



ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ  
ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ  
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ & ΥΠΟΛΟΓΙΣΤΩΝ  
ΡΟΗ ΛΟΓΙΣΜΙΚΟΥ ΚΑΙ ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

## ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΑΝΑΠΤΥΞΗ ΕΚΠΑΙΔΕΥΤΙΚΟΥ ΠΑΙΧΝΙΔΙΟΥ ΚΥΚΛΟΦΟΡΙΑΚΗΣ ΑΓΩΓΗΣ

### **“Traffic Education Simulator 2021”**

του φοιτητή

**Σιαφάκα Παρασκευά-Σωκράτη (711151083)**

Επιβλέπων Καθηγητής  
**Γεώργιος Μπαρδής, Επίκουρος Καθηγητής**

Αιγάλεω, Οκτώβριος 2021





ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ  
ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ  
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ & ΥΠΟΛΟΓΙΣΤΩΝ  
ΡΟΗ ΛΟΓΙΣΜΙΚΟΥ ΚΑΙ ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

## ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΑΝΑΠΤΥΞΗ ΕΚΠΑΙΔΕΥΤΙΚΟΥ ΠΑΙΧΝΙΔΙΟΥ ΚΥΚΛΟΦΟΡΙΑΚΗΣ ΑΓΩΓΗΣ

### “Traffic Education Simulator 2021”

Ον/μο: **Σιαφάκας Παρασκευάς-Σωκράτης**

Αριθμός Μητρώου: **711151083**

Επιβλέπων Καθηγητής: **Γεώργιος Μπαρδής, Επίκουρος Καθηγητής**

Εγκρίθηκε από την 3μελή εξεταστική επιτροπή των:

--	--	--

Γεώργιος Μπαρδής  
Επίκουρος Καθηγητής

Αθανάσιος Βουλόδημος  
Επίκουρος Καθηγητής

Παναγιώτα Τσελέντη  
Ε.ΔΙ.Π.

Αιγάλεω, Οκτώβριος 2021







**UNIVERSITY OF WEST ATTICA  
SCHOOL OF ENGINEERING  
DEPARTMENT OF INFORMATICS AND COMPUTER  
ENGINEERING  
SOFTWARE AND INFORMATION SYSTEMS FLOW**

# **DESIGN AND DEVELOPMENT OF A TRAFFIC EDUCATION COMPUTER GAME**

## **“Traffic Education Simulator 2021”**

**Full Name: Sifakas Paraskevas-Sokratis**

**ID Number: 71151083**

**Supervisor: Georgios Bardis, Assistant Professor**

Egaleo, October 2021



## ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ

Ο κάτωθι υπογεγραμμένος **Σιαφάκας Παρασκευάς-Σωκράτης**, με αριθμό μητρώου **711151083**, φοιτητής του **Πανεπιστημίου Δυτικής Αττικής της Σχολής Μηχανικών του Τμήματος Πληροφορικής και Υπολογιστών**, δηλώνω υπεύθυνα ότι:

«Είμαι συγγραφέας αυτής της διπλωματικής εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, οι όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε ακριβώς είτε παραφρασμένες, αναφέρονται στο σύνολό τους, με πλήρη αναφορά στους συγγραφείς, τον εκδοτικό οίκο ή το περιοδικό, συμπεριλαμβανομένων και των πηγών που ενδεχομένως χρησιμοποιήθηκαν από το διαδίκτυο. Επίσης, βεβαιώνω ότι αυτή η εργασία έχει συγγραφεί από μένα αποκλειστικά και αποτελεί προϊόν πνευματικής ιδιοκτησίας τόσο δικής μου, όσο και του Ιδρύματος. Παράβαση της ανωτέρω ακαδημαϊκής μου ευθύνης αποτελεί ουσιώδη λόγο για την ανάκληση του πτυχίου μου»

Ο Δηλών,

Σιαφάκας Παρασκευάς-Σωκράτης



(υπογραφή)



## Περίληψη

Κατά της διάρκεια της συναναστροφής του με τον έξω κόσμο, είναι δεδομένο ότι ο άνθρωπος οφείλει να τηρεί κάποιους βασικούς κανόνες, οι οποίοι εξασφαλίζουν την ασφάλειά του ίδιου και των γύρω του. Ένα κομμάτι αυτών των κανόνων είναι αυτοί που απαρτίζουν τη **κυκλοφοριακή αγωγή**. Από πολύ μικρή ηλικία, οι άνθρωποι στο ρόλο του μαθητή ενημερώνονται και εκπαιδεύονται για τους βασικούς κανόνες κυκλοφοριακής αγωγής, κάτι που εξασφαλίζει την ασφάλειά τους στη καθημερινή ζωή ως πεζοί και ως επιβάτες ενός οχήματος. Οι προσπάθειες αυτές έχουν συσπειρωθεί γύρω από ενημερώσεις των μαθητών μέσω προφορικού λόγου, **διαδραστικών ψηφιακών μέσων** και δημιουργία φορητών ή μονίμων πάρκων κυκλοφοριακής αγωγής.

Με τη τεχνολογία να γίνεται αναπόσπαστο κομμάτι των ανθρώπων σε όλο και μικρότερη ηλικία, είναι αναγκαίο να προσαρμόσουμε αυτές τις προσπάθειες στις τάξεις της σύγχρονης κοινωνίας. Με αυτό κατά νου, θα εστιάσουμε στα **ψηφιακά μέσα** και κυρίως στα **βιντεοπαιχνίδια**. Παρόλο που τα **εκπαιδευτικά βιντεοπαιχνίδια** δεν είναι τίποτα καινούργιο στη ψηφιοποιημένη κοινωνία του σήμερα, κατά ικανό ποσοστό έχουν αποδειχθεί είτε μη ικανά ως προς το σκοπό που καλούνται να επιτελέσουν, είτε υστερούν ενδιαφέροντος και προσοχής από το κοινό στο οποίο απευθύνονται.

Συνεπώς, καλούμαστε να ερευνήσουμε τι οφείλει να περιέχει ένα εκπαιδευτικό βιντεοπαιχνίδι κυκλοφοριακής αγωγής σε συνδυασμό με το πως είναι δυνατόν να κεντρίσει το ενδιαφέρον των μαθητών στο μέγιστο δυνατό βαθμό. Παράλληλα, θα εμβαθύνουμε στις **τεχνολογίες** που θα μας βοηθήσουν στην υλοποίηση του έργου αυτού, εξετάζοντας και παρουσιάζοντας πως λειτουργούν και το ρόλο που επιτελούν. Τέλος, θα προσπαθήσουμε να αναλύσουμε πως η εφαρμογή ενός τέτοιου μέσου θα είναι επιτυχής, τονίζοντας ποιος θα πρέπει να είναι ο ρόλος των δασκάλων στη διαδικασία, πως οι μαθητές θα ενημερωθούν επιτυχώς και θα ψυχαγωγηθούν στο βαθμό που απαιτείται και εν τέλει, ποιος θα είναι ο μακροπρόθεσμος ρόλος μια τέτοιας προσπάθειας στο χώρο της **εκπαίδευσης** και πως μπορεί να εξελιχθεί.

### Λέξεις Κλειδιά

Κυκλοφοριακή Αγωγή, Εκπαιδευτικά Βιντεοπαιχνίδια, Διαδραστικά Ψηφιακά Μέσα, Μαθητές - Εκπαιδευτικοί, Τεχνολογίες Ανάπτυξης Βιντεοπαιχνιδιών



# Abstract

During their interaction with the outside world, it is a known fact that all human beings owe to obey a number of basic rules that ensure the safety of them and their fellow human beings. A portion of these rules constitute the code of **traffic education**. From a very young age, people are informed and educated about the basic rules of traffic education as students, something that ensures their safety in their daily life as pedestrians or as passengers of a vehicle. These efforts have been concentrated around informing students through speeches, **interactive digital media** and through the creation of portable or permanent traffic education parks.

With technology becoming an integral part of humanity in a gradually younger age, it is important for us to adjust the aforementioned efforts in the trends of today's society. With that in mind, we will focus on **digital media** and primarily **video games**. Despite **educational video games** being nothing new in today's digitalized society, they have been statistically proven to be either inefficient towards the goal they are called to accomplish, or they lack the interest and attention from the audience they are oriented towards.

Therefore, we are called to research what an educational video game owes to include in combination with how it is possible for it to trigger the interest of the students to the highest possible degree. Meanwhile, we will delve deeply into the **technologies** that will help us in the implementation of this project, examining and demonstrating how they work and what is the role they accomplish. Finally, we will attempt to analyze how the implementation of a project like this will be successful, emphasizing what is the role of the teachers in the whole process, how the students will be informed successfully and will be entertained to the required degree and eventually, which is going to be the long-term role of an effort like this in **education** as a whole and how it can be evolved.

## Keywords

Traffic Education, Educational Video Games, Interactive Digital Media, Students - Teachers, Gaming Development Technologies





## Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου κύριο Γεώργιο Μπαρδή για την υποστήριξή του καθ' όλη τη διάρκεια ανάπτυξης αυτής της διπλωματικής εργασίας, καθώς και για την εμπιστοσύνη που μου έδειξε. Επιπλέον θα ήθελα να ευχαριστήσω την οικογένεια και τους φίλους μου για την υπομονή και τη κατανόηση που επέδειξαν κατά τη διάρκεια των σπουδών μου και τέλος θα ήθελα να ευχαριστήσω όλα αυτά τα άτομα που μου έδωσαν το κίνητρο να αγκαλιάσω τη σχολή και να προσπαθήσω περισσότερο για το μέλλον μου.

# Πίνακας Περιεχομένων

<b>Περίληψη</b>	<b>9</b>
<b>Abstract</b>	<b>11</b>
<b>Ευχαριστίες</b>	<b>13</b>
<b>Πίνακας Περιεχομένων</b>	<b>14</b>
<b>Κατάλογος Εικόνων</b>	<b>16</b>
<b>1. Εισαγωγή</b>	<b>17</b>
1.1 Περιγραφή Προβλήματος - Στόχος Εργασίας	17
1.2 Δομή Εργασίας	18
<b>2. Θεωρητικό Μέρος</b>	<b>19</b>
2.1 Βιντεοπαιχνίδια	19
2.1.1 Βασικές Κατηγορίες Βιντεοπαιχνιδιών	19
2.1.2 Σοβαρά Βιντεοπαιχνίδια	19
2.1.3 Κατηγορίες Σοβαρών Βιντεοπαιχνιδιών	21
2.1.4 Δομή Σοβαρών Βιντεοπαιχνιδιών	21
2.1.5 Το Μοντέλο G/P/S (Gameplay/Purpose/Scope Model)	22
2.2 Κυκλοφοριακή Αγωγή	23
2.2.1 Στατιστικά Στοιχεία	24
2.2.2 Η Κυκλοφοριακή Αγωγή Στην Εκπαίδευση	25
2.2.3 Κυκλοφοριακή Αγωγή και Βιντεοπαιχνίδια	26
2.3 Τεχνολογίες Ανάπτυξης Βιντεοπαιχνιδιών	29
2.3.1 Μηχανές Παιχνιδιών (Game Engines)	29
2.3.2 Γραφικά Υπολογιστών (Computer Graphics)	31
2.3.3 Η Γλώσσα Προγραμματισμού C#	33
<b>3. Ανάπτυξη Βιντεοπαιχνιδιού</b>	<b>35</b>

3.1 Περιγραφή	35
3.2 Πορεία Ανάπτυξης Παιχνιδιού	35
3.2.1 Δημιουργία Περιβάλλοντος Βιντεοπαιχνιδιού	36
3.2.2 Δημιουργία Αυτοκινήτου Παίκτη	39
3.2.3 Δημιουργία Μενού Παιχνιδιού	41
3.2.4 Δημιουργία Επιπέδων	46
<b>4. Εκτέλεση Βιντεοπαιχνιδιού &amp; Παρατηρήσεις</b>	<b>52</b>
4.1 Περιγραφή Επιπέδων	52
4.2 Προβλήματα Που Συναντήθηκαν	54
<b>5. Αξιολόγηση &amp; Προτάσεις Βελτίωσης</b>	<b>56</b>
5.1 Προσωπική Αξιολόγηση	56
5.2 Αξιολόγηση Από Τρίτους	57
5.3 Προτάσεις Βελτίωσης	58
<b>Βιβλιογραφία</b>	<b>59</b>
<b>Παράρτημα Κώδικα</b>	<b>61</b>

# Κατάλογος Εικόνων

<b>Εικόνα 2.1</b> Pong(1972)	21
<b>Εικόνα 2.2</b> Pulse!!(2007) - Ένα παράδειγμα σοβαρού βιντεοπαιχνιδιού	23
<b>Εικόνα 2.3</b> Σχεδιάγραμμα Ταξινόμησης που αναπαριστά το μοντέλο G/P/S	25
<b>Εικόνα 2.4</b> Στατιστικά στοιχεία θανάτων στους δρόμους από τον WHO (2018)	27
<b>Εικόνα 2.5</b> Παράδειγμα μιας τρισδιάστατης σκηνής	35
<b>Εικόνα 3.1</b> Η default σκηνή του Unity	38
<b>Εικόνα 3.2</b> Κάτοψη του περιβάλλοντος έπειτα από τη δημιουργία του οδικού δικτύου	39
<b>Εικόνα 3.3</b> Το μενού ρυθμίσεων κάθε αντικειμένου	38
<b>Εικόνα 3.4</b> Μια διασταύρωση με σηματοδότες	40
<b>Εικόνα 3.5</b> Κάτοψη του περιβάλλοντος με τα κτίρια που έχουν προστεθεί	40
<b>Εικόνα 3.6</b> Το αυτοκίνητο του παίκτη σε ακίνητη κατάσταση	41
<b>Εικόνα 3.7</b> Παράδειγμα collider τροχού του αυτοκινήτου	41
<b>Εικόνα 3.8</b> Η θέα του παίκτη έπειτα από την εφαρμογή του CameraFollow	42
<b>Εικόνα 3.9</b> Το αυτοκίνητο του παίκτη σε στιγμιότυπο περιστροφής	43
<b>Εικόνα 3.10</b> Η λειτουργία OnClick() του Unity	44
<b>Εικόνα 3.11</b> Το μενού επιλογής επιπέδου του παιχνιδιού	45
<b>Εικόνα 3.12</b> Το μενού παύσης του παιχνιδιού	46
<b>Εικόνα 3.13</b> Το πλαίσιο καλωσορίσματος του παιχνιδιού	47
<b>Εικόνα 3.14</b> Το πλαίσιο περιγραφής επιπέδου του παιχνιδιού	47
<b>Εικόνα 3.15</b> Το πλαίσιο ολοκλήρωσης επιπέδου του παιχνιδιού	48
<b>Εικόνα 3.16</b> Το πλαίσιο αποστολών του παιχνιδιού	49
<b>Εικόνα 3.17</b> Ένας ανιχνευτής συγκρούσεων	50
<b>Εικόνα 3.18</b> Μοντέλο πεζού στην αρχική του κατάσταση	51
<b>Εικόνα 3.19</b> Στιγμιότυπο πεζών που περπατάνε στη πόλη	52
<b>Εικόνα 3.20</b> Στιγμιότυπο οχημάτων που κυκλοφορούν στη πόλη	52
<b>Εικόνα 3.21</b> Waypoint Navigator	53
<b>Εικόνα 4.1</b> Παράδειγμα αυτοκινήτου που παρεκτρέπεται από τη πορεία του	57

# 1. Εισαγωγή

Η βιομηχανία των παιχνιδιών, από την ίδρυσή της μέχρι σήμερα, όχι μόνο έχει καθιερωθεί ριζικά στις ζωές των ανθρώπων σαν ένα χόμπι, αλλά έχει εξελιχθεί σε τέτοιο βαθμό που μπορεί να διηγηθεί μία ιστορία ή να διδάξει ένα σημαντικό μάθημα με την ίδια ή ακόμα και μεγαλύτερη απόδοση από ένα βιβλίο ή ένα ντοκιμαντέρ. Αυτό δεν πέρασε απαρατήρητο από αρκετούς επαγγελματικούς και εκπαιδευτικούς φορείς, που άρχισαν να βλέπουν τα βιντεοπαιχνίδια ως ένα εργαλείο μετάδοσης γνώσης που θα ήταν πιο προσιτό και φιλικό για την επόμενη γενιά μαθητών και εργαζομένων. Έτσι, η κατηγορία των σοβαρών παιχνιδιών δημιουργήθηκε και άρχισε να εξελίσσεται με σταθερά βήματα, προσφέροντας περιεχόμενο αρχικά σε πειραματικό βαθμό και έπειτα σε ενισχυτικό βαθμό, κάτι που ισχύει μέχρι σήμερα. Δεν είναι λίγοι οι φορείς οι οποίοι ενώ ακολουθούν το παραδοσιακό τρόπο εκπαίδευσης και ενημέρωσης των εργαζομένων ή των μαθητών τους (χρήση εκπαιδευτών, βιβλίων και παρουσιάσεων), αρχίζουν σιγά σιγά να δοκιμάζουν τις γνώσεις που έχουν αποκτηθεί με τη χρήση ενός βιντεοπαιχνιδιού ως ενδιάμεσο βήμα ανάμεσα στη θεωρία και τη πράξη, χωρίς όμως αυτό να σημαίνει ότι δεν υπάρχουν φορείς που έχουν εναποθέσει την εμπιστοσύνη τους εξ ολοκλήρου στα σοβαρά βιντεοπαιχνίδια για να αντικαταστήσουν όλα τα παραπάνω.

Ειδικότερα στο χώρο της παιδείας, όπου η ραγδαία εξέλιξη της τεχνολογίας έχει βρει σπίτι στη καθημερινότητα των νέων σε όλο και μικρότερη ηλικία, δεν είναι λίγες οι προσπάθειες που έχουν γίνει ώστε η διεξαγωγή του μαθήματος να ξεφύγει από το στερεότυπο που θέλει τον μαθητή ως ένα παθητικό δέκτη γνώσης σε μια πιο διαδραστική εμπειρία που θα δίνει την ευκαιρία στον μαθητή να παίρνει πρωτοβουλίες, αλλά και να απορροφά την απαραίτητη γνώση σε ένα περιβάλλον που είναι πιο οικείο σε αυτόν. Ένα τέτοιο παράδειγμα είναι οι ενημερώσεις των μαθητών για θεμελιώδη θέματα που επηρεάζουν τη καθημερινή τους ζωή, όπως η κυκλοφοριακή αγωγή. Ενώ στο παρελθόν ήταν ευθύνη της τροχαίας και των δασκάλων να ενημερώνουν τους μαθητές μέσω ομιλιών και παρουσιάσεων, πλέον δεν είναι λίγες οι διαδραστικές πηγές στο διαδίκτυο και κυρίως σε ακαδημαϊκές πλατφόρμες που προσπαθούν να πετύχουν το ίδιο μέσω απλών και δημιουργικών βιντεοπαιχνιδιών.

## 1.1 Περιγραφή Προβλήματος - Στόχος Εργασίας

Παρόλο που όπως αναφέρθηκε η παρουσία των σοβαρών βιντεοπαιχνιδιών στην εκπαίδευση δεν είναι κάτι καινούργιο, έχει παρατηρηθεί ότι αρκετές φορές δεν καταφέρνουν να κερδίσουν τη προσοχή των μαθητών. Αυτό συμβαίνει είτε επειδή στο παιχνίδι δεν έχει δοθεί αρκετή προσοχή στο κομμάτι της διασκέδασης ή επειδή οι μαθητές δεν μπορούν να ταυτιστούν αρκετά με το τρόπο εκπαίδευσης που τους παρέχεται. Συνεπώς, ο στόχος μας είναι η δημιουργία ενός σοβαρού εκπαιδευτικού βιντεοπαιχνιδιού που να αφορά τη κυκλοφοριακή αγωγή δίνοντας έμφαση στις σύγχρονες απαιτήσεις που έχουν οι μαθητές από ένα βιντεοπαιχνίδι. Εντός αυτού, θα προσπαθήσουμε να εντάξουμε όσες περισσότερες τεχνικές σχεδίασης μπορούμε (π.χ. δημιουργία μοναδικών μονοπατιών που θα

χρησιμοποιούνται αυτόματα από πεζούς και αυτοκίνητα εντός του περιβάλλοντος), ενώ παράλληλα θα προσπαθήσουμε να βρούμε τη τέλεια ισορροπία ανάμεσα στο εκπαιδευτικό περιεχόμενο και το περιεχόμενο που υπάρχει καθαρά για ψυχαγωγία. Τέλος, θα προσπαθήσουμε να αξιολογήσουμε τη προσπάθειά μας, να προτείνουμε τρόπους βελτίωσης, αλλά και να επιτρέψουμε σε τρίτους (μαθητές Δημοτικού) να δοκιμάσουν το παιχνίδι και να δώσουν τη δική τους αξιολόγηση.

## **1.2 Δομή Εργασίας**

Η εν λόγω διπλωματική εργασία είναι χωρισμένη σε πέντε κεφάλαια. Το πρώτο κεφάλαιο απαρτίζεται από μια γενική εισαγωγή στην εργασία, προετοιμάζοντας τον αναγνώστη για το θέμα με το οποίο πραγματεύεται. Στο δεύτερο κεφάλαιο βρίσκεται το θεωρητικό μέρος της εργασίας, που δίνει στον αναγνώστη όλες τις απαραίτητες βασικές γνώσεις για να κατανοήσει τη τρόπο δημιουργίας του βιντεοπαιχνιδιού που θα συναντηθεί στο επόμενο κεφάλαιο. Έτσι, στο τρίτο κεφάλαιο περιγράφεται αναλυτικά η φιλοσοφία πίσω από την ανάπτυξη του βιντεοπαιχνιδιού, ενώ κάθε πτυχή του εξηγείται βήμα βήμα. Στο τέταρτο κεφάλαιο αναφερόμαστε στην εκτέλεση του βιντεοπαιχνιδιού και στις παρατηρήσεις οι οποίες σημειώθηκαν, με ιδιαίτερη αναφορά στα προβλήματα που συναντήθηκαν. Στο πέμπτο κεφάλαιο γίνεται μια αντικειμενική προσωπική αξιολόγηση της εργασίας και αναφέρονται οι αξιολογήσεις που έγιναν από τρίτους στο βιντεοπαιχνίδι. Επιπλέον, αναφέρονται προσωπικές απόψεις για το πως μπορεί να βελτιωθεί το τελικό προϊόν σε μελλοντικές προσπάθειες. Τέλος, σε παράρτημα θα υπάρχουν κάποια χρήσιμα κομμάτια κώδικα που συνδέονται με το τρίτο κεφάλαιο και θα δίνουν τη δυνατότητα στον αναγνώστη να τα αναλύσει αλλά και να κάνει και ο ίδιος τη δική του προσπάθεια βασιζόμενος σε αυτά.

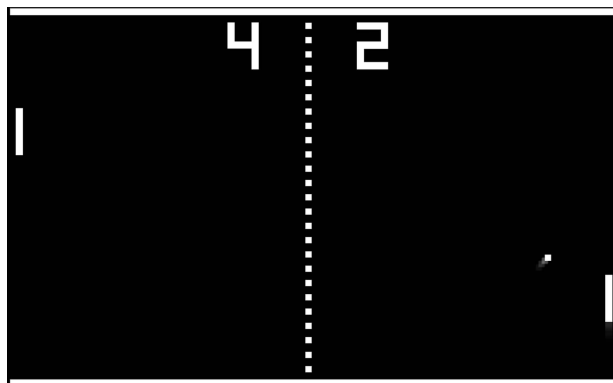
## 2. Θεωρητικό Μέρος

### 2.1 Βιντεοπαιχνίδια

Ως βιντεοπαιχνίδι (video game), ορίζουμε ένα ηλεκτρονικό παιχνίδι που εμπλέκει την ανθρώπινη αλληλεπίδραση με μια διεπαφή χρήστη με σκοπό τη παραγωγή ψηφιακής αναπληροφόρησης σε μια συσκευή αναπαραγωγής εικόνας. Σε αυτή τη κατηγορία υπάγονται όλες οι συσκευές προβολής που μπορούν να παράγουν δισδιάστατες ή τρισδιάστατες εικόνες. Τα ηλεκτρονικά συστήματα που χρησιμοποιούνται για την αναπαραγωγή βιντεοπαιχνιδιών είναι γνωστά ως πλατφόρμες. Με τον όρο “πλατφόρμα”, αναφερόμαστε σε ένα συγκεκριμένο συνδυασμό ηλεκτρονικών εξαρτημάτων που σε συνεργασία με ένα ή περισσότερα λογισμικά, επιτρέπουν τη λειτουργία ενός βιντεοπαιχνιδιού.

