



Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης
Σχολή Τεχνολογικών Εφαρμογών
Τμήμα Μηχανολογίας

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

**ΜΕΘΟΔΟΛΟΓΙΕΣ ΠΡΟΒΟΛΗΣ ΤΡΙΣΔΙΑΣΤΑΤΩΝ ΓΡΑΦΙΚΩΝ ΜΕ
ΔΥΝΑΤΟΤΗΤΑ ΚΙΝΗΣΗΣ ΣΤΟ ΔΙΑΔΥΚΤΙΟ**

ΣΠΟΥΔΑΣΤΗΣ: ΧΡΥΣΑΚΗΣ ΙΩΑΝΝΗΣ ΑΜ:3646

ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ: ΔΡ. ΠΕΤΟΥΣΗΣ ΜΑΡΚΟΣ

ΕΥΧΑΡΙΣΤΙΕΣ

Θα ήθελα να ευχαριστήσω τον εισηγητή καθηγητή Δρ. Πετούση Μάρκο για την καθοδήγηση του στην εργασία και την οικογένεια μου για την στήριξή τους στην προσπάθεια μου.

Περιεχόμενα

ΕΙΣΑΓΩΓΗ	5
1 ΑΠΕΙΚΟΝΙΣΗ ΤΡΙΣΔΙΑΣΤΑΤΩΝ ΓΡΑΦΙΚΩΝ ΣΤΟΝ ΥΠΟΛΟΓΙΣΤΗ	6
2 ΠΕΡΙΓΡΑΦΗ ΠΡΟΒΟΛΗΣ ΓΡΑΦΙΚΩΝ	8
2.1 Τρισδιάστατα γραφικά στο διαδίκτυο.	8
2.2 Κινούμενα γραφικά στο διαδίκτυο.	9
3 ΧΡΗΣΙΜΟΠΟΙΟΥΜΕΝΑ ΛΟΓΙΣΜΙΚΑ ΚΑΙ ΕΦΑΡΜΟΓΕΣ	9
3.1 ΠΕΡΙΓΡΑΦΗ ΤΩΝ ΛΟΓΙΣΜΙΚΩΝ	10
3.1.1 AUTODESK MAYA	10
3.1.2 OKINO COMPUTER GRAPHICS-POLYTRANS	11
3.1.3 HEADUS UVLAYOUT-ROADKILL	12
3.1.4 PROENGINEER	12
3.1.5 SWIFT 3D	13
3.1.6 ADOBE FLASH PLAYER	14
3.1.7 ADOBE FLASH ILLUSTRATOR	17
3.1.8 COSMOPLAYER-CORTONA PLAYER	17
3.2 Η ΠΡΑΚΤΙΚΗ ΕΦΑΡΜΟΓΗ ΤΩΝ ΛΟΓΙΣΜΙΚΩΝ	19
3.2.1 AUTODESK MAYA	19
3.2.2 OKINO COMPUTER GRAPHICS-POLYTRANS	20
3.2.3 HEADUS UVLAYOUT-ROADKILL	21
3.2.4 PROENGINEER	22
3.2.5 SWIFT 3D	24
3.2.6 ADOBE FLASH PLAYER	25
3.2.7 ADOBE FLASH ILLUSTRATOR	26
3.2.8 COSMOPLAYER-CORTONA PLAYER	27
4 ΑΝΑΛΥΤΙΚΗ ΠΕΡΙΓΡΑΦΗ ΤΩΝ ΕΦΑΡΜΟΓΩΝ ΣΤΑ ΛΟΓΙΣΜΙΚΑ	28
4.1 Εφαρμογές σε Cortona	29
4.1.1 Κινούμενη σφαίρα τριών διαστάσεων που εκτελεί διαδρομή $A \rightarrow B \rightarrow A$ με σταθερή ταχύτητα και περιστροφή, με το πάτημα ενός κουμπιού	29
4.1.2 Δύο κινούμενες σφαίρες που συγκρούονται και επιστρέφουν στις αρχικές τους θέσεις με το πάτημα ενός κουμπιού.	38
4.2 Εφαρμογές σε προγράμματα Adobe	43
4.2.1 Ρολόι δύο διαστάσεων με δείκτες ταλάντωσης ανά δευτερόλεπτο	43
4.2.2 Ρολόι δύο διαστάσεων με μηχανικούς δείκτες	55
4.2.3 Ρολόι Rolex τρισδιάστατης γεωμετρίας δισδιάστατης απεικόνισης με μηχανικούς δείκτες.	63

4.3	Εφαρμογή σε Autodesk Maya	64
5	ΣΥΜΠΕΡΑΣΜΑΤΑ-ΠΕΡΙΟΡΙΣΜΟΙ-ΠΡΟΒΛΗΜΑΤΑ	92
6	ΕΠΙΛΟΓΟΣ	98
7	ΒΙΒΛΙΟΓΡΑΦΙΑ	99

ΕΙΣΑΓΩΓΗ

Η απεικόνιση κινούμενων τρισδιάστατων γραφικών στον ηλεκτρονικό υπολογιστή είναι πλέον πολύ διαδεδομένη σε πλήθος εφαρμογών στη μηχανολογία, στο διαδίκτυο και αλλού. Για τη δημιουργία των απεικονίσεων αυτών, αρχικά απαιτείται η ανάπτυξη της τρισδιάστατης γεωμετρίας. Η ανάπτυξη της γεωμετρίας γίνεται με τον ορισμό στο κατάλληλο λογισμικό, στον ηλεκτρονικό υπολογιστή σχημάτων. Στη συνέχεια, με τη χρήση κατάλληλων μεθοδολογιών και αλγορίθμων, επιτυγχάνεται η προσομοίωση της κίνησης τους στο χώρο για την απεικόνιση διαδικασιών του πραγματικού κόσμου στον ηλεκτρονικό υπολογιστή, όπως είναι η λειτουργία ενός μηχανισμού.

Σκοπός της παρούσης εργασίας είναι η διερεύνηση των μεθοδολογιών και των αλγορίθμων που έχουν αναπτυχθεί για την προβολή τρισδιάστατων γραφικών με δυνατότητα κίνησης στο διαδίκτυο. Θα περιγραφούν οι μεθοδολογίες αυτές και τα λογισμικά εργαλεία που χρησιμοποιούνται στον ηλεκτρονικό υπολογιστή (opengl, directx, vml, κλπ.). Θα γίνει επίσης περιγραφή των λογισμικών εργαλείων που χρησιμοποιούνται για την απεικόνιση κινούμενων τρισδιάστατων γραφικών σε διαδικτυακές εφαρμογές (macromedia flash, κλπ.).

Για την επίδειξη των δυνατοτήτων των μεθοδολογιών που θα περιγραφούν, θα αναπτυχθεί ένα παράδειγμα εφαρμογής σε κάποιο από τα διαθέσιμα λογισμικά εργαλεία.

Επιγραμματικά τα βήματα για την επιτυχή ολοκλήρωση της πτυχιακής εργασίας από το σπουδαστή είναι:

Αναζήτηση στη βιβλιογραφία πληροφοριών για τις μεθοδολογίες απεικόνισης τρισδιάστατων γραφικών στο διαδίκτυο.

Αναζήτηση στη βιβλιογραφία των λογισμικών εργαλείων που χρησιμοποιούνται για την απεικόνιση κινούμενων τρισδιάστατων γραφικών στον ηλεκτρονικό υπολογιστή.

Περιγραφή των μεθοδολογιών και των λογισμικών εργαλείων.

Ανάπτυξη παραδείγματος εφαρμογής με τη χρήση κατάλληλου λογισμικού εργαλείου.
Συγγραφή και παρουσίαση της πτυχιακής εργασίας.

1 ΑΠΕΙΚΟΝΙΣΗ ΤΡΙΣΔΙΑΣΤΑΤΩΝ ΓΡΑΦΙΚΩΝ ΣΤΟΝ ΥΠΟΛΟΓΙΣΤΗ

Στη σημερινή εποχή η εξέλιξη είναι συνάρτηση της τεχνολογικής προόδου και οι ανάγκες του ανθρώπου σε κάθε τομέα ξεπερνούν μια απλώς “δισδιάστατη” οπτική για τη ζωή. Κατά συνέπεια, η αντικατάσταση των γραφικών δύο διαστάσεων από τα γραφικά τριών διαστάσεων (Εικόνα 1) ήταν δεδομένη και αναγκαία, αφού προσφέρουν φυσική αίσθηση του βάθους ως συστατικό της ανθρώπινης όρασης. Τα τρισδιάστατα γραφικά αναλυτικότερα, συνιστούν ένα σύνολο κατασκευασμένων εικόνων, με την βοήθεια των οποίων αναπαρίστανται αντικείμενα σε τρεις διαστάσεις.



Εικόνα 1-Τρισδιάστατα γραφικά

Κάθε εικόνα περιέχει 3 συντεταγμένες, οι οποίες αναπαριστούν ένα σημείο ή ένα διάνυσμα στον τρισδιάστατο χώρο. Με τον συνδυασμό πλήθους διανυσμάτων τα οποία αναπαρίστανται και υπό μορφή πινάκων, είναι δυνατόν να σχηματιστούν απλά αντικείμενα (κώνοι, κύβοι κτλ). Συνηθέστατα, γίνεται χρήση ενός μοντέλου πολυγώνων (κυρίως τριγώνων), σύμφωνα με το οποίο, η πλευρά κάθε πολυγώνου απεικονίζεται από ένα διάνυσμα ή πίνακα, ώστε να γίνει συνδυασμός των πολυγώνων και να σχηματιστεί μια εικόνα. Η ποιότητα της εικόνας, εξαρτάται από το μέγεθος και το πλήθος των πολυγώνων που την συνθέτουν. Με την ολοκλήρωση της γεωμετρίας του αντικειμένου, ακολουθεί ο φωτισμός και η δημιουργία σκιάσεων. Αυτό επιτυγχάνεται με την κάλυψη των πολυγώνων με κάποια υφή χρώματος και με τον καθορισμό της φωτεινότητας των πολυγώνων, ανάλογα με τη θέση της πηγής φωτός (Εικόνα 2).



Εικόνα 2-Τρισδιάστατα γραφικά

2 ΠΕΡΙΓΡΑΦΗ ΠΡΟΒΟΛΗΣ ΓΡΑΦΙΚΩΝ

Η προβολή τρισδιάστατων γραφικών και κινούμενων γραφικών στο διαδίκτυο γίνεται μέσω διαφόρων τεχνολογιών διαδικτυακής τρισδιάστατης απεικόνισης, πιο διαδεδομένες από τις οποίες είναι : α) Hoops Stream format β) Java 3D γ) VRML δ) X3D ε) Shockwave 3D. Για την περιγραφή της μεθοδολογίας μιας τρισδιάστατης διαδικτυακής προβολής, θα χρησιμοποιηθεί η τεχνολογία VRML.

2.1 Τρισδιάστατα γραφικά στο διαδίκτυο.

Ένα τρισδιάστατο αντικείμενο στο διαδίκτυο αποτελείται ουσιαστικά από μια σειρά κόμβων (Nodes), οι οποίοι συνιστούν τύπους αντικειμένων και από μια σειρά πεδίων (fields), τα οποία συνιστούν τις ιδιότητες των αντικειμένων. Κατά συνέπεια, η απόδοση σχήματος στο επιθυμητό σχέδιο επιτυγχάνεται με την δημιουργία ενός κόμβου **“Shape”** και των πεδίων **“appearance”** και **“geometry”**. Τα βασικά γεωμετρικά σχήματα που μπορούν να εισαχθούν στο πεδίο “geometry” με την μορφή κόμβου, είναι τα: a) “box”, “cone”, “cylinder” και “sphere”. Στην συνέχεια, αφού ολοκληρωθεί το σχήμα, ορίζονται η τοποθέτηση/μετατόπιση στον χώρο, εισάγοντας το πεδίο **“translation”** και η ρύθμιση του μεγέθους του σχήματος, εισάγοντας το πεδίο **“size”**, οι διαστάσεις του οποίου περιγράφονται από τους 3 άξονες συντεταγμένων (x,y,z). Η απόδοση χρώματος στο αντικείμενο, επιτυγχάνεται με την προσθήκη ενός επιπλέον κόμβου **“Material”** στον κόμβο “Appearance” και την εισαγωγή του πεδίου “diffuseColor” στον κόμβο “Material”. Σε περίπτωση απόδοσης υφής στο αντικείμενο, προστίθεται στον κόμβο “Appearance” το πεδίο Texture ImageTexture μέσω αποθηκευμένης εικόνας στον υπολογιστή (πχ.url“ball.jpg”).

2.2 Κινούμενα γραφικά στο διαδίκτυο.

Για την προβολή κινούμενων γραφικών στο διαδίκτυο, είναι απαραίτητη η εισαγωγή ενός μετρητή (**Timer**) και παρεμβολέων (**Interpolators**): θέσης (**PositionInterpolator**), προσανατολισμού (**OrientationInterpolator**), κλίμακας (**ScalarInterpolator**), χρώματος (**ColorInterpolator**), συντεταγμένων (**CoordinateInterpolator**). Η μεθοδολογία αυτής της προβολής περιγράφεται ως εξής: Ο μετρητής εισάγεται στο πρόγραμμα, για να συντονίζει τις χρονικές ενέργειες (κίνησης, ταχύτητας, περιστροφής, αλλαγής χρώματος, έντασης ήχου κλπ). Η δομή του είναι η εξής: DEF timer TimeSensor { cycleInterval **x** loop TRUE }, όπου **x** η χρονική διάρκεια σε δευτερόλεπτα, που θα διαρκέσει η κίνηση. Η βασικότερη λειτουργία του μετρητή (TimeSensor) είναι ότι παράγει **events** όσο περνάει ο χρόνος (sec). Τα events αυτά, στέλνονται στον εκάστοτε παρεμβολέα (κίνησης, προσανατολισμού, κλίμακας κτλ), ο οποίος με την σειρά του θα εξάγει αυτά τα events στον κόμβο Transform, ώστε να τροφοδοτηθεί το πεδίο του Translate το οποίο εν τέλει θα αλλάξει την θέση του αντικειμένου στον χώρο και θα προκύψει κίνηση στο τρισδιάστατο αντικείμενο.

3 ΧΡΗΣΙΜΟΠΟΙΟΥΜΕΝΑ ΛΟΓΙΣΜΙΚΑ ΚΑΙ ΕΦΑΡΜΟΓΕΣ

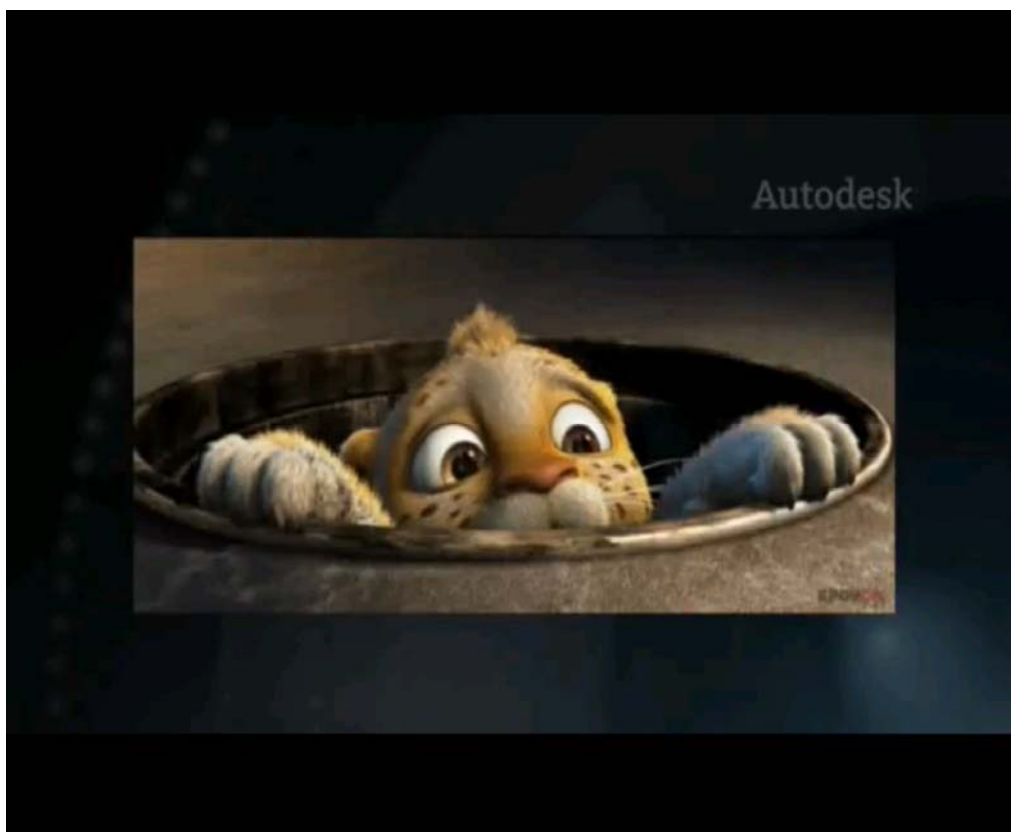
Για τις ανάγκες της εργασίας χρησιμοποιήθηκαν τα παρακάτω λογισμικά:

- 1) **AYTODESK MAYA**
- 2) **OKINO COMPUTER GRAPHICS-POLYTRANS**
- 3) **HEADUS UVLAYOUT-ROADKILL**
- 4) **PROENGINEER**
- 5) **SWIFT 3D**
- 6) **ADOBE FLASH PLAYER**
- 7) **ADOBE FLASH ILLUSTRATOR**
- 8) **COSMOPLAYER-CORTONA PLAYER**

3.1 ΠΕΡΙΓΡΑΦΗ ΤΩΝ ΛΟΓΙΣΜΙΚΩΝ

3.1.1 AUTODESK MAYA

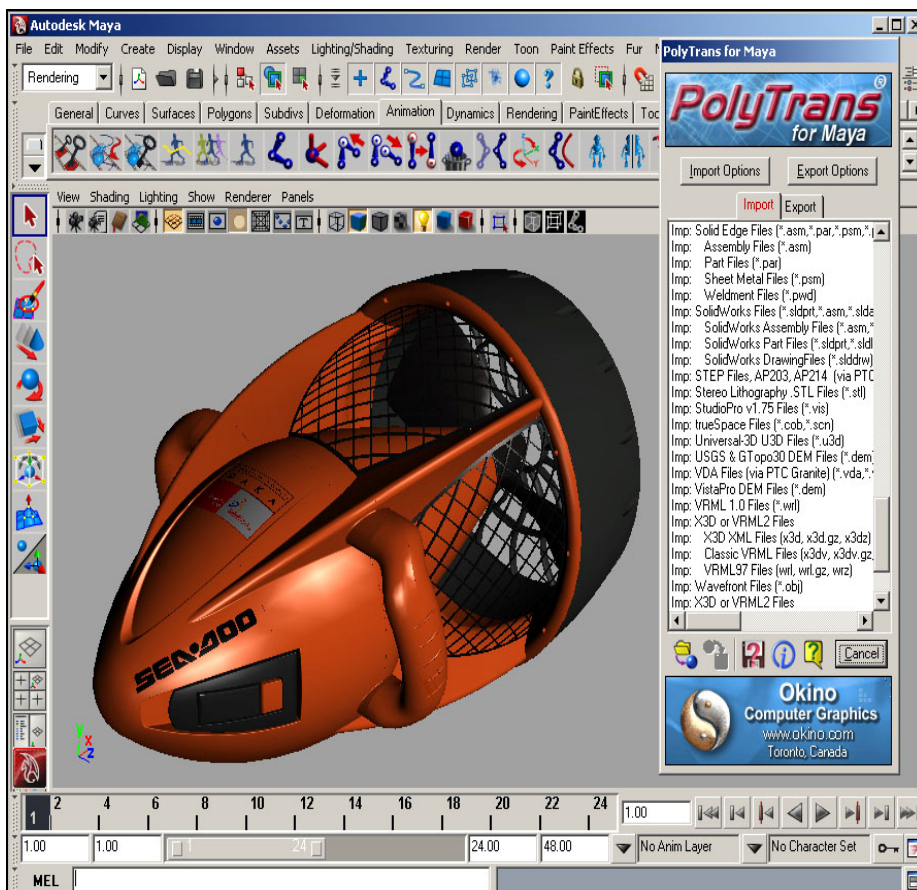
Είναι ένα λογισμικό τρισδιάστατων γραφικών για τον υπολογιστή, το οποίο χρησιμοποιείται για την δημιουργία διαδραστικών τρισδιάστατων εφαρμογών, συμπεριλαμβανομένων παιχνιδιών, βίντεο, ταινιών κινουμένων σχεδίων (Εικόνα 3), τηλεοπτικών σειρών ή οπτικών εφέ. Ο χρήστης ορίζει έναν εικονικό χώρο εργασίας (scene), όπου έχει την δυνατότητα να επεξεργαστεί και να τροποποιήσει τα μέσα/εφαρμογές ενός συγκεκριμένου σχεδίου. Τα σχέδια μπορούν να αποθηκευτούν σε διάφορες μορφές αρχείων με προεπιλογή το .mb (maya binary). Το Maya διαθέτει ένα σύστημα αρχιτεκτονικής κομβικού γραφήματος. Τα στοιχεία των χώρων εργασίας βασίζονται σε κόμβους καθένας από τους οποίους διαθέτει τα δικά του χαρακτηριστικά και ρυθμίσεις.



