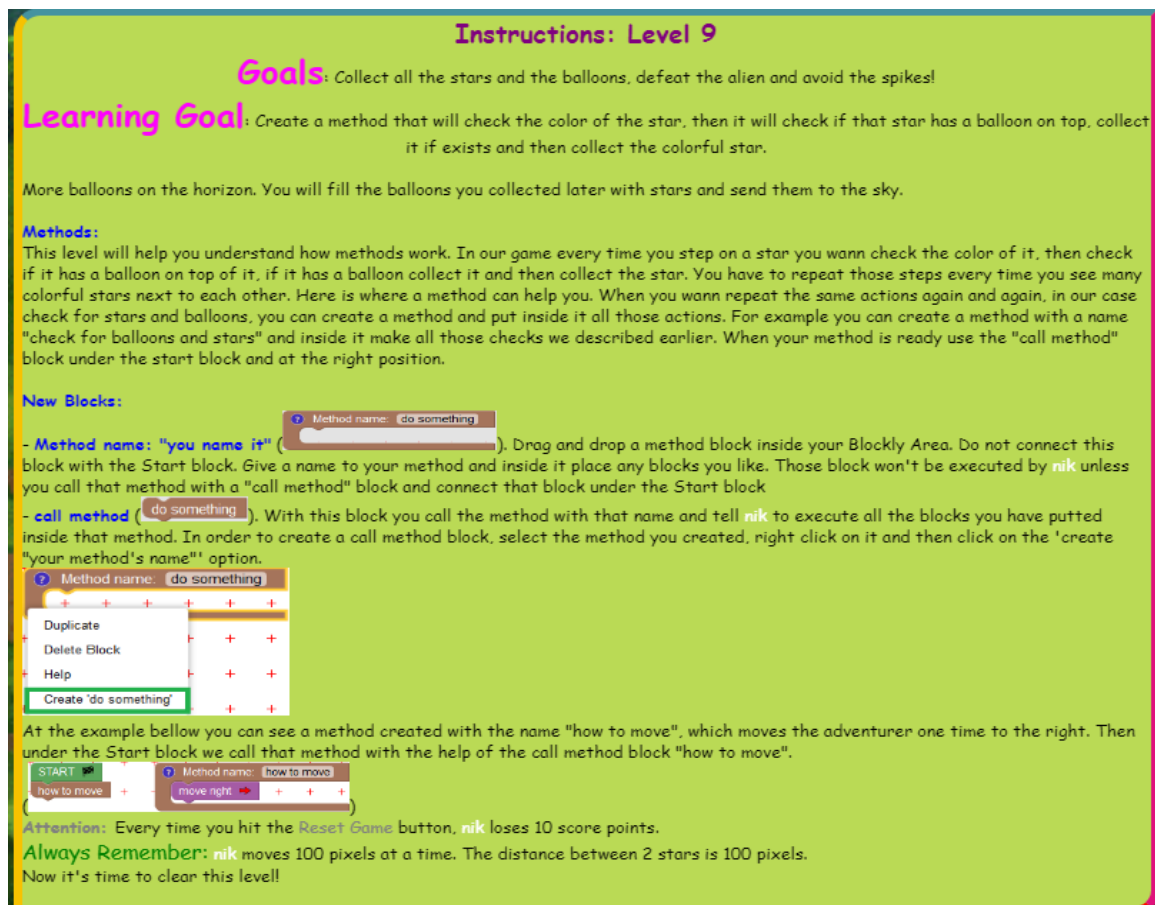
Πίνακας 21: Έλεγχοι περιοχής Program Output εβδομού επιπέδου

Νίκη	
if(totalBlocksUsed == 24)	<p>Congratulations. You used the minimum number of blocks in order to complete Level 8. You are rewarded with +100 points, added to your current Score! Your total score is: '+((blocklyPoints + totalScore + levelScore) - Number(losePoints()))</p>
else if(totalBlocksUsed > 24)	<p>-Tip: If you used two or more blocks of the same type under each other, then you should use a repeat N times block to earn more points</p> <p>-Well done. You completed Level 8. However you used more blocks than Level 8 needed so you are rewarded with +40 points, added to your current Score! Your total score is: '+((blocklyPoints + totalScore + levelScore) - Number(losePoints()))</p>
else if(totalBlocksUsed < 24)	<p>Well done. You completed Level 8. You probably though didn't use the "if"(condition) block the right way. You should try to use them in order to get more points. You are rewarded with +40, added to your current Score.'" '+((blocklyPoints + totalScore + levelScore) - Number(losePoints()))</p>
Ήττα	
Αν ηττηθεί (γενικά μηνύματα)	<p>-Tip: Check the image at the instructions which explains how nested if works. Inside your nested if block you should only use the jump and collect option balloon block. After that, use the collect option star block outside of your Nested if block and inside your first If block</p> <p>-Remember: The adventurer must step on the colorful star in order to collect it with the collect "option" star block</p>

3.4.10 Ένατο επίπεδο παιχνιδιού - Method

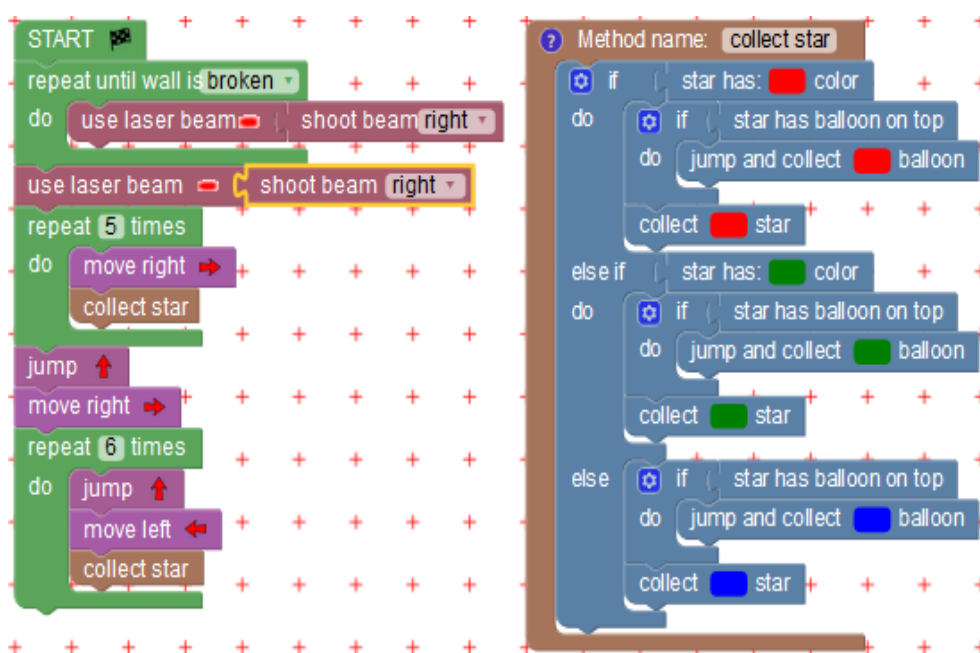


Εικόνα 64: Οθόνη ένατου επιπέδου



Εικόνα 65: Οδηγίες ένατου επιπέδου

Στο ένατο επίπεδο γίνεται μια εισαγωγή στην έννοια των μεθόδων. Ο χρήστης δημιουργώντας μια μέθοδο για τον έλεγχο και τη συλλογή των πολύχρωμων αστεριών και μπαλονιών θα μάθει δύο βασικά χαρακτηριστικά της Υπολογιστικής Σκέψης που παρουσιάστηκαν στην ενότητα 2.3. Το πρώτο είναι ο εντοπισμός του μοτίβου στο πρόβλημα προς επίλυση και το δεύτερο είναι η αφαιρετική σκέψη. Χρησιμοποιώντας αυτά τα δυο χαρακτηριστικά ο παίκτης καλείται να βρει τη λύση του επιπέδου.

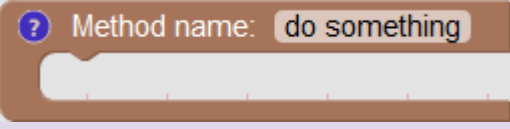
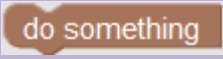
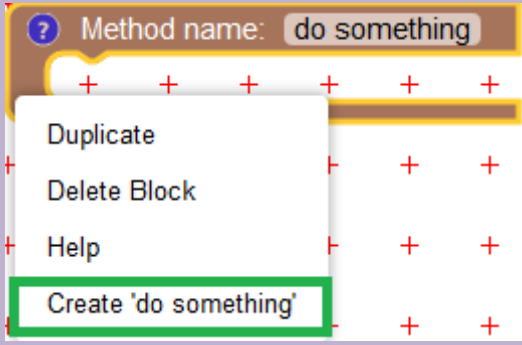


Εικόνα 66: Βέλτιστη λύση ένατου επιπέδου

Πίνακας 22: Στόχοι, έννοιες και μπλοκ ένατου επιπέδου

<p>Στόχοι επιπέδου</p>	
<p>Στόχος</p>	<p>Ο χρήστης θα πρέπει να δημιουργήσει μια μέθοδο για τον έλεγχο και τη συλλογή των αστεριών και των μπαλονιών καθώς και να καλέσει τη μέθοδο στον αλγόριθμο του.</p>

<p>Διδακτικός στόχος</p>	<p>Ο χρήστης θα έρθει σε μια πρώτη επαφή με την έννοια των μεθόδων και τη χρησιμότητά τους</p>
<p>Έννοιες</p>	
<p>Methods</p>	<p>This level will help you understand how methods work. In our game every time you step on a star you want to check the color of it, then check if it has a balloon on top of it, if it has a balloon collect it and then collect the star. You have to repeat those steps every time you see many colorful stars next to each other. Here is where a method can help you. When you want to repeat the same actions again and again, in our case check for stars and balloons, you can create a method and put inside it all those actions. For example you can create a method with a name "check for balloons and stars" and inside it make all those checks we described earlier. When your method is ready use the "call method" block under the start block and at the right position.</p>
<p>Μπλοκ</p>	

 <p>Μπλοκ δημιουργίας μεθόδου</p>	<p>Method name: "you name it". Drag and drop a method block inside your Blockly Area. Do not connect this block with the Start block. Give a name to your method and inside it place any blocks you like. Those blocks won't be executed by the explorer unless you call that method with a "call method" block and connect that block under the Start block</p>
 <p>Μπλοκ κλήσης μεθόδου</p>	<p>Call method. With this block you call the method with that name and tell the explorer to execute all the blocks you have putted inside that method. In order to create a call method block, select the method you created, right click on it and then click on the 'create "your method's name"' option.</p>  <p>Παράδειγμα δημιουργίας μπλοκ κλήσης μεθόδου</p>

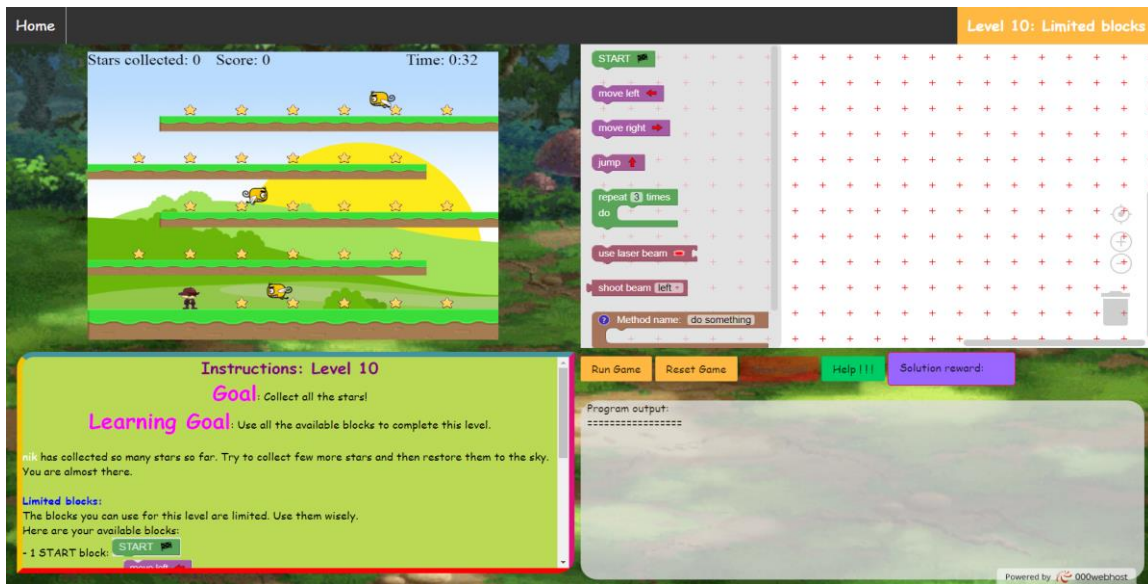
Στον Πίνακα 23 παρουσιάζονται οι έλεγχοι στην περιοχή Program Output

Πίνακας 23: Έλεγχοι περιοχής Program Output ένατου επιπέδου

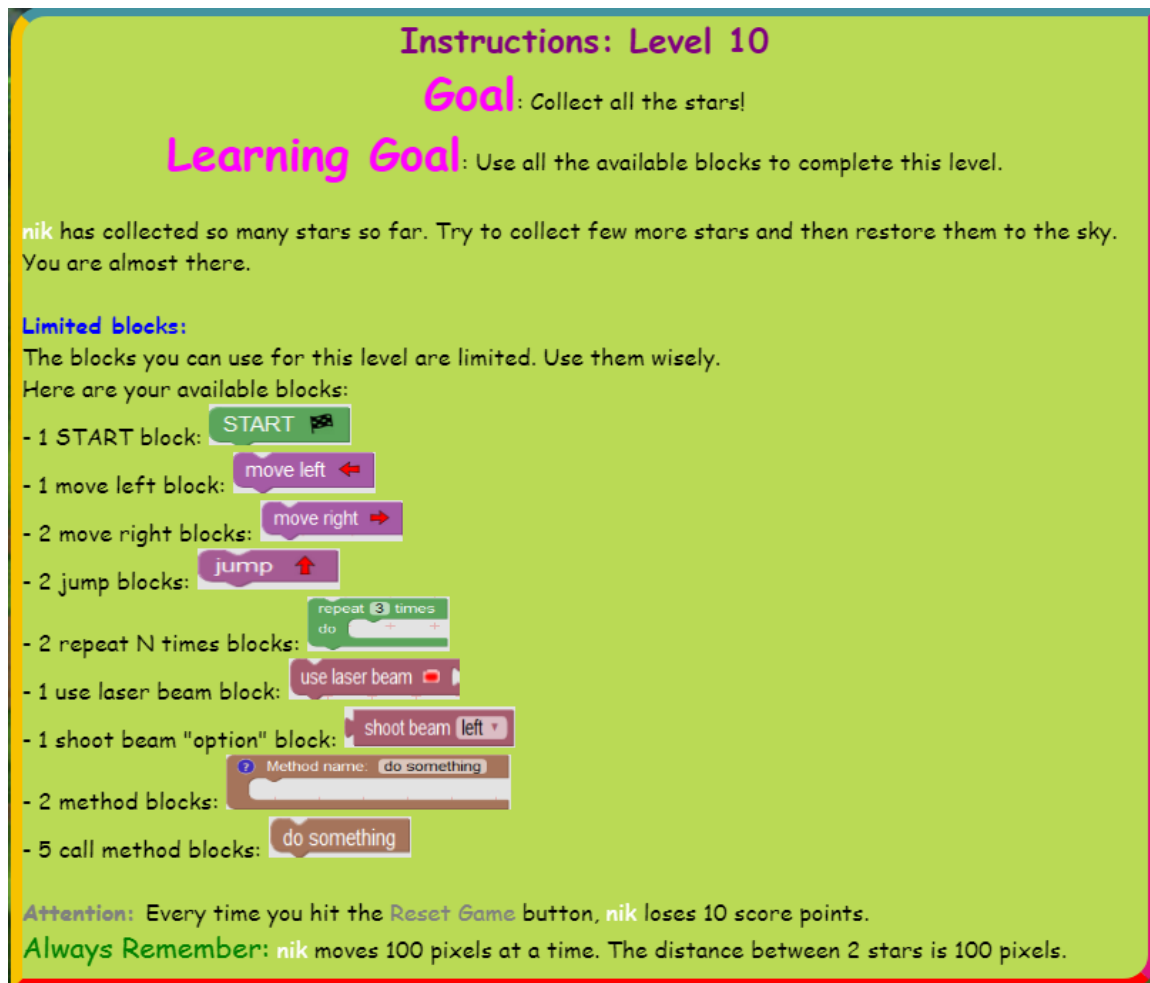
Νίκη	
if(totalBlocks Used === 31)	<p>Congratulations. You used the minimum number of blocks in order to complete Level 9. You are rewarded with +100 points, added to your current Score! Your total score is: '+((blocklyPoints + totalScore + levelScore) - Number(losePoints()))</p>

<p>else if(totalBlocks Used > 31)</p>	<p>-Tip: If you used two or more blocks of the same type under each other, then you should use a repeat N times block to earn more points</p> <p>-Well done. You completed Level 9. However you used more blocks than Level 9 needed so you are rewarded with +40 points, added to your current Score! Your total score is: ' +((blocklyPoints + totalScore + levelScore) - Number(losePoints()))</p>
<p>else if(totalBlocks Used < 31)</p>	<p>Well done. You completed Level 9. You probably though didn't use the "if"(condition) block the right way. You should try to use them in order to get more points. You are rewarded with +40, added to your current Score.'" +((blocklyPoints + totalScore + levelScore) - Number(losePoints()))</p>
<p>Ήττα</p>	
<p>Αν ηττηθεί (γενικά μηνύματα)</p>	<p>-Tip: Create a method and give it a name you like. The method should not be connected with the Start block. Inside the method add blocks that will help you check the color of the star the adventurer steps on, collect balloons and stars. First thing you have to check is the color of the star with an "if" block. Then inside that "if" add a second (nested)"if" block that will check if the star has a balloon on top of it. If it has a balloon place a "jump and collect option balloon" block inside your nested if. After that and outside your nested "if" block use "collect option star" block. Do this for all the colors of the stars. When you are done with your method, call it under your start block at the right position</p> <p>- Remember: The adventurer must step on the colorful star in order to collect it with the collect "option" star block</p>

3.4.11 Δέκατο επίπεδο παιχνιδιού - Limited Blocks



Εικόνα 67: Οθόνη δέκατου επιπέδου



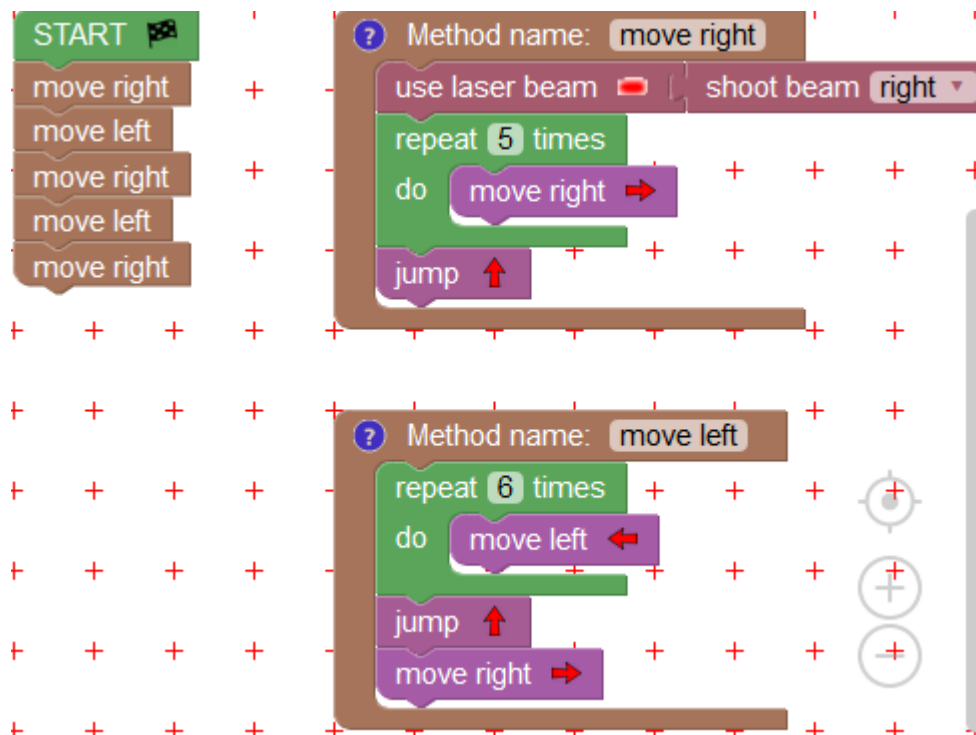
Εικόνα 68: Οδηγίες δέκατου επιπέδου

Τα διαθέσιμα μπλοκ στο δέκατο επίπεδο είναι περιορισμένα. Ο χρήστης θα πρέπει να αξιοποιήσει κατάλληλα τα διαθέσιμα μπλοκ για να βρει τη λύση. Στόχος του χρήστη είναι να βρει το μοτίβο που επαναλαμβάνεται και να δημιουργήσει τις κατάλληλες μεθόδους.

