



ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

**ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΥΛΟΠΟΙΗΣΗ ΠΑΙΧΝΙΔΙΟΥ ΔΥΟ
ΔΙΑΣΤΑΣΕΩΝ ΓΙΑ ΥΠΟΛΟΓΙΣΤΗ, ΤΥΠΟΥ ARCADE,
ΜΕ ΧΡΗΣΗ HTML & JAVASCRIPT.**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΤΟΥ

Χαράλαμπου Ελευθεριάδη

Βόλος, 14/12/ 2013

Η σελίδα αυτή είναι σκόπιμα λευκή.



ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ
ΥΠΟΛΟΓΙΣΤΩΝ

ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΥΛΟΠΟΙΗΣΗ ΠΑΙΧΝΙΔΙΟΥ ΔΥΟ ΔΙΑΣΤΑΣΕΩΝ ΓΙΑ ΥΠΟΛΟΓΙΣΤΗ, ΤΥΠΟΥ ARCADE, ΜΕ ΧΡΗΣΗ HTML & JAVASCRIPT.

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΤΟΥ

Χαράλαμπου Ελευθεριάδη

Επιβλέποντες :

ΤΣΟΜΠΑΝΟΠΟΥΛΟΥ ΠΑΝΑΓΙΩΤΑ	ΜΠΟΖΑΝΗΣ ΠΑΝΑΓΙΩΤΗΣ
ΕΠΙΚΟΥΡΟΣ ΚΑΘΗΓΗΤΡΙΑ	ΑΝΑΠΛΗΡΩΤΗΣ ΚΑΘΗΓΗΤΗΣ

Εγκρίθηκε από την διμελή εξεταστική επιτροπή την ??????????????

(Υπογραφή)

.....

ΤΣΟΜΠΑΝΟΠΟΥΛΟΥ ΠΑΝΑΓΙΩΤΑ,
ΕΠΙΚΟΥΡΟΣ ΚΑΘΗΓΗΤΡΙΑ
ΠΑΝΕΠΙΣΤΗΜΙΟΥ ΘΕΣΣΑΛΙΑΣ

(Υπογραφή)

.....

ΜΠΟΖΑΝΗΣ ΠΑΝΑΓΙΩΤΗΣ,
ΑΝΑΠΛΗΡΩΤΗΣ ΚΑΘΗΓΗΤΗΣ
ΠΑΝΑΠΙΣΤΗΜΙΟΥ ΘΕΣΣΑΛΙΑΣ

(Υπογραφή)

.....

ΧΑΡΑΛΑΜΠΟΣ ΕΛΕΥΘΕΡΙΑΔΗΣ

Διπλωματούχος Μηχανικός Ηλεκτρονικών Υπολογιστών, Τηλεπικοινωνιών και
Δικτύων του Τμήματος Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών,
Πανεπιστημίου Θεσσαλίας

© 2014 – All rights reserved

*Αφιερώνω
αυτή τη διπλωματική εργασία,
στην οικογένεια μου.*

Πίνακας περιεχομένων

1	ΕΙΣΑΓΩΓΗ	Σφάλμα! Δεν έχει οριστεί σελιδοδείκτης.1
1.1	ΠΕΡΙΓΡΑΦΗ.....	Σφάλμα! Δεν έχει οριστεί σελιδοδείκτης.1
1.2	ΑΝΤΙΚΕΙΜΕΝΟ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ.....	Σφάλμα! Δεν έχει οριστεί σελιδοδείκτης.2
1.3	ΤΡΟΠΟΣ ΔΟΜΗΣΗΣ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ ...	Σφάλμα! Δεν έχει οριστεί σελιδοδείκτης.2
2	ΤΑ ΠΑΙΧΝΙΔΙΑ ΣΤΟΥΣ ΗΛΕΚΤΡΟΝΙΚΟΥΣ ΥΠΟΛΟΓΙΣΤΕΣ....	Σφάλμα! Δεν έχει οριστεί σελιδοδείκτης.3
2.1	Ο ΡΟΛΟΣ ΤΟΥ ΠΑΙΧΝΙΔΙΟΥ ΣΤΟΥΣ Η/Υ ΓΙΑ ΤΟΝ ΧΡΗΣΤΗ	Σφάλμα! Δεν έχει οριστεί σελιδοδείκτης.3
2.2	ΕΙΔΗ ΗΛΕΚΤΡΟΝΙΚΩΝ ΠΑΙΧΝΙΔΙΩΝ ΠΟΥ ΥΠΑΡΧΟΥΝ	4
2.3	Η ΚΑΤΗΓΟΡΙΑ ΠΑΙΧΝΙΔΙΩΝ ΤΥΠΟΥ ARCADE.....	6
3	ΓΛΩΣΣΕΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ ΓΙΑ ΤΗΝ ΔΗΜΙΟΥΡΓΙΑ ΠΑΙΧΝΙΔΙΩΝ ..	10
3.1	ΠΛΕΟΝΕΚΤΗΜΑΤΑ & ΜΕΙΟΝΕΚΤΗΜΑΤΑ ΓΛΩΣΣΩΝ.....	11
3.2	ΠΕΡΙΓΡΑΦΗ ΤΩΝ ΓΛΩΣΣΩΝ HTML & JAVASCRIPT	13
4	ΣΧΕΔΙΑΣΜΟΣ ΠΑΙΧΝΙΔΙΟΥ	15
4.1	ΠΕΡΙΓΡΑΦΗ ΣΕΝΑΡΙΟΥ ΠΑΙΧΝΙΔΙΟΥ	15

4.2	ΤΡΟΠΟΣ ΧΡΗΣΗΣ ΚΑΙ ΑΠΑΙΤΗΣΕΙΣ ΠΑΙΧΝΙΔΙΟΥ	19
4.3	ΔΟΜΙΚΑ ΣΤΟΙΧΕΙΑ ΠΑΙΧΝΙΔΙΟΥ	20
4.4	ΜΕΘΟΔΟΣ ΔΟΜΗΣΗΣ ΠΑΙΧΝΙΔΙΟΥ	27
5	ΥΛΟΠΟΙΗΣΗ ΠΑΙΧΝΙΔΙΟΥ	29
5.1	ΠΕΡΙΓΡΑΦΗ ΤΟΥ HTML ΑΡΧΕΙΟΥ (Ace of Spades.html).....	29
5.2	ΠΕΡΙΓΡΑΦΗ ΤΟΥ JAVASCRIPT ΑΡΧΕΙΟΥ (aos.js).....	32
5.3	ΟΡΙΣΜΟΣ ΜΕΤΑΒΛΗΤΩΝ ΚΑΙ ΔΗΜΙΟΥΡΓΙΑ ΑΝΤΙΚΕΙΜΕΝΩΝ.....	32
5.4	ΥΛΟΠΟΙΗΣΗ ΑΝΤΙΚΕΙΜΕΝΩΝ ΚΑΙ ΣΥΝΑΡΤΗΣΕΩΝ	34
6	ΕΠΙΛΟΓΟΣ	46
	Βιβλιογραφία.....	47
	ΠΑΡΑΡΤΗΜΑ Α	50

1

ΕΙΣΑΓΩΓΗ

1.1 ΠΕΡΙΓΡΑΦΗ

Σκοπός της διπλωματικής εργασίας είναι η μεθοδολογία, η οργάνωση και ο τρόπος υλοποίησης ενός παιχνιδιού για υπολογιστή, με τη χρήση μιας συγκεκριμένης γλώσσας προγραμματισμού.

Στον τομέα των *video games* και ειδικότερα των *game applications*, τα μέσα και οι μέθοδοι που έχουν αναπτυχθεί, δίνουν μία μεγάλη ποικιλία από εργαλεία και επιλογές, ούτως ώστε να δημιουργηθεί ένα καινούργιο παιχνίδι σε οποιοσδήποτε από τις αντιληπτές διαστάσεις (2-D / 3-D). Η γλώσσα προγραμματισμού υλοποίησης και τα βοηθητικά εργαλεία/προγράμματα επεξεργασίας εικόνας και ήχου, αποτελούν προσωπική επιλογή για τον εκάστοτε *game developer*, ανάλογα με τις ικανότητές του, την εξοικείωση του με μια συγκεκριμένη γλώσσα προγραμματισμού και τον τρόπο που θέλει αυτός να δημιουργήσει το παιχνίδι που έχει σχεδιάσει.

Στα πλαίσια αυτής της εργασίας θα ακολουθηθεί, θα υλοποιηθεί και θα εξετασθεί μία τέτοια μέθοδος σχεδίασης και υλοποίησης παιχνιδιού για PC.

1.2 ΑΝΤΙΚΕΙΜΕΝΟ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ

Στα πλαίσια αυτής της εργασίας, θα γίνει η σχεδίαση, η ανάλυση και η υλοποίηση ενός παιχνιδιού δύο διαστάσεων (2-D), για PC (*stand alone*). Το παιχνίδι θα είναι τύπου *Arcade* και θα προορίζεται για έναν παίχτη (*single player game*) χωρίς την υποστήριξη του διαδικτύου (*offline game*). Η υλοποίηση θα γίνει με τη χρήση JavaScript και HTML, καθώς και με προγράμματα διαμόρφωσης εικόνας/γραφικών (Paint.NET, Photo Shop) και ήχου (Virtual DJ), για τα οπτικοακουστικά μέρη. Το παιχνίδι θα “τρέχει” και θα είναι συμβατό με όλους τους διαδεδομένους περιηγητές (*web browsers*): Google Chrome, Mozilla Firefox, Internet Explorer, Opera, Safari.

Το παιχνίδι θα αποτελείται από ορισμένα έτοιμα αντικείμενα εικόνας και ήχου, που σκοπό έχουν την, όσο το δυνατόν, πιο ρεαλιστική προσέγγιση για τον παίχτη, καθώς και στοιχεία που αποτελούν προσωπικές δημιουργίες του προγραμματιστή (*backgrounds*, *menu screens*). Όλα αυτά θα εναρμονιστούν με τη χρήση κατάλληλων συναρτήσεων και αλγορίθμων για το βέλτιστο, επιθυμητό αποτέλεσμα.

1.3 ΤΡΟΠΟΣ ΔΟΜΗΣΗΣ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ

Η ακολουθία των βημάτων δόμησης της διπλωματικής εργασίας είναι η εξής:

- I. «Σύλληψη» γενικής ιδέας και στυλ παιχνιδιού
(ταχύτητα/ ικανότητα/ γνώσεις/ αντανakλαστικά/ παρατηρητικότητα & 2D/3D).
- II. Απόφαση για το είδος του παιχνιδιού
(*arcade/action/ story line/ role playing game/ first person/ non-hero game/ quiz*).
- III. Απόφαση για τη συμβατότητα του παιχνιδιού και τον αριθμό παιχτών που θα υποστηρίζει.
- IV. Δημιουργία σεναρίου παιχνιδιού και σχεδιασμός κλίμακας δυσκολίας.
- V. Δημιουργία/συλλογή αντικειμένων εικόνας και ήχου, με τα κατάλληλα προγράμματα, που θα συνθέτουν το παιχνίδι.
- VI. Επιλογή τρόπου αναπαράστασης γραφικών και ηχητικών στοιχείων.
- VII. Επιλογή επιθυμητής γλώσσας προγραμματισμού για την υλοποίηση.
- VIII. Σχεδιασμός και προγραμματισμός αλγορίθμων και συναρτήσεων.
- IX. Εντατικός έλεγχος και διόρθωση σφαλμάτων για όλα τα πιθανά σενάρια ενεργειών.

2

ΤΑ ΠΑΙΧΝΙΔΙΑ ΣΤΟΥΣ

Η/Υ

Στο κεφάλαιο αυτό θα γίνει μια περιγραφή του ρόλου και της σημασίας των παιχνιδιών στους ηλεκτρονικούς υπολογιστές. Θα γίνει αναφορά στα διάφορα είδη παιχνιδιών που υπάρχουν και μια εμβάθυνση στο είδος *arcade*, το οποίο υλοποιείται στη συγκεκριμένη διπλωματική εργασία.

2.1 Ο ΡΟΛΟΣ ΤΩΝ ΠΑΙΧΝΙΔΙΩΝ ΣΤΟΥΣ Η/Υ ΓΙΑ ΤΟΝ ΧΡΗΣΤΗ

Το παιχνίδι σαν έννοια, αποτελούσε και αποτελεί από αρχαιοτάτων χρόνων, την ιδανικότερη μορφή ψυχαγωγίας για τον άνθρωπο σε όλες τις στιγμές της ζωής του. Σε ορισμένες ηλικίες μάλιστα, είναι αναγκαία διαδικασία για τη σωστή ανάπτυξη του ανθρώπου, τόσο σε νοητικό όσο και σε ψυχικό επίπεδο. Πάνω στο παιχνίδι έχει στηριχτεί ολόκληρη βιομηχανία εμπορίου και συνεπώς ήταν αναπόφευκτη και η προσθήκη του στον κόσμο της τεχνολογίας, ειδικά με την έξαρση της τεχνολογικής ανάπτυξης τις τρεις τελευταίες δεκαετίες. Αυτό είχε σαν αποτέλεσμα να ιδρυθούν πολλές και μεγάλες εταιρίες σχεδίασης και παραγωγής ηλεκτρονικών παιχνιδιών, επενδύοντας εκατομμύρια δολάρια για τη δημιουργία ενός και μόνο παιχνιδιού κάθε φορά!