Εικόνα 3-Animation στο Autodesk Maya

3.1.2 OKINO COMPUTER GRAPHICS-POLYTRANS

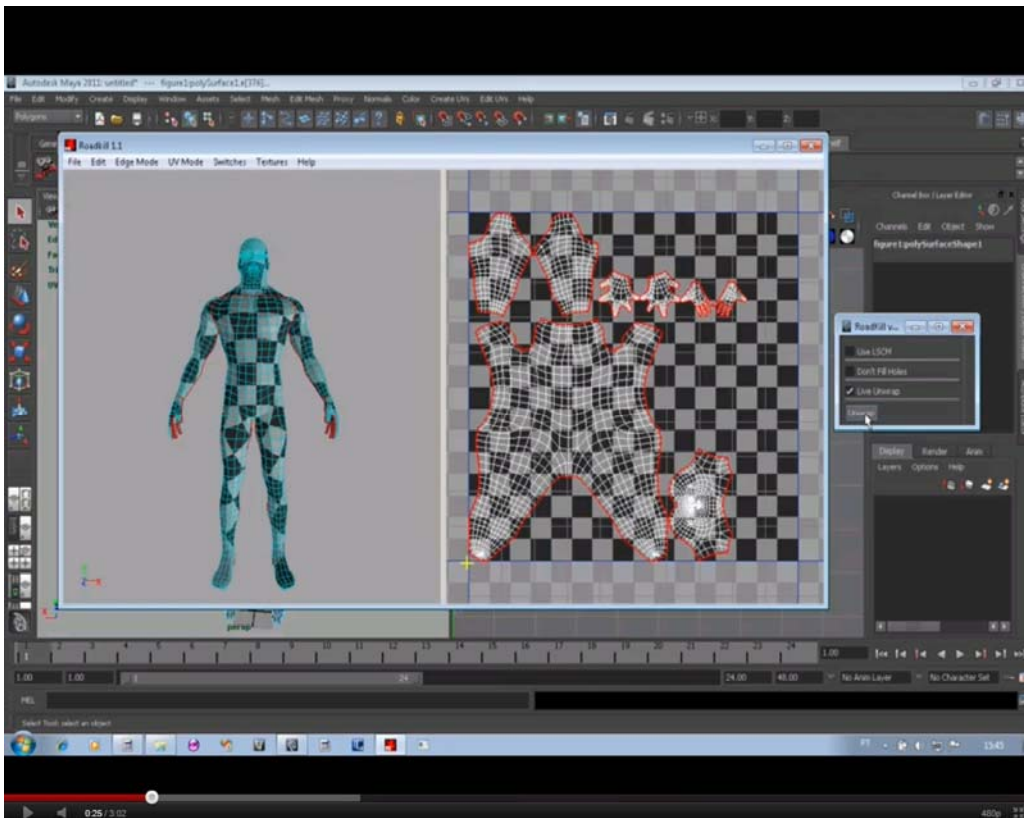
Τα polytrans είναι ένα εκτεταμένο και ακριβές λογισμικό τρισδιάστατων εφαρμογών, το οποίο διαθέτει προηγμένη Φώτο-ρεαλιστική μηχανή τρισδιάστατου rendering (Εικόνα 4), εργαλειοθήκη τρισδιάστατης γεωμετρίας και μηχανή μετατροπής animation, με σκοπό να πραγματοποιεί πολύπλοκες μεταφράσεις μεταξύ των πιο διαδεδομένων τρισδιάστατων τύπων αρχείων. Αποτελεί μια ουδέτερη γέφυρα μεταξύ όλων των κύριων εφαρμογών των λογισμικών όπως: 3ds Max, Maya, Lightwave, Softimage XSI, Cinema-4D, Collada, DirectX, FBX, U3D, XAML, X3D/VRML2 και άλλων.



Εικόνα 4-Τρισδιάστατο Rendering

3.1.3 HEADUS UVLAYOUT-ROADKILL

Αποτελούν αυτόνομα λογισμικά για δημιουργία και επεξεργασία των συντεταγμένων υφής (UVs) για τρισδιάστατα πλέγματα και επιφάνειες υποδιαίρεσης. Χρησιμοποιούνται σε εφαρμογές παιχνιδιών και σε βιομηχανίες οπτικών εφέ (Εικόνα 5). Προσφέρουν υψηλή ποιότητα στην υφή των μοντέλων, σε σημαντικά μικρότερο χρόνο σε σχέση με τις παραδοσιακές μεθόδους.

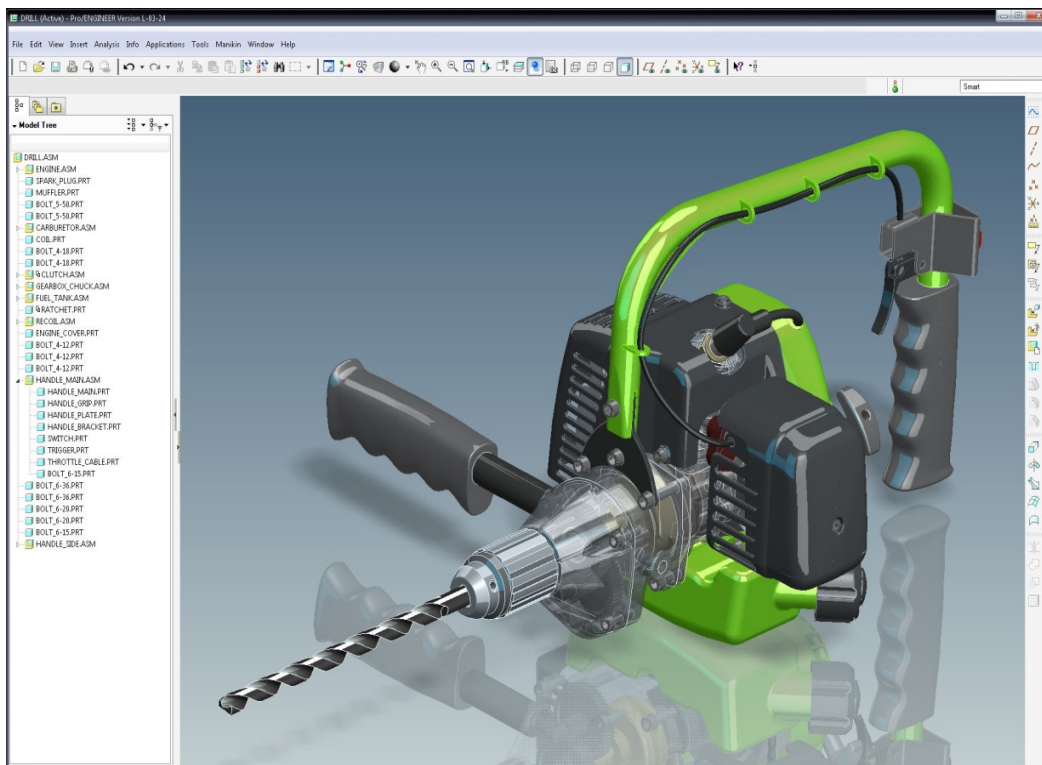


Εικόνα 5-Δημιουργία οπτικών εφέ

3.1.4 PROENGINEER

Το Pro / **ENGINEER** είναι ένα παραμετρικό, ενσωματωμένο, τρισδιάστατο CAD/CAM/CAE λογισμικό. Δημιουργήθηκε από την Εταιρεία Παραμετρικής Τεχνολογίας (Parametric Technology Corporation) η οποία ήταν η πρώτη στην αγορά που διέθετε παραμετρική, βασισμένη σε χαρακτηριστικά, συνειρμική, στερεά μοντελοποίηση. Το

λογισμικό έχει εφαρμογή σε πλατφόρμες Microsoft Windows και παρέχει στερεά μοντελοποίηση, μοντελοποίηση συναρμολόγησης καθώς και ανάλυση πεπερασμένων στοιχείων (Εικόνα 6).



Εικόνα 6-Rendering σε Pro-Engineer

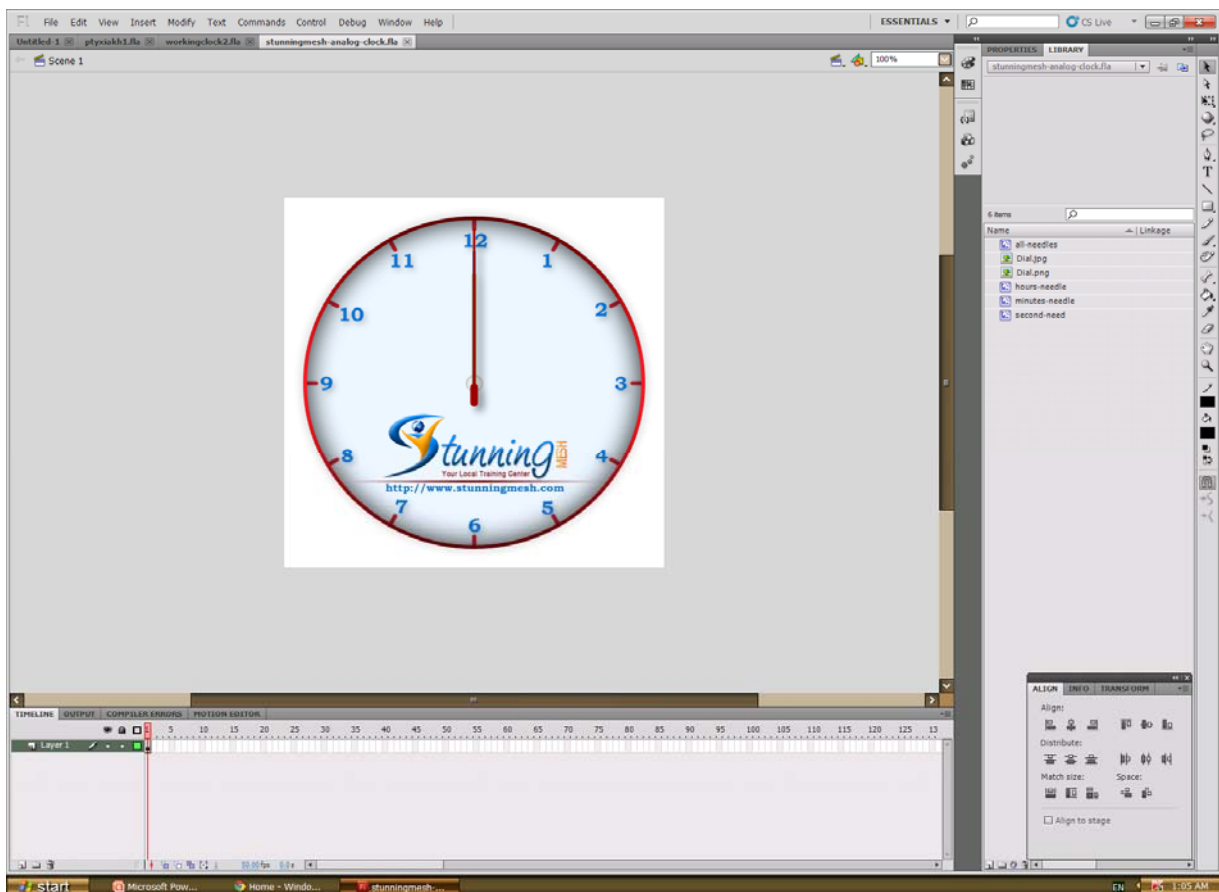
3.1.5 SWIFT 3D

Το SWIFT 3D είναι ένα λογισμικό που ανέπτυξε η εταιρεία electric rain, το οποίο επιτρέπει στον χρήστη να δημιουργεί ή να εισάγει μοντέλα τριών διαστάσεων, την κίνηση, τον χειρισμό και την εξαγωγή τους για χρήση σε εφαρμογές του adobe flash. Στις πιο πρόσφατες εκδόσεις του, μάλιστα, υποστηρίζει μια δέσμη νέων τύπων αρχείων τόσο για λογισμικά με δισδιάστατες όσο και τρισδιάστατες εφαρμογές:

Flash CS5 (.FLA και .XFL), Silverlight (WPF XAML), Papervision 3D v2, QuickTime (MOV, AVI, FLV) και πολλά άλλα.

3.1.6 ADOBE FLASH PLAYER

Είναι ένα λογισμικό, το οποίο παρέχει έναν ευρύ συνδυασμό τεχνολογιών για την δημιουργία, επεξεργασία και προβολή διανυσματικών γραφικών και animation με δυνατότητα εξαγωγής και χρήσης τους στο διαδίκτυο. Αρχικά, ήταν σχεδιασμένο για την υποστήριξη 2D animation (Εικόνα 7), (Εικόνα 8), αλλά πλέον στις τελευταίες εκδόσεις του επιδέχεται και εφαρμογές οι οποίες δίνουν την αίσθηση του 3D animation και διαθέτει πολύ ισχυρότερα εργαλεία με περισσότερα και πλουσιότερα χαρακτηριστικά. Χρησιμοποιεί διανυσματικά γραφικά για να ελαχιστοποιήσει το μέγεθος των αρχείων και δημιουργεί αρχεία εξοικονομώντας εύρος ζώνης και χρόνο φόρτωσης.



Εικόνα 7-2D animation

Δυνατότητες:

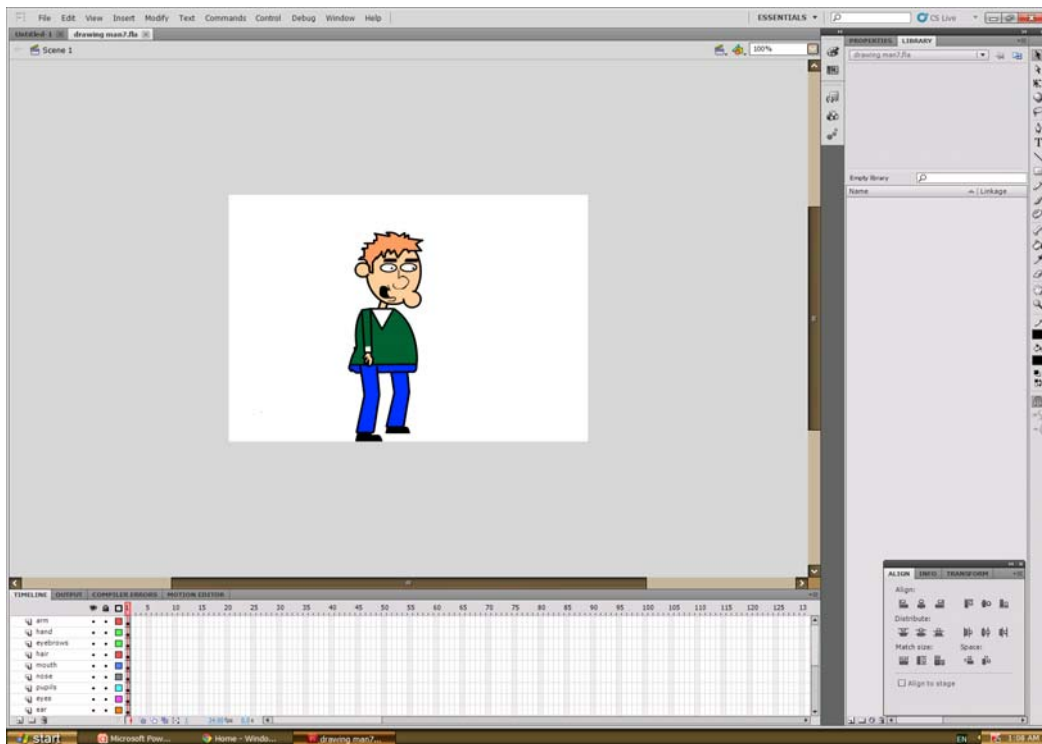
- Δημιουργία κινουμένων εικόνων, οι οποίες μπορούν να περιλαμβάνουν και στοιχεία πλοήγησης όπως κουμπιά και μενού.
- Δημιουργία αυτόνομων προβολών .
- Χρησιμοποίηση περισσότερων γραφικών και μεγαλύτερου έλεγχου αυτών, διαθέτοντας εργαλεία για την μετάδοση διανυσματικών εικόνων.

Εργαλεία και ορολογία:

- *Actionscript*: Είναι η γλώσσα προγραμματισμού που χρησιμοποιεί το flash για την δημιουργία αλληλεπιδράσεων και ειδικών εφέ.
- *Keyframe*: Είναι ένα καρέ το οποίο επισημαίνει μια αλλαγή στις ιδιότητες ενός αντικειμένου σε σχέση με το προηγούμενο καρέ. Το 1^ο καρέ του διαγράμματος ροής χρόνου πρέπει να είναι ένα καρέ κλειδί.
- *Layer*: Τα επίπεδα ή στρώσεις είναι σχήματα ομαδοποίησης και οργάνωσης των αντικειμένων με κατακόρυφη διάταξη, δηλαδή τοποθετούνται το ένα πάνω στο άλλο, στα οποία λαμβάνει χώρα το animation. Με τα επίπεδα μπορούμε να οργανώσουμε λογικά τα περιεχόμενα μιας ταινίας.
- *Library*: Είναι ένας μηχανισμός οργάνωσης του flash που μας δίνει την δυνατότητα να αποθηκεύσουμε, να διαχειριζόμαστε και να εμφανίζουμε σύμβολα στο flash καθώς και να εισάγουμε bitmap γραφικά, αρχεία ήχου και βίντεο.
- *Timeline*: Αντιπροσωπεύει τη χρονολογική σειρά των καρέ μιας ταινίας ή ενός μέρους της. Υπάρχει ένα διάγραμμα ροής χρόνου για κάθε επίπεδο, το οποίο διατηρείται σε καρέ, καθένα από τα οποία μπορεί να συσχετίζεται με μια ενέργεια ή ένα εφέ. Δηλαδή, μας επιτρέπει να δούμε τα περιεχόμενα ενώ εργαζόμαστε, μαζί με μια σειρά χρονικών τμημάτων.

- *Tweening*: Είναι μια διαδικασία η οποία παράγει την κίνηση μεταξύ δυο καρτέκλειδιών. Κάθε ενδιάμεσο καρέ μεταβάλλεται ελαφρά σε σχέση με το προηγούμενο του, πράγμα που επιταχύνει την ομαλή εξέλιξη της δράσης της ταινίας.

