



ΔΗΜΟΚΡΙΤΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΡΑΚΗΣ

Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

Τομέας Λογισμικού και Ανάπτυξης Εφαρμογών

## Διπλωματική Εργασία

---

# Τριδιάστατη Οπτικοποίηση Δεδομένων Αισθητήρων

Σαλέπης Πέτρος

AEM: 56582

Επιβλέπουσα Καθηγήτρια:

κ. Κατσίρη Ελευθερία

Ξάνθη 2019



## ΠΕΡΙΛΗΨΗ

---

Η σύγχρονη μεταφορά δεδομένων και η επικοινωνία των μέσων γίνεται πια αποκλειστικά μέσω του διαδικτύου. Εκεί, διαμοιράζονται δεδομένα με σκοπό την απομακρυσμένη επικοινωνία και την επίτευξη διαδραστικότητας με τον χρήστη. Ένας κύριος τρόπος επικοινωνίας μέσω του διαδικτύου αποτελεί η επικοινωνία με τη χρήση του πρωτοκόλλου μεταφοράς υπερκειμένου (HTTP) το οποίο υποστηρίζεται από πλήθος περιηγητών διαδικτύου αλλά και που πλαισιώνεται από ποικίλες γλώσσες και τεχνικές προγραμματισμού.

Για τη μελέτη της επεξεργασίας δεδομένων πριν και μετά τη μετάδοση με HTTP επικοινωνία, υλοποιήθηκε το κλασικό παιχνίδι Simon Says σε περιβάλλον Second Life. Μέσω αυτού, δημιουργήθηκε ένα προϊόν το οποίο είναι διαθέσιμο στη διαδικτυακή αγορά του Second Life και με το οποίο οι χρήστες μπορούν να παίξουν στο ίδιο το Second Life. Παρόλα αυτά, προσφέρεται η επιπλέον δυνατότητα απομακρυσμένου παιχνιδιού στο οποίο μια οποιαδήποτε συσκευή με έναν διαδικτυακό περιηγητή, μπορεί να επικοινωνήσει με το Second Life και να συνεχίσει το ίδιο παιχνίδι. Το παιχνίδι στη νέα συσκευή δεν απαιτεί εγκατάσταση εφαρμογών ή κάτι παραπάνω από σύνδεση στο διαδίκτυο καθώς χρησιμοποιεί οποιονδήποτε διαδικτυακό περιηγητή (Web Browser). Τέλος, εξετάστηκε και η δυνατότητα επικοινωνίας του παιχνιδιού με συσκευή σειριακών δεδομένων, οπότε υλοποιήθηκε ειδικός περιηγητής ικανός να επικοινωνήσει με μια πλακέτα μικροελεγκτή Arduino.

Το repository της εργασίας: <https://github.com/petros3432/SLSimonSays> .

Η σελίδα στο Second Life Marketplace του προϊόντος: <https://marketplace.secondlife.com/p/Simon-Saysv3/18036281> .

## ABSTRACT

---

Modern data transfer and media communication is now done exclusively through the internet. There, data is shared for the purpose of remote communication and user interaction. One of the main means of communication over the internet is communication using the Hypertext Transfer Protocol (HTTP) which is supported by a multitude of web browsers and is also supported by a variety of programming languages and techniques.

To study data processing before and after HTTP communication, the classic Simon Says game was implemented in Second Life. Through this, a product was created that is available on Second Life's online marketplace, with which users can play in Second Life itself. However, there is also the option of remote play where any device with a web browser can communicate with Second Life and continue the same game. Playing on the new device does not require installation of applications or anything more than an internet connection, as it can use any web browser. Finally, the game's ability to communicate with a serial data device was examined, so a dedicated browser capable of communicating with an Arduino microcontroller board was implemented.

Repository of the Product: <https://github.com/petros3432/SLSimonSays>.

The page in the Second Life Marketplace of the product: <https://marketplace.secondlife.com/p/Simon-Saysv3/18036281>.

## ΠΕΡΙΕΧΟΜΕΝΑ

---

ΠΕΡΙΛΗΨΗ .....	3
ABSTRACT .....	4
ΠΕΡΙΕΧΟΜΕΝΑ.....	5
ΛΙΣΤΑ ΕΙΚΟΝΩΝ .....	9
ΚΕΦΑΛΑΙΟ 1: <b>Αντικείμενο και Δομή</b> .....	13
1.1 ΑΝΤΙΚΕΙΜΕΝΟ ΕΡΓΑΣΙΑΣ .....	13
1.2 ΣΚΟΠΟΣ ΤΗΣ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ .....	13
1.3 ΔΟΜΗ ΤΗΣ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ.....	14
ΚΕΦΑΛΑΙΟ 2: <b>Second Life και Εργαλεία Server</b> .....	15
2.1 ΕΙΣΑΓΩΓΗ.....	15
2.2 SECOND LIFE .....	15
2.2.1 ΙΣΤΟΡΙΑ .....	16
2.2.2 ΟΙΚΟΝΟΜΙΑ.....	18
2.2.3 LSL (Linden Scripting Language) .....	19
2.2.4 SECOND LIFE WORLD.....	21
2.3 ΙΔΙΟΤΗΤΕΣ ΑΝΤΙΚΕΙΜΕΝΩΝ SECOND LIFE .....	22
2.3.1 ΑΛΛΑΓΗ ΧΡΩΜΑΤΟΣ .....	22
2.3.2 ΥΦΕΣ ΑΝΤΙΚΕΙΜΕΝΩΝ (TEXTURES).....	23
2.3.3 ΕΠΙΠΛΕΟΝ ΚΕΙΜΕΝΟ (FLOATING TEXT).....	24
2.4 ΔΙΑΔΙΚΤΥΑΚΟΣ ΕΞΥΠΗΡΕΤΗΤΗΣ (WEB SERVER).....	24
2.4.1 HTML ΣΕΛΙΔΑ .....	25
2.4.1.1 ΓΛΩΣΣΑ HTML .....	25
2.4.1.2 CSS ΓΡΑΦΙΚΑ .....	26
2.4.1.3 ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΜΕ JAVASCRIPT.....	27

2.4.1.4 AJAX .....	27
2.4.2 ΡΗΡ ΣΕΛΙΔΑ.....	28
2.4.3 HTTP ΕΠΙΚΟΙΝΩΝΙΑ .....	28
2.5 ΣΥΝΟΨΗ ΚΕΦΑΛΑΙΟΥ .....	30
2.6 ΒΙΒΛΙΟΓΡΑΦΙΑ .....	31
<b>ΚΕΦΑΛΑΙΟ 3 Οδηγίες Χρήσεις πακέτου Simon Says .....</b>	<b>34</b>
3.1 ΕΙΣΑΓΩΓΗ.....	34
3.2 ΚΑΝΟΝΕΣ ΤΟΥ SIMON SAYS.....	34
3.3 ΕΓΚΑΤΑΣΤΑΣΗ .....	35
3.3.1 ΑΠΑΙΤΗΣΕΙΣ ΓΗΣ .....	35
3.3.2 ΣΤΟΙΧΕΙΑ ΠΟΥ ΧΡΗΣΙΜΟΠΟΙΟΥΝΤΑΙ .....	37
3.3.2.1 SECOND LIFE .....	37
3.3.2.2 HOST SERVER .....	39
3.3.2.3 ARDUINO WEB BROWSER .....	39
3.3.3 ΤΡΟΠΟΣ ΥΛΟΠΟΙΗΣΗΣ .....	39
3.3.3.1 ΥΛΟΠΟΙΗΣΗ ΜΕ MARKETPLACE .....	40
3.3.3.2 ΥΛΟΠΟΙΗΣΗ ΕΚ ΤΟΥ ΜΗΔΕΝΟΣ .....	46
3.4 ΛΕΙΤΟΥΡΓΙΑ .....	48
3.4.1 ΤΟ INTERFACE ΣΤΟ SECOND LIFE.....	48
3.4.2 ΧΡΗΣΗ ΑΠΟΚΛΕΙΣΤΙΚΑ ΣΤΟ SECOND LIFE .....	50
3.4.3 ΧΡΗΣΗ ΜΕΣΩ ΦΥΛΛΟΜΕΤΡΗΤΗ (WEB BROWSER).....	53
3.4.4 ΔΙΕΠΑΦΗ ΣΕΛΙΔΑΣ .....	55
3.4.5 ΑΠΕΛΕΥΘΕΡΩΣΗ URL .....	57
3.4.6 ΕΠΕΞΕΡΓΑΣΙΑ ΑΝΤΙΚΕΙΜΕΝΩΝ ΚΑΙ DEBUGGING .....	57
3.4.7 ΣΗΜΕΙΩΣΕΙΣ .....	59
3.5 ΣΥΝΟΨΗ ΚΕΦΑΛΑΙΟΥ .....	59
<b>ΚΕΦΑΛΑΙΟ 4: Ανάλυση και Επεξεργασία Δεδομένων .....</b>	<b>60</b>

4.1 ΕΙΣΑΓΩΓΗ.....	60
4.2 SECOND LIFE .....	61
4.2.1 ΛΕΙΤΟΥΡΓΙΑ ΕΠΙΚΟΙΝΩΝΙΑΣ ΠΑΙΧΝΙΔΙΟΥ SIMON SAYS .....	63
4.2.2 ΛΕΙΤΟΥΡΓΙΑ ΔΕΥΤΕΡΕΥΟΝΤΩΝ ΕΠΙΚΟΙΝΩΝΙΩΝ .....	65
4.2.3 ΛΕΙΤΟΥΡΓΙΑ ΜΕΤΑΦΟΡΑΣ ΔΕΔΟΜΕΝΩΝ ΑΠΟ ΚΑΙ ΠΡΟΣ ΤΗΝ HTTP ΣΥΝΔΕΣΗ .....	67
4.2.4 ΤΟ ΑΝΤΙΚΕΙΜΕΝΟ INFO .....	68
4.3 WEB SERVER .....	69
4.3.1 ΔΕΔΟΜΕΝΑ ΠΡΟΣ SECOND LIFE .....	70
4.3.1.1 ΠΑΡΑΓΩΓΗ ΣΤΗ ΣΕΛΙΔΑ HTML .....	70
4.3.1.2 Η ΕΝΤΟΛΗ WINDOW.OPEN .....	71
4.3.1.3 ΤΟ ΠΕΔΙΟ IFRAME .....	72
4.3.1.4 ΑΠΟΛΑΒΗ ΔΕΔΟΜΕΝΩΝ ΣΤΟ SECOND LIFE .....	73
4.3.1.5 QR CODE .....	74
4.3.1.6 ΔΙΑΧΕΙΡΙΣΗ ΤΩΝ ΔΕΔΟΜΕΝΩΝ ΑΠΟ ΤΟ SECOND LIFE .....	74
4.3.2 ΔΕΔΟΜΕΝΑ ΑΠΟ SECOND LIFE.....	75
4.3.2.1 ΑΠΟΣΤΟΛΗ ΑΠΟ SECOND LIFE ΣΕ PHP ΑΡΧΕΙΟ .....	75
4.3.2.2 ΕΠΕΞΕΡΓΑΣΙΑ ΣΕ PHP .....	76
4.3.2.3 ΠΡΟΣΠΑΘΕΙΑ ΑΝΑΓΝΩΣΗΣ ΔΕΔΟΜΕΝΩΝ ΜΕ PHP .....	76
4.3.2.4 ΧΡΗΣΗ AJAX .....	77
4.3.2.5 ΑΝΑΓΝΩΣΗ ΔΕΔΟΜΕΝΩΝ ΣΤΗΝ HTML ΣΕΛΙΔΑ ΜΕ AJAX.....	77
4.3.2.6 ΕΠΕΞΕΡΓΑΣΙΑ ΔΕΔΟΜΕΝΩΝ ΣΤΗΝ HTML ΣΕΛΙΔΑ .....	78
4.4 ΠΡΩΤΟΚΟΛΛΑ ΕΠΙΚΟΙΝΩΝΙΑΣ ΚΑΙ ΠΕΡΙΟΡΙΣΜΟΙ.....	81
4.4.1 ΠΡΩΤΟΚΟΛΛΑ ΚΑΙ ΠΕΡΙΟΡΙΣΜΟΙ ΣΤΟ SECOND LIFE .....	81
4.4.1.1 ΚΑΝΑΛΙΑ ΕΠΙΚΟΙΝΩΝΙΑΣ .....	81
4.4.1.2 ΠΑΡΑΛΑΒΗ ΔΕΔΟΜΕΝΩΝ.....	82
4.4.1.3 ΑΠΟΣΤΟΛΗ ΔΕΔΟΜΕΝΩΝ .....	82
4.4.1.4 ΔΕΣΜΕΥΣΗ URL .....	83

4.4.2 ΠΡΩΤΟΚΟΛΛΑ ΚΑΙ ΠΕΡΙΟΡΙΣΜΟΙ ΣΤΟΝ WEB SERVER.....	84
4.4.2.1 ΕΠΙΚΟΙΝΩΝΙΑ HTML ΠΡΟΣ SECOND LIFE.....	84
4.4.2.2 IFRAME και SSL .....	85
4.4.2.3 ΕΠΙΚΟΙΝΩΝΙΑ SECOND LIFE ΠΡΟΣ ΡΗΡ.....	85
4.4.2.4 ΧΡΗΣΗ AJAX .....	86
4.5 ΣΥΝΟΨΗ ΚΕΦΑΛΑΙΟΥ .....	86
4.6 ΒΙΒΛΙΟΓΡΑΦΙΑ .....	87
<b>ΚΕΦΑΛΑΙΟ 5: Επικοινωνία Πλακέτας ARDUINO με Second Life .....</b>	<b>89</b>
5.1 ΕΙΣΑΓΩΓΗ.....	89
5.2 ΠΛΗΡΟΦΟΡΙΕΣ.....	89
5.2.1 ARDUINO .....	89
5.2.2 Visual Studio .....	90
5.3 ΕΓΚΑΤΑΣΤΑΣΗ .....	91
5.3.1 ΕΓΚΑΤΑΣΤΑΣΗ ΠΛΑΚΕΤΑΣ ARDUINO .....	92
5.3.2 ΕΓΚΑΤΑΣΤΑΣΗ Simon Browser .....	94
5.4 ΤΡΟΠΟΣ ΧΡΗΣΗΣ .....	94
5.5 ΛΕΙΤΟΥΡΓΙΑ .....	95
5.5.1 ΤΡΟΠΟΣ ΛΕΙΤΟΥΡΓΙΑΣ ARDUINO .....	95
5.5.2 ΤΡΟΠΟΣ ΛΕΙΤΟΥΡΓΙΑΣ SIMON BROWSER .....	96
5.6 ΣΥΝΟΨΗ ΚΕΦΑΛΑΙΟΥ .....	97
5.7 ΒΙΒΛΙΟΓΡΑΦΙΑ .....	97
<b>ΚΕΦΑΛΑΙΟ 6: Συμπεράσματα .....</b>	<b>99</b>
ΕΠΙΛΟΓΟΣ.....	100



## ΛΙΣΤΑ ΕΙΚΟΝΩΝ

---

Εικόνα 2.1. Στατιστικά που δημοσιοποίησε το Second Life εν όψει των εορτασμών για τα 10 χρόνια λειτουργίας του, το 2013. ....	17
Εικόνα 2.2. Το προκαθορισμένο script "Hello, Avatar". ....	20
Εικόνα 2.3.Κώδικας JavaScript που χρησιμοποιεί CSS για κουμπί με animation. ....	26
Εικόνα 2.4. Τρόπος επικοινωνίας client-server χωρίς AJAX και με AJAX. ....	28
Εικόνα 3.1. Μπάρα τοποθεσίας στο Second Life στην οποία φαίνονται οι απαγορεύσεις που ισχύουν. ....	35
Εικόνα 3.2. Μπάρα τοποθεσίας όπου φαίνονται πολλαπλές απαγορεύσεις building και scripting. ....	35
Εικόνα 3.3. Επιλογή της ενότητας World. ....	36
Εικόνα 3.4. Επιλογή της ενότητας World map (Ctrl+M για συντομία). ....	36
Εικόνα 3.5. Εμφάνιση του συνολικού κόσμου του Second Life και τηλεμεταφορά στην τοποθεσία Dreams. ....	37
Εικόνα 3.6. Η σελίδα του project στο Marketplace του Second Life. ....	40
Εικόνα 3.7. Προβολή του κουμπιού για την τοποθέτηση της παραγγελίας. ....	40
Εικόνα 3.8. Εμφάνιση του κουμπιού "Buy now" για την ολοκλήρωση της αγοράς. ....	41
Εικόνα 3.9. Η παραγγελία έχει πραγματοποιηθεί και το project θα παραδοθεί σύντομα. ....	41
Εικόνα 3.10. Άνοιγμα Inventory μέσα στο Second Life Viewer. ....	42
Εικόνα 3.11. Εντοπισμός του νεοαποκτηθέντος Project στο Inventory. ....	42
Εικόνα 3.12. Εντοπισμός του νεοαποκτηθέντος Project στο Inventory. ....	42
Εικόνα 3.13. Περιεχόμενο των οδηγιών στο Notecard. ....	43
Εικόνα 3.14. Το object της εργασίας στον φάκελο που αποστάλθηκε από το Marketplace. ....	43
Εικόνα 3.15. Εφαρμογή του αντικειμένου στον κόσμο του Second Life (Rez) σε μη ιδανικό σημείο. ....	44
Εικόνα 3.16. Παραμετροποίηση του object με Edit. ....	44
Εικόνα 3.17. Μετακίνηση της πλακέτας στους άξονες Z και Y. ....	45
Εικόνα 3.18. Μετακίνηση της πλακέτας στους άξονες Z και Y. ....	45
Εικόνα 3.19. Περιστροφή της ταμπλέτας γύρω από τον άξονα Y. ....	45
Εικόνα 3.20. Επιλογή του σημείου Down για τέλεια καθετότητα. ....	46
Εικόνα 3.21 Το interface του Project "Simon Says.v3". ....	48
Εικόνα 3.22. Το κουμπί START. ....	49
Εικόνα 3.23. Το κουμπί CONNECTOR. ....	49
Εικόνα 3.24. Το κουμπί INFO. ....	49

Εικόνα 3.25. Το κουμπί HARD RESET.....	49
Εικόνα 3.26. Η εναλλαγή του κουμπιού START σε RESET .....	50
Εικόνα 3.27. Τα 4 χρωματιστά κουμπιά.....	50
Εικόνα 3.28. Μήνυμα από το αντικείμενο «Simon Says.v3».....	50
Εικόνα 3.29. Τα προβαλλόμενα μηνύματα στην περίπτωση λάθος επιλογής χρώματος του χρήστη. 51	
Εικόνα 3.30. Το παράθυρο με πληροφορίες που εμφανίζεται πατώντας το κουμπί INFO.....	52
Εικόνα 3.31. Το URL που εμφανίζει το κουμπί "CONNECTOR".....	54
Εικόνα 3.32. Το παράθυρο του φυλλομετρητή του Second Life. ....	54
Εικόνα 3.33. Η διεπαφή της σελίδας όπως φαίνεται σε Web Browser υπολογιστή (PC). ....	56
Εικόνα 3.34. Η διεπαφή της σελίδας όπως εμφανίζεται σε Web Browser κινητής συσκευής (smartphone).....	56
Εικόνα 3.35. Προβαλλόμενο μήνυμα σε περίπτωση «HARD RESET».....	57
Εικόνα 3.36. Το κουμπί Unlink που καταστεί δυνατή την επί μέρους επεξεργασία του αντικειμένου. .....	58
Εικόνα 4.1. Εκτέλεση της συνάρτησης SimonGame με το κλικάρισμα του κόκκινου ή του πράσινου κουμπιού.....	70
Εικόνα 4.2. Εκτέλεση της συνάρτησης StartGame με το κλικάρισμα του κουμπιού "START GAME". .	70
Εικόνα 4.3. Η συνάρτηση SimonGame.....	71
Εικόνα 4.4. Η συνάρτηση StartGame. ....	71
Εικόνα 4.5. Η εξαγωγή της διεύθυνσης του URL του Second Life. ....	71
Εικόνα 4.6. Το πεδίο iframe. ....	72
Εικόνα 4.7. Το συμβάν http_request και ο έλεγχος μεθόδου. ....	73
Εικόνα 4.8. Browser του Second Life με τον παραχθέν QR κωδικό. ....	74
Εικόνα 4.9. Η κατηγοριοποίηση των δεδομένων.....	74
Εικόνα 4.10. Η απόκτηση των δεδομένων από το αντικείμενο PARENT.....	75
Εικόνα 4.11. Το αρχείο PHP.....	76
Εικόνα 4.12. Απόκτηση δεδομένων μέσω AJAX.....	78
Εικόνα 4.13. Έλεγχος δεδομένων και εκτέλεση flashing κουμπιών.....	79
Εικόνα 4.14. Η συνάρτηση που είναι υπεύθυνη για το blinking (flashing) του κόκκινου κουμπιού....	79
Εικόνα 4.15. Περίπτωση εμφάνισης χαρακτήρα «Ο» στα δεδομένα υποδεικνύοντας «GAME OVER». .....	80
Εικόνα 4.16. Η συνάρτηση zero.....	80
Εικόνα 4.17. Μήνυμα που υπάρχει στην εγκυκλοπαίδεια του Second Life. ....	83
Εικόνα 5.1. Συνδεσμολογία Arduino με κουμπιά. ....	92

Εικόνα 5.2. Μήνυμα επιτυχής φόρτωσης κώδικα στη πλακέτα Arduino. ....	93
Πίνακας 2.1. Τα χρώματα της εργασίας στο Second Life. ....	21
Πίνακας 4.1. Εύρος ακρόασης διάφορων συναρτήσεων εντός του Second Life. ....	80
Γράφημα 4.1. Η διάταξη επικοινωνίας του συνολικού Project. ....	59
Γράφημα 4.2. Η επικοινωνία των αντικειμένων στο Second Life. ....	61
Γράφημα 4.3. Απεικόνιση της επικοινωνίας εντός του Web Server. ....	68



## ΚΕΦΑΛΑΙΟ 1: Αντικείμενο και Δομή

### 1.1 ANTIKEIMENO ΕΡΓΑΣΙΑΣ

Στο σημερινό τεχνολογικό τοπίο, η μεταφορά δεδομένων και η επικοινωνία διαφορετικών αντικειμένων, τόσο σε εικονικό περιβάλλον όσο και στον πραγματικό κόσμο, αποτελεί έδαφος συνεχούς έρευνας και ανάπτυξης.

Ειδικότερα, στο προσκήνιο εμφανίζονται τεχνολογίες αναμετάδοσης (broadcast) διαφόρων μέσων, όπως είναι οι ταινίες (πχ Netflix) και παιχνίδια (πχ το επερχόμενο Google Stadia). Στη βάση των συγκεκριμένων τεχνολογιών υφίσταται η βασική επικοινωνία μέσω του διαδικτύου με τα διαφορετικά μέσα.

Μια τεχνολογία που χρησιμοποιείται ευρέως σήμερα για επικοινωνία διαφορετικών μέσων μέσω του διαδικτύου, είναι η επικοινωνία HTTP η οποία, αν και δεν αποτελεί σύγχρονη τεχνολογία των τελευταίων ετών, υποστηρίζεται από νέα εργαλεία που κάνουν την εμφάνιση τους καθημερινώς, με αποτέλεσμα το συγκεκριμένο πρωτόκολλο να χρησιμοποιείται ανελλιπώς.

Ωστόσο, η HTTP επικοινωνία αποτελεί ένα μόνο κομμάτι για την επίτευξη επικοινωνίας εικονικών αντικειμένων με άλλες συσκευές. Οπότε, ιδιαίτερη μέριμνα απαιτείται στην επεξεργασία των δεδομένων τόσο πριν την αποστολή τους μέσω HTTP επικοινωνίας, όσο και κατά την παραλαβή τους, ειδικά όταν η επικοινωνία είναι αμφίδρομη.

### 1.2 ΣΚΟΠΟΣ ΤΗΣ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ

Τη μελέτη και την περαιτέρω έρευνα πάνω στη διαχείριση των δεδομένων σε μια HTTP επικοινωνία, καλείται να καλύψει η συγκεκριμένη διπλωματική εργασία μέσω του κύριου προϊόντος που δημιουργήθηκε και ακολούθως μέσω των διαφορετικών λειτουργιών που υλοποιήθηκαν σε διαφορετικών ειδών συσκευές.

Για τη δημιουργία κεντρικού προϊόντος το οποίο θα επεξεργαζόταν και θα μετέδιδε δεδομένα επιλέχθηκε να υλοποιηθεί ένα διαδραστικό παιχνίδι, το Simon Says.

Ωστόσο, για τη δημιουργία του παιχνιδιού και την υλοποίηση του πλήθους των λειτουργιών απαιτήθηκε κάποιου είδους περιβάλλοντος το οποίο θα επέτρεπε την σχετικά εύκολη δημιουργία του παιχνιδιού αλλά και την επίτευξη της επικοινωνίας.

Για αυτούς τους σκοπούς επιλέχθηκε το εικονικό κοινωνικό δίκτυο Second Life το οποίο επιτρέπει τη δημιουργία, τον προγραμματισμό αλλά και την επικοινωνία του παιχνιδιού με ευκολία ενώ με αυτόν τον τρόπο εξερευνώνται και οι πολλαπλές άλλες δυνατότητες του.

Ακόμη, ερευνήθηκε η διαχείριση και η μεταφορά των δεδομένων του παιχνιδιού αμφίδρομα σε εξωτερική συσκευή, μέσω HTTP επικοινωνίας, ενώ ακόμη και για τις εξωτερικές συσκευές ερευνήθηκε και η χρήση σειριακών δεδομένων τα οποία προκύπτουν από πλακέτα μικροελεγκτή.

### 1.3 ΔΟΜΗ ΤΗΣ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ

Το **2<sup>ο</sup> Κεφάλαιο** παρουσιάζει το αρχικό περιβάλλον στο οποίο γίνεται η υλοποίηση της εργασίας και είναι το Second Life. Εκεί αναφέρονται ιστορικά στοιχεία της πλατφόρμας ενώ αναλύονται οπτικά χαρακτηριστικά του περιβάλλοντος που χρησιμοποιούνται. Ακόμη γίνεται η παρουσίαση τεχνολογιών, κυρίως του οπτικού τομέα που χρησιμοποιούνται για τη δημιουργία Web Server ο οποίος επικοινωνεί με το Second Life.

Το **3<sup>ο</sup> Κεφάλαιο** παρουσιάζει τις οδηγίες χρήσης του τελικού προϊόντος στο Second Life, αναλύει τους τρόπους εγκατάστασης και χρήσης του προϊόντος τόσο εντός Second Life όσο και στον Web Server ενώ τέλος αναφέρει τα διαφορετικά αρχεία κώδικα που χρησιμοποιήθηκαν.

Το **4<sup>ο</sup> Κεφάλαιο** αναλύει σε βάθος τον τρόπο λειτουργίας του προϊόντος στο Second Life, τον τρόπο λειτουργίας του Web Server ενώ γίνεται μελέτη της επεξεργασίας των δεδομένων σε κάθε στάδιο της επικοινωνίας. Τέλος μελετώνται οι διαφορετικοί περιορισμοί και τα πρωτόκολλα που ισχύουν σε κάθε επίπεδο της επικοινωνίας.

Το **5<sup>ο</sup> Κεφάλαιο** αναλύει έναν διαφορετικό τρόπο εξωτερικής επικοινωνίας με το κεντρικό πρόγραμμα στο Second Life και περιλαμβάνει έναν μικροελεγκτή. Εκεί περιέχονται ο τρόπος εγκατάστασης της πλακέτας του μικροελεγκτή, ο τρόπος χρήσης του αλλά και ο τρόπος με τον οποίο λειτουργεί.

Το **6<sup>ο</sup> Κεφάλαιο** περιγράφει το αποτέλεσμα της δημιουργίας του συγκεκριμένου προϊόντος, της χρησιμότητας του ενώ αναφέρει τυχόν καθυστερήσεις που εμφανίζονται σε αυτό.

## ΚΕΦΑΛΑΙΟ 2: Second Life και Εργαλεία Server

### 2.1 ΕΙΣΑΓΩΓΗ

Το Second Life αποτελεί κοινωνικό μέσο το οποίο χρησιμοποιούν εδώ και πολλά χρόνια εκατομμύρια χρήστες. Πέρα από την επικοινωνία μεταξύ των χρηστών παρέχει οικόπεδα εντός του κόσμου του Second Life, έναντι κανονικών χρημάτων, στα οποία οι χρήστες μπορούν να δημιουργήσουν μια οικία ή κάποιο άλλο κατασκεύασμα, όπως το επιθυμούν και το φαντάζονται.

Ακόμα, παρέχει πληθώρα εργαλείων στους χρήστες τους, με τα κυριότερα να είναι το χτίσιμο (building) και ο προγραμματισμός (scripting) αντικειμένων. Και τα δύο εργαλεία, βρίθουν από λεπτομερείς οδηγούς, παραμετροποιημένα εργαλεία άλλων χρηστών διαθέσιμα στη διαδικτυακή αγορά του Second Life ( Second Life Marketplace) και δημιουργικές δυνατότητες οι οποίες σπάνια εμφανίζονται σε περιβάλλοντα χρήστη, σε τέτοια μεγέθη. Μέσα σε αυτές συγκαταλέγονται δυνατότητες δημιουργίας εντυπωσιακών, εμφανισιακά, κατασκευών, παραμετροποίηση της εμφάνισης του χαρακτήρα του χρήστη με πολύ μεγάλη λεπτομέρεια και η δημιουργία πολύπλοκων πλατφόρμων εντός του εικονικού κόσμου, που αυξάνουν την προσομοίωση του Second Life, ως ολόκληρος κόσμος, σε πολύ ρεαλιστικά επίπεδα.

Μερικές από αυτές τις δυνατότητες, επιτρέπουν στους χρήστες του Second Life να δημιουργούν αντικείμενα τα οποία επικοινωνούν με εξωτερικό Web Server, επιτρέποντας την HTTP επικοινωνία.

Σε αυτούς τους Web Server, μπορούν να τοποθετηθούν αρχεία τα οποία διαχειρίζονται δεδομένα από και προς το Second Life. Τα αρχεία αυτά είναι σελίδες HTML οι οποίες χρησιμοποιούν CSS κομμάτια κώδικα και JavaScript εντολές ώστε να δημιουργήσουν ένα φιλικό περιβάλλον χρήστη. Ταυτόχρονα γίνεται χρήση προγραμματισμένων γραφικών που καταστούν λειτουργικό το όλο εγχείρημα ενώ προσθέτουν εμφανισιμότητα η οποία αυξάνει την αποδεκτικότητα τους.

### 2.2 SECOND LIFE

Το Second Life αποτελεί διαδικτυακό εικονικό κόσμο ο οποίος διέπεται δωρεάν από εγγεγραμμένους χρήστες στην πλατφόρμα του, μέσω του αυθεντικού Second Life Viewer ή μέσου ενναλακτικών Third-Party Viewer[1][2]. Τα πολλαπλά του χαρακτηριστικά, το έκαναν γρήγορα πολύ δημοφιλές ενώ μέχρι και σήμερα επιλέγεται από χιλιάδες κόσμου για την ανάπτυξη και διεκπεραίωση διάφορων έργων (project) κυρίως εκπαιδευτικού χαρακτήρα. Ταυτόχρονα, η διαρκής υποστήριξη του από τη μητρική εταιρεία Linden Lab την οποία απασχολεί η τακτική ανανέωση των περιεχομένων και των

δυνατοτήτων του, φροντίζει την διατήρηση αρκετών τακτικών χρηστών και την έλευση νέων[3]. Η υπηρεσία του Second Life, απευθύνεται σε ανθρώπους άνω των 16 ετών, με εξαίρεση έφηβους 13 με 15 ετών που μπορούν να χρησιμοποιούν το Second Life για να επισκεφθούν ορισμένους εκπαιδευτικούς προορισμούς.

Το κύριο χαρακτηριστικό του συγκεκριμένου εικονικού κόσμου είναι ότι δεν αποτελεί παιχνίδι με την πλήρη έννοια του όρου, δηλαδή δεν εμπεριέχει κάποιον στόχο επίτευξης τον οποίο οι χρήστες προσπαθούν να εκπληρώσουν[4]. Οι χρήστες του Second Life, ονομάζονται και κάτοικοι, δημιουργούν μια εικονική αναπαράσταση του εαυτού τους (avatar) και έχουν τη δυνατότητα να αλληλεπιδρούν με τον κόσμο, με αντικείμενα και με άλλους χρήστες. Έχουν τη δυνατότητα να περιηγηθούν σε ολόκληρο τον κόσμο του Second Life, γνωστό και ως grid, να επισκεφθούν εκατοντάδες προορισμούς που δημιούργησε το ίδιο το Second Life ή και άλλοι χρήστες (Destinations), να γνωρίσουν άλλους χρήστες, να κοινωνικοποιηθούν, να συμμετάσχουν σε ατομικές και ομαδικές δραστηριότητες, να κατασκευάσουν και να δημιουργήσουν πολύπλοκα αντικείμενα που μπορούν και να προγραμματίσουν, να αγοράσουν και να πουλήσουν αντικείμενα εντός και εκτός του Second Life, να ανταλλάξουν εικονική ιδιοκτησία και να προσφέρουν εικονικές υπηρεσίες σε κάποιον άλλον χρήστη. Οι δυνατότητες που παρέχονται είναι πάρα πολλές, ενώ και οι ίδιοι οι χρήστες διατηρούν τον ζωντανό εικονικό κόσμο παρέχοντας και δημιουργώντας διαρκώς νέα πράγματα.

Στο Second Life, ανά διαστήματα συμμετείχαν πολλοί φορείς, εταιρείες και καλλιτέχνες που προκάλεσαν έκπληξη. Μερικοί από αυτούς του φορείς είναι εκκλησιαστικές οργανώσεις του Χριστιανισμού και του Ισλάμ ( LifeChurch.tv και Islam Online), προξενία των Μαλδιβών, της Σουηδίας, της Σερβίας και πολλών άλλων χωρών ενώ και το Ιρλανδικό συγκρότημα U2 έχει πραγματοποιήσει ζωντανή συναυλία εντός του Second Life.

Τέλος, τα τελευταία χρόνια υπάρχει μεγάλη κινητικότητα από ομάδες και κοινότητες σχετικές με σεξουαλικά κινήματα, καθώς η κοινωνική πτυχή της ψηφιακής πλατφόρμας βοηθάει τα άτομα που αισθάνονται κοινωνικά αποκλεισμένα.

### 2.2.1 ΙΣΤΟΡΙΑ

Το Second Life αναπτύχθηκε από την εταιρεία Linden Lab η οποία εδρεύει στο San Francisco. Κυκλοφόρησε στις 23 Ιουνίου του 2003 ενώ μέχρι και σήμερα απασχολεί περίπου 800 χιλιάδες χρήστες[5].

Ήδη από το 1999 ο Philip Rosedale ίδρυσε την εταιρεία Linden Lab με σκοπό της την ανάπτυξη υπολογιστικών εξαρτημάτων για χρήσεις Εικονικής Πραγματικότητας (Virtual Reality). Στο αρχικό της



στάδιο, η Linden Lab δημιούργησε μια κατασκευή με οθόνες, οι οποίες φοριόντουσαν στην πλάτη του χρήστη, όμως δεν κατάφερε να κάνει το συγκεκριμένο εγχείρημα, διαθέσιμο στην αγορά[6].

Έτσι, η εταιρεία άλλαξε κατεύθυνση και στράφηκε στη δημιουργία ενός εικονικού τρισδιάστατου κόσμου, που ονόμασε Linden World, στον οποίο οι χρήστες θα μπορούν να παίζουν αλλά και να κοινωνικοποιούνται. Η προσπάθεια αυτή εν τέλει, μετατράπηκε στο γνωστό Second Life.



Εικόνα 2.1. Στατιστικά που δημοσιοποίησε το Second Life εν όψει των εορτασμών για τα 10 χρόνια λειτουργίας του, το 2013.

Το 2005 και το 2006, το Second Life δέχθηκε την προσοχή των ΜΜΕ. Ενδιαφέρον αποτελεί η περίπτωση του avatar Anshe Chung ο οποίος έγινε γνωστός στο ευρύ κοινό εξαιτίας της οικονομικής επιτυχίας που δέχτηκε, μέσω των οικονομικών ευκαιριών εντός του Second Life[7]. Εκείνη την περίοδο εμφανίστηκε κατακόρυφη αύξηση των χρηστών στο Second Life.

Ο μέγιστος αριθμός ταυτόχρονων συνδεδεμένων χρηστών, εμφανίστηκε το 2009 και ήταν 88.200[5]. Όμως, από το 2010, παρουσιάστηκε πτώση στους συνδεδεμένους χρήστες με την εταιρεία να δικαιολογεί την πτώση εξαιτίας της αλλαγής πολιτικής και όρων λειτουργίας που είχαν σαν αποτέλεσμα να διαγραφούν πολλοί ψεύτικοι χρήστες (bot). Την ίδια χρονιά ωστόσο, απολύθηκε το 30% του εργατικού δυναμικού της εταιρείας[5]. Μέχρι τότε όμως οι εγγεγραμμένοι λογαριασμοί είχαν φτάσει τους 21.3 εκατομμύρια χρήστες[8].

### 2.2.2 ΟΙΚΟΝΟΜΙΑ

Το Second Life διαθέτει δική του οικονομία η οποία επιτρέπει την αγορά, την πώληση αλλά και τη μίσθωση περιουσίας εντός του Second Life. Χρησιμοποιεί δικό του χρηματοπιστωτικό σύστημα, ενώ το νόμισμα του ονομάζεται Linden Dollar (L\$). Ένα Linden Dollar αντιστοιχεί σε 0.00299 ευρώ (EUR) ή 0.00327 αμερικανικά δολάρια (USD)[9]. Το 2013, εν μέσω 10 ετών λειτουργίας του Second Life, διακινήθηκαν εντός του εικονικού κόσμου, 3,2 δισεκατομμύρια δολάρια Αμερικής (USD) [10] (Εικόνα 2.1).