Πλέον τα παιχνίδια στους ηλεκτρονικούς υπολογιστές , αλλά και σε επίπεδο κονσόλας, έχουν φτάσει σε τέτοιο σημείο εξέλιξης που παρέχουν στο χρήστη την όσο το δυνατόν ρεαλιστικότερη αναπαράσταση περιβάλλοντος και ενεργειών, τα οποία σε συνδυασμό με τον απόλυτο έλεγχο, μεταφέρουν με μεγάλη επιτυχία τον παίχτη στον κόσμο του παιχνιδιού. Επίσης, στα περισσότερα σύγχρονα παιχνίδια, ο χρήστης μπορεί να χρησιμοποιήσει το διαδίκτυο και να παίζει παράλληλα με πολλούς διαφορετικούς παίχτες ανά τον κόσμο (*MMOG – Massive Multiplayer Online Game*), κάτι που κάνει το εκάστοτε παιχνίδι να ξεφεύγει από τα όρια του απρόσωπου παιχνιδιού για έναν παίχτη (*single player game*). Στα *MMOG*, ο παίχτης μπορεί να συναναστρέφεται διαδικτυακά με άλλους παίχτες, είτε σε επίπεδο συνεργασίας είτε σαν αντίπαλος, κάτι που τον βοηθάει να βελτιώνει τις δεξιότητές του στην ομάδα και τις ατομικές του ικανότητες.

2.2 ΕΙΔΗ ΗΛΕΚΤΡΟΝΙΚΩΝ ΠΑΙΧΝΙΔΙΩΝ ΠΟΥ ΥΠΑΡΧΟΥΝ

Παρατηρείται μια τεράστια ποικιλία ηλεκτρονικών παιχνιδιών για Η/Υ αλλά και για πλατφόρμες ηλεκτρονικών παιχνιδιών (*PlayStation, Nintendo, X-box 360*), που στόχο έχουν να ικανοποιήσουν τις επιθυμίες του κάθε παίχτη. Παρακάτω παρουσιάζεται μια συνοπτική λίστα με τα είδη σύγχρονων τρισδιάστατων παιχνιδιών που υπάρχουν και μια ενδεικτική εικόνα (*screenshot*) ενός παιχνιδιού από κάθε είδος:

- Πάζλ - *Puzzle games* : (**Tetris, Minesweeper**)



Εικόνα 1- Tetris

- Παιχνίδια ρόλων - Role playing games : (World of Warcraft, Lineage, Diablo)



Εικόνα 2 - World Of Warcraft

- Παιχνίδια βολών πρώτου ή τρίτου βαθμού - First/Third person shooter : (Doom, Quake, Unreal, Counter Strike, Max Payne, Splinter Cell, Tomb Rider)



Εικόνα 3 - Counter Strike

- Παιχνίδια στρατηγικής - Strategy games : (Civilization, Age of Empires, Starcraft, Red Alert)



Εικόνα 4 - Age Of Empires

- Παιχνίδια περιπέτειας - *Adventure games* : (**Broken Sword, Assassin's Creed, Space Quest, Myst, Syberia**)



Εικόνα 5 - Assassin's Creed

- Παιχνίδια αθλητισμού - *Sports games* : (**FIFA, NBA Live, Pro Evolution Soccer**)



Εικόνα 6 - FIFA

- Παιχνίδια αγώνων - *Racing games* : (**Need for Speed, Gran Turismo, Formula 1, WRC, Moto-GP**)



Εικόνα 7 - Need For Speed

- Παιχνίδια εξομοίωσης - *Simulation games* : (**The Sims, SimCity, MS Flight Simulator**)



Εικόνα 8 - The Sims

2.3 Η ΚΑΤΗΓΟΡΙΑ ΠΑΙΧΝΙΔΙΩΝ ΤΥΠΟΥ ARCADE

«Πρόγονος» όμως, όλων των παραπάνω ειδών είναι μια ειδική κατηγορία παιχνιδιών, δύο διαστάσεων, αυτή των *arcade games*, πάνω στην οποία βασίζεται και η παρούσα διπλωματική εργασία. Στα μέσα της δεκαετίας του '70 άρχισαν να κάνουν την εμφάνιση τους πλατφόρμες παιχνιδιών σε διάφορα καταστήματα διασκέδασης και ψυχαγωγίας, οι οποίες χρειάζονταν την εισαγωγή κέρματος για να δώσουν τη δυνατότητα στο χρήστη να παίξει το παιχνίδι (*coin operated arcade games*). Οι πλατφόρμες αυτές υποστήριζαν αρχικά μόνο ένα συγκεκριμένο παιχνίδι, ενώ αργότερα εξελίχθηκαν στο να μπορούν να δίνουν στον παίχτη το δικαίωμα να επιλέξει ποιο είδος επιθυμεί να παίξει.

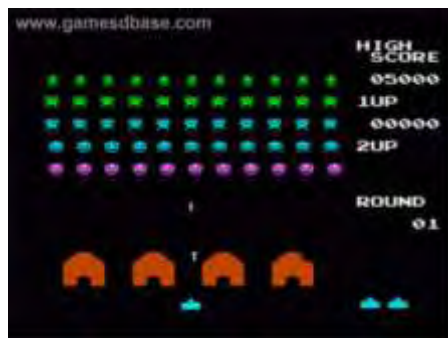
Τα περισσότερα *arcade games* έχουν ένα μικρό αριθμό επιπέδων (*levels*), απλό περιβάλλον γραφικών, απλό και άμεσο τρόπο χειρισμού (*game-play*) και ραγδαία αύξηση δυσκολίας. Αυτό διότι στην ουσία, ο παίχτης νοικιάζει το παιχνίδι για έναν συγκεκριμένο χρόνο ο οποίος επηρεάζεται από τις «ζωές» που έχει στο παιχνίδι και το πώς μπορεί να τις διαχειριστεί για να φτάσει όσο πιο μακριά μπορεί στην πορεία του παιχνιδιού, σημειώνοντας συγκεκριμένο σκορ κάθε φορά. Σκοπός των παικτών είναι να καταφέρουν να βάλουν το όνομά τους (συνήα ψευδώνυμο – *nickname*), στην κορυφή της λίστας των σκορ του παιχνιδιού (*high score*). Το γεγονός ότι έπρεπε να βελτιώσουν τις ικανότητες τους στο παιχνίδι με το ελάχιστο δυνατό χρονικό και χρηματικό κόστος, έκανε τα *arcade games* ιδιαίτερα δύσκολα και ανταγωνιστικά μεταξύ των παιχτών. Μερικά μάλιστα από αυτά ξεπερνούν σε δυσκολία τα σύγχρονα τρισδιάστατα παιχνίδια των πολλών εκατομμυρίων, ακόμα και σήμερα.



Εικόνα 9 - Πλατφόρμα arcade (Pac-Man)

Τα πιο διαδεδομένα *arcade games* είναι τα εξής:

➤ *Space Invaders* (1978)



Εικόνα 10 - Space Invaders

➤ *Galaxian* (1979)



Εικόνα 11 - Galaxian

➤ *Pac-Man* (1980)



Εικόνα 12 - Pac-Man

➤ *Battlezone* (1980)



Εικόνα 13 - Battlezone

➤ *Defender (1980)*



Εικόνα 14 - Defender

➤ *Bosconian (1981)*



Εικόνα 15 - Bosconian

Κάθε παιχνίδι, ανεξαρτήτου είδους ή κονσόλας, μπορεί να υλοποιηθεί σε οποιαδήποτε γλώσσα προγραμματισμού. Ωστόσο, η γλώσσα που κυριαρχεί στη βιομηχανία παραγωγής ηλεκτρονικών παιχνιδιών, είναι η C++. Η C++ χρησιμοποιείται ευρέως διότι είναι μια γλώσσα χαμηλού επιπέδου (βρίσκεται πιο κοντά σε επίπεδο επεξεργαστή), πράγμα που την κάνει «γρηγορότερη» σε σχέση με άλλες γλώσσες υψηλού επιπέδου. Είναι όμως και πιο επίπονη σε συγγραφή κώδικα από άλλες γλώσσες, όπως η *Java*. Χρησιμοποιείται για τη δημιουργία των περισσότερων μεγάλων και σύνθετων παιχνιδιών κι αυτό γιατί το ζητούμενο είναι η βέλτιστη δυνατή απόκριση για τον χρήστη (*speed over usability*).

3.1 ΠΛΕΟΝΕΚΤΗΜΑΤΑ & ΜΕΙΟΝΕΚΤΗΜΑΤΑ ΓΛΩΣΣΩΝ

ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ ΓΙΑ ΤΗΝ ΔΗΜΙΟΥΡΓΙΑ ΠΑΙΧΝΙΔΙΩΝ

ΓΛΩΣΣΑ	ΠΛΕΟΝΕΚΤΗΜΑΤΑ	ΜΕΙΟΝΕΚΤΗΜΑΤΑ
Assembly	Γλώσσα χαμηλού επιπέδου, που ελαχιστοποιεί το κόστος υπολογισμών στον επεξεργαστή. Βέλτιστη για συγκεκριμένες εργασίες και μέρη ενός παιχνιδιού.	Δεν ενδείκνυται για δημιουργία μεγάλων σε έκταση παιχνιδιών. Δύσκολη στην εκμάθηση, επιρρεπής σε σφάλματα, αργή εξέλιξη παραγωγής, μη μεταφέρσιμη.
C	Ευρέως διαδεδομένη και γνωστή. Παρέχεται υποστήριξη από πολλά βοηθητικά εργαλεία και βιβλιοθήκες.	Δεν είναι αντικειμενοστραφής γλώσσα, πράγμα που αποτελεί δυσκολία για τη συγγραφή μεγάλων προγραμμάτων ή τη χρήση πολλαπλών πλατφόρμων.
C++	Ευρέως διαδεδομένη και γνωστή για τη δημιουργία πολύπλοκων παιχνιδιών. Αντικειμενοστραφής γλώσσα με υποστήριξη από μεγάλο αριθμό βοηθητικών εργαλείων.	Η παραγωγή κώδικα κοστίζει σε θέματα διαχείρισης και σωστής κατανομής πόρων. Συχνά απαιτείται αρκετός χρόνος για τη μεταγλώττιση του κώδικα.
C#	Αντικειμενοστραφής γλώσσα, με μηχανισμούς αυτόματης διαχείρισης μνήμης. Δυνατότητα ελέγχου και τροποποίησης των αντικειμένων του προγράμματος εν ώρα τρεξίματος.	Περιορισμένη χρήση σε συγκεκριμένες πλατφόρμες της <i>Microsoft</i> (<i>Windows</i> , <i>Xbox</i>). Ο «καθαρισμός» της μνήμης και η απομάκρυνση άχρηστων δεδομένων, μπορεί να κοστίσουν σε χρόνο απόκρισης.

<p>Java</p>	<p>Αντικειμενοστραφής γλώσσα, εύκολα μεταφέρσιμη σε διαφορετικές πλατφόρμες. Μηχανισμοί αυτόματης διαχείρισης μνήμης. Δυνατότητα ελέγχου και τροποποίησης των αντικειμένων του προγράμματος εν ώρα τρεξίματος. Μεγάλος αριθμός βοηθητικών βιβλιοθηκών και εργαλείων.</p>	<p>Μη διαθέσιμη σε μεγάλες κονσόλες ηλεκτρονικών παιχνιδιών. Ο «καθαρισμός» της μνήμης και η απομάκρυνση άχρηστων δεδομένων, μπορεί να κοστίσουν σε χρόνο απόκρισης.</p>
<p>Scripting languages (JavaScript, Python, Lua, etc.)</p>	<p>Γλώσσες προγραμματισμού που οι εντολές εκτελούνται ταυτόχρονα (όχι ακολουθιακά). Αποτελούν μέρος των <i>web browsers</i> με τους οποίους γίνεται η επικοινωνία σε επίπεδο χρήστη. Αντικειμενοστραφείς γλώσσες που δε χρειάζονται μεταγλώττιση. Ευέλικτες, γρήγορες και εύκολες στη χρήση και στη συγγραφή κώδικα.</p>	<p>Προβλήματα συμβατότητας με διαφορετικούς <i>web browsers</i>. Θέματα ασφάλειας κώδικα. Απαραίτητη χρήση <i>web browser</i>.</p>

Όπως έχει αναφερθεί, η υλοποίηση του παιχνιδιού την οποία καλύπτει η διπλωματική εργασία, θα γίνει σε *JavaScript* με τη χρήση *HTML*.

3.2 ΠΕΡΙΓΡΑΦΗ ΤΩΝ ΓΛΩΣΣΩΝ HTML & JAVASCRIPT

[-] HTML (Hyper Text Markup Language):

Η *HTML* είναι η γλώσσα που «καταλαβαίνουν» οι *web browsers* για την αναπαράσταση των οπτικοακουστικών δεδομένων και έχει ως βασικό της χαρακτηριστικό, τη διασύνδεση με άλλα έγγραφα τύπου *HTML*. Στην *HTML* δε γίνεται διάκριση πεζών και κεφαλαίων γραμμμάτων μέσα στα *tags* (*non case-sensitive language*).