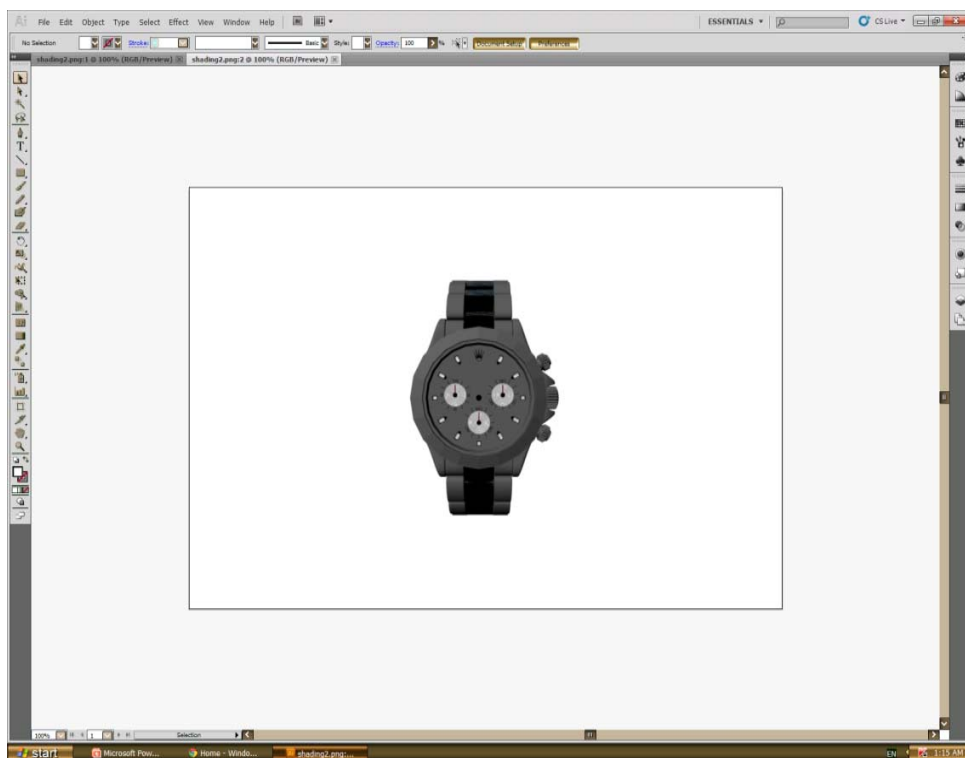
Τύπος αρχείων: Τα αρχεία που δημιουργούμε με το flash ονομάζονται ταινίες (movies) και έχουν την επέκταση *.fla* ενώ τα εκτελέσιμα αρχεία του flash, αυτά δηλαδή που θα εμφανιστούν ενσωματωμένα σε μια σελίδα στο internet ή θα μπορούν να τρέξουν σαν αυτόνομες εφαρμογές, έχουν την επέκταση *.swf* .



Εικόνα 8-2D animation

3.1.7 ADOBE FLASH ILLUSTRATOR

Το Adobe Illustrator είναι μια εφαρμογή γραφικών, που επιτρέπει στο χρήστη να δημιουργήσει πολύπλοκα και ελκυστικά σχέδια με εφέ τύπου, όπως σχήματα και λογότυπα, χρησιμοποιώντας διακριτικά διανυσματικά γραφικά και ακριβή εργαλεία σχεδίασης. Διαθέτει, τέλος, έναν συνδυασμό κειμένου και γραφικών για την δημιουργία διαφημίσεων (Εικόνα 9).

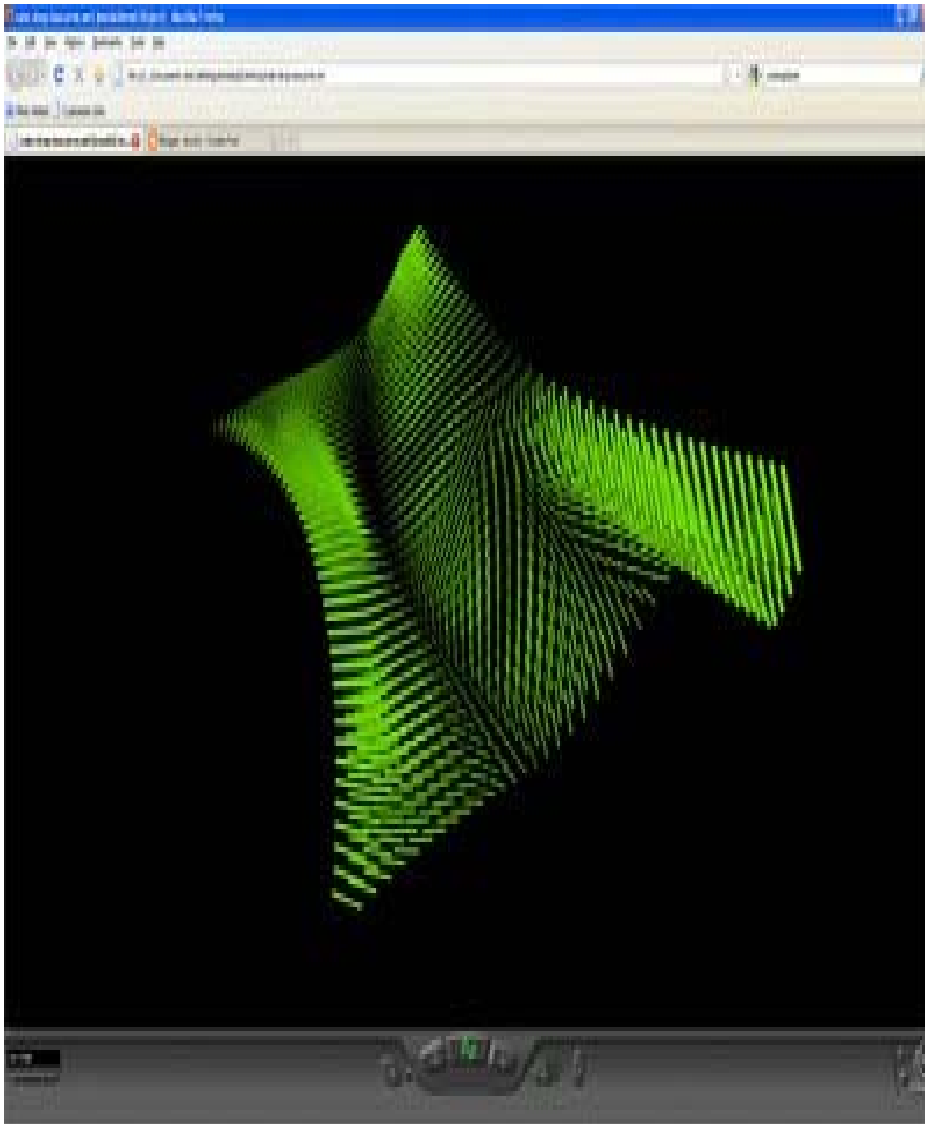


Εικόνα 9

3.1.8 COSMOPLAYER-CORTONA PLAYER

Τα προγράμματα Cosmo player και Cortona, αποτελούν δύο γρήγορα και διαδραστικά διαδικτυακά προγράμματα τρισδιάστατης απεικόνισης, ιδανικά για την προβολή

τρισδιάστατων μοντέλων στο διαδίκτυο (Εικόνα 10). Λειτουργούν ως VRML plug-ins για τα δημοφιλή προγράμματα περιήγησης του διαδικτύου (Internet Explorer, Netscape Browser, Mozilla, Firefox, Google Chrome κτλ) και ως εφαρμογές γραφείου (Microsoft PowerPoint, το Microsoft Word, κτλ).

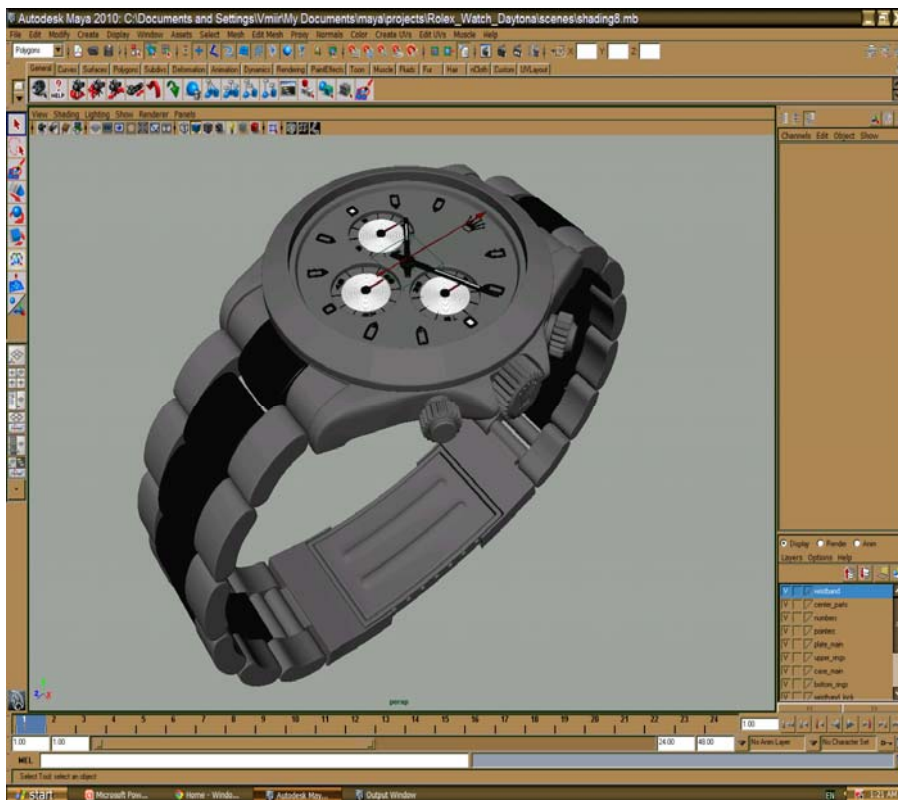


Εικόνα 10

3.2 Η ΠΡΑΚΤΙΚΗ ΕΦΑΡΜΟΓΗ ΤΩΝ ΛΟΓΙΣΜΙΚΩΝ

3.2.1 ΑΥΤΟΔΕΣΚ ΜΑΥΑ

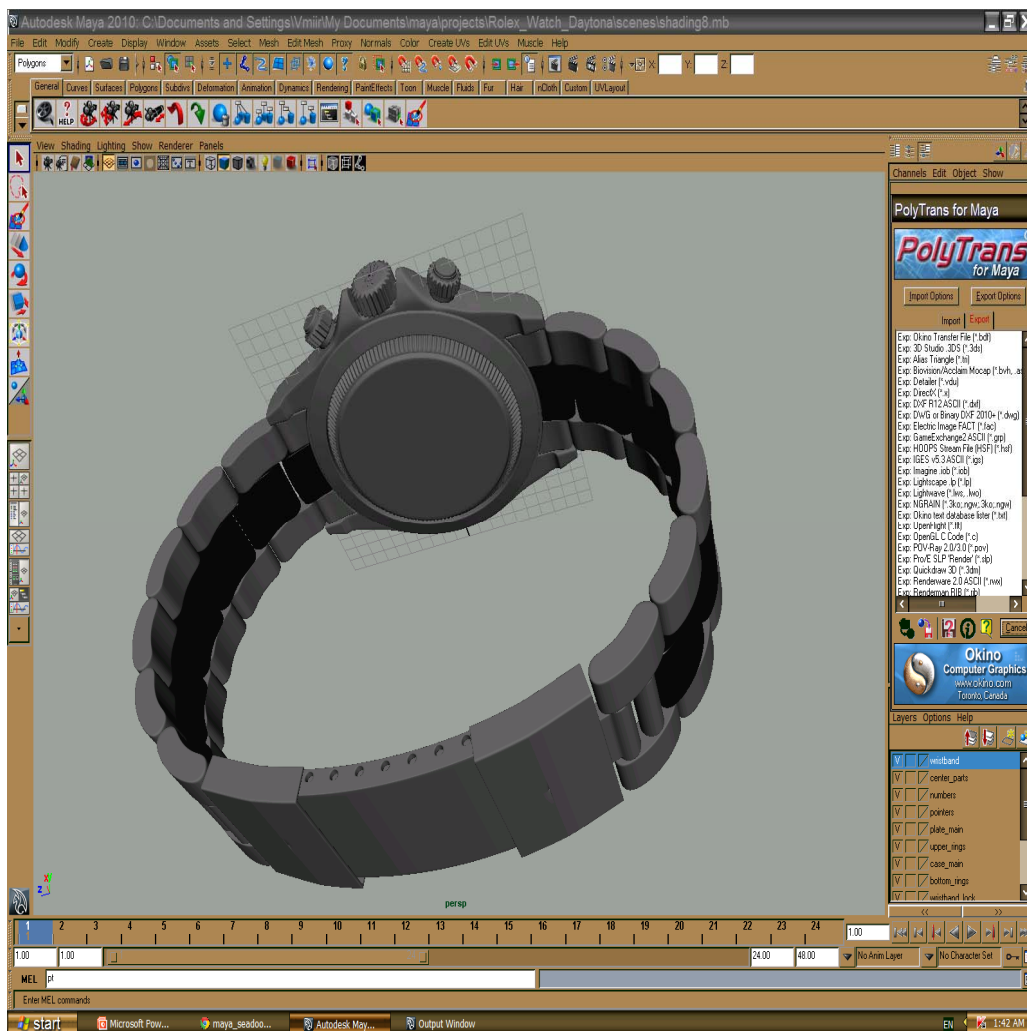
Το εν λόγω λογισμικό, χρησιμοποιήθηκε στην εργασία για τον σχεδιασμό της τρισδιάστατης γεωμετρίας του ρολογιού ROLEX DAYTONA (Εικόνα 11). Στα πλαίσια του προγράμματος υπήρξε ανάγκη να χρησιμοποιηθούν άλλα δύο λογισμικά, τα οποία αποτελούν, ουσιαστικά, plugins του Maya: α) **OKINO COMPUTER GRAPHICS-POLYTRANS** και β) i) **HEADUS UVLAYOUT** ii) **ROADKILL**.



Εικόνα 11

3.2.2 OKINO COMPUTER GRAPHICS-POLYTRANS

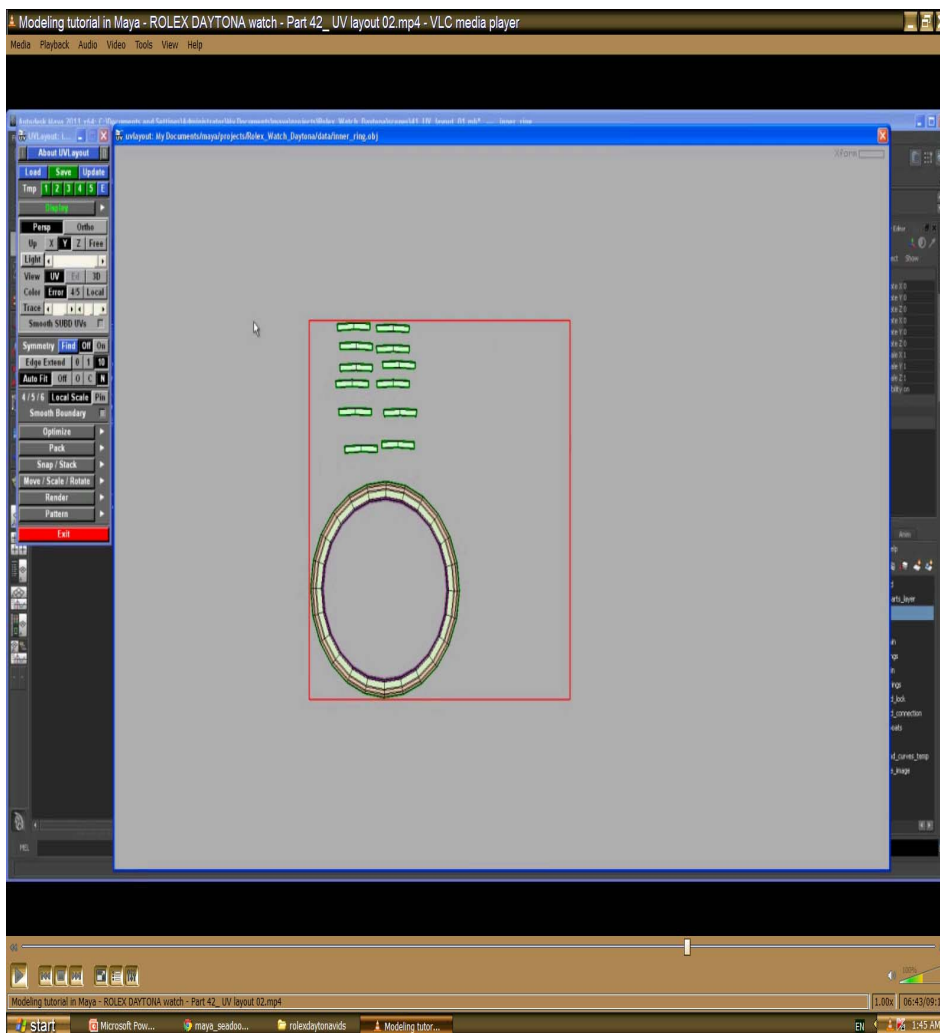
Τα polytrans χρησιμοποιήθηκαν στην εργασία, για να γίνει δυνατή η εξαγωγή της μορφής αρχείου της τρισδιάστατης γεωμετρίας του ρολογιού από το Autodesk Maya σε λογισμικά τρισδιάστατων εφαρμογών (Pro Engineer, Autodesk 3Dsmax), δισδιάστατων εφαρμογών (adobe flash player) και διαδικτυακών εφαρμογών (Cortona, Cosmo player) (Εικόνα 12).



Εικόνα 12

3.2.3 HEADUS UVLAYOUT-ROADKILL

Τα παραπάνω λογισμικά χρησιμοποιήθηκαν ως plugins στο Autodesk Maya, για την δημιουργία και επεξεργασία της υφής σε διάφορα σημεία της γεωμετρίας του ρολογιού (Εικόνα 13).

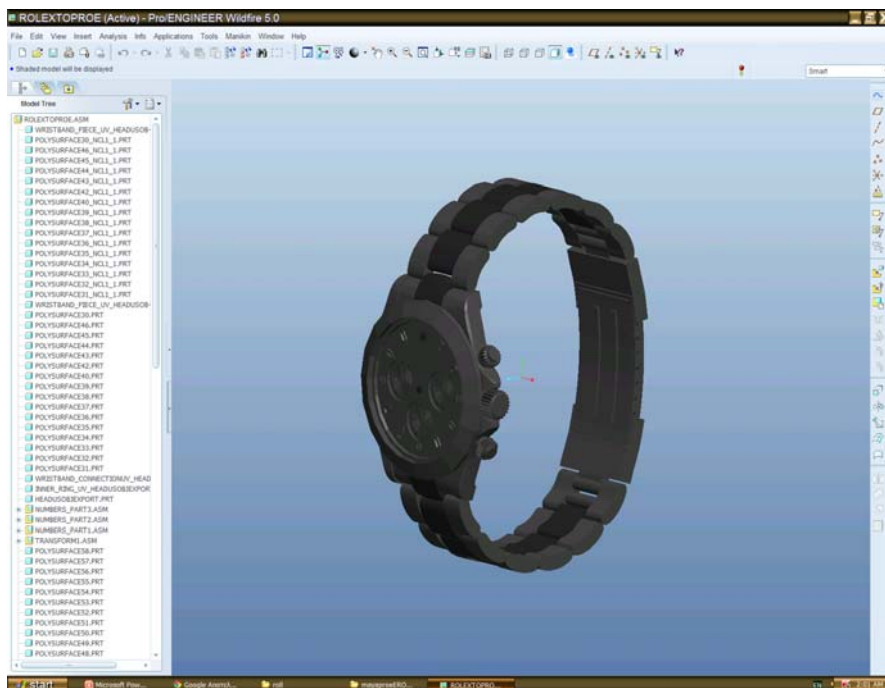


Εικόνα 13

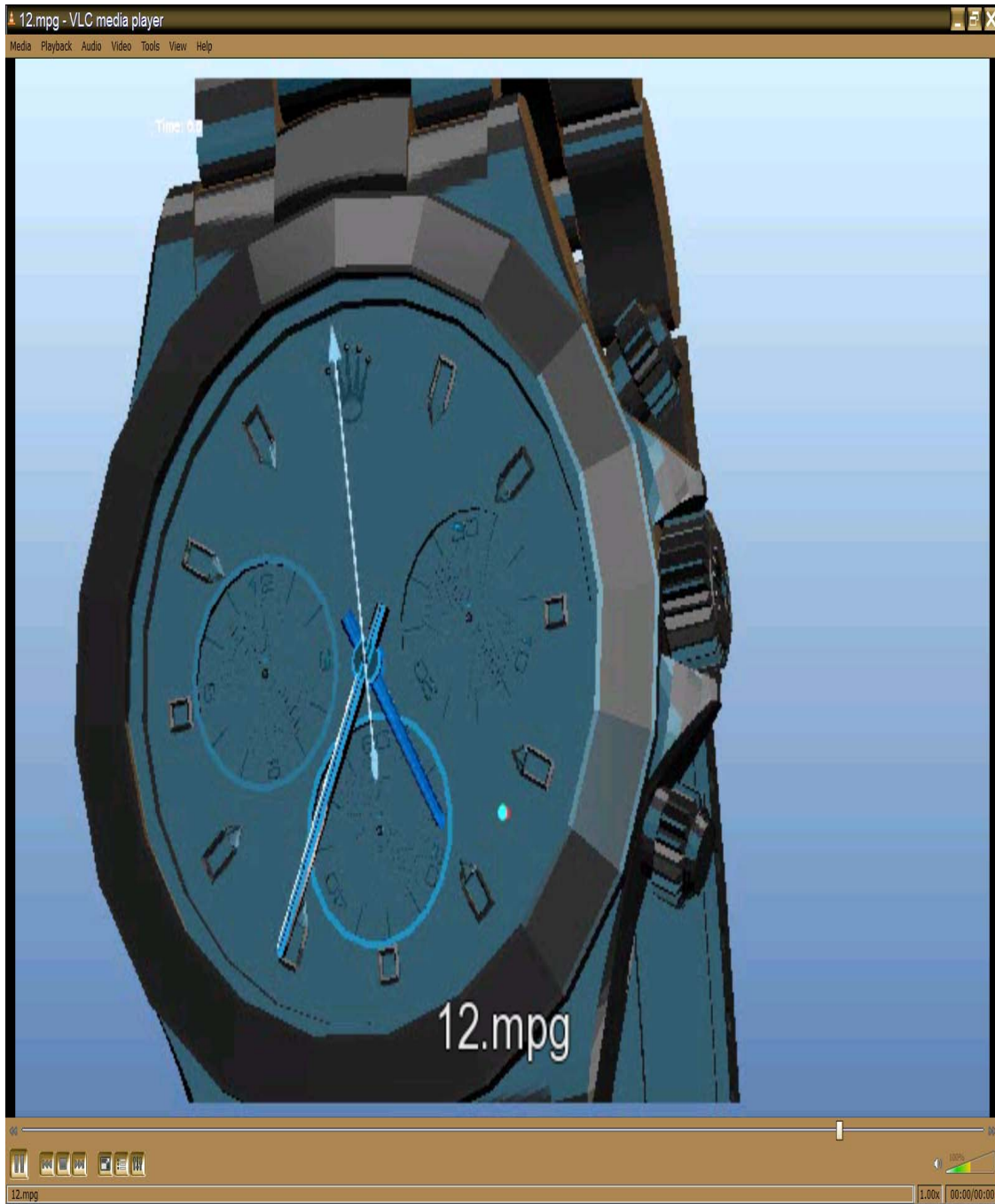
3.2.4 PROENGINEER

Το ProEngineer χρησιμοποιήθηκε στην εργασία για δύο κυρίως λόγους:

- A) Ως μεσάζων λογισμικό, ώστε να μεταφερθεί σε αυτό η γεωμετρία του ρολογιού από το Autodesk Maya σε μορφή αρχείου .fbx (Εικόνα 14) και να είναι εφικτό να εξαχθεί σε μορφή αρχείου που να είναι συμβατή με το adobe flash player (.png, .obj, .stl, .iges) και την Vnml. Η χρήση του αυτή οφείλεται στο γεγονός του ότι στο Autodesk Maya, ένα σχέδιο δεν μπορεί να εξαχθεί παρά μόνο με συγκεκριμένες μορφές αρχείου (.mb μον).
- B) Μεταφέρθηκε στο Pro Engineer η τρισδιάστατη γεωμετρία του ρολογιού, ώστε μέσω των εργαλείων του (mechanism) να γίνει επεξεργασία στους δείκτες του, ορίζοντας στον καθένα κινητήρες servo και αποδίδοντας στην συνέχεια ταχύτητες περιστροφής. Τέλος, έγινε κινηματική ανάλυση και εξαγωγή του ολοκληρωμένου αρχείου σε μορφή .ani (Εικόνα 15).



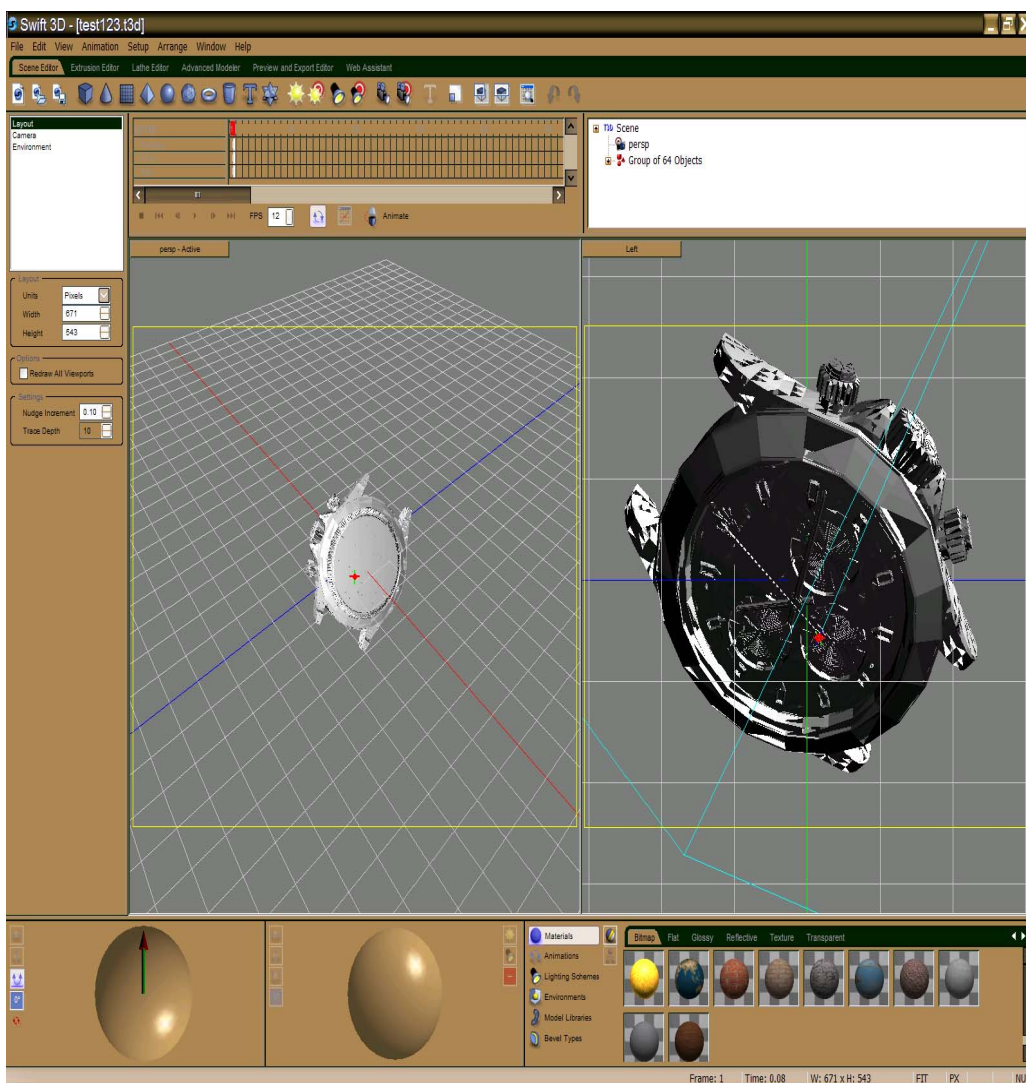
Εικόνα 14



Εικόνα 15

3.2.5 SWIFT 3D

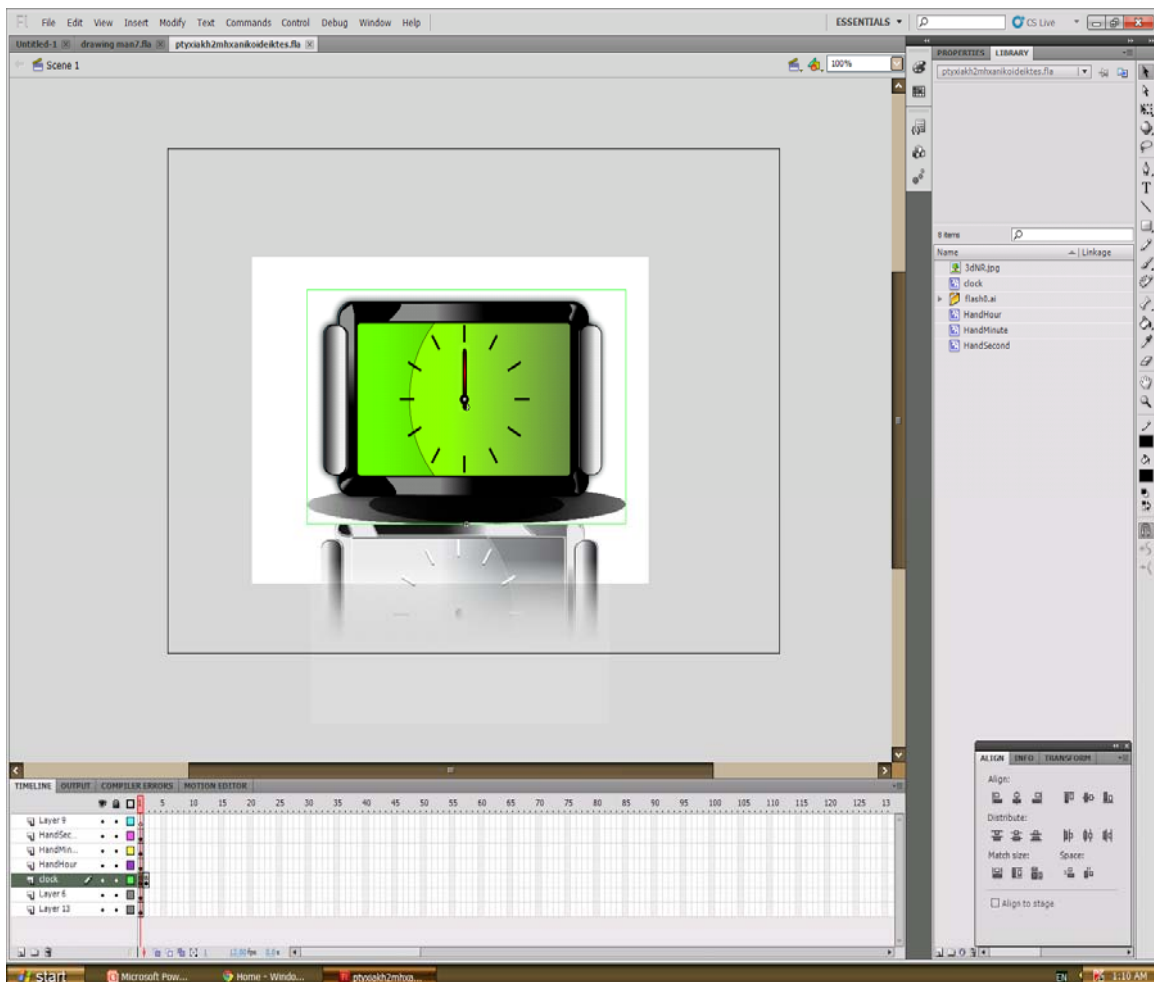
Το SWIFT 3D, χρησιμοποιήθηκε στην εργασία ως μεσάζον λογισμικό για την μετατροπή και μεταφορά αρχείων από το Autodesk Maya (3D) σε adobe flash player (2D) (Εικόνα 16).



Εικόνα 16

3.2.6 ADOBE FLASH PLAYER

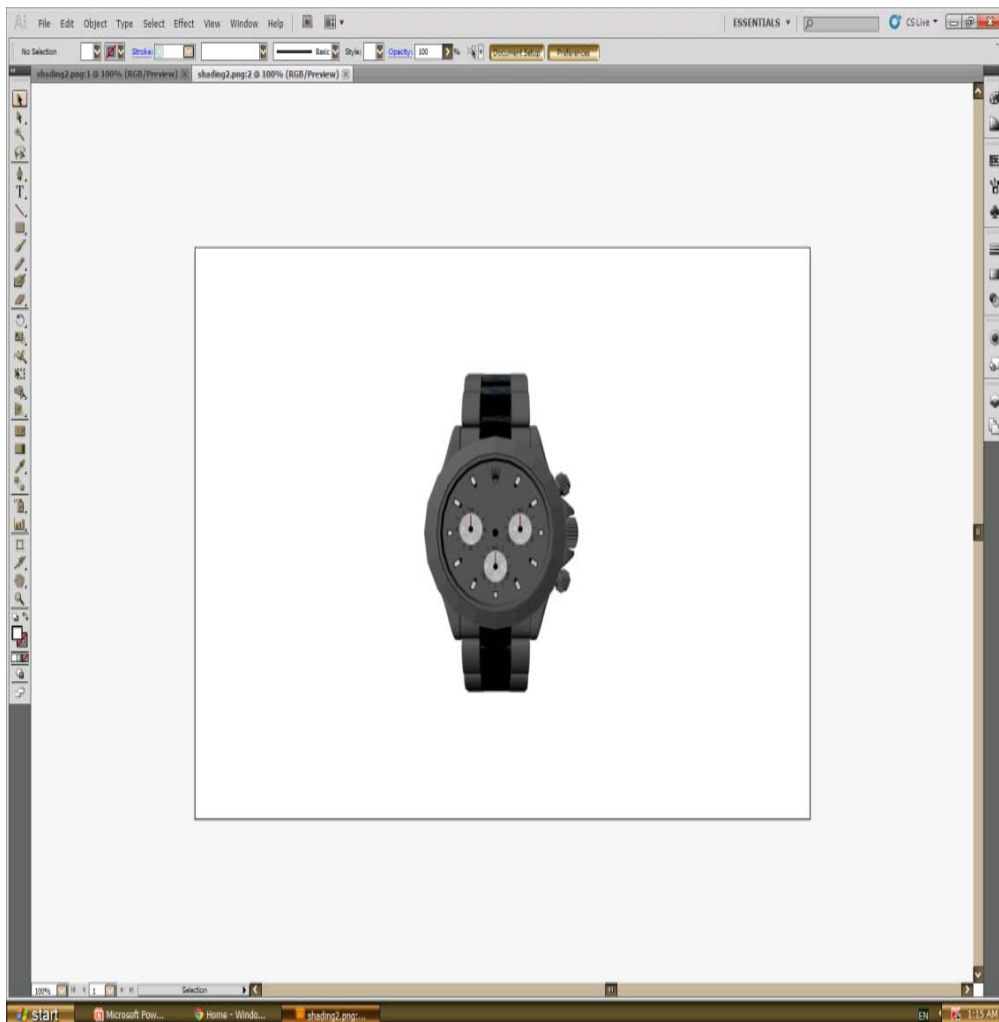
Το adobe flash player, χρησιμοποιήθηκε στα πλαίσια της εργασίας για την εισαγωγή της ολοκληρωμένης γεωμετρίας του ρολογιού και τους δείκτες του ξεχωριστά (Εικόνα 17), την σύνδεσή και αλληλεπίδρασή τους μέσω κατάλληλων εργαλείων του προγράμματος, την ανάπτυξη κώδικα χρησιμοποιώντας γλώσσα προγραμματισμού του flash player, ώστε να λειτουργούν οι δείκτες του ρολογιού βάσει της σωστής ώρας και τέλος την δυνατότητα εξαγωγής του ολοκληρωμένου αρχείου και εφαρμογή του στο διαδίκτυο.



Εικόνα 17

3.2.7 ADOBE FLASH ILLUSTRATOR

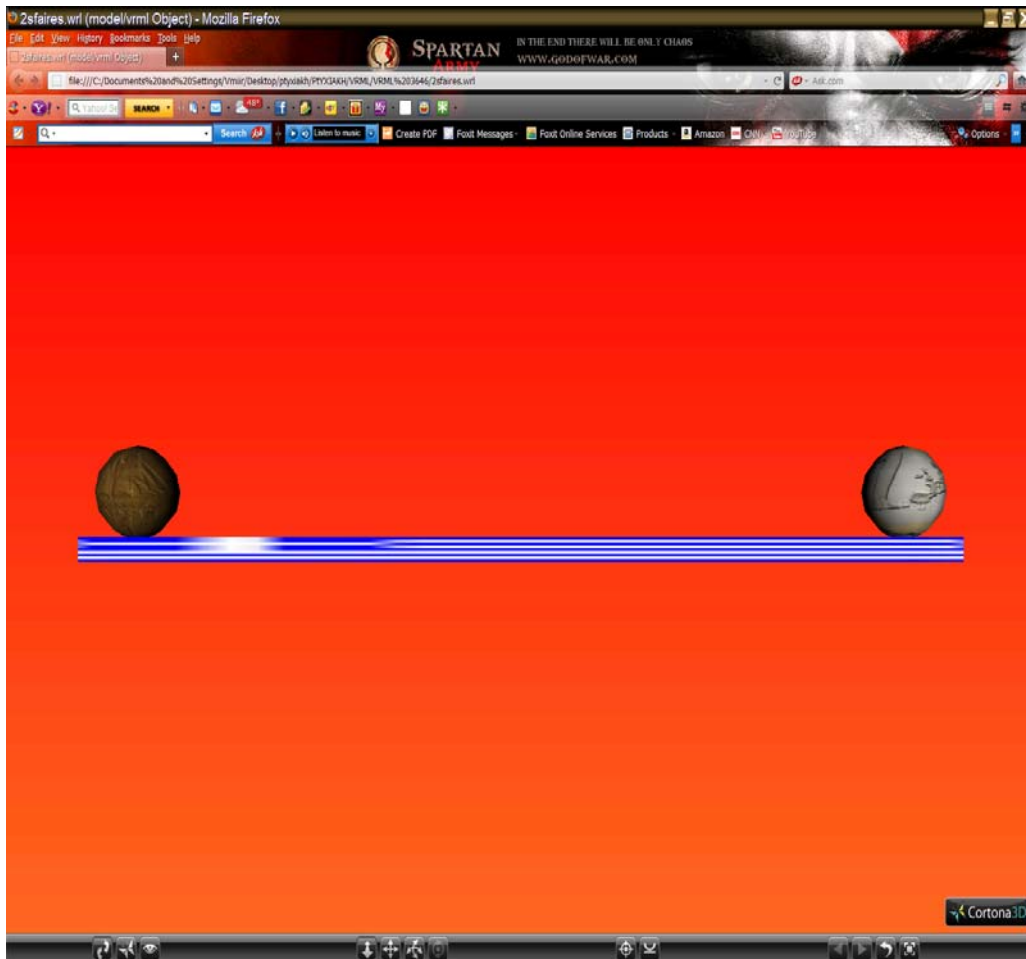
Το Adobe Illustrator χρησιμοποιήθηκε στην εφαρμογή του ρολογιού δύο διαστάσεων, με σκοπό να δημιουργήσει την ψευδαίσθηση ενός τρισδιάστατου μοντέλου. Αυτό επιτεύχθηκε με την δημιουργία αλλά και επεξεργασία των σκιάσεων, του φωτισμού και του ειδώλου του ρολογιού (Εικόνα 18).



Εικόνα 18

3.2.8 COSMOPLAYER-CORTONA PLAYER

Τα εν λόγω προγράμματα χρησιμοποιήθηκαν στα πλαίσια της εργασίας, για την προβολή τρισδιάστατων αλλά και κινούμενων γραφικών στο διαδίκτυο. Χρησιμοποιήθηκε κώδικας σε VRML για την δημιουργία αρχικά μίας κινούμενης κυλιόμενης σφαίρας πάνω σε οριζόντια πλατφόρμα, η οποία εκτελούσε μια διαδρομή με το πάτημα ενός κουμπιού και τελικά για την δημιουργία δύο ίδιων σφαιρών αντίστοιχα που εκτελούσαν κατοπτρικές κινήσεις, συγκρούονταν και επανέρχονταν στις αρχικές τους θέσεις, επίσης με το πάτημα ενός κουμπιού (Εικόνα 19).