Πίνακας 24: Στόχοι, έννοιες και μπλοκ δέκατου επιπέδου

Στόχοι επιπέδου	
Στόχος	Ο χρήστης θα πρέπει να βρει τη λύση του επιπέδου αξιοποιώντας τον περιορισμένο αριθμό των διαθέσιμων μπλοκ
Διδακτικός στόχος	Ο χρήστης θα πρέπει να εντοπίσει το μοτίβο του παιχνιδιού και να σκεφτεί αφαιρετικά τον σχεδιασμό των μεθόδων του
Έννοιες	
-	
Μπλοκ	
-	

Στον Πίνακα 25 παρουσιάζονται οι έλεγχοι της περιοχής Program Output.



Εικόνα 69: Βέλτιστη λύση δέκατου επιπέδου

Πίνακας 25: Έλεγχοι περιοχής Program Output δέκατου επιπέδου

Νίκη	
if(totalBlocksUsed === 17)	<p>Congratulations. You used the minimum number of blocks in order to complete Level 10. You are rewarded with +100 points, added to your current Score! Your total score is: '+((blocklyPoints + totalScore + levelScore) - Number(losePoints()))</p>
Ήττα	
Αν ηττηθεί (γενικά μηνύματα)	<p>- Tip: Under your Start block you should only use "call method" type of blocks. Next to your Start block try to create 2 methods. Inside your first method, you should use your laser beam, move to the right multiple times and then jump block. At your second method you should move left multiple times then jump and move right block.</p>

3.4.12 Εντέκατο επίπεδο παιχνιδιού - Variable



Εικόνα 70: Οθόνη εντέκατου επιπέδου

Instructions: Level 11

Goal: Collect all the stars, brake the wall!

New Goal: Find the password to open the door. 

Finally **nik** made it to the last level of Blocklyland. There is a door that needs a password in order to open. **nik** knows the password. **To reveal the password use your left click and hold it on top of nik or inside your game area.**

New blocks:

- **set "variable" to "your input":**  Think variables as a container where you can store some data. At the example image the variable is called 'password'. By using that name we can access the data we stored inside it. The set 'variable' to 'your input' block allows you to store 'your input' data (which is number 5 at the example image) inside a variable called 'password'
- **open door with password: "variable"**  With this block you can try to open the door by using the variable name you created earlier. If you entered the right password inside your variable (at the example image the variable is called password) then the door will open. It is important to set the password variable first and then try to open the door.

Attention: Every time you hit the **Reset Game** button, **nik** loses 10 score points.

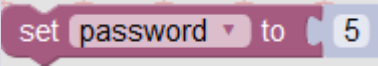
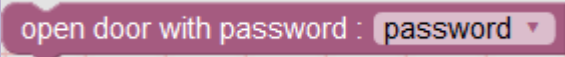
Always Remember: **nik** moves 100 pixels at a time. The distance between 2 stars is 100 pixels.

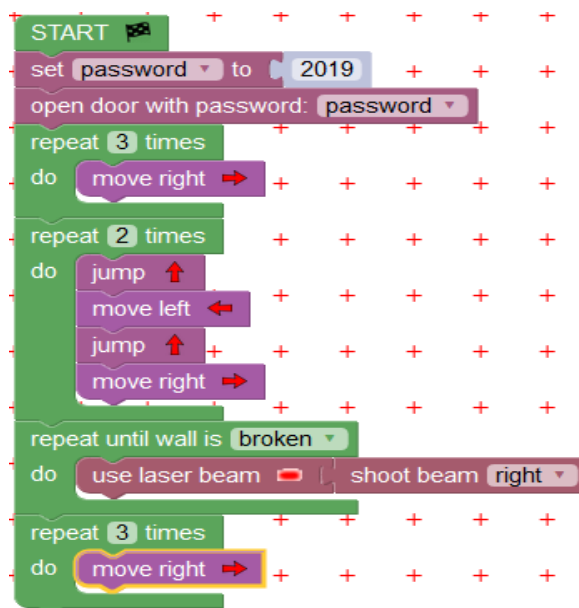
Εικόνα 71: Οδηγίες εντέκατου επιπέδου

Στο ενδέκατο επίπεδο ο χρήστης θα γνωρίσει την έννοια των μεταβλητών. Μέσα στο παιχνίδι ο χρήστης θα πρέπει να βρει το συνθηματικό για να ανοίξει η πόρτα. Για να μάθει το συνθηματικό θα πρέπει να κάνει αριστερό κλικ πάνω στον εξερευνητή ο οποίος το γνωρίζει. Έπειτα θα πρέπει να χρησιμοποιήσει το συνθηματικό που έμαθε, να το αποθηκεύσει στη μεταβλητή password και τέλος να ξεκλειδώσει την πόρτα χρησιμοποιώντας την τιμή της μεταβλητής.

Πίνακας 26: Στόχοι, έννοιες και μπλοκ δέκατου επιπέδου

Στόχοι επίπεδου	
Στόχος	Ο χρήστης θα πρέπει να βρει το συνθηματικό για να ξεκλειδώσει την πόρτα και να μαζέψει όλα τα αστέρια
Διδακτικός στόχος	Ο χρήστης θα έρθει σε μια πρώτη επαφή με την έννοια των μεταβλητών

Έννοιες	
<p>Hidden password</p>	<p>The explorer knows the password. To reveal the password use your left click and hold it on top of the explorer or inside your game area.</p>
Μπλοκ	
 <p>Μπλοκ ανάθεσης τιμής σε μεταβλητή</p>	<p>set "variable" to "your input". Think variables as a container where you can store some data. At the example image the variable is called 'password'. By using that name we can access the data we stored inside it. The set 'variable' to 'your input' block allows you to store 'your input' data (which is number 5 at the example image) inside a variable called 'password'</p>
 <p>Μπλοκ "open door with password: 'variable'"</p>	<p>open door with password: "variable". With this block you can try to open the door by using the variable name you created earlier. If you entered the right password inside your variable (at the example image the variable is called password) then the door will open. It is important to set the password variable first and then try to open the door.</p>



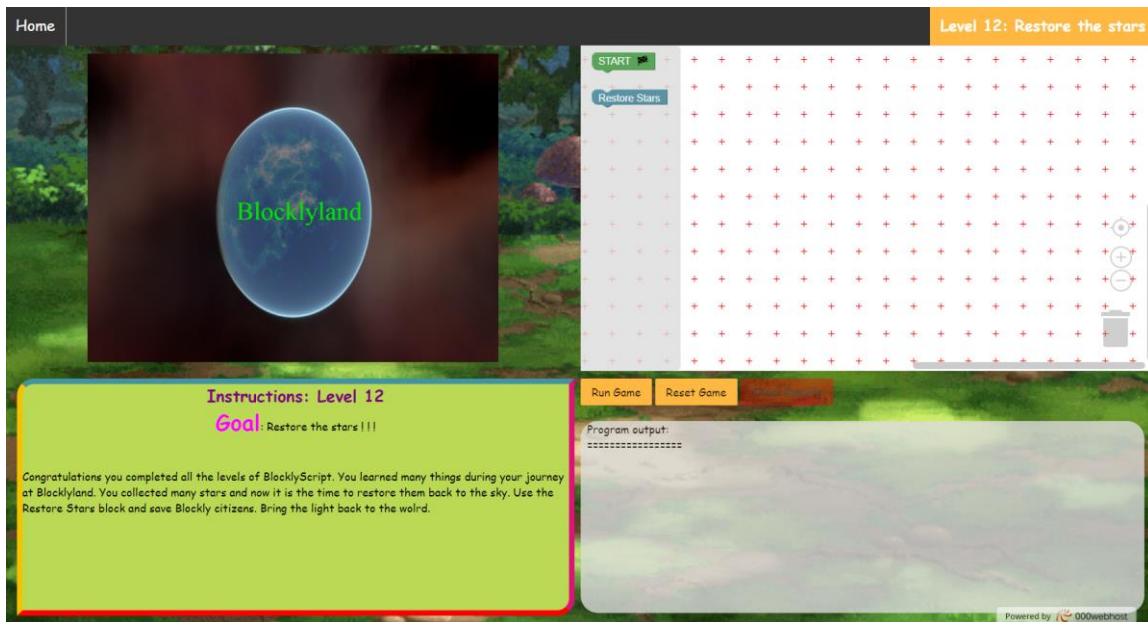
Εικόνα 72: Βέλτιστη λύση εντέκατου επιπέδου

Στον πίνακα 27 παρουσιάζονται οι έλεγχοι της περιοχής Program Output.

Πίνακας 27: Έλεγχοι περιοχής Program Output εντέκατου επιπέδου

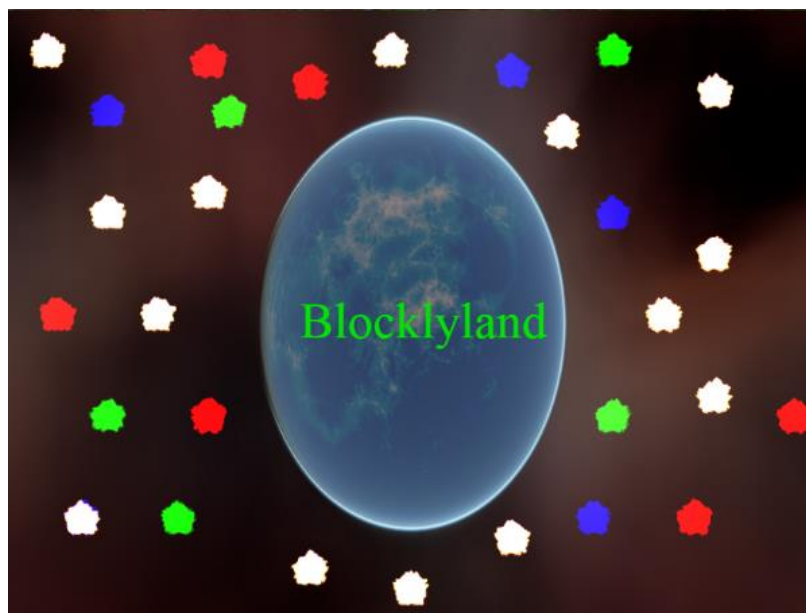
Νίκη	
if(totalBlocksUsed == 15)	Congratulations. You used the minimum number of blocks in order to complete Level 11. You are rewarded with +100 points, added to your current Score! Your total score is: '+((blocklyPoints + totalScore + levelScore) - Number(losePoints()))
else if(totalBlocksUsed > 15)	Well done. You completed Level 11. However you used more blocks than Level 11 needed so you are rewarded with +40 points, added to your current Score! Your total score is: '+((blocklyPoints + totalScore + levelScore) - Number(losePoints()))
Ήττα	
Αν ηττηθεί (γενικά μηνύματα)	Tip: In order to open the door you need a password. Click and hold your left click on top of your adventurer to reveal the password. Type that password inside your variable and then open the door using that variable.

3.4.13 Δωδέκατο επίπεδο παιχνιδιού - Restore the stars



Εικόνα 73: Οθόνη δωδέκατου επιπέδου

Στο δωδέκατο επίπεδο του παιχνιδιού ο χρήστης έχει τερματίσει ουσιαστικά το παιχνίδι μαζεύοντας όλα τα πεφταστέρια της Blocklyland. Χρησιμοποιώντας τις δύο διαθέσιμες εντολές του θα στείλει τα αστέρια πίσω στο διάστημα και θα ολοκληρώσει το στόχο που παιχνιδιού. Έπειτα μπορεί να δει τα στατιστικά από το κάθε επίπεδο του καθώς και να εκτυπώσει μια βεβαίωση που αποδεικνύει την ολοκλήρωση του παιχνιδιού.



Εικόνα 74: Τα αστέρια επέστρεψαν στην θέση του στο διάστημα

Level	Name	Score	Time(min.)	First Time Won	Times Reseted	Stars	Solution Reward
1	Movement	100	1:34	1:0	2	5	🌟
2	Jump	210	3:37	3:6	1	7	🌟
3	Repeat 'n' times	440	2:41	2:41	0	13	🌟
4	Repeat While	670	3:53	3:53	2	15	🌟
5	Repeat Until	990	5:41	5:41	1	19	🌟
6	Simple if	1200	1:11	1:11	0	15	🌟
7	If - else	1440	2:33	2:33	0	14	🌟
8	Nested if	1670	4:44	4:44	2	15	🌟
9	Method	1950	5:39	5:39	2	20	🌟
10	Limited Blocks	2330	3:8	3:8	1	29	🌟
11	Variable	2530	1:16	1:16	0	10	🌟
Total	-	2530	35:57	-	11	162	-

Εικόνα 75: Οθόνη αποτελεσμάτων παιχνιδιού

Στην εικόνα 96 παρουσιάζονται το σκορ του παίκτη ανά επίπεδο, ο συνολικός χρόνος που αφιέρωσε, ο χρόνος που το κέρδισε για πρώτη φορά, ο αριθμός των επανεκκινήσεων, τα αστέρια που συγκέντρωσε και η λύση που έδωσε (βέλτιστη ή απλή).

Home Certification



Congratulations

Bobi

You helped **Bobi** to complete all the levels of BlocklyScript and restore the fallen stars back to the sky.
Blocklyland is saved by you.

Score: **2530** Time: **35:57**

Print Certification

Powered by 

Εικόνα 76: Βεβαίωση ολοκλήρωσης του BlocklyScript

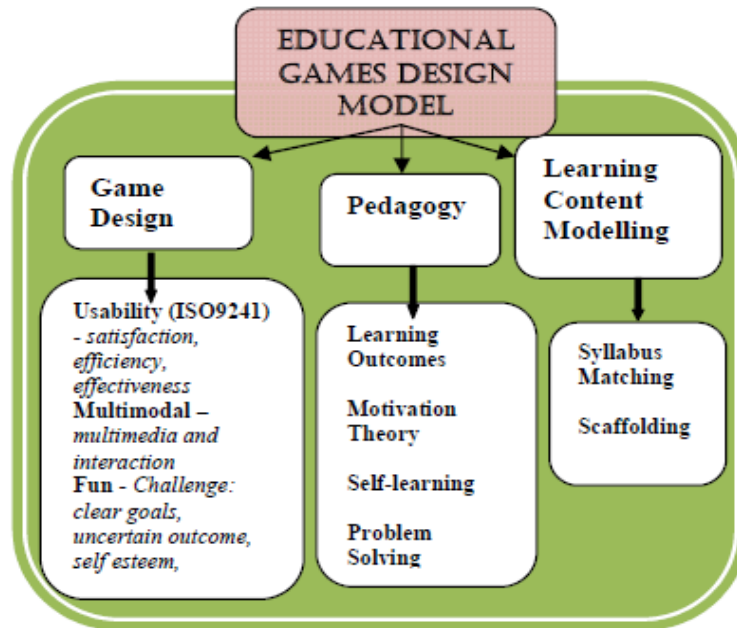
4 Σχεδίαση και ανάπτυξη του παιχνιδιού

4.1 Πλαίσιο σχεδίασης

Για την ανάπτυξη ενός παιχνιδιού σοβαρού σκοπού απαραίτητη προϋπόθεση αποτελεί η επιλογή του κατάλληλου πλαισίου σχεδίασης. Ένα πλαίσιο σχεδίασης βοηθάει στη σχεδίαση του παιχνιδιού παρέχοντας τους βασικούς πυλώνες πάνω στους οποίους πρέπει να στηρίζεται ένα παιχνίδι σοβαρού σκοπού για να πετύχει τους στόχους του. Μάλιστα υποστηρίζεται ότι ο συνδυασμός ενός καλά σχεδιασμένου παιχνιδιού με τους παιδαγωγικούς του στόχους βελτιστοποιεί τα οφέλη και την αποτελεσματικότητα του στην εκπαίδευση. Για τους εξής λόγους έγινε μια εκτενής έρευνα με σκοπό την επιλογή του κατάλληλου πλαισίου σχεδίασης που ικανοποιεί τις ανάγκες του BlocklyScript. Σε μια έρευνα που έγινε από τους Maliarakis, Satratzemi & Xinogalos (2014) παρουσιάζονται αρκετά από τα διαθέσιμα πλαίσια σχεδίασης ανάπτυξης παιχνιδιών σοβαρού σκοπού αναφέροντας για κάθε ένα τα χαρακτηριστικά του.

Με βάση αυτά επιλέχθηκε το πλαίσιο σχεδίασης εκπαιδευτικών παιχνιδιών που προτάθηκε από τους Ibrahim & Jaafar (2009). Το μοντέλο αυτό σχεδιάστηκε με στόχο να υποστηρίζει την αυτοδιδασκαλία και την αυτοαξιολόγηση του μαθητή μέσα στο παιχνίδι. Το μοντέλο αυτό βασίζεται σε τρεις βασικούς άξονες. Οι άξονες είναι ο σχεδιασμός του παιχνιδιού (Game Design), οι παιδαγωγικοί στόχοι (Pedagogy) και η μοντελοποίηση του μαθησιακού περιεχομένου (Learning Content Modelling), όπως φαίνεται στην Εικόνα 77.

Ο παράγοντας της σχεδίασης του παιχνιδιού αφορά την ευχρηστία, την πολυτροπικότητα και την ψυχαγωγία που προσφέρει. Η ευχρηστία του παιχνιδιού βασίζεται στην ικανοποίηση, στην αποδοτικότητα και στην αποτελεσματικότητα του και θα πρέπει να ελέγχεται μέσω του πρωτοτύπου ευχρηστίας ISO 9241 (Pinelle and Wong, 2008). Η πολυτροπικότητα είναι πολύ σημαντική για τα παιχνίδια διότι συνδυάζει τα πολυμέσα, όπως είναι το κείμενο, τα γραφικά, τον ήχο, το βίντεο και τις κινούμενες εικόνες, με την διαδραστικότητα με την οποία αλληλεπιδρούν οι μαθητές και παίρνουν άμεση ανατροφοδότηση. Η ψυχαγωγία ξεχωρίζει με τα άλλα δύο χαρακτηριστικά και παρέχει κίνητρο και δέσμευση στους παίκτες.



Εικόνα 77: Πλαίσιο σχεδίασης εκπαιδευτικού παιχνιδιού (Ibrahim & Jaafar, 2009)

Ο δεύτερος παράγοντας είναι τα παιδαγωγικά θέματα του παιχνιδιού και αφορά τον τελικό βαθμό επίτευξης των διδακτικών αποτελεσμάτων. Το παιχνίδι θα σχεδιαστεί βάσει των τριών πρώτων επιπέδων της ταξινόμησης εκπαιδευτικών αποτελεσμάτων Bloom (γνώση, κατανόηση, εφαρμογή). Επιπλέον η θεωρία θα πρέπει να παρέχει κίνητρο στους μαθητές και να υποστηρίζει την αυτόνομη μάθηση. Τέλος θα πρέπει να αναπτύσσει τις ικανότητες του μαθητή στην επίλυση προβλημάτων.

Η μοντελοποίηση του περιεχομένου αποτελεί τον τρίτο παράγοντα του πλαισίου σχεδίασης. Ο συγκεκριμένος παράγοντας υποστηρίζει ότι τα μαθήματα του παιχνιδιού θα πρέπει να σχεδιαστούν με κατάλληλο τρόπο έτσι ώστε να υποστηρίζουν την αυτοδιδασκαλία και να πετύχουν τους διδακτικούς του στόχους. Επιπλέον ο τρόπος παρουσίασης της του εκπαιδευτικού υλικού θα πρέπει να γίνεται με ομαλή και προοδευτική μέθοδο με στόχο να παρακινήσει το ενδιαφέρον του παίκτη.

Στην συνέχεια στον Πίνακα 28, 29 και 30 παρουσιάζονται τα χαρακτηριστικά του παιχνιδιού που ικανοποιούν τους παράγοντες του πλαισίου σχεδίασης.