Οι πιο δημοφιλείς πλατφόρμες στις μέρες μας είναι οι παιχνιδοκονσόλες (game consoles), που αποτελούν ειδικά σχεδιασμένες συσκευές για αναπαραγωγή βιντεοπαιχνιδιών και κατανάλωση ψηφιακών μέσων με τη χρήση μιας τηλεόρασης (π.χ. PlayStation, Xbox, Nintendo Wii) ή σε φορητές οθόνες (π.χ. Nintendo DS, PlayStation Portable, κινητά τηλέφωνα) και προσωπικούς υπολογιστές. Όταν τα βιντεοπαιχνίδια άνθισαν στις αρχές της δεκαετίας του ‘70, η κυριότερη πλατφόρμα αναπαραγωγής τους ήταν οι μηχανές arcade (arcade machines) και ήταν ειδικά σχεδιασμένες για να μπορούν να μπορούν να αναπαράγουν αποκλειστικά ένα βιντεοπαιχνίδι. [GK15] Με τη πάροδο του χρόνου όμως, αυτή η πλατφόρμα άρχισε να ελαχιστοποιείται λόγω του υπέρογκου μεγέθους της και λόγω της ανικανότητας αναπαραγωγής πολλαπλών βιντεοπαιχνιδιών, ειδικά μετά την εισαγωγή των οικιακών πλατφορμών που έγιναν ιδιαίτερα δημοφιλείς τη δεκαετία του ‘80.

Σε συνδυασμό με την οπτική απεικόνιση, τα βιντεοπαιχνίδια προσφέρουν περαιτέρω μέσα παροχής αλληλεπίδρασης και πληροφορίας στους παίκτες, με το κυριότερο από αυτά να είναι ο ήχος, που αναπαράγεται μέσω ακουστικών και ηχείων. Επιπλέον μέσα παρέχονται μέσω απτικών περιφερειακών όπως η δόνηση (vibration) και η αναγκαστική ανατροφοδότηση (force feedback). [GK15]



Εικόνα 2.1 Pong(1972) - Ένα από τα πιο δημοφιλή βιντεοπαιχνίδια της πρώιμης εποχής της βιομηχανίας

### 2.1.1 Βασικές Κατηγορίες Βιντεοπαιχνιδιών

Παρά το γεγονός ότι είμαστε συνηθισμένοι στο να κατηγοριοποιούμε τα βιντεοπαιχνίδια ως προς το κύριο ψυχαγωγικό περιεχόμενο που προσφέρουν (π.χ παιχνίδια δράσης, αθλητικά παιχνίδια, παιχνίδια τρόμου), στη πραγματικότητα ο πιο βασικός τρόπος διαχωρισμού τους βρίσκεται ιεραρχικά ένα βήμα προς τα πάνω και αφορά το βασικό προσανατολισμό τους. Συνεπώς, το σύνολο των βιντεοπαιχνιδιών, χωρίζονται στα παιχνίδια ψυχαγωγίας (entertainment games) και στα λεγόμενα σοβαρά παιχνίδια (serious games).

Τα παιχνίδια ψυχαγωγίας, είναι σχεδιασμένα για να παρέχουν ευχαρίστηση στον παίκτη, καθώς ο ίδιος είναι συνεχώς απασχολημένος με δραστηριότητες εντός αυτών. [AK09]. Αποτελεί κατά συντριπτικό ποσοστό τη πιο δημοφιλή κατηγορία εκ των δύο, καθώς έχει ταυτιστεί στη συνείδηση των παικτών με το πως ορίζεται ένα βιντεοπαιχνίδι, δηλαδή ένα ψηφιακό μέσο που δίνει προτεραιότητα στην αναψυχή του παίκτη μέσω ενδιαφέρουσων ιστοριών ή μέσω δημιουργικών τρόπων παιχνιδιού (gameplay).

### 2.1.2 Σοβαρά Βιντεοπαιχνίδια

Στην ερώτηση “Τι ορίζουμε ως σοβαρό βιντεοπαιχνίδι;” έχουν δοθεί διαφορετικές ερμηνείες με τη πάροδο του χρόνου. Η πρώτη εμφάνιση του όρου serious game έγινε το 1970 και αναφέρεται σε προσομοιώσεις και παιχνίδια ως μέσα βελτίωσης της εκπαίδευσης, εντός και εκτός του σχολικού περιγύρου. Τα παραδείγματα που παρουσιάστηκαν αφορούσαν είτε μεγάλα συστήματα υπολογιστή, είτε παιχνίδια βασισμένα σε χαρτί και μολύβι, αφού ακόμα η βιομηχανία βιντεοπαιχνιδιών δεν είχε ακόμα εδραιωθεί. [A70] [DAJ11] Παρά τις προσπάθειες που έχουν γίνει ως προς την εύρεση ενός καθολικά αποδεκτού ορισμού, έχει εκφραστεί σκεπτικισμός ως προς το αν αυτό είναι δυνατόν. Αυτό γιατί η βιομηχανία που περικλείει τα σοβαρά βιντεοπαιχνίδια, απαρτίζεται από μια ευρύ φάσμα επαγγελματικών πεδίων, όπως η εκπαίδευση, η εθνική άμυνα, η πολιτική κτλ., που δεν συμφωνούν πάντα με το τι είναι και δεν είναι μέρος ενός σοβαρού βιντεοπαιχνιδιού [C07]. Παρά αυτές τις διαφορές, είναι γενικά αποδεκτό από όλα τα πεδία πως ένα σοβαρό βιντεοπαιχνίδι οφείλει να εκμεταλλευτεί το ενδιαφέρον των ανθρώπων στα βιντεοπαιχνίδια για σκοπούς που ξεφεύγουν από το πεδίο της απλής διασκέδασης. [DAJ11]

Με βάση τα παραπάνω λοιπόν, ένας απλός τρόπος για να προσδιορίσουμε το τι πρεσβεύουν τα σοβαρά βιντεοπαιχνίδια είναι να τα ορίσουμε με βάση τον τελικό τους σκοπό και όχι το τρόπο λειτουργίας/δημιουργίας τους. Αυτό σημαίνει ότι ένα σοβαρό βιντεοπαιχνίδι έχει ως τελικό σκοπό την ενημέρωση ή εκπαίδευση του παίκτη για ένα στόχο που έχει εφαρμογή στη πραγματική ζωή και όχι την ψυχαγωγή ή ικανοποίησή του μέσω μια σειρά αποστολών όπως συμβαίνει στα βιντεοπαιχνίδια ψυχαγωγίας. Ένα εύκολα κατανοητό παράδειγμα ενός σοβαρού βιντεοπαιχνιδιού είναι η προσομοίωση εξέτασης των σημάτων οδικής κυκλοφορίας, καθώς απαρτίζεται από στοιχεία που βρίσκουμε σε ένα παραδοσιακό βιντεοπαιχνίδι (αλληλεπίδραση χρήστη με διεπαφή χρήστη με τη μορφή κουίζ, χρήση ήχου και μια δομημένη σειρά “αποστολών”) αλλά ο τελικός του σκοπός είναι η προετοιμασία του



παίκτη για την τελική εξέταση των σημάτων ώστε να μπορέσει σε μελλοντικό χρόνο να αποκτήσει το δίπλωμα οδήγησής του.



Εικόνα 2.2 *Pulse!!*(2007) - Ένα παράδειγμα σοβαρού βιντεοπαιχνιδιού με σκοπό την εκπαίδευση στο χώρο της υγείας

### 2.1.3 Κατηγορίες Σοβαρών Βιντεοπαιχνιδιών

Για να μπορέσουμε να προσδιορίσουμε καλύτερα τη χρησιμότητα και την εφαρμογή των σοβαρών βιντεοπαιχνιδιών στη σημερινή κοινωνία και τη προσωπική μας ζωή, είναι αναγκαίο να χωριστούν σε ειδικές κατηγορίες. Ο τρόπος κατηγοριοποίησης τους λειτουργεί με γνώμονα το τελικό σκοπό τον οποίο επιτελούν, αλλά και ποιον τομέα της ζωής του ανθρώπου επηρεάζουν. Έχοντας αυτό σαν δεδομένο, τα σοβαρά βιντεοπαιχνίδια χωρίζονται ως εξής:

- **Παιχνίδια Εκπαίδευσης:** Είναι γεγονός πως τα βιντεοπαιχνίδια έχουν συνεισφέρει σημαντικά στον εμπλουτισμό της εκπαίδευσης με την πάροδο του χρόνου. Αυτό γιατί τα βιντεοπαιχνίδια από την φύση τους, προσθέτουν ένα στοιχείο ψυχαγωγίας που βοηθάει στη διατήρηση προσοχής του παίκτη και τον παροτρύνει να ασχοληθεί περισσότερο με τη πληροφορία που του παρουσιάζεται. Σε αυτή τη κατηγορία εντάσσονται οι προσομοιώσεις και η λεγόμενη παιχνιδοποίηση (gamification). [DD18] Τελικός σκοπός αυτής της κατηγορίας βιντεοπαιχνιδιών είναι να προσφέρουν στον παίκτη απαραίτητες γνώσεις θα πρέπει να εφαρμοστούν στον επαγγελματικό ή ακαδημαϊκό του περίγυρο.
- **Παιχνίδια Ενημέρωσης & Πειθούς:** Αυτή η κατηγορία βιντεοπαιχνιδιών έχει ως στόχο την καθοδήγηση του παίκτη σε ένα συμπέρασμα, χωρίς όμως να επηρεάζει την ελευθερία του ως προς το αν εκείνος συμφωνεί ή διαφωνεί με αυτό. [DD18] Θεωρείται γενικά μια ευαίσθητη κατηγορία, καθώς δεν είναι εύκολο να δημιουργηθεί και να δομηθεί σωστά ένα τέτοιο βιντεοπαιχνίδι. Χρησιμοποιούνται κυρίως για να

φέρουν το παίκτη στη θέση άλλων ευάλωτων κοινωνικών ομάδων ή για αλλαγή αρνητικών συμπεριφορών/συνηθειών.

#### 2.1.4 Δομή Σοβαρών Βιντεοπαιχνιδιών

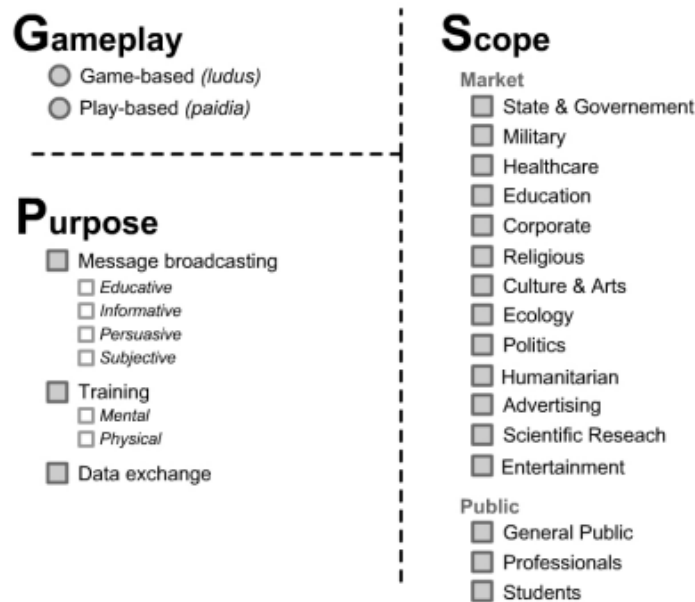
Άλλο ένα πολύ σημαντικό κριτήριο κατηγοριοποίησης αυτής του είδους βιντεοπαιχνιδιών, αφορά τη δομή τους. Η δομή αυτή ποικίλει, ανάλογα με τη γραμμικότητα του παιχνιδιού (linearity) και με το αν υπάρχει κάποιος τελικός στόχος για την ολοκλήρωσή του. Όταν αναφερόμαστε στη γραμμικότητα ενός βιντεοπαιχνιδιού, αναφερόμαστε στο βαθμό ελευθερίας που έχει ο παίκτης από ένα μέρος Α σε ένα μέρος Β. Αν ένα παιχνίδι χαρακτηρίζεται από γραμμικότητα (linear), τότε ο παίκτης γενικά καθοδηγείται από το ίδιο το παιχνίδι στο τελικό στόχο, χωρίς να έχει τη δυνατότητα να παρεκκλίνει από το στόχο του, ενώ τα γεγονότα του παιχνιδιού συμβαίνουν διαδοχικά. Αντιθέτως, αν ένα παιχνίδι δεν χαρακτηρίζεται από γραμμικότητα (non-linear), τότε ο παίκτης έχει τη δυνατότητα να παρεκκλίνει από τη πορεία που θα τον οδηγήσει στο τελικό στόχο, είτε για λόγους εξερεύνησης, είτε για ικανοποίηση άλλων στόχων του παιχνιδιού που δεν συνδέονται καθόλου με τον τελικό στόχο. Η παραπάνω εξήγηση, θα μας βοηθήσει να καταλάβουμε καλύτερα τις κατηγορίες δομής που εξηγούνται παρακάτω:

- **Παιχνίδια προσανατολισμένα στη διαδικασία (Process-oriented Games):** Σε αυτή τη κατηγορία υπάγονται τα παιχνίδια που δεν έχουν κάποιο συγκεκριμένο τελικό σκοπό, αλλά το περιεχόμενό τους απορροφάται από τον παίκτη μέσω συνεχών και επαναλαμβανόμενων δραστηριοτήτων. Τέτοια παιχνίδια είναι αυτά που επικεντρώνονται στην εξερεύνηση, στη προσομοίωση και στη λήψη αποφάσεων. [DD18] Για αυτό το λόγο είναι συνήθως μη γραμμικά και μπορούν να χρησιμοποιηθούν για μεγάλο χρονικό διάστημα, καθώς ο παίκτης έχει τη δυνατότητα να εκπαιδευτεί ή να βελτιωθεί περαιτέρω με κάθε αλληλεπίδρασή του με αυτά.
- **Παιχνίδια προσανατολισμένα στο αποτέλεσμα (Outcome-oriented Games):** Σε αυτή τη κατηγορία ανήκουν τα παιχνίδια όπου ο τελικός στόχος είναι η κύρια εστίασή τους. Περιέχουν δραστηριότητες και στόχους που εκπαιδεύουν το παίκτη με σκοπό να αποκτήσουν μια ή περισσότερες δεξιότητες στο τέλος αυτών, είτε επαγγελματικές, είτε πνευματικές. [DD18] Για το λόγο αυτό θεωρούνται γραμμικά παιχνίδια και χρησιμοποιούνται συνήθως μια φορά. Παραδείγματα εφαρμογής τους είναι η εκπαίδευση ενός νέου εργαζόμενου και η ενημέρωση μαθητών σε μια ειδική ημερίδα.

#### 2.1.5 Το Μοντέλο G/P/S (Gameplay/Purpose/Scope Model)

Επιστρέφοντας πίσω στην αναφορά που έγινε για το πως είναι σχετικά δύσκολο να δοθεί μια ερμηνεία για το τι απαρτίζει ένα σοβαρό βιντεοπαιχνίδι και πως αυτό μπορεί να ενταχθεί με σιγουριά σε μια κατηγορία, δημιουργήθηκε ένα μοντέλο που προσπαθεί να διευκολύνει τη διαδικασία αυτή. Ένα σοβαρό βιντεοπαιχνίδι, αποτελείται από δύο διαστάσεις, αυτή της “σοβαρότητας” που περιέχει τους σκοπούς και της εφαρμογής του

παιχνιδιού στη κανονική ζωή και αυτή του “gameplay” που περιέχει όλα τα στοιχεία που χτίζουν το παιχνίδι μέσα στο οποίο χτίζεται η γνώση. Για να συνδυαστούν αρμονικά αυτές οι δύο διαστάσεις, δημιουργήθηκε το μοντέλο G/P/S, το οποίο δανείζεται το παράδειγμα σκοπού και αγοράς (purpose & market) και προσθέτει το τρόπο παιχνιδιού ως κριτήριο. Συγκεκριμένα, το μοντέλο G/P/S βασίζεται στις παρακάτω 3 όψεις:



Εικόνα 2.3 Σχεδιάγραμμα Ταξινόμησης που αναπαριστά το μοντέλο G/P/S

- **Τρόπος Παιχνιδιού (Gameplay):** Αναφέρεται όπως φανερώνει και το όνομα στο τρόπο παιχνιδιού που εφαρμόζεται. Αυτή η όψη παρέχει πληροφορίες για τη δομή του παιχνιδιού, δηλαδή απαντά στην ερώτηση “Πως παίζεται;”.
- **Σκοπός (Purpose):** Αναφέρεται στο τελικό σκοπό για τον οποίο δημιουργήθηκε το παιχνίδι. Αυτή η όψη συνυπολογίζει ταυτόχρονα όλους τους σκοπούς εκτός από αυτόν της διασκέδασης που ήταν εξ αρχής στο πλάνο του σχεδιαστή και απαντά στην ερώτηση “Ποιος είναι ο σκοπός του;”
- **Πεδίο Εφαρμογής (Scope):** Αναφέρεται στις στοχευμένες εφαρμογές του παιχνιδιού. Αυτή η όψη υποδεικνύει τις προτεινόμενες χρήσεις του παιχνιδιού, το κοινό στο οποίο απευθύνεται και απαντά στην ερώτηση “Για ποιον προορίζεται;”

Λαμβάνοντας αυτές τις συνιστώσες, το μοντέλο G/P/S σκοπεύει να παρέχει ένα έδαφος συνεννόησης για όλα τα επαγγελματικά πεδία τα οποία περικλείουν τα σοβαρά βιντεοπαιχνίδια και να βοηθήσει τη κατηγοριοποίησή τους ανάλογα με το τι μπορεί να προσφέρει κάθε παιχνίδι και σε ποια ομάδα ανθρώπων. Συνεπώς, για τη διευκόλυνση εύρεσης του παιχνιδιού που ικανοποιεί όλα τα κριτήρια που αναζητά ένα άτομο, δημιουργήθηκε ένα σχεδιάγραμμα ταξινόμησης (Εικόνα 2.1). Με τη χρήση αυτού, κάθε παιχνίδι ταξινομείται εύκολα, ανάλογα τα κριτήρια που ικανοποιεί, κάτι που αποτρέπει επικείμενες αστοχίες ως προς την επιλογή ενός παιχνιδιού. [DAJ11]

## 2.2 Κυκλοφοριακή Αγωγή

Ο θεσμός της κυκλοφοριακής αγωγής (traffic education) αναφέρεται σε κάθε είδους επίσημης ή ανεπίσημης ενημέρωσης που στοχεύει στην εκμάθηση και βελτίωση των γνώσεων, της διορατικότητας, των ικανοτήτων και της συμπεριφοράς που οφείλουμε να έχουμε για να μπορούμε να συμμετέχουμε στο κυκλοφοριακό δίκτυο με ασφάλεια. Η παροχή της κυκλοφοριακής αγωγής γίνεται κυρίως από τα σχολεία, αλλά και από οργανισμούς οδικής ασφάλειας και την οικογένεια. Η κυκλοφοριακή αγωγή είναι απαραίτητη για όλα τα μέλη του οδικού δικτύου σε όλους τους ρόλους (οδηγός αυτοκινήτου, πεζός, ποδηλάτης κτλ.) και συνεπώς δεν προορίζεται αποκλειστικά για παιδιά. Η συνεχής παροχή κυκλοφοριακής αγωγής στοχεύει στη σταθερή και συνεχή ενημέρωση των ανθρώπων σε όλα τα στάδια της ζωής τους και σε όλους τους ρόλους που πρόκειται να υιοθετήσουν.

Παρόλο που ακόμη δεν έχουμε ακριβή στατιστικά ως προς το πως η κυκλοφοριακή αγωγή έχει επηρεάσει τις πιθανότητες δημιουργίας ατυχήματος, κάποιες εκτιμήσεις δείχνουν ότι μπορεί να επηρεάσει τη συμπεριφορά και τη στάση των ανθρώπων, πράγμα που οδηγεί σε μεγαλύτερη εμπιστοσύνη ανάμεσα στα μέλη του οδικού δικτύου ανά πάσα στιγμή. Για να είναι άκρως αποτελεσματικό ένα πρόγραμμα κυκλοφοριακής αγωγής θα πρέπει να ικανοποιεί κάποιες σημαντικές απαιτήσεις που είναι οι εξής:

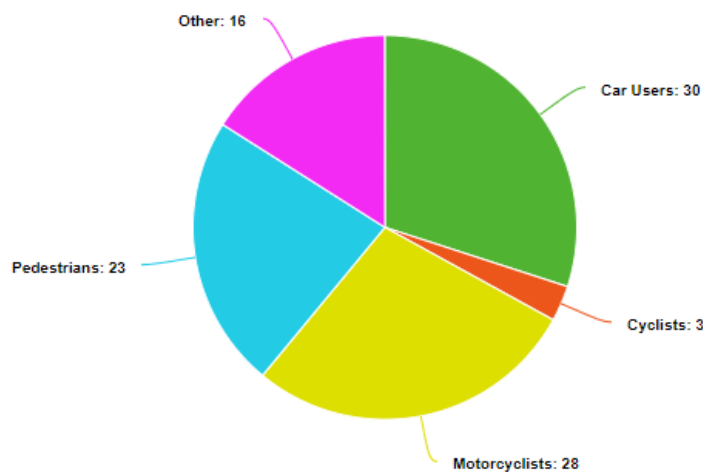
- Θα πρέπει να επικεντρώνεται στις συμπεριφορές που είναι αποδεδειγμένο ότι έχουν ξεκάθαρη σχέση με την οδική ασφάλεια. (κατανάλωση αλκοόλ, υπερβολική ταχύτητα, μη τήρηση του κώδικα οδικής κυκλοφορίας)
- Θα πρέπει να επικεντρώνεται στις ηλικιακές ομάδες που παρουσιάζουν τα περισσότερα συμπεριφορικά προβλήματα, καθώς αυτές είναι οι ομάδες που είναι περισσότερο ικανές για βελτίωση της συμπεριφοράς τους. (ανήλικοι, νέοι οδηγοί που ανήκουν στην ηλικιακή ομάδα 18-24 ετών)
- Θα πρέπει να λαμβάνει υπόψη γιατί αυτές οι ομάδες έχουν αυτή τη συμπεριφορά. (απειρία, έλλειψη ωριμότητας)
- Θα πρέπει να προσφέρει το ενδεχόμενο εκμάθησης μέσω των εμπειριών που ήδη έχουν κατά την επαφή τους με το οδικό δίκτυο.