Εικόνα 19

4 ΑΝΑΛΥΤΙΚΗ ΠΕΡΙΓΡΑΦΗ ΤΩΝ ΕΦΑΡΜΟΓΩΝ ΣΤΑ ΛΟΓΙΣΜΙΚΑ

Οι εφαρμογές που αναπτύχθηκαν στην συγκεκριμένη εργασία με τα χρησιμοποιούμενα λογισμικά, συνοψίζονται στον ακόλουθο πίνακα:

ΕΦΑΡΜΟΓΕΣ	ΛΟΓΙΣΜΙΚΑ
Κινούμενη σφαίρα τριών διαστάσεων που εκτελεί διαδρομή $A \rightarrow B \rightarrow A$ με σταθερή ταχύτητα και περιστροφή, με το πάτημα ενός κουμπιού	Cortona (Vrml)
Δύο κινούμενες σφαίρες που συγκρούονται και επιστρέφουν στις αρχικές τους θέσεις με το πάτημα ενός κουμπιού.	
Ρολόι δύο διαστάσεων με δείκτες ταλάντωσης ανά δευτερόλεπτο	Adobe Illustrator & Adobe flash player
Ρολόι δύο διαστάσεων με μηχανικούς δείκτες	
Ρολόι Rolex τρισδιάστατης γεωμετρίας δισδιάστατης απεικόνισης με μηχανικούς δείκτες	
Ρολόι ROLEX σε τρισδιάστατη γεωμετρία	Autodesk Maya

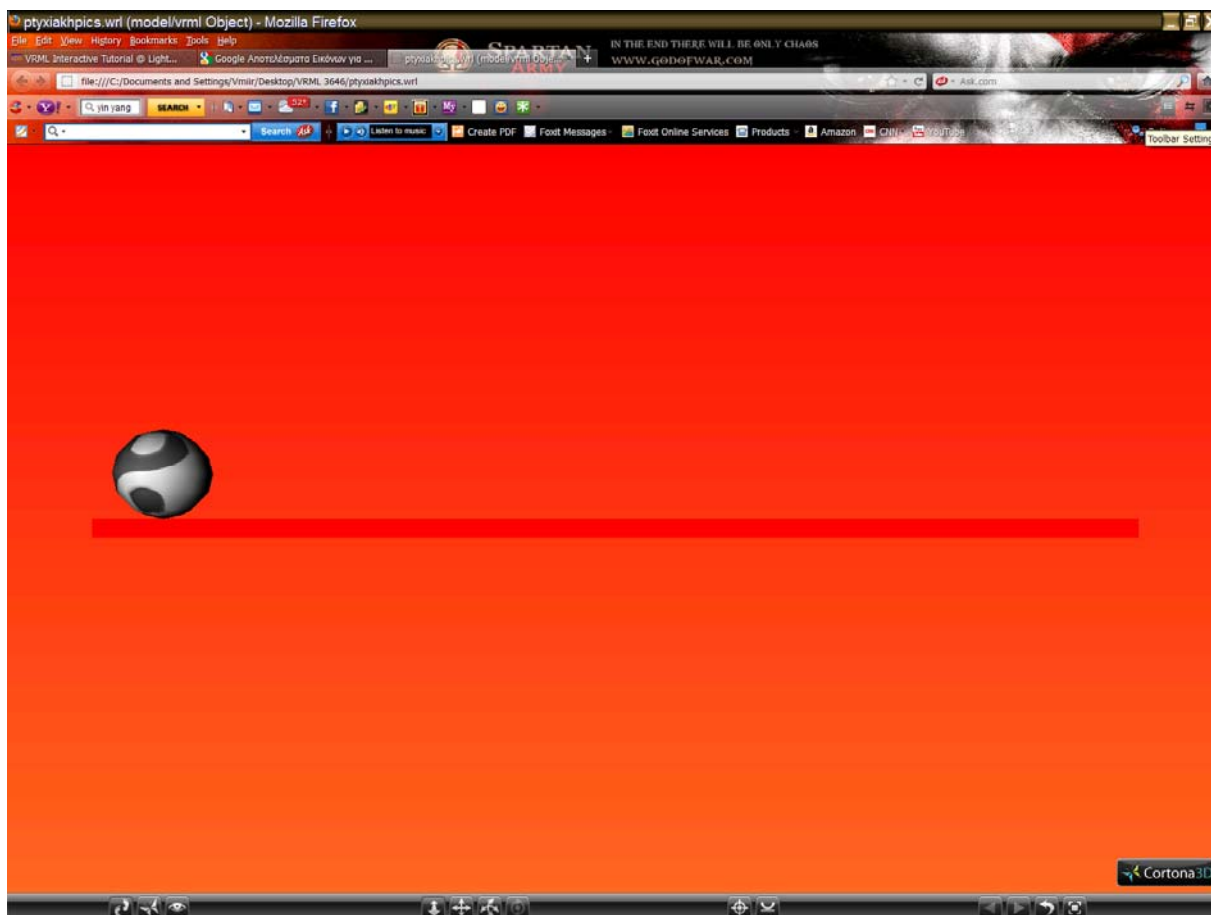
4.1 Εφαρμογές σε Cortona

4.1.1 Κινούμενη σφαίρα τριών διαστάσεων που εκτελεί διαδρομή $A \rightarrow B \rightarrow A$ με σταθερή ταχύτητα και περιστροφή, με το πάτημα ενός κουμπιού

Δημιουργήθηκε, αρχικά, ένα απλό αρχείο κειμένου .txt με κατάληξη .wrl ώστε ανοίγοντας το μέσω ενός Browser (Mozilla Firefox , Google chrome) σε συνεργασία με κατάλληλο λογισμικό (Cosmo player, Cortona), να προβάλλεται το περιβάλλον της VRML. Στην συνέχεια, γίνεται περιγραφή του κώδικα: Εισάγεται η επικεφαλίδα #VRML V2.0 utf8, η οποία συνιστά την βιβλιοθήκη της VRML και είναι απαραίτητη.

4.1.1.1 ΒΗΜΑ 1: Κτίσιμο της σκηνής :

Για τον καθορισμό των συντεταγμένων της πλατφόρμας, εισάγεται ο κόμβος "Transform" στον οποίο προστίθεται το πεδίο "Translate" με συντεταγμένες στους τρεις άξονες '0 - 0.2 0'. Για τον καθορισμό του σχήματος της πλατφόρμας, εισάγεται ο κόμβος shape στον οποίο συμπληρώνονται τα πεδία appearance και geometry. Στον κόμβο "Appearance" συμπληρώνεται ο κόμβος "Material", ο οποίος χρησιμοποιείται για τον καθορισμό του χρώματος του αντικειμένου, δίνοντας για τιμές στο πεδίο "diffuseColor" τρεις πραγματικούς αριθμούς από το 0 έως το 1 (μέγιστη τιμή). Οι τρεις αριθμοί αντιστοιχούν στα χρώματα κόκκινο, πράσινο και μπλε (πρότυπο RGB). Το πεδίο "geometry" έχει ως τιμή τον κόμβο "Box" και με τη σειρά του ο κόμβος "Box" έχει ένα συμπληρωμένο πεδίο "size" με τις διαστάσεις '22 0.4 2'. Κατόπιν, επαναλαμβάνεται το **ΒΗΜΑ 1** για την δημιουργία της σφαίρας στο άνω αριστερό άκρο της πλατφόρμας. Ωστόσο, στον κόμβο "Material", εισάγεται μια υφή texture: ImageTexture { url "Rolex.jpg"} και στο πεδίο "geometry" εισάγεται ο κόμβος "Sphere" με ακτίνα 1 (Εικόνα 20).



Εικόνα 20-Δημιουργία της οριζόντιας πλατφόρμας και της σφαίρας

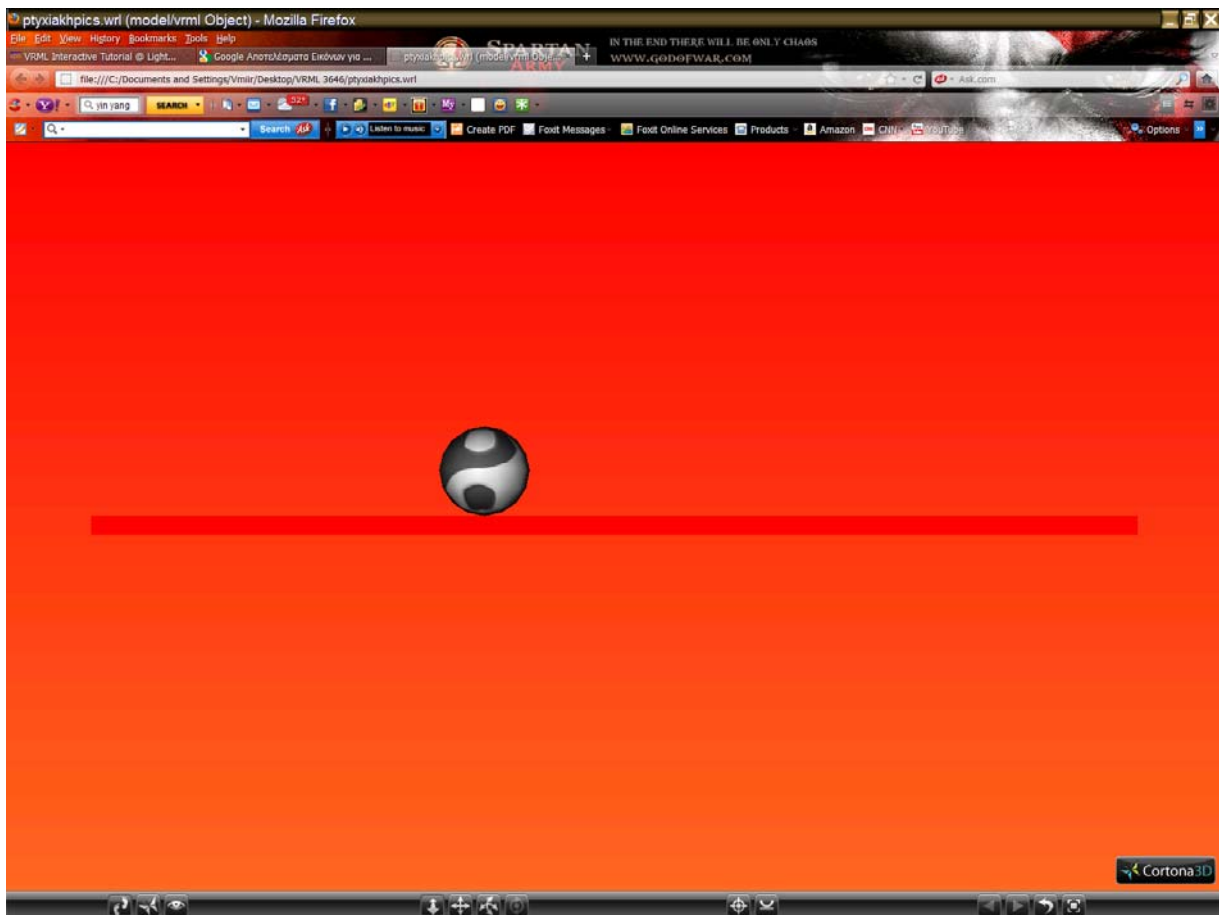
Η σκηνή προβάλλεται διαφορετικά με την χρήση διαφορετικού προγράμματος προβολής. Κάποια προγράμματα προβολής απεικονίζουν ολόκληρη τη σκηνή και στην προκειμένη εφαρμογή, ολόκληρη την πλατφόρμα με την σφαίρα στο άκρο της, ενώ κάποια άλλα δείχνουν μόνο μέρος της πλατφόρμας χωρίς η σφαίρα να είναι ορατή. Τα συγκεκριμένα προγράμματα προβολής συμφωνούν με τις επίσημες προδιαγραφές, οι οποίες ορίζουν μια αρχική θέση για το πρόγραμμα ως προεπιλογή. Οι προηγούμενες εκδόσεις προγραμμάτων προβολής δεν χρησιμοποιούν την προβολή θέσης βάσει προεπιλογής, έναντι υπολογίζουν μία θέση όπου είναι ορατή ολόκληρη η σκηνή. Το πρόγραμμα προβολής που χρησιμοποιείται στην εφαρμογή χρησιμοποιεί την θέση προεπιλογής κι ως εκ τούτου δεν είναι ορατή ολόκληρη η σκηνή. Αυτό συμβαίνει διότι το μάτι του θεατή βρίσκεται πολύ κοντά στην σκηνή. Η θέση προεπιλογής για το

πρόγραμμα προβολής είναι '0 0 10'. Επομένως αρκεί το μάτι του θεατή να απομακρυνθεί λίγο από την σκηνή, ώστε να προβληθεί ολόκληρη. Χρησιμοποιείται ο κόμβος "Viewpoint" μέσω της εντολής **Viewpoint { position 0 0 20 }**.

4.1.1.2 ΒΗΜΑ 2: Animation μέρος I.

Στο σημείο αυτό περιγράφεται η κίνηση της σφαίρας απ' το αριστερό άκρο της πλατφόρμας, στο δεξιό (Εικόνα 21). Σε γενικές γραμμές, τα animation προϋποθέτουν την ύπαρξη ενός μετρητή για τον έλεγχο του ρυθμού. Στην VRML οι μετρητές βρίσκουν εφαρμογή στους **Timesensors**. Ορίζουμε το χρονικό διάστημα στο οποίο η σφαίρα θα κινηθεί απ' το αριστερό στο δεξιό άκρο της πλατφόρμας (15 sec), εκφράζοντάς το στο γενικότερο κομμάτι κώδικα του μετρητή: `DEF timer TimeSensor { cycleInterval 15 loop FALSE}`. Με τον όρο DEF ορίζουμε ένα όνομα στον μετρητή, ώστε το πρόγραμμα να μπορεί να τον "καλέσει" αργότερα. Το πεδίο "loop" ορίζεται αληθές "TRUE" ώστε να επαναλαμβάνεται το animation. Ο κόμβος του μετρητή μπορεί να τοποθετηθεί σε οποιοδήποτε σημείο του κώδικα. Για την αλλαγή της θέσης της σφαίρας εισάγεται στον κώδικα ένας παρεμβολέας θέσης (Position Interpolator), μέσω του οποίου η σφαίρα θα κινηθεί από την αρχική της θέση '-10 1 0', στο αντίθετο άκρο της πλατφόρμας '10 1 0'. Ο κόμβος ορίζεται με το παρακάτω κομμάτι κώδικα: `DEF pi PositionInterpolator { key [0 1] keyValue [-10 1 0, 10 1 0] }`, ο οποίος μπορεί, επίσης, να τοποθετηθεί οπουδήποτε μέσα στον κώδικα. Η κεντρική ιδέα είναι η εξής: Ο παρεμβολέας θέσης (PositionInterpolator) εξάγει events με τρισδιάστατες τιμές οι οποίες θα χρησιμοποιηθούν για να αλλάξουν το πεδίο "translation" του κόμβου "Translation" στο οποίο ορίζεται η σφαίρα (σχήμα, συντεταγμένες). Ωστόσο ο παρεμβολέας θέσης δεν μπορεί να εξάγει events χωρίς να δέχεται events. Σ' αυτό το σημείο, εισχωρεί ο κόμβος του "Timesensor", ο οποίος παράγει events όσο περνάει ο χρόνος. Τα παραγόμενα events στέλνονται στον "PositionInterpolator", ο οποίος με την σειρά του εξάγει events στον κόμβο του "Transform" ώστε τελικά να αλλάξει το πεδίο του "translate" (συντεταγμένες της σφαίρας στο χώρο). Για να είναι, όμως, σε θέση ο κόμβος "Transform" να δεχτεί events απ' τον παρεμβολέα θέσης θα πρέπει να του οριστεί ένα

όνομα μέσω του DEF. Ο κόμβος του “Timesensor” εξάγει ένα event που λέγεται “*fraction_changed*” το οποίο δέχεται τιμές από 0 έως 1, περιγράφοντας την κλασματική ποσότητα του χρόνου από το διάστημα του κύκλου (cycleInterval) που έχει παρέλθει. Η τιμή αυτή (0-1), τροφοδοτείται στον PositionInterpolator μέσω του event *set_fraction*. Η διαδικασία αυτή επιτυγχάνεται με την χρήση των **ROUTES**, τα οποία ουσιαστικά αποτελούν μέσα με τα οποία καλούνται οι κόμβοι μέσα στον κώδικα και μεταφέρονται τα events από μετρητή (TimeSensor), στον εκάστοτε παρεμβολέα και τελικά στο Transform και στο translate. Η διαδικασία αυτή περιγράφεται στα παρακάτω κομμάτια κώδικα:
ROUTE timer.fraction_changed TO pi.set_fraction ROUTE pi.value_changed TO ball_tr.set_translation.



Εικόνα 21-Κίνηση της σφαίρας στην πλατφόρμα

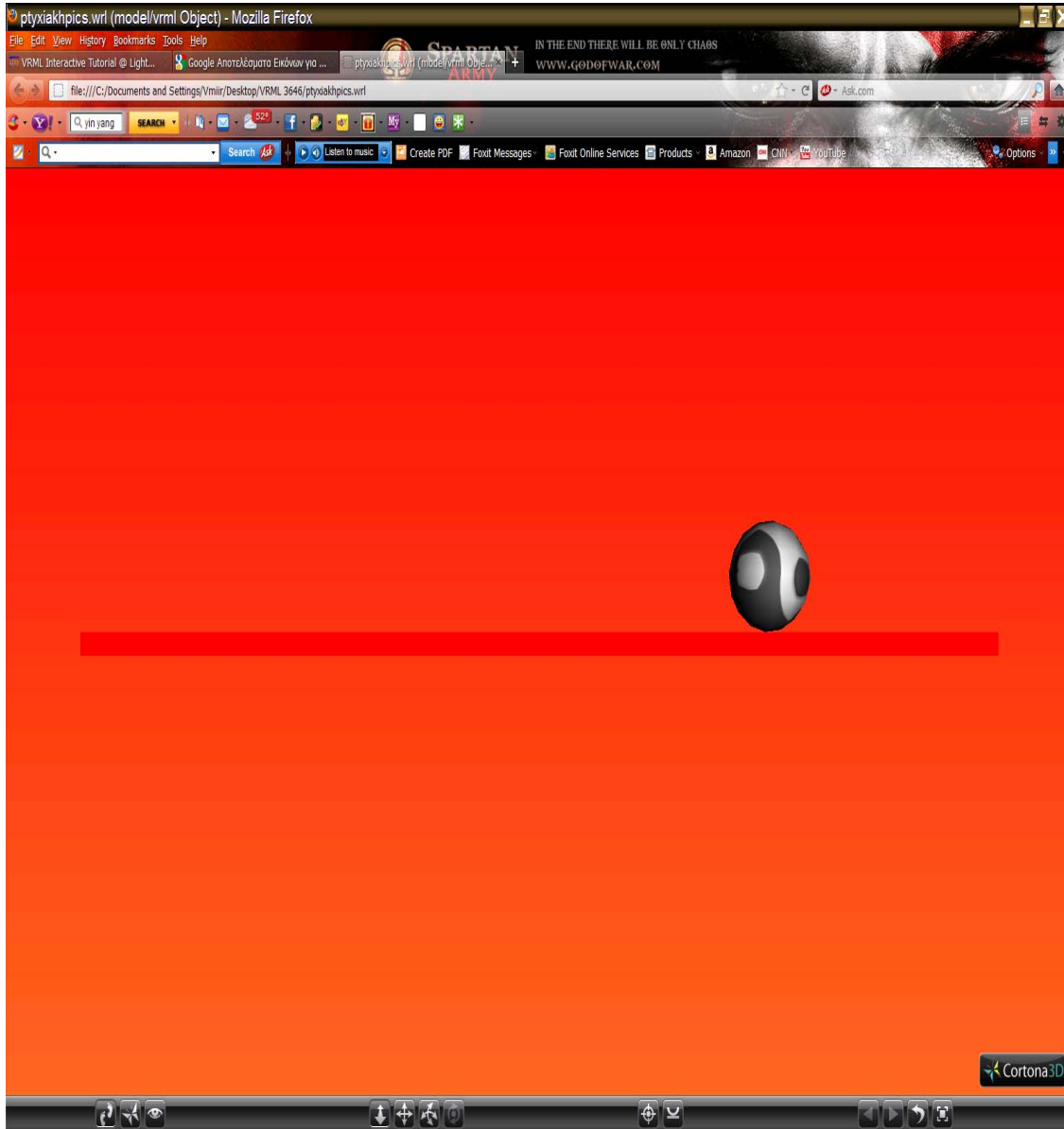
4.1.1.3 ΒΗΜΑ 3: Animation μέρος II

Στο βήμα αυτό, περιγράφεται η διαδικασία περιστροφής της σφαίρας κατά την κίνηση, ώστε η τελευταία να είναι πιο ρεαλιστική (Εικόνα 22), (Εικόνα 23). Για τον σκοπό αυτό, εισάγεται στον κώδικα ο παρεμβολέας προσανατολισμού (**OrientationInterpolator**), σύμφωνα με το παρακάτω κομμάτι κώδικα: DEF οι OrientationInterpolator { key [0 0.071 0.142 0.214 0.285 0.357 0.428 0.5 0.571 0.642 0.714 0.785 0.857 0.928 1] keyValue [0 0 1 0, 0 0 1 -3.14, 0 0 1 -6.28, 0 0 1 -9.42, 0 0 1 -12.56, 0 0 1 -15.7, 0 0 1 -18.84, 0 0 1 -21.6, 0 0 1 -18.84, 0 0 1 -15.7, 0 0 1 -12.56, 0 0 1 -9.42, 0 0 1 -6.28, 0 0 1 -3.14, 0 0 1 0] } .