Ωστόσο, μεγάλης σημασίας παράγοντας στην ανάπτυξη της οικονομίας της πλατφόρμας, αποτέλεσε η επιχειρηματική δράση, η οποία άνθισε εντός Second Life και που δημιούργησε νέες προοπτικές και χρήσεις. Η δράση αυτή σχετίστηκε γύρω από τη δημιουργία εικονικών αντικειμένων και προγραμμάτων τα οποία μπορούν να αποτελέσουν εμπορικά αγαθά. Ακόμη, προσέφερε πλήρη δικαιώματα στους δημιουργούς των αντικειμένων, κάτι που επέτρεψε σε φιλόδοξους δημιουργούς, να τοποθετήσουν στην αγορά του Second Life (Second Life Marketplace) πάνω από 2,1 εκατομμύρια αντικείμενα[10]. Ειδικό ενδιαφέρον παρουσιάζει η Ailin Graef[11] (πιο γνωστή με το avatar Anshe Chung) η οποία το 2006 έγινε η πρώτη εκατομμυριούχος του Second Life, δουλεύοντας ως μεσίτρια εικονικών οικοπέδων, αγοράζοντας, πουλώντας και μισθώνοντας εικονική γη. Με αρχικό ποσό τα US\$9.95 κατάφερε εντός 2μηνων να αποκομίσει πάνω από 1 εκατομμύριο αμερικανικά δολάρια (USD)[12].

Τα αντικείμενα που διατίθενται εντός Second Life Marketplace ποικίλουν από ρούχα, αξεσουάρ, χτενίσματα και κοσμήματα για τα avatar των χρηστών, μέχρι κατασκευές, παιχνίδια, οχήματα και έργα τέχνης. Ακόμη υπηρεσίες μπορούν να γίνουν αντικείμενο αγοροπωλησίας και αυτές μπορούν να είναι :

- 1) Διοίκηση Επιχειρήσεων
- 2) Ψυχαγωγία
- 3) Δημιουργία Πρωτότυπου (custom-made) Περιεχόμενου
  - Χτίσιμο (Building)
  - Δημιουργία Υφών (Texturing)
  - Προγραμματισμός (Scripting)
  - Δημιουργία Γραφικών (Animating)
  - Καλλιτεχνική Επιμέλεια (Art Direction)
  - Παραγωγός/Χρηματοδότης έργου (Position of Producer/Project Funder)

Στο Second Life Marketplace έχουν όλοι οι χρήστες δωρεάν πρόσβαση, ενώ κάθε αντικείμενο φέρει δική του τιμή σε L\$ (για τα δωρεάν αντικείμενα είναι L\$0). Κάθε χρήστης μπορεί να πουλήσει δικό του δημιούργημα στο Second Life Marketplace αρκεί να έχει καταχωρήσει έναν τρόπο πληρωμής (συνήθως λογαριασμό PayPal) και να έχει πραγματοποιήσει μια χρηματική συναλλαγή με αυτόν τον τρόπο πληρωμής (ελάχιστο ποσό συναλλαγής είναι τα 2.5 δολάρια Αμερικής (US\$), δηλαδή 640 Linden Dollars (L\$))[13].

### 2.2.3 LSL (Linden Scripting Language)

Η LSL αποτελεί τη γλώσσα προγραμματισμού του Second Life στα αρχεία script που χρησιμοποιούνται στα αντικείμενα και στα avatar του κόσμου. Αποτελεί προϊόν συλλογικής προσπάθειας από την εταιρεία Linden και την κοινότητα του Second Life με στόχο την παροχή ακριβούς και ανοιχτής τεκμηριωμένης πηγής πόρων για προγραμματιστές κάθε επιπέδου.

Η LSL βασίζεται κυρίως στις γλώσσες Java και C εισάγοντας σε αυτές ειδικές συναρτήσεις και εντολές που διευκολύνουν τον προγραμματισμό στο Second Life[14]. Τα προγράμματα που κατασκευάζονται με LSL ονομάζονται scripts και έχουν πολλές χρήσεις, εντός του Second Life. Τα script μπορούν να εισαχθούν σε αντικείμενα αλλά όχι σε avatar, ωστόσο avatar μπορούν να φορούν αντικείμενα που εμπεριέχουν scripts[14].

Η εκτέλεση των Second Life script συμβαίνει στον Second Life Server (sims) και όχι στον client (Viewer), παρόλο που η επεξεργασία του προγράμματος συμβαίνει στον client[14]. Τα αποτελέσματα της εκτέλεσης του προγράμματος στέλνονται στον client μέσω του διαδικτύου (network)[14].

Σημαντικό στοιχείο λειτουργίας της LSL γλώσσας είναι η έμφαση που προδίδει στα πεδία «Καταστάσεις» και «Συμβάντα» (“States” and “Event”) τα οποία ουσιαστικά χρησιμεύουν για την επικοινωνία του script με τον εικονικό κόσμο. Πολλές ρεαλιστικές κινήσεις του εικονικού κόσμου μπορούν να χαρακτηριστούν ως διαφορετικά «state» και «event» όπως είναι το εικονικό άνοιγμα και κλείσιμο μιας πόρτας, το άναμμα ή σβήσιμο μιας λάμπας κ.α.[14].

Ο ελάχιστος αριθμός καταστάσεων «state» που μπορεί να έχει ένα script, είναι ένας και αυτή η κατάσταση θα είναι η προκαθορισμένη κατάσταση (default state)[14].

```

default
{
    state_entry()
    {
        llSay(0, "Hello, Avatar!");
    }

    touch_start(integer total_number)
    {
        llSay(0, "Touched.");
    }
}

```

Εικόνα 2.2. Το προκαθορισμένο script "Hello, Avatar".

Ένα συμβάν (event) μπορεί να θεωρηθεί ως ένας σκανδαλισμός και τα event που μπορούν να προκύψουν στο Second Life είναι προκαθορισμένα. Μπορούν να προκληθούν και από αντικείμενα και από avatar που αλληλεπιδρούν με τον κόσμο ή να δημιουργηθούν εντός script[14]. Τα συμβάντα (event) σκανδαλίζουν τους χειριστές συμβάντος (event handlers) τα οποία μπορούν να προγραμματιστούν κατά το δοκούν. Ένα παράδειγμα συμβάντος είναι το touch\_start το οποίο σκανδαλίζεται όταν ένας χρήστης «ακουμπάει» το αντικείμενο (touch). Για να ακουμπήσει ένα αντικείμενο, ο χρήστης θα πρέπει απλά να επιλέξει το αντικείμενο με το αριστερό κλικ του ποντικιού.

Το πιο απλό πρόγραμμα που μπορεί να γραφτεί στο Second Life, είναι και αυτό που εμφανίζεται αυτόματα όταν ο χρήστης επιλέξει εντός του εικονικού κόσμου να δημιουργήσει νέο script. Το συγκεκριμένο script απλά εμφανίζει στον χρήστη το κείμενο «Hello, Avatar!» μόλις εκείνος εμφανίσει το αντικείμενο, ενώ όταν ακουμπήσει το αντικείμενο, εμφανίζεται το κείμενο «Touched.»(Εικόνα 2.2).

Το συγκεκριμένο script διαθέτει 1 state (default), 2 events (state\_entry και touch\_start) και 2 functions (llSay(0, "Hello, Avatar!") και llSay(0, "Touched.")).

Τα script έχουν πολλές χρησιμότητες με μερικές από αυτές να επιτρέπουν τους χρήστες να επικοινωνούν, να κάνουν αντικείμενα να μετακινούνται, να αλλάζουν μέγεθος και χρώμα και τέλος να αποτελούν μέρος μιας πολυπλοκότερης προγραμματιστικής οντότητας, όπως σε αυτήν την εργασία.

#### 2.2.4 SECOND LIFE WORLD

Ο κόσμος του Second Life εκτελείται στον Second Life Viewer (client) ο οποίος βρίσκεται στον υπολογιστή του χρήστη με τη βοήθεια αρκετών χιλιάδων server της Linden Lab. Ο client υλοποιεί τον τρισδιάστατο κόσμο χρησιμοποιώντας OpenGL τεχνολογία. Η συγκεκριμένη τεχνολογία εκδόθηκε το 1992 από την εταιρεία Silicon Graphics, πριν από 27 χρόνια, και εν μέρει αυτό δικαιολογεί την αίσθηση παλιών γραφικών στο Second Life[15].

Επίσης, ο κόσμος του Second Life απαρτίζεται από περιοχές (regions). Το σύνολο των περιοχών του Second Life ονομάζεται grid. Κάθε περιοχή του Second Life έχει μέγεθος 256 επί 256 μέτρα και εκτελείται σε έναν αποκλειστικό πυρήνα του πολυπύρηνου server[5].

Σε κάθε region μπορούν να δημιουργηθούν, αγοραστούν ή πολλαπλασιαστούν αντικείμενα. Κάθε αντικείμενο στον κόσμο του Second Life αναφέρεται ως asset.

Αυτά περιλαμβάνουν:

- Τρισδιάστατα πολυγωνικά αντικείμενα που είναι πιο γνωστά ως Primitive Mesh (prims για συντομία).
- Ψηφιακές Εικόνες (αναφέρονται ως textures) που διακοσμούν τα prims.
- Ψηφιοποιημένα αρχεία ήχου.
- Εμφάνιση και σχήμα του avatar.
- Υφές δέρματος του avatar.
- Προγράμματα LSL (LSL scripts).
- Πληροφορίες γραμμένες σε κάρτες (notecards)

Κάθε ένα από αυτά τα asset φέρει έναν μοναδικό κωδικό UUID με το οποίο μπορεί να αναφερθεί και να επεξεργαστεί[16]. Όλα τα asset του Second Life είναι αποθηκευμένα σε συστοιχίες αποθήκευσης Isilon Systems (Isilon Systems storage clusters)[17]. Μέχρι το 2007 το συνολικό μέγεθος όλων των αντικειμένων των χρηστών μετρήθηκε ότι καταλαμβάνει 100 terabytes στους server χωρητικότητας[5]. Οι server που έχουν αποθηκευμένα τα αντικείμενα είναι διαφορετικοί από τους server που εκτελούν την κάθε περιοχή (region), όμως με τη φόρτωση (load) κάθε καινούργιου αντικειμένου σε μια περιοχή, οι προσομοιωτές της περιοχής (region simulators) κάνουν αίτηση για τις πληροφορίες του αντικειμένου στους server των αντικειμένων (asset). Οι προσομοιωτές περιοχής είναι γνωστοί και ως sims.

Επιπλέον, σε κάθε server υπάρχει φυσική προσομοίωση (physics simulation) που ελέγχει τις αλληλεπιδράσεις και τις συγκρούσεις όλων των αντικειμένων στην περιοχή. Τα αντικείμενα είναι

δυνατόν είτε να έχουν φυσική υπόσταση είτε να μην έχουν ενώ ακόμα είναι δυνατόν να μετακινούνται ή και όχι.

Τέλος, τα avatar των χρηστών μέσα στον κόσμο του Second Life, μεταχειρίζονται ως φυσικά αντικείμενα, ώστε να έχουν τη δυνατότητα να αλληλεπιδρούν με άλλα αντικείμενα.

Από το 2014 χρησιμοποιείται η μηχανή φυσικής προσομοίωσης Havok 2011.2 για όλες τις δυναμικές του κόσμου[18]. Η εταιρεία Linden Lab προωθεί γενικά τις τεχνολογίες ανοικτού κώδικα και χρησιμοποιεί δωρεάν και ανοιχτού κώδικα λογισμικό όπως Apache, MySQL και Linux[19].

## 2.3 ΙΔΙΟΤΗΤΕΣ ΑΝΤΙΚΕΙΜΕΝΩΝ SECOND LIFE

Τα αντικείμενα στο Second Life επιτρέπουν σε πολύ μεγάλο βαθμό την παραμετροποίηση των φυσικών τους ιδιοτήτων αλλά και των λειτουργιών τους. Αυτές οι παραμετροποιήσεις μπορούν να συμβούν είτε με τη χρήση του εύκολου εργαλείου Edit που προσφέρει ο Second Life Viewer για την επεξεργασία των αντικειμένων είτε μέσω προγραμματισμού του αντικειμένου (scripting).

Στη συγκεκριμένη εργασία, οι παραμετροποιήσεις που λαμβάνουν χώρα στα αντικείμενα είναι η αλλαγή του χρώματος τους, η αλλαγή των υφών τους (textures) και η τοποθέτηση επιπλέοντος κειμένου (floating text). Επίσης πραγματοποιείται επικοινωνία μεταξύ των αντικειμένων η οποία θα καλυφθεί στο Κεφάλαιο 4 λεπτομερώς.

### 2.3.1 ΑΛΛΑΓΗ ΧΡΩΜΑΤΟΣ

Τα χρώματα των αντικειμένων στο Second Life μπορούν να αλλάζουν μέσω scripting με την εντολή `llSetColor( vector color, integer face )`. Το διάνυσμα `color` ορίζει το χρώμα στο οποίο θα αλλάξει το αντικείμενο ενώ η μεταβλητή `face` ορίζει την όψη του αντικειμένου η οποία θα αλλάξει χρώμα. Η μεταβλητή `face` μπορεί να είναι 0 ή κάποιος φυσικός αριθμός, ανάλογα τον αριθμό των όψεων του αντικειμένου (για παράδειγμα, ο κύβος έχει 6 όψεις, άρα οι αριθμοί των όψεων είναι από 0 έως 5). Ακόμη, μπορεί να είναι -1 (ή ολογράφως `ALL_SIDES`) έτσι ώστε να αλλάξουν χρώμα όλες οι όψεις του αντικειμένου[20].

Το διάνυσμα του χρώματος έχει τη μορφή `<0.0, 0.0, 0.0>` όπου κάθε δεκαδικός αριθμός δηλώνει ένα εκ των χρωμάτων κόκκινου, πράσινου και μπλε (`<r, g, b>`)[20].

Τα χρώματα που χρησιμοποιούνται σε αυτήν την εργασία είναι τα χρώματα κόκκινο, πράσινο, μπλε και κίτρινο σε φωτεινές και σκούρες εκδοχές τους (BRIGHT and DARK).

Τα χρώματα που χρησιμοποιήθηκαν εντός του Second Life:

ΚΟΚΚΙΝΟ ΣΚΟΤΕΙΝΟ	0.6, 0.2, 0.0
ΚΟΚΚΙΝΟ ΦΩΤΕΙΝΟ	1.0, 0.0, 0.0
ΠΡΑΣΙΝΟ ΣΚΟΤΕΙΝΟ	0.0, 0.6, 0.0
ΠΡΑΣΙΝΟ ΦΩΤΕΙΝΟ	0.0, 1.0, 0.0
ΜΠΛΕ ΣΚΟΤΕΙΝΟ	0.0, 0.0, 1.0
ΜΠΛΕ ΦΩΤΕΙΝΟ	0.0, 0.8, 1.0
ΚΙΤΡΙΝΟ ΣΚΟΤΕΙΝΟ	0.8, 0.6, 0.0
ΚΙΤΡΙΝΟ ΦΩΤΕΙΝΟ	1.0, 1.0, 0.0

Πίνακας 2.1. Τα χρώματα της εργασίας στο Second Life.

### 2.3.2 ΥΦΕΣ ΑΝΤΙΚΕΙΜΕΝΩΝ (TEXTURES)

Οι υφές των αντικειμένων στο Second Life μπορούν να αλλάζουν μέσω του προγραμματισμού του αντικειμένου. Οι υφές που χρησιμοποιούνται μπορεί να είναι μοτίβο υλικών ή φυσικών επιφανειών όμως μπορούν και να προβάλλουν οποιαδήποτε εικόνα. Έτσι, μια εικόνα στην οποία έχει γραφτεί ένα κείμενο, μπορεί να χρησιμεύσει σαν υφή (texture) ενός αντικειμένου, κι έτσι το αντικείμενο να φέρει μια επιγραφή.

Για να μπορέσει ένα αντικείμενο να φέρει μια πρωτότυπη υφή, αυτή θα πρέπει να φορτωθεί (upload) στον Second Life Viewer. Ο χρήστης έχει την επιλογή να χρησιμοποιήσει την υφή δωρεάν αλλά για περιορισμένο χρόνο (μέχρι να κλείσει ο viewer) ή να πληρώσει το ποσό των \$L10 ώστε να αποθηκευτούν μόνιμα.

Τα αντικείμενα μπορούν να αλλάζουν textures με την εντολή `llSetTexture(string texture, integer face)`. Η μεταβλητή κειμένου texture περιέχει το όνομα της υφής που θα χρησιμοποιηθεί. Σημαντικό είναι εδώ να τονιστεί πως για να χρησιμοποιηθεί το όνομα του texture, αυτό θα πρέπει να αποθηκευτεί

στα περιεχόμενα (Contents) του αντικειμένου. Η μεταβλητή face, όπως και στην αλλαγή χρώματος, ορίζει ποιες πλευρές του αντικειμένου θα φέρουν το καινούργιο texture[21]. Τέλος, η συγκεκριμένη εντολή προκαλεί παύση του προγράμματος για 0,2 δευτερόλεπτα (sleep) [21].

### 2.3.3 ΕΠΙΠΛΕΟΝ ΚΕΙΜΕΝΟ (FLOATING TEXT)

Σε κάθε αντικείμενο του Second Life υπάρχει η δυνατότητα να τοποθετηθεί, σε σχέση με αυτό, κείμενο το οποίο «επιπλέει» πάνω από το αντικείμενο. Αυτό το κείμενο μπορεί να προσφέρει στους χρήστες οδηγίες ή πληροφορίες, χωρίς αυτοί να χρειάζεται να αλληλοεπιδράσουν με το αντικείμενο ακόμα. Με αυτόν τον τρόπο το επιπλέον κείμενο μπορεί να χρησιμοποιηθεί ως εναρκτήριο κείμενο που προτρέπει τους χρήστες να πατήσουν το κουμπί εκκίνησης ή να ξεκινήσουν ένα παιχνίδι.

Floating text μπορεί να προστεθεί σε ένα αντικείμενο με την εντολή llSetText (string text, vector color, float alpha) [22]. Η μεταβλητή text περιέχει το κείμενο που θα προβληθεί με μέγιστο μέγεθος τα 254 byte, το διάνυσμα color το χρώμα το οποίο θα φέρει το κείμενο ενώ η μεταβλητή alpha ορίζει τον βαθμό διαφάνεια του κειμένου. Αν η τιμή της μεταβλητής είναι 0.0 τότε το κείμενο είναι πλήρως διαφανές ενώ αν είναι 1.0 τότε είναι πλήρως αδιαφανές[22].

Το κείμενο μπορεί να τοποθετηθεί σε διάφορα ύψη χρησιμοποιώντας επανειλημμένα τον χαρακτήρα «\n» πριν το κείμενο για χαμηλότερο ύψος ή μετά για υψηλότερο.

## 2.4 ΔΙΑΔΙΚΤΥΑΚΟΣ ΕΞΥΠΗΡΕΤΗΤΗΣ (WEB SERVER)

Στη συγκεκριμένη εργασία γίνεται χρήση διαδικτυακού εξυπηρετητή. Παρόλα αυτά, εξαιτίας του μεγάλου κόστους συντήρησης ενός server, επιλέχθηκε η φιλοξενία ιστοσελίδας (web hosting) ώστε να μπορούν να αναρτηθούν κυρίως 2 σελίδες στο διαδίκτυο. Οπότε ως Web Server θα θεωρείται ο Host που φιλοξενεί τις σελίδες που παράχθηκαν σε αυτήν την εργασία.

Έτσι, επιλέχθηκε το δωρεάν πακέτο φιλοξενίας της ιστοσελίδας 000webhost.com καθώς αυτό κάλυπτε και τον διαθέσιμο χώρο που απαιτούνταν, ο οποίος ήταν και ελάχιστος (11KB), και επέτρεπε το φόρτωμα αρχείων (upload) όχι μόνο μέσω FTP σύνδεσης μέσω Filezilla αλλά διέθετε εργαλείο διαχείρισης του διαθέσιμου χώρου, πράγμα που διευκόλυνε τις δοκιμές και τις επιλύσεις των προβλημάτων (debugging).

Στον web host, 2 είναι τα κυρίως αρχεία που χρειάστηκαν να ανεβούν ενώ δημιουργείται αυτόματα και ένα τρίτο.



Το ένα αρχείο είναι αρχείο HTML το οποίο κάνει χρήση κομματιών κώδικα CSS, JavaScript ενώ χρησιμοποιεί επίσης τεχνικές AJAX και iframe.

Το δεύτερο αρχείο είναι ένα απλό αρχείο PHP το οποίο δημιουργεί κιόλας το τρίτο αρχείο που είναι ένα απλό αρχείο κειμένου text.

Στο σύνολο τους, τα αρχεία αυτά επικοινωνούν με την πλατφόρμα του Second Life χάρη στην HTTP επικοινωνία η οποία διασφαλίζει την μεταφορά δεδομένων και το φόρτωμα των ιστοσελίδων.

#### 2.4.1 HTML ΣΕΛΙΔΑ

Η HTML σελίδα αποτελεί το κεντρικό σημείο διαχείρισης τόσο των δεδομένων της εργασίας όσο και του περιβάλλοντος που αλληλοεπιδρά ο χρήστης. Χρησιμοποιεί HTML γλώσσα ως τον σκελετό, τη κύρια δομή της σελίδας και μέσω αυτής αλληλοεπιδρά ο χρήστης αλλά και οι υπόλοιπες γλώσσες που εμπεριέχονται σε αυτήν.

Η HTML σελίδα έχει κατάληξη .html ενώ το μέγεθος της αποτελεί 10 KiloBytes(KB). Κάθε φορά που η διεύθυνση της σελίδας φορτώνεται σε κάποιον περιηγητή (φυλλομετρητή) τότε το σύνολο της σελίδας φορτώνεται μέσω HTTP, συμπεριλαμβανομένων και των κομματιών κώδικα CSS και JavaScript. Τα συγκεκριμένα κομμάτια κώδικα θα μπορούσαν να αποτελούν και ξεχωριστά αρχεία χωρίς αυτό να επηρεάζει σε μεγάλο βαθμό τη λειτουργικότητα.

Αξίζει να τονιστεί πως αν στη σελίδα HTML εμπεριεχόταν και το αρχείο php, τότε η κατάληξη της σελίδας θα ήταν .php και οι ιδιότητες τις σελίδας θα ήταν αλλαγμένες.

##### 2.4.1.1 ΓΛΩΣΣΑ HTML

Η γλώσσα HTML (HyperText Markup Language) αποτελεί την κύρια γλώσσα σήμανσης για ιστοσελίδες ενώ αποτελεί το δομικό υλικό κάθε σελίδας στο διαδίκτυο. Η HTML έχει αρκετά απλή δομή, ενώ το σύνολο των εντολών της εκτελείται και φορτώνεται ακαριαία και όχι γραμμή-γραμμή. Αυτό οφείλεται στη χρήση ετικετών οι οποίες τοποθετούνται μέσα στον κώδικα εκεί που θα εμφανίζονται κιόλας στον περιηγητή.

Τέτοιες ετικέτες ορίζουν τα κύρια μέρη μιας HTML σελίδας όπως είναι η αρχή τη σελίδας <html>, η κεφαλίδα <head> και το σώμα <body>. Εντός αυτών μπορούν να οριστούν τίτλοι <title>, επικεφαλίδες <h1> και παράγραφοι <p>.

Η σήμανση στη γλώσσα HTML σκοπεύει κυρίως στην εμφάνιση κειμένου και στους διαφορετικούς τρόπους εμφάνισης του.

Η δομή μιας σελίδας HTML μπορεί να εκτελέσει πολύ πιο πολύπλοκες λειτουργίες μόνο με τη χρήση CSS για τα γραφικά και JavaScript για τις συναρτήσεις.

#### 2.4.1.2 CSS ΓΡΑΦΙΚΑ

Η γλώσσα CSS (Cascading Style Sheets) είναι γλώσσα υπολογιστή που χρησιμοποιείται για τη ρύθμιση της εμφάνισης ενός εγγράφου. Χρησιμοποιείται σε συνδυασμό με αρχεία HTML (και XHTML) κι έτσι αποτελεί την κύρια πηγή γραφικών και γενικότερα της εικόνας μιας ιστοσελίδας. Στις σύγχρονες ιστοσελίδες συνηθώς χρησιμοποιείται αποκλειστικά μόνο η CSS για την εμφάνιση της καθώς τα εργαλεία εμφάνισης που προσφέρει η HTML είναι πενιχρά και ξεπερασμένα σε σχέση με της CSS. Κώδικας με γλώσσα CSS μπορεί να είναι αποθηκευμένος σε ξεχωριστό αρχείο από τη σελίδα HTML, με κατάληξη .css και στην HTML σελίδα θα πρέπει να καλείται ονομαστικά. Αλλιώς μπορεί να είναι εμφωλευμένος στον κώδικα της HTML, χρησιμοποιώντας τις ετικέτες <style> και </style> ώστε να διαχωριστεί ο κώδικας της HTML με τον κώδικα CSS.

Με τη γλώσσα CSS μπορούν να παραμετροποιηθούν πολλαπλές ιδιότητες αντικειμένων, κουμπιών και κειμένων της HTML. Μερικές από αυτές τις παραμετροποιήσεις περιλαμβάνουν το χρώμα, το μέγεθος, τη διαφάνεια, τις σκιές και τη γραμματοσειρά διαφορετικών περιπτώσεων σε μια ιστοσελίδα. Ακόμη, μπορούν να δημιουργηθούν κινούμενα γραφικά (animations), όπως στον κώδικα της Εικόνας 2.3, με τη βοήθεια JavaScript κώδικα.

```
document.getElementById('button1').style.transform = 'translateY(-50%)';
document.getElementById('button1').style.visibility = 'hidden';
setTimeout(function() {document.getElementById('button2').style.transform = 'translateY(40%)';}, 300);
setTimeout(function() {document.getElementById('button2').style.visibility = 'visible';}, 300);
```

Εικόνα 2.3.Κώδικας JavaScript που χρησιμοποιεί CSS για κουμπί με animation.

Στον κώδικα της Εικόνας 2.3 χρησιμοποιείται το πεδίο style ώστε να μπορούν να διαπεραστούν συναρτήσεις CSS. Έτσι, το κουμπί με όνομα «button1» μεταλλάσσεται με τη με τη χρήση της εντολής style.transform στην οποία ορίζεται η τιμή «translateY(-50%)». Η τιμή αυτή ορίζει πως στον άξονα Y (κατακόρυφος άξονας) θα υπάρξει μετακίνηση του αντικειμένου «button1» κατά -50%(μετακίνηση προς τα πάνω) απόσταση του τωρινού μεγέθους του. Η ακριβώς επόμενη γραμμή ορίζει πως η εμφάνιση του συγκεκριμένου αντικειμένου θα γίνει κρυφή, δηλαδή «hidden». Στη συνέχεια, παρόμοια διαχείριση επικρατεί και στο αντικείμενο με όνομα «button2», μόνο που εκεί η μετακίνηση

προκύπτει προς τα πάνω (+40%), το κουμπί γίνεται ορατό, ενώ όλα αυτά συμβαίνουν εντός της συνάρτησης `setTimeout` της JavaScript που ορίζει χρονικές καθυστερήσεις σε εκτελέσεις.

#### 2.4.1.3 ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΜΕ JAVASCRIPT

Η γλώσσα JavaScript αποτελεί γλώσσα προγραμματισμού που χρησιμοποιείται κυρίως για την ασύγχρονη ανταλλαγή δεδομένων μεταξύ του χρήστη και του client. Έχει τη δυνατότητα να αλλάζει δυναμικά την όψη μιας ιστοσελίδας ενώ επιτρέπει κίολας δημιουργία σεναρίων εντός της ιστοσελίδας, κάτι το οποίο δεν επιτρέπει η γλώσσα HTML.

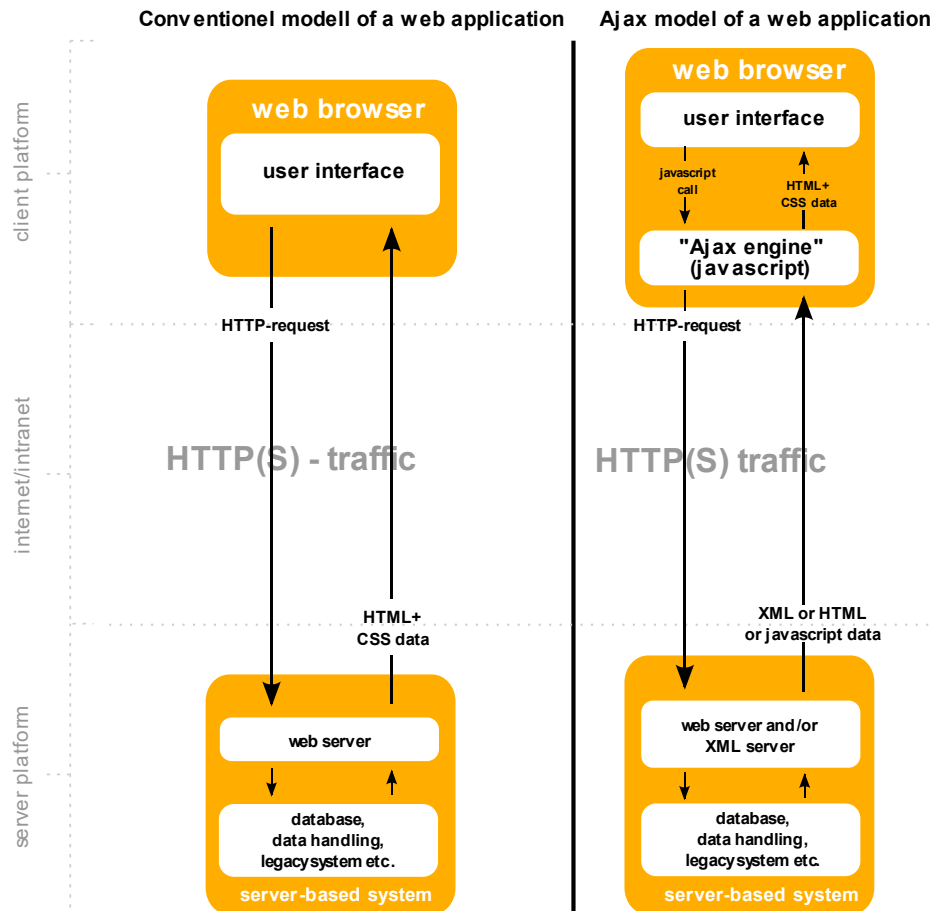
Η συγκεκριμένη γλώσσα είναι βασισμένη σε διαφορετικά προγραμματιστικά παραδείγματα υποστηρίζοντας αντικειμενοστραφές, προστακτικό και συναρτησιακό προγραμματισμό. Παρόλο που το όνομα της γλώσσας φέρει τη λέξη Java, στην πραγματικότητα έχει μεγαλύτερη σχέση με τη γλώσσα C παρά με τη γλώσσα Java καθώς η σημασιολογία τους είναι διαφορετική.

Κομμάτια κώδικα JavaScript μπορούν να εισαχθούν εμφωλευμένα σε μια σελίδα html με τη χρήση της ετικέτας `<script>` για την αρχή του κώδικα JavaScript και την ετικέτα `</script>` για το τέλος της. Μέσω της συγκεκριμένης γλώσσας υλοποιούνται και οι τεχνικές AJAX. Τα κομμάτια κώδικα εκτελούνται εντός του Web Browser (Web Client) καθώς φορτώνεται η ιστοσελίδα από εξειδικευμένο διερμηνέα εντός του Web Browser[23].

#### 2.4.1.4 AJAX

Η AJAX (Asynchronous Javascript And XML) αποτελεί ένα σύνολο προγραμματιστικών τεχνικών που χρησιμοποιούνται στις ιστοσελίδες κυρίως από την πλευρά του πελάτη. Με τη χρήση AJAX μπορεί να επιτευχθεί μεταφορά δεδομένων από και προς μια ιστοσελίδα, χωρίς αυτή να χρειάζεται να ανανεωθεί[24]. Έτσι, υπάρχει δυναμική ανανέωση του περιεχόμενου χωρίς ο χρήστης να βιώνει καθυστερήσεις και ανανεώσεις της ιστοσελίδας. Όμως με τη χρήση AJAX μπορούν να διαπεραστούν δεδομένα μόνο σε θέσεις του ίδιου server που βρίσκεται και η σελίδα HTML (Same origin policy)[25]. Η πύλη (port) που χρησιμοποιείται για την HTTP επικοινωνία είναι η 80 (default) και δεν υπάρχει δυνατότητα να αλλάξει [25].

Ο συνηθέστερος τρόπος χρησιμοποίησης AJAX γίνεται με την συνάρτηση `XMLHttpRequest` η οποία αποτελεί αντικείμενο της JavaScript.



Εικόνα 2.4. Τρόπος επικοινωνίας client-server χωρίς AJAX και με AJAX

#### 2.4.2 PHP ΣΕΛΙΔΑ

Η PHP σελίδα χρησιμοποιεί τη γλώσσα προγραμματισμού PHP (Hypertext Preprocessor) η οποία αποτελεί εργαλείο για δημιουργία ιστοσελίδων με δυναμικά περιεχόμενα. Έτσι, σε αντίθεση με CSS και JavaScript, PHP αρχεία μπορούν να εκτελούνται χωρίς απαραίτητα να αναφέρονται σε κάποια σελίδα HTML. Με αυτόν τον τρόπο, αρχεία PHP σε έναν Server μπορούν να επικοινωνούν μεταξύ τους ή ακόμα και να εγγράφουν και να διαβάζουν άλλου είδους αρχεία (πχ TXT αρχεία, HTML σελίδες) χωρίς απαραίτητα να εμφανίζεται κάποια έξοδος τους στη σελίδα HTML.

#### 2.4.3 HTTP ΕΠΙΚΟΙΝΩΝΙΑ

Το HTTP (Hypertext Transfer Protocol) αποτελεί πρωτόκολλο επικοινωνίας υπερκειμένων (hypertext). Ως υπερκείμενο ορίζεται ένα κείμενο το οποίο δεν διέπεται γραμμικά αλλά οι πληροφορίες του μπορούν να διαπεραστούν και με άλλους τρόπους επιτρέποντας την ελεύθερη πλοήγηση του χρήστη σε αυτό[25].

Η HTTP επικοινωνία λειτουργεί με τρόπο αίτησης-απάντησης (request-response) ανάμεσα στον πελάτη και τον εξυπηρετητή (client-server). Ένας client μπορεί να είναι ένας απλός Web Browser ενώ server μπορεί να είναι ο υπολογιστής που κάνει host μια ιστοσελίδα. Ο client πραγματοποιεί αίτηση (request) στον server ο οποίος του απαντάει (response) με μήνυμα. Στο μήνυμα μπορεί να περιέχεται ο κωδικός κατάστασης της HTTP επικοινωνίας καθώς και τα δεδομένα που ζήτησε ο client, τα οποία βρίσκονται στο σώμα (body) του μηνύματος.

Για να μπορέσει να πραγματοποιηθεί HTTP επικοινωνία, ένας client ξεκινάει μια αίτηση ορίζοντας μια TCP (Transmission Control Protocol) σύνδεση σε μια συγκεκριμένη πύλη (port) του Server. Έτσι, ένας HTTP server «ακούει» σε εκείνη την πύλη για τις αιτήσεις του client. Μια απάντηση του server έχει τη μορφή «HTTP/1.1 200 OK» και επιπλέον ένα μήνυμα στο οποίο το σώμα μπορεί να εμπεριέχονται τα δεδομένα που ζήτησε ο client.

Το πεδίο «200 OK» αποτελεί τους κωδικούς κατάστασης μιας HTTP επικοινωνίας υποδηλώνοντας αν η αίτηση ή η απάντηση είναι πετυχημένη ή όχι.

Οι κωδικοί κατάστασης χωρίζονται ως εξής:

- Πληροφορία (Informational) 1XX
- Επιτυχής (Successful) 2XX
- Ανακατεύθυνση (Redirection) 3XX
- Σφάλμα Πελάτη (Client Error) 4XX
- Σφάλμα Εξυπηρετητή (Server Error) 5XX

Στο πρωτόκολλο HTTP/1.1 αμφίδρομη διαρκής επικοινωνία είναι δυνατή σε αντίθεση με τις παλαιότερες εκδόσεις του πρωτοκόλλου στις οποίες η σύνδεση μεταξύ client και server χανόταν μετά από μόνο μία επικοινωνία αίτησης-απάντησης.

Για την πραγματοποίηση αίτησης (request) σε μια επικοινωνία HTTP υπάρχουν πολλοί διαφορετικοί μέθοδοι.

Τέτοιοι είναι οι:

- GET
- HEAD
- POST
- PUT
- DELETE

- TRACE
- OPTIONS
- CONNECT
- PATCH

Οι κυριότερες μέθοδοι είναι οι GET και POST.

Με τη μέθοδο GET αιτούνται συγκεκριμένα δεδομένα και δε συμβαίνουν άλλες παρενέργειες.

Με τη μέθοδο POST αιτείται από τον Server να αποδεχτεί τα δεδομένα τα οποία αποστέλλονται με την αίτηση.

Στη συγκεκριμένη εργασία λαμβάνουν χώρα 3 συνδέσεις HTTP επικοινωνίας οι οποίες έχουν διαφορετική φορά. Μια HTTP σύνδεση με client τον Web Server και server τον Second Life Server στην οποία μεταφέρονται δεδομένα στο Second Life μέσω της πύλης 12046.

Μία HTTP σύνδεση με client τον Second Life Viewer και Server τον Web Server στον οποίο μεταφέρονται δεδομένα από το Second Life στον Web Server μέσω της πύλης 80, που είναι και η προκαθορισμένη πύλη για την HTTP επικοινωνία.

Και τέλος μία HTTP σύνδεση μέσω AJAX όπου client είναι η σελίδα HTML και server ο ίδιος ο Web Server. Και σε αυτήν την HTTP σύνδεση, η πύλη που χρησιμοποιείται είναι η προκαθορισμένη 80.

## 2.5 ΣΥΝΟΨΗ ΚΕΦΑΛΑΙΟΥ

Το Second Life αποτελεί ένα μοναδικό μέσο κοινωνικής δικτύωσης το οποίο χαρίζει στους χρήστες του, πέρα από την κοινωνικοποίηση τους, και την ευκαιρία να δημιουργήσουν ευφάνταστα έργα αλλά και να εισαχθούν στον προγραμματισμό. Παράλληλα, η διαρκής στήριξη που παρέχεται ανά τα χρόνια σε αυτόν τον εικονικό κόσμο, τόσο από τα μέλη του και την κοινότητα των παικτών, όσο και από την ιδιοκτήτη εταιρεία, έχουν μετατρέψει σε αυτά τα 16 χρόνια λειτουργίας του το Second Life σε μία πλήρη δεύτερη ζωή στην οποία ο καθένας μπορεί να ασχοληθεί με ότι σκεφτεί.

Στη συγκεκριμένη εργασία όμως, το Second Life καταφέρνει και επικοινωνεί με το διαδίκτυο. Στην άλλη άκρη της επικοινωνίας βρίσκεται ένας Web Server που δημιουργήθηκε για τις ανάγκες τις εργασίας και που περιέχει στοιχεία απαραίτητα για την επεξεργασία, αποστολή και λήψη δεδομένων στο Second Life.