Βασικό δομικό της *HTML* είναι η ετικέτα (*tag*) και κάθε διακριτικό της περικλείεται σε `<>`.

Τα περισσότερα διακριτικά ομαδοποιούνται σε ζεύγη. Το πρώτο δείχνει την αρχή και το δεύτερο το τέλος επίδρασης του συγκεκριμένου διακριτικού.

Πχ. `<HTML> ... </HTML>`.

Σε κάποιο *tag* μπορεί να δίνονται επιπλέον χαρακτηριστικά τα οποία γράφονται μέσα στον ορισμό του *tag*.

Πχ. `<...attribute_name = "value"...>`.

Ο τρόπος με τον οποίον μπορεί να δοθεί διαδραστικότητα (*interaction*) μεταξύ του χρήστη και ενός *HTML* εγγράφου, είναι μέσω της χρήσης *scripts* που είναι γραμμένα στη γλώσσα *JavaScript*.

[-] JavaScript:

Η *JavaScript* είναι μία διαδεδομένη δυναμική, *scripting* γλώσσα προγραμματισμού για τη δημιουργία εφαρμογών, παιχνιδιών για υπολογιστή, καθώς και για προγραμματισμό σε επίπεδο εξυπηρετητή (*server side*), που χρησιμοποιείται στα πλαίσια ενός *HTML* εγγράφου για *interaction* μεταξύ του χρήστη και του εγγράφου *HTML*. Μοιράζεται μερικές ιδιότητες των γλωσσών *Java* και *C*, αλλά από μόνη της αποτελεί μια διαφορετική, αυτούσια γλώσσα προγραμματισμού. Βασίζεται στη χρήση συναρτήσεων πρώτης τάξης (συναρτήσεις μπορούν να δίνονται ως ορίσματα σε άλλες συναρτήσεις για την επιστροφή ή όχι τιμών), γεγονός που την κάνει να ακολουθεί τις αρχές τόσο του συναρτησιακού, όσο και του αντικειμενοστραφούς προγραμματισμού.

Η JavaScript ενσωματώνεται σε HTML σελίδες με τη χρήση του διακριτικού :

```
<script language="Javascript">  
    ...κώδικας...  
</script>
```

«Κατεβαίνει» μαζί με το *HTML* έγγραφο και εκτελείται από τον *web browser* σε επίπεδο χρήστη (*client-side*).

Οι βασικοί λόγοι επιλογής της *JavaScript* ως γλώσσα υλοποίησης του παιχνιδιού της εργασίας είναι οι εξής:

Ταχύτητα: Λόγω του ότι είναι *client-side*, ο κώδικας εκτελείται αμέσως, χωρίς να είναι απαραίτητη οποιαδήποτε μορφή επικοινωνίας με κάποιον απομακρυσμένο *server* για την αποστολή μιας απάντησης.

Απλότητα: Είναι ιδιαίτερα απλή γλώσσα στην εκμάθηση και στην εφαρμογή της. Γνωρίζοντας κάποιος να προγραμματίζει σε *C* και/ή *Java*, μπορεί εύκολα να εξοικειωθεί με την *JavaScript*.

Συμβατότητα: Η *JavaScript* μπορεί να συνδυαστεί και να συνυπάρξει αρμονικά και με άλλες γλώσσες προγραμματισμού, αλλά και μέσα σε *scripts* γραμμένα σε διαφορετική γλώσσα, ενώ μπορεί να χρησιμοποιηθεί σε μια μεγάλη γκάμα εφαρμογών. Σε αντίθεση με άλλες γλώσσες, μπορεί επίσης να χρησιμοποιηθεί σε οποιαδήποτε *web page* ανεξάρτητα από την επέκταση αρχείου (*file extension*).

4

ΣΧΕΔΙΑΣΜΟΣ

ΠΑΙΧΝΙΔΙΟΥ

Σε αυτό το κεφάλαιο θα γίνει μία εκτενής περιγραφή του παιχνιδιού που υλοποιείται. Αρχικά, θα περιγραφεί το σενάριο του παιχνιδιού, στη συνέχεια ο τρόπος χρήσης του και τέλος τα στοιχεία από τα οποία αυτό αποτελείται. Θα παρουσιαστούν και ορισμένες εικόνες (*screenshots*), για την πιο ολοκληρωμένη αντίληψη του παιχνιδιού.

4.1 ΠΕΡΙΓΡΑΦΗ ΣΕΝΑΡΙΟΥ ΠΑΙΧΝΙΔΙΟΥ

Το παιχνίδι που υλοποιείται έχει τίτλο : **ACE OF SPADES** και μοιάζει σε στυλ με το *DEFENDER* (*arcade game* του 1980 – *Βλ. Σελίδα 9,Εικόνα 14*). Ο κεντρικός ήρωας, τον οποίον χειρίζεται ο παίχτης, είναι ένα αεροσκάφος το οποίο ονομάζεται : **BOMBER**. Ο παίχτης έχει τη δυνατότητα να μετακινεί το αεροσκάφος προς οποιαδήποτε κατεύθυνση μέσα στα πλαίσια περιβάλλοντος που εξελίσσεται το παιχνίδι και να πυροβολεί ή να αποφεύγει τους επικείμενους εχθρούς που θα εμφανίζονται.

Οι εχθροί που καλείται να αντιμετωπίσει ο ήρωας, είναι ένα σύνολο από χαρτιά τράπουλας, τα οποία κινούνται συνεχώς, μόνο από δεξιά προς τα αριστερά της οθόνης παιχνιδιού. Το κάθε χαρτί που θα εμφανιστεί, θα είναι τυχαίο (δηλ. οποιοδήποτε από τα 52 φύλλα που έχει μια τράπουλα) και θα εμφανίζεται και σε τυχαία θέση στην οθόνη παιχνιδιού καθ' όλη τη διάρκεια εκτέλεσής του.

Αριστερά από την οθόνη παιχνιδιού, θα υπάρχει ένα πλαίσιο (*table*), στο οποίο θα εμφανίζεται ένας τυχαίος συνδυασμός πέντε φύλλων της τράπουλας. Ο παίχτης καλείται να πυροβολήσει τα σωστά φύλλα που του προτείνονται στο *table* και μάλιστα με τη σωστή σειρά με την οποία εμφανίζονται. Αν πυροβολήσει το σωστό φύλλο, τότε το φύλλο αυτό αφαιρείται από την προτεινόμενη πεντάδα του *table*.

Κάθε φύλλο έχει μια συγκεκριμένη αξία η οποία αποτελεί και την τιμή διαμόρφωσης του σκορ του ήρωα. Αν ο παίχτης πυροβολήσει το σωστό προτεινόμενο φύλλο, η αξία του συγκεκριμένου φύλλο προστίθεται στο σκορ, ενώ αν πυροβολήσει λάθος φύλλο η συγκεκριμένη αξία (του λάθος φύλλου), αφαιρείται από το σκορ του ήρωα. Το σκορ μεταβάλλεται μόνο με την πρόσκρουση της σφαίρας (*bullet*) σε κάποιο φύλλο. Αν το σκορ πέσει κάτω από το 0, το παιχνίδι σταματάει (*Game Over*) και ο παίχτης πρέπει να ξεκινήσει πάλι από το πρώτο στάδιο. Το σκορ του συγκεκριμένου παιχνιδιού δεν αρχίζει από το 0, όπως θα περίμενε κανείς, αλλά από το 50. Αυτό γίνεται έτσι ώστε ο παίχτης να έχει το περιθώριο να πυροβολήσει ένα ή περισσότερα λάθος φύλλα, χωρίς τον άμεσο τερματισμό του παιχνιδιού (πχ. μπορεί δύο φύλλα να επικαλύπτονται).

Τα φύλλα και οι αξίες τους παρουσιάζονται στον παρακάτω πίνακα:

CARD	VALUE
2 of ♠ (two of spades) 2 of ♣ (two of clubs) 2 of ♥ (two of hearts) 2 of ♦ (two of diamonds)	2
3 of ♠ (three of spades) 3 of ♣ (three of clubs) 3 of ♥ (three of hearts) 3 of ♦ (three of diamonds)	3
4 of ♠ (four of spades) 4 of ♣ (four of clubs) 4 of ♥ (four of hearts) 4 of ♦ (four of diamonds)	4

5 of ♠ (five of spades) 5 of ♣ (five of clubs) 5 of ♥ (five of hearts) 5 of ♦ (five of diamonds)	5
6 of ♠ (six of spades) 6 of ♣ (six of clubs) 6 of ♥ (six of hearts) 6 of ♦ (six of diamonds)	6
7 of ♠ (seven of spades) 7 of ♣ (seven of clubs) 7 of ♥ (seven of hearts) 7 of ♦ (seven of diamonds)	7
8 of ♠ (eight of spades) 8 of ♣ (eight of clubs) 8 of ♥ (eight of hearts) 8 of ♦ (eight of diamonds)	8
9 of ♠ (nine of spades) 9 of ♣ (nine of clubs) 9 of ♥ (nine of hearts) 9 of ♦ (nine of diamonds)	9
10 of ♠ (ten of spades) 10 of ♣ (ten of clubs) 10 of ♥ (ten of hearts) 10 of ♦ (ten of diamonds)	10
J of ♠ (Jack of spades) J of ♣ (Jack of clubs) J of ♥ (Jack of hearts) J of ♦ (Jack of diamonds)	10

Q of ♠ (Queen of spades) Q of ♣ (Queen of clubs) Q of ♥ (Queen of hearts) Q of ♦ (Queen of diamonds)	10
K of ♠ (King of spades) K of ♣ (King of clubs) K of ♥ (King of hearts) K of ♦ (King of diamonds)	10
A of ♣ (Ace of clubs) A of ♥ (Ace of hearts) A of ♦ (Ace of diamonds)	11
A of ♠ (Ace of spades)	21

Με την έναρξη του παιχνιδιού, δίνεται ένας συγκεκριμένος αριθμός «ζωών» (*lives*), τις οποίες ο παίχτης πρέπει να διαχειριστεί όσο το δυνατόν πιο σωστά, για να προχωρήσει πιο μακριά στα επίπεδα (*levels*) του παιχνιδιού. Αν το αεροσκάφος συγκρουστεί με κάποιο/α από τα επερχόμενα φύλλα/εχθρούς, τότε αφαιρούνται τόσες ζωές όσες και το πλήθος των φύλλων με τα οποία συγκρούστηκε ο ήρωας. Αν ο αριθμός των διαθέσιμων ζωών του παίχτη φτάσει στο 0, το παιχνίδι σταματάει (*Game Over*) και ο παίχτης πρέπει να ξεκινήσει πάλι από το πρώτο στάδιο. Αξίζει να σημειωθεί εδώ, ότι ακόμα και αν το φύλλο με το οποίο συγκρούεται ο ήρωας είναι προτεινόμενο στο *table*, θα αφαιρεθεί ζωή και το σκορ θα παραμείνει αμετάβλητο.

Το παιχνίδι θα αποτελείται από τέσσερα επίπεδα (*levels/stages*). Κάθε επίπεδο διαθέτει ξεχωριστό περιβάλλον (*background*) και μουσική επένδυση (*background music*). Για να μεταβεί ο παίχτης σε επόμενο επίπεδο, θα πρέπει να πυροβολήσει και τα πέντε προτεινόμενα φύλλα με τη σωστή σειρά. Όταν αλλάζει το επίπεδο (δηλ. έχει αδειάσει το *table* με τον προτεινόμενο συνδυασμό), τότε το *table* ξαναγεμίζει/ανανεώνεται με μία νέα, διαφορετική, τυχαία πεντάδα προτεινόμενων φύλλων. Κατά τη μετάβαση από το ένα επίπεδο στο άλλο, αυξάνεται η ταχύτητα των επερχόμενων φύλλων και του αεροσκάφους, με αποτέλεσμα να κλιμακώνεται η δυσκολία του παιχνιδιού.

Επίσης, κατά την μετάβαση σε ανώτερο επίπεδο, ο παίχτης ανταμείβεται με μια συγκεκριμένη προσαύξηση σκορ και ζωών (*bonus score & bonus lives*), ούτως ώστε να βοηθηθεί στα επερχόμενα, δυσκολότερα στάδια του παιχνιδιού.

Η ολοκλήρωση/τερματισμός του παιχνιδιού θα επιτυγχάνεται όταν ο παίχτης καταφέρει να ολοκληρώσει και το τελευταίο επίπεδο δυσκολίας, ανανεώνοντας έτσι το τελικό, πιθανόν νέο, υψηλότερο σκορ (*new high score*).