Πέρα όμως από τα προγράμματα κυκλοφοριακής αγωγής, όπως αναφέραμε, εξίσου σημαντικός είναι και ο ρόλος της οικογένειας ως προς τα παιδιά από πολύ μικρή ηλικία. Αυτό γίνεται εφικτό μέσω των γονέων, οι οποίοι με ανεπίσημο και φιλικό τρόπο, καθοδηγούν τα παιδιά τους στη καθημερινή τους ζωή. Για παράδειγμα, μπορούν να υποδείξουν πιθανούς κινδύνους ενώ περπατάνε προς το σχολείο ή το σπίτι ή να εξηγήσουν συγκεκριμένες καταστάσεις ή συμπεριφορές ενώ βρίσκονται σε ένα μέσο μεταφοράς. Αυτός είναι ένας αποδοτικός πρώτος τρόπος ενημέρωσης των παιδιών που λόγω του νεαρού της ηλικίας τους δεν μπορούν να μετακινηθούν ακόμα ανεξάρτητα. Αυτό όμως δεν σημαίνει πως οι γονείς δεν μπορούν να έχουν θετική επίδραση στα παιδιά τους, ακόμα και όταν ανεξαρτητοποιηθούν, καθώς όπως αναφέρθηκε παραπάνω, είναι θεμιτό η κυκλοφοριακή αγωγή να είναι κομμάτι όλων, καθ' όλη τη διάρκεια της ζωής τους. [\[SWOV17\]](#)

## 2.2.1 Στατιστικά Στοιχεία

Για να καταλάβουμε καλύτερα και με μια πιο ώριμη ματιά την αναγκαιότητα της κυκλοφοριακής αγωγής στη σημερινή κοινωνία, αρκεί να κοιτάξουμε τα αρνητικά στατιστικά που συνδέονται με τη θνησιμότητα λόγω οδικών ατυχημάτων. Ο Παγκόσμιος Οργανισμός Υγείας (WHO), έπειτα από έρευνα που πραγματοποιήθηκε το 2018, ανακοίνωσε μόνο εκείνη τη χρονιά πάνω από 1.3 εκατομμύρια άνθρωποι έχασαν τη ζωή τους στην άσφαλο, πράγμα που αντιστοιχεί σε έναν θάνατο κάθε 23 δευτερόλεπτα. Αναλυτικότερα το 2018 έχασαν τη ζωή τους:

- Πάνω από 390.000 οδηγοί επιβατικών αυτοκινήτων (ένας κάθε 80 δευτερόλεπτα)
- Πάνω από 40.000 ποδηλάτες (ένας κάθε 13 λεπτά)
- Πάνω από 370.000 μοτοσυκλετιστές (ένας κάθε 83 δευτερόλεπτα)
- Πάνω από 310.000 πεζοί (ένας κάθε 101 δευτερόλεπτα) [[WHO18](#)]



Εικόνα 2.4 Στατιστικά στοιχεία θανάτων στους δρόμους από τον WHO (2018)

## 2.2.2 Η Κυκλοφοριακή Αγωγή Στην Εκπαίδευση

Είναι φανερό λοιπόν πως υπάρχει ένα μακρύς δρόμος για την ελαχιστοποίηση αυτού του φαινομένου και γι' αυτό η κυκλοφοριακή αγωγή θεμελιώνεται όλο και περισσότερο στη καθημερινή μας ζωή από όλο και μικρότερη ηλικία μέσα από το θεσμό της εκπαίδευσης. Ήδη από το νηπιαγωγείο, όλα τα παιδιά έχουν μια πρώτη επαφή με τη κυκλοφοριακή αγωγή μέσω των δασκάλων τους, οι οποίοι μέσω πολύ απλών παραδειγμάτων, τους μεταδίδουν τους βασικούς κανόνες ασφαλείας (υποχρεωτική χρήση ζώνης ασφαλείας, πέρασμα του δρόμου μέσω των διαβάσεων πεζών και με ποια ένδειξη του φαναριού κτλ.)

Η εκπαίδευση αυτή μέχρι πρότινος αναβαθμιζόταν σημαντικά στο δημοτικό σχολείο όπου μέσω μαθημάτων όπως η Ευέλικτη Ζώνη, οι μαθητές είχαν την ευκαιρία να εξερευνήσουν αναλυτικότερα τον Κώδικα Οδικής Κυκλοφορίας. Συγκεκριμένα, γινόταν αναφορά στο πως οι μαθητές μπορούσαν να προστατέψουν περισσότερο τον εαυτό τους, αλλά και σε σημαντικούς κανόνες που οφείλουν να ακολουθήσουν ως πεζοί και ως επιβάτες. Το μάθημα της Ευέλικτης Ζώνης δυστυχώς έχει πλέον αντικατασταθεί, κυρίως επειδή



απέτυχε να κερδίσει τη προσοχή των μαθητών, αλλά και γιατί οι εκπαιδευτικοί σε μεγάλο ποσοστό, δεν το εκμεταλλεύτηκαν σωστά. Ιδιαίτερα, το κεφάλαιο της κυκλοφοριακής αγωγής ήταν το τελευταίο κεφάλαιο στα βιβλία, που σημαίνει ότι δεν διδασκόταν κατά τη διάρκεια της σχολικής χρονιάς λόγω έλλειψης χρόνου (εκτός κι αν ένας εκπαιδευτικός επισκεπτόταν και δίδασκε το κεφάλαιο αυτό κατ'εξαίρεση). Για το λόγο αυτό, αρκετά σχολεία σε συνδυασμό με δημόσιους φορείς όπως η Ελληνική Αστυνομία, πραγματοποιούν ειδικές συναντήσεις με τους μαθητές με σκοπό την ενημέρωσή τους για τα ίδια θέματα, αλλά εντός ενός περιβάλλοντος που είναι σίγουρο ότι θα απορροφήσουν τις παρεχόμενες πληροφορίες. Εξηγούν τους κανόνες συμπεριφοράς αναλυτικότερα, αναφέρονται στις σημαντικότερες πινακίδες σήμανσης που ενδέχεται να συναντήσουν οι μαθητές εκτός του σχολικού περιγύρου και απαντούν σε όλες τις ερωτήσεις που ενδέχεται να δημιουργηθούν μετά το πέρας της παρουσίασης.

Πέρα όμως από τους συμβατικούς τρόπους εκπαίδευσης, με τη πάροδο του χρόνου έχουν πραγματοποιηθεί συναντήσεις που ξεφεύγουν από τη παραδοσιακή ιδέα που θέλει τους μαθητές να δέχονται παθητικά γνώση μέσω του προφορικού λόγου και παρουσιάσεων σε μια αίθουσα. Το κυριότερο παράδειγμα αυτού ήταν μια πρωτότυπη ιδέα που είχε ο ιδιωτικός τηλεοπτικός σταθμός 0-6 TV, ο οποίος σε συνεργασία με την Ελληνική Αστυνομία, δημιούργησε το πρώτο φορητό πάρκο κυκλοφοριακής αγωγής στην Ελλάδα το 2007. Στόχος τους, ήταν να επισκεφτούν όλα τα σχολεία του Νομού Αττικής και να δώσουν την ευκαιρία στους μαθητές να μάθουν για τους κανόνες της κυκλοφοριακής αγωγής με έναν πρωτότυπο και διασκεδαστικό τρόπο. Συγκεκριμένα, αφού είχε προηγηθεί μια ενημέρωση των μαθητών, παρόμοια με ότι αναφέρθηκε στην προηγούμενη παράγραφο, οι μαθητές μπορούσαν μετά με τη παρουσία ενός φορητού κυκλοφοριακού πάρκου που κατασκευαζόταν στο προαύλιο κάθε σχολείου και τη χρήση ηλεκτρικών αυτοκινήτων να έχουν τη πρώτη τους επαφή ως οδηγοί σε ένα ασφαλές περιβάλλον. Το πάρκο, διέθετε διαβάσεις πεζών, πινακίδες σήμανσης και φανάρια στα οποία οι μαθητές έπρεπε να αντιδράσουν χρησιμοποιώντας τις γνώσεις που είχαν αντλήσει. Η προσπάθεια αυτή έληξε ένα χρόνο μετά, αφού είχε ολοκληρωθεί η επίσκεψη σε όλα τα σχολεία και άφησε άκρως θετικές εντυπώσεις σε μαθητές και εκπαιδευτικούς. Προσπάθειες σαν αυτή, δημιούργησαν τα θεμέλια για μια πιο διαδραστική εμπειρία σε ότι αφορά τη κυκλοφοριακή αγωγή, αφού ήταν φανερό πως οι μαθητές ήθελαν να αξιοποιήσουν το ενδιαφέρον και τη προσοχή τους με έναν πιο ενεργητικό τρόπο.

### **2.2.3 Κυκλοφοριακή Αγωγή και Βιντεοπαιχνίδια**

Η συσχέτιση της κυκλοφοριακής αγωγής με τα βιντεοπαιχνίδια, έχει αποκτήσει ιδιαίτερο ενδιαφέρον τις τελευταίες δύο δεκαετίες, έπειτα από μελέτες που έχουν πραγματοποιηθεί πάνω στις θετικές επιδράσεις που μπορεί να έχουν τα βιντεοπαιχνίδια σε ανθρώπινες λειτουργίες που συνδέονται με τη συμπεριφορά εντός ενός οδικού δικτύου. Συγκεκριμένα, μια έρευνα σε σχολές οδηγών στη Σουηδία το 2005 και το 2006 [BHM06], όταν ακόμα δηλαδή τα βιντεοπαιχνίδια δεν ήταν ακόμα ένα τόσο διαδεδομένο μέσο ψυχαγωγίας συγκριτικά με σήμερα και η σχέση τους με την εκπαίδευση ήταν ελάχιστη, έδειξαν ενδιαφέροντα αποτελέσματα μεταξύ των ανθρώπων που είχαν μια εμπειρία με αυτά

έναντι αυτών που δεν είχαν. Αφού οι συμμετέχοντες χωρίστηκαν σε δύο κατηγορίες έπειτα από ενδελεχή εξέταση (χρήστες και μη χρήστες βιντεοπαιχνιδιών), προέκυψαν τα εξής αποτελέσματα που εξήχθησαν στην αρχή και στο τέλος της εκπαίδευσής τους:

Ικανότητα	Μη Χρήστες Βιντεοπαιχνιδιών	Χρήστες Βιντεοπαιχνιδιών
Διανομή προσοχής	3.1	4.3
Χειρισμός καταστάσεων που απαιτούν γρήγορες αποφάσεις	3.0	4.8
Εκτίμηση κινδύνου σε κρίσιμες καταστάσεις	3.9	4.4
Γενική οδηγική ικανότητα	3.9	4.9
Στάση ως προς	Μη Χρήστες Βιντεοπαιχνιδιών	Χρήστες Βιντεοπαιχνιδιών
Οδήγηση διατηρώντας αποστάσεις ασφαλείας	3.7	4.5
Όρια ταχύτητας	4.9	4.9
Άλλους οδηγούς/πεζούς	4.9	4.8
Γενική στάση	5.4	5.3

Πίνακας 2.1 Σύγκριση χρηστών και μη χρηστών βιντεοπαιχνιδιών ως προς τη στάση και την ικανότητά τους στην αρχή της εκπαίδευσής τους. [\[BHM06\]](#)

Παρατηρούμε πως στην αρχή της εκπαίδευσής τους, οι χρήστες βιντεοπαιχνιδιών έχουν σαφές προβάδισμα και στις τέσσερις εξεταζόμενες κατηγορίες ικανότητας πράγμα που συσχετίζει, έστω και σε μικρό βαθμό τα βιντεοπαιχνίδια με γενική βελτίωση της κριτικής ικανότητας και των αντανακλαστικών, ενώ έκπληξη προκαλεί το γεγονός ότι από την αρχή της εκπαίδευσής τους, οι χρήστες βιντεοπαιχνιδιών είχαν μεγαλύτερη έφεση στην οδήγηση. Αντιθέτως, παρατηρούμε ότι τα αποτελέσματα στις τέσσερις εξεταζόμενες κατηγορίες στάσης είναι πιο μεικτά, με τους χρήστες βιντεοπαιχνιδιών να έχουν προβάδισμα στη διατήρηση αποστάσεων ασφαλείας κατά τη διάρκεια της οδήγησης αλλά να υστερούν ελαφρώς σε ότι αφορά τη στάση τους προς τα άλλα μέλη του οδικού δικτύου αλλά και στη γενική τους στάση. Σε ότι αφορά τα όρια ταχύτητας και οι δύο κατηγορίες παρουσιάζουν τα ίδια αποτελέσματα. Το προβάδισμα που έχουν οι χρήστες βιντεοπαιχνιδιών στη διατήρηση αποστάσεων ασφαλείας, μπορεί να εξηγηθεί λόγω της κριτικής ικανότητας που χρειάζεται για να ικανοποιηθεί αυτό το κριτήριο, πράγμα που δεν ισχύει στις υπόλοιπες τρεις κατηγορίες, όπου η εμπειρία ως οδηγός και ο χαρακτήρας του κάθε εξεταζόμενου παίζουν μεγαλύτερο ρόλο.

Ικανότητα	Μη Χρήστες Βιντεοπαιχνιδιών	Χρήστες Βιντεοπαιχνιδιών
Διανομή προσοχής	4.8(+1.7)	5.7(+1.4)
Χειρισμός καταστάσεων που απαιτούν γρήγορες αποφάσεις	4.6(+1.6)	5.9(+1.1)
Εκτίμηση κινδύνου σε κρίσιμες καταστάσεις	5.3(+1.4)	5.8(+1.4)
Γενική οδηγική ικανότητα	5.4(+1.5)	6.1(+1.2)
Στάση ως προς	Μη Χρήστες Βιντεοπαιχνιδιών	Χρήστες Βιντεοπαιχνιδιών
Οδήγηση διατηρώντας αποστάσεις ασφαλείας	5.3(+1.6)	6.0(+1.5)
Όρια ταχύτητας	5.9(+1.0)	5.9(+1.0)
Άλλους οδηγούς/πεζούς	5.7(+0.8)	5.8(+1.0)
Γενική στάση	5.9(+0.5)	6.2(+0.9)

Πίνακας 2.2 Σύγκριση χρηστών και μη χρηστών βιντεοπαιχνιδιών ως προς τη στάση και την ικανότητά τους στο τέλος της εκπαίδευσής τους. [\[BHM06\]](#)

Παρατηρούμε πως μετά το πέρας της εκπαίδευσής, υπάρχουν κάποιες σημαντικές διαφορές στα αποτελέσματα και τις συγκρίσεις των δύο κατηγοριών. Σε ότι αφορά τις τέσσερις εξεταζόμενες κατηγορίες ικανότητας, τα αποτελέσματα παραμένουν υπέρ των χρηστών βιντεοπαιχνιδιών με σημαντική διαφορά από τους μη χρήστες. Παρά τη διατήρηση των αποτελεσμάτων όμως, βλέπουμε πως οι μη χρήστες εμφάνισαν μεγαλύτερη βελτίωση στις βαθμολογίες τους. Αυτή η βελτίωση μπορεί να αποδοθεί στο γεγονός ότι οι μη χρήστες βιντεοπαιχνιδιών δεν είχαν εξ αρχής τη δυνατότητα να εξασκήσουν αυτές τις ικανότητες, οπότε και είχαν μεγαλύτερο περιθώριο βελτίωσης από τους χρήστες. Γενικότερα σε ότι αφορά τις φυσικές ικανότητες του ανθρώπου, αυτές αυξάνονται λογαριθμικά. Αυτό σημαίνει ότι ενώ στην αρχή υπάρχει δραματική βελτίωση, στη συνέχεια ο ρυθμός βελτίωσης ελαχιστοποιείται όλο και πιο γρήγορα μέχρι να μην υπάρχει εμφανής βελτίωση όσο φτάνουμε το φυσικό ταβάνι των δυνατοτήτων μας.

Προχωρώντας στις τέσσερις εξεταζόμενες κατηγορίες στάσης, τα αποτελέσματα πλέον είναι αρκετά διαφορετικά συγκριτικά με τη προηγούμενη σύγκριση. Ενώ στα όρια ταχύτητας βλέπουμε παρόμοια βελτίωση/συμμόρφωση, στις υπόλοιπες κατηγορίες οι χρήστες βιντεοπαιχνιδιών φαίνεται να έχουν παρουσιάσει τη μεγαλύτερη βελτίωση, ενώ η βαθμολογία τους πλέον υπερβαίνει αυτή των μη χρηστών, χωρίς όμως μεγάλη διαφορά στις τελευταίες δύο που υστερούσαν. Λόγω του μικρών διαφορών, δε μπορούμε να πούμε με ασφάλεια ότι η χρήση βιντεοπαιχνιδιών είχε μεγαλύτερο ρόλο από την αντίδραση που είχαν οι εξεταζόμενοι στην εκπαίδευση σε συνδυασμό με το προσωπικό τους χαρακτήρα, αλλά αν



κοιτάζουμε τα αποτελέσματα συγκεντρωτικά, μπορούμε να συμπεράνουμε ότι είναι εφικτή μια ενδεχόμενη εφαρμογή της κυκλοφοριακής αγωγής σε ένα βιντεοπαιχνίδι, πόσο μάλλον στη κοινωνία του σήμερα που έχει ταυτιστεί με τη τεχνολογία.

## 2.3 Τεχνολογίες Ανάπτυξης Βιντεοπαιχνιδιών

Για να δημιουργηθεί ένα βιντεοπαιχνίδι, απαιτείται ένας συνδυασμός τεχνολογιών, όπου κάθε μία έχει πρωταρχικό ρόλο σε τουλάχιστον ένα κομμάτι του. Από την εικόνα, τον ήχο, τον σχεδιασμό περιβάλλοντος μέχρι τους στόχους και το τρόπο παιχνιδιού, ένα βιντεοπαιχνίδι απαιτεί άριστη εκτέλεση σε όλους αυτούς τους τομείς ώστε να μπορεί να γίνει αρεστό στο κοινό το οποίο απευθύνεται. Για το λόγο αυτό εδώ και αρκετά χρόνια, η ανάπτυξη ενός βιντεοπαιχνιδιού φαινόταν ως ένα δύσκολο έως και ακατόρθωτο έργο για τους περισσότερους ανθρώπους οι οποίοι ευελπιστούσαν να ασχοληθούν με τη βιομηχανία. Αυτό το χάσμα πλέον έχει γεφυρωθεί, πράγμα που αντανακλάται στη ραγδαία αύξηση των ανεξάρτητων βιντεοπαιχνιδιών (indie games) τα τελευταία χρόνια. Με τη χρήση μηχανών παιχνιδιών (game engines) όπως η Unity και η Unreal Engine, χιλιάδες επιτυχημένα βιντεοπαιχνίδια έχουν δημιουργηθεί από μικρές ομάδες ανθρώπων ή ακόμα και ένα άτομο, αφού προσφέρουν μια έτοιμη πλατφόρμα για την ανάπτυξη βιντεοπαιχνιδιών όλων των ειδών. Έτσι, σε συνδυασμό με τις γνώσεις του χρήστη και την ενσωμάτωση ποικίλων τεχνολογιών και επιστημονικών κλάδων όπως τα γραφικά υπολογιστών και η τεχνητή νοημοσύνη, είναι δυνατόν να αναπτυχθεί ένα βιντεοπαιχνίδι πιο εύκολα και αποτελεσματικά από ποτέ.

### 2.3.1 Μηχανές Παιχνιδιών (Game Engines)

Ως μηχανή παιχνιδιού, ορίζουμε μια συλλογή εργαλείων λογισμικού ή διεπαφών προγραμματισμού εφαρμογών (APIs), κατασκευασμένη για την βελτιστοποίηση ενός βιντεοπαιχνιδιού και τυπικά περιέχουν τουλάχιστον μια μηχανή σχεδιασμού δύο ή τριών διαστάσεων (2D/3D Rendering Engines). Επιπροσθέτως, κάποιες από τις πιο εύρωστες μηχανές όπως η Unity, μπορούν να περιέχουν πολλά και διαφορετικά εξαρτήματα όπως μηχανές αναπαράστασης συγκρούσεων, φυσικής κίνησης και ήχου, εργαλεία δικτύωσης, multiplayer και εικονικής πραγματικότητας, καθώς και εξαρτήματα φωτισμού, σκίασης, υλικού αντικειμένων κτλ. Τέλος, τέτοιες μηχανές μπορούν να προσφέρουν εξαρτήματα τεχνητής νοημοσύνης, επεξεργασίας περιβάλλοντος και ένα ενσωματωμένο περιβάλλον ανάπτυξης κώδικα με τη χρήση γλωσσών προγραμματισμού όπως η C# ή η JavaScript.

Κατά το ξεκίνημα της βιομηχανίας βιντεοπαιχνιδιών, όλα τα παιχνίδια αναπτύσσονταν με τη χρήση προσαρμοσμένων μηχανών σχεδιασμού που ήταν σχεδιασμένες ειδικά για κάθε υπό ανάπτυξη παιχνίδι. Όσο η βιομηχανία εξελισσόταν, σιγά σιγά οι κορυφαίες εταιρείες άρχισαν να χρησιμοποιούν ιδιόκτητες μηχανές που χρησιμοποιούνταν αποκλειστικά για την ανάπτυξη των δικών τους βιντεοπαιχνιδιών. Χρειάστηκαν αρκετά χρόνια ώστε να κυκλοφορήσει κάτι που να μοιάζει έστω και λίγο με τις μηχανές του σήμερα,

με ένα τέτοιο παράδειγμα να είναι το Adventure Construction Kit που κυκλοφόρησε το 1985 από την Electronic Arts για κονσόλες της εποχής. [\[G16\]](#)



*Εικόνα 2.4 Ένα δωμάτιο κατασκευασμένο με τη χρήση προσαρμοσμένων πλακιδίων μέσω του Adventure Construction Kit*

Τα τελευταία χρόνια, ο αριθμός των διαθέσιμων μηχανών στην αγορά έχει αυξηθεί δραματικά. Η Unreal Engine που κυκλοφόρησε το 1998, ίσως είναι η κορυφαία μηχανή ανάπτυξης βιντεοπαιχνιδιών αυτή τη στιγμή. Από τη κυκλοφορία της, ήταν εξ αρχής διαθέσιμη με άδεια σε όλα τα ενδιαφερόμενα στούντιο. Διαθέσιμη μέχρι και σήμερα, η Unreal Engine έχει υποστεί σημαντικές αλλαγές για να μπορεί να ανταποκριθεί στις σύγχρονες ανάγκες της βιομηχανίας και γι' αυτό το λόγο κατέχει μεγάλο μερίδιο εντός αυτής. [\[G16\]](#)

### **2.3.1.1 Μηχανές Παιχνιδιών και Ανεξάρτητη Ανάπτυξη Βιντεοπαιχνιδιών**

Ανεξάρτητη ανάπτυξη βιντεοπαιχνιδιών (indie game development) ονομάζουμε τη διαδικασία δημιουργίας ενός ή περισσότερων βιντεοπαιχνιδιών από μια μικρή ομάδα ανθρώπων ή ακόμα και ένα άτομο, χωρίς κάποια ιδιαίτερη οικονομική ή τεχνική στήριξη, πράγματα που είναι παρόντα σε όλους τους τίτλους βιντεοπαιχνιδιών που έχουν αναπτυχθεί από οικονομικά και πληθυσμιακά ισχυρότερες εταιρείες. Παρόλο που δεν είναι τίποτα καινούργιο στη βιομηχανία, η αγορά ανεξάρτητων βιντεοπαιχνιδιών αυξήθηκε δραματικά από το 2008 και μετά λόγω δύο σημαντικών παραγόντων.

Πρώτον, οι μεγαλύτερες υπηρεσίες διάθεσης βιντεοπαιχνιδιών σε ψηφιακή μορφή όπως το XBOX Live, το PlayStation Network, το Steam κτλ, έχουν δώσει την ευκαιρία σε πολλούς ανεξάρτητους προγραμματιστές να παρουσιάσουν, να προωθήσουν και να κυκλοφορήσουν τις δημιουργίες τους σε ένα τεράστιο κοινό χρηστών, χωρίς να χρειάζεται να συνεργαστούν με κάποια μεγάλη εκδοτική εταιρεία.