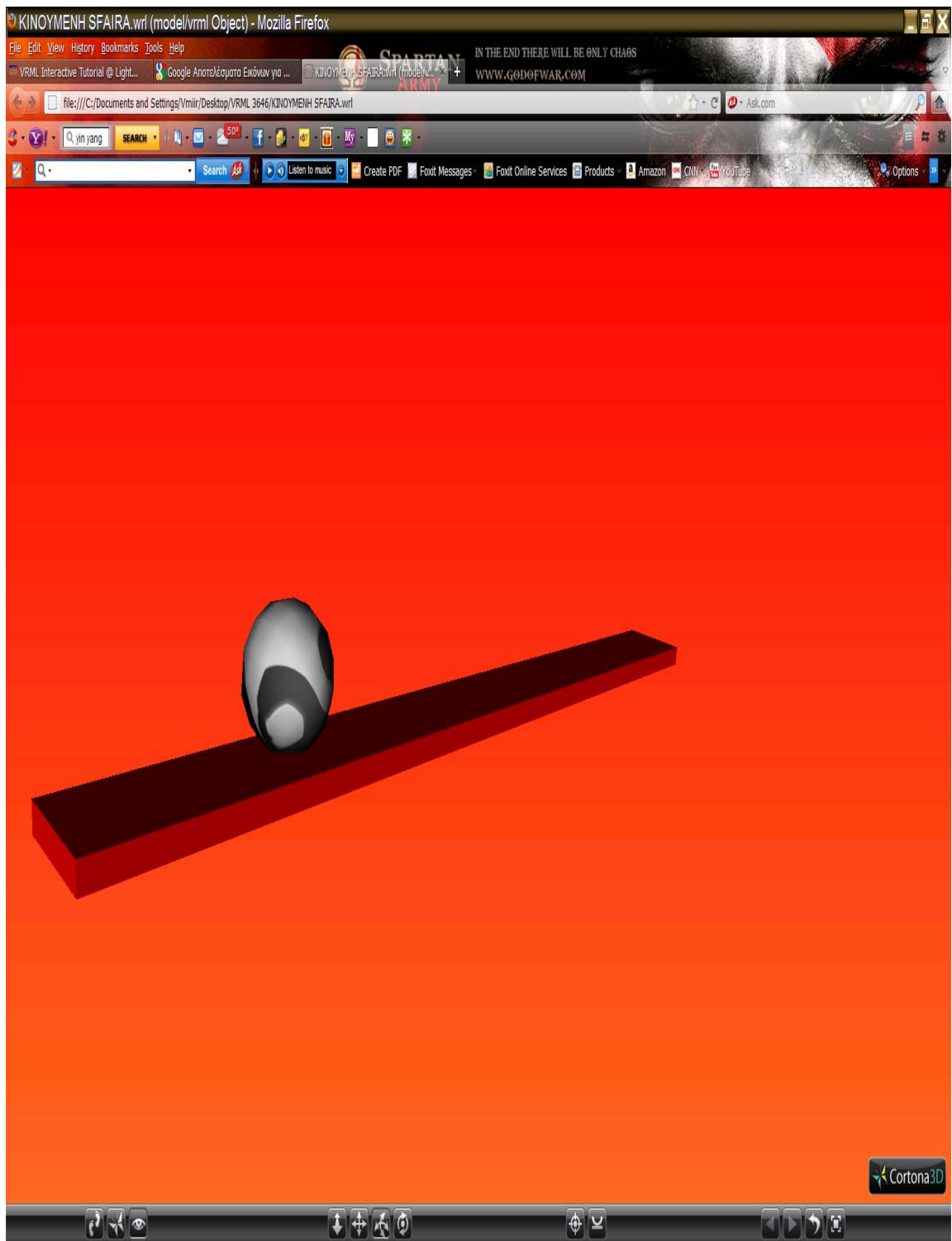
Γενικά, ο ρόλος του κόμβου του παρεμβολέα, είναι να περιγράφει τις αλλαγές που προκύπτουν κατά τη διάρκεια της κίνησης. Λαμβάνει μία δεκαδική τιμή **eventIn value(set_fraction)** και κάνοντας χρήση των τιμών **key** και **keyValue** οι οποίες ορίζονται μέσα στον κόμβο, υπολογίζει μία καινούργια τιμή **eventOut(value_change)**, η οποία μπορεί να χρησιμοποιηθεί για να αλλάξει τις ιδιότητες άλλων κόμβων. Το πεδίο **key** είναι μία λίστα από δεκαδικές τιμές, οι οποίες συνήθως αναπαριστούν σημεία σε κλασματικό χρόνο. Το πεδίο **keyValue**, περιέχει μία λίστα από αντίστοιχες τιμές eventOut για κάθε τιμή του πεδίου **key**. Όταν η τιμή eventIn βρίσκεται μεταξύ των τιμών του πεδίου key, η τιμή του eventOut καθορίζεται με την χρήση γραμμικής παρεμβολής. Στην εφαρμογή, οι τιμές του **key** αναπαριστούν τις θέσεις/σημεία σε κλασματικό χρόνο στις οποίες θα βρίσκεται η σφαίρα. Χρησιμοποιήθηκαν 15 τιμές βάσει της τιμής του cycle Interval του μετρητή TimeSensor, σύμφωνα με την οποία η διαδρομή της σφαίρας θα διαρκέσει 15 δευτερόλεπτα. Οι τιμές του **keyValue**, αναπαριστούν την αντίστοιχη περιστροφή της σφαίρας για κάθε τιμή του **key**. Έτσι για παράδειγμα, στο σημείο 0.071, η σφαίρα θα περιστραφεί κατά 3.14 μοίρες, στο σημείο 0.285 θα περιστραφεί κατά -12.56 μοίρες κοκ. Εν τέλει, ο OrientationInterpolator δέχεται τις τιμές (fraction_changed) απ' τον μετρητή, τις στέλνει (value_changed) στον κόμβο του Transform και εν συνεχεία στο πεδίο Translate για να αλλάξουν οι συντεταγμένες της σφαίρας και να περιστραφεί, όπως φαίνεται στο κομμάτι του κώδικα:

```
ROUTE timer.fraction_changed TO oi.set_fraction
```

ROUTE oi.value_changed TO ball_tr.set_rotation



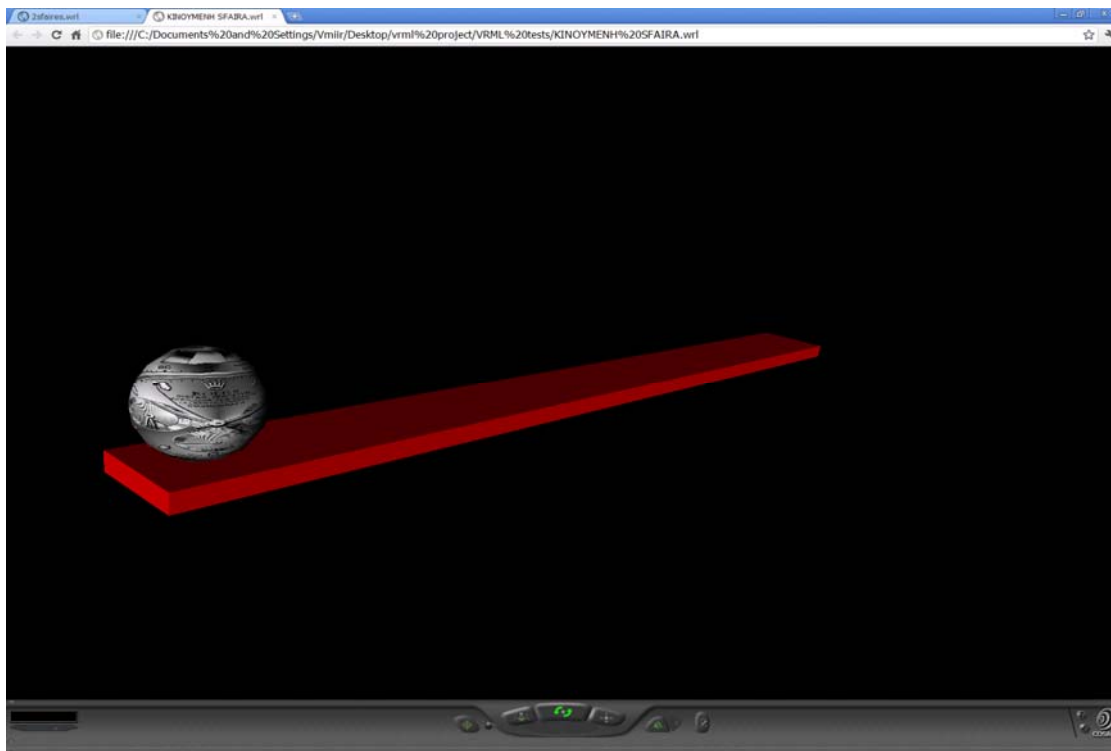
Εικόνα 22-Περιστροφή της σφαίρας κατά την κίνηση



Εικόνα 23-Περιστροφή της σφαίρας κατά την κίνηση

4.1.1.4 ΒΗΜΑ 4: Έναρξη της κίνησης με κλικ του ποντικιού πάνω στην σφαίρα.

Προκειμένου η κίνηση της σφαίρας να ενεργοποιείται με ένα κλικ, θα πρέπει η κίνηση να μην επαναλαμβάνεται συνεχώς, όπως συνέβαινε μέχρι το σημείο αυτό. Πρέπει κατά συνέπεια, να μετατραπεί το πεδίο loop του TimeSensor σε FALSE: DEF timer TimeSensor { cycleInterval 8 loop FALSE }. Στην συνέχεια, προστίθεται ένας αισθητήρας αφής (TouchSensor) στο ίδιο γκρουπ με την σφαίρα: DEF ball_tr Transform { translation -10 1 0 children [Shape { appearance Appearance { material Material { texture ImageTexture { url "cone.jpg" } } geometry Sphere { } } DEF ball_sensor TouchSensor { }] }. Ο TouchSensor θα παράγει το event (touchTime), όταν ο χρήστης κάνει κλικ πάνω στο σχήμα που βρίσκεται στο ίδιο γκρουπ με αυτόν. Το event αυτό, ενεργοποιεί το start_time του μετρητή και κατ' επέκταση το animation (Εικόνα 24). Παρακάτω προβάλλεται ο συνολικός κώδικας.



Εικόνα 24-Ενεργοποίηση της κίνησης με κλικ πάνω στην σφαίρα

#VRML V2.0 utf8

```
Transform {  
translation 0 -0.2 0  
children Shape {  
appearance Appearance { material  
Material { diffuseColor 7 0 0}}  
geometry Box {size 22 0.4 2}  
}  
}
```

```
DEF tr Transform {  
translation -10 1 0  
children[  
Shape {  
appearance Appearance {  
material Material { }  
texture ImageTexture {  
url "rolex.jpg"}  
}  
geometry Sphere { }  
}  
DEF ball_sensor TouchSensor {  
]  
}  
Viewpoint { position 0 0 20 }
```

```
DEF timer TimeSensor {  
cycleInterval 15  
loop FALSE  
}  
DEF pi PositionInterpolator {  
key [ 0 0.5 1 ]  
keyValue [-10 1 0, 10 1 0, -10 1 0]  
}  
ROUTE timer.fraction_changed TO  
pi.set_fraction  
ROUTE pi.value_changed TO  
tr.set_translation  
  
DEF oi OrientationInterpolator {  
key [0 0.071 0.142 0.214 0.285 0.357  
0.428 0.5 0.571 0.642 0.714 0.785  
0.857 0.928 1]  
keyValue [ 0 0 1 0, 0 0 1 -3.14,  
0 0 1 -6.28, 0 0 1 -9.42, 0 0 1 -12.56,  
0 0 1 -15.7, 0 0 1 -18.84, 0 0 1 -21.6 , 0  
0 1 -18.84 , 0 0 1 -15.7  
, 0 0 1 -12.56, 0 0 1 -9.42, 0 0 1 -6.28, 0  
0 1 -3.14, 0 0 1 0]  
}  
}
```

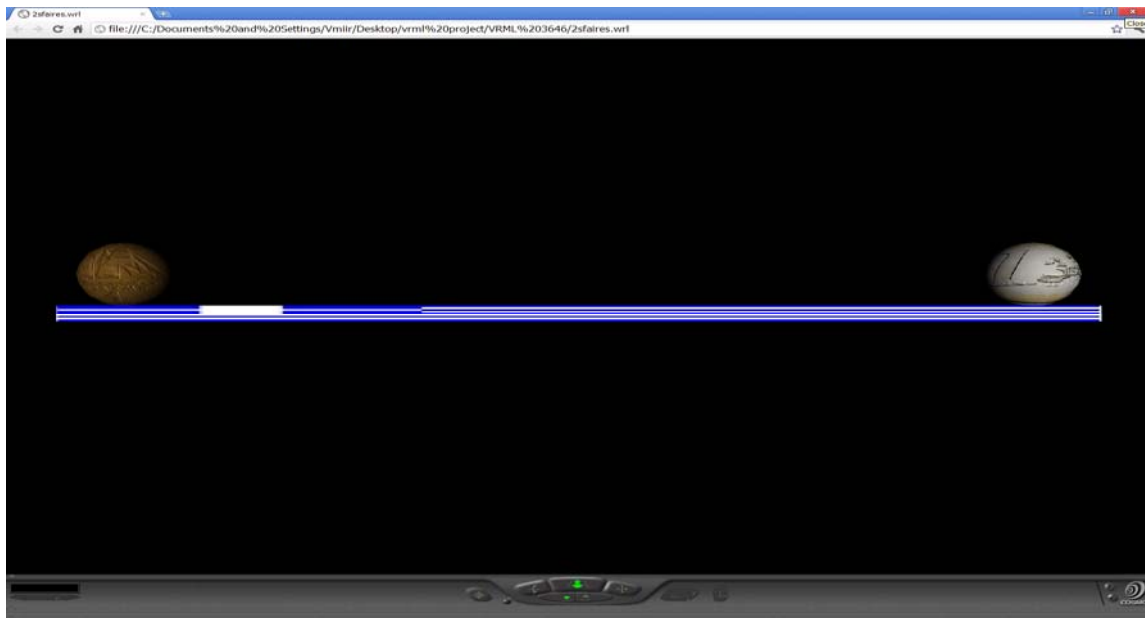
**ROUTE timer.fraction_changed TO
oi.set_fraction**

**ROUTE ball_sensor.touchTime TO
timer.set_startTime**

**ROUTE oi.value_changed TO
tr.set_rotation**

4.1.2 Δύο κινούμενες σφαίρες που συγκρούονται και επιστρέφουν στις αρχικές τους θέσεις με το πάτημα ενός κουμπιού.

Σκοπός της εφαρμογής αυτής είναι η κίνηση δύο σφαιρών, με ίδια ταχύτητα και περιστροφή, οι οποίες ξεκινάνε από τα δύο αντίθετα άκρα μίας πλατφόρμας, συγκρούονται στο μέσο της και επιστρέφουν στις αρχικές τους θέσεις, όπου και τερματίζεται η κίνηση. Η ενεργοποίηση της κίνησης, γίνεται με ένα κλικ του ποντικιού πάνω σε μία από τις σφαίρες. Οι κόμβοι και τα πεδία που χρησιμοποιήθηκαν στην εφαρμογή αυτή, περιγράφονται αναλυτικά στην εφαρμογή 4.1.1, επομένως θα περιγραφεί η δομή, οι προσθήκες και οι τροποποιήσεις που έγιναν σε σχέση με την εφαρμογή 4.1.1(Εικόνα 25)



Εικόνα 25

Αρχικά, δημιουργείται η γεωμετρία της οριζόντιας πλατφόρμας:

```
#VRML V2.0 utf  
  
Transform {  
translation 0 -0.2 0  
  
children Shape {  
appearance Appearance { material  
Material {} texture ImageTexture { url  
"GF.jpg"}  
}  
  
geometry Box {size 22 0.5 2}  
}  
}
```

Στην συνέχεια, εισάγεται η πρώτη σφαίρα που ορίζεται ως **DEF tr1** και ο αισθητήρας αφής που ορίζεται ως **DEF ball_sensor** :

```
DEF tr1 Transform {  
translation -10 1 0  
  
children [  
Shape {  
appearance  
Appearance { material Material { }  
texture ImageTexture { url "DC.jpg"}  
}  
  
geometry Sphere { }  
}  
  
DEF ball_sensor TouchSensor { }
```

]

Αντίστοιχα, δημιουργείται η δεύτερη σφαίρα με το όνομα **DEF tr2**:

Χρησιμοποιείται η εντολή `viewpoint`, ώστε να είναι εμφανής όλη η σκηνή: Εισάγονται δύο μετρητές, ένας για κάθε σφαίρα, ώστε καθένας να παράγει τιμές που θα αποστέλλονται στους παρεμβολείς θέσης και προσανατολισμού της καθεμίας και καθένας απ' τους παρεμβολείς να στέλνει τις υπολογισμένες τιμές στο πεδίο `Translate` της κάθε σφαίρας, ώστε ν' αλλάζουν οι συντεταγμένες κίνησής τους:

Κατόπιν, εισάγονται στον κώδικα δύο παρεμβολείς θέσης αντίστοιχα, έχοντας ίδια πεδία `key` που περιγράφουν τις τρεις χρονικές στιγμές (0,0.5,1), που θα καθορίσουν τις 3 θέσεις του καθενός οι οποίες περιγράφονται στο `keyValue`.