4.2 ΤΡΟΠΟΣ ΧΡΗΣΗΣ ΚΑΙ ΑΠΑΙΤΗΣΕΙΣ ΠΑΙΧΝΙΔΙΟΥ

Η κίνηση του αεροσκάφους επιτυγχάνεται με τη χρήση των πλήκτρων κατεύθυνσης : ←, ↑, →, ↓, ή με τη χρήση των πλήκτρων: **A,W,D,S**, τα οποία είναι εξίσου διαδεδομένα για την κίνηση του εκάστοτε ήρωα στα περισσότερα παιχνίδια ηλεκτρονικών υπολογιστών. Επίσης, παρέχεται η δυνατότητα διαγώνιας κίνησης (πάνω & αριστερά, πάνω & δεξιά, κάτω & αριστερά, κάτω & δεξιά). Αν πατηθούν ταυτόχρονα οι συνδυασμοί πλήκτρων: ↑ & ↓, ή ← & → (**W & S** ή **A & D**), το αεροσκάφος παραμένει στην ίδια θέση αφού δεν υπάρχει επιτρεπτή, λογική κίνηση. Οι σφαίρες που φεύγουν από το αεροσκάφος εκτοξεύονται όταν ο παίχτης πατήσει το πλήκτρο: **Space Bar**.

Κατά την εκκίνηση του παιχνιδιού, δηλαδή με το άνοιγμα του *web browser*, εμφανίζεται η αρχική/κύρια οθόνη του παιχνιδιού (*Main Menu Screen*), στην οποία θα υπάρχουν τα εξής διαθέσιμα κουμπιά (*main menu buttons*) :

- **Play Game** : έναρξη παιχνιδιού.
- **How To Play** : μετάβαση στην οθόνη οδηγιών παιχνιδιού (*Manual Screen*).
Στην οθόνη οδηγιών, γίνεται μια σύντομη περιγραφή του τρόπου με τον οποίον παίζεται το παιχνίδι (σκοπός / περιορισμοί) και παρουσίαση των πλήκτρων κίνησης και δράσης (*game control keys*). Υπάρχουν επίσης, δύο επιπλέον κουμπιά (**Play Game** και **Return To Menu**), με τα οποία ο παίχτης έχει τη δυνατότητα, είτε να επιλέξει την έναρξη του παιχνιδιού άμεσα, είτε να επιστρέψει στην αρχική οθόνη (*Main Menu Screen*).
- **High Score**: εμφάνιση παραθύρου με το τρέχον υψηλότερο σκορ που έχει επιτευχθεί.
- **Exit** : έξοδος από το παιχνίδι με κλείσιμο του παραθύρου του *web browser* (αν ο χρήστης έχει ανοιχτές παραπάνω από μία καρτέλες στον *web browser* και πατήσει την επιλογή **Exit**, κλείνει μόνο η καρτέλα στην οποία «έτρεχε» το παιχνίδι).

Κατά τον τερματισμό του παιχνιδιού (ο παίχτης χάνει ή ολοκληρώνει το παιχνίδι επιτυχώς), εμφανίζεται η οθόνη τερματισμού (*Game Over Screen*), η οποία και αυτή με τη σειρά της διαθέτει τα κουμπιά : **Return To Menu** (για επιστροφή στην αρχική οθόνη) και **Exit** (για έξοδο από το παιχνίδι).

Όταν γίνει εκκίνηση του παιχνιδιού, κάτω από την κεντρική οθόνη περιβάλλοντος, θα εμφανίζονται δύο επιπλέον κουμπιά:

- **Return To Menu** : για άμεση επιστροφή στην αρχική οθόνη.
- **Pause** : για παύση της ροής του παιχνιδιού (όσο ο παίχτης βρίσκεται σε κατάσταση *pause*, το κουμπί **Pause** αντικαθίσταται από ένα νέο κουμπί: **Resume**, για τη συνέχεια του παιχνιδιού).

Οι ελάχιστες απαιτήσεις που χρειάζονται για να τρέξει το παιχνίδι σε έναν οποιονδήποτε ηλεκτρονικό υπολογιστή, είναι η ύπαρξη της τελευταίας έκδοσης ενός από τους υπάρχοντες *web browsers* : Google Chrome, Mozilla Firefox, Internet Explorer, Opera, Safari και 50 Mb ελεύθερου αποθηκευτικού χώρου για τα αρχεία και τους φακέλους του παιχνιδιού.

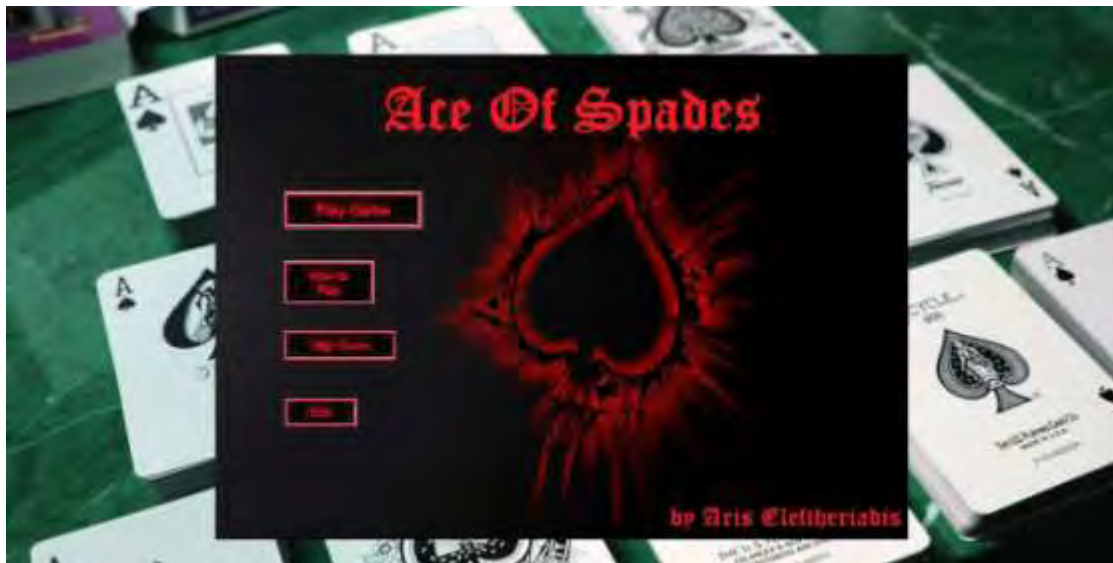
4.3 ΔΟΜΙΚΑ ΣΤΟΙΧΕΙΑ ΠΑΙΧΝΙΔΙΟΥ

Το παιχνίδι είναι βασισμένο σε έτοιμα οπτικοακουστικά στοιχεία, δηλαδή έτοιμα αρχεία εικόνας και ήχου, τα οποία έχουν τροποποιηθεί κατάλληλα για να καλύπτουν τις απαιτήσεις του παιχνιδιού (βλ. Βιβλιογραφία για τις πηγές των αρχείων). Η επιλογή αυτή έχει ως σκοπό να προσδώσει στον παίχτη την πιο ρεαλιστική απεικόνιση περιβάλλοντος και αντικειμένων σε δύο διαστάσεις.

[-] ΣΤΟΙΧΕΙΑ ΕΙΚΟΝΑΣ :

Με το άνοιγμα του παραθύρου στο οποίο θα τρέχει το παιχνίδι, το *background* της καρτέλας του *web browser* αντικαθίσταται από μια συγκεκριμένη στατική εικόνα (*web browser background*) και στο κέντρο του παραθύρου εμφανίζεται το κύριο πλαίσιο στο οποίο διαδραματίζεται το παιχνίδι. Το πλαίσιο αυτό θα ανανεώνεται συνεχώς και θα αλλάζει τα στοιχεία εικόνας και ήχου, ανάλογα με την κατάσταση στην οποία βρίσκεται το παιχνίδι κάθε φορά.

-Αρχικά, στο κύριο πλαίσιο (διαστάσεις σε *pixels*: 1000x700) εμφανίζεται η αρχική οθόνη (*main menu screen*).



Εικόνα 16 - Web browser Background & Main Menu Screen

-Αν ο παίχτης επιλέξει να πατήσει το κουμπί **How To Play**, τότε στο κύριο πλαίσιο εμφανίζεται η οθόνη οδηγιών.



Εικόνα 17 – Manual Screen

-Αν ο παίχτης επιλέξει να πατήσει **Play Game** (σε οποιαδήποτε από τις δύο οθόνες), τότε εμφανίζεται στα αριστερά ένα νέο πλαίσιο (το *table* με τα πέντε προτεινόμενα φύλλα), τα κουμπιά **Return To Menu** και **Pause/Play** και το κεντρικό πλαίσιο καθαρίζεται για να εκκινήσει το παιχνίδι με νέα στοιχεία εικόνας.

Στο σημείο αυτό αρχίζει και η αλληλεπίδραση των αντικειμένων του παιχνιδιού τα οποία παρουσιάζονται παρακάτω:

- ❖ **Game Backgrounds:** εικόνες πόλεων φωτισμένων τη νύχτα, που μαζί με την ταχύτητα κίνησης και το κομμάτι μουσικής, αποτελούν τα διακριτικά για κάθε επίπεδο δυσκολίας («πίστες» - *stages*).



Εικόνα 18 - Πρώτο Επίπεδο (Stage 1)



Εικόνα 19 - Δεύτερο Επίπεδο (Stage 2)



Εικόνα 20 - Τρίτο Επίπεδο (Stage 3)



Εικόνα 21 - Τέταρτο Επίπεδο (Stage 4)

- ❖ **Bomber:** το αεροσκάφος που χειρίζεται ο παίχτης. Πρόκειται για μία εικόνα ενός αεροσκάφους του Β' Παγκοσμίου Πολέμου.



Εικόνα 22 - The Bomber

- ❖ **Bullet:** η σφαίρα που φεύγει από το αεροσκάφος, όταν ο παίχτης πυροβολεί. Εικόνα μιας φλεγόμενης σφαίρας εν ώρα πυροβολισμού.



Εικόνα 23 - Bomber's Bullet

- ❖ **Deck of Cards:** το σύνολο των φύλλων της τράπουλας (η παρακάτω εικόνα έχει διασπαστεί σε 52 αντικείμενα εικόνας, ούτως ώστε το κάθε ένα να αποτελεί ένα ξεχωριστό φύλλο).

2	3	4	5	6	7	8	9	10	J	Q	K	A
♥	♥	♥	♥	♥	♥	♥	♥	♥	♥	♥	♥	♥
2	3	4	5	6	7	8	9	10	J	Q	K	A
♦	♦	♦	♦	♦	♦	♦	♦	♦	♦	♦	♦	♦
2	3	4	5	6	7	8	9	10	J	Q	K	A
♣	♣	♣	♣	♣	♣	♣	♣	♣	♣	♣	♣	♣
2	3	4	5	6	7	8	9	10	J	Q	K	A
♠	♠	♠	♠	♠	♠	♠	♠	♠	♠	♠	♠	♠

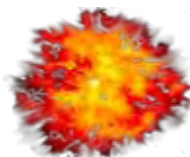
Εικόνα 24 - Deck of Cards

- ❖ **Explosion:** η εικόνα που εμφανίζεται σαν γραφικό εφέ τη στιγμή της πρόσκρουσης μιας σφαίρας με οποιοδήποτε φύλλο της τράπουλας.



Εικόνα 25 - Explosion effect

- ❖ **Impact:** η εικόνα που εμφανίζεται σαν γραφικό εφέ τη στιγμή της πρόσκρουσης του αεροσκάφους με οποιοδήποτε φύλλο της τράπουλας.



Εικόνα 26 - Impact effect

- ❖ **Table:** το επιπλέον πλαίσιο που εμφανίζεται αριστερά του κύριου πλαισίου, με την έναρξη του παιχνιδιού, μέσα στο οποίο τοποθετείται ο τυχαίος συνδυασμός των πέντε προτεινόμενων φύλλων.



Εικόνα 27 - Table with recommended cards to shoot

- ❖ **In-play buttons:** τα κουμπιά, **Return to Menu** και **Pause/Play**, που εμφανίζονται κάτω από το κύριο πλαίσιο με την έναρξη του παιχνιδιού.



Εικόνα 28 - Buttons Back to Menu & Pause



Εικόνα 29 - Buttons Back to Menu & Resume

- ❖ **Lives counter:** ο μετρητής που δείχνει τον αριθμό ζωών του παίχτη και εμφανίζεται στο πάνω-αριστερά μέρος του κύριου πλαισίου παιχνιδιού.



Εικόνα 30 - Lives display

- ❖ **Score counter:** ο μετρητής που δείχνει το σκορ του παίχτη και εμφανίζεται στο πάνω-δεξιά μέρος του κύριου πλαισίου παιχνιδιού.