Δεύτερον, κατά τη διάρκεια της τελευταίας δεκαετίας, η αγορά έχει δει πολλαπλές μηχανές παιχνιδιών να κυκλοφορούν. Πλέον υπάρχουν μηχανές που μπορούν να

υποστηρίζουν και να κυκλοφορήσουν παιχνίδια για κάθε πλατφόρμα που υπάρχει αυτή τη στιγμή, από κονσόλες μέχρι υπολογιστές και κινητά. Εκτός αυτού, με ακόμα και τις πιο εύρωστες μηχανές να είναι διαθέσιμες σε λογικές τιμές, αυτό έχει οδηγήσει σε μια εξισορρόπηση της αγοράς σε σημείο που πλέον ανεξάρτητα μικρά στούντιο μπορούν να ανταγωνιστούν ή ακόμα και να ξεπεράσουν τις μεγαλύτερες εταιρείες της βιομηχανίας. [\[G16\]](#)

### 2.3.2 Γραφικά Υπολογιστών (Computer Graphics)

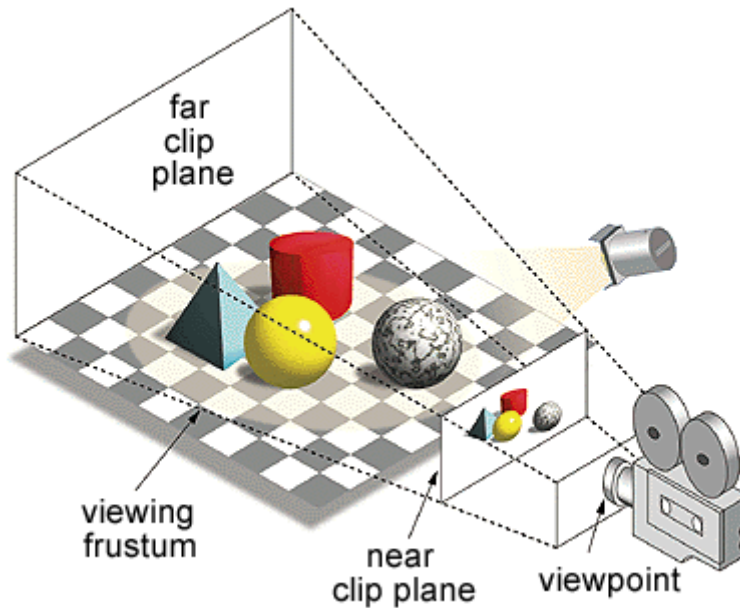
Με τον όρο “Γραφικά Υπολογιστών”, αναφερόμαστε σε οτιδήποτε έχει να κάνει με τη δημιουργία ή το χειρισμό εικόνων σε έναν υπολογιστή συμπεριλαμβανομένων των κινούμενων εικόνων. Είναι ένα ευρύ πεδίο που αλλάζει και εξελίσσεται συνεχώς. Στους πρώτους επιτραπέζιους υπολογιστές, τα περιεχόμενα της οθόνης διαχειρίζονταν απευθείας από τη κεντρική μονάδα επεξεργασίας (CPU). Για παράδειγμα, για να σχηματιστεί μια γραμμή στην οθόνη, η CPU έτρεχε μια επαναλαμβανόμενη διαδικασία κατά την οποία τοποθετούμε το κατάλληλο χρώμα για κάθε εικονοστοιχείο (pixel) που υπήρχε στη σχηματιζόμενη γραμμή, δαπανώντας έτσι αρκετό χρόνο και χωρητικότητα της CPU. Επιπλέον, η επίδοση των γραφικών ήταν πολύ αργή συγκριτικά με το τι ισχύει σήμερα. Στις μέρες μας, οι υπολογιστές είναι σαφώς ταχύτεροι αλλά η μεγαλύτερη αλλαγή κρύβεται στο γεγονός ότι η επεξεργασία των γραφικών πλέον γίνεται από ένα ειδικό εξάρτημα που ονομάζεται μονάδα επεξεργασίας γραφικών (GPU). Μια GPU, περιέχει επεξεργαστές που πραγματοποιούν όλους τους υπολογισμούς γραφικών και συγκεκριμένα, περιέχει έναν μεγάλο αριθμό επεξεργαστών που λειτουργούν παράλληλα για να επιταχύνουν σημαντικά όλες τις γραφικές λειτουργίες. Επιπλέον, περιέχει τη δική του αποκλειστική μνήμη για την αποθήκευση εικόνων και λιστών με συντεταγμένες. Οι GPU έχουν ταχύτατη πρόσβαση σε δεδομένα που είναι αποθηκευμένα στη μνήμη τους, πολύ ταχύτερη από τη ταχύτητα πρόσβασης που έχουν στη κύρια μνήμη του υπολογιστή.

Για να σχεδιάσουμε μια γραμμή ή για να εκτελέσουμε κάποια άλλη γραφική λειτουργία, η CPU πρέπει να στείλει τις εντολές μαζί με τα απαραίτητα δεδομένα στη GPU, η οποία είναι υπεύθυνη για την εκτέλεση αυτών των εντολών. Η σειρά εντολών που είναι κατανοητές από τη GPU σχηματίζουν τη διεπαφή προγραμματισμού εφαρμογών της (API). Η OpenGL είναι ένα τέτοιο παράδειγμα μιας API γραφικών και υποστηρίζεται από τις περισσότερες GPU αυτή τη στιγμή με την έννοια ότι μπορούν να κατανοήσουν τις εντολές της ή ότι τουλάχιστον μπορούν να τις μεταφράσουν σε εντολές που είναι κατανοητές από τη GPU. Η OpenGL δεν είναι η μοναδική API γραφικών. Οι πιο γνωστές εναλλακτικές είναι οι ιδιόκτητες API που χρησιμοποιούνται από την Apple και τη Microsoft. Το λογισμικό Mac OS της Apple χρησιμοποιεί μια API γνωστή ως Metal, ενώ η Microsoft την Direct3D. Η OpenGL μπορεί παρόλα αυτά να χρησιμοποιηθεί αρμονικά με τα δύο λογισμικά. Επιπλέον, ένα μια πρόσφατη ανοιχτή API ονόματι Vulkan έχει οριστεί ως διάδοχος της OpenGL και είναι μια πολύπλοκη API χαμηλού επιπέδου που είναι σχεδιασμένη περισσότερο με γνώμονα τη ταχύτητα και την αποδοτικότητα παρά την ευκολία χρήσης και πιθανότητα να μην αντικαταστήσει την OpenGL ποτέ πλήρως. [\[E21\]](#)

### 2.3.2.1 Τρισδιάστατα Γραφικά (3D Graphics)

Όταν αναφερόμαστε στα τρισδιάστατα γραφικά, εννοούμε τη δημιουργία, απεικόνιση και χειρισμό αντικειμένων σε έναν υπολογιστή σε τρεις διαστάσεις (ύψος, πλάτος, βάθος). Όλα τα τρισδιάστατα αντικείμενα, μπορούν να υποστούν περιστροφή και να προβληθούν από όλες τις γωνίες και επίσης να υποστούν αλλαγή μεγέθους. Τα αντικείμενα μπορούν να έχουν γνωρίσματα (attributes) που σε συνδυασμό με καθολικές ιδιότητες της σκηνής μέσα στην οποία βρίσκονται καθορίζουν την εμφάνιση τους. Συνήθως το εύρος των βασικών αντικειμένων είναι πολύ περιορισμένο και συνεπώς πολύπλοκα αντικείμενα όπως μια σφαίρα ή ένα πολύγωνο δημιουργούνται ως συνδυασμός των βασικών αντικειμένων (π.χ. συνδυασμός τριγώνων). Για μπορέσουμε να δημιουργήσουμε δισδιάστατες εικόνες, η σκηνή προβάλλεται σε δύο διαστάσεις αντί για τρεις, όπως όταν βγάζουμε μια φωτογραφία ενός δωματίου. Αυτό το αποτέλεσμα αποτελεί συνδυασμό τριών παραγόντων:

- **Γεωμετρία:** Αναφέρεται στις γεωμετρικούς μετασχηματισμούς που είναι η κλιμάκωση (scaling), η περιστροφή (rotation) και η μετάφραση (translation). Η κλιμάκωση είναι υπεύθυνη για τον ορισμό μεγέθους των διαστάσεων του αντικειμένου, η περιστροφή για τον προσανατολισμό του αντικειμένου γύρω από έναν ή περισσότερους άξονες και η μετάφραση για τις συντεταγμένες του αντικειμένου στο χώρο.
- **Εμφάνιση:** Αναφέρεται στο υλικό (material) του αντικειμένου και τις ιδιότητές του όπως η υφή (texture), καθώς και στον φωτισμό του (lighting). Η υφή είναι ένας τρόπος διαφοροποίησης των ιδιοτήτων του υλικού σε όλο το εύρος του, όπως διαφορετικοί χρωματισμοί, αλλαγή ορατότητας και αλλαγή του μεγέθους τοπικά. Ο φωτισμός είναι ένα απαραίτητο εξάρτημα για την ομαλή εμφάνιση της σκηνής και μπορούν να υπάρχουν πολλαπλοί φωτισμοί. Κάθε πηγή φωτός μπορεί να έχει το δικό της χρώμα, ένταση, προσανατολισμό και θέση. Το φως αυτών των πηγών θα αλληλεπιδράσει με τις ιδιότητες του υλικού κάθε αντικειμένου στη σκηνή. Η υποστήριξη για τον φωτισμό σε ένα γραφικό σύστημα μπορεί να ποικίλει από σχετικά απλή έως και αρκετά πολύπλοκη και υπολογιστικά εντατική.
- **Εικόνα:** Ο τελικός στόχος των τρισδιάστατων γραφικών είναι να παράγουν δισδιάστατες εικόνες του τρισδιάστατου κόσμου. Η μετατροπή από 3D σε 2D εμπλέκει τις ιδιότητες της προβολής (projection) και της θέασης (viewing). Μια σκηνή φαίνεται διαφορετική από διαφορετικές γωνίες θέασης. Για να δημιουργήσουμε μια γωνία θέασης, πρέπει να προσδιορίσουμε τη θέση του θεατή και τη κατεύθυνση προς την οποία κοιτάζει. Επίσης είναι απαραίτητο να προσδιορίσουμε μια “άνω” κατεύθυνση, που θα δείχνει προς τα πάνω στη τελική εικόνα. Αυτή η κατεύθυνση μπορεί να ερμηνευθεί ως μια εικονική κάμερα που τοποθετούμε στη σκηνή. Όταν η γωνία θέασης δημιουργηθεί, η σκηνή όπως φαίνεται από αυτήν μπορεί να προβληθεί σε 2D.



Εικόνα 2.5 Παράδειγμα μιας τρισδιάστατης σκηνής. Η τρέχουσα θέα βασίζεται στη γωνία της κάμερας και στις πηγές φωτισμού.

Το τελικό βήμα είναι να αναθέσουμε χρώματα σε όλα τα εικονοστοιχεία της 2D εικόνας. Αυτή η διαδικασία ονομάζεται ραστεροποίηση (rasterization), ενώ η όλη διαδικασία δημιουργίας της εικόνας ορίζεται ως απόδοση (rendering) της σκηνής. [\[E21\]](#)

### 2.3.3 Η Γλώσσα Προγραμματισμού C#

Η C# είναι μια απλή, μοντέρνα, αντικειμενοστρεφής και type-safe (προσπαθεί να αποτρέψει τη δημιουργία εσφαλμένων εντολών) γλώσσα προγραμματισμού. Εκτός από την αντικειμενοστρέφεια της, η C# περιέχει υποστήριξη για εξαρτηματοστρεφή προγραμματισμό (component-oriented). Ο σύγχρονος σχεδιασμός λογισμικού εξαρτάται όλο και περισσότερο σε εξαρτήματα λογισμικού στη μορφή αυτοδύναμων και αυτο-περιγραφόμενων πακέτων λειτουργικότητας. Το κλειδί σε τέτοιου είδους εξαρτήματα, είναι ότι παρουσιάζουν ένα προγραμματιστικό μοντέλο με ιδιότητες, μεθόδους και συμβάντα, έχουν γνωρίσματα που παρέχουν δηλωτικές πληροφορίες για το εξάρτημα και ενσωματώνουν τη δική τους τεκμηρίωση. Η C# παρέχει γλωσσικές κατασκευές για να μπορέσει να υποστηρίξει άμεσα αυτές τις ιδέες, κάνοντάς την πολύ φυσική γλώσσα με την έννοια ότι δημιουργεί και χρησιμοποιεί λογισμικά εξαρτήματα.

Πολλά χαρακτηριστικά της C# βοηθούν στη κατασκευή εύρωστων και ανθεκτικών εφαρμογών. Για παράδειγμα ο χειρισμός εξαιρέσεων (exception handling) παρέχει μια δομημένη και επεκτάσιμη προσέγγιση στην εύρεση σφαλμάτων σε ένα πρόγραμμα και την ανάκτηση του και η συλλογή “απορριμμάτων” (garbage collection) αυτόματα ανακτά τη κατελιμμένη μνήμη από αντικείμενα που δε χρησιμοποιούνται.

Η C# έχει ένα ενοποιημένο σύστημα τύπων. Όλοι οι τύποι της C#, συμπεριλαμβανομένων των βασικών τύπων όπως ο int και ο double, κληρονομούν από ένα τύπο αντικειμένου μονής ρίζας. Συνεπώς, όλοι οι τύποι μοιράζονται ένα σύνολο κοινών

λειτουργιών και τιμές κάθε τύπου μπορούν να αποθηκευτούν, να μεταφερθούν και να λειτουργήσουν με ένα σταθερό τρόπο. Επιπλέον, η C# υποστηρίζει και αναφορικούς τύπους ορισμένους από το χρήστη, αλλά και τύπους τιμών, επιτρέποντας έτσι δυναμική κατανομή αντικειμένων αλλά και γραμμική αποθήκευση ελαφρών δομών.

Για να εξασφαλίσουμε ότι τα προγράμματα και οι βιβλιοθήκες της C# μπορούν να εξελιχθούν με το πέρασμα του χρόνου με αρμονικό τρόπο, έχει δοθεί ιδιαίτερη έμφαση στην εκδοτικότητα (versioning) του σχεδιασμού της C#. Πολλές γλώσσες προγραμματισμού δεν δίνουν ιδιαίτερη προσοχή σε αυτό το πρόβλημα και ως αποτέλεσμα, προγράμματα που είναι γραμμένα σε τέτοιες γλώσσες αλλοιώνονται πολύ πιο συχνά όταν νέες εκδόσεις εξαρτώμενων βιβλιοθηκών κυκλοφορούν. [\[HTWG10\]](#)

## 3. Ανάπτυξη Βιντεοπαιχνιδιού

### 3.1 Περιγραφή

Σκοπός μας είναι η δημιουργία ενός εκπαιδευτικού βιντεοπαιχνιδιού εκμάθησης της κυκλοφοριακής αγωγής που να απευθύνεται σε μαθητές δημοτικού. Η δημιουργία του θα πραγματοποιηθεί με τη χρήση της μηχανής βιντεοπαιχνιδιών Unity και είναι τρισδιάστατης μορφής (3D). Ο παίκτης, μέσω συνεχούς καθοδήγησης από το ίδιο το παιχνίδι θα πρέπει να ολοκληρώσει επιτυχώς έναν ορισμένο αριθμό επιπέδων, όπου κάθε ένα από αυτά θα τον τοποθετεί σε ένα ρεαλιστικό σενάριο στη θέση του οδηγού. Στο σύνολό του το παιχνίδι θα περιέχει:

- Λειτουργικό μενού με επιλογή αλλαγής ρυθμίσεων (Main Menu & Settings Menu)
- Τέσσερα επίπεδα κλιμακούμενης δυσκολίας εκτέλεσης αλλά και δημιουργίας από τη πλευρά του δημιουργού, με στόχο τη διαφήμιση των γενικών δυνατοτήτων ενός τέτοιου παιχνιδιού
- Μεγάλο περιβάλλον παιχνιδιού που να προσφέρει τη δυνατότητα εξερεύνησης αλλά και ποικίλων συνδυασμών αποστολών
- Εισαγωγή αυτοκινήτων και πεζών μη χειρίσιμων από τον παίκτη (NPCs) για να συνεισφέρουν στην εκπαίδευση του παίκτη αλλά και για να κάνουν το περιβάλλον πιο ζωντανό και ενδιαφέρον
- Ψηφιακό εκπαιδευτή ο οποίος θα καθοδηγεί τον παίκτη σε ό,τι αφορά τις αποστολές που πρέπει να ολοκληρώσει, την εκπαίδευσή του σε σχέση με το τι πρέπει να διδαχθεί από κάθε αποστολή αλλά και στο να προσφέρει γενικές οδηγίες (π.χ. χειρισμός αυτοκινήτου)
- Μικρές λεπτομέρειες για τη βελτίωση της γενικής εικόνας του παιχνιδιού (μουσική και κατάλληλες εικόνες)

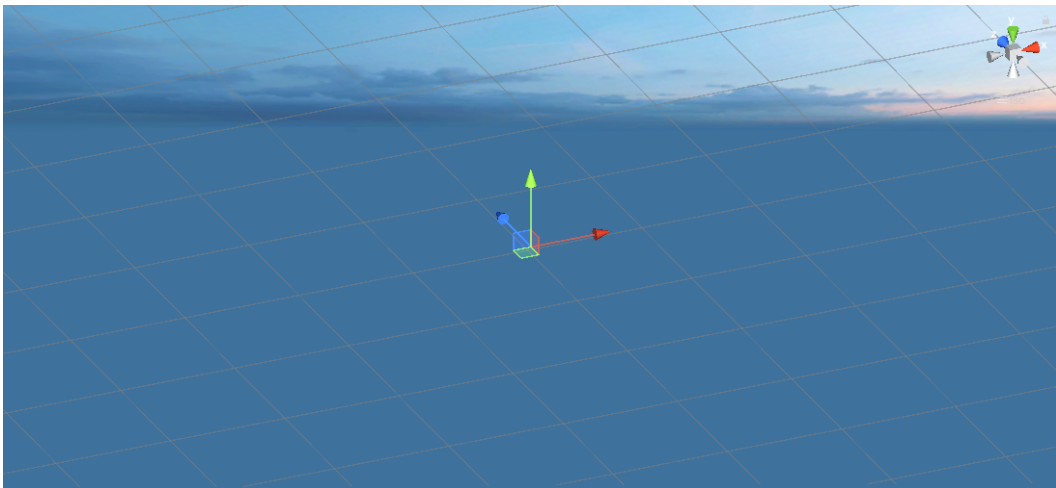
### 3.2 Πορεία Ανάπτυξης Παιχνιδιού

Για να υλοποιηθεί σωστά και με επιτυχία το βιντεοπαιχνίδι θα ξεκινήσουμε με τη δημιουργία του περιβάλλοντος πάνω στο οποίο θα βασιστούν όλα τα επίπεδα. Αυτό θα βοηθήσει σημαντικά στη μετέπειτα ανάπτυξη των επιπέδων σε ότι αφορά τη διαφοροποίηση των αποστολών αλλά και στην εξοικονόμηση χρόνου, καθώς ένα μεγάλο περιβάλλον είναι πιο εύκολο να αξιοποιηθεί σε βάθος πολλαπλών επιπέδων συγκριτικά με τη δημιουργία νέου περιβάλλοντος για κάθε επίπεδο. Στη συνέχεια, θα ενσωματώσουμε το όχημα που θα χειρίζεται ο χρήστης εφαρμόζοντας τον κατάλληλο μηχανισμό κίνησης και χειρισμού πάνω σε αυτό. Έπειτα, θα ξεκινήσει η κατασκευή του αρχικού μενού και του μενού ρυθμίσεων ώστε να έχουμε μια ξεχωριστή βάση σκηνών. Στη συνέχεια θα ετοιμάσουμε το μενού επιλογής επιπέδου που θα μας δώσει τη δυνατότητα να επιλέξουμε τα διαφορετικά επίπεδα

που θα δημιουργηθούν, ενώ αμέσως μετά θα δημιουργήσουμε τα επίπεδα με βάση τις απαιτήσεις που έχουν οριστεί με τη προσθήκη ενός μενού παύσης, κοινό για όλα τα επίπεδα που θα συνδέει τα μενού με το κύριο μέρος του παιχνιδιού. Τέλος, θα γίνει μια πειραματική δοκιμή του παιχνιδιού για να σημειωθούν και να διορθωθούν προβλήματα που ενδέχεται να προκύψουν κατά την ανάπτυξή του. Η πορεία αυτή εξηγείται αναλυτικά παρακάτω.

### 3.2.1 Δημιουργία Περιβάλλοντος Βιντεοπαιχνιδιού

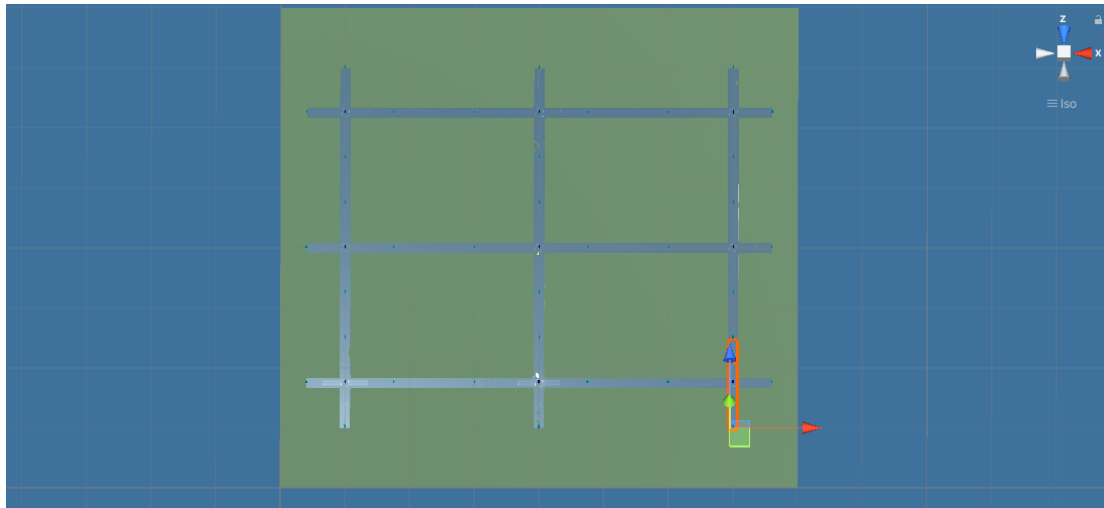
Όταν ανοίγουμε το Unity και δημιουργούμε το νέο μας project, οδηγούμαστε στη διεπαφή χρήστη όπου έχουμε όλα μας τα εργαλεία αλλά και τη σκηνή στην αρχική της κατάσταση. Εντός της σκηνής έχουμε μια κάμερα, μια πηγή φωτισμού και ένα εικονικό δάπεδο που μας υποδεικνύει που είναι το default ύψος για να δημιουργήσουμε το περιβάλλον μας. Αρχικά, προσθέτουμε ένα Skybox στη σκηνή που θα αποτελέσει τον ουρανό του περιβάλλοντος και θα συμβάλλει στο να γίνει το παιχνίδι πιο ευπαρουσίαστο.