Αυτά τα στοιχεία περιλαμβάνουν σελίδες HTML και PHP οι οποίες κάνουν χρήση γλωσσών προγραμματισμού ως εργαλεία, όπως είναι η CSS και η JavaScript ενώ χειρίζονται και πακέτα προγραμματιστικών τεχνικών όπως είναι η AJAX.

Τέλος, η επικοινωνία ώστε να επιτυγχάνεται η μεταφορά των δεδομένων υλοποιείται με συνδέσεις HTTP με τις οποίες αιτούνται και απαντούν πελάτες και εξυπηρετητές τόσο στο Second Life όσο και στον Web Browser.

## 2.6 ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1]. Second Life Downloads, <https://secondlife.com/support/downloads/>, 2018.
- [2]. Second Life Wiki, [http://wiki.secondlife.com/wiki/Third\\_Party\\_Viewer\\_Directory](http://wiki.secondlife.com/wiki/Third_Party_Viewer_Directory), 2018.
- [3]. El. Wikipedia, Second Life, [https://el.wikipedia.org/wiki/Second\\_Life](https://el.wikipedia.org/wiki/Second_Life), 2018.
- [4]. If Second Life Isn't A Game What It Is,  
[http://www.nbcnews.com/id/17538999/ns/technology\\_and\\_science-games/t/if-second-life-isnt-game-what-it/#.U9uyeEi49yw](http://www.nbcnews.com/id/17538999/ns/technology_and_science-games/t/if-second-life-isnt-game-what-it/#.U9uyeEi49yw), 2007.
- [5]. En. Wikipedia, Second Life, [https://en.wikipedia.org/wiki/Second\\_Life](https://en.wikipedia.org/wiki/Second_Life), 2018.
- [6]. Au, Wagner James. The Making of Second Life, pg. 19. New York: Collins.
- [Εικόνα 2.1]. Linden Lab, Infographic 10 years of Second Life,  
<https://www.lindenlab.com/releases/infographic-10-years-of-second-life>.
- [7]. Web Archives, Business Week,  
[https://web.archive.org/web/20070105191002/http://www.businessweek.com/magazine/content/06\\_18/b3982001.htm](https://web.archive.org/web/20070105191002/http://www.businessweek.com/magazine/content/06_18/b3982001.htm), 2001
- [8]. Second Life, Active Users tool, <http://secondlife.com/xmlhttp/secondlife.php>.
- [9]. Currency Rate, Linden Dollar to USD, <https://ld.currencyrate.today/usd>.
- [10]. Linden Lab, Infographic 10 years of Second Life,  
<https://www.lindenlab.com/releases/infographic-10-years-of-second-life>.

- [11]. Web Archives, Ailin Graef,  
[https://web.archive.org/web/20061212224408/http://money.cnn.com/magazines/business2/business2\\_archive/2005/12/01/8364581/index.htm?cnn=yes](https://web.archive.org/web/20061212224408/http://money.cnn.com/magazines/business2/business2_archive/2005/12/01/8364581/index.htm?cnn=yes), 2005.
- [12]. New World Notes, <https://nwn.blogs.com/nwn/2009/03/million.html>, 2009.
- [13]. Second Life, Buy Linden Dollars, <https://secondlife.com/my/lindex/buy.php?lang=en-US>.
- [14]. Second Life Wiki, LSL, [http://wiki.secondlife.com/wiki/Getting\\_started\\_with\\_LSL](http://wiki.secondlife.com/wiki/Getting_started_with_LSL).
- [15]. Web Archives, Open GL Software,  
<https://web.archive.org/web/20041031094901/http://www.sgi.com/products/software/opengl/overview.html>, 2004.
- [16]. Second Life Wiki, Key, <http://wiki.secondlife.com/wiki/Key>.
- [17]. En. Wikipedia, Isilon Clusters, <https://en.wikipedia.org/wiki/Dell EMC Isilon>.
- [18]. Second Life Wiki, Server,  
[http://wiki.secondlife.com/wiki/Release\\_Notes/Second\\_Life\\_Server/11#11.11.09.244706\\_2](http://wiki.secondlife.com/wiki/Release_Notes/Second_Life_Server/11#11.11.09.244706_2).
- [19]. Information Week, Open Doors to Innovation,  
<https://www.informationweek.com/open-doors-to-innovation/d/d-id/1036952>, 2005.
- [20]. Second Life Wiki, llSetColor, <http://wiki.secondlife.com/wiki/llSetColor>.
- [21]. Second Life Wiki, llSetTexture, <http://wiki.secondlife.com/wiki/llSetTexture>.
- [22]. Second Life Wiki, llSetText, <http://wiki.secondlife.com/wiki/llSetText>.
- [23]. HowStuffWorks, JavaScript, <https://computer.howstuffworks.com/javascript.htm>.
- [EIKONA 2.4]. Wikimedia Commons, Daniel S. Haischt,  
<https://commons.wikimedia.org/wiki/File%3AAjax-vergleich.svg>.
- [24]. El. Wikipedia, AJAX, [https://el.wikipedia.org/wiki/Ajax\\_\(προγραμματισμός\)](https://el.wikipedia.org/wiki/Ajax_(προγραμματισμός)).
- [25]. Developer Mozilla, Same Origin Policy,  
[https://developer.mozilla.org/en-US/docs/Web/Security/Same-origin\\_policy](https://developer.mozilla.org/en-US/docs/Web/Security/Same-origin_policy), 2019.



- [26]. Web Archives, Όταν το κείμενο εκδίδεται την τηλεόραση..., <https://web.archive.org/web/20070927212329/http://www.archive.gr/modules.php?name=News&file=article&sid=125&mode=&order=0&thold=0>, 2007.

## ΚΕΦΑΛΑΙΟ 3 Οδηγίες Χρήσεις πακέτου Simon Says

### 3.1 ΕΙΣΑΓΩΓΗ

Το παρόν κεφάλαιο αποτελεί εγχειρίδιο χρήσης για το Project SLSimonSays. Το Project αυτό υλοποιεί το κλασικό παιχνίδι Simon Says σε περιβάλλον Second Life παρέχοντας ταυτόχρονα επιλογές επικοινωνίας διπλής ροής χρησιμοποιώντας HTTP επικοινωνία μέσω οποιουδήποτε Web Browser. Θα πρέπει να τονιστεί επίσης, ότι το συγκεκριμένο Project αποτελεί όχι μόνο την υλοποίηση του παιχνιδιού Simon Says αλλά και τη μελέτη του Cloud Gaming με χρήση λειτουργικού παιχνιδιού-κελύφους στην πλευρά του αποδέκτη.

Τα αρχεία του Project συμπεριλαμβάνονται στην in-world Second Life υλοποίηση και παρέχονται δωρεάν μέσω του Second Life Marketplace στον παρακάτω σύνδεσμο:

<https://marketplace.secondlife.com/p/Simon-Saysv3/18036281> .

Επίσης τα αρχεία είναι διαθέσιμα σε repository του GitHub, για εκ του μηδενός αποδόμηση και επαναδημιουργία οποιουδήποτε σκέλους του Project, στον εξής σύνδεσμο:

<https://github.com/petros3432/SLSimonSays> .

### 3.2 ΚΑΝΟΝΕΣ ΤΟΥ SIMON SAYS

Οι κανόνες του παιχνιδιού Simon Says είναι απλοί.

- Το παιχνίδι δημιουργεί μια ακολουθία από τυχαία χρώματα (ανάλογα τα διαθέσιμα χρώματα, στην περίπτωση της συγκεκριμένης εργασίας είναι τέσσερα).
- Το παιχνίδι προβάλλει την ακολουθία στον χρήστη αναβοσβήνοντας τα χρωματιστά κουμπιά επιλογής που διαθέτει.
- Ο χρήστης καλείται να εισάγει την ακολουθία των χρωμάτων που παρακολούθησε στη σωστή σειρά.
- Εάν η προσπάθεια του χρήστη είναι σωστή, τότε ένα νέο τυχαίο χρώμα προστίθεται στην πρότερη ακολουθία με τα χρώματα και επαναπροβάλλεται με τον ίδιο τρόπο στον χρήστη.
- Το παιχνίδι τελειώνει όταν ο χρήστης εισάγει κάποιο λάθος χρώμα, που σημαίνει ότι η σωστή ακολουθία των χρωμάτων του διέφυγε.
- Το σκορ του παιχνιδιού προκύπτει από τον μέγιστο αριθμό χρωμάτων που είχε η τελευταία ακολουθία όπου ο χρήστης βρήκε αλάνθαστα.

### 3.3 ΕΓΚΑΤΑΣΤΑΣΗ

Για να καταστεί δυνατό ο χρήστης να παίξει με το συγκεκριμένο Project υπάρχουν ορισμένες απαιτήσεις. Ο χρήστης θα πρέπει να έχει ενεργό λογαριασμό στο Second Life καθώς και κάποιον Viewer του Second Life ώστε να μπορεί να συνδεθεί στον κόσμο του. Ο χρήστης θα πρέπει να έχει πρόσβαση σε In-World γη η οποία επιτρέπει το χτίσιμο (Building) και τον προγραμματισμό (Scripting) καθώς και το άνοιγμα του αρχείου από το Marketplace (Rez). Τα δικαιώματα της εκάστοτε γης και οι απαγορεύσεις της στο Second Life μπορούν να γίνουν εύκολα αντιληπτές είτε μέσω της κεντρικής μπάρας στον Viewer είτε μέσω του μενού World > About This Land.

#### 3.3.1 ΑΠΑΙΤΗΣΕΙΣ ΓΗΣ

Τα επιθυμητά δικαιώματα στο Second Life έχουν σημαντικό ρόλο στην διεκπεραίωση οποιοδήποτε εγχειρημάτων από χρήστες εντός του Second Life καθώς επηρεάζουν σε τι βαθμό μπορεί να φανεί δημιουργικός κανείς εντός του εικονικού κόσμου.

Στον κόσμο που βρίσκεται ο χρήστης είναι επιθυμητό η κεντρική μπάρα να μοιάζει με την Εικόνα 3.1:



Εικόνα 3.1. Μπάρα τοποθεσίας στο Second Life στην οποία φαίνονται οι απαγορεύσεις που ισχύουν.

Στη δεξιά πλευρά φαίνονται οι απαγορεύσεις (στη συγκεκριμένη περίπτωση η μόνη απαγόρευση που υπάρχει είναι να μη μπορεί ο χρήστης να «σκουντήσει» άλλους παίκτες). Μία περιοχή (region) χωρίς τα επιθυμητά δικαιώματα θα είχε μια μπάρα όπως στην Εικόνα 3.2. Επίσης δεξιά, ο κύβος υποδηλώνει την απαγόρευση χτισίματος (building) ενώ το σήμα του εγγράφου υποδηλώνει απαγόρευση προγραμματισμού (scripting).



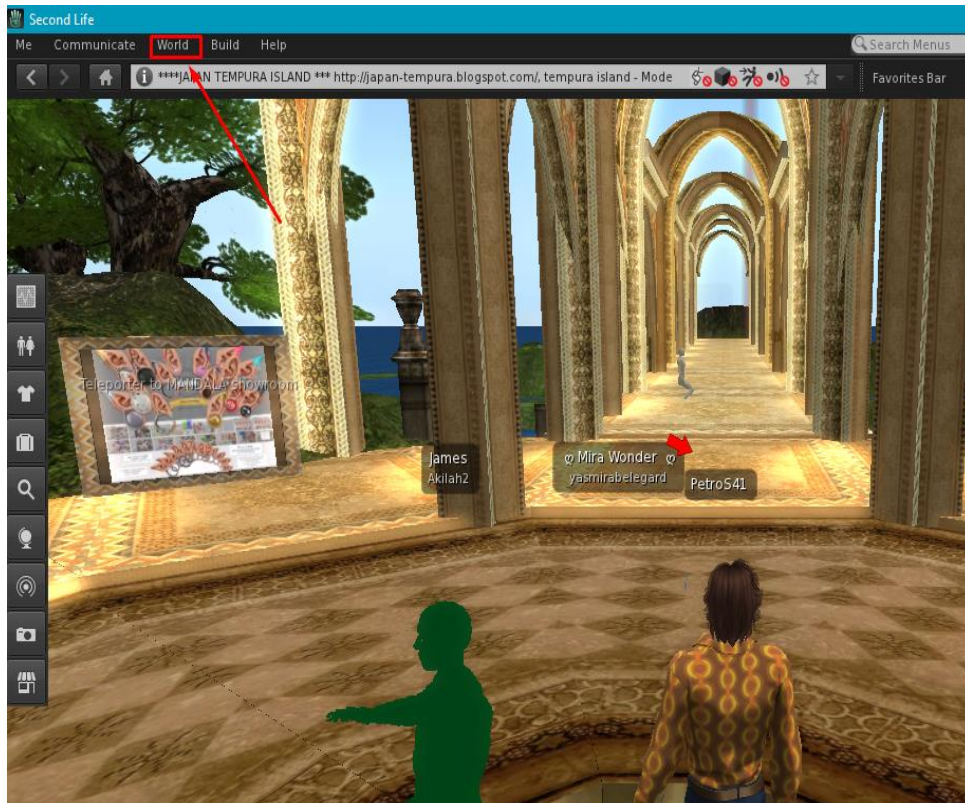
Εικόνα 3.2. Μπάρα τοποθεσίας όπου φαίνονται πολλαπλές απαγορεύσεις building και scripting.

Περιοχή με τα κατάλληλα δικαιώματα μπορεί να αγοραστεί ή να ενοικιαστεί από το Second Life με κανονικά χρήματα (ποσά που ανέρχονται έως και τα εκατοντάδες δολάρια).

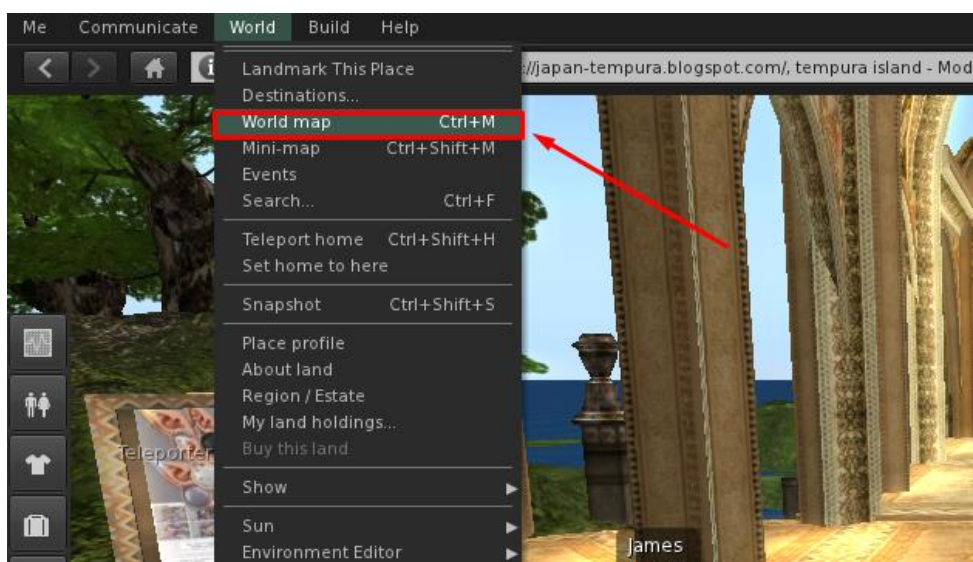
Παρόλα αυτά, υπάρχει η δυνατότητα ο χρήστης να ταξιδέψει και να επισκεφθεί περιοχές οι οποίες, αν και δε βρίσκονται στην ιδιοκτησία του χρήστη, επιτρέπουν τη δημιουργία έργων και δίνουν τη δυνατότητα στους χρήστες να δοκιμάσουν, να παίξουν και να χτίσουν πολλαπλά project. Αξίζει να σημειωθεί ότι αυτοί οι χώροι είναι επιβλεπόμενοι και μερικές φορές διαγράφουν όλα τα αντικείμενα του χώρου ανά μερικές ώρες. Τέτοιες περιοχές ονομάζονται Sandboxes και μια λίστα με διαθέσιμους

τέτοιους προορισμούς μπορεί να βρεθεί εδώ:  
«<https://community.secondlife.com/knowledgebase/english/building-tips-r13/>».

Μια τέτοια περίπτωση είναι η περιοχή Dreams. Για να περιηγηθεί εκεί ο χρήστης, χρειάζεται να ανοιχθεί το παράθυρο World και μετά να επιλεγεί World>World Map ή να πατηθεί εξ αρχής Ctrl+M.



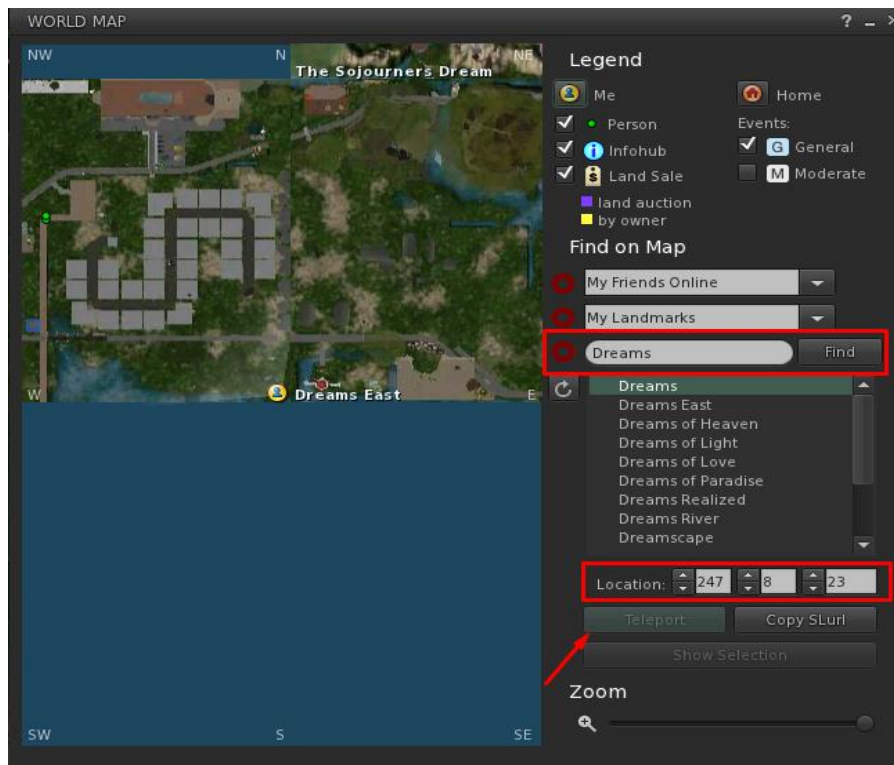
Εικόνα 3.3. Επιλογή της ενότητας World.



Εικόνα 3.4. Επιλογή της ενότητας World map (Ctrl+M για συντομία).

Στη συνέχεια, στο παράθυρο που ακολουθεί, ο χρήστης χρειάζεται να εισάγει τον προορισμό Dreams στο πλαίσιο δίπλα στο κουμπί Find, που φαίνεται στην εικόνα ενώ στα πλαίσια Location, να εισάγει τους αριθμούς 247, 8 και 23 αντίστοιχα και να πατήσει το κουμπί Teleport.

Με αυτόν τον τρόπο ο χρήστης τηλεμεταφέρεται στην περιοχή Dreams και μάλιστα σε σημείο του χάρτη που δεν επηρεάζει τις υπόλοιπες κατασκευές άλλων χρηστών που βρίσκονται ήδη εκεί.



Εικόνα 3.5. Εμφάνιση του συνολικού κόσμου του Second Life και τηλεμεταφορά στην τοποθεσία Dreams.

### 3.3.2 ΣΤΟΙΧΕΙΑ ΠΟΥ ΧΡΗΣΙΜΟΠΟΙΟΥΝΤΑΙ

Για την ορθή λειτουργία του παιχνιδιού Simon Says καθώς και για την οπτική αλλά και τη λειτουργική βελτιστοποίηση του, χρησιμοποιήθηκαν αρκετά αρχεία με κώδικα αλλά και εικαστικές προσθήκες, το καθένα με διαφορετική χρήση κι εφαρμογή.

#### 3.3.2.1 SECOND LIFE

Μέσα στον κόσμο του Second Life χρησιμοποιούνται 9 αρχεία κώδικα (ονομάζονται scripts στο Second Life) καθώς και 3 αρχεία εικόνας που χρησιμεύουν ως Texture. Τα αρχεία κώδικα ή scripts είναι γραμμένα στη γλώσσα LSL (Linden Scripting Language), μια γλώσσα προγραμματισμού που έχει ως βάση τη γλώσσα C. Αυτός είναι και ο λόγος που τα αρχεία κώδικα διατίθενται με κατάληξη «.c» και όχι με κάποια άλλη.

Τα αρχεία είναι τα εξής:

- parent\_v8.c
- CONNECTOR\_v4.c
- RED\_v4.c
- GREEN\_v4.c
- BLUE\_v4.c
- YELLOW\_v4.c
- STARTRESET\_v2.c
- HARD\_RESET.c
- INFO\_v1.c
- INFO.png
- START.png
- RESET.png

Όλα τα αρχεία κώδικα επικοινωνούν μεταξύ τους στον κόσμο του Second Life μέσω εντολών Say και Listen, πιο συγκεκριμένα με την εντολή llsay() και τη συνάρτηση συμβάντος (Event Function) listen(). Ανάμεσα σε 2 object που μεταχειρίζονται διαφορετικά script εφαρμόζεται επικοινωνία σε ξεχωριστό channel ώστε να αποφεύγονται οι παρεμβολές, εκτός κι αν αυτές είναι επιθυμητές.

Στον κόσμο του Second Life ο χρήστης έχει τη δυνατότητα να δημιουργεί πολλαπλά αντικείμενα (ανάλογα τα δικαιώματα της εκάστοτε περιοχής του Second Life, όπως προαναφέρθηκε) και σε αυτά να καταχωρεί αρχεία κώδικα αλλά και αρχεία υφής (textures).

Έτσι, σε αυτό το project κάθε αρχείο κώδικα είναι αποθηκευμένο σε ένα ξεχωριστό αντικείμενο (object) του Second Life, ενώ 3 από αυτά τα object διαθέτουν και από ένα αρχείο εικόνας όπου χρησιμεύει σαν texture.

Συγκεκριμένα, χρησιμοποιούνται 4 αντικείμενα-κουμπιά το καθένα με ξεχωριστό χρώμα (κόκκινο, πράσινο, μπλε και κίτρινο) στα οποία ανατίθεται κι από ένα script (RED\_v4.c , GREEN\_v4.c , BLUE\_v4.c και YELLOW\_v4.c αντίστοιχα). Ακόμη, υπάρχει η βασική πλακέτα στην οποία εφαρμόζουν τα 4 προαναφερθέντα κουμπιά εισόδου και προβολής της χρωματικής ακολουθίας. Στη συγκεκριμένη πλακέτα ανατίθεται το script «parent\_v8.c» το οποίο αποτελεί και τον “εγκέφαλο” του παιχνιδιού.

Επιπλέον, σημαντικός είναι ο ρόλος του κουμπιού CONNECTOR το οποίο είναι υπεύθυνο για τη σύνδεση του παιχνιδιού με το διαδίκτυο (αρχείο CONNECTOR\_v4.c) ενώ και τα κουμπιά START και HARD\_RESET δε μπορούν να παραλειφθούν (αρχεία STARTRESET\_v2.c και HARD\_RESET.c αντίστοιχα). Το κουμπί STARTRESET επίσης κάνει χρήση των 2 αρχείων εικόνας (START.png και RESET.png) και τα χρησιμοποιεί ως texture.

Τέλος, το κουμπί INFO που παρέχει πληροφορίες για το παιχνίδι καθώς και οδηγίες, περιλαμβάνει το script «INFO\_v1.c» καθώς και την εικόνα «INFO.png».

### 3.3.2.2 HOST SERVER

Πέρα από τα αρχεία του Second Life το συγκεκριμένο project κάνει χρήση αρχείων διαδικτυακού χαρακτήρα, τα οποία είναι ανεβασμένα σε έναν Host Server και εξυπηρετούν την πολυεπίπεδη μεταφορά των δεδομένων του Second Life.

Τα αρχεία είναι τα εξής:

- siwonsimon.html
- sl.php

Το αρχείο siwonsimon.html είναι γραμμένο σε HTML ενώ περιέχει εισχωρήσεις από CSS, JAVASCRIPT και εφαρμογή AJAX (Asynchronous JavaScript and XML). Το αρχείο «sl.php» είναι εξ 'ολοκλήρου γραμμένο σε γλώσσα PHP ενώ παράγει τοπικά στον Host ένα αρχείο TEST.txt το οποίο μεταβάλλεται δυναμικά από το PHP αρχείο.

### 3.3.2.3 ARDUINO WEB BROWSER

Προκειμένου να είναι δυνατόν να συνδεθεί μια πλακέτα Arduino στον Host και να επικοινωνήσει με το Second Life χρησιμοποιούνται 1 αρχείο Arduino και 1 πρόγραμμα φτιαγμένο στο Windows Visual Studio και έχει ως βάση τη Visual Basic.NET.

Τα αρχεία είναι τα εξής:

- Simon\_Browser.vb
- BUTTONS.ino

### 3.3.3 ΤΡΟΠΟΣ ΥΛΟΠΟΙΗΣΗΣ

Στο σημείο αυτό ο χρήστης μπορεί να επιλέξει τον τρόπο με τον οποίο θα υλοποιήσει τη φυσική υπόσταση στο Second Life του παιχνιδιού SLSimonSays. Για αυτόν τον σκοπό υπάρχουν δύο διαθέσιμοι τρόποι. Ο πρώτος, κι ο πιο απλός, τρόπος χρησιμοποιεί το Second Life Marketplace ώστε να αποκτήσει άμεσα και τα ζητούμενα object που χρησιμοποιούνται στο Second Life και τα scripts που είναι υπεύθυνα για τη λειτουργία του project. Ο δεύτερος τρόπος παρέχει τα λειτουργικά scripts όμως εισάγει τον χρήστη στο εργαλείο Build και του επιτρέπει να σχεδιάσει και να μεταλλάξει την εξωτερική εικόνα του παιχνιδιού στον κόσμο του Second Life.



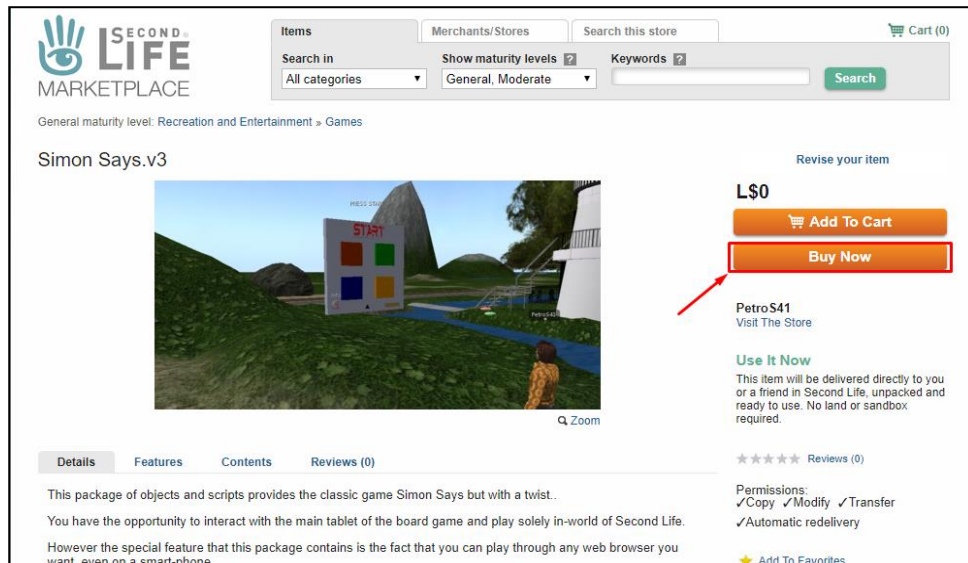
### 3.3.3.1 ΥΛΟΠΟΙΗΣΗ ΜΕ MARKETPLACE

Ο πιο απλός τρόπος είναι να μεταβεί στη σελίδα του Marketplace του Second Life στο οποίο βρίσκεται ως προϊόν το παιχνίδι SLSimonSays.

Η φυσική υπόσταση του παιχνιδιού (objects) και τα αρχεία προγραμματισμού (scripts) διατίθενται δωρεάν (L\$ 0).

Μεταβαίνοντας στο σύνδεσμο :

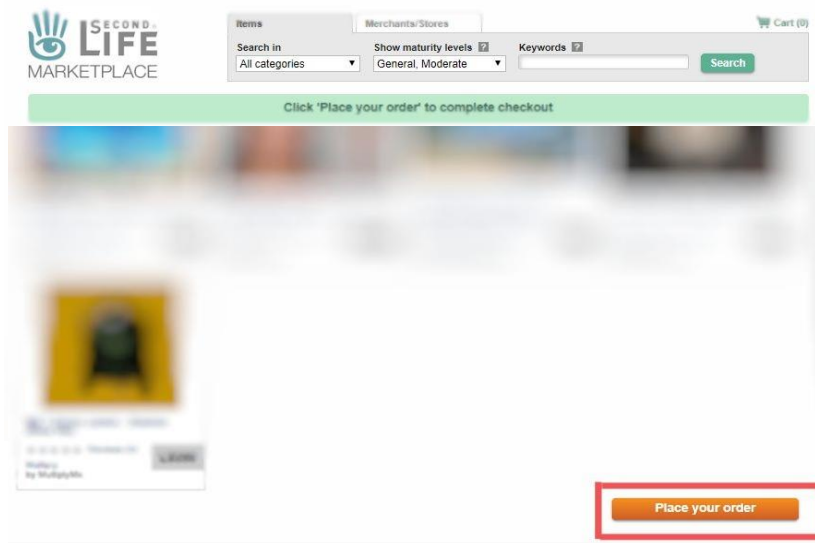
<https://marketplace.secondlife.com/p/Simon-Saysv3/18036281>



Εικόνα 3.6. Η σελίδα του project στο Marketplace του Second Life.

Το κουμπί Buy Now, επιλέγεται.

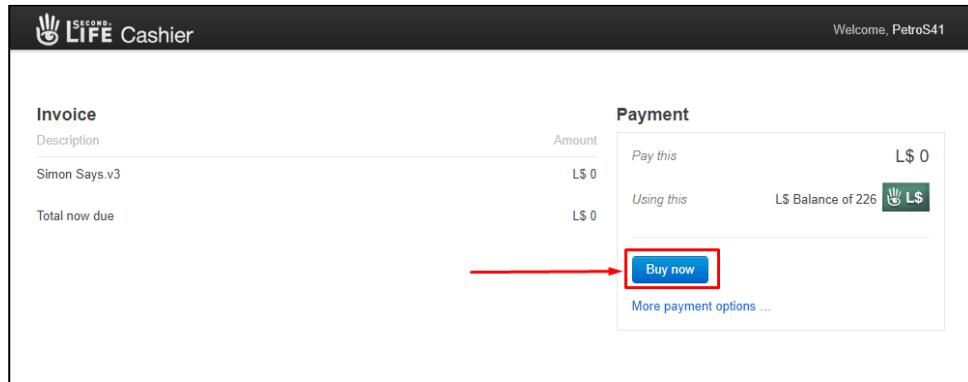
Στη συνέχεια, στην νέα σελίδα, ο ενδιαφερόμενος πρέπει να κατέβει στο τέλος της σελίδας όπου υπάρχει το κουμπί «Place your Order».



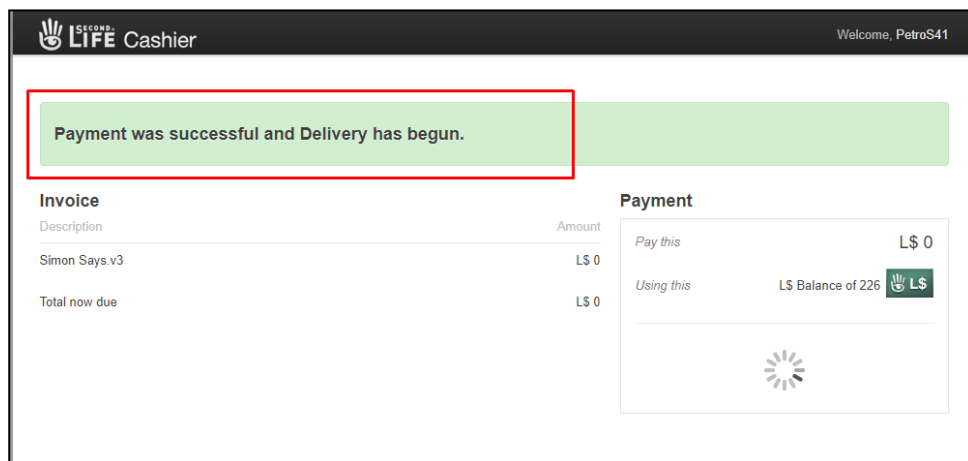
Εικόνα 3.7. Προβολή του κουμπιού για την τοποθέτηση της παραγγελίας.



Έπειτα, επιλέγεται το κουμπί Buy Now ώστε να ολοκληρωθεί η αγορά. Σε αυτήν τη σελίδα υπενθυμίζεται ότι το κόστος της αγοράς είναι μηδενικό, ενώ στη συνέχεια υπάρχει ενημέρωση ότι η αποστολή του πακέτου είναι καθοδόν.



Εικόνα 3.8. Εμφάνιση του κουμπιού "Buy now" για την ολοκλήρωση της αγοράς.



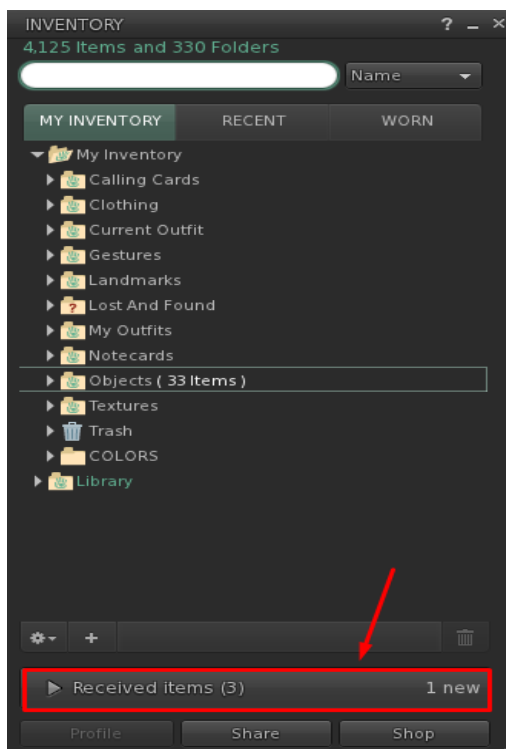
Εικόνα 3.9. Η παραγγελία έχει πραγματοποιηθεί και το project θα παραδοθεί σύντομα.

Στο Second Life Viewer τώρα, κι ενώ ο χρήστης πρέπει να βρίσκεται σε περιοχή του Second Life με τα απαιτούμενα δικαιώματα, όπως αναφέρθηκε και προηγουμένως, μεταβαίνει στο Inventory. Αυτό μπορεί να γίνει από την πλαϊνή βοηθητική μπάρα ή πατώντας Ctrl+I.

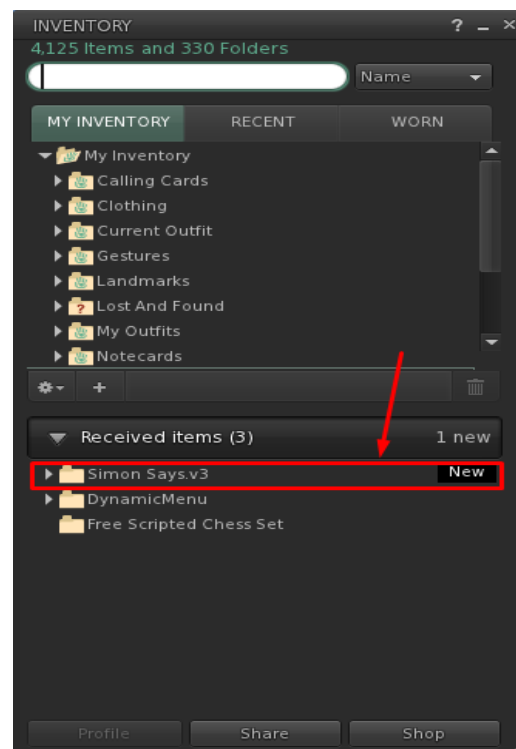


Εικόνα 3.10. Άνοιγμα Inventory μέσα στο Second Life Viewer.

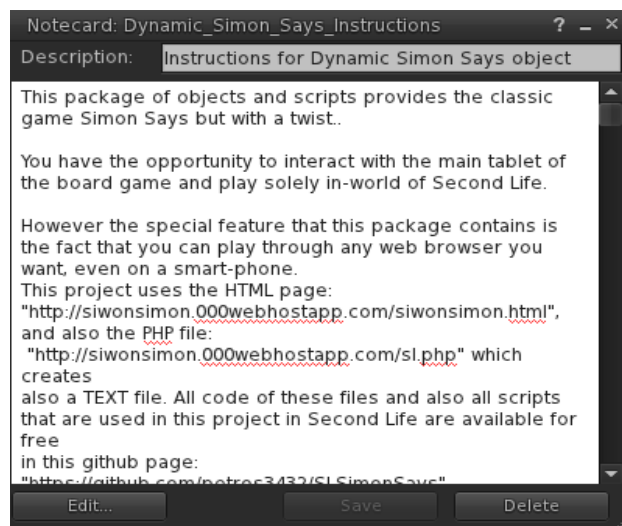
Έτσι, ανοίγει το Inventory όπου μπορεί να βρει την κατηγορία Received Items. Κλικάροντας πάνω στη συγκεκριμένη περιοχή, εμφανίζεται και ο φάκελος που περιέχει το Project SLSimonSays, «Simon Says.v3».



Εικόνα 3.11. Εντοπισμός του νεοαποκτηθέντος Project στο Inventory.