Πίνακας 28: Χαρακτηριστικά σχεδίασης του BlocklyScript (Game design)

Παράγοντες (Ibrahim & Jaafar, 2009)	Σχεδιαστική απόφαση
Ευχρηστία (Usability)	
Ικανοποίηση (Satisfaction)	<p>Το περιβάλλον του παιχνιδιού είναι φιλικό προς τον παίκτη και η πλοήγηση του σε αυτό είναι εύκολη.</p> <p>Το ενδιαφέρον σενάριο σε συνδυασμό με τα ελκυστικά γραφικά και τους ήχους κάθε επιπέδου παρέχουν αισθητική ικανοποίηση στον χρήστη. Επιπλέον η αποθήκευση του σκορ, του χρόνου και η κονκάρδα επιβράβευσης με την ολοκλήρωση ενός επιπέδου δίνουν χαρά στον παίκτη βλέποντας την προσπάθειά του να ανταμείβεται. Ακόμη δίνει τη δυνατότητα στον χρήστη μέσω της αποθήκευσης της προόδου του στην βάση δεδομένων να παίζει το παιχνίδι όποτε διαθέτει ελεύθερο χρόνο.</p> <p>Η ενθάρρυνση για τον εντοπισμό της βέλτιστης λύσης δημιουργεί ένα αίσθημα ικανοποίησης στον παίκτη.</p>
Αποδοτικότητα (Efficiency)	<p>Το παιχνίδι παρέχεται δωρεάν στο διαδίκτυο και υποστηρίζεται από τους πιο δημοφιλή περιηγητές.</p> <p>Η θεωρία παρουσιάζεται μέσω λιτού και περιεκτικού κειμένου σε συνδυασμό με εικόνες και απλών παραδειγμάτων που εφαρμόζουν τη θεωρία. Επίσης η κατηγοριοποίηση των μπλοκ εντολών και η αντιστοίχιση χρωμάτων διευκολύνει την κατανόηση της λειτουργικότητας τους από τον μαθητή</p>
Αποτελεσματικότητα (Effectiveness)	<p>Η ενσωμάτωση του κουμπιού βοήθειας υπενθυμίζει όλες τις έννοιες που διδάχτηκε ο παίκτης στα προηγούμενα επίπεδα σε περίπτωση που τις έχει ξεχάσει.</p> <p>Η παρουσίαση μιας και μοναδικής έννοιας ανά επίπεδο συντελεί στην αποτελεσματικότερη εκμάθησή της.</p> <p>Οι έλεγχοι και η ανατροφοδότηση για τις απαντήσεις του</p>

	<p>χρήστη τον βοηθούν να κατανοήσει ακριβώς το σημείο που μπορεί να έκανε λάθος</p>
<p>Πολυτροπικότητα (Multimodal)</p>	
<p>Πολυμέσα και αλληλεπιδράσεις (multimedia and interaction)</p>	<p>Η ενσωμάτωση αντικειμένων για συλλογή (απλά αστέρια, πολύχρωμα αστέρια και μπαλόνια), η αποφυγή εμποδίων (καρφιά), η αντιμετώπιση εχθρών (ιπτάμενοι εξωγήινοι), η καταστροφή εμποδίων (τοίχοι) και η εξερεύνηση μυστικών συνθηματικών διατηρούν αμείωτο το ενδιαφέρον του παίκτη.</p> <p>Η χαλαρή και χαρούμενη μουσική στο υπόβαθρο βοηθάει τον χρήστη να σκεφτεί και να χαλαρώσει στην προσπάθειά του να σχεδιάσει τον κατάλληλο αλγόριθμο για την επίλυση του επιπέδου.</p> <p>Η εμφάνιση κατάλληλων μηνυμάτων και ήχων στο τέλος της εκτέλεσης του αλγορίθμου του χρήστη διαχωρίζουν την επιτυχημένη από την εσφαλμένη λύση.</p> <p>Ο χρήστης αλληλεπιδρά με τον εξερευνητή σέρνοντας και αφήνοντας τα διαθέσιμα μπλοκ εντολών στην περιοχή του συντάκτη Blockly και συνδέοντας τα κατάλληλα μεταξύ τους.</p> <p>Επιπλέον υπάρχουν διάφοροι τύποι μπλοκ που ζητούν μια είσοδο από τον χρήστη. Η είσοδος αυτή μπορεί να είναι η επιλογή του κατάλληλου χρώματος, η επιλογή της κατάλληλης συνθήκης, η μορφοποίηση του ή η πληκτρολόγηση ενός αλφαριθμητικού.</p> <p>Το πολύχρωμο περιβάλλον προκαλεί ευχάριστη διάθεση στον παίκτη.</p> <p>Στο τελευταίο επίπεδο του παιχνιδιού ο παίκτης εκπληρώνει τον στόχο του σεναρίου δημιουργώντας αντικείμενα αστεριών στο παιχνίδι.</p>
<p>Ψυχαγωγία - Πρόκληση (Fun - Challenge)</p>	

Ξεκάθαροι στόχοι (clear goals)	Οι στόχοι σε κάθε επίπεδο είναι ξεκάθαροι και εμφανίζονται στην περιοχή των οδηγιών του επιπέδου. Ο βασικός στόχος είναι να συλλέξει ο εξερευνητής όλα τα διαθέσιμα αστέρια ή/και μπαλόνια του επιπέδου με τη βοήθεια του αλγορίθμου του παίκτη.
Αβέβαιο αποτέλεσμα (uncertain outcome)	Ο παίκτης θα πρέπει να βρει την λύση του επιπέδου για να μπορέσει να συνεχίσει στα επόμενα επίπεδα αλλά και να δει τα αποτελέσματα της λύσης του.
Αυτοπεποίθηση (self esteem)	Η παρουσία του σκορ και των κονκάρδων, η εμφάνιση κατάλληλα διαμορφωμένων μηνυμάτων για την εύρεση της βέλτιστης λύσης και η επανεκκίνηση των επιπέδων χωρίς περιορισμούς ενισχύουν την αυτοπεποίθηση του παίκτη.

Πίνακας 29: Παιδαγωγικά θέματα του BlocklyScript (Pedagogy)

Παράγοντες (Ibrahim & Jaafar, 2009)	Σχεδιαστική απόφαση
Μαθησιακά αποτελέσματα (Learning outcomes)	Παίζοντας ο χρήστης θα μάθει βασικές έννοιες του προγραμματισμού, όπως είναι η ακολουθιακή εκτέλεση των εντολών, οι βρόχοι, οι εντολές if, η περιορισμένη χρήση εντολών και οι μεταβλητές
Θεωρία παρακίνησης (Motivation Theory)	Καθώς ο παίκτης ανεβαίνει επίπεδα εμπλουτίζει τις γνώσεις του πάνω στις βασικές έννοιες του προγραμματισμού. Η θεωρία παρουσιάζεται μέσω περιεκτικού κειμένου μαζί με εικόνες και παραδείγματα εφαρμογής των εντολών. Ο χρήστης διατηρεί το ενδιαφέρον του για τη θεωρία που ήδη έχει μάθει διότι τη χρησιμοποιεί σε συνδυασμό με τις έννοιες του νέου επιπέδου για να βρει τη λύση.
Αυτοδιδασκαλία (Self-learning)	Αρχικά το παιχνίδι δεν προϋποθέτει γνώσεις προγραμματισμού από τον χρήστη. Σε κάθε επίπεδο ο χρήστης προσπαθεί να αφομοιώσει μια νέα προγραμματιστική

	<p>έννοια. Καθώς ανεβαίνει τα επίπεδα καλείται να χρησιμοποιήσει τις νέες έννοιες σε συνδυασμό με τη θεωρία που ήδη διάβασε.</p> <p>Το παιχνίδι επιτρέπει στον χρήστη να ανατρέχει στη θεωρία σε κάθε επίπεδο καθώς και την επανεκκίνηση του χωρίς να υπάρχει κάποιος περιορισμός στις προσπάθειες επανασχεδίασης του αλγορίθμου.</p>
<p>Επίλυση προβλημάτων (Problem solving)</p>	<p>Σε κάθε επίπεδο ο χρήστης εφαρμόζει τη θεωρία που έμαθε με στόχο να σχεδιάσει έναν αλγόριθμο και να λύσει το επίπεδο καθοδηγώντας τον εξερευνητή.</p> <p>Με τη βοήθεια που παίρνει από την περιοχή "Program output" μπορεί να καταλάβει ακριβώς το σημείο που έκανε το λάθος και να διαβάσει χρήσιμες συμβουλές για τη σωστή εφαρμογή της θεωρίας.</p>

Πίνακας 30: Μοντελοποίηση μαθησιακού περιεχομένου του BlocklyScript (Learning Content Modeling)

Παράγοντες (Ibrahim & Jaafar, 2009)	Σχεδιαστική απόφαση
<p>Κατάλληλο πρόγραμμα σπουδών (Syllabus matching)</p>	<p>Για την επιλογή της κατάλληλης σειράς παρουσίασης της θεωρίας λήφθηκαν υπόψη δύο σημαντικοί παράγοντες. Αρχικά μελετήθηκαν παρόμοια παιχνίδια σοβαρού σκοπού που εφαρμόζουν οπτικές γλώσσες προγραμματισμού και επιλέχθηκε εκείνο που παρουσιάζει πιο ομαλά τις προγραμματιστικές έννοιες σε έναν αρχάριο σύμφωνα με την κρίση του συγγραφέα. Έπειτα προσαρμόστηκε η σειρά της θεωρίας στις ανάγκες του BlocklyScript.</p> <p>Η σειρά των επιπέδων επιλέχθηκε από το Rapid Router: https://www.codeforlife.education/rapidrouter/</p>
<p>Προοδευτικότητα-</p>	<p>Η ροή παρουσίασης της θεωρίας γίνεται ομαλά. Αρχικά ο</p>

**Εξελιξιμότητα
(Scaffolding)**

μαθητής μαθαίνει την ακολουθιακή εκτέλεση. Έπειτα γνωρίζει του βρόχους επανάληψης. Συνεχίζοντας γνωρίζει τις εντολές "if" και στα τελευταία επίπεδα έρχεται σε μια πρώτη επαφή με τις μεθόδους και τις μεταβλητές. Ακόμη ο χρήστης μπορεί να αφιερώσει όσο χρόνο επιθυμεί σε κάθε επίπεδο διαβάζοντας τη θεωρία που έχει ήδη μάθει.

4.2 Πλαίσιο ανάπτυξης

Για την ανάπτυξη του παιχνιδιού συνδυάστηκαν δύο διαθέσιμα εργαλεία που παρέχονται δωρεάν στο διαδίκτυο. Το πρώτο είναι η βιβλιοθήκη Blockly για την οποία έγινε αναφορά στο κεφάλαιο 2.4.3. Το δεύτερο εργαλείο είναι το πλαίσιο ανάπτυξης διαδικτυακών 2D παιχνιδιών Phaser. Στην Εικόνα 78 φαίνεται το λογότυπο του Phaser.



Εικόνα 78: Λογότυπο Phaser

Το Phaser είναι ένα από τα πιο δημοφιλή πλαίσια ανάπτυξης παιχνιδιών για τους εξής λόγους:

- **Κόστος:** Το Phaser παρέχεται δωρεάν στους προγραμματιστές ανάπτυξης διαδικτυακών παιχνιδιών
- **Κοινότητα:** Παρέχει μια μεγάλη κοινότητα από προγραμματιστές και υπάρχουν ομάδες σε διάφορα μέσα κοινωνικής δικτύωσης όπως είναι το Twitter, το Slack και το Discord. Μέσα από αυτά δημοσιεύονται σύνδεσμοι από ανεπτυγμένα παιχνίδια, εικόνες και περιεχόμενα από τις νέες τάσεις στις τεχνολογίες ανάπτυξης παιχνιδιών. Επιπλέον το Phaser παρέχει την δυνατότητα ενημέρωσης των χρηστών μέσω ηλεκτρονικού ταχυδρομείου.

- **Ευχρηστία:** Είναι εύκολο στην εγκατάσταση και γρήγορο στην εκμάθηση. Παρέχει πληθώρα από οδηγούς και παραδείγματα μέσα από τα οποία ο χρήστης μαθαίνει να αξιοποιεί τις δυνατότητες του πλαισίου. Επιπλέον ο προγραμματιστής μπορεί να λύσει τυχόν απορίες του μέσω του φόρουμ που διαθέτει η επίσημη ιστοσελίδα.
- **Γλώσσα:** Υποστηρίζει τις γλώσσες JavaScript και TypeScript. Η JavaScript είναι μια από τις δημοφιλέστερες γλώσσες προγραμματισμού για την δημιουργία ιστοσελίδων που υποστηρίζουν την επικοινωνία τους με τον πελάτη (client side). Επιτρέπει την ανταλλαγή ασύγχρονων δεδομένων και την δυναμική αλλαγή του περιεχομένου του εγγράφου. Η TypeScript είναι μια εξελιγμένη εκδοχή της JavaScript.
- **Άλλα χαρακτηριστικά:**
 - Ενσωματώνει μηχανή για τον έλεγχο των νόμων της φυσικής του περιβάλλοντος (physics engine)
 - Εύκολη φόρτωση γραφικών και ήχων και παρέχει πληθώρα συναρτήσεων για την επεξεργασία αυτών.
 - Τρεις βασικές καταστάσεις του παιχνιδιού preload, create και update.
 1. preload - υπεύθυνη για την φόρτωση στοιχείων (assets)
 2. create - δημιουργεί όλα τα αντικείμενα του παιχνιδιού
 3. update - περιέχει τη λογική του παιχνιδιού και ενημερώνεται κάθε εικόνα (frame)
 - Αναλυτική τεκμησίωση για την κατανόηση των κλάσεων και των συναρτήσεων που παρέχει το Phaser

4.3 Γραφικά και ήχοι παιχνιδιού

Η επιλογή των γραφικών και των ήχων συνάδει με το σενάριο του παιχνιδιού. Τα γραφικά και οι ήχοι του παιχνιδιού διατίθενται δωρεάν στο διαδίκτυο και επιλέχθηκαν έπειτα από την αναζήτηση στις παρακάτω ιστοσελίδες.

Γραφικά παιχνιδιού:

- <https://opengameart.org/>
- <https://www.gameart2d.com/>
- www.itch.io

Ήχοι παιχνιδιού:

- <https://opengameart.org/>
- <https://freesound.org/>
- <https://www.soundjay.com/>
- <https://www.pacdv.com/sounds/>

Επιπλέον για την επεξεργασία των γραφικών και τη δημιουργία των στοιχείων (sprites) χρησιμοποιήθηκαν τα εξής δωρεάν εργαλεία:

- <https://draeton.github.io/stitches/> (Υπεύθυνο για την δημιουργία των sprite sheet)
- <https://onlinepngtools.com/create-transparent-png> (Υπεύθυνο για τη διαφάνεια των εικόνων)
- <https://resizeimage.net/> (Υπεύθυνο για την τροποποίηση των διαστάσεων μιας εικόνας)

4.4 Αρχιτεκτονική Λογισμικού

4.4.1 Είσοδος - Εγγραφή στο παιχνίδι

Για την καλύτερη οργάνωση των αρχείων που είναι υπεύθυνα για την είσοδο και την εγγραφή στη βάση δεδομένων μελετήθηκαν αρκετοί οδηγοί και ο τελικός σκελετός βασίζεται στον σχεδιασμό που προτείνει ο μηχανικός λογισμικού Awa Melvine, ο οποίος ασχολείται με τη δημιουργία ιστοσελίδων και την ανάπτυξη εφαρμογών σε συσκευές Android και IOS. Μπορεί κανείς να δει τη συνεισφορά του στο σύνδεσμο: <https://github.com/AwaMelvine?tab=repositories>

Η πρώτη ενέργεια που πρέπει να κάνει ο χρήστης για να έχει πρόσβαση στο παιχνίδι είναι να συνδεθεί στη βάση δεδομένων του παιχνιδιού στην οποία αποθηκεύεται η πρόοδος του. Για να συνδεθεί στην βάση θα πρέπει να έχει δημιουργήσει τον δικό του λογαριασμό με τα στοιχεία που έδωσε κατά την εγγραφή του. Για τον λόγο αυτό δημιουργήθηκαν δύο σελίδες, μια σελίδα για την είσοδο του χρήστη (αρχείο login.php) και μια σελίδα για την εγγραφή του (αρχείο register.php). Όπως φαίνεται και από τις καταλήξεις των αρχείων για την επικοινωνία του χρήστη με τη βάση δεδομένων χρησιμοποιήθηκε η γλώσσα PHP που είναι μια server-side γλώσσα προγραμματισμού. Στον Κώδικα 1 παρουσιάζεται το αρχείο register.php.

```

1  <?php include('server.php'); ?>
2  <!DOCTYPE html>
3  <html>
4  <head>
5      <title>User registration system</title>
6      <link rel="stylesheet" type="text/css" href="regStyle.css">
7  </head>
8  <body>
9
10
11     <br>
12     <br>
13     <br>
14     <span>BlocklyScript</span>
15     <div class="header">
16         <h2>Register</h2>
17     </div>
18
19     <form method="post" action="register.php">
20         <?php include('errors.php'); ?>
21         <div class="input-group">
22             <label>Username</label>
23             <input type="text" name="username" value="<?php echo $username; ?>">
24         </div>
25         <div class="input-group">
26             <label>Password</label>
27             <input type="password" name="password_1">
28         </div>
29         <div class="input-group">
30             <label>Confirm Password</label>
31             <input type="password" name="password_2">
32         </div>
33         <br>
34         <div class="input-group">
35             <button type="submit" name="register" class="btn">Register</button>
36         </div>
37         <br>
38         <p>Already a member? <a href="login.php">Login</a></p>
39     </form>
40 </body>
41 </html>

```

Κώδικας 1: Εγγραφή χρήστη στην βάση δεδομένων (register.php)

Ξεκινώντας από το αρχείο register.php ο κώδικάς του περιέχει μια φόρμα (form) με μέθοδο (method) post και ενέργεια (action) "register.php", η οποία είναι υπεύθυνη να στείλει τα δεδομένα που θα καταχωρήσει ο χρήστης στα πεδία της στον server (αρχείο server.php). Ακόμη στην αρχή της φόρμας υπάρχει ένα πλαίσιο για την εμφάνιση μηνυμάτων αποτυχημένης εγγραφής στο σύστημα (αρχείο errors.php).

Παρόμοια είναι και η διάταξη του κώδικα στο αρχείο login.php με την διαφορά ότι η ενέργεια (action) ονομάζεται "login.php". Έτσι ο server διαχωρίζει από ποια σελίδα του έρχονται τα δεδομένα και ανάλογα με την ενέργεια κάνει τους αντίστοιχους ελέγχους. Στον Κώδικα 2 γίνονται οι έλεγχοι στο αρχείο server.php για την εγγραφή του χρήστη (το όνομα πρέπει να είναι μοναδικό).

```

11 if(isset($_POST['register'])) {
12     $username = mysqli_real_escape_string($db, $_POST['username']);
13     $password_1 = mysqli_real_escape_string($db, $_POST['password_1']);
14     $password_2 = mysqli_real_escape_string($db, $_POST['password_2']);
15
16     if(empty($username)) {
17         array_push($errors, "Username is required");
18     }
19
20     if(!empty($username)) {
21         $query = "SELECT username FROM users WHERE username='$username'";
22         $result = mysqli_query($db, $query);
23         if(mysqli_num_rows($result) == 1) {
24             array_push($errors, "Username already exists. Try a new one.");
25         }
26     }
27
28     if(empty($password_1)) {
29         array_push($errors, "Password is required");
30     }
31
32     if($password_1 != $password_2) {
33         array_push($errors, "The two password do not match");
34     }
35
36     if(count($errors) == 0) {
37         $password = md5($password_1);
38         $sql = "INSERT INTO users (username, password)
39             VALUES ('$username', '$password')";
40         mysqli_query($db, $sql);
41         $_SESSION['username'] = $username;
42         $_SESSION['success'] = "You are now logged in";
43         header('location: Game/index.php');
44     }
45 }
46

```

Κώδικας 2: Έλεγχοι δεδομένων εγγραφής χρήστη στην βάση δεδομένων (server.php)

Στην Κώδικα 3 γίνονται οι έλεγχοι στο αρχείο server.php για την είσοδο του χρήστη στο παιχνίδι (το όνομα και ο κωδικός πρέπει να υπάρχουν στην βάση δεδομένων).

```

48  if(isset($_POST['login'])) {
49      $username = mysqli_real_escape_string($db, $_POST['username']);
50      $password = mysqli_real_escape_string($db, $_POST['password']);
51
52
53      if(empty($username)) {
54          array_push($errors, "Username is required");
55      }
56      if(empty($password)) {
57          array_push($errors, "Password is required");
58      }
59
60      if(count($errors) == 0 ) {
61          $password = md5($password);
62          $query = "SELECT * FROM users WHERE username='$username' AND
password='$password'";
63          $result = mysqli_query($db, $query);
64          if(mysqli_num_rows($result) == 1){
65              while($row = mysqli_fetch_assoc($result))
66              {
67                  $playerData[] = $row;
68              }
69
70              $_SESSION['username'] = $username;
71              $_SESSION['success'] = "You are now logged in";
72              header('location: Game/index.php');
73          }else{
74              array_push($errors, "Wrong username/password combination");
75          }
76      }
77  }

```

Κώδικας 3: Έλεγχοι δεδομένων εισόδου χρήστη στην βάση δεδομένων (server.php)

Με την επιτυχή σύνδεση ή εγγραφή του χρήστη δημιουργούνται δύο μεταβλητές συνεδρίας username και success για το ψευδώνυμο και την επιτυχημένη σύνδεση του αντίστοιχα (π.χ. γραμμές 70-71 Κώδικα 3), οι οποίες διατηρούνται μέχρι ο χρήστης να βγει από το σύστημα ή να κλείσει τον περιηγητή, και στη συνέχεια μεταφέρεται στην αρχική οθόνη του παιχνιδιού (π.χ. γραμμή 72 Κώδικα 3).

4.4.2 Οργάνωση βάσης δεδομένων του παιχνιδιού

Για την διαχείριση της σχεσιακής βάσης δεδομένων MySQL επιλέχθηκε ένα από τα πιο δημοφιλή εργαλεία λογισμικού, το οποίο ονομάζεται phpMyAdmin. Το εργαλείο αυτό περιέχει μια απλοϊκή διεπαφή χρήστη για την οργάνωση και τη συντήρηση της βάσης δεδομένων και παρέχει δικαιώματα διαχειριστή στον χρήστη. Πρόκειται ουσιαστικά για ένα εργαλείο που εκτελεί εντολές MySQL με την βοήθεια γραφικών. Οι πιο συχνά χρησιμοποιούμενες λειτουργίες είναι η διαχείριση των βάσεων δεδομένων, των πινάκων, των στηλών, των σχέσεων, των ευρετηρίων, των χρηστών και των δικαιωμάτων.

Όσον αφορά το BlocklyScript, στο παράρτημα Α υπάρχει ένας συνοπτικός οδηγός ο οποίος δείχνει τον τρόπο της δωρεάν φιλοξενίας του παιχνιδιού σε συνδυασμό με τη βάση δεδομένων του από την ιστοσελίδα 000webhost.com. Η βάση του παιχνιδιού ονομάζεται "blocklyland" και περιέχει έναν πίνακα users με τους χρήστες. Ο πίνακας users αποτελείται από 98 στήλες και περιέχει όλα τα στοιχεία από την πρόοδο του χρήστη στο παιχνίδι. Πιο συγκεκριμένα στις στήλες αποθηκεύονται τα εξής χαρακτηριστικά ταυτοποίησης του χρήστη κατά την είσοδο του:

- Το όνομα - Πρωτεύον κλειδί (username - Primary key)
- Ο κωδικός σε κωδικοποίηση md5 (password)

Έπειτα για κάθε επίπεδο αποθηκεύονται στις στήλες της βάσης τα δεδομένα του Πίνακα 31 (Σημείωση: X είναι ο αριθμός του αντίστοιχου επιπέδου).