Εικόνα 3.1 Η default σκηνή του Unity έπειτα από τη προσθήκη προσαρμοσμένου Skybox

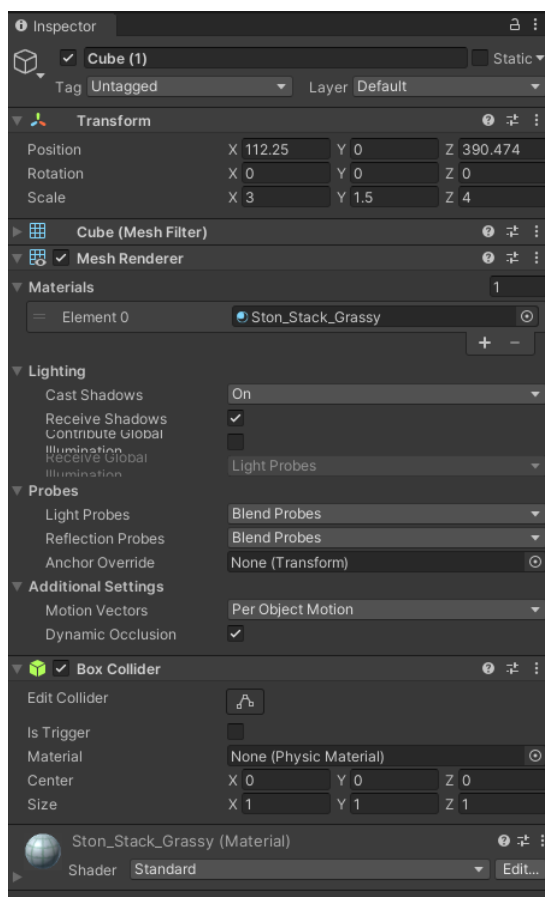
Στη συνέχεια, θα ξεκινήσουμε με την υλοποίηση της αρχικής μορφής του περιβάλλοντος, διαμορφώνοντας τους δρόμους γύρω από τους οποίους θα λάβει χώρα το παιχνίδι μας. Στην αρχή θα φτιάξουμε ένα έδαφος μέσα από τις κατασκευαστικές επιλογές του Unity, το οποίο θα έχει πράσινο χρώμα και θα σηματοδοτεί ότι υπάρχει παρουσία γρασιδιού, κάνοντας μετέπειτα πιο εύκολη τη διαφοροποίηση των υπόλοιπων στοιχείων του περιβάλλοντος. Στη συνέχεια, θα υλοποιήσουμε το οδικό δίκτυο του παιχνιδιού πράγμα που θα γίνει με τη βοήθεια του Road Architect. Το Road Architect είναι ένα πακέτο ειδικά σχεδιασμένο για το Unity και προσφέρει εύκολη και γρήγορη κατασκευή ενός πλήρους εξοπλισμένου οδικού δικτύου ανάλογα με τις απαιτήσεις του δημιουργού. Λειτουργεί με ένα απλό σύστημα κόμβων που τοποθετούνται κατάλληλα από τον χρήστη και κάθε φορά που δύο κόμβοι συνδέονται μεταξύ τους δημιουργείται αυτόματα ένας δρόμος μονής ή διπλής κατεύθυνσης με αριθμό λωρίδων που ορίζεται από τον χρήστη. Αν δύο συνδυασμοί κόμβων συναντιούνται, τότε αυτόματα δημιουργείται μια διασταύρωση, πλήρως εξοπλισμένη με φανάρια ή κατάλληλες πινακίδες για ομαλή διέλευση της κυκλοφορίας.





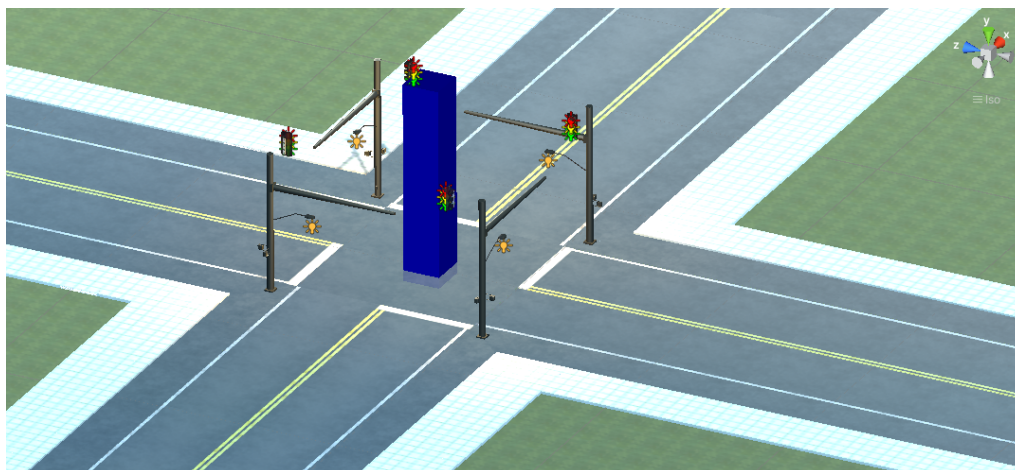
Εικόνα 3.2 Κάτοψη του περιβάλλοντος έπειτα από τη δημιουργία του οδικού δικτύου

Επόμενο μας βήμα είναι να περικλείσουμε το οδικό δίκτυο που δημιουργήθηκε, ώστε να διαχωρίσουμε το δρόμο από τα κτίρια που θα τοποθετηθούν στη συνέχεια, αλλά και για να κάνουμε δυνατή τη διέλευση των πεζών εντός της πόλης, όπως συμβαίνει και στη κανονική ζωή. Για να το πετύχουμε αυτό, θα μορφοποιήσουμε κατάλληλα τον κύβο που μας προσφέρει έτοιμο το Unity και θα του δώσουμε το κατάλληλο texture για να πάρει τη μορφή πεζοδρομίου.



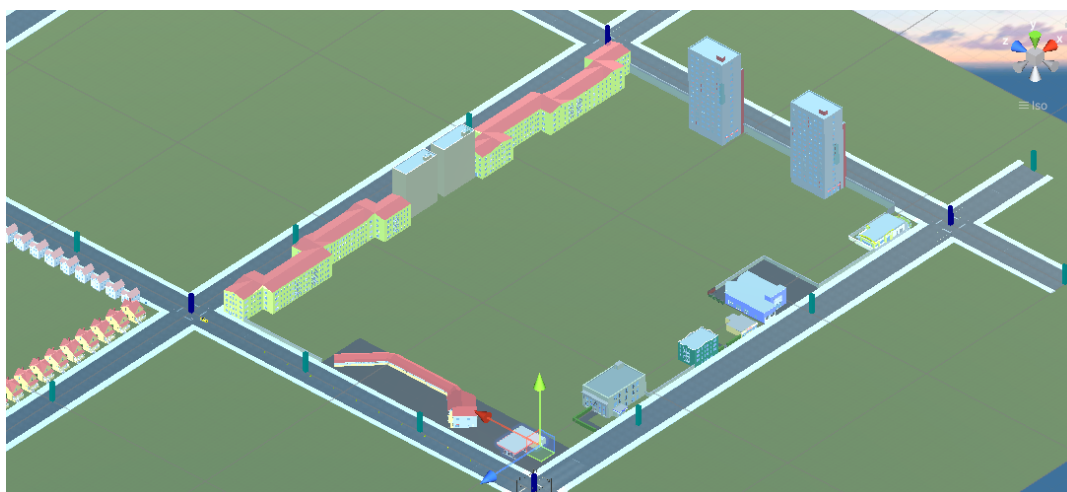
Για να γίνει αυτό, μέσα στον επιθεωρητή του αντικειμένου μας στην επιλογή “Materials” προσθέτουμε το αντίστοιχο texture, ενώ με την επιλογή “Transform” μορφοποιούμε τον κύβο ώστε να είναι λίγο πιο ψηλός σε ύψος από τον οδικό δίκτυο και να έχει ικανό βάθος και πλάτος για να υπάρχει αρκετός χώρος για την κυκλοφορία των πεζών. Μόλις το αντικείμενό μας είναι έτοιμο, θα πολλαπλασιαστεί τόσες φορές όσες χρειάζεται για να καλυφθεί ολοκληρωτικά το οδικό δίκτυο και θα τοποθετηθεί κατάλληλα ώστε να διαχωρίζει το έδαφος με τους δρόμους που τοποθετήθηκαν. Επιπλέον όλα τα αντικείμενα θα έχουν τα αντίστοιχα colliders τους ενεργοποιημένα, ώστε όταν ένα αυτοκίνητο χτυπήσει ένα πεζοδρόμιο ή ένα φανάρι να μην “περάσει” από μέσα του (ghosting) αλλά και ώστε όλα τα αντικείμενα να μείνουν στο χώρο και να μην “πέσουν” εκτός της σκηνής.

Εικόνα 3.3 Το μενού ρυθμίσεων κάθε αντικείμενου



*Εικόνα 3.4 Μια διασταύρωση με σηματοδότες στο ολοκληρωμένο οδικό δίκτυο, έπειτα από την τοποθέτηση των πεζοδρομίων*

Θεωρητικά, το περιβάλλον αυτό είναι αρκετό για την ομαλή υλοποίηση του παιχνιδιού μας αλλά θα επιχειρήσουμε να το κάνουμε πιο ενδιαφέρον αλλά και πιο διασκεδαστικό στα μάτια του παίκτη. Αυτό θα γίνει με τη προσθήκη κτιρίων και λοιπών κατασκευών που θεωρητικά μπορεί να συναντήσει κανείς σε μια πόλη. Είναι σημαντικό να εκμεταλλευτούμε το διαθέσιμο χώρο κατάλληλα, αλλά ταυτόχρονα είναι θεμιτό να μην υπερφορτώσουμε το παιχνίδι ώστε να μπορεί να είναι εκτελέσιμο ομαλά και σε υπολογιστές με χαμηλές επιδόσεις. Όλα τα αντικείμενα τοποθετούνται με απλό “drag & drop” στο σημείο που θέλουμε, φροντίζοντας πάντα τα colliders τους να είναι ενεργοποιημένα.



*Εικόνα 3.5 Κάτοψη του περιβάλλοντος με τα κτίρια που έχουν προστεθεί*

Κατά την τοποθέτηση των κτιρίων, φροντίζουμε ο παίκτης να μην έχει πρόσβαση στο χώρο πίσω από αυτά, κάτι που επιτυγχάνουμε με την τοποθέτηση τοίχων ή παρόμοιων εμποδίων σε όλο το εύρος των περιοχών που περιέχουν κτήρια. Τέλος, στη περίπτωση μας διαλέξαμε να απενεργοποιήσουμε τις φυσικές σκιές των αντικειμένων, γιατί με τη τρέχουσα επιλογή φωτισμού, το περιβάλλον ήταν σκοτεινό εμποδίζοντας έτσι την όραση των παικτών.

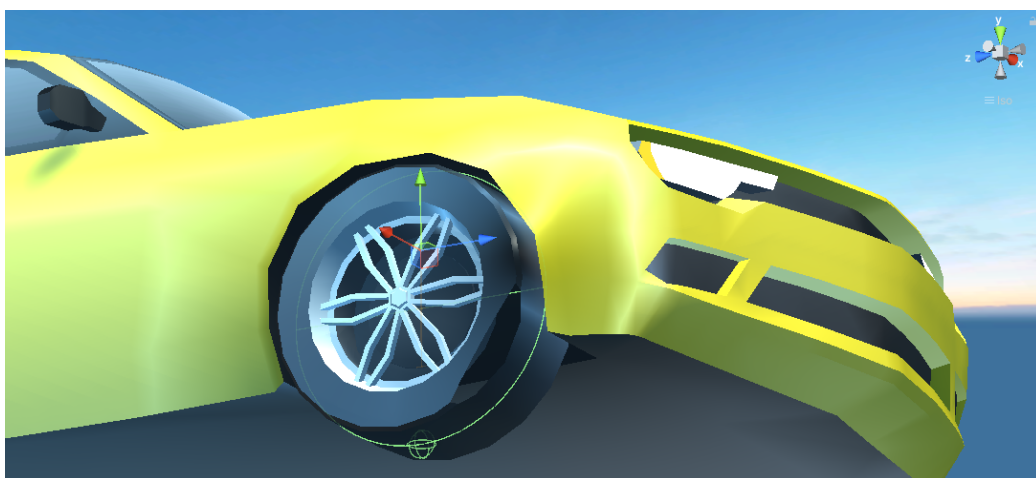
### 3.2.2 Δημιουργία Αυτοκινήτου Παίκτη

Το επόμενο βήμα αφού δημιουργήσουμε το περιβάλλον του παιχνιδιού είναι να υλοποιήσουμε κατάλληλα το χειρισμό αλλά και τα physics του αυτοκινήτου που θα χρησιμοποιήσει ο παίκτης. Ως βάση, θα χρησιμοποιήσουμε ένα από τα έτοιμα αυτοκίνητα που προσφέρει τα πακέτα του Unity. Το αυτοκίνητο χωρίζεται σε δύο μέρη, το κυρίως μέρος (body) και τις ρόδες (wheels). Οι ρόδες και το σώμα του αυτοκινήτου έχουν τα δικά τους ξεχωριστά colliders, ενώ σε κάθε μέρος θα προσθέσουμε το εξάρτημα “Rigidbody” που μας δίνει τη δυνατότητα να προσθέτουμε την απαραίτητη βαρύτητα στο αυτοκίνητό μας ώστε να αντιδρά ρεαλιστικά στις εντολές που θα του δίνουμε. (δηλαδή να παραμένει σταθερά στο έδαφος, να αποφευχθεί η πιθανότητα να αναποδογυρίσει κτλ)



Εικόνα 3.5 Το αυτοκίνητο του παίκτη σε ακίνητη κατάσταση

Όπως έχουμε ήδη αναφέρει τα colliders χρησιμεύουν ώστε το αυτοκίνητο να μη μπορεί να περάσει μέσα από άλλα αντικείμενα που δεν θέλουμε, όπως τα κτίρια ή το πεζοδρόμιο, ενώ είναι αυτόματα σε σχήμα κατάλληλο ώστε να “αγκαλιάζουν” με ακρίβεια το σώμα και τις ρόδες του αυτοκινήτου.



Εικόνα 3.6 Παράδειγμα collider τροχού του αυτοκινήτου

Στη συνέχεια, θα ρυθμίσουμε τη κάμερα ώστε να ακολουθεί κατάλληλα το αυτοκίνητο κατά τη κίνησή του. Το παιχνίδι, ώντας τρισδιάστατης φύσεως, δεν θα μπορέσει να επηρετήσει κατάλληλα μια “top-down” θέα. “Top-down” σημαίνει ότι η κάμερα βρίσκεται στον ουρανό και κοιτάζει προς το περιβάλλον ως κάτοψη (βλέπε Εικόνα 3.2). Έτσι, όπως στα περισσότερα κλασσικά 3D βιντεοπαιχνίδια, θα τοποθετήσουμε τη κάμερα ακριβώς πίσω από το αυτοκίνητο και ελαφρώς πιο πάνω από το ύψος του ώστε ο παίκτης να έχει μια καθαρή εικόνα για το που κατευθύνεται. Για να το πετύχουμε αυτό, θα φτιάξουμε ένα script σε C# ονόματι [CameraFollow](#) που θα κατευθύνει τη κάμερα στη σωστή διεύθυνση ως προς τη μετακίνηση (translation) και τη περιστροφή (rotation) της. Το script, θα μας δίνει τη δυνατότητα να ρυθμίσουμε την αρχική θέση της κάμερας, το στόχο (αυτοκίνητο παίκτη) και το ρυθμό μεταβολής της μετακίνησης και της περιστροφής της κάμερας.



Εικόνα 3.7 Η θέα του παίκτη έπειτα από την εφαρμογή του *CameraFollow*

Το τελευταίο βήμα είναι να δώσουμε “ζωή” στο αυτοκίνητο του παίκτη, δηλαδή να του δώσουμε τη δυνατότητα κίνησης. Ένα αυτοκίνητο πρέπει να επιταχύνει, να επιβραδύνει, να μπορεί να κάνει όπισθεν και να μπορεί να στρίψει δεξιά και αριστερά. Για την υλοποίηση όλων των παραπάνω, θα δημιουργήσουμε πάλι ένα script σε C# ονόματι [CarController](#), το οποίο θα μας δίνει τη δυνατότητα να πραγματοποιούμε όλες τις παραπάνω ενέργειες με τη χρήση του πληκτρολογίου. Ταυτόχρονα, το script θα ελέγχει κάθε τροχό ξεχωριστά και θα αναθέτει ενέργειες στη κάθε μία ανάλογα με τη τρέχουσα ενέργεια. Για παράδειγμα για την επιτάχυνση και την περιστροφή του αυτοκινήτου, εντολές δέχονται μόνο οι δύο μπροστινοί τροχοί με τους δύο πίσω τροχούς να περιστρέφονται ανάλογα, όπως δηλαδή συμβαίνει και με τα αυτοκίνητα με μπροστινή κίνηση στη πραγματική ζωή. Σε ό,τι αφορά την επιβράδυνση, και οι τέσσερις τροχοί συμβάλλουν στην επιβράδυνση του. Αυτές οι ενέργειες είναι δυνατόν να συμβούν χάρη στο γεγονός ότι μπορούμε να συνδέσουμε τους τροχούς και τα colliders τους στο script, ώστε κάθε φορά να ανταποκρίνονται μόνο οι τροχοί που θέλουμε. Ταυτόχρονα, μας δίνεται η δυνατότητα να ρυθμίσουμε μόνοι μας τις ρυθμού επιτάχυνσης και επιβράδυνσης, καθώς και τη μέγιστη τιμή περιστροφής των μπροστινών τροχών για το καλύτερο δυνατό χειρισμό.





*Εικόνα 3.8 Το αυτοκίνητο του παίκτη σε στιγμιότυπο περιστροφής μετά την εφαρμογή του CarController*

### 3.2.3 Δημιουργία Μενού Παιχνιδιού

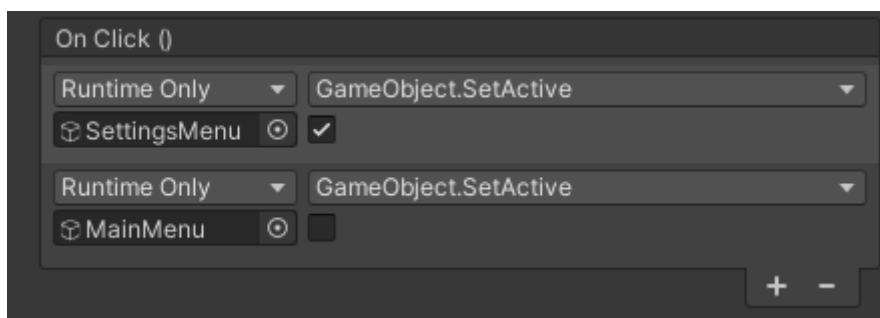
Αφού έχουμε ολοκληρώσει το βασικό κορμό γύρω από τον οποίο θα σχηματιστεί ολόκληρο το παιχνίδι, ήρθε η ώρα να δημιουργήσουμε τα απαραίτητα μενού τα οποία θα κάνουν πιο φυσική τη μετάβαση του παίκτη στο παιχνίδι αλλά και θα δημιουργήσει έναν ατέρμονο κύκλο ενεργειών έτσι ώστε όσο ο παίκτης βρίσκεται εντός του παιχνιδιού, να μπορεί να εισέλθει και να εξέλθει ελεύθερα από οποιαδήποτε σκηνή επιθυμεί. Για τη δημιουργία όλων των μενού θα χρησιμοποιηθούν απλά πλαίσια (panels) με κουμπιά (buttons) που υπάρχουν έτοιμα στο Unity και προστίθενται στη σκηνή με απλό drag & drop. Στη συνέχεια εμείς θα μορφοποιήσουμε κατάλληλα την εμφάνισή τους και έπειτα με χρήση απλών script σε C# θα καθορίσουμε τη λειτουργία του κάθε εξαρτήματος.

- **Κυρίως Μενού (Main Menu)**



*Εικόνα 3.9 Το αρχικό μενού του παιχνιδιού*

Για το αρχικό μενού, και τα υπόλοιπα panel που θα συνδέονται με αυτό, θα χρησιμοποιήσουμε ένα background το οποίο φτιάχτηκε και προσαρμόστηκε κατάλληλα με τη βοήθεια του λογισμικού επεξεργασίας γραφικών GIMP. Πάνω από αυτό, θα προσθέσουμε τρία κουμπιά που θα έχουν το δικό τους background, ώστε να “φωτίζουν” κάθε φορά που ο χρήστης περνάει με το ποντίκι από πάνω τους. Το πρώτο κουμπί με την ένδειξη “Stage Select”, θα οδηγεί τον παίκτη σε μια νέα οθόνη που θα εμφανίζει όλες τις διαθέσιμα επίπεδα. Το δεύτερο κουμπί με την ένδειξη “Settings”, θα οδηγεί τον χρήστη στο μενού ρυθμίσεων, ενώ το τελευταίο κουμπί με την ένδειξη “Exit”, τερματίζει το παιχνίδι και οδηγεί τον χρήστη πίσω στην επιφάνεια εργασίας. Κάθε κουμπί λειτουργεί κατάλληλα με τη χρήση της OnClick() που υπάρχει έτοιμη. Εκεί ο χρήστης μπορεί να προσθέσει εύκολα τις ενέργειες που θα εκτελεί κάθε κουμπί (π.χ. κλείσιμο/άνοιγμα σκηνής).



Εικόνα 3.10 Η λειτουργία OnClick() του Unity

- **Μενού Ρυθμίσεων**

Ο σκοπός του μενού ρυθμίσεων είναι να δώσει τη δυνατότητα στον χρήστη να αλλάξει βασικές ρυθμίσεις του παιχνιδιού σε περίπτωση που δεν είναι ικανοποιημένος με τις αρχικές ή σε περίπτωση που ο υπολογιστής του δεν μπορεί να ανταπεξέλθει στις απαιτήσεις του παιχνιδιού.



Εικόνα 3.11 Το μενού ρυθμίσεων του παιχνιδιού

Στη περίπτωση του παιχνιδιού μας, προσφέρονται τρεις βασικές ρυθμίσεις. Η πρώτη ρύθμιση είναι αυτή του ήχου. Με τη χρήση ενός slider, ο χρήστης μπορεί να αυξομειώσει τον ήχο ανάλογα με τις προτιμήσεις του, αφού το εν λόγω slider είναι συνδεδεμένο με ένα βασικό Audio Mixer του Unity που ελέγχει όλους τους ήχους του παιχνιδιού. Η δεύτερη ρύθμιση είναι αυτή της ποιότητας γραφικών. Προσφέρονται τρεις επιλογές (Χαμηλή, Μέση και Υψηλή) και ο στόχος τους είναι να προσαρμόσουν τα γραφικά του παιχνιδιού με τέτοιο τρόπο ώστε το παιχνίδι να μπορεί να εκτελεστεί ομαλά ακόμα και σε υπολογιστές χαμηλών επιδόσεων. Τέλος, προσφέρεται και η ρύθμιση της ανάλυσης που ρυθμίζει τις διαστάσεις του output του παιχνιδιού ανάλογα με την ανάλυση που υποστηρίζει ο υπολογιστής και η οθόνη του χρήστη.

Και οι τρεις επιλογές συνδέονται με το C# script ονόματι [SettingsMenu](#), ώστε να δίνονται στο παιχνίδι οι κατάλληλες τιμές ανάλογα με τις επιλογές του χρήστη. Ιδιαίτερη αναφορά αξίζει να γίνει στο κομμάτι του κώδικα που αφορά την ανάλυση. Με την εντολή “Screen.resolutions”, το script επιστρέφει όλες τις δυνατές αναλύσεις που υποστηρίζει η οθόνη του χρήστη και τις αποθηκεύει σε μονοδιάστατο πίνακα. Ο πίνακας αυτός εμφανίζεται στο dropdown menu στις ρυθμίσεις και μπορεί να έχει διαφορετικές τιμές για κάθε χρήστη. Μόλις ο χρήστης διαλέξει την επιθυμητή ανάλυση, το script λαμβάνει το πλάτος και το ύψος της επιλεγμένης ανάλυσης και το εφαρμόζει.