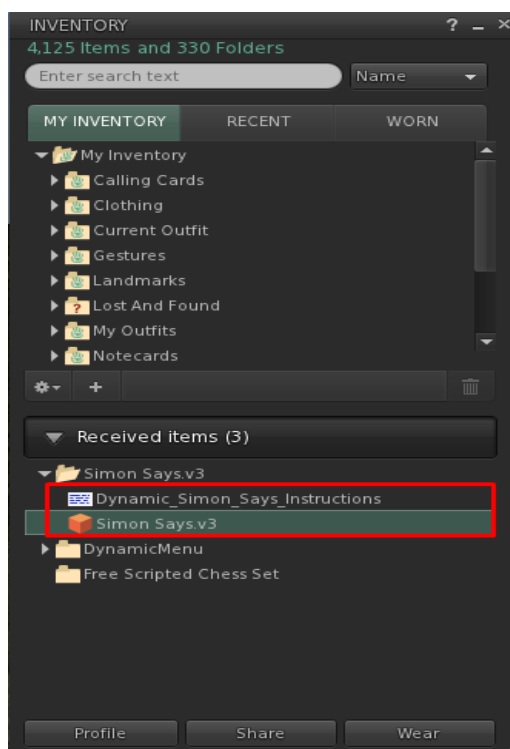
Εικόνα 3.12. Εντοπισμός του νεοαποκτηθέντος Project στο Inventory.



Εικόνα 3.13. Περιεχόμενο των οδηγιών στο Notecard.

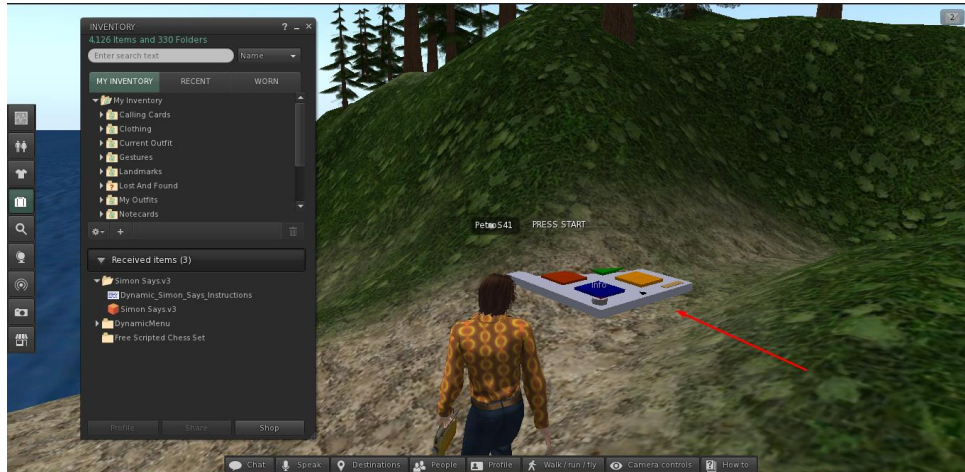
Σε αυτόν τον φάκελο υπάρχει ένα Object ονόματι «Simon Says.v3» και μια notecard με όνομα «Dynamic\_Simon\_Says\_Instructions» στην οποία περιέχονται οδηγίες χρήσης στα Αγγλικά.

Για να χρησιμοποιηθεί το παιχνίδι, ο χρήστης απλά πρέπει να κλικάρει και να σύρει το αρχείο «Simon Says.v3» οπουδήποτε στην οθόνη στο Second Life.



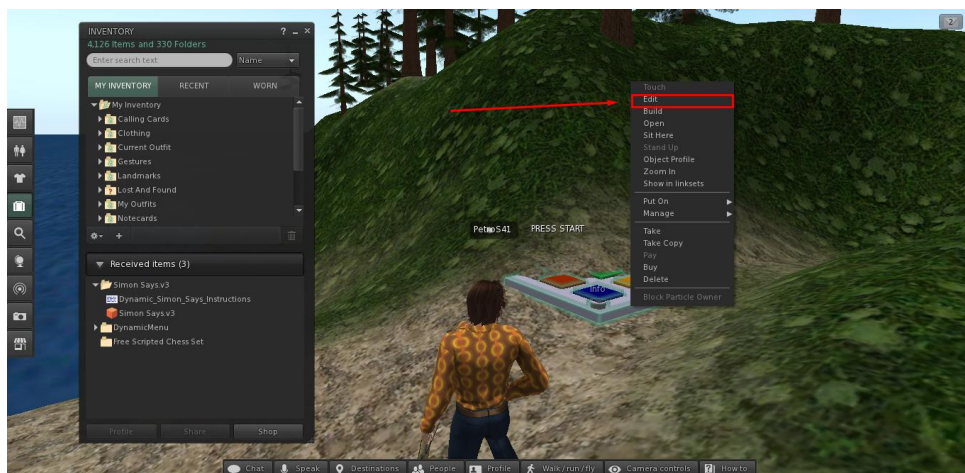
Εικόνα 3.14. Το object της εργασίας στον φάκελο που αποστάλθηκε από το Marketplace.

Σε αυτό το σημείο, υπάρχει μια μικρή πιθανότητα το αντικείμενο να εισαχθεί στον κόσμο, όχι με τον πλέον βέλτιστο τρόπο και να μην είναι δυνατή η χρήση του, όπως για παράδειγμα φαίνεται στην Εικόνα 3.15.



Εικόνα 3.15. Εφαρμογή του αντικειμένου στον κόσμο του Second Life (Rez) σε μη ιδανικό σημείο.

Εδώ, με δεξί κλικ επιλέγεται το αντικείμενο και κλικάρεται η επιλογή Edit, κι έτσι ο χρήστης μεταφέρεται στο εργαλείο του builder απ' όπου μπορεί να μετακινήσει, να περιστρέψει, ακόμα και να μεγεθύνει το αντικείμενο.



Εικόνα 3.16. Παραμετροποίηση του object με Edit.



Για να μετακινήσει το αντικείμενο, απλά πρέπει να επιλέξει έναν από τους τρεις άξονες (ουσιαστικά τα βελιάκια τους) με τα διαφορετικά χρώματα, και με το ποντίκι κι έχοντας πατημένο το αριστερό κλικ μπορεί να μετακινήσει το αντικείμενο προς την κατεύθυνση του επιλεγμένου άξονα.



Εικόνα 3.18. Μετακίνηση της πλακέτας στους άξονες Z και Y.



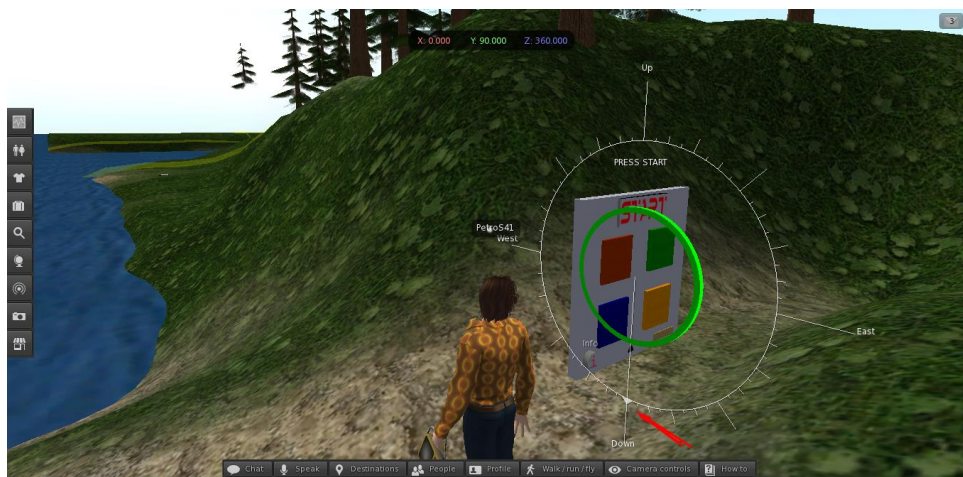
Εικόνα 3.17. Μετακίνηση της πλακέτας στους άξονες Z και Y.

Εφόσον μετακινήσει το αντικείμενο στη θέση που επιθυμεί, και χωρίς να βγει από τη λειτουργία Build, μπορεί να το περιστρέψει. Αυτό γίνεται έχοντας πατημένο το κουμπί CTRL. Με αυτόν τον τρόπο θα εμφανιστούν 3 χρωματιστά δαχτυλίδια γύρω από το αντικείμενο. Με τον ίδιο τρόπο με πριν, κι έχοντας πατημένο το Ctrl, μπορεί να επιλεγθεί με το ποντίκι ένα δαχτυλίδι και μετακινώντας το ποντίκι στην κατεύθυνση του κάθε δαχτυλιδιού, μπορεί να περιστρέφει το αντικείμενο προς κάθε φορά.



Εικόνα 3.19. Περιστροφή της ταμπλέτας γύρω από τον άξονα Y.

Αξίζει σημείωσης το ότι αν μετακινηθεί το ποντίκι, πάντα έχοντας πατημένο το CTRL και το αριστερό κλικ, σε μία από τις λευκές γραμμές, είτε αυτές με τις ονομασίες Up, Down, East, West, είτε αυτές χωρίς όνομα, τότε το αντικείμενο περιστρέφεται σε αυτές τις προκαθορισμένες θέσεις εύκολα. Με αυτόν τον τρόπο, απλά στοχεύοντας στο Up, είναι δυνατόν ο χρήστης να είναι σίγουρος ότι το αντικείμενο θα είναι τέλεια ευθυγραμμισμένα κάθετο στον Second Life κόσμο. Τέτοιου είδους εργαλεία με προσοχή πάντοτε στη λεπτομέρεια καταστούν τη λειτουργία Build του Second Life μια εξαιρετική σχεδιαστική πλατφόρμα την οποία επιλέγουν πολλοί χρήστες μέχρι και σήμερα για τη δημιουργία πολλών εικαστικών και μη, αντικειμένων.



Εικόνα 3.20. Επιλογή του σημείου Down για τέλεια καθετότητα.

Σε αυτό το σημείο το παιχνίδι Simon Says είναι έτοιμο για χρήση.

### 3.3.3.2 ΥΛΟΠΟΙΗΣΗ ΕΚ ΤΟΥ ΜΗΔΕΝΟΣ

Εάν έχει ολοκληρωθεί η υλοποίηση του project μέσω Marketplace, τότε αυτού του είδους η υλοποίηση μπορεί να παραληφθεί.

Σε αυτό το σημείο, για να ολοκληρωθεί η υλοποίηση εκ του μηδενός, απαιτείται η πρόσβαση στα αρχεία script που χρησιμοποιούνται στο Second Life. Ακόμη, αν ο ενδιαφερόμενος επιθυμεί να παραμετροποιήσει ή να συμβουλευτεί τις σελίδες που χρησιμοποιεί η εργασία για τους web-browsers, μπορεί να συμβουλευτεί τα αρχεία siwonsimon.html και sl.php καθώς επίσης και το αρχείο siwonsimon\_debugging.html το οποίο είναι εξοπλισμένο με εργαλεία debugging.

Τα αρχεία στο σύνολο τους μπορούν να βρεθούν στο εξής repository του GitHub: <https://github.com/petros3432/SLSimonSays> ενώ οι σελίδες HTML και η σελίδα PHP μπορούν να διαπεραστούν στα ακόλουθα link: <http://siwonsimon.000webhostapp.com/siwonsimon.html>, [http://siwonsimon.000webhostapp.com/siwonsimon\\_debugging.html](http://siwonsimon.000webhostapp.com/siwonsimon_debugging.html), <http://siwonsimon.000webhostapp.com/sl.php>.

Στο Second Life τώρα, ο χρήστης, εφόσον βρίσκεται σε περιοχή του Second Life με τα κατάλληλα δικαιώματα για building και scripting, μπορεί να αρχίσει τη δημιουργία του παιχνιδιού Simon Says. Αρχικά, χρειάζεται να πατηθεί το δεξί κλικ οπουδήποτε στον κόσμο και στη συνέχεια η επιλογή Build. Ο μηχανισμός Build είναι ένα εύκολο εργαλείο που επιτρέπει τη δημιουργία ποικίλων σχημάτων και μορφών ενώ σε συνεργασία με το scripting, είναι δυνατό να υλοποιηθούν σχεδόν ό,τι μπορεί να φανταστεί ο χρήστης. Οπότε, είναι στην κρίση και στη δημιουργικότητα του χρήστη να μεταλλάξει, να προσθέσει ή και να εφαρμόσει νέα αντικείμενα σε αυτά που συμπεριλαμβάνονται σε αυτό το Project. Όμως, δε θα ήταν συνετό ο χρήστης να μειώσει τα απαιτούμενα αντικείμενα, εκτός κι αν γίνει αντίστοιχη παραμετροποίηση και σύζευξη των script.

Το παιχνίδι Simon Says που υλοποιείται σε αυτό το Project κάνει χρήση 9 αρχείων script τα οποία είναι σε ισάριθμα object. Για τη σωστή υλοποίηση του εγχειρήματος, ο χρήστης θα πρέπει να δημιουργήσει τουλάχιστον 9 διαφορετικά αντικείμενα καθώς αριθμός μικρότερος από αυτόν δεν είναι αποδεκτός.

Για να εισαχθούν script σε ένα object αρκεί ο χρήστης να περιηγηθεί στην κατηγορία Contents και να επιλέξει το κουμπί New Script. Έτσι, αυτόματα, θα δημιουργηθεί ένα προκαθορισμένο αρχείο script, με όνομα «New Script», το οποίο περιέχει το απλό πρόγραμμα εκφώνησης «Hello World» με το άγγιγμα του object. Σε αυτό το αρχείο, ανοίγοντας το με διπλό κλικ, ο χρήστης έχει πρόσβαση στον κώδικα. Οπότε, αρκεί να διαγραφεί το περιεχόμενο του script που εμφάνιζε το «Hello World», το άνοιγμα των αρχείων C με κάποιου είδους εργαλείο επεξεργασίας (π.χ. Notepad++), η αντιγραφή των περιεχομένων των αρχείων C και η επικόλληση τους στο ήδη ανοιγμένο script στο Second Life. Τέλος, χρειάζεται να πατηθεί το κουμπί Save, αν υπάρχουν συντακτικά λάθη θα εμφανιστούν στην κονσόλα και το script είναι έτοιμο για χρήση.

Ο χρήστης, μπορεί να ξεκινήσει με τη δημιουργία 5 αντικειμένων οποιουδήποτε σχήματος και στο κάθε ένα θα εισάγει ένα από τα αρχεία κώδικα RED\_v4.c, GREEN\_v4.c, BLUE\_v4.c, YELLOW\_v4.c και parent\_v8.c. Το κάθε ένα από αυτά τα αρχεία ορίζει το χρώμα του object με την εντολή `llSetColor()`, οπότε αν ο χρήστης επιθυμεί να αλλάξει κάποιο από τα χρώματα των αντικειμένων, μπορεί να το κάνει παραμετροποιώντας τη συγκεκριμένη εντολή. Τα πρώτα 4 αντικείμενα με ονόματα RED, GREEN, BLUE και YELLOW αποτελούν τα χρωματιστά κουμπιά τα οποία ελέγχει το αντικείμενο parent, με το script parent\_v8.c. Οπότε, θεωρείται συνετό τα 4 χρωματιστά κουμπιά να εγκατασταθούν πάνω ή τουλάχιστον κοντά στο object parent.

Ακόμη, είναι υποχρεωτικό να δημιουργηθούν άλλα 3 κουμπιά, CONNECTOR (με script CONNECTOR\_v4.c), STARTRESET (με script STARTRESET\_v1.c και texture START.png και RESET.png) και τέλος ένα κουμπί HARD\_RESET (με script HARD\_RESET.c). Και αυτά τα κουμπιά αποτελούν μέρος του

In-World Second Life Game Interface οπότε είναι επιθυμητό να τοποθετηθούν κοντά ή και πάνω στην κεντρική πλακέτα.

Τέλος, μπορεί τελείως προαιρετικά να τοποθετηθεί ένα επιπλέον κουμπί INFO (με script INFO\_v1.c) το οποίο προσφέρει χρήσιμες πληροφορίες στον χρήστη μέσα στο Second Life. Το κουμπί αυτό δεν επικοινωνεί με κάποιο άλλο κομμάτι του έργου οπότε η απουσία του δε θα επηρεάσει με κάποιον τρόπο τη λειτουργικότητα.

### 3.4 ΛΕΙΤΟΥΡΓΙΑ

Η λειτουργία του Project SLSimonSays βασίζεται στην πολλαπλών επιπέδων επικοινωνία του Second Life με εξωτερικούς Server του διαδικτύου ενώ επιτρέπει και τη ξεχωριστή χρήση στο Second Life.

#### 3.4.1 TO INTERFACE ΣΤΟ SECOND LIFE

Εφόσον η υλοποίηση του project έγινε μέσω του Second Life Marketplace, τότε το παιχνίδι ως σύνολο θα είναι ίδιο με αυτό της Εικόνας 3.21.



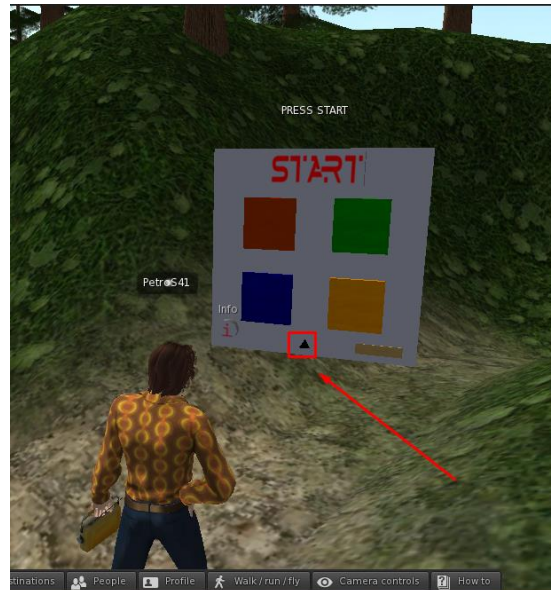
Εικόνα 3.21 To interface του Project "Simon Says.v3".

Εδώ είναι ορατή η κεντρική πλακέτα στην οποία εφαρμόζουν τα κουμπιά χρήσης. Σε αυτό είναι ορατά το START κουμπί (Εικόνα 3.22), το κουμπί CONNECTOR (Εικόνα 3.23), το κουμπί HARD\_RESET (Εικόνα 3.24) και το κουμπί INFO (Εικόνα 3.25). Το κουμπί START έχει διπλή μορφή και εμφανίζει και το κουμπί RESET αν πατηθεί (Εικόνα 3.27).





Εικόνα 3.22. Το κουμπί START.



Εικόνα 3.23. Το κουμπί CONNECTOR.

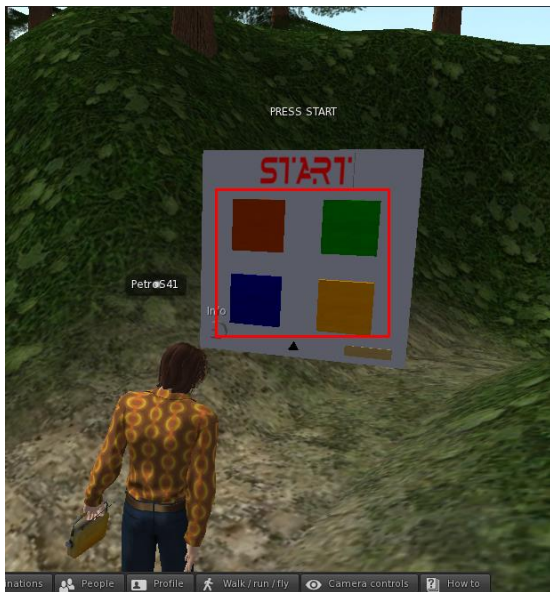
Ακόμη είναι ορατά τα τετράγωνα κουμπιά εισόδου με χρώματα Κόκκινο, Πράσινο, Μπλε και Κίτρινο (Εικόνα 3.26) τα οποία αναβοσβήνουν.



Εικόνα 3.25. Το κουμπί HARD RESET.



Εικόνα 3.24. Το κουμπί INFO.



Εικόνα 3.27. Τα 4 χρωματιστά κουμπιά.



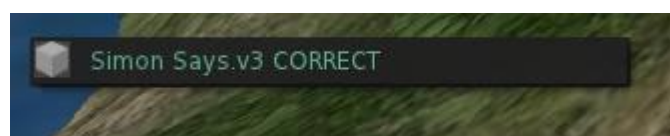
Εικόνα 3.26. Η εναλλαγή του κουμπιού START σε RESET

### 3.4.2 ΧΡΗΣΗ ΑΠΟΚΛΕΙΣΤΙΚΑ ΣΤΟ SECOND LIFE

Εφόσον ο χρήστης βρίσκεται σε κατάλληλο χώρο και εφόσον έχει εγκαταστήσει το παιχνίδι Simon Says στο Second Life με κάποια από τις παραπάνω υλοποιήσεις, έχει τη δυνατότητα να παίξει το κλασικό παιχνίδι εντός του Second Life.

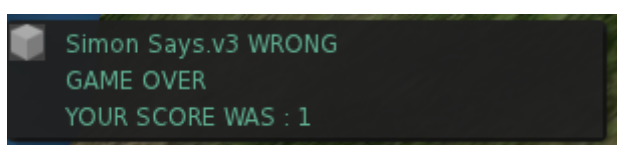
Για να ξεκινήσει η προβολή της χρωματικής ακολουθίας, ο χρήστης αρκεί να πατήσει το κουμπί START. Τότε, ένα από τα τέσσερα χρωματιστά κουμπιά θα αναβοσβήσει (αλλάζει το χρώμα στη φωτεινή του έκδοση) επιδεικνύοντας το χρώμα που πρέπει να πατήσει ο χρήστης για να συνεχιστεί το παιχνίδι. Εφόσον ο χρήστης «πατήσει» το συγκεκριμένο χρωματιστό κουμπί, επιλέγοντας με το αριστερό κλικ του ποντικιού (ταυτίζεται με τη λειτουργία Touch), η ακολουθία χρωμάτων θα συνεχιστεί, προσθέτοντας ακόμη ένα χρώμα στην ήδη υπάρχουσα. Έτσι θα αναβοσβήσει το πρώτο κουμπί (για 0.5 δευτερόλεπτα) και μετά από ένα σύντομο χρονικό διάστημα (1 δευτερόλεπτο) θα αναβοσβήσει το δεύτερο κουμπί. Το πρώτο κουμπί είναι ίδιο με αυτό που προβλήθηκε μόνο, οπότε γίνεται κατανοητό ότι η ακολουθία σε κάθε γύρο που ακολουθεί σωστή απάντηση του χρήστη, διευρύνεται και δεν παράγεται ξανά από την αρχή. Για παράδειγμα, η ακολουθία Κόκκινο-Κόκκινο-Μπλε, μετά από σωστή επιλογή του χρήστη δια διαμορφωθεί σε Κόκκινο-Κόκκινο-Μπλε-Κίτρινο.

Εάν ο χρήστης επιλέξει σωστά τα κουμπιά, τότε ένα μήνυμα «CORRECT» θα εμφανιστεί στη δημόσια συνομιλία του (Chat, στο channel 0) όπως φαίνεται στην Εικόνα 3.28.



Εικόνα 3.28. Μήνυμα από το αντικείμενο «Simon Says.v3».

Στην περίπτωση τώρα, που ο χρήστης κάνει κάποια λάθος επιλογή, τότε το παιχνίδι θα τον ενημερώσει πάλι μέσω της συνομιλίας με το μήνυμα «WRONG» ακολουθούμενο από το μήνυμα «GAME OVER». Ακόμη, μετά από 500 ms (0.5 δευτερόλεπτα), εμφανίζεται μήνυμα με το σκορ του παίκτη σε μορφή «YOUR SCORE WAS:» (Εικόνα 3.29). Το σκορ υπολογίζεται σύμφωνα με το μέγιστο αριθμό χρωμάτων σε μια ακολουθία, όπου ο παίκτης κατάφερε να μαντέψει σωστά. Άρα το σκορ εκφράζει τον αριθμό των χρωμάτων στην ακολουθία που ο παίκτης έκανε λάθος επιλογή, μείον 1. Ξεχωριστή περίπτωση πρέπει να θεωρείται η ακολουθία με ένα μόνο χρώμα κατά την οποία το σκορ διαμορφώνεται στο 0 και όχι στο -1 (αρνητικό σκορ δεν έχει υπόσταση). Έτσι, λαμβάνονται υπόψιν όλες οι πιθανές εκδοχές του σκορ, το οποίο αποκτά μόνο θετικές τιμές ή τη μηδενική τιμή.



*Εικόνα 3.29. Τα προβαλλόμενα μηνύματα στην περίπτωση λάθος επιλογής χρώματος του χρήστη.*

Ταυτοχρόνως, μπορεί να γίνει αντιληπτό ότι με το ξεκίνημα του παιχνιδιού και την επιλογή του κουμπιού «START» από τον χρήστη, το κουμπί «START» αλλάζει μορφή και αναφέρει την επιλογή «RESET» στη μπροστινή όψη του (εντεύθεν και το όνομα STARTRESET του κουμπιού).

Με την επιλογή του κουμπιού «RESET» τώρα, επιτυγχάνεται η μερική επαναφορά του παιχνιδιού (Soft Reset). Η λειτουργία «Soft Reset» διαφέρει από τη λειτουργία «Hard Reset» που έχει αναφερθεί και θα αναλυθεί στη συνέχεια. Αυτό που πετυχαίνει το «Soft Reset» είναι να σταματάει η εξέλιξη του τρέχοντος παιχνιδιού και να επανέρχεται σε αδρανή κατάσταση. Αυτή η λειτουργία μπορεί να φανεί χρήσιμη σε περιπτώσεις όπου υπάρξει σφάλμα κατά την εκτέλεση του παιχνιδιού, χαθεί προσωρινά η σύνδεση του Second Life Viewer με το διαδίκτυο ή εάν ο παίκτης επιθυμεί να αποσυρθεί από τον τρέχον γύρο του παιχνιδιού.

Στη συγκεκριμένη περίπτωση, πατώντας το «RESET» κουμπί, το object δεν ενημερώνει τον χρήστη ότι το παιχνίδι τερματίστηκε και αρχικοποιήθηκε, αντ' αυτού επαναφέρει την όψη του κουμπιού στην αρχική, με την όψη του «START». Ο λόγος που προτιμήθηκε η μη εμφάνιση μηνυμάτων διαλόγου στο χρήστη είναι η επιθυμία ταχύτητας και αμεσότητας σε αυτήν τη λειτουργία καθώς η φύση των προβλημάτων τα οποία αντιμετωπίζει είναι άμεσα συνδεδεμένη με τη χρονική καθυστέρηση. Επιπλέον καθυστέρηση, ακόμη και μικρή, για την εμφάνιση μηνύματος, θεωρήθηκε περιττή.





Εικόνα 3.30. Το παράθυρο με πληροφορίες που εμφανίζεται πατώντας το κουμπί INFO.

Τέλος, ιδιαίτερη χρησιμότητα κατέχει το κουμπί «INFO». Με το πάτημα του συγκεκριμένου κουμπιού ανοίγει ένα νέο παράθυρο στην επιφάνεια εργασίας του Second Life και πληροφορεί τον χρήστη για τις δυνατότητες του παιχνιδιού που έχει μπροστά του ενώ παρέχει ταυτόχρονα οδηγίες χρήσης στα Αγγλικά. Περιλαμβάνει συνολικά 4 διαφορετικές καρτέλες, για διαφορετικές λειτουργίες του παιχνιδιού και προσφέρει περιήγηση μεταξύ αυτών μέσω ισάριθμων κουμπιών. Οι διαφορετικές σελίδες έχουν ως περιεχόμενο μια εισαγωγική σελίδα με πληροφορίες και υπενθυμίσεις για το παιχνίδι, μια σελίδα που υποδεικνύει ότι το παιχνίδι είναι δυνατόν να παιχτεί χρησιμοποιώντας αποκλειστικά το Second Life ή και έναν διαδικτυακό φυλλομετρητή (Web Browser) και 2 σελίδες με υποδείξεις για τους 2 διαφορετικούς τρόπους λειτουργίας. Πέρα από τα 4 κουμπιά «How to Play», «Second Life», «Web Browser» και «Back», η συνάρτηση `llDialog()` του Second Life προσφέρει και τα κουμπιά «Block» και «Ignore» σε κάθε σελίδα τα οποία επιτρέπουν τη διακοπή εμφάνισης των εν λόγω καρτελών. Παρόλα αυτά, το παράθυρο των καρτελών μπορεί να τερματιστεί εύκολα πατώντας το «X» στην δεξιά κορυφή της εκάστοτε καρτέλας.

Σε αυτό το σημείο υπενθυμίζεται ότι η φυσική υπόσταση του Project στο Second Life αποτελείται από πολλαπλά αντικείμενα, όπως η ταμπλέτα `parent` και τα κουμπιά `HARD_RESET` και `INFO`, αλλά και `script` μέσα σε αυτά όπως τα αρχεία `parent_v8` και άλλα. Το τελικό προϊόν το οποίο διανέμεται και από το Second Life Marketplace, φέρει την ονομασία «Simon Says.v3» και αυτό είναι το όνομα που θα εμφανίζεται στο Second Life τις περισσότερες φορές όσο το αντικείμενο παραμένει αυτούσιο. Φυσικά υπάρχουν εξαιρέσεις κατά τις οποίες το όνομα που εμφανίζεται στο παράθυρο Chat υποδεικνύει το όνομα κάποιου κουμπιού. Ο λόγος που συμβαίνει αυτό έχει να κάνει με την προσπάθεια του παιχνιδιού ώστε να εξοικειωθεί ο χρήστης με το User Interface του καθώς και με τις διαφορετικές λειτουργίες.

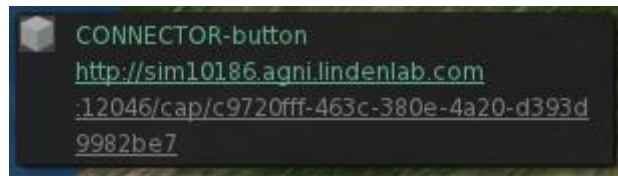
Το όνομα «Simon Says.v3» παραμένει στις περιπτώσεις που τα μηνύματα έχουν άμεση σχέση με τις συνθήκες παιχνιδιού (Gameplay) και όχι με τη συνδεσιμότητα του.

### 3.4.3 ΧΡΗΣΗ ΜΕΣΩ ΦΥΛΛΟΜΕΤΡΗΤΗ (WEB BROWSER)

Μια από τις ιδιαίτερες δυνατότητες αυτού του Project είναι η ικανότητα του να επικοινωνεί με οποιονδήποτε Web Browser αντλώντας και στέλνοντας δεδομένα σε πραγματικό χρόνο. Με αυτόν τον τρόπο επιτυγχάνεται Cloud Gaming Service κατά την οποία δε συμβαίνει μεταφορά πληροφορίας μέσω εικόνας αλλά μεταφορά πληροφορίας μέσω δεδομένων εισόδου και εξόδου. Για αυτήν τη χρήση οι ειδικά διαμορφωμένες σελίδες HTML και PHP που χρησιμοποιούνται, λειτουργούν ως κελύφη του παιχνιδιού καθώς είναι προγραμματισμένα να εκτελούν εντολές ανάλογα τα δεδομένα εισόδου. Παρόλα αυτά, οι σελίδες που αναπτύχθηκαν δεν αποτελούν λειτουργικά παιχνίδια από μόνα τους καθώς αυτό δεν ήταν επιθυμητό και με αυτόν τον τρόπο χάνονται πολλά από τα πλεονεκτήματα αυτής της υπηρεσίας όπως είναι η απλότητα του εκτελούμενου κώδικα, το μικρό μέγεθος και η ταχύτητα εκτέλεσης. Σε αντίθετη περίπτωση, κατά την επεξεργασία των δεδομένων θα εμφανιζόντουσαν μεγάλες χρονικές καθυστερήσεις αφού θα έπρεπε σε κάθε Web Browser το παιχνίδι Simon Says να εκτελείται στην πλήρη μορφή του.

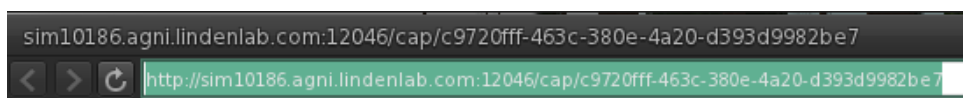
Για να μπορεί τώρα ο χρήστης να παίξει το παιχνίδι μέσω ενός Web Browser θα πρέπει να έχει ολοκληρώσει τα προηγούμενα βήματα της εγκατάστασης και της υλοποίησης του Project στο Second Life. Ακόμη, θα πρέπει να βρίσκεται στο Second Life, σε region με τα κατάλληλα δικαιώματα και με το Simon Says να βρίσκεται στην κατάσταση ηρεμίας όπου δεν εκτελείται κάποιο παιχνίδι.

Σε αυτό το σημείο ο χρήστης θα πρέπει να πατήσει (touch event) το μικρό μαύρο τριγωνικό κουμπί (κουμπί CONNECTOR) που βρίσκεται στο κάτω κεντρικό σημείο της πλακέτας parent από το Project Simon Says.v3. Αυτομάτως θα δημιουργηθεί ένα μοναδικό URL το οποίο μπορεί να χρησιμοποιηθεί για την HTTP επικοινωνία. Αυτό το μοναδικό URL θα εμφανιστεί στη συνομιλία του χρήστη και θα έχει τη μορφή της Εικόνας 3.31. Από εκεί, ο χρήστης έχει τη δυνατότητα να επιλέξει (κλικάρει) αυτό το link ώστε να ανοίξει ξεχωριστός φυλλομετρητής εντός του Second Life. Σημειώνεται πως σε περίπτωση που ο χρήστης επιθυμεί να ξαναδεί το παραχθέν URL μπορεί να το κάνει είτε αν επισκεφθεί το ιστορικό συνομιλίας του, είτε εάν ξαναπατήσει το κουμπί CONNECTOR. Το συγκεκριμένο κουμπί είναι προγραμματισμένο να επανεμφανίζει το URL που έχει παραχθεί ή να παράγει καινούργιο σε περίπτωση που το ήδη υπάρχον απελευθερώθηκε από τον χρήστη (Released URL).



Εικόνα 3.31. Το URL που εμφανίζει το κουμπί "CONNECTOR".

Όπως αναφέρθηκε, εάν ο χρήστης επιλέξει να κλικάρει το URL που παράχθηκε τότε θα ανοίξει ένας εσωτερικός φυλλομετρητής του Second Life ο οποίος περιλαμβάνει ένα QR code (Εικόνα 3.32) ενώ ο header του φυλλομετρητή είναι το URL που παράχθηκε.



Take a QR code with your smartphone.



Εικόνα 3.32. Το παράθυρο του φυλλομετρητή του Second Life.

Η χρήση του παραγόμενου URL γίνεται σε συνδυασμό με τη σελίδα : «<http://siwonsimon.000webhostapp.com/siwonsimon.html>» στην οποία ο χρήστης μπορεί να αλληλεπιδράσει με το παιχνίδι Simon Says χωρίς να βρίσκεται σε πρόσβαση με το Second Life.

Για να λειτουργήσει η HTTP σύνδεση το παραχθέν URL θα πρέπει να τοποθετηθεί μετά τη σελίδα του HOST siwonsimon και η μορφή της θα πρέπει να είναι η εξής : «<http://siwonsimon.000webhostapp.com/siwonsimon.html?http://sim10186.agni.lindenlab.com:12046/cap/c9720fff-463c-380e-4a20-d393d9982be7>». Προφανώς και κάθε φορά που γίνεται generated ένα URL από το project, αυτό θα είναι διαφορετικό από κάθε προηγούμενο ενώ το προηγούμενο URL που αποδεσμεύεται δε θα έχει πια κάποια χρήση.

Άρα για να μπορέσει κάποιος να έχει πρόσβαση στο παιχνίδι Simon Says μέσω HTTP σύνδεσης χρειάζεται τη σελίδα «<http://siwonsimon.000webhostapp.com/siwonsimon.html>» και το παραχθέν URL τοποθετημένο μετά το link της σελίδας. Σημαντικό είναι ανάμεσα στη σελίδα του HOST και το παραχθέν URL, να τοποθετηθεί ο χαρακτήρας του αγγλικού ερωτηματικού «?».

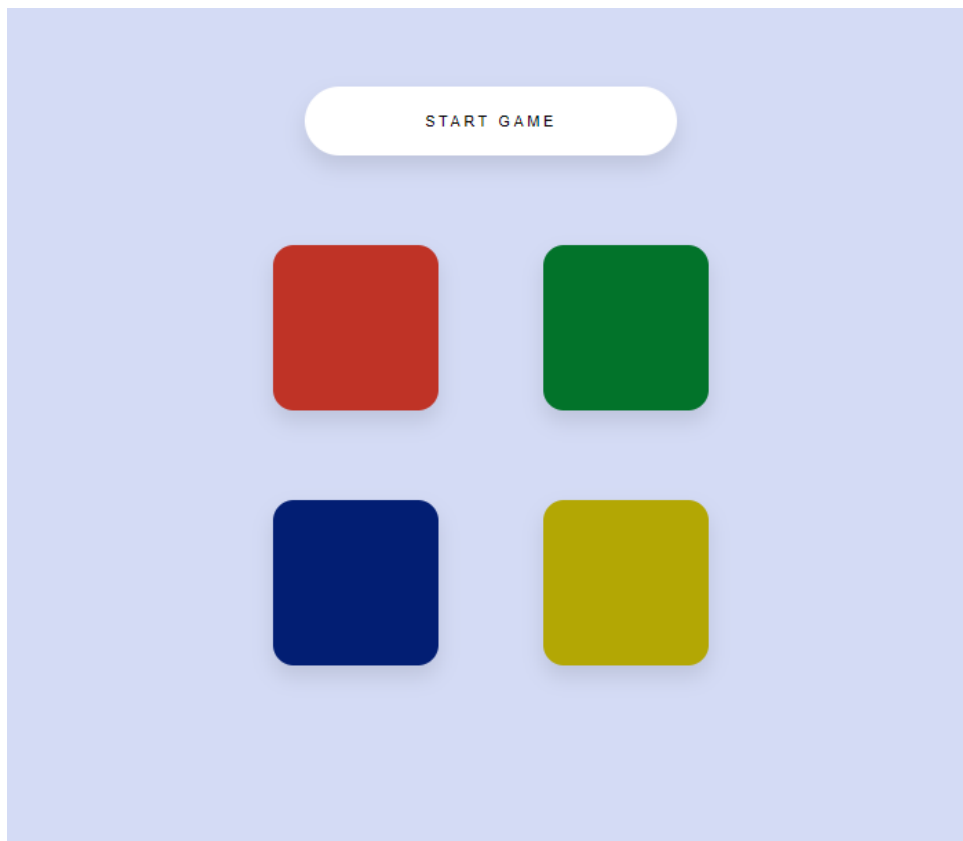
Για να μπορέσει να διαπεραστεί το συγκεκριμένο σύνολο από τη σελίδα και το URL, ο χρήστης μπορεί να αντιγράψει χειροκίνητα το παραχθέν URL από την επικεφαλίδα του φυλλομετρητή του Second Life, να το προσθέσει μετά από το link της σελίδας «<http://siwonsimon.000webhostapp.com/siwonsimon.html?>» (εδώ έχει προστεθεί το αγγλικό ερωτηματικό που χρειάζεται για ευκολία κατανόησης) και να μεταβεί σε αυτήν.