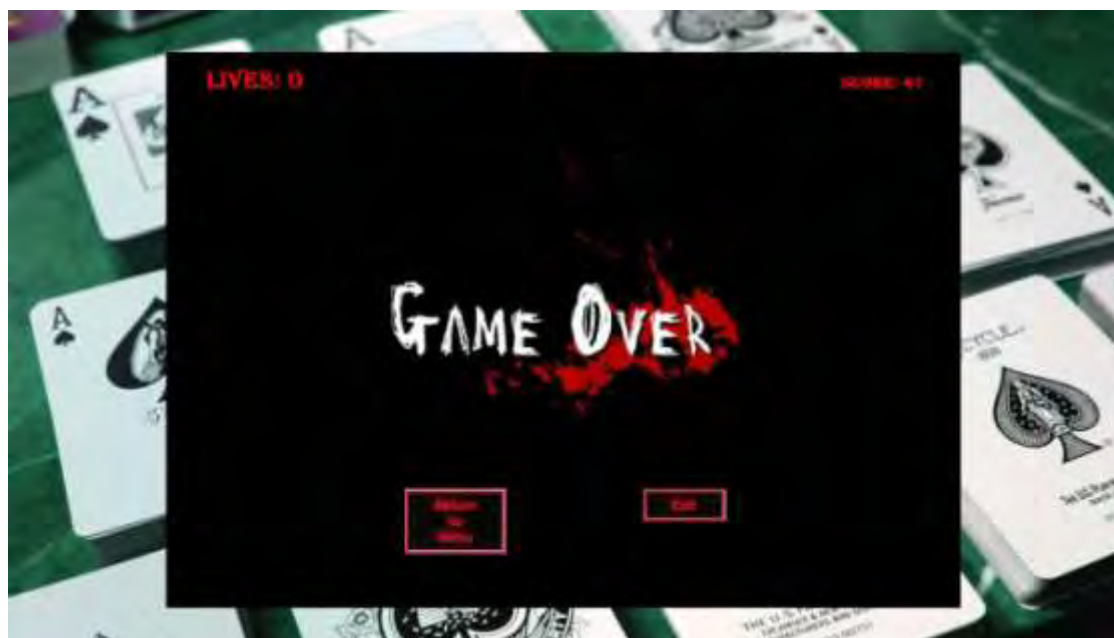
Εικόνα 31 - Score display

-Ο ταυτόχρονος συνδυασμός όλων των παραπάνω στοιχείων εικόνας, έχει σαν αποτέλεσμα την παραγωγή του παρακάτω στιγμιότυπου εν ώρα εξέλιξης του παιχνιδιού :



Εικόνα 32 - Στιγμιότυπο παιχνιδιού, εν ώρα δράσης

-Με τον τερματισμό του παιχνιδιού, γίνεται εκκαθάριση όλων των παραπάνω στοιχείων από το κύριο πλαίσιο, εξαφάνιση του πλαισίου προτεινόμενων φύλλων και φόρτωσης της οθόνης τερματισμού.



Εικόνα 33 - Game Over screen

[-] ΣΤΟΙΧΕΙΑ ΗΧΟΥ :

Στο παιχνίδι έχουν ενσωματωθεί και ορισμένα ηχητικά αρχεία για να αυξήσουν το ενδιαφέρον του παίκτη και για να γίνει πιο συναρπαστική η εμπειρία παιχνιδιού. Συγκεκριμένα, έχουν εισαχθεί ηχητικά εφέ (από έτοιμη *online* βιβλιοθήκη ήχων στο διαδίκτυο) για τις βασικές ενέργειες της αλληλεπίδρασης του αεροσκάφους με τις σφαίρες που εκτοξεύει, με τα επερχόμενα φύλλα και με τον τρόπο τερματισμού του παιχνιδιού. Υπάρχουν δηλαδή, τέσσερα διαφορετικά ηχητικά εφέ εκρήξεων για κάθε ένα από τα παρακάτω γεγονότα:

1. Όταν το αεροσκάφος εκτοξεύει σφαίρα.
2. Όταν γίνεται πρόσκρουση της σφαίρας με κάποιο φύλλο της τράπουλας.
3. Όταν γίνεται πρόσκρουση του αεροσκάφους με κάποιο φύλλο της τράπουλας.
4. Όταν τελειώσουν οι ζωές του παίκτη ή όταν το σκορ γίνει αρνητικό – εμφάνιση οθόνης τερματισμού.

Για την πιο διασκεδαστική εξέλιξη του παιχνιδιού, έχουν ενσωματωθεί ορισμένα μουσικά κομμάτια, διαφορετικά για κάθε επίπεδο και για κάθε οθόνη που εμφανίζεται στο κύριο πλαίσιο παιχνιδιού. Κατά τη μετάβαση από το ένα επίπεδο στο άλλο, το τρέχον μουσικό κομμάτι που αναπαράγεται, σταματάει και αρχίζει η αναπαραγωγή του επομένου. Το ίδιο συμβαίνει και κατά την εναλλαγή στις στατικές οθόνες (*main menu screen & manual screen/ game over screen*). Τα μουσικά κομμάτια μαζί με τους καλλιτέχνες που τα ερμηνεύουν, ανήκουν στο στυλ μουσικής «Rock n' Roll» και παρουσιάζονται στον παρακάτω πίνακα (*playlist*) :

Song Title	Artist	Play in :
Game Over	Christopher Ashmore	Main menu & manual screen
Youth Gone Wild	Skid Row	Stage 1
Big City Nights	Scorpions	Stage 2
The Chase Is Better Than The Catch	Motörhead	Stage 3
Ace Of Spades	Motörhead	Stage 4
Whorehouse Blues	Motörhead	Game over screen

4.4 ΜΕΘΟΔΟΣ ΔΟΜΗΣΗΣ ΠΑΙΧΝΙΔΙΟΥ

Οι τρόποι δόμησης ενός παιχνιδιού για ηλεκτρονικό υπολογιστή διαφέρουν, ανάλογα με την πολυπλοκότητα και τις απαιτήσεις που πρέπει να καλυφθούν. Στο παιχνίδι που σχεδιάζεται στα πλαίσια της παρούσας εργασίας, η μέθοδος που επιλέχτηκε είναι η εξής:

Αρχικά συγκεντρώνονται όλα τα απαραίτητα αρχεία ήχου και εικόνας που θα αποτελούν τα αντικείμενα του παιχνιδιού. Τα αρχεία ήχου είναι σε μορφή .mp3. Τα αρχεία εικόνας μετατρέπονται σε μορφή .png (*Portable Network Graphics*). Τα αρχεία .png περιλαμβάνουν παλέτα χρωμάτων 24-bit, που με απλά λόγια σημαίνει ότι οι εικόνες θα έχουν τεράστια γκάμα χρωμάτων με αποτέλεσμα να "μη χαλάει η ποιότητά τους". Επίσης, υποστηρίζουν πολύ καλό *transparency*. Το *transparency* είναι μια ιδιότητα που κάνει το γραφικό να φαίνεται χωρίς *background*. Τα μεγάλα πλεονεκτήματα των αρχείων .png σε αντίθεση με τα .gif ή .jpeg, είναι η πολύ καλή απόδοση χρωμάτων και η πολύ καλή ποιότητα απόδοσης του *transparency*.

Στη συνέχεια, διαμορφώνεται κάθε αρχείο εικόνας έτσι ώστε να έχει την επιθυμητή μορφή και τις επιθυμητές διαστάσεις με τη χρήση κάποιων προγραμμάτων διαμόρφωσης και μετατροπής εικόνας. Για τους σκοπούς της εργασίας, χρησιμοποιήθηκαν τα βοηθητικά προγράμματα: **Paint.NET** και **Photo Shop** για εμπλουτισμό και μετατροπή διαστάσεων. Τα κουμπιά (*buttons*) δημιουργούνται με online γεννήτριες κουμπιών (*online button generators*) και ενσωματώνονται στο περιβάλλον του παιχνιδιού, ενώ η διαδραστικότητά τους δίνεται μέσω των *HTML* και *JavaScript*, με τη χρήση κατάλληλων εντολών.

Το επόμενο στάδιο δόμησης, είναι η δημιουργία ενός ειδικού αρχείου εικόνας που ονομάζεται **sprite image**, το οποίο πρόκειται για ένα αρχείο εικόνας μεγάλων διαστάσεων καθώς αποτελείται από ένα σύνολο εικόνων μικρότερων διαστάσεων. Ουσιαστικά, όλες οι εικόνες που συνθέτουν το παιχνίδι τοποθετούνται η μία δίπλα/κάτω στην/από την άλλη (σωστά στοιχισμένες, με τη χρήση εργαλείου χάρακα), δημιουργώντας έτσι το ενιαίο αρχείο εικόνας *sprite*. Το *sprite* αρχείο θα χρησιμοποιηθεί ως πηγή (*source*) από την *JavaScript* για την επιλογή σχεδίασης κάθε αντικειμένου του παιχνιδιού. Με πιο απλά λόγια, η *JavaScript*, με τη χρήση κατάλληλων συναρτήσεων σχεδίασης, θα επιλέγει ποιο στοιχείο του *sprite* αρχείου θα «ζωγραφίζει» κάθε φορά, με βάση τις συντεταγμένες που έχει το στοιχείο αυτό μέσα στο *sprite* αρχείο. Το *sprite* αρχείο του παιχνιδιού της εργασίας φαίνεται παρακάτω:



Εικόνα 34 - Sprite Image

Σε τελικό στάδιο, δημιουργείται ένας φάκελος μέσα στον οποίον τοποθετούνται τα αρχεία ήχου (.mp3) , το *sprite* αρχείο (.png), το αρχείο εικόνας (.jpeg) για το *background* του *browser* και τα αρχεία κώδικα *HTML* και *JavaScript* (.html και .js αντίστοιχα).

5

ΥΛΟΠΟΙΗΣΗ

ΠΑΙΧΝΙΔΙΟΥ

Σε αυτό το κεφάλαιο θα γίνει λεπτομερής ανάλυση του τρόπου υλοποίησης του παιχνιδιού χρησιμοποιώντας τις γλώσσες *HTML* και *JavaScript*. Θα γίνει περιγραφή των αντικειμένων, των αλγορίθμων και των συναρτήσεων του κώδικα υλοποίησης και του πώς όλα αυτά επηρεάζουν την εξέλιξη και την αλληλεπίδραση των οπτικοακουστικών στοιχείων του παιχνιδιού. Ο συνολικός κώδικας χωρίζεται σε δύο αρχεία: το αρχείο `.html` και το αρχείο `.js`.

5.1 ΠΕΡΙΓΡΑΦΗ ΤΟΥ HTML ΑΡΧΕΙΟΥ (Ace of Spades.html)

Όπως αναφέρθηκε στο Κεφάλαιο 3, η *HTML* είναι η γλώσσα που καταλαβαίνει ο *web browser*. Με το άνοιγμα του `.html` αρχείου ανοίγει και το παράθυρο του *web browser* στο οποίο θα τρέχει το παιχνίδι. Για να δει κάποιος τον κώδικα της *HTML*, αρκεί να πατήσει, οπουδήποτε μέσα στο παράθυρο του *web browser*, δεξί κλικ και στη συνέχεια inspect element. Τότε εμφανίζεται στο κάτω μέρος του παραθύρου ένα πλαίσιο με τα στοιχεία και τον κώδικα της *HTML*.

Ολόκληρος ο κώδικας της *HTML* περικλείεται μέσα στα *tags*:

`<html lang="en"> ... </html>` , όπου ορίζεται η γλώσσα που θα χρησιμοποιηθεί από τον *web browser*.

Στη συνέχεια αποδίδεται ένα όνομα (**Ace of Spades**) στο παράθυρο/καρτέλα στο οποίο τρέχει το παιχνίδι:

<head>

<title>Ace of Spades</title>

</head>

Ακολούθως, δημιουργείται το κύριο σώμα της *HTML* και ορίζεται το *background* του *web browser*, με τη χρήση των *tags*:

<body background="bg.jpeg"> ... </body>, μέσα στα οποία ορίζονται τα περιεχόμενα της *HTML*, όπως εικόνα, καμβάς, ήχος, *hyperlinks*, κείμενο, κτλ.

Στο συγκεκριμένο *body*, ορίζονται τα εξής περιεχόμενα με τα αντίστοιχα *tags*:

- Στοιχεία ήχου με αναγνωριστικά (*id*) και την πηγή στην οποία βρίσκονται (*source*). Αν κάποιος *web browser* δεν υποστηρίζει *audio tags*, εμφανίζεται ενημερωτικό μήνυμα πχ.

<audio id="gameOver"

src="sounds/Christopher Ashmore - Game Over.mp3">

Your Browser Does Not Support the HTML audio tag!

</audio>

(Παρόμοια για όλα τα στοιχεία ήχου του παιχνιδιού)

- Στοιχεία καμβά, τα οποία απαιτούνται για τη δημιουργία και επεξεργασία γραφικών στον *web browser*. Αυτό επιτυγχάνεται μόνο με τη χρήση *script* (στη συγκεκριμένη περίπτωση, θα γίνει χρήση της *JavaScript*). Κάθε καμβάς έχει τις εξής ιδιότητες: ένα αναγνωριστικό (*id*), πλάτος (*width*), ύψος (*height*) και θέση που θα τοποθετηθεί μέσα στο παράθυρο (*style*). Στο παιχνίδι της εργασίας θα χρησιμοποιηθούν έξι καμβάδες, οι οποίοι είναι :

1. Καμβάς που διαχειρίζεται το background του παιχνιδιού
2. Καμβάς που διαχειρίζεται τα φύλλα της τράπουλας
3. Καμβάς που διαχειρίζεται το αεροσκάφος
4. Καμβάς που διαχειρίζεται το σκορ του παίκτη
5. Καμβάς που διαχειρίζεται τις ζωές του παίκτη
6. Καμβάς που διαχειρίζεται το table με τα πέντε προτεινόμενα φύλλα

Η σειρά που δηλώνονται είναι σημαντική, διότι ο κάθε επόμενος καμβάς θα επικαλύπτει τον προηγούμενό του, με αποτέλεσμα όλοι οι καμβάδες να συνυπάρχουν πάνω στον κύριο καμβά του παιχνιδιού, που είναι ο *background canvas*, και αποτελεί και το κύριο πλαίσιο του παιχνιδιού.