Πίνακας 31: Στήλες βάσης δεδομένων για την πρόοδο του χρήστη στο παιχνίδι

Δεδομένα προόδου χρήστη στην βάση δεδομένων	
Στήλες βάσης δεδομένων	Περιγραφή
levelXwon	Ένας ακέραιος αριθμός που δηλώνει αν χρήστης κέρδισε το X επίπεδο.
levelXscore	Ακέραιος αριθμός για το σκορ του χρήστη στο X επίπεδο
levelXtime	Ακέραιος αριθμός για τον χρόνο που αφιέρωσε συνολικά ο χρήστης στο X επίπεδο
levelXfirsttimewon	Ακέραιος αριθμός για την πρώτη φορά που νίκησε ο χρήστης στο X επίπεδο
levelXtimesreseted	Ακέραιος αριθμός για τις επαναλήψεις στο X επίπεδο
levelXattempts	Ένα κείμενο με τις προσπάθειες αλληλοσύνδεσης των μπλοκ εντολών του χρήστη στο X επίπεδο
levelXstars	Ακέραιος αριθμός που δηλώνει τα αστέρια που μάζεψε ο χρήστης στην τελευταία του προσπάθεια στο X επίπεδο
levelXbadge	Ακέραιος αριθμός για τον έλεγχο της λύσης της τελευταίας προσπάθειας του χρήστη στο X επίπεδο

Όπως έχει ήδη αναφερθεί τα παιχνίδια σοβαρού σκοπού είναι ένας ενδιαφέρον και εναλλακτικός τρόπος στην εκπαίδευση μικρών μαθητών. Παίζοντας ο χρήστης μαθαίνει νέες έννοιες σε έναν συγκεκριμένο τομέα (π.χ προγραμματισμός) διαβάζοντας την θεωρία και εφαρμόζοντας την με σκοπό να νικήσει το επίπεδο. Ωστόσο σύμφωνα με τους Maliarakis, Satratzemi & Xinogalos (2014) ένα εμπλουτισμένο εκπαιδευτικό παιχνίδι θα πρέπει να καταγράφει την πρόοδο και τα αποτελέσματα των παικτών με σκοπό την περαιτέρω ανάλυση τους από τους ερευνητές (learning analytics).

Στο εκπαιδευτικό παιχνίδι BlocklyScript τα δεδομένα της βάσης μπορούν να χρησιμοποιηθούν κυρίως για να εντοπιστούν οι δυσκολίες που αντιμετωπίζουν οι παίκτες στην κατανόηση τη θεωρίας. Συγκεκριμένα μπορεί να μελετηθεί ο χρόνος που αφιέρωσε ο χρήστης για να κατανοήσει την θεωρία στο αντίστοιχο επίπεδο. Επιπλέον ο ερευνητής μπορεί να δει τον χρόνο που έκανε ο χρήστης μέχρι να κερδίσει για πρώτη φορά το επίπεδο και συγκρίνοντας τον συγκεκριμένο χρόνο με τον συνολικό χρόνο του επιπέδου να καταλάβει εάν ο χρήστης αφιέρωσε παραπάνω χρόνο για βελτιώσει τον αλγόριθμο του. Ακόμη πολύ σημαντικό στοιχείο είναι ο αριθμός των επανεκκινήσεων του κάθε επιπέδου και υποδηλώνει πόσο εύκολο ή δύσκολο είναι για τον παίκτη να εφαρμόσει τη θεωρία που έμαθε. Τέλος η στήλη κειμένου με τις προσπάθειες του παίκτη βοηθάει τον ερευνητή να καταλάβει ακριβώς τα σημεία του αλγορίθμου στα οποία ο παίκτης έκανε το λάθος.

4.4.3 Λήψη και ενημέρωση των δεδομένων της βάσης

Με την επιτυχή σύνδεση του χρήστη στην αρχική οθόνη του παιχνιδιού ξεκινάει και η διαδικασία λήψης των στοιχείων του από την βάση δεδομένων. Με βάση το όνομα του χρήστη εκτελείται η κατάλληλη δήλωση SQL (στο αρχείο server.php) η οποία επιστρέφει όλα τα δεδομένα του χρήστη όπως φαίνεται στον Κώδικα 4.

```
85 if(isset($_SESSION['username'])) {
86     $playerData = array();
87     $query = "SELECT * FROM users WHERE username='".$_SESSION['username']."'";
88     $result = mysqli_query($db, $query);
89     while($row = mysqli_fetch_assoc($result))
90     {
91         $playerData[] = $row;
92     }
93     $_SESSION['playerData'] = json_encode($playerData);
94 }
```

Κώδικας 4: Λήψη δεδομένων συνδεδεμένου χρήστη (server.php)

Έπειτα τα δεδομένα αυτά μεταφέρονται στο αρχείο της αρχικής οθόνη (index.php) και μέσω ενός script τα δεδομένα αποθηκεύονται σε μεταβλητές της τοπικής μνήμης του περιηγητή, έτσι ώστε να είναι διαθέσιμες από κάθε επίπεδο του παιχνιδιού. Αυτό φαίνεται στον Κώδικα 5.

```
217 <script>
218
219
220
221 var data = <?php echo $_SESSION['playerData']; ?>;
222 console.log(data);
223
224 var userName = data[0].username;
225 var gender = data[0].gender;
226
227 if(gender !== '0')
228 {
229     localStorage.setItem("playerGender", gender);
230     document.getElementById("myModal").style.zIndex = "-100";
231 }
232
233 localStorage.setItem("explorerName", userName);
234 //level 1
235     var level1won = data[0].level1won;
236     var level1score = data[0].level1score;
237     var level1start = data[0].level1start;
238     var level1moveleft = data[0].level1moveleft;
239     var level1moveright = data[0].level1moveright;
240     var level1time = data[0].level1time;
241     var level1firsttimewon = data[0].level1firsttimewon;
242     var level1timesreseted = data[0].level1timesreseted;
243     var level1attempts = data[0].level1attempts;
244     var level1stars = data[0].level1stars;
245
246     localStorage.setItem("userName", userName);
247     localStorage.setItem("level1won", level1won);
248     localStorage.setItem("level1score", level1score);
249     localStorage.setItem("level1start", level1start);
250     localStorage.setItem("level1moveleft", level1moveleft);
251     localStorage.setItem("level1moveright", level1moveright);
252     localStorage.setItem("level1time", level1time);
253     localStorage.setItem("level1firsttimewon", level1firsttimewon);
254     localStorage.setItem("level1timesreseted", level1timesreseted);
255     localStorage.setItem("level1attempts", level1attempts);
256     localStorage.setItem("level1stars", level1stars);
257     :
```

Κώδικας 5: Αποθήκευση δεδομένων χρήστη στην τοπική μνήμη του περιηγητή (index.php)

Στο σημείο αυτό τα δεδομένα είναι αποθηκευμένα στην τοπική μνήμη του περιηγητή και μπορούν να χρησιμοποιηθούν από τα επίπεδα του παιχνιδιού. Τα δεδομένα που αντιστοιχούν στο εκάστοτε επίπεδο διαβάζονται μέσα από το αρχείο phaserScript.js μια φορά στην έναρξη του επιπέδου και ανανεώνονται με την επανεκκίνηση του. Στον Κώδικα 6 απεικονίζεται η αρχικοποίηση των δεδομένων.


```

1      var playerName = localStorage.getItem("explorerName") || "player";
2      var playerGender = localStorage.getItem("playerGender") || "girl";
3
4
5          var userName = localStorage.getItem("userName") || '';
6          var level1won = Number(localStorage.getItem("level1won")) || 0;
7          var level1score;
8          var level1start;
9          var level1moveleft;
10         var level1moveright;
11         var level1time;
12         var level1firsttimewon;
13         var level1timesreseted = Number(localStorage.getItem(
14             "level1timesreseted")) || 0;
15         var level1attempts;
16         var level1stars;

```

Κώδικας 6: Αρχικοποίηση δεδομένων χρήστη (Level1/phaserScirpt.js)

Στην Κώδικα 7 φαίνεται η ενημέρωση των δεδομένων κατά τη δημιουργία-επανεκκίνηση του κόσμου του παιχνιδιού.

```

61      // =====
62      // Create
63      create() {
64
65
66
67
68         level1score = Number(localStorage.getItem("level1score")) || 0;
69         level1start = Number(localStorage.getItem("level1start")) || 0;
70         level1moveleft = Number(localStorage.getItem("level1moveleft")) ||
71             0;
72         level1moveright = Number(localStorage.getItem("level1moveright"))
73             || 0;
74         level1time = Number(localStorage.getItem("level1time")) || 0;
75         level1firsttimewon = Number(localStorage.getItem(
76             "level1firsttimewon")) || 0;
77
78         level1attempts = localStorage.getItem("level1attempts") || ' ';
79         level1stars = Number(localStorage.getItem("level1stars")) || 0;

```

Κώδικας 7: Ανανέωση δεδομένων χρήστη κατά την δημιουργία του παιχνιδιού (Level1/phaserScirpt.js)

Έπειτα εφόσον ο χρήστης έχει σχεδιάσει και εκτελέσει τον αλγόριθμό του, μετά την ολοκλήρωση όλων των εντολών ενημερώνονται ανάλογα πρώτα οι μεταβλητές της τοπικής μνήμης του περιηγητή για το επίπεδο και έπειτα η βάση δεδομένων. Ο κώδικας που αναλαμβάνει τη συγκεκριμένη ενέργεια είναι ενσωματωμένος στη συνάρτηση update της κλάσης Phaser.Scene, η οποία τρέχει τον κώδικα ανά εικόνα (frame) του παιχνιδιού. Για τον λόγο αυτό ελέγχεται η στιγμή που θα τελειώσει ο αλγόριθμος

(blocklyScriptCompleted) και αν έχει εκτελεστεί μια και μοναδική φορά η ενημέρωση των δεδομένων της τοπικής μνήμης αλλά και της βάσης δεδομένων (runOnce). Αυτό δείχνει ο Κώδικας 8.

```
249     if(starCollected === 5 && blocklyScriptCompleted && !runOnce) {
250
251         if(level1won === 0){
252             localStorage.setItem("level1firsttimewon", level1time +
                time);
253         }
254
255         level1won = 1;
256
257     }
258
259     if(blocklyScriptCompleted && !runOnce){
260         ///////////////
261         localStorage.setItem("level1won", level1won);
262         localStorage.setItem("level1score", totalScore +
                blocklyPoints);
263         localStorage.setItem("level1start", startBlock);
264         localStorage.setItem("level1moveleft", moveLeft);
265         localStorage.setItem("level1moveright", moveRight);
266         localStorage.setItem("level1time", level1time + time);
267         localStorage.setItem("level1timesreseted",
                level1timesreseted + gameTimesReseted);
268         localStorage.setItem("level1attempts", level1attempts +
                blocks + "\n\n");
269         localStorage.setItem("level1stars", starCollected);
270         localStorage.setItem("level1badge", levelBadge);
271
272
273         const xhr = new XMLHttpRequest();
274         xhr.open("POST", "level1Update.php");
275         xhr.setRequestHeader("Content-type",
                "application/x-www-form-urlencoded");
276         xhr.send("username="+userName+"&level1won="+level1won+
                "&level1start="+startBlock+"&level1moveleft="+moveLeft+
                "&level1moveright="+moveRight+"&level1time="+level1time +
                time+"&level1score="+ (totalScore + blocklyPoints)+
                "&level1firsttimewon="+ Number(localStorage.getItem(
                "level1firsttimewon"))+"&level1timesreseted="+
                (level1timesreseted + gameTimesReseted)+"&level1attempts="+
                ( level1attempts + blocks + "\n\n")+ "'&level1stars="+
                starCollected + "&level1badge="+levelBadge+"&gender='"+
                playerGender+"'");
277
278         runOnce = true;
279         blocklyScriptCompleted = false;
280
281     }
```

Κώδικας 8: Ενημέρωση δεδομένων προόδου χρήστη στην τοπική μνήμη και αποστολή στον server (level1Update.php) για ενημέρωση της βάσης (Level1/phaserScirpt.js)

Όπως φαίνεται στις γραμμές 273-274 του Κώδικα 8 για τη μεταφορά των δεδομένων στον server (level1Update.php) χρησιμοποιείται το αντικείμενο XMLHttpRequest, το οποίο στέλνει τα δεδομένα από το υπόβαθρο χωρίς να ανανεώνεται η σελίδα. Έπειτα στο αρχείο level1Update.php γίνεται η ενημέρωση των κατάλληλων πεδίων της βάσης δεδομένων, όπως δείχνει ο Κώδικας 9.

```
index.php | phaserScript.js | level1Update.php
1 <?php
2
3 $username = $_POST['username'];
4 $level1won = $_POST['level1won'];
5 $level1start = $_POST['level1start'];
6 $level1moveleft = $_POST['level1moveleft'];
7 $level1moveright = $_POST['level1moveright'];
8 $level1time = $_POST['level1time'];
9 $level1stars = $_POST['level1stars'];
10 $level1attempts = $_POST['level1attempts'];
11 $level1score = $_POST['level1score'];
12 $level1firsttimewon = $_POST['level1firsttimewon'];
13 $level1timesreseted = $_POST['level1timesreseted'];
14 $level1badge = $_POST['level1badge'];
15
16 $gender = $_POST['gender'];
17
18
19
20 $db = mysqli_connect('localhost', 'root', '', 'blocklyscript');
21
22 $sql = "UPDATE users SET level1won=$level1won, level1start=$level1start,
level1moveleft=$level1moveleft, level1moveright=$level1moveright, level1time=
$level1time, level1stars=$level1stars, level1attempts=$level1attempts,
level1score=$level1score, level1firsttimewon=$level1firsttimewon,
level1timesreseted=$level1timesreseted, level1badge=$level1badge, gender=$gender
WHERE username='$username'";
23
24 mysqli_query($db, $sql);
25
26 ?>
```

**Κώδικας 9: Ενημέρωση δεδομένων των πεδίων της βάσης
(Level1/level1Update.php)**

4.4.4 Σχεδίαση επιπέδων παιχνιδιού


Για την ανάπτυξη του BlocklyScript μελετήθηκαν αρκετά διαδικτυακά παιχνίδια παρόμοιου τύπου και στα περισσότερα από αυτά με την επιτυχή ολοκλήρωση ενός επιπέδου ο χρήστης οδηγούταν μέσω ενός συνδέσμου σε μια νέα html σελίδα με το επόμενο επίπεδο. Η ίδια στρατηγική εφαρμόστηκε και στο BlocklyScript με στόχο την καλύτερη οργάνωση των επιπέδων και τη διαχείριση των μεταβλητών. Επιπλέον η συγκεκριμένη στρατηγική βοηθάει στην καλύτερη διαχείριση των μπλοκ εντολών του συντάκτη Blockly και δεν χρειάζονται ανανεώσεις στην εργαλειοθήκη του. Έτσι κάθε επίπεδο του παιχνιδιού είναι αυτόνομο και η οργάνωση του αποτελείται από επτά βασικά αρχεία.

1. index.html: Υπεύθυνο για την οργάνωση της σελίδας σε τέσσερις περιοχές (παιχνιδιού, συντάκτη Blockly, οδηγιών και αποτελέσματος προγράμματος).
2. levelXUpdate.php: Υπεύθυνο για την ενημέρωση των πεδίων της βάσης μετά την εκτέλεση του αλγορίθμου του χρήστη.

3. phaserScript.js: Υπεύθυνο για τη σχεδίαση της περιοχής παιχνιδιού και της λογικής του.
4. blocklyScript.js: Υπεύθυνο για την εκτέλεση του αλγορίθμου που θα δώσει ο χρήστης κατά την εκκίνηση του παιχνιδιού στην περιοχή του συντάκτη Blockly.
5. blocks_def.js: Κώδικας για τον ορισμό των μπλοκ εντολών.
6. blocks_stub.js: Κώδικας που ορίζει τη λειτουργικότητα των μπλοκ εντολών.
7. levelStyle.css: Υπεύθυνο για τον προσαρμοσμένο σχεδιασμό των επιπέδων σε διάφορες συσκευές.

4.4.4.1 Αρχείο *Index.html*

Το πρώτο αρχείο είναι το `index.html` και αποτελεί την βασική εμφάνιση της σελίδας αλλά και τη σύνδεση όλων των παραπάνω αρχείων με αυτήν. Στον Κώδικα 10 απεικονίζεται ο τρόπος σύνδεσης του πλαισίου ανάπτυξης 2D παιχνιδιών Phaser καθώς και η σύνδεση την βιβλιοθήκης Blockly που είναι υπεύθυνη για την σχεδίαση και την λειτουργικότητα των μπλοκ εντολών.



```

1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8">
5     <title>Page title</title>
6     <script src="../acorn_interpreter.js"></script>
7     <script src="../blockly_compressed.js"></script>
8     <script src="../javascript_compressed.js"></script>
9     <script src="../blocks_compressed.js"></script>
10    <script src="../en.js"></script>
11    <script src="blocks_def.js"></script>
12    <script src="blocks_stub.js"></script>
13    <script src="
14      https://cdn.jsdelivr.net/npm/phaser@3.20.0/dist/phaser-arcade-physics.min
15      .js"></script>
16    <link rel="stylesheet" href="../levelStyle.css">

```

Κώδικας 10: Σύνδεση Phaser και Blockly στην αρχική σελίδα `index.html` του πρώτου επιπέδου

Πιο συγκεκριμένα οι γραμμές 6-10 στον Κώδικα 10 αφορούν τη σύνδεση των αρχείων της βιβλιοθήκης Blockly, ενώ στην γραμμή 13 γίνεται η σύνδεση του Phaser. Οι γραμμές 11 και 12 στον ίδιο Κώδικα αφορούν τον ορισμό και τη λειτουργικότητα αντίστοιχα των προσαρμοσμένων μπλοκ που σχεδιάστηκαν αποκλειστικά για το συγκεκριμένο παιχνίδι. Επιπλέον προς το τέλος του αρχείου `index.html` γίνεται και η σύνδεση των αρχείων `blocklyScript.js` και `phaserScript.js`. Ο Κώδικας 11 δείχνει αυτή την σύνδεση.

```

111 <script src="blocklyScript.js"></script>
112 <script src="phaserScript.js"></script>

```

Κώδικας 11: Σύνδεση αρχείων blocklyScript.js και phaserScript.js με το αρχείο index.html του πρώτου επιπέδου

Συνεχίζοντας η διάταξη της οθόνης κάθε επιπέδου έχει χωριστεί σε τέσσερις βασικές περιοχές οι οποίες έχουν αναφερθεί αναλυτικά στο κεφάλαιο 3.4.2. Ωστόσο ενδιαφέρον παρουσιάζουν η περιοχή του παιχνιδιού (πάνω αριστερή γωνία στην οθόνη του επιπέδου) και η περιοχή του συντάκτη Blockly (πάνω δεξιά γωνία στην οθόνη του επιπέδου). Οι περιοχές αυτές συνδέονται άμεσα με τα αρχεία του Κώδικα 11. Όπως φαίνεται στον Κώδικα 12 οι δύο περιοχές είναι απλά δύο κενά τμήματα (divisions) στην αρχική σελίδα του επιπέδου και έχουν μόνο τα δικά του αναγνωριστικά (id). Με βάση τα αναγνωριστικά τους θα διαμορφωθούν αυτές οι περιοχές αργότερα μέσω των δύο αρχείων του Κώδικα 11.

```

26 <div>
27   <div class="leftpanel" id="phaserDiv" style="border: 5px red
28     solid; border-radius: 25px;">
29     <div id="phaserDivCanvas"></div>
30   </div>
31   <div class="rightpanel" id="blocklyDiv" style="border: 5px blue
32     solid;"></div>
</div>

```

Κώδικας 12: Περιοχή παιχνιδιού και συντάκτη Blockly πρώτου επιπέδου (index.html)

Τέλος στο αρχείο index.html, αφού οριστούν τα μπλοκ στο αρχείο blocks_def.js, γίνεται η αρχικοποίηση της εργαλειοθήκης του συντάκτη Blockly, δηλαδή δημιουργούνται τα διαθέσιμα μπλοκ που θα μπορεί να χρησιμοποιήσει ο παίκτης στο συγκεκριμένο επίπεδο. Αυτό φαίνεται στον Κώδικα 13.

```

72 <xml id="toolbox" style="display: none">
73   <block type="start_point"></block>
74   <block type="move_left"></block>
75   <block type="move_right"></block>
76 </xml>

```

Κώδικας 13: Αρχικοποίηση διαθέσιμων μπλοκ εντολών της εργαλειοθήκης του συντάκτη Blockly του πρώτου επιπέδου (index.html)