Ευκολότερο τρόπο αποτελεί η χρήση του QR code που δημιουργείται στο Second Life. Έτσι, χρησιμοποιώντας την κάμερα ενός κινητού, ο χρήστης μπορεί να σαρώσει τον εν λόγω κωδικό και τότε αυτόματα θα περιηγηθεί στην επιθυμητή σελίδα από όπου θα μπορεί να αλληλεπιδράσει με το Simon Says στο Second Life χωρίς επιπλέον βήματα αρκεί το παιχνίδι Simon Says.v3 να είναι εκτελέσιμο στον κόσμο του Second Life.

#### 3.4.4 ΔΙΕΠΑΦΗ ΣΕΛΙΔΑΣ

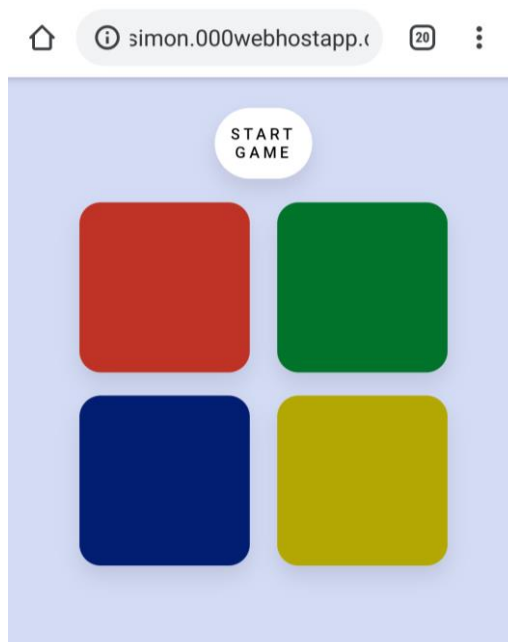
Η σελίδα που θα χρησιμοποιηθεί από τον Web Browser ώστε να επικοινωνήσει με το Second Life θα έχει τη μορφή της Εικόνας 3.33 εάν είναι προσωπικού υπολογιστή (PC) ή της Εικόνας 3.34 εάν είναι Web Browser σε κινητή συσκευή.

Εκεί ο χρήστης αρκεί να πατήσει το κουμπί «START GAME» και σύντομα κάποιο από τα 4 χρωματιστά κουμπιά θα αναβοσβήσει υποδεικνύοντας το κουμπί που πρέπει να επιλέξει ο χρήστης. Οι κανόνες του παιχνιδιού παραμένουν ίδιοι ενώ σε περίπτωση που ο χρήστης κάνει κάποιο λάθος τότε θα εμφανιστεί το μήνυμα «GAME OVER» και το σκορ του χρήστη. Επιπλέον το σκορ του χρήστη εμφανίζεται διαρκώς κατά τη διάρκεια του παιχνιδιού ενώ υπολογίζεται με τον ίδιο τρόπο όπως και στο Second Life. Όλα τα μηνύματα καθώς και το σκορ εμφανίζονται στην ίδια περιοχή της σελίδας και αλλάζουν δυναμικά με custom animations για τη μέγιστη δυνατή εμπειρία και ευκολία χρήσης. Ακόμη, θα πρέπει να τονιστεί πως στην πραγματικότητα το παιχνίδι Simon Says δεν «τρέχει» στη σελίδα του Web Browser, απλά δέχεται δεδομένα από το Second Life. Παρόλα αυτά, το σκορ του παιχνιδιού δεν αποστέλλεται από το Second Life αλλά υπολογίζεται στον Web Browser μέσω μεταβλητών μέτρησης των δεδομένων. Επιλέχθηκε αυτός ο τρόπος προσέγγισης καθώς διαπιστώθηκε ότι το μήκος του εκτελούμενου κώδικα καθώς και ο αριθμός των μεταβλητών που χρησιμοποιούνται θα ήταν μεγαλύτερος σε περίπτωση που η σελίδα του Web Browser θα έπρεπε να επεξεργάζεται δεδομένα διαφορετικής φύσης (δεδομένα της χρωματικής ακολουθίας και δεδομένα του σκορ) από ότι αν υπολογίζει το σκορ ζωντανά στον Host Server.



Εικόνα 3.33. Η διεπαφή της σελίδας όπως φαίνεται σε Web Browser υπολογιστή (PC).

Τέλος θα πρέπει να αποσαφηνιστεί ότι ενόσω κάποιος χρήστης αλληλεπιδράει με το παιχνίδι κάποιου είδους Web Browser, μπορεί οποιαδήποτε στιγμή να σταματήσει το παιχνίδι εκεί και να συνεχίσει στο Second Life χωρίς να χρειαστεί να πράξει κάτι περαιτέρω.



Εικόνα 3.34. Η διεπαφή της σελίδας όπως εμφανίζεται σε Web Browser κινητής συσκευής (smartphone).



### 3.4.5 ΑΠΕΛΕΥΘΕΡΩΣΗ URL

Όπως έχει αναφερθεί και προηγουμένως αλλά και όπως ενημερώνεται κάποιος είτε από τη σελίδα του Simon Says.v3 στο Second Life Marketplace είτε μέσω της Notecard του παιχνιδιού είτε ακόμη από την κατηγορία «INFO» του ομώνυμου κουμπιού της πλακέτας στον κόσμο του Second Life, όλα τα URL που κάποιος χρήστης παράγει για τη σύνδεση HTTP μορφής θα πρέπει να απελευθερώνονται στο τέλος χειροκίνητα. Αυτό συμβαίνει καθώς τα συγκεκριμένα URL καταλαμβάνουν φυσική θέση στη συνολική χωρητικότητα του κάθε region στον κόσμο του Second Life. Με αυτόν τον τρόπο, η μητρική εταιρεία Linden αποτρέπει την κατάχρηση του διαθέσιμου Server από τον οποιοδήποτε. Ακόμη, το Second Life διαθέτει αυτόματο σύστημα απελευθέρωσης των URL μετά από ένα εύλογο χρονικό διάστημα, όμως η αποτελεσματικότητα αυτού του συστήματος επιδέχεται περαιτέρω έρευνα. Ωστόσο, με τη χειροκίνητη απελευθέρωση URL ο χρήστης προστατεύει την αξιοπιστία και την ακεραιότητα του αντικειμένου του από πιθανές υποκλοπές δεδομένων και στοιχείων.



Εικόνα 3.35. Προβαλλόμενο μήνυμα σε περίπτωση «HARD RESET».

Για να μπορέσει ο χρήστης να απελευθερώσει χειροκίνητα τα παραχθέντα URL αρκεί να πατήσει (touch event) το παραλληλόγραμμο κουμπί του HARD\_RESET που βρίσκεται στην κάτω δεξιά γωνία της πλακέτας του Simon Says.v3. Με αυτόν τον τρόπο θα αποδεσμευτούν οι συνδέσεις HTTP και ως επακόλουθο χρήστες που διαθέτουν και λειτουργούν το συγκεκριμένο URL μέσω ενός Web Browser θα πάνσουν να μπορούν να αλληλοεπιδρούν με το παιχνίδι. Ακόμη το παιχνίδι στο Second Life θα επανέλθει στην αρχική του λειτουργία (Soft Reset) και για όλα αυτά ο χρήστης θα ενημερωθεί στο Second Life μέσω του παραθύρου της συνομιλίας, όπως φαίνεται στην Εικόνα 3.35.

Έτσι, ο χρήστης σε εκείνη τη στιγμή θα μπορεί να επαναλάβει οποιοδήποτε από τα προηγούμενα βήματα ώστε να επιλέξει τη λειτουργία του παιχνιδιού που επιθυμεί από την αρχή.

### 3.4.6 ΕΠΕΞΕΡΓΑΣΙΑ ΑΝΤΙΚΕΙΜΕΝΩΝ ΚΑΙ DEBUGGING

Σύμφωνα και με την ενότητα υλοποίησης του project στο Second Life εκ του μηδενός, ο χρήστης μπορεί να δημιουργήσει τη δικιά του εκδοχή της όψης του παιχνιδιού εντός του Second Life. Παρόλα αυτά, υπάρχει και η δυνατότητα να επεξεργαστεί και να αλλοιώσει το ήδη υπάρχον αντικείμενο που προσφέρεται μέσω του Second Life Marketplace.

Ο τρόπος για να επιτευχθεί αυτό είναι να μεταβεί στην κατηγορία Edit μέσω δεξιού κλικ πάνω στο αντικείμενο Simon Says.v3 κι εκεί να επιλέξει το κουμπί «Unlink» σύμφωνα και με την Εικόνα 3.36.



Εικόνα 3.36. Το κουμπί Unlink που καταστεί δυνατή την επί μέρους επεξεργασία του αντικειμένου.

Με αυτόν τον τρόπο κάθε επί μέρους στοιχείο της πλακέτας είναι διαθέσιμο προς επεξεργασία και παραμετροποίηση ενώ έτσι είναι και προσβάσιμα όλα τα αρχεία κώδικα (scripts) και αρχεία όψης (textures) που υπάρχουν αποθηκευμένα στα στοιχεία. Αρκεί ο χρήστης να αποχωρήσει από τη λειτουργία «Edit» μετά το «Unlink» και στη συνέχεια να επιλέξει όποιο αντικείμενο επιθυμεί, πάλι με δεξί κλικ και «Edit». Αξίζει να σημειωθεί πως ακόμα και σε αυτήν την κατάσταση, η πλακέτα Simon Says.v3 και γενικά το παιχνίδι Simon Says είναι πλήρως λειτουργικό καθώς η εντολή «Unlink» δεν την επηρεάζει στο ελάχιστο. Ο λόγος ύπαρξης της σύζευξης των αντικειμένων (Link) είναι η αδυναμία του Second Life να δέχεται ως προϊόντα, αντικείμενα τα οποία δεν έχουν συζεύξει μεταξύ τους ώστε να παράγουν ένα τελικό ομοιογενές αντικείμενο.

Πέρα όμως από την επεξεργασία των αντικειμένων υπάρχει διαθέσιμη σελίδα HTML στον Host Server όπου διευκολύνει το debugging για οποιονδήποτε επιθυμεί να παραμετροποιήσει το Project. Αυτή η σελίδα είναι πανομοιότυπη με την κανονική σελίδα που χρησιμοποιείται για την HTTP σύνδεση, μόνο που διαθέτει επιπλέον ένα κουμπί «Load» για την χειροκίνητη φόρτωση των δεδομένων του Second Life καθώς και ένα πεδίο «Screen» για την προβολή των συγκεκριμένων δεδομένων. Η διεύθυνση της σελίδας είναι η παρακάτω : [http://siwonsimon.000webhostapp.com/siwonsimon\\_debugging.html](http://siwonsimon.000webhostapp.com/siwonsimon_debugging.html) ενώ ο κώδικας της παρέχεται και στην κεντρική σελίδα της εργασίας στο GitHub.

### 3.4.7 ΣΗΜΕΙΩΣΕΙΣ

Τόσο το παραχθέν URL του Second Life όσο και η σελίδα του Host που χρησιμοποιείται δε φέρουν πιστοποιητικό SSL. Αυτό συμβαίνει διότι παρά την παροχή του Second Life εντολής παραγωγής HTTPS URL, το παραχθέν URL δε φέρει πιστοποίηση SSL με συνέπεια όλοι οι σύγχρονοι φυλλομετρητές να το εκλαμβάνουν ως απειλή και να μην επιτρέπουν να διαπεραστεί. Για αυτόν τον λόγο επιλέχθηκε να μη χρησιμοποιηθεί το SSL πιστοποιητικό (αν και αποκτήθηκε) στη σελίδα του Host Server καθώς χωρίς ενεργό HTTPS URL δε θα μπορούσε να τεθεί σε λειτουργία.

### 3.5 ΣΥΝΟΨΗ ΚΕΦΑΛΑΙΟΥ

Σε αυτό το κεφάλαιο έγινε διεξοδική ανάλυση των περιεχόντων του πακέτου Simon Says.v3 στο Second Life Marketplace, ενώ δείχθηκε και ο τρόπος εγκατάστασης του συγκεκριμένου πακέτου αλλά και η υλοποίηση του από την αρχή.

Ακόμη, αναλύθηκαν οι οδηγίες χρήσης του παιχνιδιού τόσο εντός του Second Life όσο και με τη χρήση εξωτερικού υπολογιστή ή κινητής συσκευής.

Τέλος, αναφέρθηκαν ορισμένες παρατηρήσεις οι οποίες χρήζουν της προσοχής του χρήστη όσον αφορά τον κόσμο του Second Life.

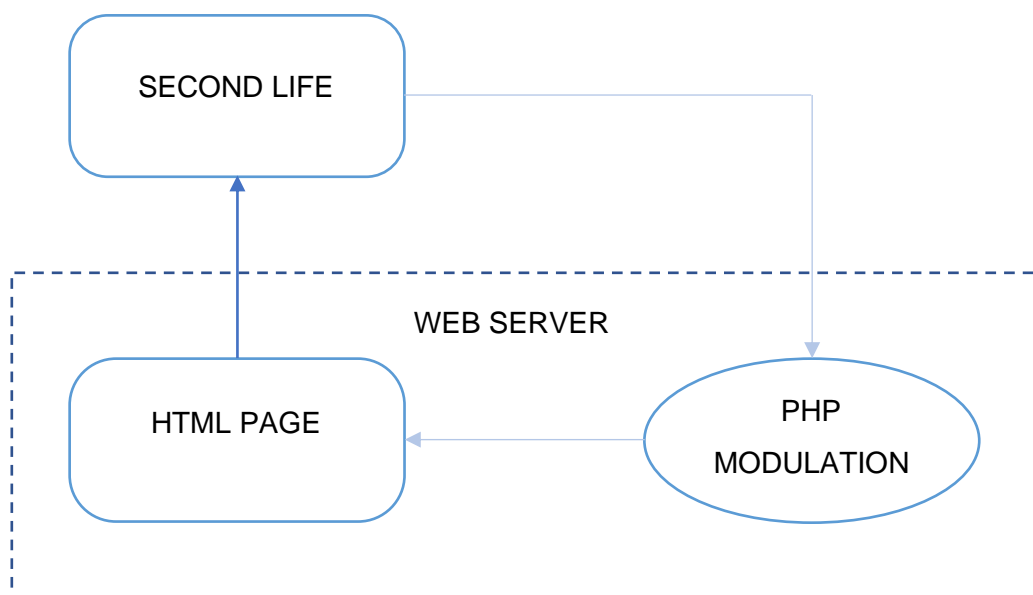
## ΚΕΦΑΛΑΙΟ 4: Ανάλυση και Επεξεργασία Δεδομένων

### 4.1 ΕΙΣΑΓΩΓΗ

Το project στο σύνολο του τίθεται σε λειτουργία εντός του Second Life και ταυτόχρονα επικοινωνεί μέσω HTTP επικοινωνίας με έναν Web Server. Το παιχνίδι Simon Says, από τη στιγμή που ενεργοποιηθεί εντός του Second Life είναι ικανό να αναγνωρίσει και να διαχειριστεί τις αλληλεπιδράσεις του χρήστη είτε εντός του Second Life είτε μέσω του Web Browser. Ακόμη, έχει τη δυνατότητα να αναγνωρίζει εναλλαγές στον τρόπο που αλληλεπιδράει ο χρήστης με αυτό και να προσαρμόζεται κατάλληλα επιτρέποντας στον χρήστη να συνεχίσει το παιχνίδι του με εναλλαγή από Second Life σε Web Browser ή και αντίστροφα.

Για να καταστεί αυτό δυνατό, χρησιμοποιούνται αρχεία κώδικα (scripts) στο Second Life που εκτελούν το παιχνίδι Simon Says ενώ ταυτόχρονα επεξεργάζονται τις αλληλεπιδράσεις του χρήστη με αυτό αλλά και τις διαύλους επικοινωνίας που υπάρχουν.

Επίσης χρησιμοποιείται ένας Web Server (Host σε αυτήν την περίπτωση) όπου επικοινωνεί με το Second Life. Συγκεκριμένα, μια σελίδα HTML περιέχει το user interface και διαχειρίζεται τα δεδομένα που αποστέλλονται αλλά και που λαμβάνονται. Η HTML σελίδα αποστέλλει απευθείας τα δεδομένα στο Second Life ενώ λαμβάνει δεδομένα από αυτό μέσω μιας επεξεργασίας που περιλαμβάνει ένα αρχείο PHP, όπως φαίνεται και στο Γράφημα 4.1. Τόσο το Second Life όσο και ο Web Server είναι απαραίτητο να είναι συνδεδεμένοι σε μία σταθερή σύνδεση Internet ώστε να καταστεί δυνατή τόσο η εκτέλεση του παιχνιδιού Simon Says στο Second Life όσο και η επικοινωνία με τον Web Server.



Γράφημα 4.1. Η διάταξη επικοινωνίας του συνολικού Project.

## 4.2 SECOND LIFE

Στο Second Life η υλοποίηση και η λειτουργία του Project επιτυγχάνεται με τη χρήση 9 διαφορετικών αρχείων κώδικα (scripts) τοποθετημένα σε ισάριθμα αντικείμενα (objects) του Second Life. Αυτά τα αντικείμενα επιτελούν ξεχωριστή λειτουργία το κάθε ένα ενώ επικοινωνούν μεταξύ τους στην πλειοψηφία τους, ώστε να είναι δυνατή η διεκπεραίωση του παιχνιδιού.

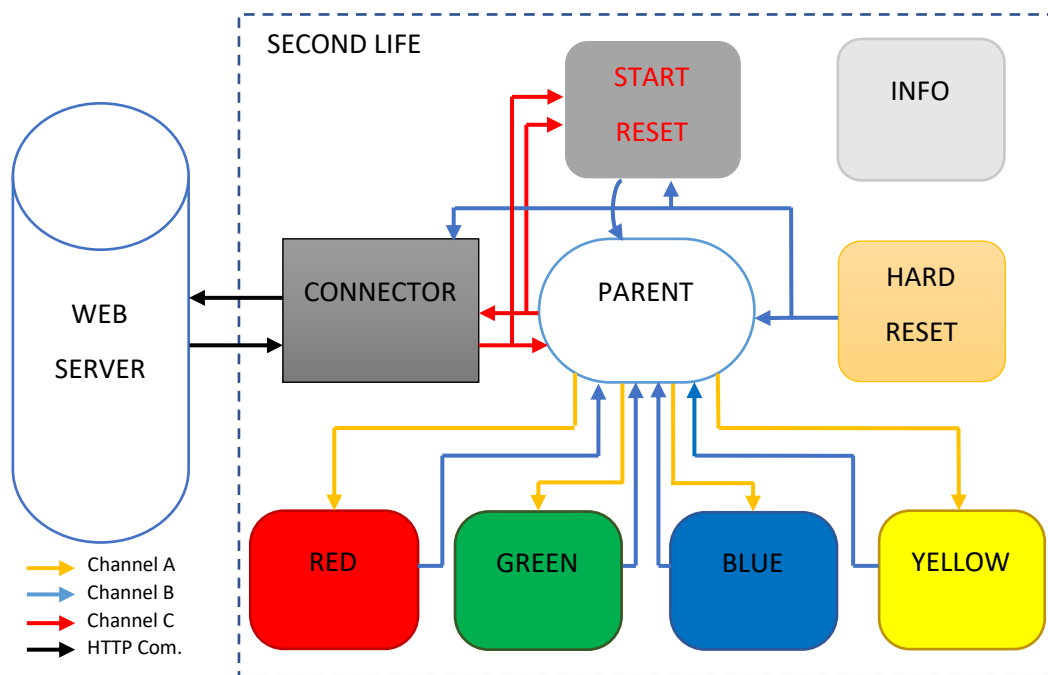
Σε ένα βασικό script στο Second Life υπάρχει ένα default state στο οποίο διεκπεραιώνονται τα ζητούμενα του κώδικα. Όμως ένα αντικείμενο μπορεί να έχει παραπάνω από μία καταστάσεις (states) στις οποίες εναλλάσσεται ανάλογα τα εξωτερικά ερεθίσματα ή προγραμματισμένες λειτουργίες. Κάθε state περιλαμβάνει ένα state\_entry(){} το οποίο αποτελεί την αρχικοποίηση της συγκεκριμένης κατάστασης. Σε αυτήν γίνονται οι αρχικοποιήσεις των μεταβλητών που χρησιμοποιούνται στη συγκεκριμένη κατάσταση (state).

Η επικοινωνία μεταξύ των αντικειμένων γίνεται χάρη στην εντολή του Second Life lISay(channel,"text") ή lShout(channel,"text") με τις οποίες αποστέλλονται δεδομένα στο εκάστοτε channel και τη συνάρτηση listen με την οποία γίνεται παραλαβή δεδομένων.

Στο συγκεκριμένο Project γίνεται χρήση 3 διαφορετικών καναλιών επικοινωνίας (channels) μέσω των οποίων τα διαφορετικά αντικείμενα του Second Life μπορούν να επικοινωνήσουν. Αυτά τα 3 κανάλια φέρουν διαφορετική αρίθμηση το κάθε ένα, ενώ η αρίθμηση τους έχει επιλεχτεί τυχαία αλλά πάντα με γνώμονα την αποφυγή τυχαίας παρεμβολής. Αυτό συμβαίνει διότι μικροί αριθμοί ή συνεχόμενοι (πχ 1, 2, 3) μπορούν να χρησιμοποιούνται τυχαία από διαφορετικούς χρήστες στην περιοχή του Second Life που μπορεί να βρίσκεται ο χρήστης κι έτσι συμπτωματικώς να προκαλέσουν ανεπιθύμητες παρεμβολές στο σύστημα κι εν τέλει τη μη σωστή λειτουργία του.

Άρα για αυτόν το λόγο, επιλέχθηκαν ως κανάλια επικοινωνίας τα Channel A = -5243212, Channel B = -5243210 και Channel C = -5154789. Οι αρνητικές τιμές της αρίθμησης δεν προκαλούν κάποιο πρόβλημα ενώ με αυτόν τον τρόπο εξασφαλίζεται ακόμα περισσότερο η τυχαιότητα των αριθμών. Ο τρόπος επικοινωνίας στο Second Life φαίνεται στο Γράφημα 4.2.

Στο Γράφημα 4.2 γίνεται εμφανής η πολύπλοκη επικοινωνία που λαμβάνει χώρα μεταξύ των διαφορετικών αντικειμένων του Second Life. Ωστόσο, σε προσπάθεια να γίνει πιο απλή η φύση της επικοινωνίας μεταξύ των αντικειμένων, τα κανάλια επικοινωνίας ομαδοποιήθηκαν στις 3 προαναφερθείσες κατηγορίες. Η ομαδοποίηση έγινε με γνώμονα τις διαφορετικές λειτουργίες που λαμβάνουν χώρα στο Second Life και τη σχέση κάθε λειτουργίας με το κεντρικό αντικείμενο PARENT. Για να καταστεί δυνατή η επικοινωνία το αντικείμενο που επιθυμεί να αποστείλει δεδομένα χρησιμοποιεί την εντολή lISay(Channel, "text").



Γράφημα 4.2. Η επικοινωνία των αντικειμένων στο Second Life.

Προφανώς με αυτόν τον τρόπο ένα αντικείμενο μπορεί να αποστέλλει πολλαπλά μηνύματα σε ένα ή και σε πολλαπλά κανάλια επικοινωνίας ταυτόχρονα. Ακόμη, πολλαπλά αντικείμενα μπορούν να ακούσουν το ίδιο μήνυμα ταυτόχρονα και να δράσουν αναλόγως. Όμως, η απολαβή των δεδομένων αποτελεί πιο πολύπλοκο εγχείρημα καθώς χρησιμοποιείται συνάρτηση συμβάντος.

Συγκεκριμένα, στην `state_entry` του αντικειμένου που λαμβάνει δεδομένα, αρχικοποιείται μια μεταβλητή απολαβής (`handle`) για το συμβάν της ακρόασης (`listen event`) [1]. Στη συγκεκριμένη αρχικοποίηση, ορίζεται το κανάλι το οποίο το αντικείμενο απολαβής θα παρακολουθεί. Ένα αντικείμενο έχει τη δυνατότητα να παρακολουθεί παραπάνω από ένα κανάλια επικοινωνίας, άρα θα πρέπει να οριστούν περισσότερες μεταβλητές `listen_handle`.

Στη συνέχεια, εκτός του `state_entry`, δημιουργείται το event `listen()` εντός του οποίου το αντικείμενο μπορεί να διαχειριστεί τα διαφορετικά channel με τη χρήση μιας συνθήκης `if` αλλά και να διαχειριστεί διαφορετικά κείμενα που έχουν αποσταλεί στο συγκεκριμένο channel, πάλι με τη χρήση μιας εμφωλευμένης συνθήκης `if`, στην υπάρχουσα `if` του καθορισμού του channel. Άρα, ένα αντικείμενο μπορεί ταυτόχρονα να «ακούει» πολλαπλά κανάλια επικοινωνίας αλλά και να «ψάχνει» διαφορετικά μηνύματα μέσα σε αυτά τα κανάλια και ανάλογα τα μηνύματα τα οποία εντοπίζει, να δρα και αναλόγως.

#### 4.2.1 ΛΕΙΤΟΥΡΓΙΑ ΕΠΙΚΟΙΝΩΝΙΑΣ ΠΑΙΧΝΙΔΙΟΥ SIMON SAYS

Η πρώτη λειτουργία είναι η επικοινωνία της κεντρικής πλακέτας PARENT με τα τέσσερα χρωματιστά κουμπιά και διεκπεραιώνεται με το κανάλι επικοινωνίας Channel A (-5243212). Μέσω αυτής της λειτουργίας το Second Life είναι δυνατόν να αποκτήσει υπόσταση και να μπορεί να παράγει τυχαίες χρωματικές ακολουθίες αλλά και να τις αναπαράγει στον χρήστη εντός του Second Life.

Το αντικείμενο PARENT επικοινωνεί με τα 4 χρωματιστά αντικείμενα RED, GREEN, BLUE και YELLOW ονοματίζοντας στον δίαυλο επικοινωνίας το όνομα εκείνου του αντικειμένου που επιλέχθηκε και πρέπει να ανάψει. Για παράδειγμα, στο κανάλι A μπορεί να εμφανιστεί το μήνυμα «B» το οποίο υποδηλώνει το χρώμα μπλε (BLUE). Στη συγκεκριμένη περίπτωση επικοινωνίας το όνομα ενός χρώματος της ακολουθίας αναφέρεται μόνο με το πρώτο γράμμα του χρώματος στα Αγγλικά. Στην πραγματικότητα, ολόκληρη η ακολουθία των χρωμάτων που παράγονται τυχαία στο αντικείμενο PARENT αναφέρεται μόνο με τα πρώτα γράμματα των χρωμάτων. Δηλαδή μια παραγόμενη ακολουθία θα μπορούσε να είναι η «RRYBR» η οποία αντιστοιχεί στα χρώματα RED, RED, YELLOW, BLUE, RED. Αυτός ο τρόπος απεικόνισης επιλέχτηκε διότι αυτού του είδους η απεικόνιση της χρωματικής ακολουθίας είναι πιο εύκολα επεξεργάσιμος κυρίως στον WEB SERVER όπου και αποστέλλεται αργότερα και προφανώς καταλαμβάνει μικρότερη μνήμη.

Η αποστολή του κάθε χαρακτήρα του χρώματος συμβαίνει ανά 1 δευτερόλεπτο ενώ υπεύθυνη για αυτή τη λειτουργία είναι η συνάρτηση `pattern_SHOUT`. Σε αυτήν, παράγεται κάθε φορά το επόμενο χρώμα της χρωματικής ακολουθίας. Η επιλογή του κάθε χρώματος γίνεται τυχαία. Τα τέσσερα διαθέσιμα χρώματα υπάρχουν σε μορφή λίστας [2] και μια εντολή `lIfRand` επιλέγει έναν τυχαίο ακέραιο αριθμό από το 0 έως το 3. Ο κάθε αριθμός υποδηλώνει και μια θέση στη συγκεκριμένη λίστα, πρακτικά δηλαδή ένα χρώμα. Η εντολή `lIfRand` επιλέγει τον αριθμό μέσω ενός προκαθορισμένου εύρους αριθμών ενώ η ακολουθία των τυχαίων αριθμών είναι σε κοινή χρήση σε ολόκληρη τη διαδικασία προσομοίωσης του Second Life [3]. Άρα η κάθε επιλογή είναι ψευδοτυχαία αφού γίνεται από προκαθορισμένη δεξαμενή αριθμών.

Η καθυστέρηση στην αποστολή των χαρακτήρων της ακολουθίας συμβαίνει διότι ένα μόνο χρώμα της πλακέτας στο Second Life θα πρέπει να είναι ανοιχτό τη φορά. Αυτή η καθυστέρηση του ενός δευτερολέπτου προκύπτει από την εντολή `lISleep(1.0)`[4] που ορίζει το χρόνο για τον οποίο ο εκτελέσιμος κώδικας θα παραμείνει ανενεργός, μέχρι να ξαναξεκινήσει από το σημείο που σταμάτησε. Εάν ολόκληρη η χρωματική ακολουθία στελνόταν ταυτόχρονα σε κάθε κουμπί τότε θα άναβαν όλα μαζί μη κάνοντας δυνατή την επιτυχή προβολή της χρωματικής ακολουθίας.

Στα χρωματικά κουμπιά ωστόσο, αφού έχει διασφαλιστεί η σωστή αναπαραγωγή της χρωματικής ακολουθίας από το αντικείμενο Parent, δεν απαιτείται σε αυτά περαιτέρω επεξεργασίας της ακολουθίας, παρά μόνο την προβολή τους. Άρα, το κάθε ένα αντικείμενο, μόλις λάβει στο κανάλι επικοινωνίας, κείμενο με το όνομα του, αλλάζει χρώμα για χρονικό διάστημα 0.5 δευτερολέπτου και επανέρχεται στην αρχική του κατάσταση. Το χρονικό όριο του μισού δευτερολέπτου σε αυτήν την περίπτωση επιτυγχάνεται μέσω της ξεχωριστής κλήσης της συνάρτησης timer() που εκτελείται μέσω της εντολής ISetTimerEvent(toff\_unit). Η τιμή της μεταβλητής toff\_unit ορίζει το χρονικό διάστημα της καθυστέρησης. Εδώ, εφόσον αλλάζει το χρώμα ολόκληρου του αντικειμένου που υποδεικνύει το χρώμα της ακολουθίας, αλλάζει και η ορισμένη κατάσταση. Έτσι από default state, τα χρωματιστά κουμπιά RED, GREEN, BLUE και YELLOW μεταβαίνουν στα αντίστοιχα state RED\_BUTTON, state GREEN\_BUTTON, state BLUE\_BUTTON και state YELLOW\_BUTTON. Σε αυτές τις καταστάσεις το μόνο που αλλάζει είναι το χρώμα των κουμπιών όπου από τη σκοτεινή (DARK) εκδοχή τους, μεταβαίνουν στη φωτεινή (BRIGHT).

Ωστόσο, αξίζει να σημειωθεί πως όση ώρα το κάθε κουμπί βρίσκεται στη διαφορετική κατάσταση (state) εκτός της προκαθορισμένης (default) δε δύναται να εκτελέσει άλλες εντολές που μπορεί να ζητήσει ο χρήστης ταυτόχρονα. Με αυτόν τον τρόπο εξασφαλίζεται μια προστασία απέναντι σε προβληματική χρήση ορισμένων χρηστών που ίσως επιθυμούν να επιλέγουν πολλαπλά κουμπιά ταυτόχρονα.

Επίσης, σε αυτήν τη λειτουργία επικοινωνίας του παιχνιδιού Simon Says εμπίπτει μόνο η μονόδρομη επικοινωνία του κεντρικού προγράμματος PARENT με τα τέσσερα χρωματιστά κουμπιά επιλογών. Η αντίθετη διαδρομή, από τα τέσσερα κουμπιά προς το αντικείμενο PARENT καλύπτεται από την επόμενη λειτουργία επικοινωνίας. Αυτό συμβαίνει επειδή η αρχική επικοινωνία του PARENT προς τα κουμπιά εξυπηρετεί τη βασική αρχή ύπαρξης του παιχνιδιού.

Η συγκεκριμένη λειτουργία θεωρείται η πιο σημαντική λειτουργία καθώς χωρίς αυτήν δεν υφίσταται Second Life. Έτσι, επιλέχθηκε αυτή η βασική λειτουργία να έχει δικό της ξεχωριστό κανάλι επικοινωνίας. Παρόλα αυτά, όλες οι επικοινωνίες, δυνητικά θα μπορούσαν να εξυπηρετούνται από μόνο ένα κανάλι επικοινωνίας και να μην επηρεάζεται με κάποιον τρόπο η εύρυθμη λειτουργία του project. Όμως με αυτόν τον τρόπο ο κώδικας γίνεται πιο πολύπλοκος και το κυριότερο, πιο δύσκολος στην ανάγνωση και μελλοντική επεξεργασία.



#### 4.2.2 ΛΕΙΤΟΥΡΓΙΑ ΔΕΥΤΕΡΕΥΟΝΤΩΝ ΕΠΙΚΟΙΝΩΝΙΩΝ

Η δεύτερη λειτουργία περιλαμβάνει την επικοινωνία σχεδόν όλων των κουμπιών με το κεντρικό αντικείμενο PARENT. Σε αυτήν τη λειτουργία τα 4 χρωματιστά κουμπιά, το κουμπί HARD RESET, το κουμπί START/RESET και το κουμπί CONNECTOR επικοινωνούν με το κεντρικό αντικείμενο PARENT αλλά και μεταξύ τους, χρησιμοποιώντας το κανάλι επικοινωνίας Channel B (-5243210).

Σε αυτή τη λειτουργία μεταδίδονται μηνύματα που παράγονται από την αλληλεπίδραση του χρήστη με το «περιβάλλον χρήστη». Αυτά τα μηνύματα περιλαμβάνουν μηνύματα SOFT RESET και HARD RESET, το μήνυμα START καθώς και τα μηνύματα επιλογής (touch) των τεσσάρων χρωματιστών κουμπιών.

Αρχικά, τα τέσσερα χρωματιστά κουμπιά RED, GREEN, BLUE και YELLOW πέρα από την προβολή της χρωματικής ακολουθίας, έχουν τη δυνατότητα να δεχτούν τα δεδομένα εισόδου του χρήστη μέσω της επιλογής Touch του χρήστη σε κάθε κουμπί. Έτσι, με το πάτημα κάθε κουμπιού από τον χρήστη, το κουμπί αποστέλλει στο κεντρικό αντικείμενο PARENT το όνομα του. Άρα, άμα πατηθεί το κουμπί GREEN θα αποσταλεί σα μήνυμα στο κανάλι B το κείμενο «GREEN». Σημειώνεται ότι δε θα αποσταλεί μόνο το αρχικό γράμμα «G» του χρώματος καθώς η επεξεργασία της συγκεκριμένης εντολής γίνεται από το κεντρικό αντικείμενο PARENT το οποίο διατηρεί αποθηκευμένη τη λίστα των διαθέσιμων χρωμάτων, και όχι στον WEB SERVER ο οποίος δεν έχει δυνατότητα επεξεργασίας των δεδομένων με αποστολή ενός χρώματος τη φορά. Επίσης πρέπει να τονιστεί πως με το πάτημα ενός από τα συγκεκριμένα χρωματιστά κουμπιά, το εκάστοτε κουμπί δεν μεταβαίνει στην ειδική κατάσταση (state) στην οποία αναβοσβήνει. Επιλέχθηκε αυτή η προσέγγιση καθώς παρατηρήθηκε επιπλέον καθυστέρηση με τη χρήση της συγκεκριμένης κατάστασης. Η συγκεκριμένη παρατήρηση δεν επηρεάζει τις υπόλοιπες περιπτώσεις όπου γίνεται κλήση της κατάστασης, καθώς σε εκείνες τις περιπτώσεις δεν υπάρχει άμεση αλληλεπίδραση του χρήστη με το αντικείμενο στο Second Life. Άρα, όταν ο χρήστης χρησιμοποιεί το παιχνίδι μόνο εντός Second Life, τα γραφικά και τα animation των διαφόρων κουμπιών πρέπει να είναι όσο το δυνατόν πιο γρήγορα αποκριτικά ή και να απουσιάζουν τελείως, όπως στη συγκεκριμένη περίπτωση. Στις περιπτώσεις που το παιχνίδι δε διαδραματίζεται εξ ολοκλήρου στο Second Life, άσχετα αν η εκτέλεση του λαμβάνει χώρα πάντα εντός αυτού, τα γραφικά και τα animation δεν επηρεάζουν κάπου την εμπειρία παιχνιδιού κι έτσι συμπεριλήφθηκαν ή δε δέχτηκαν ειδική μεταχείριση εν αντιθέσει με πριν.

Επίσης, επικοινωνία με το κεντρικό αντικείμενο PARENT μέσω του καναλιού channel B πραγματοποιεί το κουμπί START/RESET. Το συγκεκριμένο κουμπί διεκπεραιώνει δύο διαφορετικές λειτουργίες ενώ και το ίδιο το κουμπί κατέχει δύο διαφορετικές εμφανίσεις και φύσεις. Το κουμπί εναλλάσσει λειτουργία και εμφάνιση ανάλογα με το αν έχει ξεκινήσει ήδη το παιχνίδι ή αν έχει τερματιστεί.

Συγκεκριμένα, αυτή η λογική συνθήκη ανάγεται στο εάν έχει πατήσει ήδη μια φορά το κουμπί ο χρήστης ή όχι. Επομένως, εάν προβάλλεται το κουμπί START τότε σημαίνει ότι δεν έχει πατηθεί το κουμπί από τον χρήστη, τουλάχιστον όχι πριν την ολοκλήρωση του προηγούμενου παιχνιδιού. Με το πάτημα του κουμπιού τώρα, η όψη του κουμπιού αλλάζει και εμφανίζεται αυτή του RESET. Έτσι, αν πατηθεί ακόμα μια φορά το κουμπί, η όψη του επανέρχεται στην αρχική όψη αυτή του START. Η εναλλαγή των όψεων και της λειτουργίας ελέγχεται χάρη στη μεταβλητή `stateofSTART` η οποία λαμβάνει την τιμή 0 όταν είναι σε κατάσταση ηρεμίας και προβάλλει τη λέξη START ενώ γίνεται 1 όταν πατηθεί το κουμπί START. Με την επαναφορά του προγράμματος στην αρχική του κατάσταση, αυτή η μεταβλητή επαναρχικοποιείται και αποκτάει ξανά την τιμή 0.