- **Μενού Επιλογής Επιπέδου**



Εικόνα 3.11 Το μενού επιλογής επιπέδου του παιχνιδιού

Ακολουθώντας παρόμοια φιλοσοφία, το μενού επιλογής επιπέδου περιέχει 5 επιλογές με τη μορφή κουμπιών. Οι πρώτες τέσσερις, υποδεικνύουν στον χρήστη τον αριθμό του επιπέδου που πρόκειται να επιλέξει. Κάθε ένα από αυτά τα κουμπιά είναι συνδεδεμένο με το C# script [IntroPanel](#) και κάθε ένα έχει τη δική του μοναδική συνάρτηση έναρξης σκηνής μέσα σε αυτό. Επιπλέον παρακάτω έχουμε ένα επιπλέον κουμπί που μας οδηγεί πίσω στο κυρίως μενού, σε περίπτωση που ο χρήστης δεν επιθυμεί να συνεχίσει.

- **Μενού Παύσης**



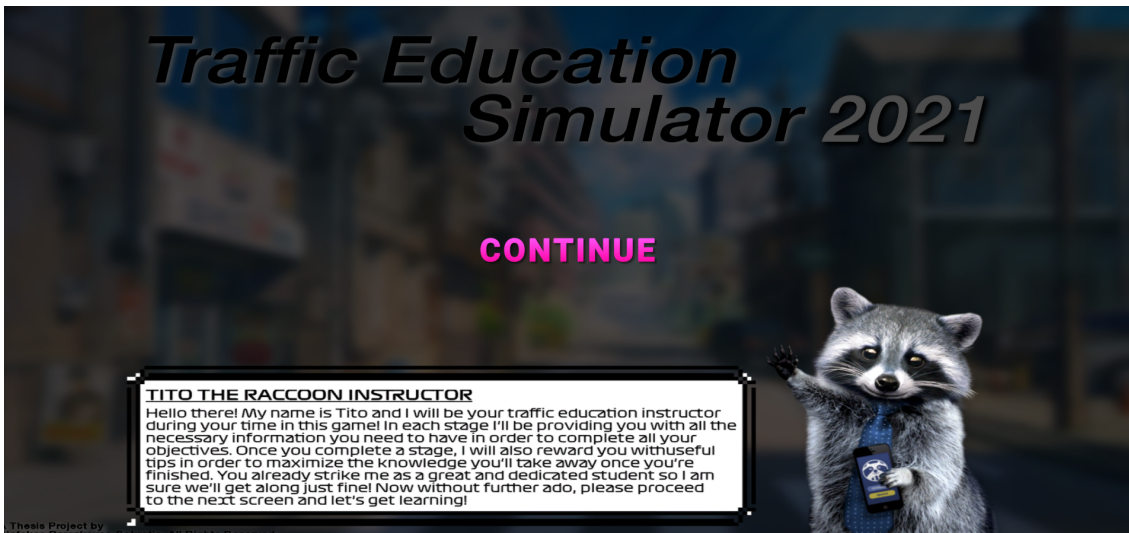
*Εικόνα 3.12 Το μενού παύσης του παιχνιδιού*

Τέλος, ολοκληρώνοντας τον αριθμό μενού που υφίστανται να υπάρχουν στο παιχνίδι μας, έχουμε το μενού παύσης. Σκοπός του είναι να δίνει τη δυνατότητα στον παίκτη να εξέλθει από το επίπεδο στο οποίο βρίσκεται ανά πάσα στιγμή, χωρίς να είναι απαραίτητο να το ολοκληρώσει. Το μενού περιέχει τρεις επιλογές. Η πρώτη επιλογή, δίνει τη δυνατότητα στον παίκτη να εξέλθει από το μενού παύσης και να συνεχίσει κανονικά το επίπεδο στο οποίο βρίσκεται. Η δεύτερη επιλογή τερματίζει το επίπεδο και επιστρέφει τον παίκτη στο κύριο μενού, ενώ η τρίτη επιλογή τερματίζει ολοκληρωτικά το παιχνίδι και επιστρέφει τον παίκτη στην επιφάνεια εργασίας. Το μενού αυτό έχει εμπλουτιστεί με ένα πιο σκοτεινό διαφανές background, ώστε να διαφοροποιείται εύκολα από τη τρέχουσα σκηνή, ενώ όταν το μενού αυτό είναι ενεργό τότε είναι σημαντικό να τονίσουμε ότι το επίπεδο σταματάει προσωρινά.

- **Πρόσθετα Διακοσμητικά Πλαίσια**

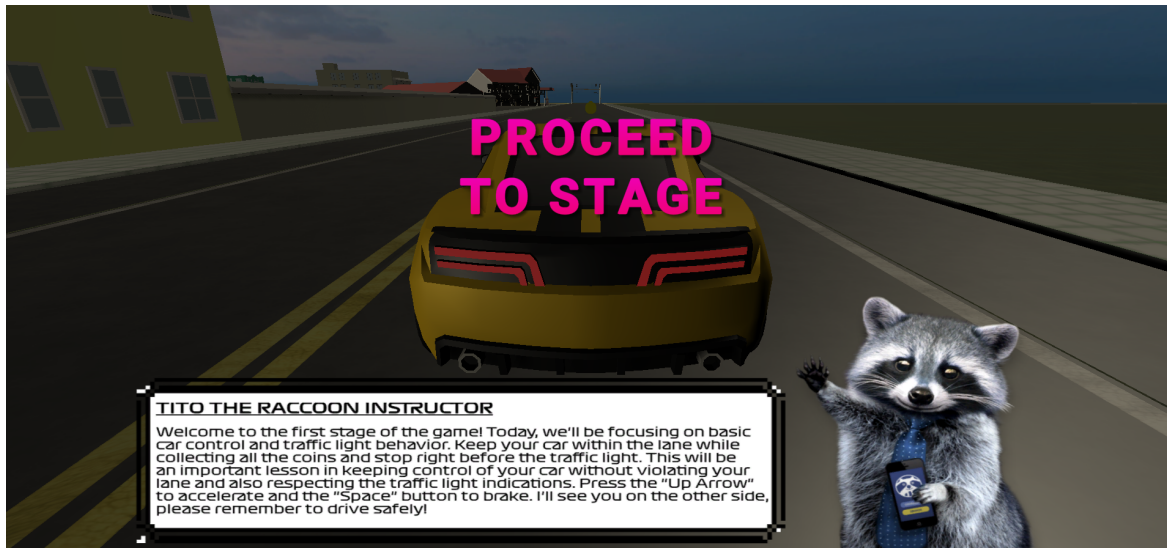
Αφού τα απαραίτητα μενού έχουν ολοκληρωθεί, μπορούμε πλέον να επικεντρωθούμε στη βελτίωση της εμφάνισης και της ροής του παιχνιδιού. Παρόλο που πρόκειται για ένα σοβαρό εκπαιδευτικό παιχνίδι, είναι επιθυμητό να έχει και κάποια διασκεδαστικά χαρακτηριστικά για μεγαλύτερη αποδοτικότητα. Στο πλαίσιο του παιχνιδιού, θα δημιουργήσουμε κάποια πλαίσια που θα ενημερώνουν και θα επιβραβεύουν τον παίκτη για τις ενέργειές του. Σε όλα τα πλαίσια που θα δημιουργήσουμε, θα ενσωματώσουμε έναν χαρακτήρα ο οποίος θα έχει το ρόλο του εκπαιδευτή. Το πρώτο πλαίσιο, θα εμφανίζεται μόλις ο παίκτης επιλέξει “Stage Select” στο κεντρικό μενού. Ο εκπαιδευτής θα καλωσορίζει τον παίκτη στο παιχνίδι και θα τον ενημερώνει για τον γενικό σκοπό του παιχνιδιού αλλά και ποιος θα είναι ο ρόλος του ίδιου σε κάθε σημείο του παιχνιδιού. Στο πλαίσιο αυτό θα υπάρχει και ένα κουμπί που ο παίκτης θα μπορεί να πατήσει όταν θα θέλει πλέον να προχωρήσει στο μενού επιλογής πίστας.





Εικόνα 3.13 Το πλαίσιο καλωσορίσματος του παιχνιδιού

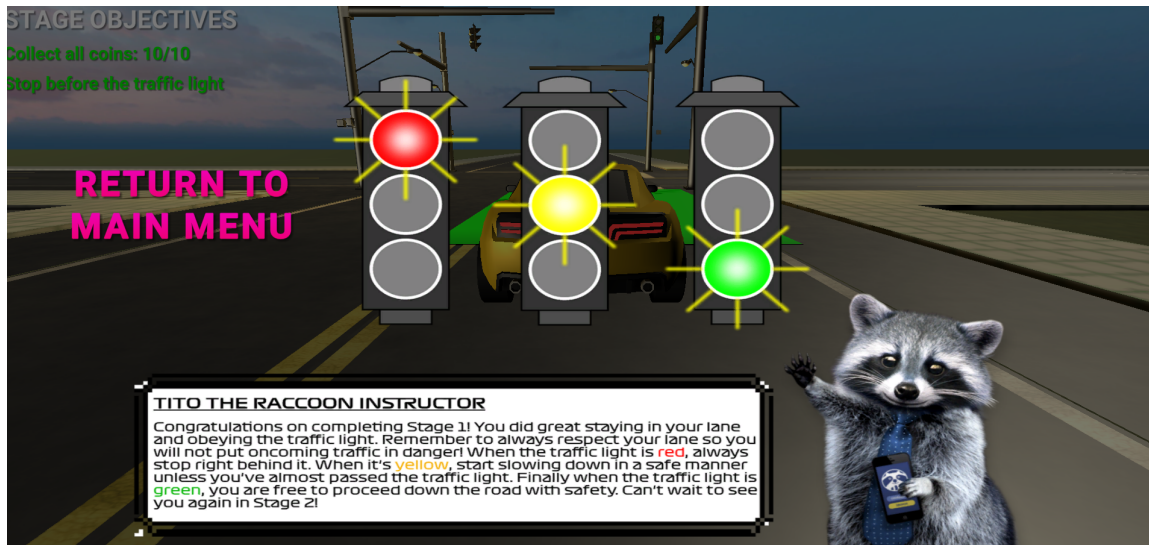
Το δεύτερο πλαίσιο που θα προσθέσουμε, θα εμφανίζεται μόλις ο παίκτης διαλέξει το επίπεδο που επιθυμεί να ολοκληρώσει. Εκεί πριν ξεκινήσει το επίπεδο, ο εκπαιδευτής θα ενημερώνει τον παίκτη για τις αποστολές που πρέπει να διεκπαιρεώσει και αν είναι αναγκαίο για το τρόπο με τον οποίο μπορεί να χειριστεί το αυτοκίνητό του. Και εκεί θα υπάρχει ένα κουμπί που ο παίκτης θα μπορεί να πατήσει όταν επιθυμεί να προχωρήσει στη διεξαγωγή του επιπέδου.



Εικόνα 3.14 Το πλαίσιο περιγραφής επιπέδου του παιχνιδιού

Το τρίτο και τελευταίο πλαίσιο που θα προσθέσουμε, θα εμφανίζεται μόλις ο παίκτης ολοκληρώσει επιτυχώς το επίπεδο στο οποίο βρίσκεται. Εκεί, ο εκπαιδευτής θα κάνει για άλλη μια φορά την εμφάνισή του, θα επιβραβεύσει τον παίκτη και θα περιγράψει τι ακριβώς διδάχτηκε στο τρέχον επίπεδο. Είναι το πιο σημαντικό πλαίσιο του παιχνιδιού, καθώς συνδέεται άμεσα με τον σκοπό του παιχνιδιού (παροχή γνώσης σχετικά με τη κυκλοφοριακή

αγωγή). Με την επιλογή ενός κουμπιού που θα βρίσκεται στο πλαίσιο, ο παίκτης θα μπορεί να επιστρέψει στο αρχικό μενού ώστε να προχωρήσει στο επόμενο επίπεδο.



Εικόνα 3.15 Το πλαίσιο ολοκλήρωσης επιπέδου του παιχνιδιού

### 3.2.4 Δημιουργία Επιπέδων

Σε αυτό το στάδιο, έχουμε στα χέρια μας ένα θεωρητικά ολοκληρωμένο παιχνίδι εντός του οποίου ο παίκτης μπορεί να οδηγήσει στη πόλη ελεύθερα, να εισέλθει και να εξέλθει από αυτήν και να μεταβεί εντός και εκτός του παιχνιδιού. Παρόλα αυτά, το παιχνίδι μας δεν έχει σκοπό, καθώς ακόμα δεν έχει αποστολές και σενάρια δημιουργημένα γύρω από αυτό. Αυτό είναι ένα άκρως απαραίτητο κομμάτι, καθώς μόνο έτσι ο παίκτης θα μπορεί να ολοκληρώσει επιτυχώς τα επίπεδα και επιπλέον αν ένα επίπεδο δεν είχε σκοπό, τότε δεν θα υπήρχε λόγος για τον παίκτη να το δοκιμάσει. Το παιχνίδι μας περιέχει τέσσερα επίπεδα τα οποία θα είναι κλιμακούμενης δυσκολίας και το καθένα είτε θα διδάσκει στον παίκτη νέες γνώσεις που σχετίζονται με τη κυκλοφοριακή αγωγή ή θα δοκιμάζει τις γνώσεις που έχει αποκτήσει μέχρι εκείνο το σημείο. Κάποια στοιχεία που θα προστεθούν θα είναι παρόντα σε όλα τα επίπεδα, ενώ κάποια άλλα θα είναι ειδικά σχεδιασμένα για μικρότερο αριθμό επιπέδων.

- **Πλαίσιο Αποστολών**

Σε κάθε επίπεδο, στο πάνω αριστερά μέρος της οθόνης, θέλουμε να αναγράφονται οι αποστολές τις οποίες πρέπει να ολοκληρώσει ο παίκτης για να μπορέσει να ολοκληρώσει το επίπεδο στο οποίο βρίσκεται. Οι αποστολές αυτές είναι μορφής κειμένου του Unity και συνδέονται με ένα ή περισσότερα στοιχεία του περιβάλλοντος μέσω ενός ή περισσότερων C# scripts. Για να ολοκληρωθεί μία αποστολή πρέπει να ικανοποιηθεί μια συνθήκη. Για παράδειγμα, αν σε μια αποστολή χρειάζεται ο παίκτης να σταματήσει πριν από μια διάβαση πεζών, τότε στο σημείο αυτό θα υπάρχει ένας ανιχνευτής σύγκρουσης (collider). Για να

ενεργοποιηθεί, θα πρέπει ο παίκτης να ακινητοποιήσει το αυτοκίνητό του εντός των ορίων που θα ορίσουμε εμείς δίνοντας στον ανιχνευτή τις ανάλογες διαστάσεις. Στον δρόμο θα είναι ορατό στον παίκτη το εύρος του ανιχνευτή αλλά στη πραγματικότητα η δουλειά θα γίνεται μέσω ενός αόρατου σχήματος, εντός του οποίου θα εισέρχεται το αυτοκίνητο. Όταν πλέον το αυτοκίνητο έχει ακινητοποιηθεί, το χρώμα του ανιχνευτή και της αποστολής στο πλαίσιο θα αλλάξει σε πράσινο, ενώ για την ολοκληρωμένη αποστολή θα υπάρχει ένα flag με αρχική τιμή 0, που θα αλλάξει σε 1 μόλις ολοκληρωθεί. Για ικανοποιηθεί η συνθήκη ολοκλήρωσης του επιπέδου, θα πρέπει κάθε flag αποστολής να πάρει τη τιμή 1. Όλα αυτά ελέγχονται μέσω του C# script [ComboTest](#), που με τη σειρά του αντλεί πληροφορίες από τα script τα οποία είναι συνδεδεμένα με τις αποστολές.



Εικόνα 3.16 Το πλαίσιο αποστολών του παιχνιδιού

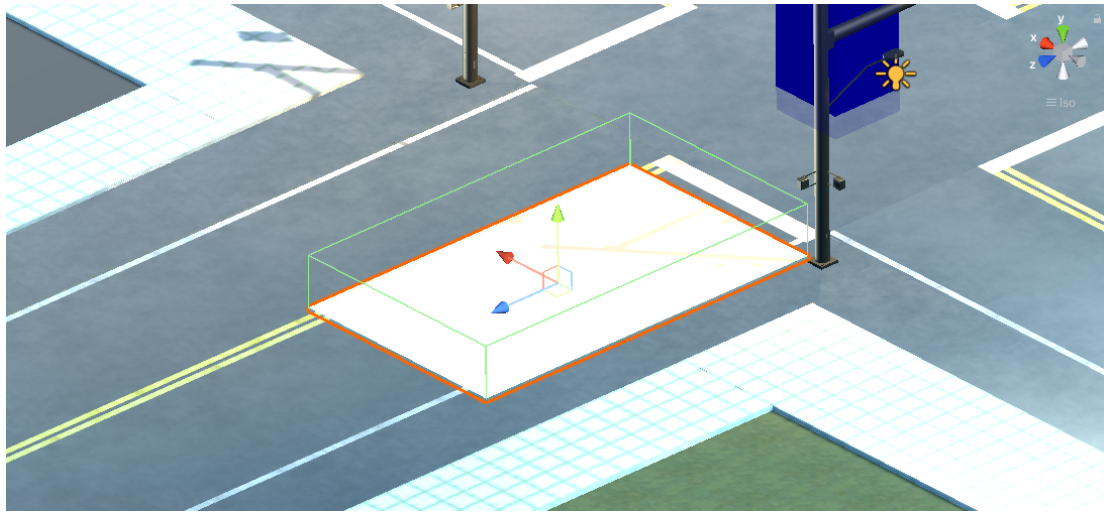
- **Ανιχνευτές Σύγκρουσης και Νομίσματα**

Η βασική φόρμουλα κάθε επιπέδου αποτελείται από αποστολές που απαιτούν από τον παίκτη να σταματήσει εντός ενός ανιχνευτή σύγκρουσης όπως αναφέρθηκε παραπάνω και να συλλέξει έναν ορισμένο αριθμό νομισμάτων που βρίσκονται γραμμικά στο περιβάλλον. Γύρω από τους ανιχνευτές σύγκρουσης θα πραγματοποιούνται τα σημαντικότερα σενάρια του επιπέδου τα οποία θα σχετίζονται άμεσα με τη κυκλοφοριακή αγωγή. Τα νομίσματα υπάρχουν στο περιβάλλον για να καθοδηγήσουν στον παίκτη προς τον τερματισμό, αλλά και για να διατηρήσουν τη προσοχή του στο παιχνίδι.

Στο κομμάτι που αφορούσε το πλαίσιο αποστολών, αναφέραμε πως λειτουργούν οι ανιχνευτές σύγκρουσης σε γενικές γραμμές. Παρακάτω θα εμβαθύνουμε περισσότερο, αναλύοντας τι συμβαίνει από πλευρά κώδικα. Κάθε ανιχνευτής συνδέεται με ένα C# script ονόματι [TriggerTest](#), το οποίο είναι υπεύθυνο για όλες τις μεταβολές που πρόκειται να υποστεί κατά τη διάρκεια του επιπέδου. Κατά την αρχικοποίηση της πίστας, το μέρος του ανιχνευτή που είναι ορατό στον παίκτη χρωματίζεται με κόκκινο και συνδέεται με το κείμενο αποστολής που του αντιστοιχεί. Όπως αναφέραμε και παραπάνω, όταν ο παίκτης ακινητοποιηθεί εντός του ανιχνευτή (κάτι που ανιχνεύεται με τη συνάρτηση

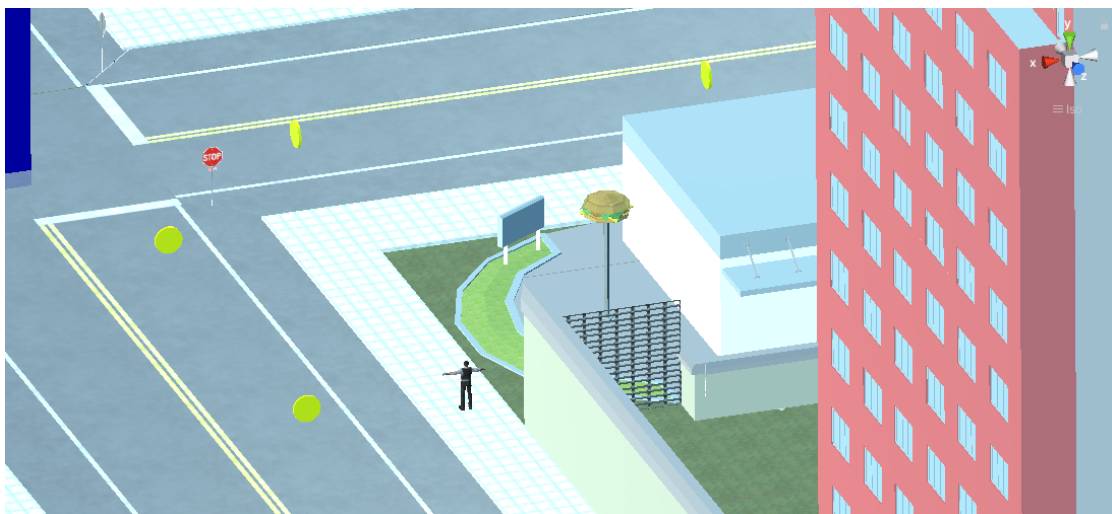


OnTriggerStay(), το χρώμα θα αλλάξει αυτόματα σε πράσινο, θα δοθεί εντολή στο κείμενο του πλαισίου αποστολών να γίνει και αυτό πράσινο και τέλος το flag του από 0 θα γίνει 1.



Εικόνα 3.17 Ένας ανιχνευτής συγκρούσεων. Το άσπρο μέρος είναι ορατό στον παίκτη και αλλάζει χρώμα ανάλογα με το αν ενεργοποιήθηκε ή όχι. Το πράσινο ορθογώνιο είναι αόρατο στον παίκτη και είναι το πεδίο στο οποίο εισέρχεται το αυτοκίνητο και ελέγχει αν είναι ακινητοποιημένο ή όχι

Μία αποστολή η οποία είναι κοινή για όλα τα επίπεδα είναι αυτή της συλλογής νομισμάτων. Τα νομίσματα αυτά είναι τροποποιημένοι κύλινδροι που παρέχονται από το Unity και σε αντίθεση με όλα τα αντικείμενα του περιβάλλοντος, δεν έχουν collider ώστε το αυτοκίνητο του παίκτη να μπορεί να περνάει από μέσα τους. Υπάρχουν δύο script που σχετίζονται με τη συλλογή νομισμάτων. Το πρώτο ονομάζεται [CollectionScript](#) και ελέγχει τη κατάσταση κάθε νομίσματος. Αν το αυτοκίνητο του παίκτη “εισέλθει” στο σώμα του νομίσματος (πράγμα που ανιχνεύεται με την εντολή OnTriggerEnter()), τότε ο τρέχων αριθμός νομισμάτων που έχουν συλλεχθεί θα αυξηθεί κατά 1 στο πλαίσιο των αποστολών, θα γίνει παραγωγή χαρακτηριστικού ήχου που θα δηλώνει ότι το νόμισμα συλλέχτηκε και τέλος, το νόμισμα θα εξαφανιστεί ολοκληρωτικά από το επίπεδο.