4.4.4.2 Αρχεία *blocks_def.js* και *blocks_stub.js*

Το αρχείο *blocks_def.js* όπως έχει ήδη αναφερθεί περιέχει τον ορισμό των μπλοκ και αφορά την οπτική σχεδίαση του. Πιο συγκεκριμένα ορίζει τον τρόπο σύνδεσης του μπλοκ με άλλα μπλοκ που βρίσκονται πριν και μετά από αυτό, αν το συγκεκριμένο μπλοκ περιμένει κάποια είσοδο ή εάν το ίδιο αποτελεί είσοδο δεδομένων από/για κάποιο άλλο μπλοκ, το χρώμα του μπλοκ, μια υπόδειξη για τον χρήστη κ.α. Ο Κώδικας 14 δείχνει τον ορισμό των τριών μπλοκ του πρώτου επιπέδου που υπάρχουν και στον Κώδικα 13.



```
1  Blockly.Blocks['move_left'] = {
2  init: function() {
3    this.appendDummyInput ()
4    .appendField("move left")
5    .appendField(new Blockly.FieldImage("../assets/arrowLeft.png", 15, 15, {
6      alt: "Left", flipRtl: "FALSE" }));
7    this.setPreviousStatement(true, null);
8    this.setNextStatement(true, null);
9    this.setColour(300);
10   this.setTooltip("Move player to the left.");
11   this.setHelpUrl("");
12  }
13 };
14
15 Blockly.Blocks['move_right'] = {
16 init: function() {
17   this.appendDummyInput ()
18   .appendField("move right")
19   .appendField(new Blockly.FieldImage("../assets/arrowRight.png", 15, 15, {
20     alt: "Right", flipRtl: "FALSE" }));
21   this.setPreviousStatement(true, null);
22   this.setNextStatement(true, null);
23   this.setColour(300);
24   this.setTooltip("Move player to the right.");
25   this.setHelpUrl("");
26 }
27 };
28
29 Blockly.Blocks['start_point'] = {
30 init: function() {
31   this.appendDummyInput ()
32   .appendField("START")
33   .appendField(new Blockly.FieldImage("../assets/startFlag.png", 15, 15, {
34     alt: "Start", flipRtl: "FALSE" }));
35   this.setNextStatement(true, null);
36   this.setColour(120);
37   this.setTooltip("This is your starting point");
38   this.setHelpUrl("");
39 }
40 };
41
```

Κώδικας 14: Ορισμός μπλοκ πρώτου επιπέδου (*block_defs.js*)

Στην συνέχεια στο αρχείο *blocks_stub.js* ορίζεται η λειτουργικότητα των μπλοκ εντολών. Στο σημείο αυτό για να οριστεί η λειτουργικότητα ενός μπλοκ υπάρχουν τρία βασικά σημεία που θα οδηγήσουν στη σωστή εκτέλεση των εντολών. Τα δύο σημεία βρίσκονται στο αρχείο *blocks_stub.js* ενώ το τρίτο στο αρχείο *blocklyScript.js*. Ξεκινώντας από τα δύο πρώτα σημεία που βρίσκονται στο αρχείο *blocks_stub.js* το πρώτο αφορά τον κώδικα που θα επιστρέψει το μπλοκ στον διερμηνέα (*interpreter*) που υπάρχει στο αρχείο *blocklyScript.js* και είναι υπεύθυνος για τη σωστή εκτέλεση του κώδικα (για τον διερμηνέα έχει γίνει σχετική αναφορά στο κεφάλαιο 2.4.3). Στον

Κώδικα 15 φαίνεται ο κώδικας που επιστρέφει το μπλοκ της κίνησης του χαρακτήρα προς τα αριστερά (move_left).

```
index.html | blocks_def.js | blocks_stub.js | blocklyScript.js
1 Blockly.JavaScript['move_left'] = function(block) {
2   // TODO: Assemble JavaScript into code variable.
3   var code = 'goLeftBlock();\n'; // make sure to keep the \n or it will not
4   work correctly
5   return code;
6 };
```

Κώδικας 15: Κώδικας του μπλοκ move_left (blocks_stub.js)

Όπως δείχνει η γραμμή 3 του Κώδικα 15 ο κώδικας που επιστρέφει το μπλοκ move_left είναι μια συνάρτηση "goLeftBlock()". Ο διερμηνέας όμως δεν γνωρίζει τη λειτουργικότητα αυτής της συνάρτησης. Έτσι το επόμενο (δεύτερο) βήμα είναι να οριστεί η συνάρτηση "goLeftBlock();" για να μπορεί να χρησιμοποιηθεί από τον διερμηνέα. Η λειτουργικότητα της συνάρτησης φαίνεται στον Κώδικα 16.

```
8 function initInterpreterGoLeft(interpreter, scope) {
9   // Ensure function name does not conflict with variable names.
10
11   Blockly.JavaScript.addReservedWords('goLeftBlock');
12   var wrapper = interpreter.createAsyncFunction(
13     function(callback) {
14       goToTheLeft = true;
15       setTimeout(function(){
16         // next stop moving to the left
17         goToTheLeft = false;
18         callback(); }, 1000);
19     });
20   interpreter.setProperty(scope, 'goLeftBlock', wrapper);
21 }
22
23 }
```

Κώδικας 16: API του μπλοκ move_left (blocks_stub.js)

Στον Κώδικα 16 η συνάρτηση "initInterpreterGoLeft" αποτελεί το API του μπλοκ move_left. Στο API αυτό δεσμεύεται το όνομα της συνάρτησης "goLeftBlock" για να μην υπάρχουν συγκρούσεις με τυχόν παρόμοια ονόματα μεταβλητών. Στη συνέχεια ορίζεται η ασύγχρονη συνάρτηση wrapper η οποία θα συνδεθεί με την goLeftBlock στη γραμμή 22 και θα τεθεί ως ιδιότητα του διερμηνέα. Η συνάρτηση goLeftBlock αναλαμβάνει τη μετακίνηση του ήρωα προς τα αριστερά για 1 δευτερόλεπτο που στο παιχνίδι μεταφράζεται ως 100 pixel.

Το τρίτο και τελευταίο βήμα αφορά την κλήση του API των μπλοκ για την αρχικοποίηση του διερμηνέα. Το βήμα αυτό θα μελετηθεί στην επόμενη ενότητα που αναφέρει το αρχείο `blocklyScript.js` (κεφάλαιο: 4.4.4.3).

4.4.4.3 Αρχείο `blocklyScript.js`

Το αρχείο αυτό είναι υπεύθυνο για τη διαχείριση του συντάκτη Blockly. Πιο συγκεκριμένα αναλαμβάνει να διαβάσει τα μπλοκ εντολών που θα τοποθετήσει ο χρήστης στην περιοχή εργασίας του συντάκτη. Αφού διαβάσει τα μπλοκ τα τοποθετεί στον διερμηνέα (interpreter) ο οποίος μέσω του API κάθε μπλοκ θα παράγει συντακτικά σωστά κώδικα σε JavaScript, ο οποίος όταν εκτελεστεί θα επηρεάσει την κίνηση του χαρακτήρα στο παιχνίδι.

Το πρώτο βήμα είναι να τοποθετηθεί ο συντάκτης Blockly στην αρχική σελίδα του επιπέδου και πιο συγκεκριμένα στην περιοχή πάνω δεξιά. Η ενσωμάτωση φαίνεται στον Κώδικα 17. Στον συγκεκριμένο Κώδικα βλέπει κανείς μερικές από τις ιδιότητες που ρυθμίζουν τον συντάκτη Blockly. Αξίζει να αναφερθεί η γραμμή 13 (toolbox) του Κώδικα 17 όπου αποτελεί την αρχικοποίηση της εργαλειοθήκης του συντάκτη με τα μπλοκ που όρισε ο προγραμματιστής στον Κώδικα 13 για το πρώτο επίπεδο του παιχνιδιού. Επίσης σημαντική είναι και η ιδιότητα της γραμμής 24 που περιέχει το μέγιστο αριθμό των στιγμιότυπων για κάθε μπλοκ (`maxInstances`). Στο παράδειγμα φαίνεται ότι ο χρήστης μπορεί να χρησιμοποιεί μόνο ένα μπλοκ εκκίνησης (Start block) κάθε φορά.


```

11  var blocklyDiv = document.getElementById('blocklyDiv');
12      var demoWorkspace = Blockly.inject('blocklyDiv',
13          {
14              toolbox: document.getElementById('toolbox'),
15              collapse : false,
16              comments : false,
17              disable : false,
18              maxBlocks : Infinity,
19              trashcan : true,
20              horizontalLayout : false,
21              toolboxPosition : 'start',
22              css : true,
23              media : 'https://blockly-demo.appspot.com/static/media/',
24              rtl : false,
25              maxInstances: {
26                  'start_point': 1
27              },
28              sounds : true,
29              oneBasedIndex : true,
30              grid : {
31                  spacing : 40,
32                  length : 9,
33                  colour : 'red',
34                  snap : true
35              },
36              zoom : {
37                  controls : true,
38                  wheel : true,
39                  startScale : 1,
40                  maxScale : 3,
41                  minScale : 0.3,
42                  scaleSpeed : 1.1
43              }
44          });

```

Κώδικας 17: Ενσωμάτωση συντάκτη Blockly στην αρχική σελίδα (index.html) του πρώτου επιπέδου (blocklyScript.js)

Συνεχίζοντας για να μπορέσει ο κώδικας των μπλοκ που θα παράγει ο διερμηνέας να επηρεάσει τη λογική του παιχνιδιού χρησιμοποιήθηκαν καθολικές μεταβλητές (global variables) οι οποίες λειτουργούν ως σημαίες και δηλώνουν μια αλλαγή κατάστασης του χαρακτήρα του παιχνιδιού. Αυτές οι μεταβλητές βρίσκονται στο αρχείο phaserScript.js του κάθε επιπέδου και θα αναφερθούν αργότερα. Στο αρχείο blocklyScript.js του κάθε επιπέδου οι πιο σημαντικές καθολικές μεταβλητές παρουσιάζονται στον Πίνακα 32.

Πίνακας 32: Σημαντικότερες καθολικές μεταβλητές στο αρχείο BlocklyScript.js

Καθολική μεταβλητή	Περιγραφή
latestCode	Περιέχει τον κώδικα που επιστρέφουν τα μπλοκ από τον αλγόριθμο που έχει σχεδιάσει ο παίκτης.
BlocklyScriptCompleted	Ενημερώνει το παιχνίδι ότι η εκτέλεση του αλγορίθμου που έδωσε ο χρήστης έχει τελειώσει.
BlocklyPoints	Είναι ο βαθμός που θα κερδίσει ο παίκτης ανάλογα με τη λύση που θα δώσει.
blocks	Είναι μια λίστα με τα μπλοκ που έδωσε ο χρήστης για εκτέλεση και με την σειρά που τα έδωσε.
outputArea	Είναι το πλαίσιο κειμένου για την περιοχή αποσφαλμάτωσης της αρχικής οθόνης.
runButton	Αποτελεί το κουμπί "Run Game" της αρχικής οθόνης του επιπέδου.
myInterpreter	Είναι ο διερμηνέας του επιπέδου, ο οποίος θα αναλάβει την σωστή εκτέλεση του κώδικα των μπλοκ εντολών.
Μεταβλητές ακέραιου αριθμού για κάθε διαθέσιμο τύπο μπλοκ στο επίπεδο	Με την εκτέλεση του αλγορίθμου αποθηκεύεται το πλήθος των μπλοκ από κάθε τύπο.

Πέρα από τις μεταβλητές το αρχείο είναι οργανωμένο σε συναρτήσεις για την ευκολότερη κατανόηση του κώδικα. Στον Πίνακα 33 παρουσιάζονται οι βασικές συναρτήσεις του αρχείου BlocklyScript.js.

Πίνακας 33: Βασικές συναρτήσεις στο αρχείο blocklyScript.js

Συνάρτηση	Περιγραφή
initApi(interpreter, scope)	<p>Αρχείο: Level1/blocklyScript.js Γραμμές Κώδικα: 58-98</p> <p>Επειδή ο διερμηνέας της JavaScript είναι ένας αμμοδόχος (sandbox) που είναι τελείως απομονωμένος από τον περιηγητή, πρέπει το API από κάθε μπλοκ να ενσωματωθεί στον διερμηνέα. Η συνάρτηση initApi καλεί τα API των μπλοκ και τα ενσωματώνει στον διερμηνέα.</p> <pre data-bbox="603 705 1342 779"> 80 initInterpreterGoLeft(interpreter, scope); 81 initInterpreterGoRight(interpreter, scope); </pre> <p>Ενσωμάτωση API μπλοκ <code>move_left</code> και <code>move_right</code> στον διερμηνέα</p>
highlightBlock(id)	<p>Αρχείο: Level1/blocklyScript.js Γραμμές Κώδικα: 104-107</p> <p>Είναι υπεύθυνη για την επισήμανση του μπλοκ που εκτελείται την συγκεκριμένη χρονική στιγμή</p>
resetStepUi(clearOutput)	<p>Αρχείο: Level1/blocklyScript.js Γραμμές Κώδικα: 109-117</p> <p>Είναι υπεύθυνη για να καθαρίσει την επισήμανση του μπλοκ που εκτελείται αλλά και το κείμενο της περιοχής αποσφαλμάτωσης.</p>
generateCodeAndLoadIntoInterpreter()	<p>Αρχείο: Level1/blocklyScript.js Γραμμές Κώδικα: 119-126</p> <p>Παράγει τον JavaScript κώδικα των μπλοκ και τον περνάει στον διερμηνέα</p>
resetInterpreter()	<p>Αρχείο: Level1/blocklyScript.js Γραμμές Κώδικα: 128-134</p> <p>Αδειάζει τον διερμηνέα και τον επαναρυθμίζει.</p>
runCode()	<p>Αρχείο: Level1/blocklyScript.js Γραμμές Κώδικα: 140-247</p> <p>Μόλις ο χρήστης πατήσει το κουμπί Run Game ξεκινάει αυτόματα η εκτέλεση αυτής της συνάρτησης.</p>

Αρχικά καθαρίζει η περιοχή της αποσφαλμάτωσης κάτω δεξιά την οθόνη και το κουμπί Run Game απενεργοποιείται. Έπειτα προσμετρούνται το πλήθος από κάθε τύπο μπλοκ που έδωσε ο χρήστης στον αλγόριθμό του και θα εμφανιστούν αργότερα στην περιοχή της αποσφαλμάτωσης όταν τελειώσει η εκτέλεση του αλγορίθμου. Στη συνέχεια περνάει ο κώδικας του αλγορίθμου στον διερμηνέα και ξεκινάει η εκτέλεση του κώδικα. Με το που τελειώσει η εκτέλεση του κώδικα ενημερώνεται η μεταβλητή blocklyScriptCompleted, αδειάζει ο διερμηνέας και σταματάει η επισήμανση του τελευταίου μπλοκ προς εκτέλεση.

Στο τέλος κάθε αρχείου blocklyScript.js υπάρχει ένας ακροατής που ακούει κάθε αλλαγή που γίνεται στο περιβάλλον εργασίας του συντάκτη Blockly και με κάθε αλλαγή που γίνεται παράγει τον αντίστοιχο κώδικα των μπλοκ και τον ενσωματώνει στον διερμηνέα. Αυτό δείχνει ο Κώδικας 18.

```
225 demoWorkspace.addChangeListener(function (event) {
226     if (!(event instanceof Blockly.Events.Ui)) {
227         // Something changed. Parser needs to be reloaded.
228         resetInterpreter();
229         generateCodeAndLoadIntoInterpreter();
230     }
231 });
```

Κώδικας 18: Ακροατής αλλαγών της περιοχής εργασίας του συντάκτη Blockly (blocklyScript.js)

4.4.4.4 Αρχείο phaserScript.js

Το αρχείο phaserScript.js αφορά την οργάνωση και την λογική του παιχνιδιού. Στο συγκεκριμένο αρχείο αξιοποιείται το πλαίσιο ανάπτυξης παιχνιδιών Phaser. Χρησιμοποιώντας το Phaser το πρώτο βήμα είναι τα καταχωρηθεί στην περιοχή του παιχνιδιού της σελίδας η οθόνη του παιχνιδιού. Στον Κώδικα 19 παρουσιάζονται οι ρυθμίσεις της οθόνης του παιχνιδιού.

```

870 var config = {
871     type: Phaser.AUTO,
872     width: 800,
873     height: 600,
874     scale: {
875         parent: document.getElementById("phaserDivCanvas"),
876         mode: Phaser.Scale.FIT,
877         autoCenter: Phaser.Scale.CENTER_BOTH
878     },
879     physics: {
880         default: 'arcade',
881         arcade: {
882             gravity: { y: 700 },
883             debug: false
884         }
885     },
886     scene: Level11
887 };
888
889 var game = new Phaser.Game(config);

```

Κώδικας 19: Ρυθμίσεις οθόνης παιχνιδιού και σύνδεση με την σελίδα εντέκατου επιπέδου (Level11/phaserScript.js)

Τρεις σημαντικές ρυθμίσεις που εμφανίζονται στον Κώδικα 19 είναι η *parent* (γραμμή 875), η *physics* (γραμμή 879) και η *scene* (γραμμή 886). Η πρώτη ρύθμιση συμβάλει στη σύνδεση της οθόνης του παιχνιδιού με το αντίστοιχο τμήμα της σελίδας του επιπέδου (*index.html*). Η δεύτερη ρύθμιση ενσωματώνει τη μηχανή για τη διαχείριση των κανόνων της φυσικής στο παιχνίδι και η τρίτη ρύθμιση που ονομάζεται σκηνή δηλώνει πιο επίπεδο θα ενσωματωθεί στην οθόνη. Στο παράδειγμα του Κώδικα 19 η σκηνή του παιχνιδιού είναι η εντεκάτη και αποτελεί μια υποκλάση της *Phaser.Scene*.

Η σκηνή αναλαμβάνει να φορτώσει τα αρχεία των γραφικών και των ήχων που θα χρησιμοποιηθούν σε αυτήν, δημιουργεί τον κόσμο και τοποθετεί τους χαρακτήρες στο επίπεδο και ελέγχει τη λογική του παιχνιδιού. Από τη μεριά του ο παίκτης ελέγχει την κίνηση του χαρακτήρα στη σκηνή τοποθετώντας τα διαθέσιμα μπλοκ τα οποία με τη σειρά τους αλλάζουν τις τιμές των καθολικών μεταβλητών του επιπέδου. Οι πιο σημαντικές μεταβλητές παρουσιάζονται στον Πίνακα 34.

Πίνακας 34: Σημαντικότερες καθολικές μεταβλητές στο αρχείο phaserScript.js

Καθολική μεταβλητή	Περιγραφή
player	Ο/Η εξερευνητής του παιχνιδιού
levelScore	Το σκορ του παιχνιδιού
platforms	Η ομάδα (group) που περιέχει όλες τις πλατφόρμες οι οποίες λειτουργούν σαν έδαφος για να πατήσει ο/η εξερευνητής
enemies	Η ομάδα (group) που περιέχει όλους τους αντιπάλους (ιπτάμενους εξωγήινους) του/της εξερευνητή/τριας
spikes	Η ομάδα (group) που περιέχει τα εμπόδια (καρφιά εδάφους) του παιχνιδιού
stars	Η ομάδα (group) που περιέχει όλα τα αστέρια του επιπέδου
starCollected	Ο αριθμός των αστεριών που κατάφερε ο χρήστης να συλλέξει
laserBeams	Η ομάδα (group) που περιέχει τις ακτίνες φωτός που μπορεί να δημιουργήσει ο/η εξερευνητής/τρια
wallHits	Ένας ακέραιος αριθμός που μετράει το πλήθος των ακτινών φωτός που έχει δεχτεί ένα τμήμα τοίχου
wallIsBroken	Ελέγχει εάν ο τοίχος έχει σπάσει ή όχι
goToTheRight	Ζητάει από τον/την εξερευνητή/τρια να κινηθεί προς τα δεξιά
goToTheLeft	Ζητάει από τον/την εξερευνητή/τρια να κινηθεί προς τα αριστερά
goJump	Ζητάει από τον/την εξερευνητή/τρια να κάνει ένα άλμα

Στην συνέχεια στον Πίνακα 35 φαίνονται οι συναρτήσεις του αρχείου Level11/phaserScript.js

Πίνακας 35: Συναρτήσεις στο αρχείο Level11/phaserScript.js

Συνάρτηση	Περιγραφή
preload()	Αρχείο: Level11/phaserScript.js Γραμμές Κώδικα: 41-67 Είναι υπεύθυνη για τη φόρτωση όλων των αρχείων ήχου και εικόνας που θα χρησιμοποιηθούν από το επίπεδο.
create()	Αρχείο: Level11/phaserScript.js Γραμμές Κώδικα: 71-136 Δημιουργεί τον κόσμο θέτοντας τις αρχικές συντεταγμένες των εμποδίων, του/της εξερευνητή/τριας, των αντιπάλων και γενικότερα των αντικειμένων του επιπέδου.
createPlatforms()	Αρχείο: Level11/phaserScript.js Γραμμές Κώδικα: 143-163 Υπεύθυνη για τη δημιουργία του εδάφους πάνω στο οποίο πατάει ο/η εξερευνητής/τρια.
createParticles()	Αρχείο: Level11/phaserScript.js Γραμμές Κώδικα: 165-176 Δημιουργεί τα εφέ που εμφανίζονται μόλις ο/η εξερευνητής/τρια συλλέξει ένα αστέρι
createPlayer()	Αρχείο: Level11/phaserScript.js Γραμμές Κώδικα: 179-209 Δημιουργεί τον/την εξερευνητή/τρια του επιπέδου καθώς και τις εικόνες για την κίνηση προς τα δεξιά και τα αριστερά
createEnemies()	Αρχείο: Level11/phaserScript.js Γραμμές Κώδικα: 213-250 Δημιουργεί τους αντιπάλους (ιπτάμενους εξωγήινους) του επιπέδου καθώς και τις εικόνες για την κίνηση προς τα δεξιά και τα αριστερά
moveEnemies()	Αρχείο: Level11/phaserScript.js