Η όψη του κουμπιού START με το που πατηθεί, πέρα από τη μετάλλαξη που υπόκειται σε κουμπί RESET, στέλνει το μήνυμα «START» στο κεντρικό αντικείμενο PARENT. Εκεί, μέσω μιας δομής ελέγχου IF επιτρέπεται να ξεκινήσει η παραγωγή της τυχαίας ακολουθίας χρωμάτων. Εάν δεν έχει πατηθεί το κουμπί START τόσο η χρωματική ακολουθία όσο και τα 4 χρωματιστά κουμπιά δε μπορούν να λειτουργήσουν.

Η εναλλακτική όψη η οποία εμφανίζει τη λέξη RESET εμφανίζεται με το που πατηθεί προηγουμένως το κουμπί START. Έτσι, το κουμπί RESET είναι διαθέσιμο καθ' όλη τη διάρκεια παιχνιδιού ενώ επιστρέφει στην αρχική του κατάσταση με το που πατηθεί ή όταν ο χρήστης χάσει στο παιχνίδι. Όταν πατηθεί το κουμπί RESET τότε μεταδίδει στο κανάλι Channel B το μήνυμα «RESET» το οποίο δέχεται μόνο το κεντρικό αντικείμενο PARENT. Αυτό έχει σαν αποτέλεσμα το κεντρικό αντικείμενο PARENT να επαναφέρεται (Reset). Ταυτόχρονα επαναφέρεται και το κουμπί START/RESET, προβάλλοντας τη λέξη START ξανά. Αυτή η διαδικασία φέρει την ονομασία SOFT RESET ώστε να ξεχωρίζει από τη διαδικασία HARD RESET που επιφέρει επιπλέον επιπτώσεις και σε άλλα αντικείμενα.

Το HARD RESET κουμπί οπότε, είναι το τελευταίο κουμπί που κάνει χρήση του καναλιού επικοινωνίας Channel B. Μέσω αυτού, όταν πατηθεί, αποστέλλει το μήνυμα «HARD\_RESET» σε πολλά αντικείμενα του project στο Second Life. Αυτά περιλαμβάνουν τα αντικείμενα PARENT, CONNECTOR και START/RESET. Στο κεντρικό αντικείμενο PARENT έχει ίδια χρήση με το προηγούμενο SOFT RESET, δηλαδή επαναφέρει τον κώδικα του συγκεκριμένου αντικειμένου. Αυτή η επαναφορά σημαίνει εμμέσως τη διακοπή του παιχνιδιού και την αρχικοποίησή του, οπότε επανέρχεται και ο κώδικας στο κουμπί START/RESET με το μήνυμα του HARD RESET. Τέλος, το συγκεκριμένο μήνυμα γίνεται αντιληπτό και από το κουμπί CONNECTOR το οποίο με αυτόν τον τρόπο αποδεσμεύει URL που είχε ήδη παράγει για την HTTP επικοινωνία.

Με αυτόν τον τρόπο δε γίνεται επαναφορά του script του κουμπιού CONNECTOR καθώς δεν υπάρχει ανάγκη για αυτό. Ακόμη, το κουμπί CONNECTOR θα αναπαράγει καινούργιο URL αν πατηθεί αυτό το

κουμπί. Σε πιθανή δεύτερη συνεχόμενη επιλογή του κουμπιού CONNECTOR χωρίς να έχει μεσολαβήσει HARD RESET τότε απλά προβάλλεται το προηγούμενος παραχθέν URL και δεν παράγεται καινούργιο με γνώμονα την οικονομία πόρων. Η αποδέσμευση των URL δεν αποτελεί αμελητέο σκοπό καθώς τα URL καταλαμβάνουν φυσική υπόσταση στα region του Second Life και καταναλώνουν ψηφιακούς πόρους οι οποίοι δεν είναι ανεξάντλητοι αλλά σχετικά πολύτιμοι.

#### 4.2.3 ΛΕΙΤΟΥΡΓΙΑ ΜΕΤΑΦΟΡΑΣ ΔΕΔΟΜΕΝΩΝ ΑΠΟ ΚΑΙ ΠΡΟΣ ΤΗΝ HTTP ΣΥΝΔΕΣΗ

Η τελευταία μορφή λειτουργία επικοινωνίας η οποία διεκπεραιώνεται εντός του Second Life περιλαμβάνει τη διαχείριση και μεταφορά των δεδομένων που είτε προέρχονται από τον Web Server είτε αποστέλλονται σε αυτόν. Ωστόσο αυτή η μορφή επικοινωνίας σε καμία περίπτωση δεν καλύπτει την ολική HTTP επικοινωνία με τον Web Server αλλά θα πρέπει να θεωρείται ως τη διαδικασία επεξεργασίας των δεδομένων πριν αποσταλούν στο διαδίκτυο ή αφού έχουν καταφτάσει από εκεί. Το κανάλι επικοινωνίας που χρησιμοποιείται είναι το Channel C (-5154789) και περιλαμβάνει την επικοινωνία μεταξύ των αντικειμένων PARENT, CONNECTOR και START/RESET. Προφανώς και δεδομένα που καταφτάνουν στο κεντρικό αντικείμενο PARENT επηρεάζουν και τα αντικείμενα RED, GREEN, BLUE και YELLOW. Όμως, το project έχει σχεδιαστεί με τέτοιο τρόπο ώστε οι λειτουργίες προβολής ενός χρώματος σε κάθε κουμπί να είναι ανεξάρτητες με τον τρόπο παιχνιδιού και να δουλεύουν με τον ίδιο τρόπο από όπου κι αν αλληλεπιδράει ο χρήστης.

Επομένως, κατά τη μεταφορά δεδομένων από τον Web Server προς τα αντικείμενα του Second Life, το αντικείμενο CONNECTOR μέσω του παραχθέντος URL, εντοπίζει τα παραγόμενα headers που δημιουργούνται στην HTML σελίδα και μέσω αυτού κατηγοριοποιεί τα μηνύματα σύμφωνα με τα 4 χρώματα «RED», «GREEN», «BLUE» και «YELLOW» αλλά και του μηνύματος «START».

Στο αντικείμενο START/RESET, το μήνυμα «START» προκαλεί την εναλλαγή της όψης του αντικειμένου στην όψη RESET. Παρόλα αυτά δεν προκαλεί νέα αποστολή του μηνύματος στο αντικείμενο PARENT καθώς αυτό θα προκαλούσε επιπλέον καθυστερήσεις αφού το μήνυμα θα έπρεπε να ληφθεί από το START/RESET και μετά να ξανααποσταλεί στο κεντρικό αντικείμενο του PARENT. Για αυτόν τον λόγο το μήνυμα «START» αποστέλλεται από τον CONNECTOR τόσο στο START/RESET όσο και στο αντικείμενο PARENT.

Στο αντικείμενο PARENT εάν φτάσει ένα μήνυμα χρώματος, προφανώς εφόσον έχει μεσολαβήσει πάτημα του κουμπιού START είτε στην HTML σελίδα είτε εντός του Second Life, αυτό αποστέλλει την πληροφορία σε ένα από τα 4 χρωματιστά κουμπιά επιλογής. Σε αυτήν την περίπτωση δεν υπάρχει λόγος απευθείας αποστολής της πληροφορίας στα κουμπιά επιλογής αφού έτσι κι αλλιώς θα υπάρξει

επεξεργασία των δεδομένων στο αντικείμενο PARENT. Έτσι, ο έλεγχος ταυτοποίησης των δεδομένων που εισέρχονται από την HTTP επικοινωνία σε σχέση με την ήδη παραχθείσα χρωματική ακολουθία, γίνεται στο κεντρικό αντικείμενο PARENT.

Από το συγκεκριμένο αντικείμενο, όμως, πάλι, ξεκινάει και η ροή δεδομένων προς την αντίθετη κατεύθυνση με προορισμό τον Web Server. Εκεί, δημιουργούνται δύο κατηγορίες δεδομένων που αποστέλλονται. Η πρώτη αφορά τα δεδομένα της χρωματικής ακολουθίας που έχει παραχθεί, τα οποία έχουν τη μορφή «RRYRBG» με το κάθε γράμμα να υποδηλώνει ένα από τα τέσσερα διαθέσιμα χρώματα. Έτσι, αυτή η ακολουθία αποστέλλεται με τη μορφή μηνύματος στο αντικείμενο CONNECTOR από το οποίο διαβιβάζεται στον Web Server.

Εδώ θα πρέπει να αναφερθεί πως ο κώδικας του αντικειμένου PARENT, χρησιμοποιώντας τη μεταβλητή `waitingForInput` ορίζει την κατάσταση στην οποία βρίσκεται το παιχνίδι και ως αποτέλεσμα τη συνάρτηση που θα εκτελεσθεί στη συνέχεια. Όταν η μεταβλητή `waitingForInput` έχει τιμή `FALSE` τότε το αντικείμενο εκτελεί τη συνάρτηση `pattern_SHOUT` στην οποία παράγεται η χρωματική ακολουθία. Εάν η μεταβλητή `waitingForInput` έχει τιμή `TRUE` τότε σημαίνει ότι έχει ήδη παράγει μία χρωματική ακολουθία και τώρα βρίσκεται σε αναμονή δεδομένων εισόδου από τον χρήστη, είτε έρθουν αυτά μέσω Web Server είτε εντός του `Second Life`. Έτσι, με την πρώτη εμφάνιση των δεδομένων εισόδου, εκτελείται η συνάρτηση `pattern_INPUT` κατά την οποία τα δεδομένα εισόδου συγκρίνονται με την ακολουθία χρωμάτων και ελέγχεται κατά πόσο είναι ίδιες. Αν δεν είναι ίδιες το παιχνίδι τερματίζεται. Στην συγκεκριμένη συνάρτηση όμως λαμβάνει χώρα και ο υπολογισμός του σκορ του χρήστη. Υπενθυμίζεται ότι μόνο ο υπολογισμός του σκορ συμβαίνει και στο `Second Life` και στον Web Server.

Η άλλη κατηγορία αφορά την περίπτωση που ο χρήστης, χρησιμοποιώντας το `Second Life` μέσω του Web Server, κάνει λάθος και χάσει στο παιχνίδι. Τότε, η ένδειξη ότι υπήρξε λάθος προκύπτει στο αντικείμενο PARENT με τη μορφή του χαρακτήρα «Ο» υποδηλώνοντας το `GAME OVER`. Αυτό το μήνυμα θα αποσταλεί πάλι στον CONNECTOR ο οποίος με τη σειρά του θα το μεταβιβάσει στον Web Server. Τέλος, ο χαρακτήρας «Ο» αποστέλλεται και στο κουμπί `START/RESET` υποδεικνύοντας του το τέλος του παιχνιδιού και άρα την επαναφορά του στην αρχική του κατάσταση και στην όψη `START`.

#### 4.2.4 TO ANTIKEIMENO INFO

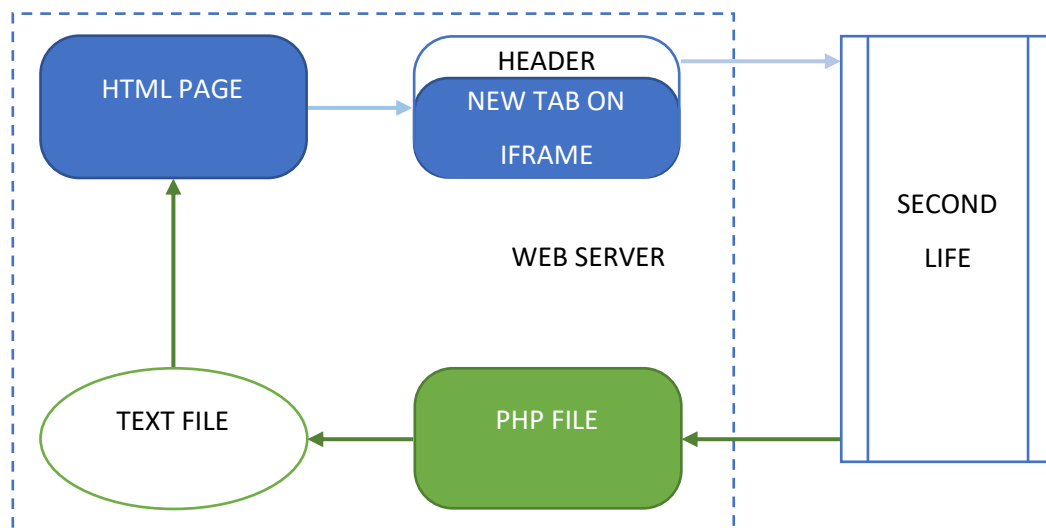
Το κουμπί-αντικείμενο INFO αποτελεί μια ξεχωριστή κατηγορία από μόνο του καθώς δεν επικοινωνεί με κάποιον τρόπο με τα υπόλοιπα αντικείμενα του `Second Life` αλλά και ούτε με τον Web Server. Παρόλα αυτά, ο τρόπος ο οποίος εμφανίζει τα παράθυρα με πληροφορίες για τον χρήστη αποτελούν

μια μορφή επικοινωνίας ελαφρώς πιο πολύπλοκη από τα βασικά μηνύματα χρήστη στο μηδενικό κανάλι (παράθυρο επικοινωνίας, chat) που δημιουργούν τα υπόλοιπα κουμπιά. Αυτό συμβαίνει διότι το συγκεκριμένο κουμπί χρησιμοποιεί την εντολή `!!Dialog()` η οποία εμφανίζει τα παράθυρα με τις πληροφορίες αλλά προβάλλει ταυτόχρονα και μια λίστα από κουμπιά. Ωστόσο, αυτά τα παράθυρα εμφανίζονται στον χρήστη μέσω επικοινωνίας στο κανάλι 1000 το οποίο ορίζεται εντός του script. Άρα, υπάρχει μεταφορά πληροφορίας στον χρήστη μέσω του καναλιού 1000 το οποίο δεν εμφανίζεται στο κεντρικό παράθυρο του chat (με αριθμό καναλιού 0) αλλά σε ένα τεχνητό παράθυρο. Όμως, όλες οι αλληλεπιδράσεις εντός αυτού του νέου παραθύρου μεταφέρονται στο κανάλι 1000 το οποίο όμως κανένα άλλο αντικείμενο δεν το χρησιμοποιεί στο Project.

#### 4.3 WEB SERVER

Στον Web Server εξειδικευμένα αρχεία HTML και PHP έχουν την δυνατότητα να παράγουν δεδομένα για το Second Life αλλά και να μπορούν να δέχονται και να αναπαράγουν δεδομένα από το Second Life.

Συγκεκριμένα, μια HTML σελίδα αποτελεί το κέντρο διαχείρισης των δεδομένων είτε αυτά προέρχονται από το Second Life είτε αποστέλλονται εκεί. Η HTML σελίδα χρησιμοποιεί ταυτόχρονα μία σελίδα PHP ενώ κατέχει εμφωλευμένα αρχεία κώδικα CSS και Javascript. Η διάρθρωση του Web Server φαίνεται στο Γράφημα 4.3.



Γράφημα 4.3. Απεικόνιση της επικοινωνίας εντός του Web Server.

Υπενθυμίζεται ότι τα αρχεία που χρησιμοποιούνται στον Διαδικτυακό Server, βρίσκονται σε δωρεάν Host της υπηρεσίας «<https://www.000webhost.com>» η οποία προσφέρει δωρεάν θέση σε Server.

#### 4.3.1 ΔΕΔΟΜΕΝΑ ΠΡΟΣ SECOND LIFE

Για να επιτευχθεί η αποστολή δεδομένων προς το Second Life γίνεται χρήση αποκλειστικά της HTML σελίδας και όχι του PHP αρχείου τα οποία βρίσκονται στον Web Server. Προφανώς για να μπορέσει να υπάρξει επικοινωνία με το Second Life, απαιτείται σταθερή σύνδεση στο διαδίκτυο, τόσο του Second Life όσο και του Web Server. Ακόμη απαιτείται το παιχνίδι Simon Says να είναι ενεργό και να εκτελείται στο Second Life ενώ επίσης θα πρέπει να υπάρχει διαθέσιμο URL το οποίο θα χρησιμοποιηθεί για την HTTP επικοινωνία.

##### 4.3.1.1 ΠΑΡΑΓΩΓΗ ΣΤΗ ΣΕΛΙΔΑ HTML

Ο χρήστης, όπως προαναφέρθηκε, έχει τη δυνατότητα να παίξει το παιχνίδι Simon Says μέσω ενός Web Browser. Το User Interface της σελίδας HTML που χρησιμοποιείται, προσφέρει τα 4 χρωματιστά κουμπιά που αποτελούν και τα κουμπιά επιλογής, καθώς και ένα κουμπί «START GAME». Κάθε φορά που ο χρήστης πατάει ένα από αυτά τα 5 κουμπιά, εντός της σελίδας HTML εκτελείται μια συνάρτηση JavaScript η οποία είναι εμφωλευμένη στον κώδικα HTML. Στην πραγματικότητα, τα 4 χρωματιστά κουμπιά εκτελούν τη συνάρτηση SimonGame() με το συμβάν onclick, όπως φαίνεται στην Εικόνα 4.1, ενώ το κουμπί «START GAME» εκτελεί τη συνάρτηση StartGame(), όπως φαίνεται στην Εικόνα 4.2.

```
<div id="PRESSR"class=REDButton onclick="SimonGame('RED')"></div>  
<div id="PRESSG"class=GREENButton onclick="SimonGame('GREEN')"></div>
```

Εικόνα 4.1. Εκτέλεση της συνάρτησης SimonGame με το κλικάρισμα του κόκκινου ή του πράσινου κουμπιού.

```
<button id="button1"class="button1" onclick="StartGame()">START GAME</button>
```

Εικόνα 4.2. Εκτέλεση της συνάρτησης StartGame με το κλικάρισμα του κουμπιού "START GAME".

Στην Εικόνα 4.1 γίνεται εμφανές ότι η εκτέλεση της συνάρτησης SimonGame() με το πάτημα του κόκκινου κουμπιού (Το id «PRESSR» αντιστοιχεί στο κόκκινο κουμπί, το «PRESSG» στο πράσινο, το «PRESSB» στο μπλε και τέλος το «PRESSY» στο κίτρινο), γίνεται με δεδομένο εισόδου της συνάρτησης, το κείμενο «RED». Αντιθέτως, η συνάρτηση StartGame() εκτελείται χωρίς δεδομένα εισόδου. Αυτό συμβαίνει διότι η συνάρτηση SimonGame() είναι κοινή και για τα τέσσερα έγχρωμα κουμπιά και λειτουργεί με τον ίδιο τρόπο και για τα τέσσερα. Η συνάρτηση StartGame() όμως λειτουργεί αποκλειστικά με το κουμπί «START GAME», οπότε δεν απαιτούνται δεδομένα εισόδου που να ξεκαθαρίζουν ποιο κουμπί πατήθηκε.

#### 4.3.1.2 Η ΕΝΤΟΛΗ WINDOW.OPEN

Ωστόσο, ακόμη μια ειδοποιός διαφορά μεταξύ των δύο συναρτήσεων, είναι ότι η συνάρτηση StartGame και η συνάρτηση SimonGame έχουν διαφορετικές χρήσεις ενώ μόνο ένα κομμάτι και των δύο συναρτήσεων είναι κοινό κι αυτό είναι η χρησιμοποίηση της εντολής window.open όπως φαίνεται και από τις εικόνες Εικόνα 4.3 και Εικόνα 4.4.

```
function SimonGame(str){
    step++;
    side_step = step;
    window.open(slurl + "?" + str, "win1");
    side_score = DataLength;
    loadDoc();
    if(step > score){
        score++;
    }
    if (step == DataLength){
        document.getElementById("button2").innerHTML = step;
        step=0;
    }
}
```

Εικόνα 4.3. Η συνάρτηση SimonGame.

```
function StartGame(){
    window.open(slurl + "?" + "START", "win1");//We send "RED" because it doesn't matter what color is sent. SL code begins
    // with every color as an input data
    //-----Animation for the START GAME button(button1) and SCORE button(button2)-----//
    document.getElementById('button1').style.transform = 'translateY(-50%)';
    document.getElementById('button1').style.visibility = 'hidden';
    setTimeout(function() {document.getElementById('button2').style.transform = 'translateY(40%)';}, 300);
    setTimeout(function() {document.getElementById('button2').style.visibility = 'visible';}, 300);
    document.getElementById("button2").innerHTML = "";
    setTimeout(function() {loadDoc();},2000);// Delay of 2s just to be sure we have got the php data sent to the html page
}
```

Εικόνα 4.4. Η συνάρτηση StartGame.

Η εντολή window.open() δημιουργεί καινούργιο παράθυρο στον Web Browser που χρησιμοποιείται για την αποστολή δεδομένων μέσω HTTP Request, ενώ στο εσωτερικό της παρένθεσης ορίζεται η διεύθυνση που θα φέρει αυτό το νέο παράθυρο. Στην περίπτωση του συγκεκριμένου project, χρειάζεται να ανοίξει ένα νέο παράθυρο ονόματος (id) «win1» το οποίο θα φέρει σε διεύθυνση μια σελίδα αποτελούμενη από τρία μέρη. Το πρώτο μέρος αυτής της διεύθυνσης θα είναι το URL που έχει παράγει το Second Life. Το συγκεκριμένο μέρος αποτελεί το περιεχόμενο της μεταβλητής slurl η οποία ορίζεται σύμφωνα με την Εικόνα 4.5.

```
var url = location.href;
var slurl = (url.slice(url.indexOf('?') + 1, url.length));
```

Εικόνα 4.5. Η εξαγωγή της διεύθυνσης του URL του Second Life.

Έτσι, η μεταβλητή `slurl` αποκτά τη διεύθυνση του URL του Second Life, αποκόπτοντας το κομμάτι της διεύθυνσης της κεντρικής σελίδας μετά τον χαρακτήρα «?». Υπενθυμίζεται ότι η κεντρική διεύθυνση που χρησιμοποιείται στον διαδικτυακό φυλλομετρητή (Web Browser) προκύπτει από τη σάρωση του QR κωδικού από το Second Life ή από τη χειροκίνητη τοποθέτηση του URL μετά τη διεύθυνση της HTML σελίδας στην οποία έχει τοποθετηθεί και ο χαρακτήρας «?». Το ίδιο URL θέλει να χρησιμοποιήσει και η μεταβλητή `slurl` και για αυτόν το λόγο αντιστρέφει αυτόματα την προηγούμενη διαδικασία.

Έτσι, επιστρέφοντας στα μέρη της νέας διεύθυνσης του νέου παραθύρου, στο δεύτερο μέρος της διεύθυνσης θα υπάρξει ο χαρακτήρας «?» ο οποίος θα χρησιμοποιείται από το Second Life για τον διαχωρισμό των συγκεκριμένων μερών της διεύθυνσης και την εξαγωγή πληροφορίας από τα δεδομένα.

Τέλος, στο τρίτο μέρος της διεύθυνσης τοποθετείται το κείμενο το οποίο περιλαμβάνει την επιλογή του χρήστη. Αυτό το κείμενο ουσιαστικά αποτελεί και τη μόνη πληροφορία που δέχεται εν τέλει το παιχνίδι Simon Says στο Second Life. Το συγκεκριμένο πεδίο μπορεί να λαμβάνει τις τιμές «RED», «GREEN», «BLUE», «YELLOW» και «START» τις οποίες δέχεται και φιλτράρει το αντικείμενο CONNECTOR του Simon Says στο Second Life.

#### 4.3.1.3 ΤΟ ΠΕΔΙΟ IFRAME

Κανονικά, με το άνοιγμα νέου παραθύρου στον διαδικτυακό φυλλομετρητή κάθε φορά που ο χρήστης πατάει ένα κουμπί, το παιχνίδι Simon Says στον φυλλομετρητή θα ήταν αδύνατο. Διότι τα πολλαπλά παράθυρα που θα εμφανιζόντουσαν αλλά και η εναλλαγή στη νέα καρτέλα θα δημιουργούσε ασύμφορο περιβάλλον χρήστη και θα εξαφάνιζε τη λειτουργικότητα του εγχειρήματος.

Σε αυτό το πρόβλημα δίνει λύση το πεδίο `iframe` το οποίο αποτελεί εμφωλευμένη σελίδα HTML εντός άλλης σελίδας HTML .

```
<iframe name="win1" src="" width=0 height=0 frameborder=0></iframe>
```

Εικόνα 4.6. Το πεδίο `iframe`.

Το πεδίο `iframe` χρησιμοποιεί το νέο παράθυρο «win1» που δημιουργείται από την εντολή `window.open()` και του προσάγει μηδενικό πλάτος και μηδενικό ύψος, κάνοντας το ουσιαστικά αόρατο. Με αυτόν τον τρόπο αποφεύγεται η δημιουργία υπαρκτού νέου παραθύρου στον φυλλομετρητή και οι αρνητικές επιπτώσεις που προκαλεί αυτό. Παρόλα αυτά, νέο παράθυρο δημιουργείται, και μπορεί να διαπεραστεί από το Second Life, όμως απλά δε μπορεί οπτικά να ανιχνευθεί.



#### 4.3.1.4 ΑΠΟΛΑΒΗ ΔΕΔΟΜΕΝΩΝ ΣΤΟ SECOND LIFE

Από τη στιγμή που το παραχθέν URL συνοδευτεί με τον χαρακτήρα «?» και στη συνέχεια τα δεδομένα του χρήστη από τον φυλλομετρητή, προκύπτει μια ζήτηση http request η οποία σκανδαλίζει (trigger) το συμβάν http\_request (event) στο αντικείμενο CONNECTOR στο Second Life. Αυτός ο σκανδαλισμός δίνει το έναυσμα στον CONNECTOR να επεξεργαστεί την επικεφαλίδα που έχει παραχθεί στον φυλλομετρητή και μέσω της εντολής `llGetHTTPHeader(id, "x-query-string")` και της `llSetContentType(id, CONTENT_TYPE_HTML)` να αποκομίσει τα δεδομένα του χρήστη που υπάρχουν στην επικεφαλίδα του νέου παραθύρου στο πεδίο `iframe`. Η εντολή `llGetHTTPHeader(id, "x-query-string")` αποκομίζει τα δεδομένα που βρίσκονται στην επικεφαλίδα ενώ το πεδίο «x-query-string» ορίζει πως τα δεδομένα θα αποκτηθούν μετά τον χαρακτήρα «?» στην επικεφαλίδα. Ακόμα, η εντολή `llSetContentType(id, CONTENT_TYPE_HTML)` είναι απαραίτητη καθώς ορίζει πως τα δεδομένα που αποκτήθηκαν μέσω της επικεφαλίδας HTTP, προέρχονται από σελίδα HTML. Το πεδίο «id» και στις δύο εντολές ορίζει την ταυτότητα του κλειδιού της ζήτησης HTTP (HTTP request key)[5].

Να σημειωθεί πως η μέθοδος της HTTP request είναι η «GET» καθώς το Second Life σκοπεύει να αποκομίσει πληροφορία από τον Server[6]. Ο έλεγχος για τη μέθοδο είναι απαραίτητος για τον κώδικα του CONNECTOR στο Second Life και τον πετυχαίνει μέσω μιας δομής ελέγχου ροής `if`, όπως φαίνεται στην Εικόνα 4.7.

```
// If a homepage server request or a URL request comes in
http_request(key id, string method, string body)
{
    // If the request is an address
    if (method == URL_REQUEST_GRANTED)
    {
        // Show your new homepage address in chat window.
        url=body;
        //llOwnerSay("URL: " + url);
    }
    // When a GET request arrives
    else if (method == "GET")
    {
        //Data from our answer is contained in the HTTP header of our page
        //HTML page generates a new window(invisible window) in which's header, our answer it's written
        string result = llGetHTTPHeader(id,"x-query-string");//Data from our answer is being kept in the variable "result"
        llSetContentType(id, CONTENT_TYPE_HTML);
    }
}
```

Εικόνα 4.7. Το συμβάν `http_request` και ο έλεγχος μεθόδου.

Επίσης, όπως φαίνεται, γίνεται ένας τυπικός έλεγχος επιβεβαίωσης της αποδοχής ζήτησης URL η οποία προσφέρει την επιλογή εμφάνισης του URL στον χρήστη. Ωστόσο, η διαχείριση των δεδομένων του χρήστη από τον φυλλομετρητή χρησιμοποιώντας HTTP επικοινωνία, συμβαίνει εντός της δομής ελέγχου `if` με τη μέθοδο να είναι «GET» (`method == "GET"`). Εκεί, χρησιμοποιώντας τις συναρτήσεις που προαναφέρθηκαν, τα δεδομένα του χρήστη αποθηκεύονται στη μεταβλητή `result`. Σε εκείνη τη στιγμή, τα δεδομένα του χρήστη είναι τύπου `string` και είναι εύκολα επεξεργάσιμα.

Στη συνέχεια, γίνεται έλεγχος της μεταβλητής `result` η οποία φέρει και τα δεδομένα του χρήστη. Αν η μεταβλητή είναι κενή, τότε θεωρείται πως το URL δεν έχει ακόμα χρησιμοποιηθεί, και για αυτό

ανοίγει ένα παράθυρο του εσωτερικού φυλλομετρητή του Second Life, εντός του οποίου υπάρχουν οι οδηγίες προς τον χρήστη να σαρώσει τον QR κωδικό ενώ επίσης η διεύθυνση σε αυτό το παράθυρο είναι το παραχθέν URL το οποίο ο χρήστης μπορεί να αντιγράψει και να το χρησιμοποιήσει ποικιλοτρόπως.

#### 4.3.1.5 QR CODE

Ο QR κωδικός δημιουργείται δυναμικά, με τη μορφή της Εικόνας 3.32, χρησιμοποιώντας τη διεύθυνση της υπηρεσίας δημιουργίας QR κωδικών «<https://api.qrserver.com/v1/create-qr-code/?data=>». Αυτή η διεύθυνση, χρησιμοποιούμενη μαζί με την διεύθυνση που είναι επιθυμητό να κωδικοποιηθεί, δημιουργεί ένα μοναδικό QR κωδικό. Εντός του Second Life προβάλλεται ως εικόνα κάτω από το γραμμοσκιασμένο κείμενο που υποδεικνύει τον χρήστη να σαρώσει τον κωδικό. Σημαντικό εδώ είναι να παρατηρηθεί πως επειδή το συγκεκριμένο κείμενο αλλά και ο QR κωδικός εμφανίζονται στον εσωτερικό φυλλομετρητή του Second Life, οι παραπάνω εντολές είναι γραμμένες σε γλώσσα HTML, εντός της εντολής `llHttpResponse()` όπως φαίνεται στην Εικόνα 4.8, όπου το νούμερο 200 αντιστοιχεί στον κωδικό κατάστασης της HTTP επικοινωνίας «OK» που ορίζει ότι η επικοινωνία ήταν επιτυχής[7]. Η μεταβλητή `baseurl` έχει τη διεύθυνση της HTML σελίδας και το «?»

```
if(result=="")
//Proper URL is our baseurl plus the url which is generated from the "llRequestURL()"
//baseurl+url navigates to the window of our custom HTML page which is built for this specific HTTP communication ONLY
//If the URL is released then the wanted HTML page will be changed but it will be generated automatically again
llHttpResponse(id,200,header +
"<br>Take a QR code with your smartphone.<br>"
"<img src='https://api.qrserver.com/v1/create-qr-code/?data=" + baseurl + url + "'/>");
```

Εικόνα 4.8. Browser του Second Life με τον παραχθέν QR κωδικό.

#### 4.3.1.6 ΔΙΑΧΕΙΡΙΣΗ ΤΩΝ ΔΕΔΟΜΕΝΩΝ ΑΠΟ ΤΟ SECOND LIFE

Εφόσον τα δεδομένα του χρήστη είναι αποθηκευμένα στη μεταβλητή `result` και εφόσον είναι γνωστές οι πιθανές τιμές που μπορεί να λάβει αυτή η μεταβλητή, δεν αποτελεί πρόβλημα η επεξεργασία των τιμών της. Έτσι, με πέντε διαφορετικές δομές ελέγχου `if`, πάντα εμφωλευμένες εντός της δομής ελέγχου για τη μέθοδο HTTP GET, γίνεται κατηγοριοποίηση των περιπτώσεων και αναλόγως της τιμής, αποστέλλεται στο κεντρικό αντικείμενο PARENT το αντίστοιχο μήνυμα μέσω του καναλιού C στο Second Life. Οι πιθανές περιπτώσεις είναι οι «RED», «GREEN», «BLUE», «YELLOW» και «START» όπως φαίνεται στην Εικόνα 4.9.

```
if(result=="RED")//If our choice is "RED", the word "RED" is sent to "parent_v8"
llSay(-5154789,"RED");
if(result=="BLUE")//If our choice is "BLUE", the word "BLUE" is sent to "parent_v8"
llSay(-5154789,"BLUE");
if(result=="GREEN")//If our choice is "GREEN", the word "GREEN" is sent to "parent_v8"
llSay(-5154789,"GREEN");
if(result=="YELLOW")//If our choice is "YELLOW", the word "YELLOW" is sent to "parent_v8"
llSay(-5154789,"YELLOW");
if(result=="START")//If we want to start the game and pressed "START", the word "START" is sent to "parent_v8"
llSay(-5154789,"START");
```

Εικόνα 4.9. Η κατηγοριοποίηση των δεδομένων.

#### 4.3.2 ΔΕΔΟΜΕΝΑ ΑΠΟ SECOND LIFE

Εάν ο Web Server μπορούσε μόνο να στείλει δεδομένα στο Second Life και δεν υπήρχε επικοινωνία διπλής κατεύθυνσης, τότε ο Web Browser θα αποτελούσε ένα απλό χειριστήριο το οποίο θα μπορούσε να χρησιμοποιηθεί μόνο μπροστά από οθόνη συσκευής που εκτελούσε το Simon Says στο Second Life. Όμως, ο στόχος αυτού του εγχειρήματος αποτελούσε η διπλής κατεύθυνσης επικοινωνία μεταξύ του Second Life και μιας εξωτερικής συσκευής. Αυτό επιτεύχθηκε με τη χρησιμοποίηση PHP και AJAX εργαλείων.

Τα δεδομένα που πρέπει να διαβιβαστούν μέσω του Web Server στον χρήστη, έχουν τη μορφή κειμένου «RGRB». Αυτά τα δεδομένα αποτελούν τη ζητούμενη χρωματική ακολουθία που πρέπει να εισάγει ο χρήστης για να προχωρήσει στο παιχνίδι. Οπότε αποτελεί ζητούμενο, η παραλαβή των δεδομένων από την HTML σελίδα και η εξαγωγή του καθενός χρώματος ώστε να είναι δυνατή η αναπαραγωγή της χρωματικής ακολουθίας στον χρήστη.

##### 4.3.2.1 ΑΠΟΣΤΟΛΗ ΑΠΟ SECOND LIFE ΣΕ PHP ΑΡΧΕΙΟ

Αρχικά, το αντικείμενο CONNECTOR παραλαμβάνει τα δεδομένα από το κεντρικό αντικείμενο PARENT και τα αποθηκεύει στη μεταβλητή PHP\_DATA. Ωστόσο, η συγκεκριμένη μεταβλητή περιέχει τη διεύθυνση του PHP αρχείου που βρίσκεται στον Host Server, ενώ επιπλέον προστίθεται η κατάληξη «?test=». Με αυτόν τον τρόπο η μεταβλητή PHP\_DATA περιέχει τη διεύθυνση «http://siwonsimon.000webhostapp.com/sl.php?test=» η οποία αποτελεί το περιεχόμενο της μεταβλητής PHP\_URL και ακόμη, μετά τον χαρακτήρα «=», τοποθετείται το κείμενο με τη χρωματική ακολουθία που έχει παραχθεί. Ένα τέτοιο παράδειγμα θα μπορούσε να είναι το «http://siwonsimon.000webhostapp.com/sl.php?test=RGRB».

Επιπλέον, πέρα από τη χρωματική ακολουθία, μπορεί να αποσταλεί ο χαρακτήρας «Ο», εκ του GAME OVER, ο οποίος αποτελεί το μήνυμα λάθους προς τον χρήστη και την ένδειξη ότι ο τωρινός γύρος του παιχνιδιού τερματίζεται.

```
//Communication with "parent_v8" for gaining the wanted sequence of colors
if (channel == -5154789){
    PHP_URL= "http://siwonsimon.000webhostapp.com/sl.php?test=";
    PHP_DATA= PHP_URL + message;//Color sequence is a string in the variable "message"(e.g. "RRYBG")
    //R stands for RED, G stands for GREEN, B stands for BLUE, Y stands for YELLOW, O stands for GAME OVER
    llHTTPRequest(PHP_DATA,[],"");//Sending the wanted sequence into a php file
    //in which is been processed and saved to a TXT file
}
```

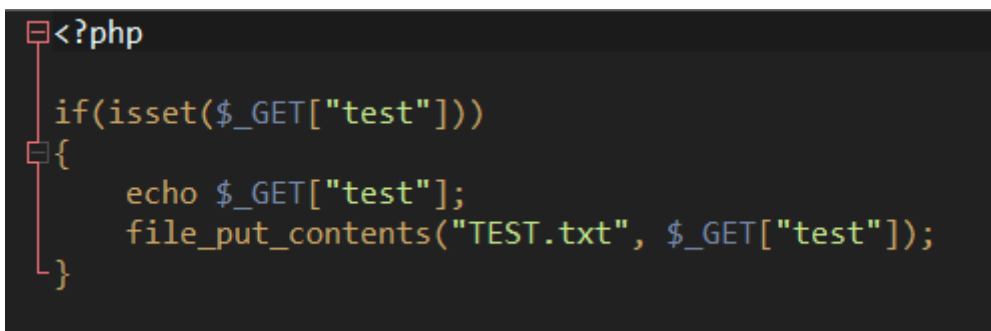
Εικόνα 4.10. Η απόκτηση των δεδομένων από το αντικείμενο PARENT.

Τα δεδομένα από το αντικείμενο PARENT μεταβιβάζονται στη μεταβλητή PHP\_DATA μέσω της μεταβλητής message χρησιμοποιώντας το συμβάν listen.

Στη συνέχεια, χρησιμοποιώντας HTTP επικοινωνία, διαπερνιέται το PHP αρχείο το οποίο χρησιμοποιεί τα δεδομένα της διεύθυνσης που είναι αποθηκευμένη στη μεταβλητή PHP\_DATA, μετά τον χαρακτήρα «?».