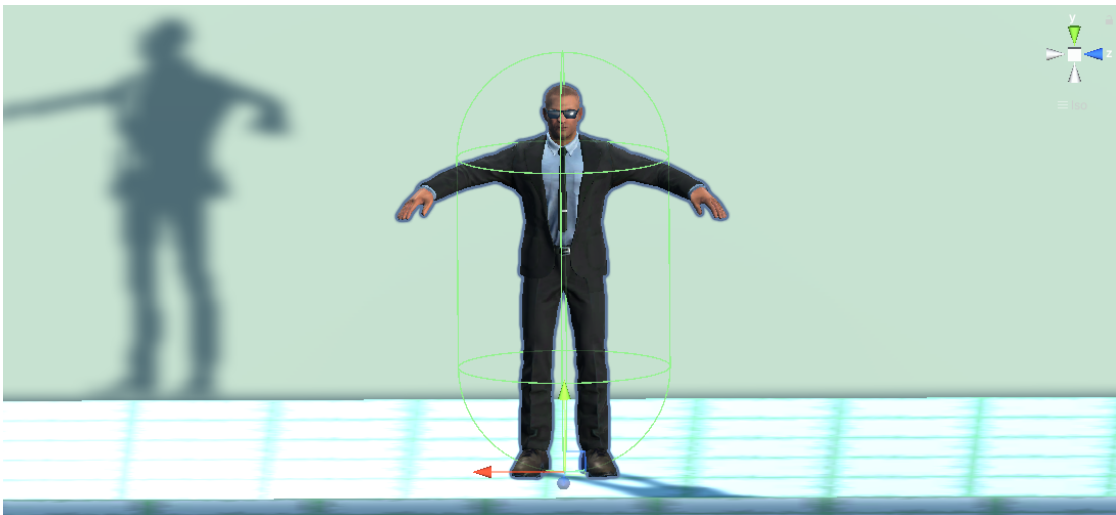


Εικόνα 3.18 Κάτοψη μέρος της πόλης στο οποίο υπάρχουν νομίσματα προς συλλογή

Το δεύτερο script, ελέγχει αν έχει ικανοποιηθεί η συνθήκη για την ολοκλήρωση της αποστολής, δηλαδή αν έχουν συλλεχθεί όλα τα νομίσματα που βρίσκονται στο επίπεδο. Ονομάζεται [CoinHandler](#) και απλά ελέγχει αν ο αριθμός νομισμάτων που έχουν συλλεχθεί μέχρι στιγμής ισούται με τον αριθμό των διαθέσιμων νομισμάτων στο επίπεδο. Όταν η συνθήκη γίνει αληθής, τότε το κείμενο στο πλαίσιο της αποστολής θα αποκτήσει πράσινο χρώμα και το flag που συνδέεται με αυτό θα αλλάξει τιμή από 0 σε 1.

- **Πεζοί και Άλλα Οχήματα**

Το τελευταίο και ίσως πιο απαιτητικό κομμάτι της ανάπτυξης αυτού του παιχνιδιού είναι η δημιουργία πεζών και οχημάτων που θα κυκλοφορούν εντός του παιχνιδιού (NPCs). Στόχος μας είναι οι NPCs να λειτουργούν ομαλά και να συμπεριφέρονται φυσικά. Η παρουσία τους θα κάνει το περιβάλλον πιο ζωντανό και ρεαλιστικό, ενώ μπορούν να χρησιμοποιηθούν και ως μέσο εκπαίδευσης του παίκτη.



Εικόνα 3.19 Μοντέλο πεζού στην αρχική του κατάσταση

Για τους πεζούς, θα χρησιμοποιήσουμε μοντέλα ανθρώπων, στα οποία θα ενσωματώσουμε κάθετα colliders σε μορφή κάψουλας (capsule colliders) όπως φαίνεται στην Εικόνα 3.17. Σε αυτά τα μοντέλα θα ενσωματώσουμε ένα C# script ονόματι [CharacterNavigationController](#), που θα είναι υπεύθυνο για τη μετακίνηση του μοντέλου. Θα λαμβάνει υπόψιν ενδεχόμενα στα οποία θα πρέπει να περιστραφεί ενώ θα έχουμε τη δυνατότητα να ρυθμίσουμε τη ταχύτητα με την οποία θα περιστρέφεται, την ελάχιστη και τη μέγιστη ταχύτητα εντός των οποίων ο ταχύτητα του πεζού θα καθορίζεται τυχαία ή αν δε το επιθυμούμε, μπορούμε να ορίσουμε καρφωτά μια σταθερή ταχύτητα. Τέλος θα δέχεται το μοντέλο του πεζού που επιθυμούμε ώστε να εφαρμοστούν σε αυτό οι παραπάνω ενέργειες σε ενδεχόμενο που χρησιμοποιήσουμε το ίδιο μοντέλο πολλαπλές φορές. Μπορεί το παραπάνω script να ελέγχει αποτελεσματικά τη μετακίνηση του πεζού αλλά αυτό δε σημαίνει πως ελέγχει και το τρόπο με τον οποίο θα μετακινείται. Με λίγα λόγια, αυτή τη στιγμή ο πεζός μπορεί να μετακινηθεί, αλλά η θέση του αλλάζει ενώ αυτός παραμένει ακίνητος, δηλαδή

χωρίς φυσική μεταβολή (περπάτημα, τρέξιμο, γενική κίνηση χεριών/ποδιών). Γι'αυτό το λόγο θα ενσωματώσουμε τη λειτουργία “Περπάτημα” στον animator του πεζού. που είναι είναι ένα έτοιμο κλιπ κίνησης και εφαρμόζεται με ένα απλό drag & drop από τα assets του Unity.



Εικόνα 3.20 Στιγμιότυπο πεζών που περπατάνε στη πόλη

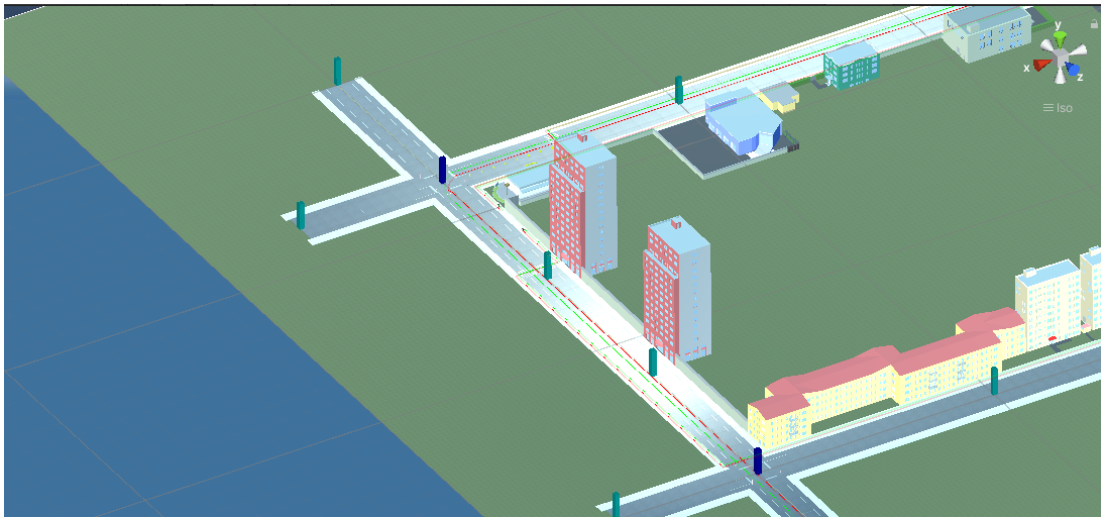
Παρόμοια φιλοσοφία θα ακολουθήσουμε και για τα αυτοκίνητα, μόνο που εκεί το έργο μας γίνεται πιο εύκολο, καθώς αρκεί να εφαρμόσουμε το [CarController](#) που αναπτύξαμε για τη κίνηση του οχήματος του παίκτη και το [CharacterNavigationController](#) που εφαρμόσαμε προηγουμένως. Το πρώτο script θα ευθύνεται για τη φυσιολογικό τρόπο κίνησης του οχήματος εντός της πόλης (στρίψιμο τροχών και ομαλή πορεία) ενώ το δεύτερο script όπως και με τους πεζούς, θα ευθύνεται για τη μετακίνησή του εντός του περιβάλλοντος.



Εικόνα 3.21 Στιγμιότυπο οχημάτων που κυκλοφορούν στη πόλη

Μπορεί τα αυτοκίνητα και οι πεζοί που βρίσκονται στο παιχνίδι μας να μπορούν να κινηθούν αυτόνομα, αλλά αυτό δε σημαίνει ότι έχουν και τη δυνατότητα να κινηθούν έξυπνα. Αυτό σημαίνει ότι στη παρούσα κατάσταση, θα κινούνται εντός του περιβάλλοντος τυχαία,

με ενδεχόμενο να συγκρουστούν σε σπία, να φύγουν εκτός του χάρτη και γενικά να κινηθούν με τρόπο επικίνδυνο και απρόβλεπτο που δεν αρμόζει σε ένα παιχνίδι κυκλοφοριακής αγωγής. Για να αντιμετωπίσουμε αυτό το πρόβλημα, θα υλοποιήσουμε ένα εσωτερικό πρόγραμμα σχεδίασης προκαθορισμένων διαδρομών με χρήση κόμβων. Το πρόγραμμα αυτό θα είναι εύκολα προσβάσιμο μέσα από το Unity ως εργαλείο και θα μας δίνει τη δυνατότητα να τοποθετήσουμε κόμβους εντός του περιβάλλοντος, που με τη σειρά τους θα υποδεικνύουν μια διαδρομή που θα μπορεί να ακολουθήσει το αυτοκίνητο ή ο πεζός.



*Εικόνα 3.22 Με πράσινο και με κόκκινο φαίνονται οι διαδρομές που δημιουργήθηκαν μέσω του Waypoint Navigator*

Χρησιμοποιώντας ένα κενό αντικείμενο ως αφετηρία, μπορούμε να προσθέσουμε κόμβους μπροστά και πίσω από τη τρέχουσα θέση, να αφαιρέσουμε έναν κόμβο ή ακόμα και να δημιουργήσουμε μια διακλάδωση στο μονοπάτι και να ρυθμίσουμε τη πιθανότητα να την ακολουθήσει το αυτοκίνητο ή ο πεζός. Αυτό μας δίνει τη δυνατότητα μιας πιο φυσικής προσομοίωσης κίνησης, αφού υπάρχει το ενδεχόμενο κάθε πεζός και αυτοκίνητο να ακολουθούν διαφορετικές διαδρομές. Ταυτόχρονα έχουμε μεριμνήσει ώστε σε ενδεχόμενο που η διαδρομή που ακολουθείται έχει αρχή και τέλος (κυρίως σε ενδεχόμενα διαδρομών πεζού), ο πεζός να μπορεί να κάνει αναστροφή και να επιστρέψει στο προηγούμενο κόμβο αντί να μείνει ακινητοποιημένος στο τερματισμό. Έτσι, μπορούμε να κρατήσουμε τα αυτοκίνητα στο δρόμο και αποκλειστικά στη λωρίδα τους, ενώ οι πεζοί θα κινούνται αποκλειστικά μέσω των πεζοδρομίων και των διαβάσεων πεζών σε περίπτωση που επιθυμούν να περάσουν απέναντι. Τέλος, για να καλύψουμε το ενδεχόμενο δημιουργίας μεγαλύτερου αριθμού πεζών ή αυτοκινήτων, θα φτιάξουμε δύο C# scripts με τα ονόματα [PedestrianSpawner](#) και [CarSpawner](#) που θα μπορούν να δημιουργήσουν τον ακριβή αριθμό αντιτύπων του ίδιου μοντέλου πεζού ή αυτοκινήτου που θα έχουμε ορίσει εμείς. Και τα δύο scripts θα λειτουργούν με τον ίδιο τρόπο και θα τοποθετούν τα μοντέλα σε τυχαίους κόμβους εντός των διαδρομών που έχουμε καθορίσει εμείς.



## 4. Εκτέλεση Βιντεοπαιχνιδιού & Παρατηρήσεις

### 4.1 Περιγραφή Επιπέδων

Έχοντας ολοκληρώσει το παιχνίδι, ήρθε η ώρα να δούμε με μια πιο αναλυτική ματιά τι προσφέρει από άποψη περιεχομένου. Όπως ήδη έχει αναφερθεί, το παιχνίδι προσφέρει τέσσερα διαφορετικά επίπεδα, τα οποία έχουν αφετηρία σε διαφορετικό σημείο της πόλης, ενώ η ίδια πόλη βρίσκεται σε διαφορετική κατάσταση κάθε φορά.

- **Επίπεδο 1**

Αποστολές:

- Σύλλεξε 10 νομίσματα
- Ακινητοποίησε το όχημά σου πριν το φανάρι

Σκοπός:

- Εκμάθηση του μαθητή σχετικά με το ρόλο των φαναριών και ποια πρέπει να είναι η συμπεριφορά του ανάλογα με τις οδηγίες του φαναριού.
- Εξοικείωση του μαθητή με το χειρισμό του αυτοκινήτου σε ό,τι αφορά την επιτάχυνση και την επιβράδυνση του.

Σε αυτό το εισαγωγικό επίπεδο, το αυτοκίνητο χρειάζεται απλά να κινηθεί σταθερά σε μια ευθεία γραμμή για να φτάσει το προορισμό του. Τα νομίσματα που είναι τοποθετημένα, βοηθούν στη καθοδήγηση του μαθητή μέχρι να έρθει η ώρα να επιβραδύνει ώστε να μπορέσει να ακινητοποιήσει το όχημά του εγκαίρως πριν το φανάρι. Επιτυχής ικανοποίηση αυτών των συνθηκών, θα οδηγήσει στη παροχή ενημερωτικού μηνύματος που θα δώσει στο μαθητή τις απαραίτητες γνώσεις γύρω από τα φανάρια, επεκτείνοντας πέρα από την κατάσταση που συναντήθηκε εντός του παιχνιδιού (σενάριο κόκκινου φαναριού).

- **Επίπεδο 2**

Αποστολές:

- Σύλλεξε 20 νομίσματα
- Ακινητοποίησε το όχημά σου πριν τη πρώτη διάβαση πεζών
- Ακινητοποίησε το όχημά σου πριν τη δεύτερη διάβαση πεζών

Σκοπός:

- Εκμάθηση του μαθητή σχετικά με τις διαβάσεις πεζών και ποιος είναι ο ρόλος τους
- Εξοικείωση του μαθητή με το χειρισμό του αυτοκινήτου σε ό,τι αφορά τη διαχείριση στροφών στο δρόμο



Στο δεύτερο επίπεδο του παιχνιδιού, θέλουμε να επεκτείνουμε στο τι διδάχτηκε ο μαθητής, προσθέτοντας επιπλέον δοκιμασίες. Εδώ, ο μαθητής θα πρέπει να ακινητοποιήσει το όχημά του δύο φορές σε δύο διαφορετικές διαβάσεις πεζών, ώστε να επιτρέψει στους πεζούς που κυκλοφορούν στο πεζοδρόμιο να περάσουν απέναντι. Ταυτόχρονα θα εξοικειωθεί με το χειρισμό του αυτοκινήτου στις στροφές, ενώ για άλλη μια φορά τα νομίσματα που θα υπάρχουν στο δρόμο θα τον καθοδηγούν καθ'όλη τη διάρκεια του επιπέδου. Επιτυχής ικανοποίηση αυτών των αποστολών, θα οδηγήσει στη παροχή ενός ακόμα μηνύματος που θα ενημερώνει το μαθητή αναλυτικότερα για το ρόλο των διαβάσεων πεζών και για το πόσο σημαντικό είναι για τον οδηγό ενός αυτοκινήτου να διατηρείται στη λωρίδα που του αναλογεί.

### ● Επίπεδο 3

#### Αποστολές:

- Σύλλεξε 20 νομίσματα
- Ακινητοποίησε το όχημά σου πριν τη πινακίδα STOP
- Ακινητοποίησε το όχημά σου πίσω από το προπορευόμενο όχημα πριν το φανάρι

#### Σκοπός:

- Εκμάθηση του μαθητή σχετικά με το ρόλο της πινακίδας STOP
- Εκμάθηση του μαθητή για τη γενική συμπεριφορά που οφείλει να έχει γύρω από άλλα οχήματα

Στο τρίτο επίπεδο του παιχνιδιού, έχουμε ένα σχετικά παρόμοιο σενάριο με το δεύτερο επίπεδο, μόνο που αυτή τη φορά η παρουσία αυτοκινήτων έχει αντικαταστήσει αυτή των πεζών. Στο πρώτο στάδιο του επιπέδου, ο μαθητής θα πρέπει να ακινητοποιήσει το όχημά του πριν τη πινακίδα STOP που υπάρχει πριν τη στροφή για να δώσει προτεραιότητα στο όχημα που ετοιμάζεται να στρίψει. Στη συνέχεια θα πρέπει να ακολουθήσει το εν λόγω όχημα χωρίς να το χτυπήσει (ελέγχοντας έτσι την ταχύτητά του) και να το ακινητοποιήσει ξανά πριν το φανάρι. Ικανοποίηση αυτών των συνθηκών θα οδηγήσει στην εμφάνιση ενημερωτικού μηνύματος που θα εξηγήσει στο μαθητή αναλυτικά το ρόλο της πινακίδας STOP και τη σπουδαιότητα της ασφαλούς οδήγησης γύρω από άλλα αυτοκίνητα.

### ● Επίπεδο 4

#### Αποστολές:

- Σύλλεξε 50 νομίσματα
- Ανεφοδιάσου στο πρατήριο καυσίμων
- Βρες το μυστικό πάρκινγκ

#### Σκοπός:

- Αναθεώρηση όσων έχει διδαχθεί ο μαθητής σε ένα πιο ανοιχτό και ελεύθερο περιβάλλον
- Ψυχαγωγήση του μαθητή ως επιβράβευση της προσοχής που επέδειξε στα προηγούμενα επίπεδα

Στο τέταρτο και τελευταίο επίπεδο του παιχνιδιού, ο μαθητής είναι ελεύθερος να εξερευνήσει τη πόλη σε όλο το εύρος της, χρησιμοποιώντας τις γνώσεις που έχει διδαχθεί ώστε να συνεισφέρει στην ομαλή κυκλοφορία του οδικού δικτύου. Η πόλη πλέον είναι γεμάτη αυτοκίνητα και πεζούς, πράγμα που σημαίνει ότι η προσοχή και η κριτική ικανότητα του μαθητή θα δοκιμαστεί στη προσπάθειά του να ολοκληρώσει και τις τρεις αποστολές. Παρόλα αυτά, με αυτό κατά νου οι αποστολές έχουν πιο χαλαρό και διασκεδαστικό τόνο ώστε να δώσουν τη δυνατότητα ψυχαγωγίας στο τέλος της εκπαίδευσης του μαθητή. Όλες οι αποστολές απαιτούν προσεκτική εξερεύνηση της πόλης σε κάθε εκατοστό της. Μόλις ολοκληρωθούν όλες οι αποστολές, ο μαθητής θα λάβει ένα συγχαρητήριο μήνυμα που θα τον ευχαριστεί για τη συμπεριφορά που επέδειξε καθ' όλη τη διάρκεια του παιχνιδιού και θα τον αποχαιρετήσει, ολοκληρώνοντας έτσι την εκπαίδευσή του.

## 4.2 Προβλήματα Που Συναντήθηκαν

Κατά την ανάπτυξη και την εκτέλεση του βιντεοπαιχνιδιού, συναντήθηκαν ορισμένα προβλήματα και εμπόδια που είναι σημαντικό να αναφερθούν και να αναλυθούν. Το πρώτο πρόβλημα, αφορά τους ανιχνευτές σύγκρουσης οι οποίοι συναντιούνται σε όλα τα επίπεδα του παιχνιδιού. Εντός αυτών των ανιχνευτών, ο παίκτης πρέπει να ακινητοποιήσει το όχημά του ώστε να ολοκληρωθούν επιτυχώς οι αποστολές που σχετίζονται με αυτούς. Δυστυχώς κατά την ανάπτυξη του παιχνιδιού δε βρέθηκε αξιόπιστος τρόπος για να ελέγξουμε πότε η ολική ταχύτητα του αυτοκινήτου ισούται με το 0, οπότε χρησιμοποιήθηκε η επιβράδυνση του αυτοκινήτου ως κριτήριο εκπλήρωσης της αποστολής. Με λίγα λόγια, αν ο παίκτης επιταχύνει σταθερά τη στιγμή που περνάει από τον ανιχνευτή, τότε η αποστολή δεν θα ολοκληρωθεί, αλλά αν ο παίκτης χρησιμοποιήσει το φρένο για να επιβραδύνει εντός του πεδίου του ανιχνευτή, τότε η αποστολή θα ολοκληρωθεί ακόμα και αν το όχημα δεν έχει ακινητοποιηθεί τελείως.

Ένα δεύτερο πρόβλημα που συναντήθηκε είναι η συμπεριφορά των αυτοκινήτων σε μαζικούς αριθμούς και παρουσιάζεται αποκλειστικά στο τέταρτο επίπεδο του παιχνιδιού. Όπως έχει αναφερθεί στο αντίστοιχο κεφάλαιο ανάπτυξης του παιχνιδιού, υπάρχει η δυνατότητα να δημιουργήσει έναν μεγάλο αριθμό αυτοκινήτων που το καθένα θα δημιουργηθεί σε ένα τυχαίο σημείο εντός του προκαθορισμένου μονοπατιού με κόμβους που έχουμε ορίσει. Σε περίπτωση όμως που το όχημα δημιουργηθεί κατά τη διάρκεια μιας στροφής του δρόμου, το ίδιο αντί να ακολουθήσει το μονοπάτι, αρχίζει να κάνει κύκλους γύρω από το σημείο που δημιουργήθηκε, παρεμποδίζοντας έτσι την κίνηση των υπολοίπων αυτοκινήτων. Αν ένα αυτοκίνητο που αντιμετωπίζει αυτό το πρόβλημα χτυπηθεί κατάλληλα

από ένα άλλο αυτοματοποιημένο αυτοκίνητο ή ακόμα και το αυτοκίνητο του ίδιου του παίκτη, τότε μπορεί να συνεχίσει κανονικά τη πορεία του.

Τέλος, ένα γενικό πρόβλημα που συναντήθηκε και γινόταν όλο και πιο εμφανές κατά την ανάπτυξη του παιχνιδιού είναι ότι το ίδιο το παιχνίδι δεν είναι ιδιαίτερα φιλικό σε υπολογιστές με χαμηλές επιδόσεις. Αυτό συμβαίνει λόγω του υπερβολικού μεγέθους του χάρτη αλλά και τη παρουσία υπερβολικών αντικειμένων (με τη μεγαλύτερη επιβάρυνση να προκαλείται από τους πεζούς και τα υπόλοιπα αυτοκίνητα εντός του περιβάλλοντος). Ως συνέπεια, αυτό περιόρισε σημαντικά τις ιδέες που υπήρχαν για περαιτέρω ανάπτυξη του παιχνιδιού όπως περισσότερα επίπεδα, πιο πυκνοκατοικημένη πόλη, επιπλέον εφέ και λειτουργίες εντός του παιχνιδιού κτλ.



*Εικόνα 4.1 Παράδειγμα αυτοκινήτου που παρεκτρέπεται από τη πορεία του και κινείται κυκλικά*

## 5. Αξιολόγηση & Προτάσεις Βελτίωσης

### 5.1 Προσωπική Αξιολόγηση

Έχοντας αναλύσει το παιχνίδι που δημιουργήθηκε και έχοντας παρατηρήσει τις περιπτώσεις στις οποίες υστερεί, μπορούμε να δούμε με μια πιο αντικειμενική ματιά αν εν τέλει ικανοποιεί το στόχο που θέσαμε. Σε ό,τι αφορά την εκπαιδευτική φύση του παιχνιδιού, μπορούμε να πούμε ότι ικανοποιεί αυτό το κριτήριο, αφού έχουμε φροντίσει ώστε ο μαθητής να μπορεί να διδαχθεί παίζοντας το παιχνίδι αλλά και διαβάζοντας τα ενημερωτικά μηνύματα που έχουν προετοιμαστεί. Τα ενημερωτικά μηνύματα παραπέμπουν στον παραδοσιακό τρόπο διδασκαλίας, αλλά ενσωματωμένα με τέτοιο τρόπο ώστε να μοιάζουν με επιβράβευση παρά σαν μια διαδικασία από την οποία πρέπει ο μαθητής να περάσει αναγκαστικά.