	<p>Γραμμές Κώδικα: 252-295</p> <p>Αναλαμβάνει την οριζόντια κίνηση των αντιπάλων στο επίπεδο</p>
moveBalloons(balloon)	<p>Αρχείο: Level11/phaserScript.js</p> <p>Γραμμές Κώδικα: 297-315</p> <p>Αναλαμβάνει την κάθετη κίνηση των αιωρούμενων μπαλονιών</p>
createSpikes()	<p>Αρχείο: Level11/phaserScript.js</p> <p>Γραμμές Κώδικα: 317-341</p> <p>Δημιουργεί τα εμπόδια του επιπέδου.</p>
createLaserBeams()	<p>Αρχείο: Level11/phaserScript.js</p> <p>Γραμμές Κώδικα: 350-354</p> <p>Δημιουργεί την ομάδα (group) που περιέχει τις ακτίνες φωτός</p>
createInstanceOFLaserBeam()	<p>Αρχείο: Level11/phaserScript.js</p> <p>Γραμμές Κώδικα: 356-373</p> <p>Δημιουργεί ένα στιγμιότυπο ακτίνας φωτός με το αντίστοιχο μπλοκ εντολής</p>
createStars()	<p>Αρχείο: Level11/phaserScript.js</p> <p>Γραμμές Κώδικα: 375-420</p> <p>Δημιουργεί τα αστέρια του επιπέδου</p>
collectStar (player, star)	<p>Αρχείο: Level11/phaserScript.js</p> <p>Γραμμές Κώδικα: 422-506</p> <p>Αναλαμβάνει του ελέγχους για τη συλλογή του αστεριού ή μπαλονιού το οποίο ακουμπάει ο/η εξερευνητής/τρια</p>
update()	<p>Αρχείο: Level11/phaserScript.js</p> <p>Γραμμές Κώδικα: 514-555</p> <p>Η συγκεκριμένη συνάρτηση εκτελείται σε κάθε εικόνα (frame) του παιχνιδιού. Ουσιαστικά αποτελεί τη λογική του παιχνιδιού και μέσα σε αυτήν γίνονται οι έλεγχοι για την κίνηση των κινούμενων</p>

	<p>χαρακτήρων, για τις συγκρούσεις μεταξύ αυτών και για τα αποτελέσματα του παιχνιδιού μετά την ολοκλήρωση του αλγορίθμου που θα δώσει ο χρήστης.</p>
hitAlien(beam, alien)	<p>Αρχείο: Level11/phaserScript.js Γραμμές Κώδικα: 557-568 Ελέγχει τη σύγκρουση μεταξύ ενός εξωγήινου και μιας ακτίνας φωτός</p>
hitWall(beam, wall)	<p>Αρχείο: Level11/phaserScript.js Γραμμές Κώδικα: 570-591 Ελέγχει τη σύγκρουση μεταξύ ενός τοίχου και μιας ακτίνας φωτός</p>
hitBomb(player, bomb)	<p>Αρχείο: Level11/phaserScript.js Γραμμές Κώδικα: 593-605 Ελέγχει τη σύγκρουση μεταξύ ενός ιπτάμενου εξωγήινου και του/της εξερευνητή/τριας.</p>
hitSpike(player, spike)	<p>Αρχείο: Level11/phaserScript.js Γραμμές Κώδικα: 607-621 Ελέγχει τη σύγκρουση μεταξύ μιας σειράς από καρφιά και του/της εξερευνητή/τριας.</p>
checkGameEndResult()	<p>Αρχείο: Level11/phaserScript.js Γραμμές Κώδικα: 623-821 Ελέγχει το αποτέλεσμα μετά την εκτέλεση του αλγορίθμου που έδωσε ο χρήστης εμφανίζοντας τα κατάλληλα μηνύματα στην περιοχή της αποσφαλμάτωσης. Επιπλέον ενημερώνει τη βάση δεδομένων με τα αποτελέσματα της προσπάθειας του χρήστη.</p>
losePoints()	<p>Αρχείο: Level11/phaserScript.js Γραμμές Κώδικα: 831-838 Υπολογίζει τους πόντους που θα χάσει ο παίκτης από το συνολικό του σκορ βάσει του πλήθους των</p>

	επαναλήψεων του επιπέδου
resetGame()	Αρχείο: Level11/phaserScript.js Γραμμές Κώδικα: 843-858 Αναλαμβάνει την επανεκκίνηση του επιπέδου.
checkWall()	Αρχείο: Level11/phaserScript.js Γραμμές Κώδικα: 860-865 Ελέγχει εάν ο τοίχος έσπασε ή όχι.
updateTimer()	Αρχείο: Level11/phaserScript.js Γραμμές Κώδικα: 138-141 Ανανεώνει ανά δευτερόλεπτο το κείμενο με τον χρόνο που αφιέρωσε ο χρήστης στο επίπεδο.

5 Αξιολόγηση παιχνιδιού

Για την πιλοτική αξιολόγηση του παιχνιδιού στάλθηκε μια πρόσκληση μέσω ηλεκτρονικού ταχυδρομείου σε δασκάλους/ες της Πρωτοβάθμιας και Δευτεροβάθμιας εκπαίδευσης. Στους συμμετέχοντες δόθηκε ένα διάστημα δεκαπέντε ημερών για να παίξουν το παιχνίδι διαδικτυακά ακολουθώντας τον σύνδεσμο που περιείχε η πρόσκληση και έπειτα να αξιολογήσουν το παιχνίδι απαντώντας ανώνυμα σε ένα ειδικά σχεδιασμένο ερωτηματολόγιο.

5.1 Μεθοδολογία

Το ερωτηματολόγιο που χρησιμοποιήθηκε βασίζεται στο μοντέλο αξιολόγησης εκπαιδευτικών παιχνιδιών MEEGA+, το οποίο σχεδιάστηκε από τους Petri et al. (2018). Το μοντέλο MEEGA+ αφορά κυρίως την αξιολόγηση εκπαιδευτικών παιχνιδιών που στοχεύουν στην εκπαίδευση των χρηστών πάνω στον τομέα της Επιστήμης των Υπολογιστών. Επιπλέον αξίζει να σημειωθεί ότι οι ερωτήσεις διαμορφώθηκαν με τέτοιο τρόπο ώστε να απευθύνονται στους δασκάλους/ες και όχι στους μαθητές.

Πιο συγκεκριμένα το MEEGA+ μοντέλο αξιολογεί την ποιότητα του εκπαιδευτικού παιχνιδιού χωρίζοντας τις ερωτήσεις που περιέχει σε δύο ποιοτικούς παράγοντες. Ο πρώτος ποιοτικός παράγοντας αφορά την εμπειρία του παίκτη (Player Experience) παίζοντας το παιχνίδι και περιλαμβάνει υποκατηγορίες ερωτήσεων που αξιολογούν την εστιασμένη προσοχή (Focused Attention) του παίκτη, τη διασκέδαση (Fun) του, τις προκλήσεις (Challenge) που αντιμετώπισε, την κοινωνική αλληλεπίδραση (social interaction) του με άλλους παίκτες, την ικανοποίηση (Satisfaction) του, την αυτοπεποίθηση (Confidence) του, τη συνάφεια (Relevance) του παιχνιδιού με ένα συγκεκριμένο πεδίο που αποτελεί και τον στόχο του παιχνιδιού, τα μαθησιακά αποτελέσματα (Perceived Learning) του παιχνιδιού και την προστασία του χρήστη από σφάλματα (User error protection). Ο δεύτερος ποιοτικός παράγοντας αφορά την χρηστικότητα (Usability) και οι ερωτήσεις που περιλαμβάνει αξιολογούν την ικανότητα μάθησης (Learnability), τη λειτουργικότητα (Operability), την αισθητική (Aesthetics) και την προσιτότητα (Accessibility).

Με βάση αυτά το ερωτηματολόγιο του παιχνιδιού χωρίζεται σε τέσσερις ενότητες:

- Η πρώτη ενότητα αφορά τα *δημογραφικά δεδομένα* και περιλαμβάνει έξι ερωτήσεις.
- Η δεύτερη ενότητα αφορά την *χρηστικότητα* του παιχνιδιού και περιέχει εννιά ερωτήσεις.
- Η τρίτη ενότητα αφορά την *εμπειρία του παίκτη* και περιλαμβάνει εικοσιοκτώ ερωτήσεις και
- η τελευταία ενότητα περιλαμβάνει δύο προαιρετικές ερωτήσεις ανοιχτού τύπου που αφορούν την αξιολόγηση των δυνατών και των αδύναμων στοιχείων του παιχνιδιού.

Στο ερωτηματολόγιο δεν συμπεριλήφθηκαν οι ερωτήσεις του μοντέλου MEEGA+ που αφορούν στην κοινωνική αλληλεπίδραση με άλλους παίκτες, μιας και το στοιχείο αυτό δεν υποστηρίζεται από το προς αξιολόγηση παιχνίδι. Για τις ερωτήσεις της δεύτερης και τρίτης ενότητας χρησιμοποιήθηκε η κλίμακα Likert από το -2 έως το 2 στην οποία το -2 ισοδυναμεί με το "Διαφωνώ κάθιστα", το -1 ισοδυναμεί με το "Διαφωνώ", το 0 ισοδυναμεί με το "Ούτε συμφωνώ ούτε διαφωνώ", το 1 ισοδυναμεί με το "Συμφωνώ" και το 2 ισοδυναμεί με το "Συμφωνώ απόλυτα". Τέλος για την δεύτερη και τρίτη ενότητα υπολογίστηκαν οι διάμεσες τιμές και τα ποσοστά των απαντήσεων σε κάθε ερώτηση.

5.2 Συμμετέχοντες

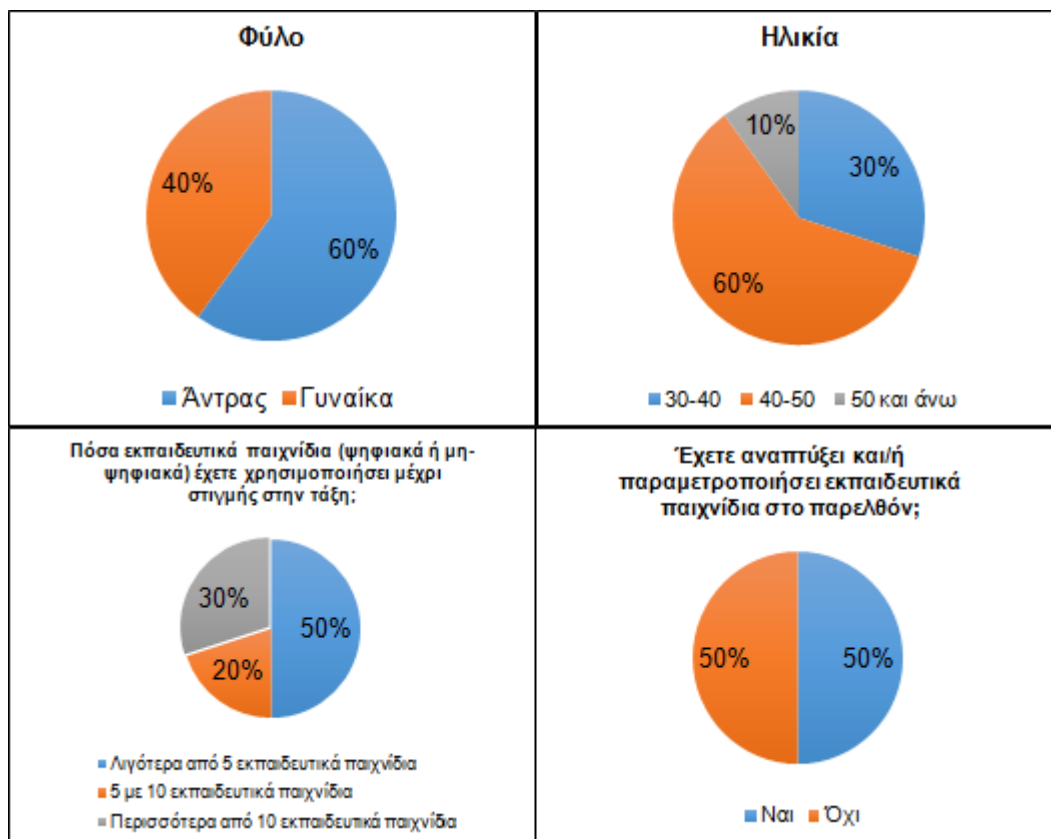
Όπως ήδη αναφέρθηκε στην πιλοτική αξιολόγηση του παιχνιδιού συμμετείχαν δάσκαλοι/ες Πληροφορικής της Πρωτοβάθμιας και Δευτεροβάθμιας εκπαίδευσης.

Συνολικά 10 δάσκαλοι/ες έπαιξαν το παιχνίδι BlocklyScript και έπειτα αξιολόγησαν την αναμενόμενη συνεισφορά του στην εκπαίδευση των μαθητών. Στην ενότητα 5.3 παρουσιάζονται αναλυτικά τα αποτελέσματα της αξιολόγησης.

5.3 Αποτελέσματα αξιολόγησης

5.3.1 Δημογραφικά δεδομένα

Στην αξιολόγηση συμμετείχαν δέκα έμπειροι εκπαιδευτικοί Πληροφορικής από τους οποίους το 30% διδάσκει σε παιδιά δημοτικού, το 30% διδάσκει σε παιδιά γυμνασίου και το 40% διδάσκει σε παιδιά λυκείου. Στην Εικόνα 79 παρουσιάζονται τα δημογραφικά στοιχεία των συμμετεχόντων και αξίζει να σημειωθεί ότι το 70% είναι άνω των 40 ετών (60% 40-50, 10% 50 ή μεγαλύτερος/η), στοιχείο που υποδηλώνει ότι έχουν αρκετά χρόνια εμπειρίας στην εκπαίδευση μαθητών. Επίσης σημαντικό δεδομένο είναι ότι το 50% έχει χρησιμοποιήσει πάνω από πέντε εκπαιδευτικά παιχνίδια (20% από 5 έως 10, 30% πάνω από 10) στην τάξη και είναι σε θέση να αξιολογήσουν τα θετικά και τα αρνητικά στοιχεία ενός τέτοιου τύπου παιχνιδιού. Τέλος, το 50% των εκπαιδευτικών έχει αναπτύξει ή προσαρμόσει ένα εκπαιδευτικό παιχνίδι και γνωρίζει τον τρόπο που πρέπει να σχεδιαστεί για να μπορέσει να πετύχει τους στόχους του. Παρόλο λοιπόν το μικρό δείγμα συμμετεχόντων στη μελέτη, αξίζει να σημειωθεί ότι πρόκειται για εκπαιδευτικούς που έχουν πλούσια εμπειρία στο πεδίο των εκπαιδευτικών παιχνιδιών.



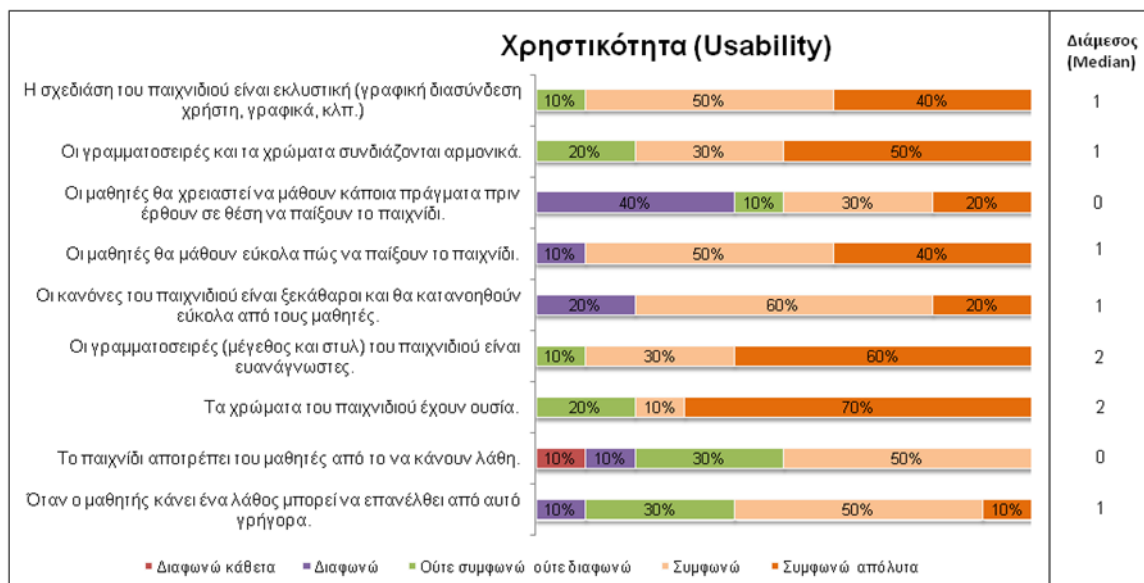
Εικόνα 79: Δημογραφικά δεδομένα αξιολόγησης

5.3.2 Χρηστικότητα

Στην Εικόνα 80 παρουσιάζονται τα θετικά στοιχεία όσον αφορά τη σχεδίαση του παιχνιδιού, την ευκολία εκμάθησης του και την υποστήριξη που αυτό προσφέρει στους χρήστες για να εντοπίσουν τα λάθη τους. Πιο αναλυτικά το 50% των συμμετεχόντων συμφωνεί και το 40% συμφωνεί απόλυτα ότι το παιχνίδι είναι ελκυστικό. Για την κατάλληλη επιλογή της γραμματοσειράς και των χρωμάτων το 30% συμφωνεί και το 50% συμφωνεί απόλυτα. Μάλιστα το 70% συμφωνεί απόλυτα ότι η επιλογή των χρωμάτων είναι σημαντική για το παιχνίδι.

Επιπλέον το 90% των συμμετεχόντων συμφωνεί ή συμφωνεί απόλυτα ότι οι μαθητές θα μάθουν εύκολα πώς να το χειρίζονται. Ωστόσο τα αποτελέσματα για το εάν οι μαθητές θα πρέπει να γνωρίζουν κάποια βασικά πράγματα πριν παίξουν το παιχνίδι είναι αμφιλεγόμενα, με το 50% των συμμετεχόντων να συμφωνεί ή να συμφωνεί απόλυτα, το 10% να παραμένει ουδέτερο και το 40% να διαφωνεί. Επομένως το σημείο αυτό επιδέχεται βελτιώσεις με την εισαγωγή περισσότερων παραδειγμάτων και επιπέδων θεωρίας για την ομαλότερη ένταξη των μαθητών στις έννοιες του προγραμματισμού.

Επιπροσθέτως το παιχνίδι θεωρείται ότι βοηθάει τον μαθητή να εντοπίσει τα πιθανά λάθη του και να τα διορθώσει σχετικά γρήγορα με το 50% να συμφωνεί και το 10% να συμφωνεί απόλυτα. Το παιχνίδι όμως δεν είναι βέβαιο εάν αποτρέπει τον μαθητή να κάνει λάθη κατά την σχεδίαση των αλγορίθμων του καθώς έχει διάμεσο 0. Βέβαια ο μαθητής κάνοντας λάθη μαθαίνει μέσα από αυτά και εφόσον κατανοήσει το λάθος του και το διορθώσει είναι πιθανότερο να μην το ξανακάνει σε σχέση με κάποιον άλλο μαθητή που απλά ακολούθησε την αυτόματη διόρθωση που θα του προσφέρει το σύστημα.



Εικόνα 80: Η σχεδίαση του παιχνιδιού

5.3.3 Εμπειρία παίκτη

Στην Εικόνα 81 παρουσιάζονται τα αποτελέσματα των ερωτήσεων που αφορούν την εμπειρία του χρήστη παίζοντας το παιχνίδι. Οι ερωτήσεις της Εικόνας 81 ομαδοποιούνται σε κατηγορίες ανάλογα με τον τύπο τους σύμφωνα με το μοντέλο MEEGA+. Στη συνέχεια ακολουθεί ο σχολιασμός των αποτελεσμάτων της εμπειρίας του παίκτη ξεκινώντας από την πρώτη κατηγορία των ερωτήσεων (όπως παρουσιάζονται στην Εικόνα 81) μέχρι και την τελευταία κατηγορία.

Αυτοπεποίθηση (Confidence). Στην πρώτη κατηγορία της Εικόνας 81 παρουσιάζονται τα αποτελέσματα που αφορούν στις απόψεις των εκπαιδευτικών σχετικά με την αυτοπεποίθηση που θα έχουν οι μαθητές παίζοντας το παιχνίδι. Όλοι οι συμμετέχοντες συμφωνούν ή συμφωνούν απόλυτα ότι με τη δομή και το εκπαιδευτικό υλικό του παιχνιδιού οι μαθητές θα αισθανθούν σίγουροι ότι θα μάθουν παίζοντας. Επίσης το μεγαλύτερο ποσοστό πιστεύει ότι το εκπαιδευτικό παιχνίδι θα είναι εύκολο για τους μαθητές (διάμεσος 1).