#### 4.3.2.2 ΕΠΕΞΕΡΓΑΣΙΑ ΣΕ PHP

Το PHP αρχείο που βρίσκεται αποθηκευμένο στον Host Server, το οποίο θα πρέπει υποχρεωτικά να βρίσκεται στον ίδιο Host με το αρχείο HTML, παραλαμβάνει τα δεδομένα τα οποία στάλθηκαν από το Second Life, τα επεξεργάζεται με μια απλή συνάρτηση η οποία χρησιμοποιεί τη μεταβλητή test και εξάγει τα δεδομένα σε ένα αρχείο κειμένου (txt).



```
<?php
if(isset($_GET["test"]))
{
    echo $_GET["test"];
    file_put_contents("TEST.txt", $_GET["test"]);
}
```

Εικόνα 4.11. Το αρχείο PHP.

Εντός του αρχείου PHP, ελέγχεται αν η μεταβλητή test έχει τιμή, ενώ αναπαράγει την τιμή της μεταβλητής και ταυτόχρονα την εγγράφει στο αρχείο TEST.txt. Κάθε φορά που εγγράφονται δεδομένα στο συγκεκριμένο αρχείο, τα παλιότερα δεδομένα διαγράφονται. Το συγκεκριμένο αρχείο κειμένου δημιουργείται αυτόματα στον Host που βρίσκεται το php αρχείο ενώ αν υπάρχει αρχείο με το ίδιο όνομα, το διαγράφει και στη θέση του δημιουργεί το προαναφερθέν.

Το αρχείο PHP επιλέχθηκε να μην τοποθετηθεί εμφωλευμένα εντός της σελίδας HTML, εν αντιθέσει με όλα τα υπόλοιπα εργαλεία που χρησιμοποιήθηκαν, καθώς η αποτελεσματικότητα και η καθυστέρηση που δημιουργούνταν, δεν ήταν οι επιθυμητές.

#### 4.3.2.3 ΠΡΟΣΠΑΘΕΙΑ ΑΝΑΓΝΩΣΗΣ ΔΕΔΟΜΕΝΩΝ ΜΕ PHP

Κατά τον σχεδιασμό της σελίδας HTML ήταν επιθυμητό η σελίδα να μην ανανεώνεται κάθε φορά που στον χρήστη έπρεπε να παρουσιαστούν νέα δεδομένα αλλά ούτε και ο χρήστης να μπει σε διαδικασία να κάνει χειροκίνητες ενέργειες ώστε να φορτωθούν τα δεδομένα στη σελίδα HTML. Ο πρώτο τρόπος που υλοποιήθηκε όριζε έναν PHP κώδικα το οποίο αποτελούσε μέρος της σελίδας HTML, που όμως χρειαζόταν να διατηρεί τη σελίδα HTML σε μόνιμη ανανέωση[8]. Με αυτόν τον τρόπο η σελίδα είχε

πρόσβαση σε νέα δεδομένα από το Second Life αυτόματα. Παρόλα αυτά, μια μόνιμη ανανέωση της σελίδας για αυτόματη φόρτιση με νέα δεδομένα, απαιτεί σταθερή σύνδεση στο διαδίκτυο χωρίς την παραμικρή ανωμαλία ενώ και ο φυλλομετρητής που έχει φορτώσει τη σελίδα θα πρέπει να ανανεώνει τη σελίδα χωρίς την παραμικρή καθυστέρηση. Κάτι τέτοιο φαντάζει ουτοπικό γιατί σε όλες τις συνδέσεις ίντερνετ υπάρχουν διακυμάνσεις ενώ όλοι οι φυλλομετρητές, αργά ή γρήγορα εμφανίζουν μικρές ανωμαλίες στην απόδοση τους. Αυτές οι ανωμαλίες δεν είναι αρκετές για να ζημιώσουν την εμπειρία χρήσης όμως αποδεικνύονται υπεραρκετές για να σταματήσει η σελίδα PHP να φορτώνει δεδομένα. Για αυτόν τον λόγο εγκαταλείφθηκε αυτή η κατεύθυνση και προτιμήθηκε η υλοποίηση με AJAX (Asynchronous JavaScript And Xml).

#### 4.3.2.4 ΧΡΗΣΗ AJAX

Η AJAX αποτελεί πακέτο τεχνικών η οποία επέτρεψε στη σελίδα HTML να φορτώνει στο παρασκήνιο, δεδομένα από το αρχείο κειμένου TEST.txt χωρίς να χρειάζεται ανανέωση της σελίδας. Αποτελεί το κυριότερο εργαλείο για την ανάγνωση των δεδομένων από το Second Life καθώς χωρίς αυτό, δε θα ήταν δυνατή η υλοποίηση του project. Για τη συγκεκριμένη εργασία επιλέχθηκε η υλοποίηση με AJAX που προσφέρει η w3schools[9] με τη χρήση του XMLHttpRequest αντικειμένου για την επικοινωνία. Σημειώνεται ότι για τη συγκεκριμένη υλοποίηση επιλέχθηκε η σύγχρονη επικοινωνία καθώς η φύση του παιχνιδιού θα προϋπέθετε πολλαπλούς επιπλέον χειριστές (handlers). Όμως, για να μπορούν τα γραφικά που δημιουργήθηκαν να λειτουργούν επαρκώς αλλά και οι κλήσεις συναρτήσεων από άλλες συναρτήσεις εντός της JavaScript να μπορούν να εκτελεστούν, επιλέχθηκε η σύγχρονη εκτέλεση. Ωστόσο, το μεγαλύτερο πρόβλημα αποτελούσαν οι κλήσεις των συναρτήσεων από άλλες συναρτήσεις. Εξαιτίας του τρόπου ανάγνωσης της χρωματικής ακολουθίας αλλά και της ανάγκης να υπάρχει διαρκής ανάγνωση για δεδομένα ώστε μήπως εμφανιστεί ο χαρακτήρας «Ο» ο οποίος θα υποδήλωνε ότι το γύρος τελείωσε, κλήσεις AJAX θα έπρεπε να συμβαίνουν διαρκώς, μετά το κάθε πάτημα κουμπιού από τον παίκτη. Αυτές οι κλήσεις, σε περίπτωση ασύγχρονης επικοινωνίας θα δυσχέραιναν τον σχεδιασμό και την εκτέλεση του κώδικα αφού θα έπρεπε να αλλάξει τελείως η φιλοσοφία πίσω από το παιχνίδι.

#### 4.3.2.5 ΑΝΑΓΝΩΣΗ ΔΕΔΟΜΕΝΩΝ ΣΤΗΝ HTML ΣΕΛΙΔΑ ΜΕ AJAX

Για να μπορεί η HTML σελίδα να φορτώνει νέα δεδομένα από το αρχείο κειμένου TEST.txt, χρησιμοποιεί το αντικείμενο XMLHttpRequest() με το οποίο ορίζει(ζητάει, request) μια καινούργια HTTP επικοινωνία η οποία ανανεώνει μόνο μέρος της σελίδας και όχι ολόκληρη τη σελίδα. Η τιμή (κλειδί, key) της HTTP επικοινωνίας αποθηκεύεται σε μια μεταβλητή xmlhttp. Στη συνέχεια,

χρησιμοποιώντας το key της επικοινωνίας, μπορεί να διαπεραστεί το αρχείο κειμένου TEST.txt με την εντολή xmlhttp.open("GET", "TEST.txt", false) ενώ με την εντολή xmlhttp.send() τα δεδομένα καταφτάνουν στον κώδικα JavaScript εντός της HTML (ουσιαστικά στην AJAX). Κατά το άνοιγμα του αρχείου κειμένου, χρησιμοποιείται η «GET» μέθοδος, ορίζεται το όνομα του αρχείου ενώ επίσης διευκρινίζεται η σύγχρονη μορφή της επικοινωνίας (false για σύγχρονη, true για ασύγχρονη).

Εδώ είναι σημαντικό να σημειωθεί πως η τεχνική AJAX, τουλάχιστον μέχρι τώρα, χρησιμοποιείται αποκλειστικά σε περιβάλλον κοινού Host (same-origin policies)[10] αν και υπάρχουν ορισμένα προγραμματιστικά εργαλεία που επιτρέπουν μερική επίλυση του συγκεκριμένου προβλήματος (πχ CORS)[11]. Αυτό σημαίνει πως δε μπορεί να επιτευχθεί επικοινωνία με AJAX εκτός του domain που χρησιμοποιεί η κεντρική σελίδα HTML.

```
function loadDoc() {  
    var xmlhttp = new XMLHttpRequest();  
    xmlhttp.open("GET", "TEST.txt", false); //AJAX call for synchronous data transfer  
    xmlhttp.send(); //Asynchronous wouldn't aide us cause we would have to add multiple handlers (time or not) for smooth running  
    DATA = xmlhttp.responseText;  
    DataLength = DATA.length;  
}
```

Εικόνα 4.12. Απόκτηση δεδομένων μέσω AJAX.

Τέλος, τα δεδομένα αποθηκεύονται στη μεταβλητή DATA μέσω της εντολής xmlhttp.responseText η οποία και θα χρησιμοποιείται από τον κώδικα JavaScript στη συνέχεια για την προβολή της ακολουθίας ή τον έλεγχο τερματισμού.

Οι βασικοί αυτοί υπολογισμοί ανήκουν στη συνάρτηση JavaScript με όνομα loadDoc() η οποία δέχεται αλλά και επεξεργάζεται τα δεδομένα από το Second Life.

#### 4.3.2.6 ΕΠΕΞΕΡΓΑΣΙΑ ΔΕΔΟΜΕΝΩΝ ΣΤΗΝ HTML ΣΕΛΙΔΑ

Τα δεδομένα χρησιμοποιούνται από τη σελίδα HTML μέσω της μεταβλητής DATA. Ο κώδικας JavaScript στη συνέχεια ελέγχει το κάθε χαρακτήρα της ακολουθίας που έχει αποσταλεί μέσω της εντολής DATA.charAt(i), όπου i θετικός αριθμός μικρότερος ή ίσος του μήκους του DATA, και ανάλογα τη τιμή του, αναβοσβήνει το αντίστοιχο χρωματιστό κουμπί για 500ms (0.5 δευτερόλεπτα). Όπως έχει προαναφερθεί, οι τιμές που μπορεί να λάβει θα είναι μια από τις 5 περιπτώσεις, «R», «G», «B», «Y» και «O».

Επίσης, πριν εκτελεστεί το flash (αναβόσβημα), το πρόγραμμα ελέγχει αν τα δεδομένα που έχει αποκομίσει, είναι τα ίδια με αυτά που αποκόμισε πριν, μέσω της μεταβλητής previousDATA. Αν συμβαίνει κάτι τέτοιο, τότε δεν εκτελείται η προβολή της ακολουθίας καθώς αυτό σημαίνει πως δεν υπήρξε αλλαγή της ακολουθίας στο μεσοδιάστημα μεταξύ συνεχόμενων ελέγχων στο αρχείο κειμένου.

Δυστυχώς, αχρείαστοι έλεγχοι στο αρχείο κειμένου TEST.txt γίνονται κατ' εξακολούθηση οπότε θα πρέπει να υπάρχει και έλεγχος στη σελίδα HTML για το αν τα δεδομένα είναι καινούργια ή απλά είναι τα παλιά. Το συγκεκριμένο πρόβλημα θα το έλυne η ασύγχρονη λειτουργία του AJAX η οποία θα ενημέρωνε τη σελίδα HTML μόνο όταν υπήρχε αλλαγή των δεδομένων στο αρχείο κειμένου TEST.txt. Όμως, όπως προαναφέρθηκε, η ασύγχρονη λειτουργία είτε θα «πάγωνε» τη σελίδα HTML όσο θα περίμενε για δεδομένα, κάνοντας το περιβάλλον και την εμπειρία χρήστη, μη βέλτιστη, είτε θα επέτρεπε στον χρήστη να προχωράει στο παιχνίδι χωρίς αυτός όμως να μη γνωρίζει πως οι κινήσεις του είναι λανθασμένες ή ότι η χρωματική ακολουθία άλλαξε. Για να δουλέψει ένα τέτοιο εγχείρημα θα έπρεπε να υπάρχει πλήθος χειριστών και περιοριστών (μεταβλητών) που θα επέτρεπαν την ομαλή λειτουργία. Παρόλα αυτά, επιλέχθηκε η σύγχρονη μέθοδος με την προσθήκη χρονικών καθυστερήσεων (ελάχιστης διάρκειας) που επιτρέπουν στην ομαλότερη λειτουργία που επιτεύχθηκε.

```
var start,end;
var i;
if (DATA.charAt(0) == ""){
    setTimeout(function() {loadDoc();},500);
}
if (previousDATA == DATA){
    setTimeout(function() {loadDoc();},500);
}
else{
    previousDATA = DATA;
    for (i = 0; i < DataLength; i++) {
        start = 1200 * i; //Start and End Time Calculations
        end = start + 500;
        if (DATA.charAt(i) == "R"){
            R_MOVE(start,end);
        }
        else if (DATA.charAt(i) == "G"){
            G_MOVE(start,end);
        }
        else if (DATA.charAt(i) == "B"){
            B_MOVE(start,end);
        }
        else if (DATA.charAt(i) == "Y"){
            Y_MOVE(start,end);
        }
    }
}
```

Εικόνα 4.13. Έλεγχος δεδομένων και εκτέλεση flashing κουμπιών.

Τέλος, η προβολή των ζητούμενων κουμπιών γίνεται με διαφορά 700ms μέσω των προσαρμοσμένων συναρτήσεων R\_MOVE, G\_MOVE, B\_MOVE και Y\_MOVE οι οποίες αλλάζουν το χρώμα του κάθε κουμπιού, στη φωτεινή τους μορφή.

```
//=====Function who makes RED Button blink when the PHP sent data is "R"=====//
function R_MOVE(start_time,end_time) {
    if (document.getElementById("PRESSR") != null) {
        setTimeout(function() {
            document.getElementById('PRESSR').style.backgroundColor = '#FF4433';
        }, start_time);
        setTimeout(function() {
            document.getElementById('PRESSR').style.backgroundColor = '#BF3326';
        }, end_time);
    }
}
```

Εικόνα 4.14. Η συνάρτηση που είναι υπεύθυνη για το blinking (flashing) του κόκκινου κουμπιού.



Αυτή η εναλλαγή, διαρκεί για 500ms και ορίζεται από τις μεταβλητές start και end που καθορίζουν πότε το κάθε κουμπί θα ξεκινήσει την εναλλαγή χρώματος και πότε θα τελειώσει, παρομοιάζοντας flash. Επειδή, η δομή επανάληψης, αν και εκτελεί μια συνάρτηση σε κάθε επανάληψη, εκτελείται ταχύτατα, χρειάστηκε να εισαχθούν οι χρονικές μεταβλητές start και end. Σε αυτήν τη χρονική επεξεργασία, η αρχή του χρόνου βρίσκεται στην προβολή του πρώτου χρώματος. Από εκεί και πέρα το κάθε επόμενο κουμπί ανάβει μετά από 700ms από όταν κλείσει το προηγούμενο.

```
else if (DATA.charAt(i) == "O"){
    setTimeout(function() {document.getElementById("button2").innerHTML = "GAME OVER";}, 500);
    if(side_step < score){
        if ((side_score == 1)&&(side_step<2)){
            setTimeout(function() {document.getElementById("button2").innerHTML = "YOUR SCORE: 0";}, 2000);
        }
        else{
            setTimeout(function() {document.getElementById("button2").innerHTML = "YOUR SCORE:"+ score;}, 2000);
        }
    }
    else{
        setTimeout(function() {document.getElementById("button2").innerHTML = "YOUR SCORE:"+ (score-1);}, 2000);
    }
    setTimeout(function() {document.getElementById('button2').style.transform = 'translateY(-50%)';}, 3500);
    setTimeout(function() {document.getElementById('button2').style.visibility = 'hidden';}, 3500);
    setTimeout(function() {document.getElementById('button1').style.transform = 'translateY(50%)';}, 3800);
    setTimeout(function() {document.getElementById('button1').style.visibility = 'visible';}, 3800);
    previousDATA=0;
    step=0;
    setTimeout(function() {zero();},3800);
}
```

Εικόνα 4.15. Περίπτωση εμφάνισης χαρακτήρα «O» στα δεδομένα υποδεικνύοντας «GAME OVER».

Τέλος, στην περίπτωση που καταφθάσει ως δεδομένο, ο χαρακτήρας «O», γίνεται υπολογισμός του σκορ ενώ μπαίνει σε λειτουργία και το animation του κουμπιού «START GAME». Ουσιαστικά, δεν προκύπτει κάποιου είδους επαναφορά στη σελίδα εκτός από τον μηδενισμό ορισμένων μεταβλητών με τη χρήση της συνάρτησης zero().

```
function zero(){
    score=0;
    side_step=0;
    side_score=0;
}
```

Εικόνα 4.16. Η συνάρτηση zero.



## 4.4 ΠΡΩΤΟΚΟΛΛΑ ΕΠΙΚΟΙΝΩΝΙΑΣ ΚΑΙ ΠΕΡΙΟΡΙΣΜΟΙ

Καθ' όλη τη διαδρομή των δεδομένων από το Second Life προς τον Web Server και από τον Web Server στο Second Life, τα δεδομένα μεταφέρονται με ποικίλους τρόπους, ενώ αν και διατηρούν την αρχική μορφή κειμένου τους, η εναλλαγή πολλαπλών πλατφόρμων και ο εμπλουτισμός τους από νέα δεδομένα χρήστη αγγίζουν πολλαπλές ιδιότητες και πρωτόκολλα που έπρεπε να ληφθούν υπόψιν κατά τη διάρκεια της εργασίας. Παρόλα αυτά, το μέγεθος των δεδομένων, αν και σε πολλές άλλες εφαρμογές θα ήταν το κυριότερο πρόβλημα και η μεγαλύτερη εστία προσοχής κατά την επεξεργασία των δεδομένων, στο συγκεκριμένο project απασχόλησε ελάχιστα την παραγωγή κώδικα αφού εκ των προτέρων υπήρχε η μέριμνα για ταχεία και αποτελεσματική μεταφορά δεδομένων, που απαιτούσε μικρά κομμάτια σε μέγεθος, δεδομένων. Ωστόσο, ακόμα κι εδώ, μπορούν να υπάρξουν ακραίες περιπτώσεις που αγγίζουν τα όρια της λειτουργίας του project και μπορούν δυνητικά να δημιουργήσουν ίσως σε σπάνιες περιπτώσεις, πρόβλημα.

### 4.4.1 ΠΡΩΤΟΚΟΛΛΑ ΚΑΙ ΠΕΡΙΟΡΙΣΜΟΙ ΣΤΟ SECOND LIFE

Στο Second Life τα δεδομένα δέχονται μεταφορά μέσω καναλιών επικοινωνίας εντός του Second Life, παραλαμβάνονται με HTTP σύνδεση από Web Server ενώ τέλος, αποστέλλονται στον Web Browser. Η πύλη (port) που χρησιμοποιείται από τον Server του Second Life για όλες τις επικοινωνίες μέσω HTTP στις οποίες λαμβάνει δεδομένα το Second Life, είναι η «12046» [14].

#### 4.4.1.1 ΚΑΝΑΛΙΑ ΕΠΙΚΟΙΝΩΝΙΑΣ

Στο Second Life τα δεδομένα διαβιβάζονται μέσω custom-made καναλιών (channel) τα οποία τα αποστέλλουν ως κείμενο στα αντικείμενα ή στους χρήστες που βρίσκονται εντός εμβέλειας

Function	Distance
llWhisper	10 metres
llSay	20 metres
llShout	100 metres
llRegionSay	region-wide
llRegionSayTo	region-wide

Πίνακας 4.1. Εύρος ακρόασης διάφορων συναρτήσεων εντός του Second Life

Τα συγκεκριμένα κανάλια μπορούν να είναι είτε θετικά είτε αρνητικά ενώ το εύρος των καναλιών είναι από -2147483648 έως 2147483647.

Το μέγεθος των μηνυμάτων μπορεί να είναι μέχρι 1023 byte στα θετικά και 254 byte στα αρνητικά. Ο συγκεκριμένος περιορισμός αποτελεί αναγνωρισμένο bug (προγραμματιστικό λάθος)[12]. Κάθε script μπορεί να «ακούει» έως και 65 κανάλια ταυτοχρόνως[13].

#### 4.4.1.2 ΠΑΡΑΛΑΒΗ ΔΕΔΟΜΕΝΩΝ

Για την παραλαβή των δεδομένων από τον Web Server, ο κώδικας στο Second Life χρησιμοποιεί το συμβάν (event) `http_request`. Όμως, τα δεδομένα προκύπτουν από την ανάγνωση της επικεφαλίδας που προκύπτει στην HTTP σύνδεση. Αυτός διέπεται μέσω της εντολής `llGetHTTPHeader(id, "x-query-string")`. Ο Header (επικεφαλίδα) περιορίζεται στα 255 byte ενώ είναι διαθέσιμος μόνο για 30 δευτερόλεπτα από τη κλήση του event `http_request[5]` το οποίο σκανδαλίζεται από την αίτηση που δημιουργεί η HTML σελίδα με τη δημιουργία νέου παραθύρου. Σημειώνεται πως τα δεδομένα που προκύπτουν από την επικεφαλίδα αποθηκεύονται σε μεταβλητή string ενώ για να είναι προσπελάσιμα θα πρέπει να οριστούν ως δεδομένα HTML με την εντολή `llSetContentType(id, CONTENT_TYPE_HTML)`. Για να χρησιμοποιηθεί το `CONTENT_TYPE_HTML` θα πρέπει η `http_request` να εκτελείται εντός Second Life, ο χρήστης (συνδεδεμένος στον Second Life Client που βλέπει τη σελίδα, δηλαδή το παραγόμενο url) να είναι ο ιδιοκτήτης του αντικειμένου και τέλος, ο χρήστης να είναι συνδεδεμένος στην ίδια περιοχή του Second Life (region) που βρίσκεται και το αντικείμενο. Τέλος, σημαντικό είναι στο παραγόμενο URL του Second Life να τοποθετείται ο χαρακτήρας «?» πριν από κάποια απάντηση ενώ θα πρέπει πριν τον χαρακτήρα «?» να υπάρχει κάθετος χαρακτήρας «/». Σε περίπτωση απώλειας του το status της HTTP σύνδεσης ορίζεται ως 500 «Server Error code»[14].

#### 4.4.1.3 ΑΠΟΣΤΟΛΗ ΔΕΔΟΜΕΝΩΝ

Το Simon Says στο Second Life στέλνει δεδομένα προς τον Web Server μέσω HTTP request, όχι στο παραγόμενο URL που χρησιμοποιήθηκε προηγουμένως, αλλά κατευθείαν στην PHP σελίδα που βρίσκεται στον Web Server. Έτσι, αυτή η HTTP επικοινωνία είναι διαφορετική από την προηγούμενη καθώς ο Server σε αυτήν την περίπτωση είναι ο Web Server και όχι ο Server του Second Life. Η εντολή που χρησιμοποιείται είναι η `llHTTPRequest( string url, list parameters, string body )`, όπου string url είναι ένα έγκυρο HTTP/HTTPS URL. Στη συγκεκριμένη περίπτωση, το sting url είναι η μεταβλητή `PHP_DATA` που περιέχει τη διεύθυνση της σελίδας PHP μαζί με τη μεταβλητή και την τιμή της που θα αποσταλούν στη σελίδα. Η λίστα παραμέτρων (list parameters) είναι κενή οπότε χρησιμοποιείται η

προκαθορισμένη (default) μέθοδος HTTP, «GET». Προσοχή απαιτείται στην τιμή της μεταβλητής της PHP σελίδας που αποστέλλεται με την διεύθυνση στην HTTP επικοινωνία καθώς κενοί χαρακτήρες, χαρακτήρες ελέγχου και άλλοι που δεν επιτρέπονται στις διευθύνσεις URL, θα προκαλούν σφάλμα εκτέλεσης (run time error). Οι χαρακτήρες που επιτρέπονται είναι σετ utf-8[15]. Επίσης, οι κλήσεις HTTP θα πρέπει να πραγματοποιούνται εντός 60 δευτερολέπτων, αλλιώς η κλήση θα χαθεί και το status code (κωδικός κατάστασης) της HTTP απάντησης θα είναι «499». Οι συνολικοί χαρακτήρες στη διεύθυνση URL που διέπεται μπορεί να έχει μέγιστο μήκος 253 χαρακτήρων [15].

Σημαντικό είναι να ειπωθεί εδώ πως επιτρέπονται 25 κλήσεις `llHttpRequest()` εντός 20 δευτερολέπτων για κάθε αντικείμενο ενώ επιτρέπονται 1000 κλήσεις `llHttpRequest()` εντός 20 δευτερολέπτων για κάθε ιδιοκτήτη. Οι συγκεκριμένες τιμές είναι πιθανό να αλλάξουν μελλοντικά καθώς αποτελούν αντικείμενο ανάπτυξης για το Second Life. Αν επιτευχθεί το ανώτατο όριο κλήσεων (throttle blocking), τότε η επιστροφή της κλήσης `llHttpRequest()` θα είναι η «NULL\_KEY». Σε τέτοια περίπτωση ο χρόνος αναμονής για επαναφορά σε κανονικές τιμές είναι τα 40 δευτερόλεπτα[15].

#### 4.4.1.4 ΔΕΣΜΕΥΣΗ URL

Συνιστάται έντονα η απελευθέρωση των διευθύνσεων URL πριν από τη δημιουργία ενός νέου ή μετά τον τερματισμό του χρόνου παιχνιδιού. Αυτό συμβαίνει επειδή οι διευθύνσεις URL τείνουν να υπάρχουν ακόμα και μετά το κλείσιμο του παιχνιδιού. Καταλαμβάνουν επίσης ένα φυσικό χώρο στη μνήμη μιας περιοχής (region) στο Second Life. Αυτό σημαίνει ότι εάν ένα in-world region έχει χωρητικότητα 1000 prim(τύπος αντικειμένου στη Second Life) τότε θα μπορούσε να γεμίσει με 1000 διευθύνσεις URL και ο ιδιοκτήτης του δεν θα ήταν σε θέση να χτίσει ή να γράψει τίποτα. Αυτό μπορεί να οδηγήσει στην απαγόρευση των χρηστών από ορισμένες περιοχές, αν δεν έχουν την ιδιοκτησία της γης. Η Second Life έχει προβλέψει αυτό το πρόβλημα και έχει υλοποιήσει μια αυτόματη απελευθέρωση URL. Ωστόσο, αυτή η έκδοση δεν λειτουργεί άμεσα και δεν έχει τέλεια απόδοση.



**Important:** Never ever forget to release a URL again which you have requested! URLs are region resources just like prims. If you take them all you can get into big trouble with the sim owner and/or estate managers.

Εικόνα 4.17. Μήνυμα που υπάρχει στην εγκυκλοπαίδεια του Second Life.

Η εντολή `llGetFreeURLs()` μπορεί να χρησιμοποιηθεί ώστε να εμφανιστεί ο αριθμός των διαθέσιμων URL της περιοχής που προορίζονται για δωρεάν δέσμευση. Η χρησιμοποίηση της εντολής `llRequestURL()` είναι περιορισμένη, αν και δεν υπάρχει ακριβή μέτρηση για το πόσες χρήσεις

επιτρέπονται, συνίσταται αυτές να είναι λιγότερες από 5 σε σύντομο χρονικό διάστημα. Η χρήση της εντολής μετά από απόρριψη εξυπηρέτησης θα είναι διαθέσιμη μετά από 1 δευτερόλεπτο. Προτείνεται να χρησιμοποιείται συνάρτηση παύσης (ISleep) για 0.6 δευτερόλεπτα ή και περισσότερο, ανάμεσα σε συνεχόμενες κλήσεις. Επιπλέον, κάθε φορά που μια περιοχή επανεκκινείται, όλα τα URL αποδεσμεύονται και καθίστανται άχρηστα[16].

Για αποδέσμευση των URL χρησιμοποιείται η εντολή IIReleaseURL() ενώ στις παρακάτω περιπτώσεις, τα URL αποδεσμεύονται αυτόματα.

- Όταν η περιοχή επανεκκινηθεί ή τεθεί εκτός σύνδεσης.
- Όταν το πρόγραμμα (script) που κατέχει τις διευθύνσεις URL επαναρυθμίζεται (reseted) ή μεταγλωττίζεται (compiled).
- Όταν το αντικείμενο που περιέχει το πρόγραμμα (script) διαγράφεται ή μεταφέρεται στο απόθεμα.

Παρόλα αυτά, όπως έχει ήδη προαναφερθεί, η αυτόματη αποδέσμευση των URL δεν είναι 100% αποτελεσματική, όπως έχει αναφερθεί από κάποιους χρήστες.

#### 4.4.2 ΠΡΩΤΟΚΟΛΛΑ ΚΑΙ ΠΕΡΙΟΡΙΣΜΟΙ ΣΤΟΝ WEB SERVER

Στον Web Server αποστέλλονται και φθάνουν δεδομένα προς και από το Second Life μέσω μιας σελίδας HTML και μιας σελίδας PHP. Για αυτήν την επικοινωνία χρησιμοποιείται HTTP επικοινωνία. Αλλά, ταυτόχρονα επιτελείται ακόμη μια επικοινωνία, αυτή μεταξύ της σελίδας HTML και της σελίδας PHP. Εκεί, χρησιμοποιείται το σύνολο εργαλείων AJAX που βασίζεται στη φόρτωση δεδομένων στην HTML σελίδα, χωρίς την ανανέωση της.

##### 4.4.2.1 ΕΠΙΚΟΙΝΩΝΙΑ HTML ΠΡΟΣ SECOND LIFE

Η σελίδα HTML επικοινωνεί με το Second Life μέσω HTTP επικοινωνίας. Η HTML σελίδα δημιουργεί ένα νέο παράθυρο περιήγησης και εισάγει στην επικεφαλίδα του, το παραχθέν URL του Second Life και επιπλέον την επιλογή του χρήστη. Αυτή η κλήση στον Server του Second Life σκανδαλίζει το πρόγραμμα στο Second Life, το οποίο αναγιγνώσκει την επικεφαλίδα. Οι περιορισμοί στην μεταφορά της επικεφαλίδας καλύφθηκαν από την παράγραφο 4.4.1.3 καθώς η HTTP επικοινωνία εκκινείται από το Second Life και τον Second Life Server.

Παρόλα αυτά, το νέο παράθυρο που δημιουργείται, τοποθετείται σε μια περιοχή iframe η οποία φέρει έναν κύριο περιορισμό όσον αφορά την ασφαλή περιήγηση.

#### 4.4.2.2 IFRAME και SSL

Στο Second Life υπάρχει ξεχωριστή εντολή που παράγει URL με SSL πιστοποιητικό και αυτή είναι η `HttpRequestSecureURL()`. Όμως, ενώ αυτή η εντολή δουλεύει και παράγεται URL με HTTPS της μορφής `"https://sim10186.agni.lindenlab.com:12043/cap/469d084a-e1ab-2e35-6e6d-95c3d6f464b0"` το URL αυτό δεν φέρει ενημερωμένο πιστοποιητικό ασφαλείας.

Αυτό έχει σαν αποτέλεσμα όλοι οι Web Browser να εντοπίζουν την κατάληξη `https` εντός του `iframe` στην HTML σελίδα αλλά μόλις η σελίδα φορτώνεται, να μην έχουν τη δυνατότητα να εντοπίσουν ισχύον πιστοποιητικό και άρα να το θεωρούν απειλή. Έτσι όλοι οι Web Browser που δοκιμάστηκαν (Mozilla, Chrome, Edge) δεν επιτρέπουν την φόρτιση της σελίδας αλλά και την επικοινωνία με το Second Life. Το πρόβλημα προκαλείται με την επίτευξη μονομερούς πιστοποίησης SSL για τη σελίδα HTML ενώ ταυτοχρόνως το URL του Second Life δε φέρει πιστοποιητικό.

Σε αυτό φέρει ευθύνη η αντιμετώπιση των πεδίων `iframe` από τους περιηγητές διαδικτύου. Η μη ασφαλής εκτέλεση `iframe` δημιουργεί προβλήματα στους Browsers και για αυτό τις απορρίπτουν απολύτως αν υπάρχει η παραμικρή αμφιβολία ασφάλειας ενώ ακόμη κι αν η σελίδα HTML είναι HTTPS, το `iframe` δε μπορεί να είναι HTTP[17]. Για αυτή την αδυναμία παροχής `https` σύνδεσης από το Second Life ευθύνεται η μητρική εταιρεία της πλατφόρμας, η Linden, η οποία ακόμα και στην εγκυκλοπαίδεια του Second Life, δεν έχει ενημερώσει τα πιστοποιητικά. Έτσι, ενώ υπήρξε προσπάθεια για παράκαμψη του προβλήματος της SSL πιστοποίησης, αποφασίστηκε πώς η εντολή `iframe` είναι ζωτικής σημασίας για τη λειτουργικότητα της εργασίας και θα πρέπει να παραμείνει δυστυχώς με HTTP σύνδεση κάτι το οποίο όμως δεν επηρεάζει κάπου τη λειτουργικότητα της.

#### 4.4.2.3 ΕΠΙΚΟΙΝΩΝΙΑ SECOND LIFE ΠΡΟΣ PHP

Για την αποστολή δεδομένων από το Second Life προς τη σελίδα PHP, η σελίδα PHP ελέγχει αν η μεταβλητή `test`, υπάρχει στην κλήση της διεύθυνσης της με την δομή ελέγχου `isset()` η οποία είναι λογικού τύπου (BOOLEAN).

Επίσης, για τη δημιουργία του αρχείου κειμένου `TEST.txt` χρησιμοποιείται η συνάρτηση `file_put_contents(filename, data, mode, context)`. Σε αυτή τα πεδία `filename` και `data` είναι υποχρεωτικά. Αν δεν υπάρχει αρχείο με το όνομα `filename`, η συνάρτηση θα το δημιουργήσει. Η συγκεκριμένη συνάρτηση απαιτεί τη χρήση έκδοσης PHP 5.0 ή μεταγενέστερη. Η επιστροφή της συγκεκριμένης συνάρτησης είναι ο αριθμός των byte που ενέγραψε στο αρχείο, σε περίπτωση επιτυχής εγγραφής ή FALSE, σε περίπτωση αποτυχίας[18].

#### 4.4.2.4 ΧΡΗΣΗ AJAX

Στη συγκεκριμένη εργασία, χρησιμοποιείται η χρήση AJAX εντός κώδικα JavaScript που βρίσκεται εμφωλευμένος στη σελίδα HTML.

Αρχικά, η χρήση AJAX λειτουργεί μόνο για επικοινωνία εντός του ίδιου ονόματος Domain, για λόγους ασφαλείας. Όλοι οι σύγχρονοι Web Browser υποστηρίζουν τη χρήση του (Chrome, Firefox, IE7+, Edge, Safari, Opera) [19].

Για την επικοινωνία, χρησιμοποιείται HTTP επικοινωνία με μέθοδο «GET». Η κατάσταση της HTTP επικοινωνίας συνήθως είναι «OK» (status code 200).

#### 4.5 ΣΥΝΟΨΗ ΚΕΦΑΛΑΙΟΥ

Η μεταφορά δεδομένων του χρήστη από και προς το Second Life κάνει χρήση πολλαπλών τεχνικών, γλωσσών προγραμματισμού και διαφορετικών πλατφόρμων οι οποίες στο μεγαλύτερο βαθμό επικοινωνούν αυτόματα.

Εντός του Second Life η μεταφορά των δεδομένων γίνεται ανάμεσα στα διαφορετικά αντικείμενα που απαρτίζουν το παιχνίδι Simon Says ενώ χρησιμοποιούνται 3 διαφορετικά κανάλια επικοινωνίας του Second Life για την επίτευξη επικοινωνίας.

Για την επικοινωνία με τον Web Server χρησιμοποιήθηκε επικοινωνία HTTP μέσω URL που παράγεται από τον Server του Second Life, ενώ η τελική διεύθυνση διέπεται με τη χρήση ενός QR κωδικού.

Στον Web Server, για να αποσταλούν δεδομένα στο Second Life, δημιουργείται νέο παράθυρο περιήγησης, στη σελίδα HTML, για κάθε διαφορετική επιλογή του χρήστη, ενώ το παράθυρο αυτό μένει αόρατο με την εγκατάσταση του σε μια περιοχή iframe. Προς τον Web Server, τα δεδομένα διαχειρίζονται από μια σελίδα PHP η οποία τα εγγράφει σε ένα αρχείο κειμένου TXT.

Η ανάγνωση δεδομένων στην HTML σελίδα από το αρχείο κειμένου επιτυγχάνεται με τη χρησιμοποίηση AJAX πράγμα το οποίο εξασφαλίζει τη μη ανανέωση της σελίδας HTML κάθε φορά που πρέπει να αναγνωστούν νέα δεδομένα.

Καθ' όλη τη μεταφορά δεδομένων σε κάθε τομέα της εργασίας, υπάρχουν περιορισμοί μεγέθους δεδομένων, κλήσης συναρτήσεων και απελευθέρωσης διευθύνσεων, οι οποίες στο μεγαλύτερο μέρος τους δεν επηρεάζουν τη λειτουργικότητα της εργασίας καθώς κάνει χρήση μικρού μεγέθους δεδομένων. Ταυτόχρονα, το κυριότερο πρωτόκολλο που ισχύει κατά την επικοινωνία και την μετάβαση των δεδομένων, είναι το πρωτόκολλο HTTP που χρησιμοποιείται εκτενώς για την κλήση (request) και απάντηση (response) με δεδομένα ανάμεσα στους φορείς.