Στη συνέχεια, είναι σημαντικό να κοιτάξουμε αν και κατά πόσο το παιχνίδι μας ανήκει στη κατηγορία των σοβαρών παιχνιδιών, χωρίς να υπάρχει κίνδυνος να παρερμηνευτεί ως ένα παιχνίδι που σχεδιάστηκε για καθαρή ψυχαγωγία. Από την αρχή της ανάπτυξής του, στοχεύσαμε ώστε το παιχνίδι να ισορροπεί αρμονικά ανάμεσα στα σοβαρά και τα ψυχαγωγικά χαρακτηριστικά του. Αυτό το πετύχαμε επικεντρώνοντας τη προσοχή του μαθητή αποκλειστικά στην εκπαίδευση στα τρία πρώτα επίπεδα και δίνοντάς του τη δυνατότητα να εφαρμόσει τις γνώσεις του σε ένα πιο χαλαρό και διασκεδαστικό τέταρτο επίπεδο. Επιπλέον, η παρουσία του εκπαιδευτή εμπνέει μεγαλύτερη οικειότητα και φιλικότητα από το πρόσωπο ενός εκπαιδευτή κυκλοφοριακής ή ενός δασκάλου στο μαθητή και συνεπώς τα εκπαιδευτικά μηνύματα που διατυπώνονται με σοβαρό τόνο, μεταδίδονται πιο αποτελεσματικά.

Τέλος, όσον αφορά το χειρισμό του παιχνιδιού, την ευκολία πλοήγησης εντός του αλλά και τη γενική λειτουργία του, έχουμε μια πιο μικτή εικόνα. Ο χειρισμός του αυτοκινήτου, είναι απλός και εύκολα κατανοήσιμος από μαθητές μικρής ηλικίας, αλλά το αυτοκίνητο σε μεγάλες ταχύτητες μπορεί να συμπεριφερθεί πιο απρόβλεπτα, ειδικά στις στροφές. Τα μενού είναι πολύ απλά και λιτά σχεδιασμένα και συνεπώς αποφεύγεται κάθε είδους δυσκολία ως προς το που πρέπει να κινηθεί ο μαθητής για να αρχίσει να παίζει, ενώ οι οδηγίες που του δίνονται για κάθε επίπεδο και κάθε αποστολή είναι ξεκάθαρες. Η γενική λειτουργία του παιχνιδιού ενώ είναι ικανοποιητική, δεν παύει να παρουσιάζει ορισμένα σημαντικά προβλήματα που αναφέρθηκαν στη τέταρτη ενότητα. Ενώ τα τρία πρώτα επίπεδα μπορούμε να πούμε ότι ικανοποιούν τις προσδοκίες μας, τα προβλήματα του τέταρτου σίγουρα δε βοηθούν τη γενική εικόνα. Βέβαια είναι σημαντικό να αναφερθεί ότι πέρα από το εκπαιδευτικό περιεχόμενο του παιχνιδιού, προσπαθήσαμε να αναδείξουμε τις δυνατότητες που μπορεί να έχει η δημιουργία ενός τέτοιου παιχνιδιού και εφόσον είναι σε πρωτότυπο στάδιο, πάντα υπάρχει η πιθανότητα εμφάνισης σφαλμάτων.

## 5.2 Αξιολόγηση Από Τρίτους

Για να μπορέσουμε να καταλάβουμε καλύτερα κατά πόσο το παιχνίδι μας εκπληρώνει το σκοπό του, διαλέξαμε δύο άτομα, ένα αγόρι και ένα κορίτσι που είναι μαθητές δημοτικού. Και τα δύο άτομα ενημερώθηκαν για το σκοπό του παιχνιδιού αλλά δεν τους δόθηκαν οδηγίες για το χειρισμό και το τρόπο πλοήγησής του με σκοπό να δούμε αν τελικά είναι αρκετά φιλικό ακόμα και σε άπειρους χρήστες. Καθώς το παιχνίδι είναι φτιαγμένο στην αγγλική γλώσσα, τους δόθηκε κείμενο με ακριβείς μεταφράσεις στα ελληνικά.

- **Αγόρι (Ετών 11)**

Το αγόρι εξέφρασε γενική ικανοποίηση για το παιχνίδι. Βρήκε εύκολη τη πλοήγηση στα μενού και εύκολα κατανοητές τις οδηγίες που του δόθηκαν ενώ χρειάστηκε λίγη βοήθεια για να ολοκληρώσει τα πρώτα τρία επίπεδα. Παρατήρησε τα προβλήματα του τέταρτου επιπέδου αλλά δήλωσε ότι γενικά δεν επηρέασαν την εμπειρία του με το παιχνίδι. Εξέφρασε ενθουσιασμό για τον χαρακτήρα του εκπαιδευτή αλλά θα ήθελε μεγαλύτερη παρουσία σημάτων εντός της πόλης. Δεν δυσκολεύτηκε με το χειρισμό του αυτοκινήτου, αλλά εξέφρασε παράπονο για την απουσία ήχου από το αυτοκίνητο την απουσία ένδειξης της τρέχουσας ταχύτητας στην οθόνη. Θεωρεί ότι είναι ένα παιχνίδι που θα μπορούσε να χρησιμοποιηθεί για την εκμάθηση της κυκλοφοριακής αγωγής αν επεκταθεί στο βαθμό που πρέπει.

- **Κορίτσι (Ετών 12)**

Το κορίτσι εξέφρασε γενική ικανοποίηση για το παιχνίδι. Βρήκε εύκολη τη πλοήγηση στα μενού ενώ χρειάστηκε μικρή βοήθεια για την κατανόηση των οδηγιών που της δόθηκαν. Δε χρειάστηκε βοήθεια για να ολοκληρώσει τα τέσσερα επίπεδα του παιχνιδιού, καθώς στη κατά τη διάρκεια επαφής της με το παιχνίδι δεν παρουσιάστηκαν προβλήματα ούτε στο τέταρτο επίπεδο. Δυσκολεύτηκε ελαφρώς με το χειρισμό του αυτοκινήτου, ειδικά όταν έβγαινε εκτός τροχιάς και έπρεπε να επιστρέψει σε κανονική πορεία για να συλλέξει τα υπόλοιπα νομίσματα. Εξέφρασε παράπονο για την απουσία μεγαλύτερης λεπτομέρειας στη πόλη σε ότι αφορά τα κτήρια, αλλά εντυπωσιάστηκε γενικά από τη συνολική εικόνα του παιχνιδιού. Θεωρεί ότι είναι ένα παιχνίδι που θα μπορούσε να χρησιμοποιηθεί για την εκμάθηση της κυκλοφοριακής αγωγής, αναγνωρίζοντας ότι η κατάσταση του παιχνιδιού είναι σχετικά πρώιμη.

### 5.3 Προτάσεις Βελτίωσης

Το αποτέλεσμα του παιχνιδιού να είναι σχετικά ικανοποιητικό, αλλά μπορεί να δεχτεί σημαντικές βελτιώσεις σε μελλοντικό χρόνο για να γίνει ποιοτικότερο και αποδοτικότερο. Κατά πρώτον, είναι σημαντικό να βελτιωθεί η γενική εμφάνιση της πόλης. Όπως είδαμε η πόλη, ενώ μεγάλη σε έκταση, δεν ήταν αρκετά πυκνή σε ότι αφορά τα κτίρια και τα αντικείμενα που υπήρχαν στο χώρο, λόγω του ότι επιβάρυναν σημαντικά τις απαιτήσεις του παιχνιδιού. Με τη χρήση πιο οικονομικών asset, είναι δυνατόν να δημιουργηθεί μια πιο ζωντανή πόλη, ακόμα και αν χρειαστεί το μέγεθος της μεταβληθεί ελάχιστα προς τα κάτω. Κατά δεύτερον, η συμπεριφορά των πεζών και των αυτοκινήτων χρήζει βελτίωσης, ώστε να αποφευχθούν τα προβλήματα που συναντήθηκαν, αλλά και να μπορούν να κινούνται εντός της πόλης πιο έξυπνα (π.χ οι πεζοί να σταματούν πριν περάσουν τη διάβαση αν νιώθουν ότι ένα αυτοκίνητο πλησιάζει, ένα αυτοκίνητο να φρενάρει απότομα αν ένας πεζός βρίσκεται στο δρόμο, να ακολουθούν μεγαλύτερο αριθμό μοναδικών διαδρομών κτλ). Επιπλέον, είναι σημαντικό το παιχνίδι να προσφέρει περισσότερα επίπεδα τα οποία θα επικεντρώνονται περισσότερο στα σήματα οδικής κυκλοφορίας. Στο παιχνίδι μας εξετάσαμε μόνο τη περίπτωση του STOP, οπότε με εμπλούτισμο της πόλης με περισσότερες πινακίδες θα μπορούμε να εμβαθύνουμε περισσότερο στο εκπαιδευτικό κομμάτι του παιχνιδιού. Επίσης, το αυτοκίνητο του παίκτη χρήζει βελτίωσης, καθώς απουσιάζει ο ήχος του κινητήρα και σημαντικές ενδείξεις όπως η τρέχουσα ταχύτητα και ο χειρισμός του αυτοκινήτου μπορεί να γίνει πιο ρεαλιστικός (κλείδωμα τροχών σε απότομο φρενάρισμα, προσομοίωση οδήγησης σε βρεγμένο οδόστρωμα). Τέλος, το παιχνίδι θα μπορούσε να εμπλουτιστεί με πρόσθετες λειτουργίες, όπως μια σειρά επιπέδων που θα βάζουν τον παίκτη στο ρόλο του πεζού ή μια λειτουργία ελεύθερης εξερεύνησης (free roaming) που θα μπορούσε να ξεκλειδωθεί μόλις ο παίκτης ολοκληρώσει την εκπαίδευσή του.

# Βιβλιογραφία

[GK15] Green, G., & Kaufman, J. C. (2015). *Video Games and Creativity*. Elsevier Gezondheidszorg. <https://doi.org/10.1016/C2014-0-00065-2>

[AK09] Asgari, M., & Kaufman, D. (2009). Motivation, Learning, and Game Design. *Handbook of Research on Effective Electronic Gaming in Education*, 1166–1182. <https://doi.org/10.4018/978-1-59904-808-6.ch067>

[A70] Abt, C. C. (1970). *Serious Games* (1st ed.). The Viking Press.

[C07] Corti, K. (2007, October 16). *Opinion: “Serious Games - Are We Really A Community?”* Game Developer. <https://www.gamedeveloper.com/pc/opinion-serious-games---are-we-really-a-community->

[DAJ11] Djaouti, D., Alvarez, J., & Jessel, J. (2011, January 1). *Classifying Serious Games: the G/P/S model*. ResearchGate. <https://dx.doi.org/10.4018/978-1-60960-495-0.ch006>

[DD18] *All About Serious Games: Types and Purposes* | DDINC. (2018, April 16). Designing Digitally. <https://www.designingdigitally.com/blog/2018/04/all-about-serious-games-types-and-purposes>

[SWOV17] SWOV (2017). Traffic education. SWOV Fact sheet, December 2017. SWOV, The Hague. <https://www.swov.nl/en/facts-figures/factsheet/traffic-education>

[WHO18] WHO | *Death on the roads*. (2018). World Health Organization. [https://extranet.who.int/roadsafety/death-on-the-roads/#ticker/all\\_road\\_users](https://extranet.who.int/roadsafety/death-on-the-roads/#ticker/all_road_users)  
<http://apps.who.int/iris/bitstream/handle/10665/277370/WHO-NMH-NVI-18.20-eng.pdf?ua=1>

[BHM06] Backlund, Per & Engström, Henrik & Johannesson, Mikael. (2006). Computer gaming and driving education. [https://www.researchgate.net/publication/228987973\\_Computer\\_gaming\\_and\\_driving\\_education](https://www.researchgate.net/publication/228987973_Computer_gaming_and_driving_education)

[G16] Gamescrye. (2016, October 3). *What is a Game Engine?* <https://gamescrye.com/blog/what-is-a-game-engine/>

[E21] Eck, D. J. (2021). *Introduction to Computer Graphics* (1.3 ed.).  
<https://math.hws.edu/graphicsbook/>

[HTWG10] Hejlsberg, A., Torgersen, M., Wiltamuth, S., & Golde, P. (2010). *C# Programming Language (Covering C# 4.0), The (Microsoft Windows Development Series)* (4th ed.). Addison-Wesley Professional.



# Παράρτημα Κώδικα

## CameraFollow.cs

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class CarSpawner : MonoBehaviour
{
    public GameObject carPrefab;
    public int carsToSpawn;
    // Start is called before the first frame update
    void Start()
    {
        StartCoroutine(Spawn());
    }

    IEnumerator Spawn()
    {
        int count = 0;
        while (count < carsToSpawn)
        {
            GameObject obj = Instantiate(carPrefab);
            Transform child = transform.GetChild(Random.Range(0, transform.childCount -
1));
            obj.GetComponent<WaypointNavigator2>().currentWaypoint =
child.GetComponent<Waypoint>();
            obj.transform.position = child.position;

            yield return new WaitForEndOfFrame();

            count++;
        }
    }
}
```

## CarController.cs

```
using System;
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class CarController : MonoBehaviour
{
    public const string HORIZONTAL = "Horizontal";
    public const string VERTICAL = "Vertical";

    public float horizontalInput;
    public float verticalInput;
    public float currentSteerAngle;
    public float currentbrakeForce;
    public bool isBraking;

    [SerializeField] public float motorForce;
    [SerializeField] public float brakeForce;
    [SerializeField] public float maxSteerAngle;

    [SerializeField] public WheelCollider frontLeftWheelCollider;
    [SerializeField] public WheelCollider frontRightWheelCollider;
    [SerializeField] public WheelCollider rearLeftWheelCollider;
    [SerializeField] public WheelCollider rearRightWheelCollider;

    [SerializeField] public Transform frontLeftWheelTransform;
    [SerializeField] public Transform frontRightWheeTransform;
    [SerializeField] public Transform rearLeftWheelTransform;
    [SerializeField] public Transform rearRightWheelTransform;

    public void FixedUpdate()
    {
        GetInput();
        HandleMotor();
        HandleSteering();
        UpdateWheels();
    }

    public void GetInput()
    {
        horizontalInput = Input.GetAxis(HORIZONTAL);
        verticalInput = Input.GetAxis(VERTICAL);
        isBraking = Input.GetKey(KeyCode.Space);
    }
}
```

```

public void HandleMotor()
{
    frontLeftWheelCollider.motorTorque = verticalInput * motorForce;
    frontRightWheelCollider.motorTorque = verticalInput * motorForce;
    currentbrakeForce = isBraking ? brakeForce : 0f;
    ApplyBraking();
}

public void ApplyBraking()
{
    frontRightWheelCollider.brakeTorque = currentbrakeForce;
    frontLeftWheelCollider.brakeTorque = currentbrakeForce;
    rearLeftWheelCollider.brakeTorque = currentbrakeForce;
    rearRightWheelCollider.brakeTorque = currentbrakeForce;
}

public void HandleSteering()
{
    currentSteerAngle = maxSteerAngle * horizontalInput;
    frontLeftWheelCollider.steerAngle = currentSteerAngle;
    frontRightWheelCollider.steerAngle = currentSteerAngle;
}

public void UpdateWheels()
{
    UpdateSingleWheel(frontLeftWheelCollider, frontLeftWheelTransform);
    UpdateSingleWheel(frontRightWheelCollider, frontRightWheelTransform);
    UpdateSingleWheel(rearRightWheelCollider, rearRightWheelTransform);
    UpdateSingleWheel(rearLeftWheelCollider, rearLeftWheelTransform);
}

public void UpdateSingleWheel(WheelCollider wheelCollider, Transform
wheelTransform)
{
    Vector3 pos;
    Quaternion rot;
    wheelCollider.GetWorldPose(out pos, out rot);
    wheelTransform.rotation = rot;
    wheelTransform.position = pos;
}
}

```

## SettingsMenu.cs

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.Audio;
using UnityEngine.UI;

public class SettingsMenu : MonoBehaviour
{
    public AudioManager audioMixer;

    public TMPro.TMP_Dropdown resolutionDropdown;

    Resolution[] resolutions;

    void Start()
    {
        resolutions = Screen.resolutions;

        resolutionDropdown.ClearOptions();

        List<string> options = new List<string>();

        int currentResolutionIndex = 0;
        for (int i = 0; i < resolutions.Length; i++)
        {
            string option = resolutions[i].width + " x " + resolutions[i].height;
            options.Add(option);

            if (resolutions[i].width == Screen.currentResolution.width && resolutions[i].height
                == Screen.currentResolution.height)
            {
                currentResolutionIndex = i;
            }
        }

        resolutionDropdown.AddOptions(options);
        resolutionDropdown.value = currentResolutionIndex;
        resolutionDropdown.RefreshShownValue();
    }

    public void SetResolution (int resolutionIndex)
    {
        Resolution resolution = resolutions[resolutionIndex];
        Screen.SetResolution(resolution.width, resolution.height, Screen.fullScreen);
    }
}
```

```
public void SetVolume (float volume)
{
    audioMixer.SetFloat("volume", volume);
}

public void SetQuality (int qualityIndex)
{
    QualitySettings.SetQualityLevel(qualityIndex);
}

public void SetFullscreen (bool isFullscreen)
{
    Screen.fullScreen = isFullscreen;
}
}
```

## IntroPanel.cs

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;

public class IntroPanel : MonoBehaviour
{
    public GameObject introPanelUI;
    public AudioSource music;

    public void StartGame()
    {
        SceneManager.LoadScene("Game");
    }

    public void StartGame2()
    {
        SceneManager.LoadScene("Game2");
    }

    public void StartGame3()
    {
        SceneManager.LoadScene("Game3");
    }

    public void StartGame4()
    {
        SceneManager.LoadScene("Game4");
    }
}
```

## ComboTest.cs

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;

public class ComboTest : MonoBehaviour
{

    private int score;
    public static bool GameFinished = false;
    public GameObject successMenuUI;
    public AudioSource music;

    // Start is called before the first frame update
    void Start()
    {
        successMenuUI.SetActive(false);
        CoinHandler.flag = 0;
        TriggerTest2.flag = 0;
        Time.timeScale = 1f;
        GameFinished = false;
    }

    // Update is called once per frame
    void Update()
    {
        score = CoinHandler.flag + TriggerTest2.flag;

        if (score == 2)
        {
            music.Stop();
            successMenuUI.SetActive(true);
            Time.timeScale = 0f;
            GameFinished = true;
        }
    }
}
```

## TriggerTest.cs

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using TMPro;

public class TriggerTest : MonoBehaviour
{

    private Color m_oldColor = Color.red;
    private Color m_textColor = new Color(1, 1, 1, 1f);
    public static int flag = 0;

    // Start is called before the first frame update
    void Start()
    {
        Renderer render = GetComponent<Renderer>();
        render.material.color = m_oldColor;
        TMP_Text missionText =
GameObject.Find("missionText3").GetComponent<TMP_Text>();
    }

    // Update is called once per frame
    void Update()
    {

    }

    void OnTriggerStay(Collider other)
    {
        if (GameManager.currVel == 0 && other.name == "Car")
        {
            Renderer render = GetComponent<Renderer>();
            render.material.color = Color.green;
            TMP_Text missionText =
GameObject.Find("missionText3").GetComponent<TMP_Text>();
            missionText.color = new Color(0, 1, 0, 1f);
            flag = 1;
        }
    }
}
```



## CollectionScript.cs

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class CollectionScript : MonoBehaviour
{
    public CoinHandler CH;
    public AudioClip coinSound;

    // Start is called before the first frame update
    void Start()
    {
        CH = GameObject.Find("Canvas").GetComponent<CoinHandler>();
    }

    // Update is called once per frame
    void Update()
    {
    }

    void OnTriggerEnter(Collider other)
    {
        if (other.name == "Car")
        {
            CH.coins++;
            AudioSource.PlayClipAtPoint(coinSound, transform.position);
            Destroy(gameObject);
        }
    }
}
```

## CoinHandler.cs

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using TMPro;

public class CoinHandler : MonoBehaviour
{

    public TMP_Text CoinText;
    public int coins;
    public static int flag = 0;

    // Update is called once per frame
    void Update()
    {
        CoinText.text = "Collect all coins: " + coins + "/10";

        if (coins == 10)
        {
            CoinText.color = new Color(0, 1, 0, 1f);
            flag = 1;
        }
    }
}
```

## CharacterNavigationController.cs

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class CharacterNavigationController : MonoBehaviour
{
    public Vector3 destination;
    public bool reachedDestination;
    public float stopDistance = 1;
    public float rotationSpeed = 120;
    public float minSpeed, maxSpeed;
    public float movementSpeed = 1;
    public Animator animator;

    private Vector3 lastPosition;
    Vector3 velocity;

    private void Awake()
    {
        animator = GetComponent<Animator>();
    }

    private void Start()
    {
        movementSpeed = Random.Range(minSpeed, maxSpeed);
    }
    private void Update()
    {
        if (transform.position != destination)
        {
            Vector3 destinationDirection = destination - transform.position;
            destinationDirection.y = 0;

            float destinationDistance = destinationDirection.magnitude;

            if (destinationDistance >= stopDistance)
            {
                reachedDestination = false;
                Quaternion targetRotation = Quaternion.LookRotation(destinationDirection);
                transform.rotation = Quaternion.RotateTowards(transform.rotation,
targetRotation, rotationSpeed * Time.deltaTime);
                transform.Translate(Vector3.forward * movementSpeed * Time.deltaTime);
            }
            else
            {

```

```
        reachedDestination = true;
    }

    velocity = (transform.position - lastPosition) / Time.deltaTime;
    velocity.y = 0;
    var velocityMagnitude = velocity.magnitude;
    velocity = velocity.normalized;
    var fwdDotProduct = Vector3.Dot(transform.forward, velocity);
    var rightDotProduct = Vector3.Dot(transform.right, velocity);

    animator.SetFloat("Horizontal", rightDotProduct);
    animator.SetFloat("Forward", fwdDotProduct);
}

lastPosition = transform.position;
}

public void SetDestination(Vector3 destination)
{
    this.destination = destination;
    reachedDestination = false;
}
}
```

## PedestrianSpawner.cs

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class PedestrianSpawner : MonoBehaviour
{
    public GameObject pedestrianPrefab;
    public int pedestriansToSpawn;
    // Start is called before the first frame update
    void Start()
    {
        StartCoroutine(Spawn());
    }

    IEnumerator Spawn()
    {
        int count = 0;
        while (count < pedestriansToSpawn)
        {
            GameObject obj = Instantiate(pedestrianPrefab);
            Transform child = transform.GetChild(Random.Range(0, transform.childCount -
1));
            obj.GetComponent<WaypointNavigator>().currentWaypoint =
child.GetComponent<Waypoint>();
            obj.transform.position = child.position;

            yield return new WaitForEndOfFrame();

            count++;
        }
    }
}
```

## CarSpawner.cs

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class CarSpawner : MonoBehaviour
{
    public GameObject carPrefab;
    public int carsToSpawn;
    // Start is called before the first frame update
    void Start()
    {
        StartCoroutine(Spawn());
    }

    IEnumerator Spawn()
    {
        int count = 0;
        while (count < carsToSpawn)
        {
            GameObject obj = Instantiate(carPrefab);
            Transform child = transform.GetChild(Random.Range(0, transform.childCount -
1));
            obj.GetComponent<WaypointNavigator2>().currentWaypoint =
child.GetComponent<Waypoint>();
            obj.transform.position = child.position;

            yield return new WaitForEndOfFrame();

            count++;
        }
    }
}
```