Πρόκληση (Challenge). Οι προκλήσεις που θα αντιμετωπίσει ο μαθητής παίζοντας είναι αρκετές διατηρώντας καταυτόν τον τρόπο αμείωτο το ενδιαφέρον του. Στη δεύτερη κατηγορία της Εικόνας 81 φαίνονται τα αποτελέσματα των ερωτήσεων που αφορούν στις προκλήσεις που προσφέρει το παιχνίδι. Όλες οι ερωτήσεις κατάφεραν να

συγκεντρώσουν διάμεσο 1 (συμφωνώ) και μάλιστα αξίζει να σημειωθεί ότι οι εκπαιδευτικοί δεν πιστεύουν ότι το παιχνίδι γίνεται μονότονο καθώς εξελίσσεται και δεν υπάρχουν στοιχεία που να το καθιστούν βαρετό. Έτσι οι μαθητές θα έχουν κίνητρο να συνεχίσουν να παίζουν ξεκλειδώνοντας νέες πίστες και μαθαίνοντας νέες έννοιες προγραμματισμού.

Ικανοποίηση (Satisfaction). Στην τρίτη κατηγορία της Εικόνας 81 παρουσιάζεται ο αναμενόμενος βαθμός ικανοποίησης που θα προσδώσει το παιχνίδι στους μαθητές σύμφωνα με τη γνώμη των εκπαιδευτικών. Το 90% των εκπαιδευτικών συμφωνεί ή συμφωνεί απόλυτα ότι οι μαθητές με τη δική τους προσωπική προσπάθεια θα καταφέρουν να ξεκλειδώσουν όλα τα επίπεδα του παιχνιδιού και θα ικανοποιηθούν με τα επιτεύγματα τους καθώς και με τις νέες έννοιες του προγραμματισμού που θα μάθουν. Τέλος, το 90% των εκπαιδευτικών που συμμετείχαν στην αξιολόγηση θα πρότειναν το συγκεκριμένο παιχνίδι σε συναδέλφους και μαθητές.

Ψυχαγωγία (Fun). Στην τέταρτη κατηγορία της Εικόνας 81 φαίνονται τα αποτελέσματα των ερωτήσεων που αφορούν στην ψυχαγωγία των μαθητών. Οι εκπαιδευτικοί συμφωνούν (διάμεσος 1) ότι οι μαθητές θα ψυχαγωγηθούν μέσω του φιλικού περιβάλλοντος του παιχνιδιού, του χαρούμενου τόνου της μουσικής και του ενδιαφέροντος σεναρίου.

Εστιασμένη Προσοχή (Focused Attention). Πέμπτη στη σειρά είναι η κατηγορία των ερωτήσεων που αφορούν την εστιασμένη προσοχή των μαθητών. Οι εκπαιδευτικοί συμφωνούν (διάμεσος 1) ότι η καλή σχεδίαση του παιχνιδιού θα καταφέρει να τραβήξει την προσοχή του μαθητή κατά την εκκίνηση του και να χάσει έτσι ο μαθητής την αίσθηση του χρόνου κατά τη διάρκεια του παιχνιδιού. Ωστόσο οι απόψεις δίστανται (διάμεσος 0) για το εάν το παιχνίδι θα καταφέρει να απορροφήσει τον παίκτη σε τέτοιο βαθμό ώστε να ξεχάσει το τριγύρω περιβάλλον του καθώς παίζει.

		Εμπειρία Παίκτη (Player Experience)				Διάμεσος (Median)	
Αυτοπρόβλεψη	Όταν πρωτοείδα το παιχνίδι, είχα την εντύπωση ότι θα είναι εύκολο για τους μαθητές.	10%	20%	60%	10%	1	
	Το περιεχόμενο και η δομή θα βοηθήσουν τους μαθητές να βεβαιωθούν ότι θα μάθουν από αυτό το παιχνίδι.	80%	20%			1	
Πρόκληση	Το παιχνίδι αποτελεί κατάλληλη πρόκληση για τους μαθητές.	10%	10%	30%	50%	1	
	Το παιχνίδι παρέχει νέες προκλήσεις (νέα εμπόδια, καταστάσεις ή παραλλαγές) σε κατάλληλο ρυθμό.	60%	40%			1	
	Το παιχνίδι δεν γίνεται μονότονο κατά την πορεία του (επανηλημμένες ή βαρετές δοκιμασίες).	10%	50%	40%		1	
Ικανοποίηση	Η ολοκλήρωση των δοκιμασιών του παιχνιδιού θα προκαλέσει ένα αίσθημα ικανοποίησης στους μαθητές.	10%	40%	50%		1	
	Η πρόοδος των μαθητών στο παιχνίδι θα οφείλεται στην προσωπική τους προσπάθεια.	10%	70%	20%		1	
	Οι μαθητές θα μείνουν ικανοποιημένοι με αυτά που θα μάθουν από το παιχνίδι.	10%	70%	20%		1	
	Θα συνιστούσα το παιχνίδι σε συναδέλφους και μαθητές.	10%	40%	50%		1	
Διασκέδαση	Οι μαθητές θα περάσουν ευχάριστα παίζοντας το παιχνίδι.	10%	50%	40%		1	
	Κατά τη διάρκεια του παιχνιδιού κάτι με έκανε να χαμογελάσω (στοιχεία του παιχνιδιού, δοκιμασία κλπ.)	30%	20%	50%		1	
Επίσημη Προσοχή	Υπάρχει κάτι ενδιαφέρον στην αρχή του παιχνιδιού που θα τραβήξει την προσοχή των μαθητών.	20%	60%	20%		1	
	Πιστεύω ότι οι μαθητές θα είναι τόσο αφοσιωμένοι με το να παίζουν, που θα χάσουν την αίσθηση του χρόνου.	10%	20%	40%	30%	1	
	Πιστεύω ότι οι μαθητές θα ξεχάσουν το άμεσο περιβάλλον τους καθώς θα παίζουν αυτό το παιχνίδι.	20%	40%	30%	10%	0	
Σχετικότητα	Το περιεχόμενο του παιχνιδιού είναι σχετικό με τα ενδιαφέροντα των μαθητών	10%	20%	40%	30%	1	
	Μου είναι ξεκάθαρο πως το περιεχόμενο του παιχνιδιού σχετίζεται με τον προγραμματισμό.	10%	40%	50%		1	
	Το παιχνίδι αποτελεί μια επαρκή μέθοδο εκμάθησης των βασικών εννοιών του προγραμματισμού.	20%	40%	40%		1	
	Προτιμώ να διδάσκω μέσω αυτού του παιχνιδιού σε σύγκριση με άλλους τρόπους διδασκαλίας (π.χ. παραδοσιακά βιβλία).	10%	20%	20%	50%	1	
Αναμενόμενα μαθησιακά αποτελέσματα	Το παιχνίδι μπορεί να βοηθήσει τον μαθητή να κατανοήσουν εύκολα τις βασικές έννοιες της "Υπολογιστικής Σκέψης".	60%	40%			1	
	Το εκπαιδευτικό υλικό του παιχνιδιού θα είναι σημαντικό για τους μαθητές μου.	10%	70%	20%		1	
	Οι μαθητές μου θα μάθουν κάτι καινούριο παίζοντας το παιχνίδι.	10%	50%	40%		1	
	Κατά την διάρκεια του παιχνιδιού οι μαθητές θα αφιερώσουν χρόνο για να διαβάσουν την θεωρία.	10%	40%	30%	20%	0	
	Η θεωρία που παρουσιάζεται σε κάθε επίπεδο θα κατανοηθεί εύκολα από τους μαθητές.	20%	20%	50%	10%	1	
	Οι μαθητές θα κατανοήσουν εύκολα την λειτουργικότητα των μπλοκ του παιχνιδιού.	10%	30%	40%	20%	1	
	Οι μαθητές θα μπορούν εύκολα να εντοπίζουν και να διορθώνουν λάθη μεταξύ των αλληλοσυνδεδεμένων μπλοκ.	10%	10%	30%	30%	20%	0
	Οι μαθητές θα διαβάσουν την περιοχή αποσφαλμάτωσης (συμβουλές, ανάλυση της περιοχής αποσφαλμάτωσης).	20%	40%	30%	10%	0	
	Οι συμβουλές που παρουσιάζονται στους μαθητές σε περίπτωση που δεν εκπληρώνουν έναν στόχο θα τους βοηθήσουν να τον ξεπεράσουν.	90%	10%			1	
	Οι συμβουλές και η ανάλυση του προγράμματος θα βοηθήσουν τους μαθητές να βελτιώσουν την ποιότητα του προγράμματος τους.	10%	70%	20%		1	

Εικόνα 81: Αποτελέσματα αξιολόγησης της εμπειρίας των χρηστών

Σχετικότητα (Relevance). Στην έκτη κατηγορία παρουσιάζονται τα αποτελέσματα της αξιολόγησης που αφορούν τη σχετικότητα του εκπαιδευτικού υλικού με τον τομέα της Πληροφορικής. Σύμφωνα με την Εικόνα 81 το περιεχόμενο του παιχνιδιού σχετίζεται με τον προγραμματισμό (40% συμφωνώ, 50% συμφωνώ απόλυτα) και με τα ενδιαφέροντα των μαθητών στα σχολεία (40% συμφωνώ, 30% συμφωνώ απόλυτα). Επιπλέον οι εκπαιδευτικοί υποστηρίζουν ότι το παιχνίδι αποτελεί μία επαρκή μέθοδο εκμάθησης των βασικών εννοιών του προγραμματισμού (80% συμφωνούν ή συμφωνούν απόλυτα) και ότι θα προτιμούσαν να χρησιμοποιήσουν το παιχνίδι έναντι ενός βιβλίου για την εκπαίδευση των μαθητών (διάμεσος 1).

Αναμενόμενα μαθησιακά αποτελέσματα (Perceived Learning). Στην τελευταία κατηγορία της Εικόνας 81 παρουσιάζονται οι ερωτήσεις που αφορούν στα αναμενόμενα μαθησιακά αποτελέσματα. Όλοι οι εκπαιδευτικοί που συμμετείχαν στην αξιολόγηση του παιχνιδιού συμφωνούν ή συμφωνούν απόλυτα (60% συμφωνούν, 40% συμφωνούν απόλυτα) ότι το παιχνίδι θα βοηθήσει τους μαθητές να κατανοήσουν εύκολα τις βασικές έννοιες της "Υπολογιστικής Σκέψης". Επιπλέον το 90% θεωρεί ότι οι μαθητές θα μάθουν νέες έννοιες προγραμματισμού, οι οποίες θα είναι αρκετά χρήσιμες για να αναπτύξουν την "Υπολογιστική τους Σκέψη". Ωστόσο υπάρχει αμφιβολία (διάμεσος 0) για το εάν οι μαθητές θα αφιερώσουν τον απαραίτητο χρόνο για να διαβάσουν τη θεωρία και τη περιοχή της αποσφαλμάτωσης και να εντοπίσουν εύκολα τα λάθη μεταξύ των αλληλοσυνδεδεμένων μπλοκ. Τέλος, θετικά είναι τα αποτελέσματα που αφορούν τη βοήθεια και τις συμβουλές που περιέχονται στην περιοχή της αποσφαλμάτωσης. Συγκεκριμένα, πάνω από το 90% των συμμετεχόντων συμφωνούν ή συμφωνούν απόλυτα ότι οι συμβουλές που προσφέρει η περιοχή της αποσφαλμάτωσης θα βοηθήσει τους μαθητές να κατανοήσουν τα λάθη τους, να επανέλθουν από αυτά και να βελτιώσουν την ποιότητα του αλγορίθμου τους.

5.3.4 Ερωτήσεις ανοιχτού τύπου

Στην ενότητα αυτή παρουσιάζονται οι απαντήσεις των συμμετεχόντων στις δύο ερωτήσεις ανοιχτού τύπου που αφορούν ποια στοιχεία τους άρεσαν στο παιχνίδι και ποια όχι. Συνολικά απάντησαν οι επτά από τους δέκα εκπαιδευτικούς και οι εντυπώσεις τους ήταν θετικές. Στην ερώτηση που αφορά τα στοιχεία που δεν τους άρεσαν στο παιχνίδι

δύο είναι κατά τη γνώμη του συγγραφέα τα πιο σημαντικά από τα οποία το ένα επιλύθηκε μετά την αξιολόγηση του παιχνιδιού. Το πρώτο στοιχείο που επιλύθηκε αφορά την απουσία της μετάφρασης του παιχνιδιού από τα αγγλικά στα ελληνικά. Τα παιδιά του Δημοτικού θα είναι δύσκολο να κατανοήσουν τη θεωρία και τις υπόλοιπες πληροφορίες του παιχνιδιού στα αγγλικά. Το παιχνίδι στην αναθεωρημένη έκδοση υποστηρίζει τόσο τα Αγγλικά όσο και τα Ελληνικά.

Το δεύτερο στοιχείο που εντόπισε μια εκπαιδευτικός ήταν ότι ο χαρακτήρας του παιχνιδιού δεν ακολουθούσε τις εντολές της καθώς πρόσθετε στον αλγόριθμο της ένα μπλοκ ανά προσπάθεια. Έπειτα από πολλές δοκιμές κατά την ανάπτυξη του παιχνιδιού και μετά την αξιολόγηση του δεν εντοπίστηκε το συγκεκριμένο πρόβλημα από τον συγγραφέα. Η μόνη περίπτωση που το παιχνίδι αντιδρούσε όπως περιγράφει η εκπαιδευτικός είναι όταν ο χρήστης πατήσει την εκτέλεση του αλγορίθμου και κατά την εκτέλεση του αλλάζει το παράθυρο του παιχνιδιού με ένα δεύτερο. Αυτό που θα συμβεί είναι το παράθυρο του παιχνιδιού να χάσει την εστίαση του και η περιοχή του παιχνιδιού, δηλαδή το Phaser, να βρίσκεται σε κατάσταση αδράνειας. Το ίδιο όμως δεν συμβαίνει και με τον διερμηνέα ο οποίος επικοινωνεί με το Phaser. Ο διερμηνέας συνεχίζει την εκτέλεση του αλγορίθμου του παίκτη και μόλις ο χρήστης εστιάζει στο παράθυρο του παιχνιδιού ο χαρακτήρας συνεχίζει να ακολουθεί τις εντολές από το σημείο που βρίσκεται ο διερμηνέας. Έπειτα από αρκετή αναζήτηση το συμπέρασμα στο συγκεκριμένο πρόβλημα είναι από την μία πλευρά ότι το Phaser με την αλλαγή της εστίασης του παραθύρου δεν μπορεί να συνεχίσει την εκτέλεση του αλλά οδηγείται αναγκαστικά σε κατάσταση αδράνειας. Από την άλλη πλευρά δεν βρέθηκαν τα κατάλληλα τεκμήρια που θα βοηθούσαν στην παύση της εκτέλεσης του αλγορίθμου από τον διερμηνέα.

Τέλος στην αναθεωρημένη έκδοση επανασχεδιάστηκε το τελευταίο επίπεδο (δωδέκατο), λαμβάνοντας υπ' όψιν την τέταρτη απάντηση του πίνακα 37, έτσι ώστε να ομορφύνει διαισθητικά και να εκπληρωθούν όλοι οι στόχοι του σεναρίου του παιχνιδιού.

Στους Πίνακες 36 και 37 παρουσιάζονται οι απαντήσεις των συμμετεχόντων στις ερωτήσεις ανοιχτού τύπου.

Πίνακας 36: Ποια είναι η γνώμη σας για το παιχνίδι; Τι σας άρεσε σε αυτό;

Ποια είναι η γνώμη σας για το παιχνίδι; Τι σας άρεσε σε αυτό;	
A/A	Απάντηση
1	Music and increasing complexity.
2	Has a similar concept and looks and feels like "scratch" type of blocks and close to the "hour of code" and students feel comfortable with this type of programming/playing.
3	Excellent game. I believe it will help students understand the basic concept of structured programming (Sequencing, Selection & Iteration). They will also have fun.
4	Graphics and Action.
5	The whole idea. Very entertaining and educational use of block programming method.
6	Είναι πολύ ωραίο το περιβάλλον και ο σχεδιασμός του παιχνιδιού.
7	Playing and learning is always fun! A nice way to teach students basic programming structures.

Πίνακας 37: Ποια είναι η γνώμη σας για το παιχνίδι; Τι δεν σας άρεσε σε αυτό;

Ποια είναι η γνώμη σας για το παιχνίδι; Τι δεν σας άρεσε σε αυτό;	
A/A	Απάντηση
1	Gameplay needs improvement.
2	I liked the game, is fun and educational.
3	Nothing.
4	Last level could be more impressive.
5	- Η περιγραφή σε κάθε επίπεδο σίγουρα θα πρέπει να γίνει στα ελληνικά αν δοκιμαστεί από μικρούς μαθητές. Ίσως και οι εντολές. - Στο επίπεδο 3 μπαίνουν ταυτόχρονα δύο νέες εντολές (repeat και laser) και ίσως δυσκολέψουν τους μαθητές (αυτό έχει να κάνει και με τη διάρκεια του μαθήματος, οι ασκήσεις στοχεύουν σε μία σειρά μαθημάτων ή σε μεμονωμένες διαλέξεις όπως η δράση Hour of Code?)
6	- I would like to have hints at every step the user needs one instead of resetting

the game and losing points.

- I also noticed that when I used step to step programming and added new blocks after each reset I had a different reaction of the hero. For example with the same code I passed the obstacle but when resetting and adding a new block at the end I fell on the obstacle

- After collecting the stars from a route and you want to go the other way now that no stars are there any more it's not clear to the student how many moves he has to do because the existence of the stars helps you count but when you have collected them there should be something left there to point that a star used to be there.

5.4 Συμπεράσματα αξιολόγησης

Τα αποτελέσματα της πιλοτικής αξιολόγησης από τους δέκα εκπαιδευτικούς Πληροφορικής της Πρωτοβάθμιας και Δευτεροβάθμιας εκπαίδευσης ήταν θετικά. Αναλυτικότερα οι ερωτήσεις που αφορούν την Αυτοπεποίθηση, την Πρόκληση, την Ικανοποίηση, την Ψυχαγωγία και τη Συνάφεια του εκπαιδευτικού παιχνιδιού κέρδισαν τις εντυπώσεις των εκπαιδευτικών συγκεντρώνοντας διάμεσο 1 (συμφωνώ). Επιπλέον τα αποτελέσματα των ερωτήσεων που αφορούν τη Χρησιμότητα του παιχνιδιού δείχνουν ότι η σχεδίαση του παιχνιδιού είναι κατάλληλα διαμορφωμένη για να τραβήξει την προσοχή των μαθητών και να εμβαθύνει την προσοχή τους στις έννοιες του προγραμματισμού και στην "Υπολογιστική Σκέψη".

Ωστόσο υπάρχουν και ορισμένα σημεία τα οποία δημιούργησαν κάποιες αμφιβολίες μεταξύ των εκπαιδευτικών και επιδέχονται βελτιώσεις. Για παράδειγμα οι απόψεις δίστανται για το εάν οι μαθητές θα διαβάσουν την περιοχή της θεωρίας και της αποσφαλμάτωσης οι οποίες είναι αρκετά σημαντικές για την πρόοδο των παικτών στο παιχνίδι. Επιπλέον μέσω των ερωτήσεων ανοιχτού τύπου προτάθηκαν κάποιες ιδέες για τη βελτίωση του παιχνιδιού, μια εκ των οποίων υλοποιήθηκε σε μετέπειτα στάδιο της συγγραφής της διπλωματικής και αφορά τη μετάφραση του παιχνιδιού.

Τέλος, ένα από τα πιο σημαντικά αποτελέσματα της αξιολόγησης του παιχνιδιού αποτελεί η γνώμη των εκπαιδευτικών πως το παιχνίδι πετυχαίνει τους εκπαιδευτικούς στόχους για τους οποίους αναπτύχθηκε με το 60% να συμφωνεί και το 40% να συμφωνεί

απόλυτα. Αυτός είναι και ο βασικός στόχος της ανάπτυξης του παιχνιδιού, δηλαδή να βοηθήσει στην ανάπτυξη της "Υπολογιστικής Σκέψης" στους μαθητές.

6 Επίλογος

6.1 Σύνοψη και συμπεράσματα

Τα παιχνίδια σοβαρού σκοπού αποτελούν έναν εναλλακτικό τρόπο στην εκπαίδευση των ανθρώπων και κυρίως των μικρών μαθητών. Κάθε εκπαιδευτικό παιχνίδι έχει ως κύριο στόχο να παρουσιάσει στον χρήστη κάποιες έννοιες που αφορούν έναν συγκεκριμένο επιστημονικό τομέα με τρόπο λιτό και ψυχαγωγικό. Ακόμη η έννοια του "παιχνιδιού" είναι γνώριμη στους μαθητές μικρής ηλικίας και ο συνδυασμός της με την εκπαίδευση κεντρίζει περισσότερο το ενδιαφέρον τους σε σύγκριση με ένα παραδοσιακό βιβλίο. Επιπλέον τα εκπαιδευτικά παιχνίδια ενισχύουν την αυτοδιδασκαλία των χρηστών παρέχοντας τους όλα τα απαραίτητα εφόδια για να μάθουν τον τρόπο χειρισμού του και τη θεωρία για ένα συγκεκριμένο πεδίο.