## 4.6 ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1]. Second Life Wiki, Listen, <http://wiki.secondlife.com/wiki/Listen>.
- [2]. Second Life Wiki, LSL List, [http://wiki.secondlife.com/wiki/Category:LSL\\_List](http://wiki.secondlife.com/wiki/Category:LSL_List).
- [3]. Second Life Wiki, lIFrand, <http://wiki.secondlife.com/wiki/lIFrand>.
- [4]. Second Life Wiki, lISleep, <http://wiki.secondlife.com/wiki/lISleep>.
- [5]. Second Life Wiki, lIGetHTTPHeader, <http://wiki.secondlife.com/wiki/lIGetHTTPHeader>.
- [6]. En. Wikipedia, POST (HTTP), [https://en.wikipedia.org/wiki/POST\\_\(HTTP\)](https://en.wikipedia.org/wiki/POST_(HTTP)).
- [7]. Hypertext Transfer Protocol -- HTTP/1.1. , <https://tools.ietf.org/html/rfc2616#section-10.2.1>, 1999, page 57.
- [8]. Stack Overflow, PHP live Read A file, <https://stackoverflow.com/questions/3218895/php-how-to-read-a-file-live-that-is-constantly-being-written-to>.
- [9]. W3Schools, JS AJAX, [https://www.w3schools.com/js/js\\_ajax\\_http.asp](https://www.w3schools.com/js/js_ajax_http.asp).
- [10]. Developer Mozilla, Same Origin Policy, [https://developer.mozilla.org/en-US/docs/Web/Security/Same-origin\\_policy](https://developer.mozilla.org/en-US/docs/Web/Security/Same-origin_policy).
- [11]. AJAX CROSS ORIGIN, HOW, <http://www.ajax-cross-origin.com/how.html>.
- [12]. Second Life, BUG with URL, <https://jira.secondlife.com/browse/BUG-41541>.
- [13]. Second Life Wiki, LSL Chat, [http://wiki.secondlife.com/wiki/Category:LSL\\_Chats](http://wiki.secondlife.com/wiki/Category:LSL_Chats).
- [14]. Second Life Wiki, LSL HTTP Server, [http://wiki.secondlife.com/wiki/LSL\\_HTTP\\_server](http://wiki.secondlife.com/wiki/LSL_HTTP_server).
- [15]. Second Life Wiki, lIHTTPRequest, <http://wiki.secondlife.com/wiki/lIHTTPRequest>, Caveats, Throttles.
- [16]. Second Life Wiki, lIRequestURL, <http://wiki.secondlife.com/wiki/lIRequestURL>.
- [17]. TinyWebGallery Blog, Why you should not mix http and https when using iframes , <https://www.tinywebgallery.com/blog/iframe-do-not-mix-http-and-https>, 2017.
- [18]. W3Schools, PHP Put Contents, [https://www.w3schools.com/php/func\\_filesystem\\_file\\_put\\_contents.asp](https://www.w3schools.com/php/func_filesystem_file_put_contents.asp).

[19]. W3Schools, JS AJAX, [https://www.w3schools.com/js/js\\_ajax\\_ht](https://www.w3schools.com/js/js_ajax_ht).



## ΚΕΦΑΛΑΙΟ 5: Επικοινωνία Πλακέτας ARDUINO με Second Life

### 5.1 ΕΙΣΑΓΩΓΗ

Σε αυτό το κεφάλαιο υλοποιείται έμμεση επικοινωνία του παιχνιδιού Simon Says στο Second Life με μια πλακέτα μικροελεγκτή ARDUINO. Η επικοινωνία επιτυγχάνεται με τη σύνδεση της πλακέτας σε υπολογιστή ο οποίος χρησιμοποιεί περιηγητή διαδικτύου (web browser) με δυνατότητα σειριακής ανάγνωσης. Ο περιηγητής πρέπει να συνδεθεί στο διαδίκτυο στη διεύθυνση «<http://siwonsimon.000webhostapp.com/ARDUINO.html>» ενώ στη συνέχεια πρέπει να προσθέσει τον χαρακτήρα «?» και μετά το URL που έχει παραχθεί από το Second Life.

Ο περιηγητής που χρησιμοποιείται αποτελεί προϊόν που παράχθηκε χειροκίνητα στο πρόγραμμα Visual Studio 2013 των Windows. Η γλώσσα που χρησιμοποιεί το Visual Studio είναι η Visual Basic με την οποία γίνεται ο χειρισμός των δεδομένων που καταφτάνουν από την πλακέτα του Arduino.

Η πλακέτα του μικροελεγκτή λειτουργεί ως χειριστήριο (controller) στο παιχνίδι του Simon Says καθώς διαθέτει συνδεδεμένα 5 κουμπιά για την αλληλεπίδραση του χρήστη με το παιχνίδι. Επίσης επικοινωνεί με τον τεχνητό Web Browser μέσω USB που είναι συνδεδεμένο στον υπολογιστή που διαθέτει τον τεχνητό Web Browser.

Παρόλα αυτά ο Web Browser που έχει παραχθεί μπορεί να χρησιμοποιηθεί χωρίς απαραίτητα στον ίδιο υπολογιστή να εκτελείται το Second Life. Άρα δεν απαιτείται άμεση πρόσβαση στο Second Life καθώς ο Web Browser έχει τη δυνατότητα να αναπαράγει και να δέχεται τη χρωματική ακολουθία που παράγεται στο Second Life.

### 5.2 ΠΛΗΡΟΦΟΡΙΕΣ

#### 5.2.1 ARDUINO

Το Arduino αποτελεί μικροελεγκτή μονής πλακέτας που είναι ανοικτού κώδικα. Στη συγκεκριμένη εργασία χρησιμοποιείται η πλακέτα Arduino Uno. Ουσιαστικά αποτελεί μια μητρική πλακέτα που περιλαμβάνει έναν μικροελεγκτή (ATmega328P τεχνολογία) η οποία μπορεί να συνδέεται με συσκευές και εξαρτήματα μέσω των 14 ψηφιακών θυρών εισόδου/εξόδου. Διαθέτει επίσης 6 αναλογικές θύρες και 1 θύρα για ρολόι 16MHz quartz κρυστάλλου[1].

Η σύνδεση της με υπολογιστή μπορεί να γίνει μέσω καλωδίου USB (type A-B) όπου εκεί μπορεί να προγραμματιστεί για να εκτελεί απλά προγράμματα, μέσω της εφαρμογής Arduino που διατίθεται από την ομώνυμη εταιρεία[1]. Μέσω του ίδιου USB λαμβάνει και ενέργεια, καθώς δεν διαθέτει μπαταρίες ενώ μπορεί να τροφοδοτηθεί με ρεύμα κι από μια θύρα μέσω AC/DC καλωδίου (προτείνεται 7 με 12 V η τάση εισόδου)[1].

Η πλακέτα Arduino χρησιμοποιεί γλώσσα Wiring (συγγενική με τη γλώσσα C++) για να προγραμματιστεί, μαζί με πολλαπλές βιβλιοθήκες, ενώ η μέγιστη χωρητικότητα κώδικα και βιβλιοθηκών που μπορεί να φέρει είναι τα 32 KB (0.5 KB εκ των οποίων είναι δεσμευμένα)[1].

### 5.2.2 Visual Studio

Το Microsoft Visual Studio αποτελεί ένα ολοκληρωμένο προγραμματιστικό περιβάλλον (Integrated Development Environment) της Microsoft. Χρησιμοποιείται για την ανάπτυξη ποικίλων άλλων προγραμμάτων υπολογιστών αλλά και για εφαρμογές κινητών τηλεφώνων και εφαρμογών διαδικτύου.

Το Visual Studio χρησιμοποιεί ποικίλα εργαλεία και υπηρεσίες της Microsoft όπως είναι τα Windows Forms, Windows Store κ.α. Χρησιμοποιεί πολλαπλές γλώσσες προγραμματισμού όπως είναι η C, C++, Visual Basic.NET (VB.NET), C# (στη συγκεκριμένη εργασία χρησιμοποιείται μόνο Visual Basic στο Visual Studio)[2].

Εκδόσεις του Visual Studio προκύπτουν ανά τακτά χρονικά διαστήματα με την πιο πρόσφατη έκδοση να είναι η «2019 version 16.3.2»[2]. Στη συγκεκριμένη εργασία γίνεται χρήση του Visual Studio 2013 καθώς αυτό ήταν το μόνο διαθέσιμο. Παρόλα αυτά το πρόγραμμα που παράγεται χρησιμοποιεί βασικές λειτουργίες του Visual Studio, οπότε δεν υπάρχει κίνδυνος παραλήψεων αν χρησιμοποιηθεί παλαιότερη έκδοση.

Ο τρόπος χρήσης του Visual Studio, αν και οι πολλαπλές λειτουργίες του και τα πολυπληθέστατα εργαλεία που κατέχει μπορεί να δημιουργούν λάθος εντύπωση, είναι απλός.

Διαθέτει «Debugger» υψηλής αποτελεσματικότητας καθώς μπορεί και εμφανίζει τιμές διαφόρων μεταβλητών κατά τη διάρκεια του «Debugging» ενώ σε περίπτωση σφάλματος δίνει τις περισσότερες φορές εμπεριστατωμένη άποψη του λάθους και τρόπου επίλυσης αυτού[3].

Στη συγκεκριμένη εργασία γίνεται χρήση της λειτουργίας Designer και πιο ειδικά της λειτουργίας «Windows Forms Designer». Η συγκεκριμένη λειτουργία επιτρέπει τη δημιουργία πολλαπλών εργαλείων εντός μιας φόρμας (ουσιαστικά κενή σελίδα προγράμματος), συμπεριλαμβανομένου και

διαδικτυακού περιηγητή (web browser) ενώ επιτρέπει την εξόπλιση του με εργαλεία με απλό drag&drop στη φόρμα (σύρσιμο των εργαλείων στο σημείο του περιηγητή που επιθυμεί ο χρήστης)[4].

Το περιβάλλον χρήστη κατά την ανάπτυξη του προγράμματος είναι συνδεδεμένο με κώδικα προγραμματισμού βασιζόμενος σε γεγονότα (event-driven programming)[2][5].

Η γλώσσα που χρησιμοποιείται στη λειτουργία Windows Forms Designer είναι η «Visual Basic .NET», μια γλώσσα προγραμματισμού βασιζόμενη σε αντικείμενα[6]. Η συγκεκριμένη γλώσσα εφαρμόζεται στη «.NET» δομή (.NET Framework), μια δομή των Windows που επιτρέπει την πολλαπλή επικοινωνία και εφαρμογή γλωσσών προγραμματισμού[7].

### 5.3 ΕΓΚΑΤΑΣΤΑΣΗ

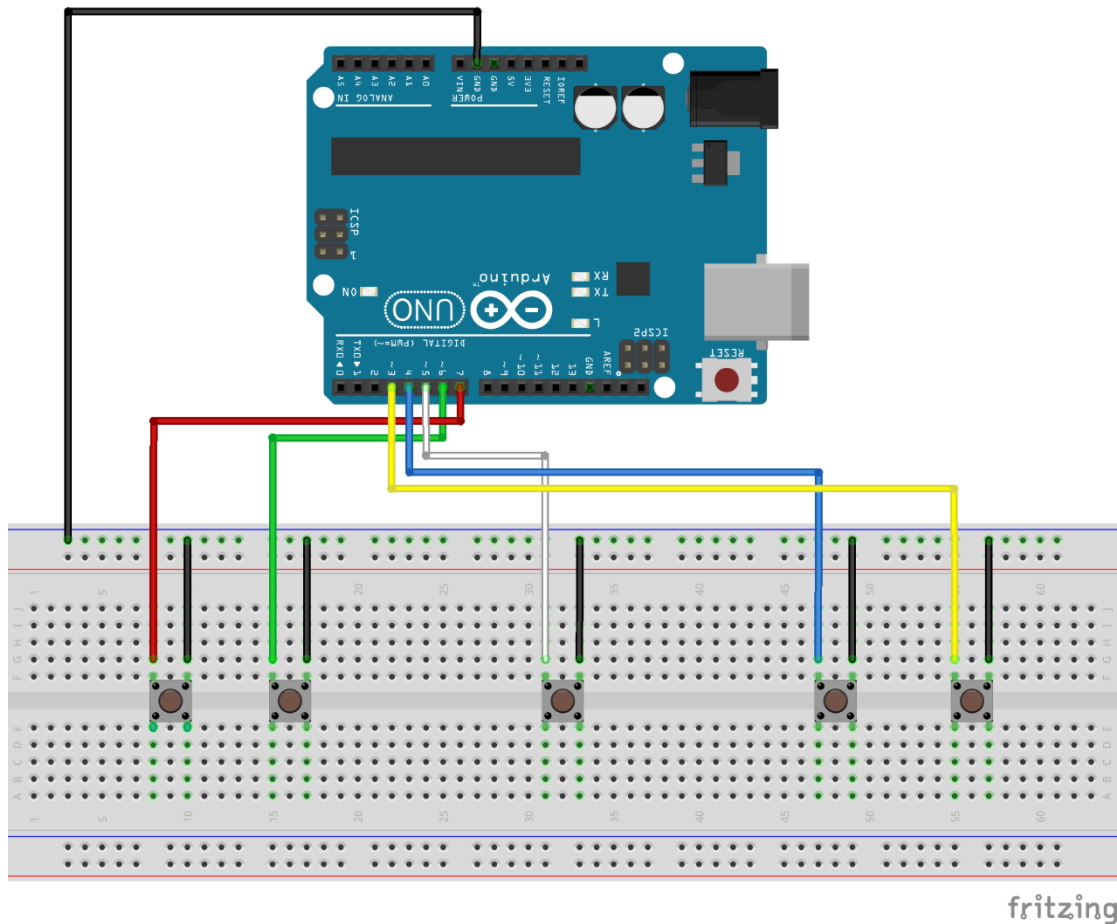
Για τη λειτουργία της συγκεκριμένης λειτουργίας απαιτείται η σωστή εγκατάσταση της πλακέτας Arduino και η εγκατάσταση του προγράμματος Simon Browser. Ακόμα εννοείται η απόκτηση και η εκτέλεση του προϊόντος Simon Says.v3 στο Second Life καθώς και η παραγωγή και κατοχή διαθέσιμου URL για την επίτευξη παιχνιδιού μέσω HTTP επικοινωνίας.

Να σημειωθεί πως η εγκατάσταση της πλακέτας Arduino και η εγκατάσταση του προγράμματος Simon Browser απαιτείται να γίνει στην ίδια συσκευή η οποία έχει και πρόσβαση στο διαδίκτυο. Το Second Life μπορεί να εκτελείται σε διαφορετική συσκευή, σε διαφορετικό δίκτυο αλλά θα πρέπει να υπάρχει κάποιου είδους επικοινωνία με τη συγκεκριμένη συσκευή για την απόκτηση του URL (ακόμη κι ένα απλό μήνυμα).

Το ζητούμενο είναι το URL να είναι γνωστοποιημένο ώστε να μπορεί να επικολληθεί εντός του Simon Browser και έτσι το πρόγραμμα να μπορεί να επικοινωνήσει με το παιχνίδι Simon Says στο Second Life.

### 5.3.1 ΕΓΚΑΤΑΣΤΑΣΗ ΠΛΑΚΕΤΑΣ ARDUINO

Με την πλακέτα του Arduino αποσυνδεδεμένη από το ρεύμα και από κάποια συσκευή (μέσω USB), απαιτείται να υλοποιηθεί η συνδεσμολογία της Εικόνας 5.1.



Εικόνα 5.1. Συνδεσμολογία Arduino με κουμπιά.

Στην Εικόνα 5.1 φαίνεται η εγκατάσταση 5 απλών κουμπιών σε μια επιφάνεια breadboard και η σύνδεση τους με την πλακέτα Arduino. Συγκεκριμένα, πάνω από έναν ακροδέκτη κάθε κουμπιού συνδέεται ο ουδέτερος αγωγός (μαύρο καλώδιο) ο οποίος καταλήγει στη γείωση της πλακέτας (θύρα με όνομα GND).

Έπειτα, σε άλλο ακροδέκτη κάθε κουμπιού συνδέεται καλώδιο με την άλλη κατάληξη του καλωδίου να καταλήγει σε μια ψηφιακή θύρα της πλακέτας. Το χρώμα κάθε καλωδίου που συνδέεται με κάθε κουμπί υποδηλώνει ένα από τα 4 χρώματα του παιχνιδιού Simon Says, ενώ το κεντρικό καλώδιο χρώματος λευκό συνδέει το κουμπί Εκκίνησης (START).

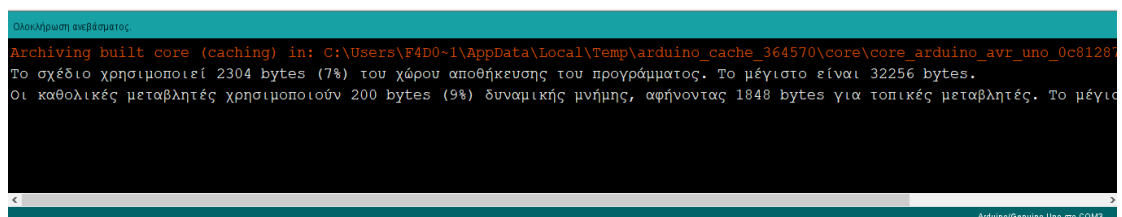
Έτσι, τα κουμπιά συνδέονται στις θύρες ως εξής:

- Κόκκινο κουμπί στη θύρα 7
- Πράσινο κουμπί στη θύρα 6
- Κουμπί Εκκίνησης στη θύρα 5
- Μπλε κουμπί στη θύρα 4
- Κίτρινο κουμπί στη θύρα 3

Τα κουμπιά να σημειωθεί πως δεν χρειάζονται τροφοδοσία από την πλακέτα Arduino καθώς οι απλές ψηφιακές θύρες παρέχουν ρεύμα έως 40mA. Παρόλα αυτά για πιο πολύπλοκες και μεγαλύτερες διατάξεις προτείνεται η χρήση αντιστάσεων 470Ω ή 1KΩ σε σειρά με κάθε κουμπί[8].

Στη συνέχεια η πλακέτα του Arduino είναι έτοιμη να συνδεθεί μέσω USB με υπολογιστή. Εκεί, χρησιμοποιώντας το πρόγραμμα «Arduino IDE» [9], ο χρήστης πρέπει να βεβαιωθεί ότι στην κατηγορία «Εργαλεία», η αναγραφόμενη θύρα που είναι συνδεδεμένη η πλακέτα Arduino Uno, είναι η «COM3». Στη συνέχεια, επισκεπτόμενος τη διεύθυνση «<https://github.com/petros3432/SLSimonSays/tree/master/Simon%20Browser%20Arduino/Arduino>» ο χρήστης μπορεί να κατεβάσει το αρχείο «BUTTON.ino» και να το ανοίξει με το πρόγραμμα «Arduino» ή να αντιγράψει τα περιεχόμενα του αρχείου σε μια κενή σελίδα του προγράμματος «Arduino».

Στη συνέχεια απαιτείται να πατηθεί το κουμπί «Ανέβασμα» ώστε να «φορτωθεί» η πλακέτα Arduino με το επιθυμητό κώδικα. Θα πρέπει να εμφανίζεται μήνυμα όπως της Εικόνας 5.2.



Εικόνα 5.2. Μήνυμα επιτυχής φόρτωσης κώδικα στη πλακέτα Arduino.

Ο χρήστης μπορεί να ελέγξει την σωστή εγκατάσταση της συνδεσμολογίας της πλακέτας, πατώντας το κουμπί «Παρακολούθηση Σειριακής» στην πάνω-δεξιά γωνία του προγράμματος Arduino (ή Εργαλεία >Παρακολούθηση Σειριακής). Εκεί, πρέπει να ορίσει τα «9600 baud» (κάτω-δεξιά στο παράθυρο) ώστε να μπορεί να παρακολουθεί τα δεδομένα που δημιουργεί το πάτημα ενός εκ των 5 εγκατεστημένων κουμπιών. Για κάθε κουμπί εμφανίζεται ένα εκ των γραμμάτων ( R, G, S, B, και Y) ένα για κάθε χρώμα και το «S» από το «START» για την εκκίνηση.

Μετά από αυτό το βήμα το πρόγραμμα «Arduino» μπορεί να κλείσει αλλά η πλακέτα δεν πρέπει να αποσυνδεθεί από τον υπολογιστή.

### 5.3.2 ΕΓΚΑΤΑΣΤΑΣΗ Simon Browser

Αρχικά ο χρήστης χρειάζεται να μεταβεί στη διεύθυνση «<https://github.com/petros3432/SLSimonSays/tree/master/Simon%20Browser%20Arduino>» και να κατεβάσει το συμπιεσμένο αρχείο «Simon Browser Arduino.zip». Στη συνέχεια πρέπει να αποσυμπίεστεί το αρχείο σε τοποθεσία του υπολογιστή του χρήστη, ώστε να εμφανίσει το αρχείο τα περιεχόμενα του. Έτσι θα δημιουργηθεί ο φάκελος «Simon Browser Arduino» ο οποίος περιλαμβάνει αρχεία.

Στη συνέχεια απαιτείται επιλογή του αρχείου «setup.exe» ώστε να ξεκινήσει η εγκατάσταση του προγράμματος. Μόλις ξεκινήσει η διαδικασία εγκατάστασης του «Simon Browser» θα εμφανιστεί μήνυμα στο οποίο ο χρήστης πρέπει να επιλέξει την επιλογή «Εγκατάσταση». Έτσι η εγκατάσταση θα ολοκληρωθεί εντός πολύ μικρού χρονικού διαστήματος και το πρόγραμμα θα ανοίξει αυτόματα. Για κάθε νέα εκκίνηση του προγράμματος ο χρήστης αρκεί να επιλέγει το αρχείο «SimonBrowser.application» καθώς η εγκατάσταση απαιτείται μόνο την πρώτη φορά χρήσης του προγράμματος.

Σε περίπτωση που ο χρήστης επιθυμεί την απεγκατάσταση του προγράμματος αυτή είναι δυνατόν να συμβεί με περιήγηση του χρήστη στο «Πίνακας Ελέγχου > Κατάργηση Εγκατάστασης ενός Προγράμματος > Κατάργηση/Αλλαγή ( στο Simon Browser)».

### 5.4 ΤΡΟΠΟΣ ΧΡΗΣΗΣ

Για την επιτυχή λειτουργία της χρήσης πλακέτας Arduino για επικοινωνία με το Second Life θα πρέπει να έχει εγκατασταθεί σωστά τόσο η πλακέτα Arduino όσο και το πρόγραμμα Simon Browser. Ακόμη θα πρέπει το Simon Says.v3 να εκτελείται παράλληλα στο Second Life και ο χρήστης της πλακέτας Arduino να έχει πρόσβαση στο παραχθέν URL του παιχνιδιού.

Έτσι, ο χρήστης αρκεί να ανοίξει το πρόγραμμα Simon Browser. Εκεί θα αντικρίσει μια λευκή σελίδα με ένα άδειο πλαίσιο κειμένου και 2 κουμπιά «GO» και «Simon» παραδίπλα.

Πατώντας το κουμπί «Simon» η διεύθυνση της σελίδας html που θα χρησιμοποιηθεί επικολλάται στο πλαίσιο κειμένου που αποτελεί το πλαίσιο εισόδου διεύθυνσης. Μετά τον χαρακτήρα «?» που συμπεριλαμβάνεται στη διεύθυνση που υπάρχει στο πλαίσιο, ο χρήστης οφείλει να επικολλήσει τη

διεύθυνση URL που έχει παράγει το Second Life και να πατήσει το κουμπί «GO» ώστε να μεταβεί στη σελίδα του παιχνιδιού (πιθανή καθυστέρηση φόρτισης της σελίδας μπορεί να εμφανιστεί).

Εκεί, χρησιμοποιώντας τα κουμπιά που είναι συνδεδεμένα με την πλακέτα Arduino, μπορεί να παίξει το παιχνίδι Simon Says όπως ακριβώς σε μια κινητή συσκευή ή σε κάποιο συμβατικό Web Browser (πχ Google Chrome). Υπενθυμίζεται ότι το κεντρικό κουμπί με το λευκό καλώδιο ξεκινάει το παιχνίδι ενώ τα υπόλοιπα 4 κουμπιά αναφέρουν και ένα από τα 4 κουμπιά του παιχνιδιού. Στη σελίδα που εμφανίζεται στο πρόγραμμα, η χρωματική ακολουθία που παράγεται στο Second Life μπορεί να αναπαραχθεί κανονικά κι έτσι δεν απαιτείται η θέαση του Second Life ώστε να χρησιμοποιηθεί η πλακέτα Arduino.

Για να κλείσει το πρόγραμμα απαιτείται απλά να πατηθεί το «X» για να κλείσει το παράθυρο ενώ η πλακέτα Arduino μπορεί να αποσυνδεθεί οποιαδήποτε στιγμή χωρίς πρόβλημα.

## 5.5 ΛΕΙΤΟΥΡΓΙΑ

Για να γίνει δυνατή η επικοινωνία της πλακέτας Arduino με το Second Life, χρησιμοποιείται ένας Web Browser που φτιάχτηκε από το μηδέν. Ο συγκεκριμένος Web Browser διέπει μια σελίδα HTML σχεδόν ολόιδια και με την ίδια αρχή λειτουργίας με αυτή των κεφαλαίων 3,4. Έτσι, η πλακέτα που συνδέεται με τον Web Browser, δεν συνδέεται άμεσα με το Second Life, αλλά λειτουργεί ως χειριστήριο ενός περιηγητή. Η δημιουργία καινούργιου Web Browser χρειάστηκε καθώς οι συμβατικοί ( Chrome, Mozilla κ.α.) δε δέχονται ως είσοδο χειρισμού σειριακά δεδομένα που είναι και τα δεδομένα που παράγει η Arduino. Έτσι χρειάστηκε η δημιουργία περιηγητή ο οποίος να μπορεί να δέχεται σειριακά δεδομένα και να τα χρησιμοποιεί για να περιηγείται στο διαδίκτυο.

### 5.5.1 ΤΡΟΠΟΣ ΛΕΙΤΟΥΡΓΙΑΣ ARDUINO

Τα κουμπιά που έχουν εγκατασταθεί στο breadboard, επικοινωνούν με την πλακέτα του Arduino μέσω των ψηφιακών pin 3 έως 7. Ο κώδικας του «BUTTONS.ino» ορίζει 9600baud στη σειριακή επικοινωνία. Αυτό σημαίνει πως μεταφέρονται το μέγιστο 9600 bit πληροφορίας το δευτερόλεπτο στο κανάλι επικοινωνίας[10]. Τα δεδομένα που μεταφέρονται είναι ένα γράμμα για κάθε κουμπί που πατήθηκε (R για το κουμπί με το κόκκινο καλώδιο, G για το κουμπί με το πράσινο καλώδιο, S για το κουμπί με το λευκό καλώδιο, B για το κουμπί με το μπλε καλώδιο και Y για το κουμπί με το κίτρινο καλώδιο). Σημαντικό είναι εδώ να τονισθεί πως ο υπολογιστής και η πλακέτα θα πρέπει να έχουν διαμορφωθεί στο ίδιο «baud rate» ώστε να επικοινωνούν αποτελεσματικά. Η ρύθμιση της πλακέτας γίνεται με την

εντολή «Serial.begin(9600)» ενώ στον υπολογιστή η ρύθμιση γίνεται κατά την ανάπτυξη του Simon Browser στο Visual Studio.

Το κανάλι επικοινωνίας που χρησιμοποιεί η πλακέτα για να επικοινωνήσει με τον υπολογιστή μέσω καλωδίου USB είναι το «COM3». COM (Communication Port) ονομάζεται η φυσική σειριακή θύρα στους υπολογιστές. Οι συγκεκριμένες θύρες ονομάζονται σειριακές καθώς τα δεδομένα μεταφέρονται 1 bit τη φορά[11]. Φυσικές σειριακές θύρες χρησιμοποιούνται και με τις USB θύρες.

### 5.5.2 ΤΡΟΠΟΣ ΛΕΙΤΟΥΡΓΙΑΣ SIMON BROWSER

Το πρόγραμμα Simon Browser έχει αναπτυχθεί στο πρόγραμμα Microsoft Visual Studio 2013. Εκεί δημιουργήθηκε μια φόρμα (Form) η οποία περιλαμβάνει έναν Web Browser, χωρίς να οριστεί συγκεκριμένο URL που θα διέπει, αντιθέτως η εισχώρηση διεύθυνσης θα επιτυγχάνεται από τον χρήστη. Αυτή η εισχώρηση διεύθυνσης υλοποιείται με τη χρήση πλαισίου κειμένου (Text Box) στο οποίο ο χρήστης μπορεί να πληκτρολογεί οποιαδήποτε διεύθυνση. Για το φόρτωμα εκάστοτε ιστοσελίδας προστέθηκε κουμπί «GO» το οποίο δίνει εντολή στον Web Browser να κατευθυνθεί στη διεύθυνση που πληκτρολόγησε ο χρήστης.

Όμως, με τα προηγούμενα βήματα δημιουργήθηκε ένας φυσιολογικός περιηγητής διαδικτύου που δεν επιλύει το πρόβλημα ανάγνωσης σειριακών δεδομένων. Για αυτόν τον σκοπό προστέθηκε αρχικά το εργαλείο «Serial Port». Εκεί ορίζεται το BaudRate στα 9600, ίδιο με της πλακέτας Arduino ενώ το όνομα της σειριακής πύλης ορίζεται το «COM3», ίδιο με της πλακέτας Arduino.

Ωστόσο, για την επιτυχή ανάγνωση των σειριακών δεδομένων χρησιμοποιείται ένα χρονόμετρο «Timer» στα 50ms ώστε να μπορεί ο περιηγητής να διαβάζει διαφορετικά δεδομένα που καταφθάνουν σειριακά. Άρα, σε επίπεδο λογισμικού μπορεί να αναγνωστεί ένα διαφορετικό πάτημα κουμπιού στην πλακέτα ανά 50ms. Επειδή αποστέλλονται χαρακτήρες από την πλακέτα Arduino, το μέγεθος των δεδομένων είναι 8bit ο κάθε χαρακτήρας, ενώ η χωρητικότητα του καναλιού επιτρέπει  $9600 \text{ baudrate} / 1000\text{ms} = 9.6 \text{ bit/ms}$ . Έτσι, δεν υπάρχει κίνδυνος απώλειας δεδομένων καθώς προστίθεται επίσης, στην πλακέτα Arduino καθυστέρηση (delay) 10ms εντός του κώδικα για αποστολή κάθε δεδομένου. Ωστόσο απώλεια δεδομένων δεν επιτρέπεται γενικότερα εξαιτίας τη μη δυνατότητας του ανθρώπου να πατήσει τόσο γρήγορα πολλαπλά κουμπιά (ο μέσος μέγιστος αριθμός «κλικ» που μπορεί να πατήσει ένας άνθρωπος εντός ενός δευτερολέπτου είναι περίπου 20, άρα 1 ανά 50 ms)[12].



Τέλος, στον διαδικτυακό περιηγητή προστέθηκε ακόμα ένα κουμπί «Simon» το οποίο προσθέτει αυτόματα τη διεύθυνση της ιστοσελίδας:

«<http://siwonsimon.000webhostapp.com/ARDUINO.html?>» για ευκολότερη πρόσβαση σε αυτήν.

Υπενθυμίζεται ότι μετά τον χαρακτήρα «?» προστίθεται το URL που παράχθηκε στο Second Life.

## 5.6 ΣΥΝΟΨΗ ΚΕΦΑΛΑΙΟΥ

Σε αυτό το κεφάλαιο εξετάστηκε η έμμεση επικοινωνία μιας πλακέτας Arduino με το project που λαμβάνει χώρα στο Second Life.

Το είδος της επικοινωνίας που χρησιμοποιείται μεταξύ του μικροελεγκτή και του Second Life έχει κι εδώ ως βάση την HTTP επικοινωνία. Ωστόσο απαιτήθηκαν επιπλέον βήματα για την ανάγνωση και την διαβίβαση των δεδομένων που παράγονται στην πλακέτα Arduino καθώς αυτά μεταφέρονται σε έναν υπολογιστή με σειριακό τρόπο.

Οπότε χρειάστηκε η δημιουργία τεχνητού διαδικτυακού περιηγητή για την ανάγνωση και επεξεργασία σειριακών δεδομένων, κάτι το οποίο δεν έχουν τη δυνατότητα να καταφέρουν οι απλοί περιηγητές (Chrome, Mozilla κλπ.).

Για τη δημιουργία τέτοιου διαδικτυακού περιηγητή χρησιμοποιήθηκε το πρόγραμμα Microsoft Visual Studio 2013 το οποίο επέτρεψε την ανάπτυξη του με συγκεκριμένα χαρακτηριστικά τα οποία απαιτούνταν.

## 5.7 ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1]. Arduino Store, Arduino Uno, <https://store.arduino.cc/arduino-uno-rev3>.
- [2]. En. Wikipedia, Microsoft Visual Studio, [https://en.wikipedia.org/wiki/Microsoft\\_Visual\\_Studio](https://en.wikipedia.org/wiki/Microsoft_Visual_Studio), 2019.
- [3]. Visual Studio 2005 Documentation, Debugger, <https://www.microsoft.com/en-us/download/details.aspx?id=55984>.
- [4]. Bind controls to data in Visual Studio, <https://docs.microsoft.com/el-gr/visualstudio/data-tools/bind-controls-to-data-in-visual-studio?view=vs-2015&redirectedfrom=MSDN>, 2016.
- [5]. En. Wikipedia, Event-driven programming, [https://en.wikipedia.org/wiki/Event-driven\\_programming](https://en.wikipedia.org/wiki/Event-driven_programming), 2019.

- [6]. En. Wikipedia, Visual Basic .NET, [https://en.wikipedia.org/wiki/Visual\\_Basic\\_.NET](https://en.wikipedia.org/wiki/Visual_Basic_.NET), 2019
- [7]. En. Wikipedia, .NET Framework, [https://en.wikipedia.org/wiki/.NET\\_Framework](https://en.wikipedia.org/wiki/.NET_Framework), 2019
- [8]. Arduino, Digital Pins, <https://www.arduino.cc/en/Tutorial/DigitalPins>.
- [9]. Arduino, Software, <https://www.arduino.cc/en/Main/software>.
- [10]. MathWorks, BaudRate, [https://www.mathworks.com/help/matlab/matlab\\_external/baudrate.html](https://www.mathworks.com/help/matlab/matlab_external/baudrate.html).
- [11]. WeboPedia, Serial Port, [https://www.webopedia.com/TERM/S/serial\\_port.html](https://www.webopedia.com/TERM/S/serial_port.html).
- [12]. Click Test, Click per Second, <https://www.click-test.com/p/game.html>.

## ΚΕΦΑΛΑΙΟ 6: Συμπεράσματα

Η επιτυχία της συγκεκριμένης διπλωματικής εργασίας είναι εύκολο να κριθεί καθώς η αποτελεσματικότητα της μπορεί να αποφασιστεί από τον βαθμό στον οποίο το τελικό προϊόν που παράχθηκε, λειτουργεί ή όχι.

Το παιχνίδι Simon Says.v3 το οποίο αναπτύχθηκε σε περιβάλλον Second Life, έγινε διαθέσιμο στο διαδικτυακό κατάστημα του Second Life, με μηδενική τιμή απόκτησης. Αυτό επέτρεψε στο προϊόν να αποκτηθεί από αρκετούς χρήστες του Second Life (10-15 άτομα εντός 2 εβδομάδων), ενώ τα ελεύθερα δικαιώματα αντιγραφής του κώδικα και παραμετροποίησης του έγιναν αντικείμενο συζήτησης μερικών παικτών σε σημείο μάλιστα να επικοινωνήσουν και με τον δημιουργό της συγκεκριμένης εργασίας, ώστε να τον συμβουλευτούν και να του προτείνουν αλλαγές ή άλλες ιδέες.

Το παιχνίδι ωστόσο Simon Says.v3 επιτρέπει και την εξ' αποστάσεως από το πρόγραμμα Second Life, πλοήγηση στο παιχνίδι. Τόσο η χρήση εντός του Second Life όσο και με την λειτουργία απομακρυσμένης απόκρισης, το παιχνίδι λειτουργούσε κανονικά χωρίς απώλεια δεδομένων.

Παρόλα αυτά, στην απομακρυσμένη λειτουργία, εμφανίζεται μικρή καθυστέρηση στην εμφάνιση των δεδομένων στην εκάστοτε συσκευή. Το συγκεκριμένο πρόβλημα, εν μέρει οφείλεται στη χρήση συναρτήσεων καθυστέρησης εντός της HTML σελίδας αλλά βάσει δοκιμών που έγιναν, μεγαλύτερο ρόλο στην εμφάνιση καθυστέρησης φαίνεται να κατέχει ο Host Server στον οποίο είναι αποθηκευμένα τα αρχεία διαδικτύου.

Είναι λογικό ως ένα σημείο η εμφάνιση καθυστέρησης εξαιτίας της HTTP επικοινωνίας και της PHP εγγραφής, ανάλογα την ταχύτητα του διαδικτύου, όμως καθ' όλη τη διάρκεια της ανάπτυξης της εργασίας η υπηρεσία Host παρουσίαζε προβλήματα καθυστέρησης αλλά και σύνδεσης FTP. Συνίσταται μελλοντικά η επιλογή Host με βελτιωμένα χαρακτηριστικά, τα οποία όμως έχουν και το ανάλογο χρηματικό κόστος. Χρηματική συναλλαγή για την εξασφάλιση καλύτερων συνθηκών Host ωστόσο υλοποιήθηκε όμως χωρίς κάποιο αποτέλεσμα καθώς η υπηρεσία 000webhost αποδείχθηκε κατώτερη των περιστάσεων τόσο στις υπηρεσίες της όσο και στην καθοδήγηση του καταναλωτή για καλύτερη εμπειρία χρήσης.

Ακόμη, η συναρμολόγηση ενός χειριστηρίου Arduino για την έμμεση επικοινωνία με την πλατφόρμα του Second Life στέφθηκε με επιτυχία αλλά και στην περίπτωση του Web Browser που αναπτύχθηκε ειδικά για την ανάγνωση και διαβίβαση των δεδομένων στο Second Life, παρατηρείται μια μεγάλη

καθυστέρηση στο «φόρτωμα» της απαραίτητης σελίδας HTML για την επικοινωνία, με πιθανότερο λόγο εμφάνισης και πάλι την αναξιοπιστία της υπηρεσίας Host.

Τέλος, παρατηρήθηκε δυσκολία ανάπτυξης επιθυμητού περιηγητή διαδικτύου καθώς η ήδη υλοποιημένη σελίδα HTML «<http://siwonsimon.000webhostapp.com/siwonsimon.html>» δε μπορούσε να εμφανιστεί κανονικά στον νέο περιηγητή. Η λύση δόθηκε μετά από πολλές δοκιμές καθώς παρατηρήθηκε αδυναμία φόρτωσης ορισμένων μόνο CSS γραφικών σε αντικείμενα <div> της HTML. Η λύση που προέκυψε ήταν η αντικατάσταση του συγκεκριμένου αντικειμένου με <button> αντικείμενα, με την αλλαγή ωστόσο να επηρεάζει το αισθητικό τομέα της σελίδας. Η νέα σελίδα που χρησιμοποιείται αποκλειστικά για την πλακέτα ARDUINO είναι η «<https://siwonsimon.000webhostapp.com/ARDUINO.html>».

## ΕΠΙΛΟΓΟΣ

Η απομακρυσμένη μεταφορά δεδομένων και η διαχείριση τους σε διαφορετικά περιβάλλοντα υπό ποικίλες συνθήκες, αποτελεί σημαντικό κομμάτι της ανάπτυξης νέων τεχνολογιών καθώς διέπει τη βασική δομή όλων των τεχνολογιών, την πληροφορία. Ωστόσο σημαντικό εργαλείο στη διαχείριση των δεδομένων είναι η οπτικοποίηση τους και η επίτευξη αλληλεπίδρασης της μηχανής με τον άνθρωπο με όσο το δυνατόν φυσικότερο τρόπο.