Ενδεικτικά παρουσιάζεται ο καμβάς του *background* του παιχνιδιού:

```
<canvas id="canvasBackground"
width="1000px"
height="700px"
style="display:block;
background:#ffffff;margin:100px auto 0px;">
</canvas>
```

- Στοιχεία *hyperlink* για την εισαγωγή και την επεξεργασία των *in-play* κουμπιών:

Return to Menu, **Pause** και **Resume** . Τα *hyperlinks* περικλείονται στα *tags*:

```
<a> ... </a>.
```

Τα αντίστοιχα *tags* έχουν παρθεί έτοιμα από *online button generator* (βλ. Βιβλιογραφία). Ενδεικτικά παρουσιάζεται το *tag* του **Return to Menu button** :

```
<a id="returnBtn" hidden href="">

</a>
```

- Στοιχείο *script* για τη διαχείριση των καμβάδων μέσω μιας *scripting* γλώσσας προγραμματισμού (όπως η *JavaScript*), μέσα στα *tags*:

```
<script> ... </script> .
```


Το στοιχείο *script* μπορεί να έχει *scripting statements* ή να «δείχνει» σε ένα εξωτερικό αρχείο *script*, με τη χρήση του χαρακτηριστικού “src”:

```
<script src="aos.js"></script> ,
```

όπου aos.js είναι το *JavaScript* αρχείο του παιχνιδιού.

5.2 ΠΕΡΙΓΡΑΦΗ ΤΟΥ JAVASCRIPT ΑΡΧΕΙΟΥ (aos.js)

Στο *JavaScript* αρχείο, βρίσκεται ουσιαστικά όλος ο κώδικας που αποτελεί και τον πυρήνα του παιχνιδιού. Περιλαμβάνει όλες τις απαραίτητες μεταβλητές, σταθερές, αντικείμενα (*objects*) και συναρτήσεις που απαιτούνται για να μεταχειριστούν τα στοιχεία του καμβά που δημιουργήθηκαν στο *.html* αρχείο με το οποίο συνδέεται.

5.3 ΟΡΙΣΜΟΣ ΜΕΤΑΒΛΗΤΩΝ ΚΑΙ ΔΗΜΙΟΥΡΓΙΑ

ΑΝΤΙΚΕΙΜΕΝΩΝ

Σε αρχικό στάδιο δημιουργούνται οι μεταβλητές που θα αντιπροσωπεύουν όλα τα στοιχεία του καμβά και τα στοιχεία ήχου. Αυτό γίνεται με την αντιστοίχιση των αναγνωριστικών κάθε καμβά και αρχείου ήχου στις αντίστοιχες μεταβλητές, μέσω της εντολής: **var x = document.getElementById('id');**

Έπειτα, για κάθε καμβά ορίζεται μία μεταβλητή με την οποία θα γίνεται η επεξεργασία του περιεχομένου (*context*) του καμβά. Αυτό επιτυγχάνεται με την εντολή:

```
var context_name = canvas_name.getContext('2d');
```

Ενδεικτικά παρουσιάζονται οι μεταβλητές επεξεργασίας καμβά για το *background* του παιχνιδιού και η μεταβλητή επεξεργασίας ήχου για ένα ηχητικό κομμάτι (αντίστοιχα και για τα υπόλοιπα στοιχεία ήχου και καμβά):

```
var canvasBackground = document.getElementById('canvasBackground');
```

```
var ctxBackground = canvasBackground.getContext('2d');
```

```
var bulletSound = document.getElementById('bulletSound');
```

Δημιουργείται αντικείμενο για τον σύνδεσμο του αρχείου *JavaScript* με το *sprite* αρχείο εικόνας (εντολή:

```
var imgSprite = new Image();
```

```
imgSprite.src = 'spr/sprite.png';
```

```
imgSprite.addEventListener('load', init, false);
```

), από το οποίο, με τη χρήση συντεταγμένων θα ζωγραφίζεται ένα συγκεκριμένο στοιχείο εικόνας του *sprite* μέσα σε κάποιον καμβά. Ο σύνδεσμος γίνεται με το άνοιγμα του παραθύρου του *web browser*.

Πιο συγκεκριμένα απαιτούνται:

- Οι συντεταγμένες (x,y) στις οποίες αρχίζει μια εικόνα στο *sprite* αρχείο
- Το πλάτος της συγκεκριμένης εικόνας
- Το ύψος της συγκεκριμένης εικόνας
- Οι συντεταγμένες (x,y) στον καμβά στις οποίες θα αρχίσει να ζωγραφίζεται η εικόνα

Ορίζονται τα αντικείμενα που θα αποτελούν τα κουμπιά (*buttons*), που εμφανίζονται σε κάθε μια από τις : *main menu screen*, *manual screen*, *game over screen* . Κάθε *button* έχει τις δικές του συντεταγμένες και το δικό του μέγεθος μέσα στο *sprite* αρχείο απ' όπου επιλέγονται.

```
Πχ. var btnPlay = new Button(sprite_x, sprite_x + width, sprite_y,
sprite_y + height);
```

Παρόμοια για όλα τα *button objects*.

Στη συνέχεια ορίζονται οι μεταβλητές που θα χρησιμοποιηθούν από τις συναρτήσεις και τα αντικείμενα του προγράμματος. Τέλος γίνεται χρήση της βασικής συνάρτησης:

window.requestAnimationFrame. Η συνάρτηση αυτή ενημερώνει τον *browser* ότι πρόκειται να ζωγραφίσει κάποιο γραφικό και απαιτεί από αυτόν να καλέσει μια συγκεκριμένη μέθοδο πριν ζωγραφίσει το επόμενο γραφικό.

5.4 ΥΛΟΠΟΙΗΣΗ ΑΝΤΙΚΕΙΜΕΝΩΝ ΚΑΙ ΣΥΝΑΡΤΗΣΕΩΝ

Μέχρι τώρα έχουν δημιουργηθεί τα *objects* για το *sprite* αρχείο και για τα *buttons* στις οθόνες του παιχνιδιού. Στη συνέχεια θα παρουσιαστούν όλα τα *objects* του προγράμματος με τις αντίστοιχες μεθόδους τους.

Bomber object :

Δημιουργία αντικειμένου του ήρωα με απόδοση των *attributes* από τα οποία αποτελείται. Αυτά είναι:

- οι συντεταγμένες της εικόνας του (x,y) στο *sprite* αρχείο
- το πλάτος και το ύψος του
- οι συντεταγμένες (x,y) του καμβά στις οποίες θα εμφανιστεί με την έναρξη του παιχνιδιού
- οι συντεταγμένες (x,y) του δεξιού, του αριστερού, του πάνω και του κάτω άκρου του αεροσκάφους
- η ταχύτητα του αεροσκάφους
- οι επιτρεπόμενες ενέργειες κίνησης και δράσης του ήρωα
- οι διαθέσιμες σφαίρες του ήρωα
- το σκορ και οι ζωές του παίχτη

Bomber functions :

- **Bomber.prototype.draw=function()**: Μέθοδος του αντικειμένου *bomber*, που ζωγραφίζει το αεροσκάφος μέσα στον αντίστοιχο καμβά. Κάθε φορά που καλείται, καθαρίζεται ο καμβάς του αεροσκάφους, ανανεώνονται οι συντεταγμένες του, γίνεται έλεγχος για την κατεύθυνση κίνησης του αεροσκάφους και για το αν προσέκρουσε σε κάποιο φύλλο. Επίσης, ελέγχεται αν έχει δοθεί εντολή από τον παίχτη να πυροβολήσει και ζωγραφίζει τις σφαίρες σε περίπτωση που αυτό ισχύει.
- **Bomber.prototype.updateCoords=function()**: Μέθοδος του αντικειμένου *bomber*, για την ενημέρωση των συντεταγμένων που περιγράφουν το αεροσκάφος, με βάση τη θέση που βρίσκεται κάθε φορά στον καμβά του παιχνιδιού (στην ουσία του *background canvas*).

- **Bomber.prototype.checkDirection=function():** Μέθοδος του αντικειμένου *bomber*, για τον έλεγχο της κατεύθυνσης κίνησης του αεροσκάφους. Επίσης, περιορίζει τη θέση του αεροσκάφους στο να βρίσκεται εντός επιτρεπτών ορίων του παιχνιδιού.
- **Bomber.prototype.drawAllBullets=function():** Μέθοδος του αντικειμένου *bomber*, που καλείται όταν το αεροσκάφος πυροβολεί. Με την κλήση της μεθόδου ζωγραφίζεται κάθε σφαίρα που φεύγει από τα αεροσκάφος και ελέγχεται αν κάποια σφαίρα συγκρούστηκε με κάποιο φύλλο. Αν ναι, τότε ζωγραφίζεται το γραφικό που αντιστοιχεί στη σύγκρουση σφαίρας-φύλλου και γίνεται η αναπαραγωγή του αντίστοιχου ηχητικού εφέ.
- **Bomber.prototype.checkShooting=function():** Μέθοδος του αντικειμένου *bomber*, που ελέγχει αν έχει δοθεί εντολή από τον παίκτη να πυροβολήσει (αν έχει πατηθεί το πλήκτρο **Space Bar**).
- **Bomber.prototype.checkImpact=function():** Μέθοδος του αντικειμένου *bomber*, που ελέγχει αν το αεροσκάφος έχει συγκρουστεί με κάποιο φύλλο. Αν ναι, τότε ζωγραφίζεται το γραφικό που αντιστοιχεί στη σύγκρουση αεροσκάφους-φύλλου, γίνεται η αναπαραγωγή του αντίστοιχου ηχητικού εφέ και η ανανέωση (μείωση) του αριθμού των ζωών του παίκτη.
- **Bomber.prototype.updateScoreCounter=function(points):** Μέθοδος του αντικειμένου *bomber*, για την ενημέρωση και ανανέωση (αύξηση / μείωση) του σκορ του παίκτη με βάση το φύλλο που έχει χτυπηθεί. Επίσης, γίνεται έλεγχος για το αν το σκορ πέσει κάτω από το 0. Αν ναι, το παιχνίδι τερματίζει.
- **Bomber.prototype.updateLivesCounter = function():** Μέθοδος του αντικειμένου *bomber*, για την ενημέρωση και ανανέωση του αριθμού των ζωών του παίκτη. Αν ο αριθμός ζωών φτάσει στο 0 το παιχνίδι τερματίζει και σε περίπτωση νέου υψηλότερου σκορ (*new high score*), εμφανίζεται παράθυρο ενημέρωσης (*alert box*).

Bullet object :

Δημιουργία αντικειμένου της σφαίρας με απόδοση των *attributes* από τα οποία αποτελείται.

Αυτά είναι:

- οι συντεταγμένες της εικόνας της (x,y) στο *sprite* αρχείο
- το πλάτος και το ύψος της
- οι συντεταγμένες (x,y) του καμβά στις οποίες θα εμφανιστεί με την έναρξη του παιχνιδιού

Bullet functions :

- **Bullet.prototype.draw=function()**: Μέθοδος του αντικειμένου *bullet*, που ζωγραφίζει κάθε σφαίρα που εκτοξεύεται με συγκεκριμένη και σταθερή ταχύτητα . Αν η σφαίρα φτάσει στο δεξί άκρο του κύριου καμβά του παιχνιδιού, χωρίς να προσκρούσει πάνω σε κάποιο φύλλο, τότε ανακυκλώνεται μέσα στον πίνακα που αποτελείται από αντικείμενα τύπου *bullet*.
- **Bullet.prototype.fire = function (startX, startY)**: Μέθοδος του αντικειμένου *bullet*, που τοποθετεί τις συντεταγμένες του αρχικού σημείου στο οποίο θα ζωγραφιστεί η σφαίρα, σύμφωνα με τις συντεταγμένες της «μύτης» (του δεξιού άκρου) του αεροσκάφους από την οποία εκτοξεύεται η σφαίρα.
- **Bullet.prototype.checkHitCard=function()**: Μέθοδος του αντικειμένου *bullet*, που ελέγχει αν κάποιο φύλλο έχει χτυπηθεί από τη συγκεκριμένη σφαίρα. Αν ναι, τότε δημιουργείται ένα νέο τυχαίο αντικείμενο φύλλου και τοποθετείται στη θέση του παλιού που χτυπήθηκε, στον πίνακα αντικειμένων τύπου *cards*. Το φύλλο που χτυπήθηκε αφαιρείται και από τον καμβά του παιχνιδιού. Επίσης, με την κλήση της μεθόδου γίνεται ανανέωση του σκορ ανάλογα με την αξία του φύλλου που χτυπήθηκε και αλλαγή του επιπέδου του παιχνιδιού, σε περίπτωση που ο παίχτης πυροβολήσει όλα τα σωστά προτεινόμενα φύλλα του τρέχοντος επιπέδου. Αν ο παίχτης ολοκληρώσει με επιτυχία όλα τα στάδια, εμφανίζεται ένα *alert box* που τον ενημερώνει για πιθανό νέο *high score* και ότι το παιχνίδι ολοκληρώθηκε.
- **Bullet.prototype.recycle=function()**: Μέθοδος του αντικειμένου *bullet*, που όταν καλείται ανακυκλώνει μία σφαίρα μέσα στον πίνακα αντικειμένων τύπου *bullet*.