Αυτοί αποτελούν μερικούς από τους βασικούς λόγους που τα εκπαιδευτικά παιχνίδια αποκτούν ολοένα και μεγαλύτερη αποδοχή στον τομέα της εκπαίδευσης και αποτέλεσε κινητήριο μοχλό για την ανάπτυξη του σοβαρού παιχνιδιού BlocklyScript. Ο βασικός στόχος της παρούσας διπλωματικής εργασίας ήταν η ανάπτυξη ενός παιχνιδιού σοβαρού σκοπού για την ανάπτυξη της "Υπολογιστική Σκέψης" σε μικρούς μαθητές με την ενσωμάτωση μιας προσαρμοσμένης γλώσσας οπτικού προγραμματισμού. Το BlocklyScript συνδυάζει τα στοιχεία ενός RPG (Role-Playing Game) με μια φιλική διεπαφή χρήστη και ένα αρκετά ενδιαφέρον σενάριο. Κατά την σχεδίαση του παιχνιδιού δόθηκε ιδιαίτερη βαρύτητα στην επιλογή των χρωμάτων και την παρουσίαση της θεωρίας παρέχοντας εικόνες και παραδείγματα έτσι ώστε να ενισχυθεί η αυτοπεποίθηση των μαθητών και να χάσουν την αίσθηση του χρόνου καθώς παίζουν.

Επιπλέον η ενσωμάτωση του οπτικού προγραμματισμού με τη βοήθεια της Blockly στο παιχνίδι διευκολύνει τους χρήστες να μάθουν τον χειρισμό του παιχνιδιού και να κατανοήσουν ευκολότερα τις έννοιες που παρουσιάζονται ανά επίπεδο. Χρησιμοποιώντας τα μπλοκ οι χρήστες μπορούν να εφαρμόσουν απευθείας τη θεωρία που διάβασαν συλλέγοντας αστέρια, νικώντας εξωγήινους και αποφεύγοντας παγίδες. Έτσι σε κάθε επίπεδο του παιχνιδιού οι χρήστες αντιμετωπίζουν νέες προκλήσεις και διατηρούν αμείωτο το ενδιαφέρον τους.

Τέλος, με βάση τα αποτελέσματα της πιλοτικής αξιολόγησης το BlocklyScript αποτελεί ένα αξιόπιστο εργαλείο για την ανάπτυξη της "Υπολογιστικής Σκέψης" σε μικρούς

μαθητές. Βέβαια για να είναι ασφαλή τα αποτελέσματα της έρευνας το παιχνίδι θα πρέπει να αξιολογηθεί από τους ίδιους τους μαθητές και επίσης να διερευνηθεί η εκπαιδευτική του αξία. Συμπερασματικά το BlocklyScript αποτελεί μια καλή προσπάθεια ανάπτυξης ενός εκπαιδευτικού παιχνιδιού που θα βοηθήσει τους μαθητές να κατανοήσουν μόνοι τους τις βασικές έννοιες του προγραμματισμού χωρίς να είναι αναγκασμένοι να το τερματίσουν με μια μόνο προσπάθεια ή να τους περιορίζει ο χρόνος.

6.2 Όρια και περιορισμοί έρευνας

Ένα από τα σημαντικότερα προβλήματα που αντιμετωπίστηκαν κατά την ανάπτυξη του παιχνιδιού είναι η έλλειψη οδηγιών για την αξιοποίηση της βιβλιοθήκης Blockly. Στην βιβλιογραφική επισκόπηση παρουσιάστηκαν τρία παιχνίδια που χρησιμοποιούν την Blockly, ωστόσο δεν υπάρχουν οδηγοί που να εξηγούν τον τρόπο σχεδιασμού και ανάπτυξής τους. Έτσι κατά την σχεδίαση του παιχνιδιού αντιμετωπίστηκαν αρκετές προκλήσεις που αφορούσαν τον τρόπο σύνδεσης του παιχνιδιού με το πλαίσιο ανάπτυξης 2D παιχνιδιών Phaser.

Επιπλέον τα παραδείγματα που προσφέρει η Blockly στην επίσημη ιστοσελίδα της βοηθούν αρκετά τον προγραμματιστή να κατανοήσει τη θεωρία, ωστόσο είναι πολύ λίγα και εξηγούν τις βασικότερες έννοιες. Ιδιαίτερος τα παραδείγματα με τον διερμηνέα της JavaScript, πάνω στα οποία βασίστηκε η σχεδίαση του παιχνιδιού, δεν αξιοποιούν όλες τις δυνατότητες που παρέχει ο διερμηνέας όπως για παράδειγμα η παύση της εκτέλεσης του αλγορίθμου.

Όσον αφορά την αξιολόγηση της εκπαιδευτικής αξίας του παιχνιδιού, όπως ήδη αναφέρθηκε, κρίνεται απαραίτητη η αξιοποίησή του στην τάξη για την εξαγωγή ασφαλών συμπερασμάτων.

6.3 Μελλοντικές επεκτάσεις

Μια από τις βασικότερες ιδέες που μπορούν να υλοποιηθούν μελλοντικά και να ενισχύσουν την επίβλεψη των δασκάλων πάνω στις επιδόσεις των μαθητών τους είναι η ενσωμάτωση διαδικτυακών τάξεων στο παιχνίδι. Για την υλοποίηση των τάξεων θα πρέπει να διαμορφωθεί κατάλληλα η βάση δεδομένων ώστε να επιτρέπει τον/την δάσκαλο/α να πληκτρολογεί το όνομα της τάξης και ένα συνθηματικό και να έχει πρόσβαση στην πρόοδο των μαθητών. Επιπλέον ο/η εκπαιδευτικός θα έχει τη δυνατότητα να προσθέτει λογαριασμούς στην τάξη και έπειτα να τους μοιράζει στους μαθητές του/της.

Επιπλέον στο παιχνίδι θα μπορούσε ο διερμηνέας να εκτελεί τον αλγόριθμο κατά βήμα ή σε διαφορετικές ταχύτητες έτσι ώστε ο χρήστης να ελέγχει ευκολότερα τις λύσεις του. Με την εκτέλεση ανά μπλοκ ο χρήστης θα μπορεί να διαπιστώσει ακριβώς τη λειτουργικότητα κάθε μπλοκ και να καταλάβει ακόμη καλύτερα τη θεωρία. Για να γίνει αυτό ωστόσο τα πρέπει να επανεξεταστούν τα μπλοκ εντολών ώστε το παιχνίδι να είναι λειτουργικό. Για παράδειγμα, εάν ο χρήστης στην εκτέλεση κατά βήμα πατούσε στο μπλοκ "jump" και μετά ακολουθούσε ένα μπλοκ "move left" τότε ο παίκτης θα έκανε ένα οριζόντιο άλμα, θα έπεφτε λόγω της βαρύτητας και μετά θα προχωρούσε αριστερά χτυπώντας πιθανός πάνω σε ένα εμπόδιο. Μια λύση στο συγκεκριμένο πρόβλημα θα ήταν να ενωθούν τα δύο αυτά μπλοκ σε ένα το οποίο να αναλαμβάνει μια πιο σύνθετη κίνηση.

Τέλος, θα μπορούσε να αφιερωθεί περισσότερος χρόνος για τον σχεδιασμό γραφικών, την επιλογή της μουσικής και την προσθήκη νέων επιπέδων. Στα νέα επίπεδα θα μπορούσε να ενσωματωθεί νέα θεωρία ή να σχεδιαστούν δυσκολότερα επίπεδα πάνω στην ίδια θεωρία. Στην πρώτη περίπτωση ο χρήστης θα μάθαινε νέες προγραμματιστικές έννοιες που θα ενίσχυαν την "Υπολογιστική του Σκέψη", ενώ στη δεύτερη περίπτωση θα αντιμετώπιζε νέες προκλήσεις και θα κατανοούσε καλύτερα τη θεωρία που έχει ήδη μάθει.

Βιβλιογραφία

De Gloria, A., Bellotti, F., & Berta, R. (2014). Serious Games for education and training. *International Journal of Serious Games*, 1(1). DOI: <https://doi.org/10.17083/ijsg.v1i1.11>

Huizinga, J. (1949). *Homo Ludens: A study of the play-element in culture*. London: Boston and Henley

Abt, Clark C. (1970). *Serious games*, New York: Viking Press

Laamarti, F., Eid M., & El Saddik, A. (2014) "An Overview of Serious Games," *International Journal of Computer Games Technology*, vol. 2014, 15 pages,. DOI: <https://doi.org/10.1155/2014/358152>

Harimurti, R., Ekohariadi, Munoto, & Asto, B. (2019). The concept of computational thinking toward information and communication technology learning. IOP Conference Series: Materials Science and Engineering. 535. 012004. DOI: <http://dx.doi.org/10.1088/1757-899X/535/1/012004>

Riley D.,D., & Hunt, K., A. (2014). *Computational Thinking for the modern problem solver*, London: CRC Press, Taylor & Francis Group

Kraus, J., & Prottzman, K. (2016). *Computational Thinking and Coding for every student: The teacher's getting-started guide*, California: Corwin

Saito, D., Washizaki, H., Fukazawa, Y. (2017). "Comparison of Text-Based and Visual-Based Programming Input Methods for First-Time Learners", *Journal of Information Technology Education: Research*, v16 p209-226 2017. DOI: <https://doi.org/10.28945/3775>

Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B. & Kafai, Y. (2009). *Scratch: Programming for All*. *Commun. ACM*. 52. 60-67. DOI: <https://doi.org/10.1145/1592761.1592779>

Giannakoulas A., Xinogalos S. (2019) Current Trends in On-line Games for Teaching Programming Concepts to Primary School Students. In: Tsitouridou M., A. Diniz J., Mikropoulos T. (eds) *Technology and Innovation in Learning, Teaching and Education. TECH-EDU 2018. Communications in Computer and Information Science*, vol 993. Springer, Cham. DOI: https://doi.org/10.1007/978-3-030-20954-4_5

Ibrahim, R., & Jaafar, A. (2009). Educational games (EG) design framework: Combination of game design, pedagogy and content modeling. In *International Conference on Electrical Engineering and Informatics* (pp. 293-298). Selangor, Malaysia. doi:10.1109/ICEEI.2009.5254771

Malliarakis, Christos & Satratzemi, Maya & Xinogalos, Stelios. (2014). Integrating Learning Analytics in an Educational MMORPG for Computer Programming. *Proceedings - IEEE 14th International Conference on Advanced Learning Technologies*, DOI: <https://doi.org/10.1109/ICALT.2014.74>.

Malliarakis, C., Satratzemi, M., & Xinogalos, S. (2014). Designing educational games for computer programming: A holistic framework. *Electronic Journal of e-Learning*, 12, 281-298.

Wing, J. (2006). Computational Thinking. *Communications of the ACM*. 49. 33-35. DOI: <https://doi.org/10.1145/1118178.1118215>.

Petri, G. & Gresse von Wangenheim, C. & Borgatto, A. (2018). MEEGA+: A Method for the Evaluation of Educational Games for Computing Education (Technical report, INCoD/GQS.05.2018.E). Brazilian Institute for Digital Convergence.

Pinelle, D. & Wong, N. (2008). Heuristic evaluation for games: Usability principles for video game design. *Conference on Human Factors in Computing Systems - Proceedings*. DOI: <https://doi.org/10.1145/1357054.1357282>.

Dr. Axel Rauschmayer, 2014 "Speaking JavaScript: An In-Dept Guide for Programmers", USA: O' Reilly.

Διαδικτυακές πηγές

CodeCombat. (χ.η.). Ανακτήθηκε 26 Δεκεμβρίου 2019, από <https://codecombat.com/>

Minecraft Hour of Code. (χ.η.). Ανακτήθηκε 26 Δεκεμβρίου 2019, από <https://code.org/minecraft>

Rapid Router. (χ.η.). Ανακτήθηκε 26 Δεκεμβρίου 2019, από <https://www.codeforlife.education/rapidrouter/>

Run Marco!. (χ.η.). Ανακτήθηκε 26 Δεκεμβρίου 2019, από <https://www.allcancode.com/hourofcode>

Blockly. (χ.η.). Ανακτήθηκε 20 Δεκεμβρίου 2019, από <https://developers.google.com/blockly>

Scratch. (χ.η.). Ανακτήθηκε 18 Δεκεμβρίου 2019, από <https://scratch.mit.edu/>

Kodu. (χ.η.). Ανακτήθηκε 15 Δεκεμβρίου 2019, από <https://www.kodugamelab.com/>

Blockly (χ.η.). Ανακτήθηκε 10 Δεκεμβρίου 2019, από <https://developers.google.com/blockly>

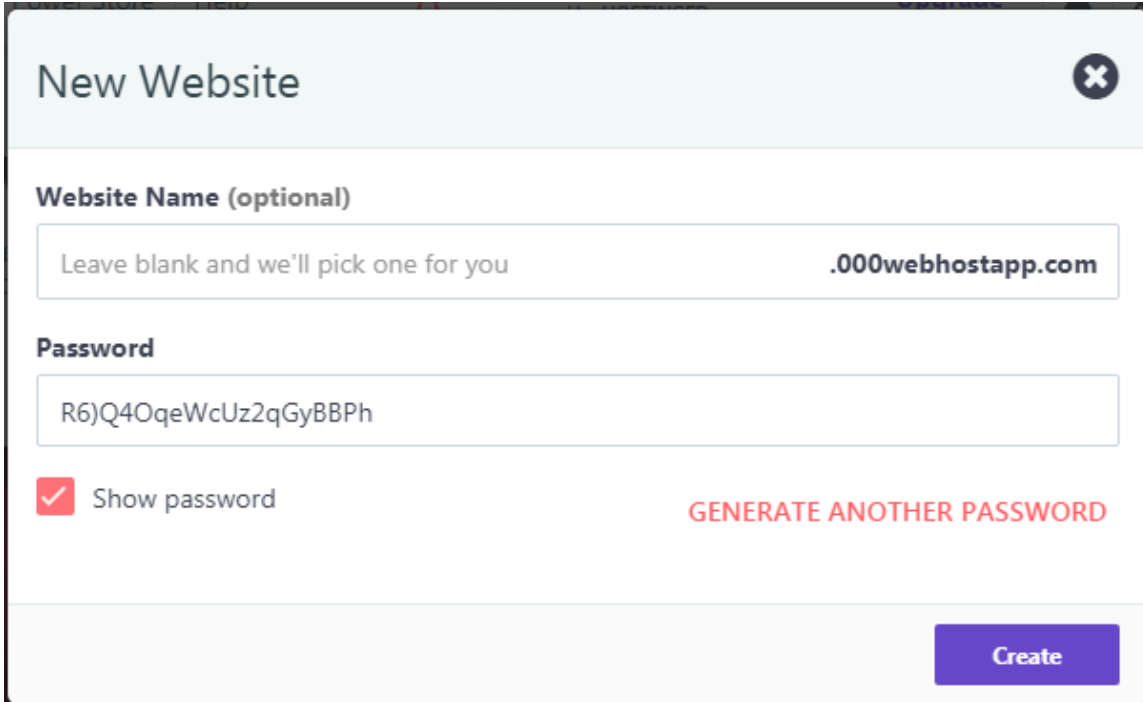
Phaser (χ.η.). Ανακτήθηκε 10 Δεκεμβρίου 2019, από <https://phaser.io/>

Παράρτημα Α

6.4 Δημοσίευση ιστοσελίδας στο Διαδίκτυο

Η δημοσίευση της ιστοσελίδας στο Διαδίκτυο έγινε με τη βοήθεια του παροχέα φιλοξενίας ιστοσελίδων 000webhost και μπορεί κανείς να τον επισκευτεί στον εξής σύνδεσμο: <https://gr.000webhost.com/>

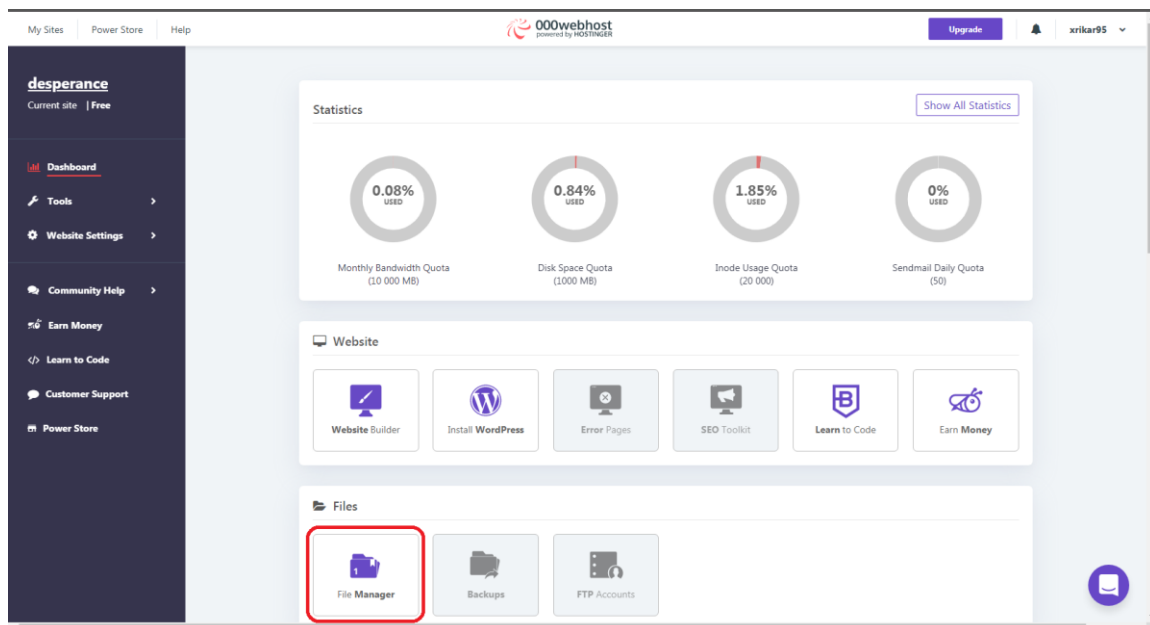
Η εγγραφή στον παροχέα 000webhost και η δημοσίευσης της ιστοσελίδας γίνεται δωρεάν και δεν περιλαμβάνονται διαφημίσεις. Εφόσον γίνει η εγγραφή στο σύστημα ο χρήστης μπορεί να εισέλθει και να συνεχίσει με τη δημοσίευση της ιστοσελίδας του. Για τη δημιουργία μιας νέας ιστοσελίδας ο χρήστης θα πρέπει να πατήσει το κουμπί "Create New Site" και να δώσει ένα όνομα και ένα συνθηματικό για την ιστοσελίδα όπως φαίνεται στην Εικόνα 82.



The image shows a web form titled "New Website" with a close button in the top right corner. The form is divided into several sections. The first section is "Website Name (optional)", which contains a text input field with the placeholder text "Leave blank and we'll pick one for you" and a dropdown menu showing ".000webhostapp.com". The second section is "Password", which contains a text input field with the password "R6)Q4OqeWcUz2qGyBBPh". Below the password field, there is a checkbox labeled "Show password" which is checked, and a red link that says "GENERATE ANOTHER PASSWORD". At the bottom right of the form, there is a blue button labeled "Create".

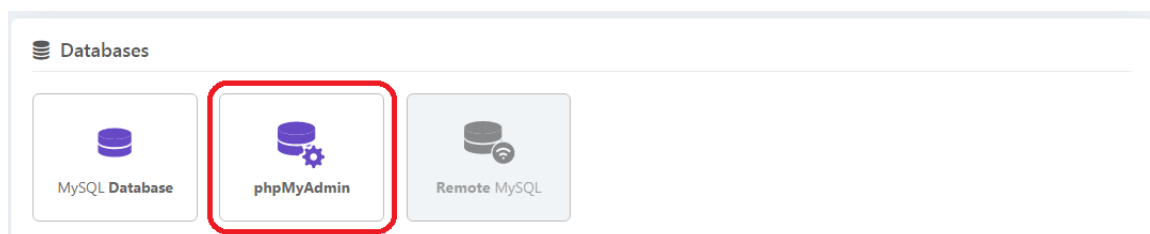
Εικόνα 82: Δημιουργία νέας ιστοσελίδας

Με τη δημιουργία της ιστοσελίδας ο χρήστης μπορεί να δει τις διαθέσιμες ιστοσελίδες του και κάνοντας αριστερό κλικ πάνω από το κουμπί "Manage Website" μπορεί να διαχειριστεί την αντίστοιχη ιστοσελίδα. Στην σελίδα διαχείρισης της ιστοσελίδας ο χρήστης μπορεί να "ανεβάσει" τα αρχεία του πατώντας το κουμπί "File Manager" όπως φαίνεται στη Εικόνα 83.



Εικόνα 83: Διαχειριστής αρχείων ιστοσελίδας

Εφόσον φορτώσει ο διαχειριστής αρχείων ο χρήστης μπορεί απλά να σύρει και να αφήσει τα αρχεία του στο παράθυρο του διαχειριστή. Επιπλέον ο χρήστης έχει τη δυνατότητα να δημιουργήσει τη δική του βάση δεδομένων από το παράθυρο διαχείρισης της ιστοσελίδας πατώντας το κουμπί "phpMyAdmin" όπως φαίνεται στην Εικόνα 84.



Εικόνα 84: Κουμπί δημιουργίας βάσης δεδομένων

Το "phpMyAdmin" είναι ένα δημοφιλές εργαλείο για τη διαχείριση βάσεων δεδομένων και υπάρχουν πολλοί οδηγοί στο διαδίκτυο για την εκμάθησή του. Με τη δημιουργία της βάσης δεδομένων ο χρήστης μπορεί μέσω PHP κώδικα στα αρχεία που "ανέβασε" να συνδεθεί με αυτήν δίνοντας τα απαραίτητα στοιχεία (όνομα βάσης, όνομα διαχειριστή και κωδικό).

Τέλος για να επισκεφτεί ο χρήστης την ιστοσελίδα του αρκεί να πληκτρολογήσει ως διεύθυνση το όνομα που έδωσε στην ιστοσελίδα ακολουθούμενο από το αλφαριθμητικό ".000webhostapp.com".