Explosion object :

Δημιουργία αντικειμένου της έκρηξης που ζωγραφίζεται με την πρόσκρουση της σφαίρας σε ένα φύλλο, με απόδοση των *attributes* από τα οποία αποτελείται. Αυτά είναι:

- οι συντεταγμένες της εικόνας της (x,y) στο *sprite* αρχείο
- το πλάτος και το ύψος της
- οι συντεταγμένες (x,y) του καμβά στις οποίες θα εμφανιστεί με την έναρξη του παιχνιδιού
- η διάρκεια (σε *frames*) που θα εμφανίζεται το γραφικό έκρηξης

Explosion functions :

- **Explosion.prototype.draw=function ()** : Μέθοδος του αντικειμένου *explosion*, που ζωγραφίζει το γραφικό της έκρηξης, κάθε φορά που γίνεται σύγκρουση μίας σφαίρας με ένα φύλλο, στις συντεταγμένες του σημείου στο οποίο έγινε η σύγκρουση.
- **Explosion.prototype.recycle=function()**:Μέθοδος του αντικειμένου *explosion*,για την ανακύκλωση των γραφικών της έκρηξης μέσα στον πίνακα αντικειμένων τύπου *explosion*.

Impact object :

Δημιουργία αντικειμένου της έκρηξης που ζωγραφίζεται με την πρόσκρουση του αεροσκάφους σε ένα φύλλο, με απόδοση των *attributes* από τα οποία αποτελείται. Αυτά είναι:

- οι συντεταγμένες της εικόνας της (x,y) στο *sprite* αρχείο
- το πλάτος και το ύψος της
- οι συντεταγμένες (x,y) του καμβά στις οποίες θα εμφανιστεί με την έναρξη του παιχνιδιού
- η διάρκεια (σε *frames*) που θα εμφανίζεται το γραφικό έκρηξης

Impact functions :

- **Impact.prototype.draw = function()**:Μέθοδος του αντικειμένου *impact*, που ζωγραφίζει το γραφικό της έκρηξης, κάθε φορά που γίνεται πρόσκρουση του αεροσκάφους με ένα φύλλο, στις συντεταγμένες του σημείου στο οποίο έγινε η πρόσκρουση.

Card object :

Δημιουργία αντικειμένου ενός φύλλου της τράπουλας, με απόδοση των *attributes* από τα οποία αποτελείται. Αυτά είναι:

- οι συντεταγμένες της εικόνας του (x,y) στο *sprite* αρχείο
- η ταχύτητα με την οποία κινείται το φύλλο
- το όνομά του
- η αξία του
- το πλάτος και το ύψος του
- οι συντεταγμένες (x,y) του καμβά στις οποίες θα εμφανιστεί με την έναρξη του παιχνιδιού

Επειδή, το κάθε φύλλο θα ζωγραφίζεται σε τυχαία θέση στην οθόνη, γίνεται χρήση της έτοιμης συνάρτησης : **Math.random()** η οποία όταν καλείται επιστρέφει έναν τυχαίο δεκαδικό αριθμό στο διάστημα 0 έως 1. Ο αριθμός αυτός θα πολλαπλασιαστεί με συγκεκριμένες τιμές για να εμφανίζονται τα φύλλα μέσα στα επιτρεπτά όρια του κεντρικού παραθύρου παιχνιδιού και στρογγυλοποιείται στο κάτω του όριο με την κλήση μιας άλλης έτοιμης συνάρτησης, της: **Math.floor()** .

Card functions :

- **Card.prototype.draw=function()**: Μέθοδος του αντικειμένου *card*, που ζωγραφίζει ένα τυχαίο φύλλο από την τράπουλα σε τυχαία θέση στην οθόνη του παιχνιδιού. Κατά τη μετάβαση από το ένα επίπεδο δυσκολίας στο άλλο, αυξάνεται η ταχύτητα του φύλλου, ανάλογα με το επίπεδο.
- **Card.prototype.checkEscaped=function()**: Μέθοδος του αντικειμένου *card*, που ελέγχει αν κάποιο φύλλο φτάσει και προσπεράσει το αριστερό άκρο της κύριας οθόνης παιχνιδιού. Στην περίπτωση αυτή το φύλλο απομακρύνεται από τον πίνακα αντικειμένων τύπου *cards*, και στη θέση του δημιουργείται ένα νέο τυχαίο φύλλο σε νέα τυχαία θέση.
- **Card.prototype.recycleCard=function()**: Μέθοδος του αντικειμένου *card*, που όταν καλείται παράγει ένα νέο τυχαίο φύλλο σε τυχαία θέση, με τη χρήση των έτοιμων συναρτήσεων που αναφέρθηκαν παραπάνω και το τοποθετεί στον πίνακα αντικειμένων τύπου *cards*.

Button object :

Δημιουργία αντικειμένου για κάθε κουμπί που βρίσκεται στις οθόνες του παιχνιδιού, με απόδοση των *attributes* από τα οποία αποτελείται. Αυτά είναι:

- οι συντεταγμένες (x,y) του πάνω ορίου του κουμπιού στο *sprite* αρχείο
- οι συντεταγμένες (x,y) του κάτω ορίου του κουμπιού στο *sprite* αρχείο
- οι συντεταγμένες (x,y) του αριστερού ορίου του κουμπιού στο *sprite* αρχείο
- οι συντεταγμένες (x,y) του δεξιού ορίου του κουμπιού στο *sprite* αρχείο

Η περιοχή του καμβά μέσα σε αυτά τα όρια αποτελεί ένα κουμπί στο οποίο ανατίθεται να εκτελέσει ένα συγκεκριμένο γεγονός.

Button functions :

- **Button.prototype.checkClicked =function():** Μέθοδος του αντικειμένου *card*, που ελέγχει αν έχει γίνει κλικ πάνω σε κάποιο κουμπί.
- **Button.prototype.checkMouseOver=function():** Μέθοδος του αντικειμένου *card*, που ελέγχει αν ο κέρσορας του ποντικιού βρίσκεται πάνω σε κάποιο κουμπί.

Στο σημείο αυτό θα παρουσιαστούν οι υπόλοιπες γενικές συναρτήσεις που δεν αποτελούν μεθόδους κάποιου συγκεκριμένου *object* και η κλήση τους μπορεί να γίνει απ' ευθείας σε οποιοδήποτε σημείο του κώδικα.

function clearCtxBackground(),

function clearCtx5card(),

function clearCtxBomber(),

function clearCtxCard() :

Συναρτήσεις για τον καθαρισμό του περιεχομένου κάθε καμβά με τη χρήση της έτοιμης μεθόδου **clearRect()**.

function init():

Η συνάρτηση αρχικοποίησης του παιχνιδιού, η οποία καλείται με το άνοιγμα του *web browser* ή με την επανέναρξη του παιχνιδιού ύστερα από τερματισμό/ολοκλήρωσή του.

function cardItem(name,value,altvalue):

Συνάρτηση για την ανάθεση ονόματος και αξίας για κάθε φύλλο της τράπουλας.

function createDeck():

Συνάρτηση για τη δημιουργία τράπουλας. Κάθε φορά δημιουργείται ένα φύλλο με τη χρήση του ονόματος και της αξίας του και εισάγεται σε μία λίστα. Η τελική λίστα που παράγεται αποτελεί και την τράπουλα με τα 52 φύλλα.

function moveBackground():

Συνάρτηση που προκαλεί την κίνηση του *background* σε κάθε στάδιο, στον κεντρικό καμβά του παιχνιδιού. Ζωγραφίζει το εκάστοτε *background*, χρησιμοποιώντας τις διαστάσεις του από το *sprite* αρχείο. Όταν φτάσει στις συντεταγμένες που αντιστοιχούν στο δεξί άκρο της εικόνας, η διαδικασία επαναλαμβάνεται, ζωγραφίζοντάς το από την αρχή, δηλ. ξεκινώντας από τις συντεταγμένες του αριστερού άκρου της εικόνας.

function spawn5cardsOnTable():

Συνάρτηση με την οποία παράγονται τα πέντε τυχαία προτεινόμενα φύλλα στο *table*, αριστερά του κεντρικού καμβά, με τη χρήση των έτοιμων συναρτήσεων **Math.floor()** και **Math.random()**, δημιουργώντας μια λίστα με πέντε αντικείμενα τύπου *cards*.

function drawOnTable():

Συνάρτηση που ζωγραφίζει στον καμβά του αριστερού *table*, τα πέντε προτεινόμενα φύλλα της λίστας που παράχθηκε προηγουμένως. Για να γίνει αυτό, γίνεται χρήση των συντεταγμένων και των διαστάσεων του κάθε φύλλου από το *sprite* αρχείο.

function spawnCard(num):

Συνάρτηση η οποία παράγει ένα συγκεκριμένο αριθμό *num* τυχαίων φύλλων από την τράπουλα, με τη χρήση των έτοιμων συναρτήσεων **Math.floor()** και **Math.random()**, δημιουργώντας μια λίστα με *num* πλήθος αντικειμένων τύπου *cards*. Η λίστα αυτή αποτελεί τα επερχόμενα φύλλα/εχθρούς και ανανεώνεται κάθε φορά που κάποιο φύλλο χτυπηθεί ή ξεπεράσει το αριστερό άκρο του κεντρικού καμβά του παιχνιδιού.

function drawAllCards():

Συνάρτηση που ζωγραφίζει τα φύλλα/εχθρούς που παράχθηκαν, σε τυχαία θέση στην οθόνη.

function manualScreen():

Συνάρτηση που ζωγραφίζει την οθόνη οδηγιών (*manual screen*) στον κεντρικό καμβά του παιχνιδιού. Με την κλήση της συνάρτησης αυτής, εισάγονται και *event listeners* στην οθόνη. Ο *event listener*, είναι ένας διαχειριστής γεγονότων που λαμβάνει κάποια ειδοποίηση όταν συμβεί ένα γεγονός συγκεκριμένου τύπου με τη χρήση της έτοιμης μεθόδου: **document.addEventListener(event, func, false)**. Δηλαδή, όταν συμβεί κάποιο γεγονός *event*, καλείται κάποια συνάρτηση *func*. Στη **manualScreen()**, εισάγονται *event listeners* για το κλικ του ποντικιού σε συγκεκριμένες συντεταγμένες και για όταν ο κέρσορας του ποντικιού βρίσκεται σε κάποια συγκεκριμένη θέση στην οθόνη οδηγιών. Έτσι, γίνεται ενεργοποίηση των αντικειμένων τύπου *button* (δηλ. των κουμπιών που έχουν δημιουργηθεί στην οθόνη οδηγιών).

function gameOverScreen():

Συνάρτηση που ζωγραφίζει την οθόνη τερματισμού του παιχνιδιού (*game over screen*).

Με την κλήση της εισάγονται και εδώ οι αντίστοιχοι *event listeners* για τα κουμπιά της οθόνης τερματισμού.

function drawMenu():

Συνάρτηση που ζωγραφίζει την κύρια οθόνη παιχνιδιού (*main menu screen*), χρησιμοποιώντας τις συντεταγμένες και διαστάσεις της αντίστοιχης εικόνας από το *sprite* αρχείο και της έτοιμης μεθόδου **drawImage()**. Η συνάρτηση αυτή καλείται μέσα στην **init()**.

function drawManual():

Συνάρτηση που ζωγραφίζει την οθόνη οδηγιών του παιχνιδιού (*manual screen*), χρησιμοποιώντας τις συντεταγμένες και διαστάσεις της αντίστοιχης εικόνας από το *sprite* αρχείο και της έτοιμης μεθόδου **drawImage()**. Η συνάρτηση αυτή καλείται μέσα στην **manualScreen()**.

function drawGameOverScreen():

Συνάρτηση που ζωγραφίζει την οθόνη τερματισμού του παιχνιδιού (*game over screen*), χρησιμοποιώντας τις συντεταγμένες και διαστάσεις της αντίστοιχης εικόνας από το *sprite* αρχείο και της έτοιμης μεθόδου **drawImage()**. Με την κλήση της συνάρτησης αυτής, αποκρύπτονται τα *in-play buttons* **Pause/Resume** και **Return to Menu** καθώς και ο καμβάς του *table* με τα προτεινόμενα φύλλα. Η συνάρτηση αυτή καλείται μέσα στην **gameOverScreen()**.

function drawBackground():

Συνάρτηση που ζωγραφίζει τα διαφορετικά *backgrounds* κάθε σταδίου του παιχνιδιού. Με την κλήση της, καθαρίζεται ο κύριος καμβάς του παιχνιδιού για να ζωγραφιστεί σε αυτόν το εκάστοτε *background*, χρησιμοποιώντας τις συντεταγμένες και διαστάσεις της αντίστοιχης εικόνας από το *sprite* αρχείο και της έτοιμης μεθόδου **drawImage()**. Επίσης, προκαλεί την έναρξη του μουσικού κομματιού που αντιστοιχεί στο συγκεκριμένο επίπεδο, σταματώντας την αναπαραγωγή του προηγούμενου.

function updateScore():

Συνάρτηση που ζωγραφίζει στον καμβά του σκορ, σε μορφή κειμένου, με τη χρήση της έτοιμης μεθόδου **fillText()**.

function updateLives():

Συνάρτηση που ζωγραφίζει στον καμβά των ζωών, σε μορφή κειμένου, με τη χρήση της έτοιμης μεθόδου **fillText()**.

function pauseButton():

Συνάρτηση που εμφανίζει το κουμπί **Pause** όταν το παιχνίδι βρίσκεται σε εξέλιξη. Όταν πατηθεί σε οποιοδήποτε επίπεδο του παιχνιδιού, το μουσικό κομμάτι που αναπαράγεται όπως και το παιχνίδι, σταματάνε και το κουμπί **Pause** αντικαθίσταται από το κουμπί **Resume**.

function resumeButton():

Συνάρτηση που εμφανίζει το κουμπί **Resume** όσο το παιχνίδι έχει διακοπεί. Όταν πατηθεί, το παιχνίδι συνεχίζει από το σημείο που σταμάτησε, το μουσικό κομμάτι συνεχίζει την αναπαραγωγή του και το κουμπί **Resume** αντικαθίσταται από το κουμπί **Pause**.

function playGame():

Κύρια συνάρτηση που καλείται όταν πατηθεί το κουμπί **Play Game** (είτε από *main menu*, είτε από *manual screen*) και προκαλεί την έναρξη του παιχνιδιού. Με την κλήση της, εμφανίζεται ο καμβάς του *table* με τα πέντε προτεινόμενα φύλλα και τα κουμπιά **Pause/Resume** και **Return to Menu**, ενώ εισάγονται και οι κατάλληλοι *event listeners* που διαχειρίζονται το γεγονός που θα συμβεί όταν κάποιο συγκεκριμένο πλήκτρο έχει πατηθεί ή όχι.

function loop():

Επαναληπτική συνάρτηση που καλείται συνέχεια, καθ' όλη τη διάρκεια του παιχνιδιού (όσο το παιχνίδι παίζεται και δεν έχει πατηθεί το κουμπί **Pause**). Ανανεώνει και ζωγραφίζει συνεχώς τους καμβάδες από τους οποίους αποτελείται το παιχνίδι. Η έναρξη της **loop()** γίνεται με τη συνάρτηση **startLoop()**, η οποία καλείται στην **playGame()**, ενώ με την **stopLoop()** σταματάει (είτε λόγω τερματισμού του παιχνιδιού, είτε λόγω επιλογής **Pause**).

function removeObjOnGameOver():

Συνάρτηση για την απομάκρυνση των αντικειμένων και τον καθαρισμό όλων των καμβάδων του παιχνιδιού, πριν την εμφάνιση της οθόνης τερματισμού.

function mouseOver(e):

Συνάρτηση διαχείρισης κάποιου γεγονότος *e*. Με την κλήση της ελέγχεται αν ο κέρσορας του ποντικιού βρίσκεται πάνω από κάποιο κουμπί στην κύρια οθόνη παιχνιδιού (*main menu screen*). Αν ναι, τότε ο κέρσορας αλλάζει μορφή, υποδεικνύοντας ότι στο συγκεκριμένο σημείο της οθόνης υπάρχει αντικείμενο κουμπιού. Η αλλαγή του κέρσορα γίνεται με την έτοιμη μέθοδο **document.body.style.cursor = 'crosshair'**. Αν ο κέρσορας απομακρυνθεί από τα όρια που συνθέτουν ένα κουμπί, τότε επιστρέφει στην αρχική του μορφή (*default*).

function mouseClicked(e):

Συνάρτηση διαχείρισης κάποιου γεγονότος *e*. Με την κλήση της ελέγχεται αν ο κέρσορας του ποντικιού έχει πατηθεί πάνω σε κάποιο κουμπί στην κύρια οθόνη παιχνιδιού (*main menu screen*). Αν ναι, τότε για κάθε κουμπί εκτελούνται οι κατάλληλες ενέργειες.

function mouseOverManualScreen(e):

Συνάρτηση διαχείρισης κάποιου γεγονότος *e*. Με την κλήση της ελέγχεται αν ο κέρσορας του ποντικιού βρίσκεται πάνω από κάποιο κουμπί στην οθόνη οδηγιών του παιχνιδιού (*manual screen*). Αν ναι, τότε ο κέρσορας αλλάζει μορφή, υποδεικνύοντας ότι στο συγκεκριμένο σημείο της οθόνης υπάρχει αντικείμενο κουμπιού. Η αλλαγή του κέρσορα γίνεται με την έτοιμη μέθοδο `document.body.style.cursor = 'crosshair'`. Αν ο κέρσορας απομακρυνθεί από τα όρια που συνθέτουν ένα κουμπί, τότε επιστρέφει στην αρχική του μορφή (*default*).

function mouseClickedManualScreen(e):

Συνάρτηση διαχείρισης κάποιου γεγονότος *e*. Με την κλήση της ελέγχεται αν ο κέρσορας του ποντικιού έχει πατηθεί πάνω σε κάποιο κουμπί στην οθόνη οδηγιών του παιχνιδιού (*manual screen*). Αν ναι, τότε για κάθε κουμπί εκτελούνται οι κατάλληλες ενέργειες.

function mouseOverGameOverScreen(e):

Συνάρτηση διαχείρισης κάποιου γεγονότος *e*. Με την κλήση της ελέγχεται αν ο κέρσορας του ποντικιού βρίσκεται πάνω από κάποιο κουμπί στην οθόνη τερματισμού του παιχνιδιού (*game over screen*). Αν ναι, τότε ο κέρσορας αλλάζει μορφή, υποδεικνύοντας ότι στο συγκεκριμένο σημείο της οθόνης υπάρχει αντικείμενο κουμπιού. Η αλλαγή του κέρσορα γίνεται με την έτοιμη μέθοδο `document.body.style.cursor = 'crosshair'`. Αν ο κέρσορας απομακρυνθεί από τα όρια που συνθέτουν ένα κουμπί, τότε επιστρέφει στην αρχική του μορφή (*default*).

function mouseClickedGameOverScreen(e):

Συνάρτηση διαχείρισης κάποιου γεγονότος *e*. Με την κλήση της ελέγχεται αν ο κέρσορας του ποντικιού έχει πατηθεί πάνω σε κάποιο κουμπί στην οθόνη τερματισμού του παιχνιδιού (*game over screen*). Αν ναι, τότε για κάθε κουμπί εκτελούνται οι κατάλληλες ενέργειες.

function checkKeyDown(e):

Συνάρτηση διαχείρισης κάποιου γεγονότος *e*. Με την κλήση της ελέγχεται αν κάποιο από τα πλήκτρα κίνησης του ήρωα έχει πατηθεί (←,↑,→,↓ ή **A,W,D,S**) ή αν έχει πατηθεί το πλήκτρο πυροβολισμού **Space Bar**. Αυτό γίνεται με τον έλεγχο των αναγνωριστικών (*key ID*) που έχει κάθε πλήκτρο του πληκτρολογίου. Αν πατηθεί κάποιο πλήκτρο κίνησης, τότε εκτελείται το αντίστοιχο σενάριο ενεργειών και το αεροσκάφος μετακινείται προς τη συγκεκριμένη κατεύθυνση. Αν πατηθεί το πλήκτρο πυροβολισμού, ένα αντικείμενο τύπου *bullet* φεύγει από το αεροσκάφος με την αναπαραγωγή του αντίστοιχου ηχητικού εφέ.

function checkKeyUp(e):

Συνάρτηση διαχείρισης κάποιου γεγονότος *e*. Η συνάρτηση αυτή ελέγχει αν ένα πλήκτρο είχε πατηθεί και «τώρα» έχει απελευθερωθεί (*key released*), ούτως ώστε να διακόπτει τα αντίστοιχα σενάρια ενεργειών σε τυχόν απελευθέρωσή του.

6

ΕΠΙΛΟΓΟΣ

Το **Ace Of Spades** είναι ένα παιχνίδι το οποίο ακολουθεί το σύνολο των βασικών γνωρισμάτων που χαρακτηρίζουν τα *arcade games* (δύο διαστάσεων, απλό στη χρήση, ραγδαία κλιμάκωση δυσκολίας) και δημιουργήθηκε στα πλαίσια της παρούσας διπλωματικής εργασίας. Οι μέθοδοι σχεδίασης και υλοποίησης του παιχνιδιού που ακολουθήθηκαν, τα οπτικοακουστικά αντικείμενα, τα βοηθητικά προγράμματα επεξεργασίας εικόνας και ήχου, όπως και η γλώσσα προγραμματισμού που χρησιμοποιήθηκε, είναι προσωπική επιλογή του σχεδιαστή/προγραμματιστή.

Βιβλιογραφία

- [1] David Flanagan: “JavaScript: The Definitive Guide”. Second Edition, January 1997
- [2] Jeremy Keith: “HTML5 for Web Designers (A Book Apart #1)”. First Edition, June 2010.
- [3] Robin Hawkes: “NY HTML5 – Game Development with HTML5 & JavaScript”. Digital Tinkerer at Rawkes on Mar 28, 2012.
- [4] Todd Anglin: “Five Benefits of Developing JavaScript Sites & Apps”. August 9, 2011.
- [5] <http://www.w3schools.com/>, W3schools web page for HTML and JavaScript tutorials.
- [6] <http://www.webestools.com>, Webestools web page for HTML and JavaScript online tools.
- [7] <http://www.jscripters.com/>, JSRIPTERS web page.
- [8] http://en.wikipedia.org/wiki/Main_Page, Wikipedia web page.
- [9] <http://onlinehtmltools.com/button-genaretor/>, Online button generator.
- [10] www.wall3.com/night-city-wallpaper-283-hd-wallpapers-background-in-abstract.html/,
image used for the first stage of the game.
- [11] <http://www.socwall.com/desktop-wallpaper/38494/lightning-in-contrast/> ,
image used for the second stage of the game.
- [12] <http://dcantoni.deviantart.com/art/Perth-City-Nightscape-IV-161138406/>,
image used for the third stage of the game.

[13] <http://wallpho.com/13751-new-york-city-at-night-2.htm#.Unj6DeJQj4w/>,

image used for the fourth stage of the game.

[14]

http://www.artwallpaperhi.com/Paintings/D_Wallpaper/ace_of_spades_1920x1080_wallpaper_42620/download_2560x1600/ , image used for the Main Menu Screen of the game.

[15] <http://exoshidaefanfic.files.wordpress.com/2013/07/game-over.jpg>,

image used for the Game Over Screen.

[16]<http://onlyhdwallpapers.com/music-and-dance/textures-playing-cards-ace-of-spades-desktop-hd-wallpaper-395679/> ,

Image used for the Manual Screen of the game.

[17] <http://www.dheinemann.com/assets/leopold/spacebar.jpg>,

image showing a Space Bar, inserted in Manual Screen of the game.

[18]http://lh3.ggpht.com/-RnbII_Y5f60/UblurdgE6QI/AAAAAAAAACZs/Vn34AdEY--0/Navigation-keys_thumb%25255B3%25255D.png?imgmax=800,

image showing the navigation keys, inserted in Manual Screen of the game.

[19] http://www.talkingproud.us/Military/Military/AirborneRecce_files/sovieta20bomber.png,

image of the Aircraft used in the game.

[20] http://www.imarvintpa.com/Mapping/Overlays/Effects/Fire/Fireball/explosion_4.png,

image of the Impact effect used in the game.

[21]

https://github.com/rjraffa/Overpass/diff_blob/09d4e10d3216c0ab32dae0082067a133ce7bd371/app/assets/images/Explosion.png?raw=true,

image of the Explosion effect used in the game.

[22] http://cdn.overclock.net/e/e1/350x700px-LL-e15b8e52_red-bluebullet1920x1080.png,

image of the Bullet firing used in the game.

- [23] <http://i.stack.imgur.com/X2GO2.png>, image of the Deck of Cards used in the game.
- [24] http://www.wallpaper4me.com/images/wallpapers/aces_of_spades_w1.jpeg, image used as the Web Browser's main background.
- [25] <http://soundbible.com/tags-boom.html>, Soundbible web page, for the sound effects used in the game .
- [26] <http://www.youtube.com/watch?v=9SdVsNqP6Oo>, Christopher Ashmore – Game Over song.
- [27] <http://www.youtube.com/watch?v=ZCpLH93Vxrc>, Skid Row – Youth Gone Wild song .
- [28] <http://www.youtube.com/watch?v=KgJ1t2S7O6M>, Scorpions – Big City Nights song .
- [29] <http://www.youtube.com/watch?v=temDHv1e5Io>, Motörhead – The Chase Is Better Than The Catch song .
- [30] <http://www.youtube.com/watch?v=vcf7DnHi54g>, Motörhead – Ace Of Spades song.
- [31] http://www.youtube.com/watch?v=5BSnEscC_V8, Motörhead – Whorehouse Blues song.
- [32] <http://www.getpaint.net/>, Paint.NET home page.
- [33] <http://www.photoshop.com/>, PhotoShop home page.
- [34] <http://www.virtualdj.com/>, Virtual DJ home page.

ΠΑΡΑΡΤΗΜΑ Α