(Πχ. Η πρώτη σφαίρα ξεκινά από αριστερά (-10), φτάνει στην μέση της

}

```
DEF tr2 Transform {  
translation 10 1 0  
children  
Shape {  
appearance  
Appearance { material Material { }  
texture ImageTexture { url "EC.jpg" }  
}  
geometry Sphere { }  
}  
}  
Viewpoint { position 0 0 20 }  
DEF timer1 TimeSensor {  
cycleInterval 7  
loop FALSE  
}  
DEF timer2 TimeSensor {  
cycleInterval 7  
loop FALSE  
}
```

πλατφόρμας (-1) και ξαναγυρίζει στην αρχική της θέση:

key [0 0.5 1]

keyValue [10 1 0, 1 1 0, 10 1 0]

}

Στην συνέχεια προστίθενται δύο παρεμβολείς προσανατολισμού για την πραγματοποίηση της κίνησης των σφαιρών με περιστροφή:

DEF pi1 PositionInterpolator {

key [0 0.5 1]

keyValue [-10 1 0, -1 1 0, -10 1 0]

DEF pi2 PositionInterpolator {

DEF oi1 OrientationInterpolator {

**key [0 0.071 0.142 0.214 0.285 0.357
0.428 0.5 0.571 0.642 0.714 0.785
0.857 0.928 1]**

keyValue [0 0 1 0, 0 0 1 -3.14,

0 0 1 -6.28, 0 0 1 -9.42, 0 0 1 -12.56,

**0 0 1 -15.7, 0 0 1 -18.84, 0 0 1 -21.6 , 0
0 1 -18.84 , 0 0 1 -15.7**

**, 0 0 1 -12.56 , 0 0 1 -9.42 , 0 0 1 -6.28 ,
0 0 1 -3.14 , 0 0 1 0]**

} DEF oi2 OrientationInterpolator {

**key [0 0.071 0.142 0.214 0.285 0.357
0.428 0.5 0.571 0.642 0.714 0.785
0.857 0.928 1]**

**keyValue [0 0 1 -21.6, 0 0 1 -18.84, 0 0
1 -15.7, 0 0 1 -12.56,**

**0 0 1 -9.42, 0 0 1 -6.28, 0 0 1 -3.14, 0 0
1 0, 0 0 1 -3.14,**

0 0 1 -6.28, 0 0 1 -9.42, 0 0 1 -12.56, 0 0 1 -15.7, 0 0 1 -18.84,

Εν τέλει, συμπληρώνονται:

Τα ROUTES όπου οι μετρητές στέλνουν τιμές στους παρεμβολείς θέσης:

Τα ROUTES όπου οι παρεμβολείς θέσης στέλνουν τιμές στους κόμβους Transform των σφαιρών:

Τα ROUTES όπου οι αισθητήρες αφής στέλνουν δεδομένα στους μετρητές για την έναρξη του animation:

Τα ROUTES όπου οι μετρητές στέλνουν τιμές στους παρεμβολείς προσανατολισμού: Τα ROUTES όπου οι παρεμβολείς προσανατολισμού στέλνουν τιμές στους κόμβους Transform των σφαιρών:

ROUTE timer1.fraction_changed TO pi1.set_fraction

0 0 1 -21.6]} }

ROUTE timer2.fraction_changed TO pi2.set_fraction

ROUTE pi1.value_changed TO tr1.set_translation

ROUTE pi2.value_changed TO tr2.set_translation

ROUTE ball_sensor.touchTime TO timer1.set_startTime

ROUTE ball_sensor.touchTime TO timer2.set_startTime

ROUTE timer1.fraction_changed TO oi1.set_fraction

ROUTE timer2.fraction_changed TO oi2.set_fraction

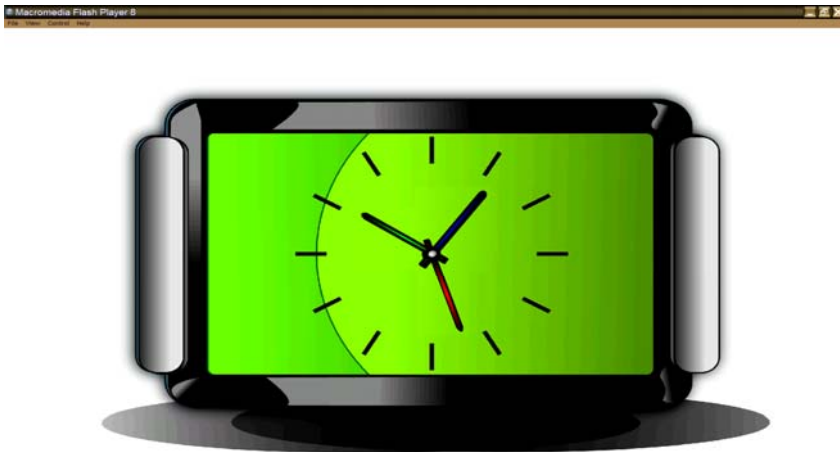
ROUTE oi1.value_changed TO tr1.set_rotation

ROUTE oi2.value_changed TO tr2.set_rotation

4.2 Εφαρμογές σε προγράμματα Adobe

4.2.1 Ρολόι δύο διαστάσεων με δείκτες ταλάντωσης ανά δευτερόλεπτο

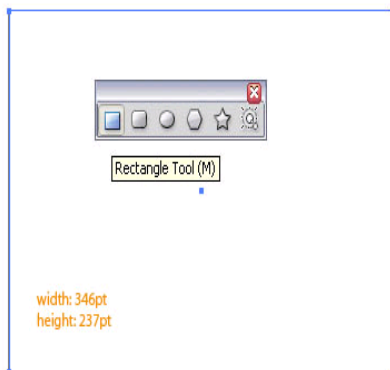
Αρχικά, θα περιγραφεί η δημιουργία ενός αναλογικού ρολογιού δύο διαστάσεων, με δείκτες ταλάντωσης ανά δευτερόλεπτο (Εικόνα 26). Για την εφαρμογή αυτή έγινε χρήση δύο προγραμμάτων: 1) Adobe Illustrator: Το πρόγραμμα χρησιμοποιήθηκε για τον σχεδιασμό του ρολογιού, την απόδοση χρωμάτων, σκιάσεων, αντανάκλασης και γενικά με σκοπό την όσο το δυνατό ρεαλιστική απεικόνιση του ρολογιού ώστε να δίνει την ψευδαίσθηση του τρισδιάστατου. 2) Adobe Flash Player: Στο πρόγραμμα αυτό, μεταφέρθηκε το αρχείο του σχεδίου του ρολογιού ώστε να σχεδιαστούν οι δείκτες και να προσαρμοστούν στο ρολόι και τέλος να γίνει εισαγωγή κώδικα ώστε να δουλεύουν οι δείκτες και να λειτουργεί το ρολόι 'διαβάζοντας' την ώρα από τα windows. Η μεθοδολογία που χρησιμοποιήθηκε στα δύο προγράμματα, περιγράφεται αναλυτικά παρακάτω:



Εικόνα 26

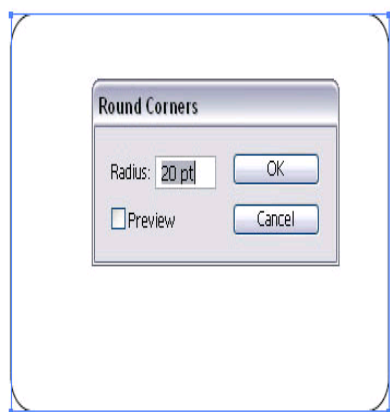
Adobe Illustrator

Αρχικά, δημιουργήθηκε ένα καινούργιο αρχείο στο Illustrator και επιλέχθηκε κατηγορία χρώματος RGB. Έπειτα επιλέχθηκε το Rectangle Tool (M) και σχεδιάστηκε ένα ορθογώνιο με πλάτος=346 και ύψος=237(Εικόνα 27):



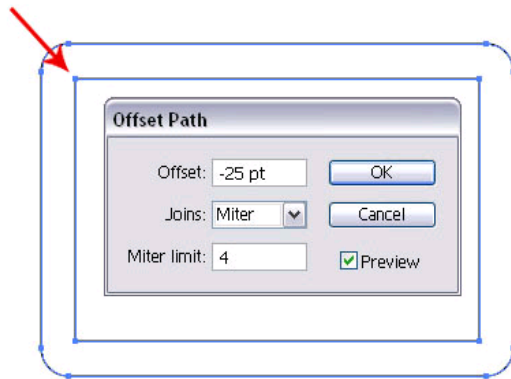
Εικόνα 27

Με τις εντολές Effect > Stylize > Rounded Corners και με το ορθογώνιο επιλεγμένο, οι άκρες του έγιναν στρογγυλεμένες (20pt.) (Εικόνα 28):



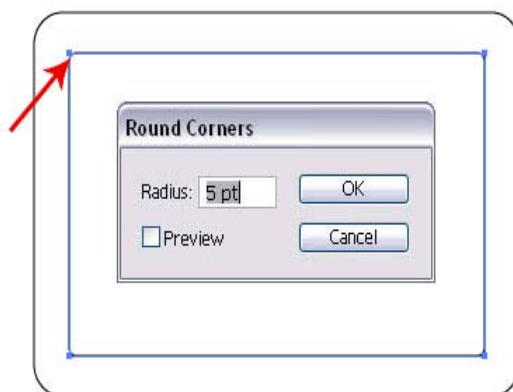
Εικόνα 28

Κατόπιν, οι άκρες πρέπει να επεκταθούν με την εντολή Object > Expand Appearance. Επιλέγεται μία στρογγυλεμένη άκρη, χρησιμοποιούνται οι εντολές Object > Path > Offset Path με offset=-25, ώστε το ορθογώνιο να συρρικνωθεί αναλογικά, δημιουργώντας έτσι δύο σχήματα (Εικόνα 29):



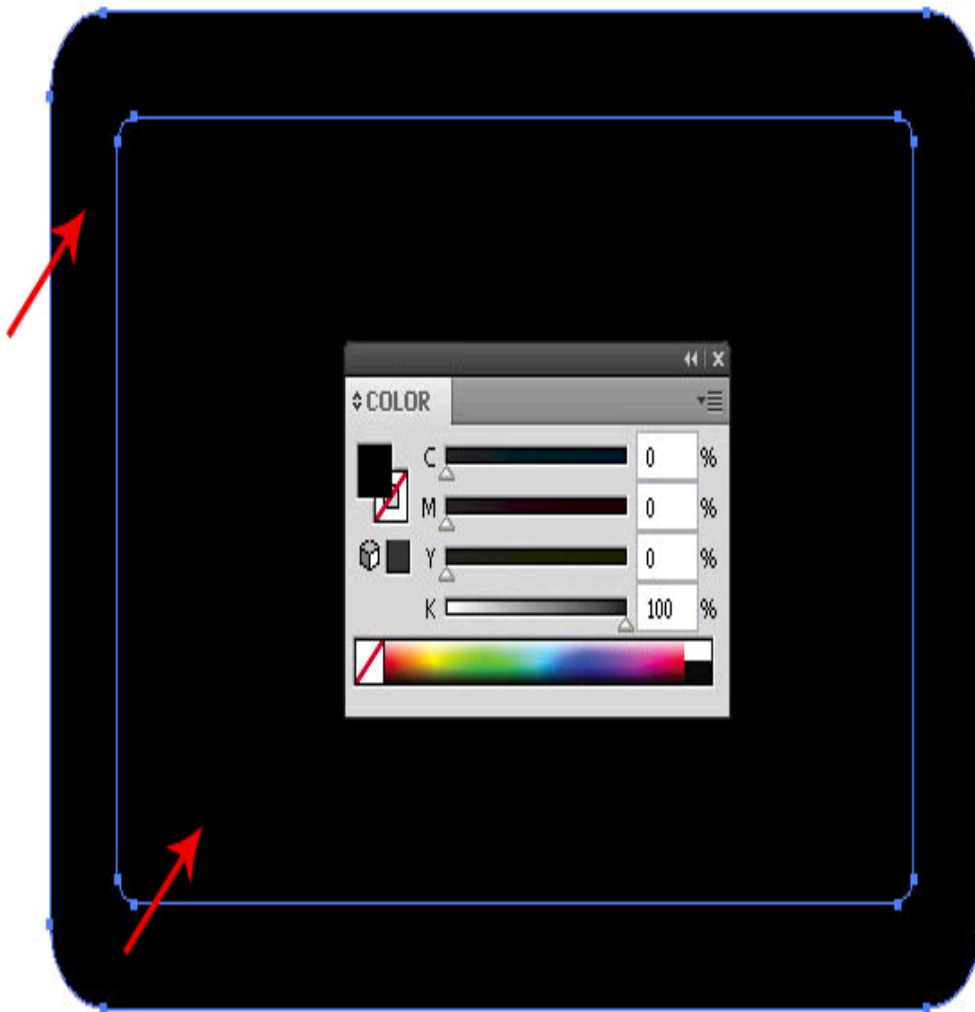
Εικόνα 29

Τα δύο σχήματα ξεχωρίζονται πατώντας Cmd / Ctrl + Shift + G. Επιλέγεται το μικρότερο εσωτερικό ορθογώνιο και εφαρμόζεται σε αυτό μία ακόμη στρογγυλή άκρη με ακτίνα 5pt. κι έπειτα επεκτείνεται ξανά (Object > Expand Appearance) (Εικόνα 30):



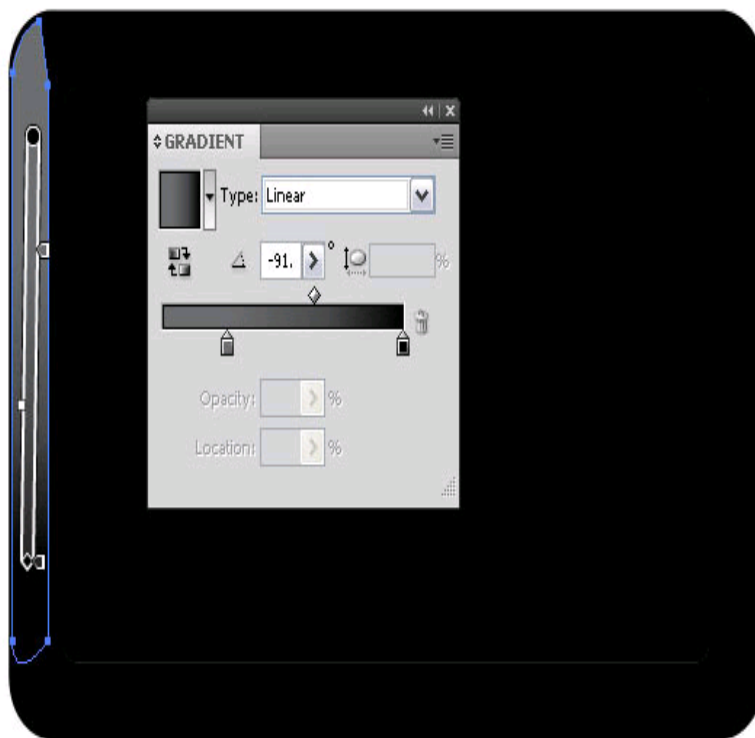
Εικόνα 30

Τα δύο στρογγυλεμένα ορθογώνια (το ένα μικρότερο απ' το άλλο) γεμίζονται με μαύρο χρώμα, αυτό θα αποτελέσει το βασικό σχήμα για το ρολόι. Το εξωτερικό/μεγαλύτερο ορθογώνιο αποτελεί το πλαίσιο και το εσωτερικό/μικρότερο αποτελεί την πρόσοψη του ρολογιού (Εικόνα 31):



Εικόνα 31

Στην συνέχεια, χρησιμοποιείται το Pen Tool(P), για να σχεδιαστεί ένα παρόμοιο σχήμα για την απόδοση εφέ φωτισμού στην αριστερή άκρη του πλαισίου. Το σχήμα χρωματίζεται με σκούρο-γκρι και γραμμική κλίση, χρησιμοποιώντας το εργαλείο Gradient Tool (G), με κάθετο προσανατολισμό. Στην κορυφή του σχήματος τοποθετείται το ανοιχτό/φωτισμένο γκρι και όσο κατευθυνόμαστε προς τη βάση του γίνεται σκούρο γκρι. Έτσι το σχήμα φαίνεται να λοξεύει προς τα έξω (Εικόνα 32):



Εικόνα 32

Κατόπιν, προστίθενται δύο παρόμοια σχήματα γραμμικής κλίσης, ένα για το πάνω μέρος του πλαισίου κι ένα για το κάτω (Εικόνα 33), (Εικόνα 34) :



Εικόνα 33



Εικόνα 34

Για τις γωνίες, σχηματίζονται μικρά ημικυκλικά σχήματα και χρωματίζονται αντίστοιχα με γραμμική κλίση (Εικόνα 35), (Εικόνα 36), (Εικόνα 37):

Κάτω αριστερά:



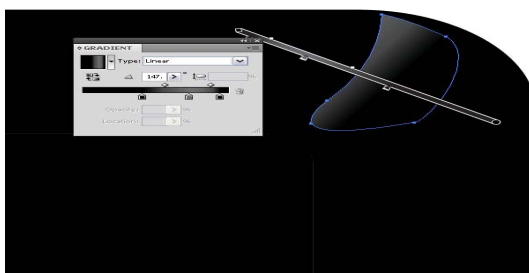
Εικόνα 35

Κάτω δεξιά:



Εικόνα 36

Πάνω δεξιά:



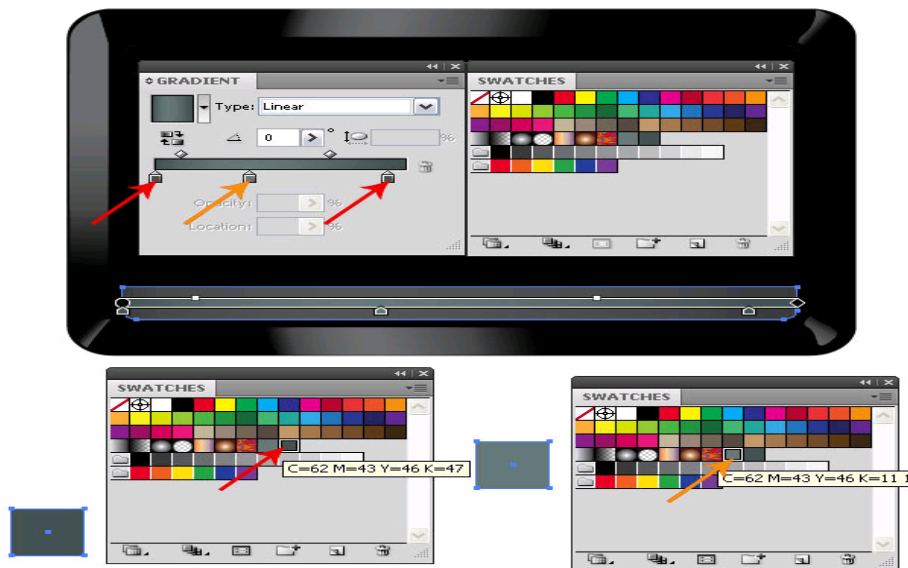
Εικόνα 37

Έτσι, προκύπτει το διπλανό σχήμα του ρολογιού, έχοντας τοποθετήσει όλους τους φωτισμούς (Εικόνα 38):



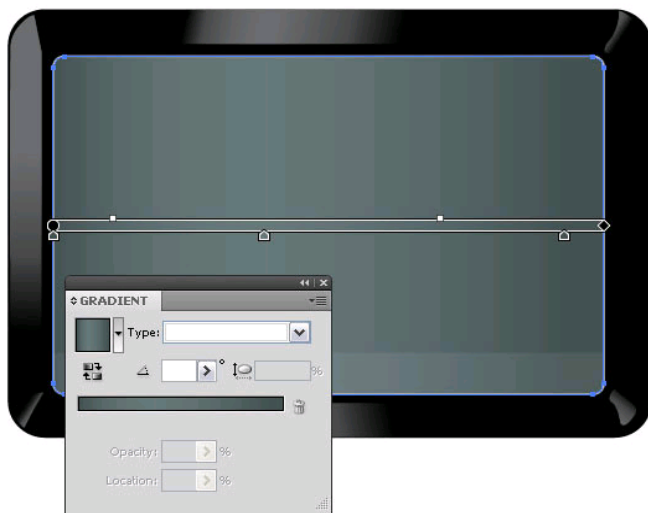
Εικόνα 38

Στην συνέχεια, γίνεται αντιγραφή του εσωτερικού ορθογωνίου, αντιγράφοντας (Cmd / Ctrl + C) και επικολλώντας το μπροστά (Cmd / Ctrl + F). Κόβεται μεγάλο μέρος της νέας επικολλημένης πρόσοψης χρησιμοποιώντας ή το Eraser Tool (Shift + E) ή το Scissors Tool (C) και γεμίζοντας το γραμμικά με ένα σκούρο χρώμα το οποίο θα γίνεται ανοιχτό (Εικόνα 39):



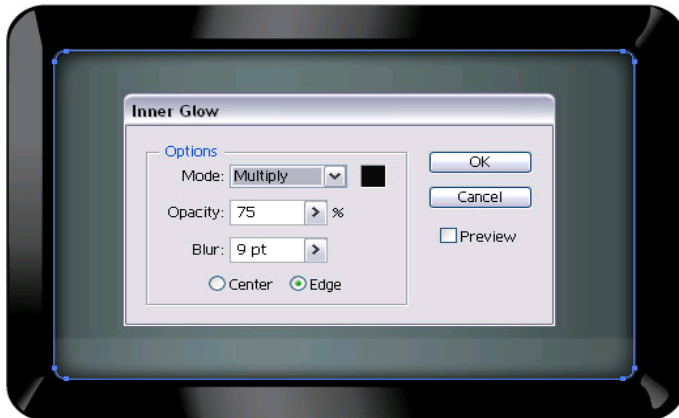
Εικόνα 39

Κατόπιν, σχεδιάζεται το φόντο της πρόσοψης του ρολογιού (Εικόνα 40). Επιλέγεται το εσωτερικό ορθογώνιο (το αρχικό σχήμα), χρωματίζεται με ανοιχτό γκρι με την ίδια γραμμική κλίση που εφαρμόστηκε στο προηγούμενο βήμα:



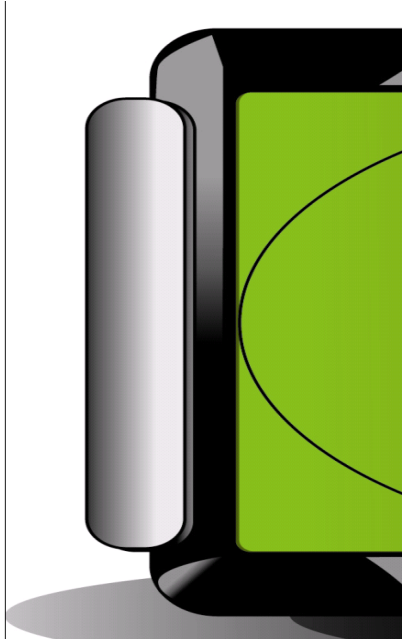
Εικόνα 40

Επιλέγεται το εσωτερικό ορθογώνιο μία εσωτερική λάμψη με τις εντολές Effect > Stylize > Inner Glow. Το εφέ αυτό, προσδίδει στο ρολόι το απαιτούμενο ένθετο στυλ (Εικόνα 41):

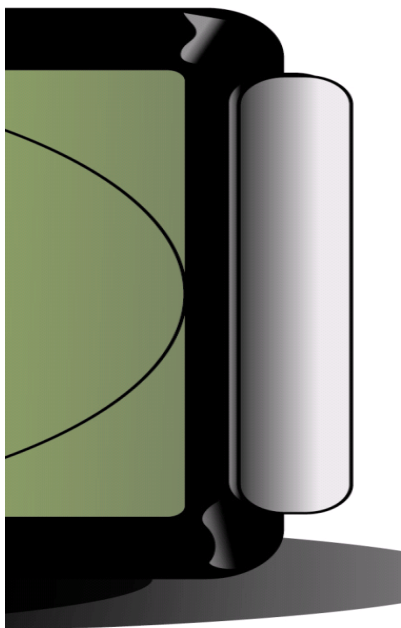


Εικόνα 41

Κατόπιν, σχεδιάζονται οι χειρολαβές δεξιά και αριστερά του πλαισίου του ρολογιού. Δημιουργείται ένα στρογγυλεμένο ορθογώνιο και γεμίζεται με άσπρο-μαύρο χρώμα γραμμικής κλίσης: Γίνεται αντιγραφή του νέου ορθογώνιου (Cmd / Ctrl + C) και επικόλληση πίσω του κάνοντας χρήση του Paste in Back command (Cmd / Ctrl + B). Το επικολλημένο ορθογώνιο, χρωματίζεται με μαύρο (Εικόνα 42). Επιλέγονται τα δύο νέα ορθογώνια μαζί (Cmd / Ctrl + G), γίνεται αντιγραφή (Cmd / Ctrl + C) και επικόλληση (Cmd / Ctrl + P) κι έπειτα γίνεται αντανάκλαση του διπλασιασμένου γκρουπ οριζόντια μέσω του Transform → Reflect. Τέλος μεταφέρεται στο δεξί άκρο του πλαισίου. Έτσι, σχεδιάζεται η δεύτερη (δεξιά) χειρολαβή (Εικόνα 43):

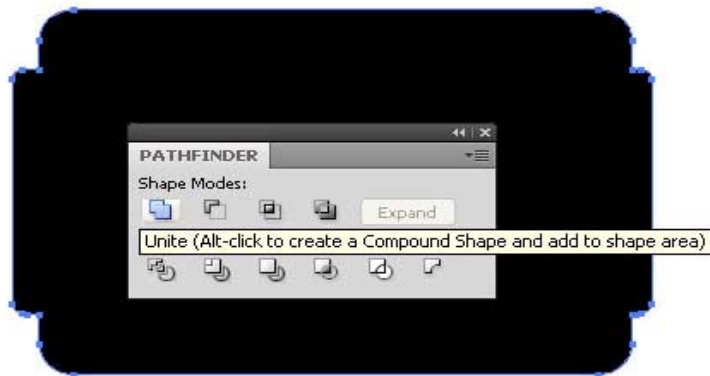


Εικόνα 42



Εικόνα 43

Για το σχεδιασμό του περιγράμματος του ρολογιού, επιλέγονται όλα τα σχήματα, γίνεται αντιγραφή και ομαδοποιούνται. Επιλέγεται το ομαδοποιημένο σχήμα και μέσω της επιλογής Unite στον πίνακα Pathfinder, το σχήμα γεμίζεται με μαύρο χρώμα (Εικόνα 44):



Εικόνα 44

Προστίθεται μία εξωτερική λάμψη (Effect > Stylize > Outer Glow), έπειτα το σχήμα με την εξωτερική λάμψη τοποθετείται πίσω απ' το ρολόι, ώστε να φαίνεται μια περιγραμματική σκιά. Για την δημιουργία αντανάκλασης του φωτός στην πρόσοψη του ρολογιού, επιλέγεται το εσωτερικό στρογγυλεμένο ορθογώνιο και του αφαιρείται ένα ελλειπτικό σχήμα, κάνοντας χρήση του εργαλείου Ellipse Tool (L) κι έπειτα χρησιμοποιείται η εντολή Minus Front στον πίνακα Pathfinder. Το ελλειπτικό σχήμα γεμίζεται μ' ένα ανοιχτό χρώμα και το επίπεδο διαφάνειας ρυθμίζεται σε multiply (Εικόνα 45):



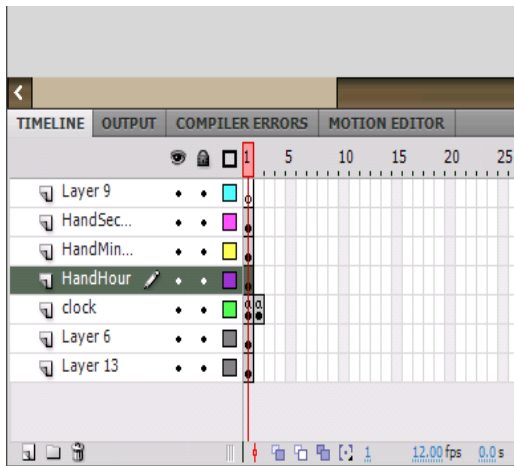
Εικόνα 45

4.2.2 Ρολόι δύο διαστάσεων με μηχανικούς δείκτες

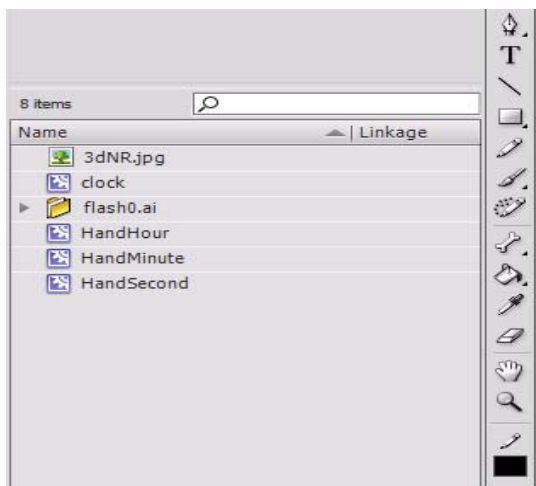
Μετά την ολοκλήρωση του σχεδιασμού της εμφάνισης του ρολογιού στο adobe illustrator, εισάγεται το αρχείο στο adobe flash για την κατασκευή των δεικτών, την εφαρμογή τους στο ρολόι και την εισαγωγή κώδικα ώστε αυτό να λειτουργεί.

ΒΗΜΑ 1: Γίνεται εισαγωγή του αρχείου του ρολογιού σε μορφή εικόνας (.jpg) με τις εντολές File >> Import >> Import to Stage. Στον πίνακα Timeline δημιουργούνται επίπεδα (layers) για 1) το ρολόι-clock 2) τον ωροδείκτη-HandHour 3) τον λεπτοδείκτη-HandMinute 4) τον δευτερολεπτοδείκτη-HandSecond (Εικόνα 46). Για κάθε ένα δημιουργείται ένα σύμβολο μορφής Movie clip, με την εντολή Insert->new symbol όπου δίδονται τα αντίστοιχα ονόματα των layers (

):

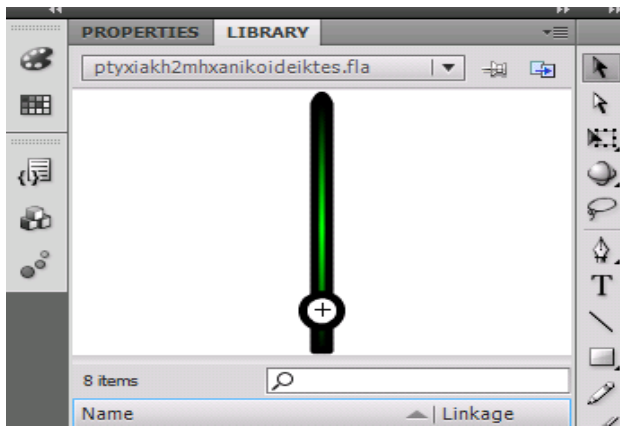


Εικόνα 46

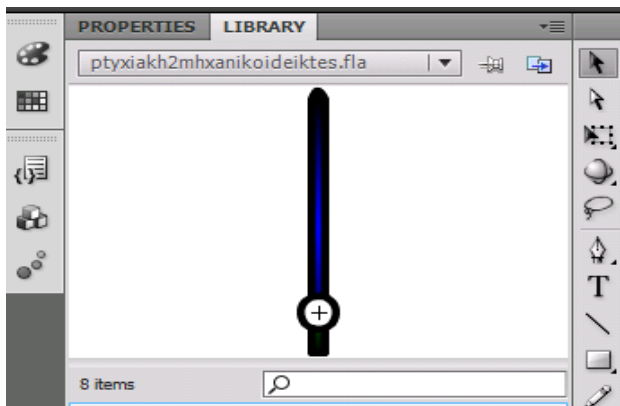


Εικόνα 47

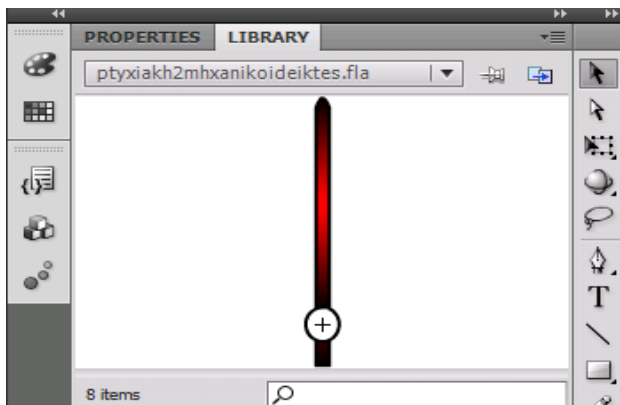
Οι δείκτες του ρολογιού σχεδιάζονται βάσει απλών ορθογωνίων εντός του προγράμματος: στη βάση καθενός σχεδιάζεται ένας κύκλος μ' ένα σταυρό εντός του, που αναπαριστά τον άξονα περιστροφής, ώστε να τοποθετηθούν και οι 3 στο ίδιο σημείο πάνω στο ρολόι. Για την ακριβή τοποθέτησή τους χρησιμοποιείται το εργαλείο align, όπου επιλέγονται ένας δείκτης και το ρολόι μαζί και κατόπιν από το μενού align, τα κουμπιά: align horizontal center και align vertical center (Εικόνα 48), (Εικόνα 49), (Εικόνα 50):



Εικόνα 48



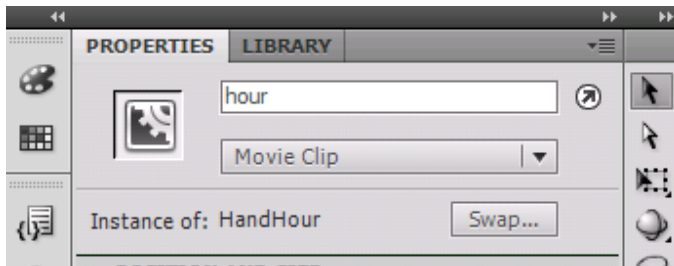
Εικόνα 49



Εικόνα 50

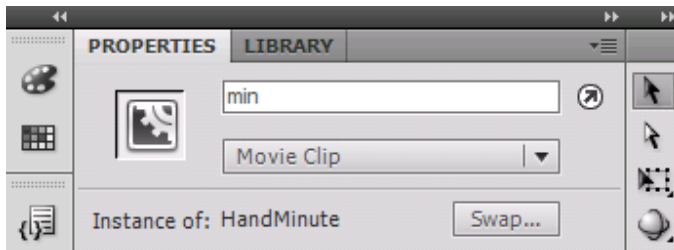
Στην συνέχεια, επιλέγονται ένας προς ένας οι δείκτες από το σχήμα και τους αποδίδονται instances μέσω του πίνακα properties (Εικόνα 51), (Εικόνα 52), (Εικόνα 53).

Για τον ωροδείκτη→hour:



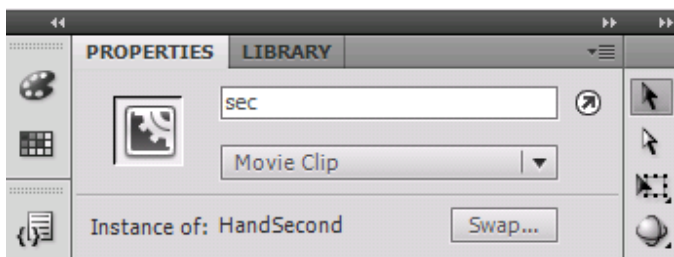
Εικόνα 51

Για τον λεπτοδείκτη→min:



Εικόνα 52

και για τον δευτερολεπτοδείκτη→sec:



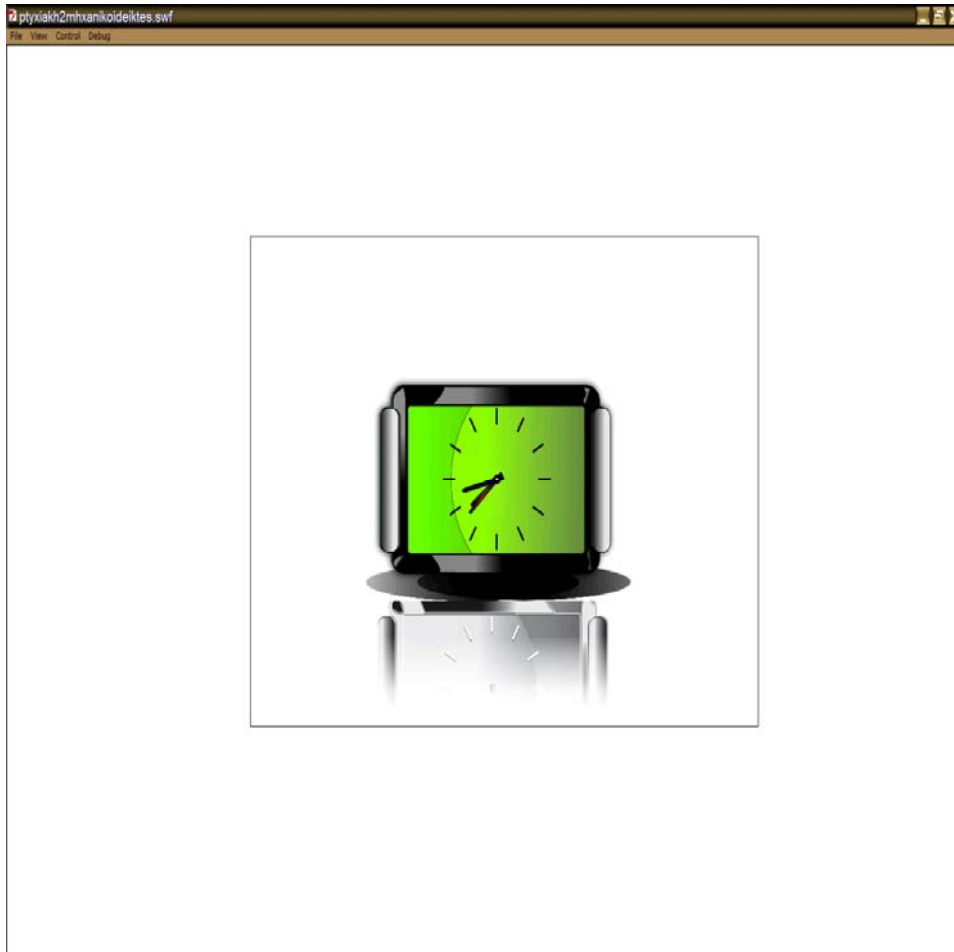
Εικόνα 53

Κώδικας για λειτουργία ρολογιού με μηχανικούς δείκτες:

Σε αυτό το σημείο εισάγεται ο κώδικας για την λειτουργία των δεικτών. Δημιουργείται ένα καινούργιο layer και στο πρώτο frame πατώντας f9 εισάγεται ο παρακάτω κώδικας:

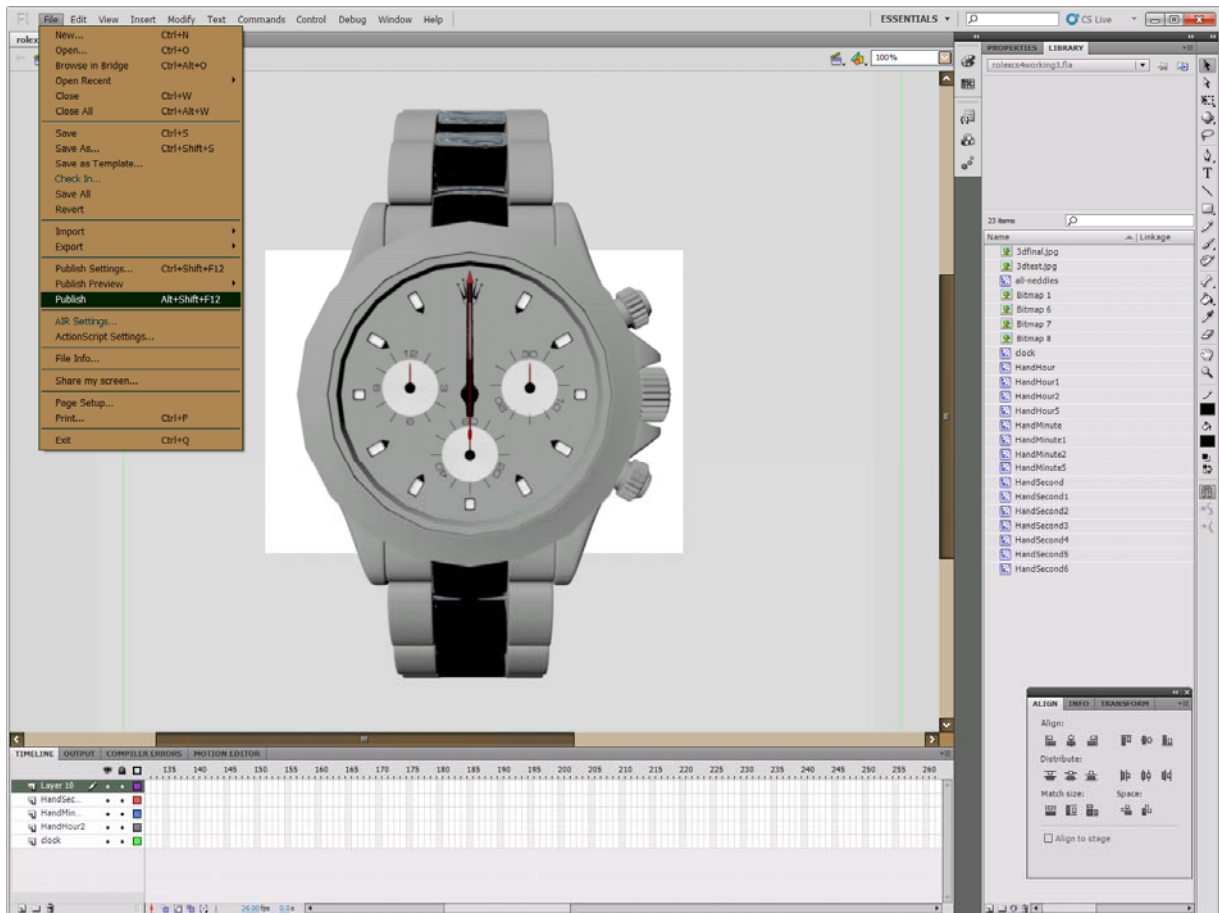
```
abc = new Date();  
a = abc.getHours();  
b = abc.getMinutes();  
c = abc.getSeconds();  
d = abc.getMilliseconds();  
e = a + (b/60) + (c/3600) + (d/3600000);  
f = b + (c/60) + (d/60000);  
g = c + (d/1000);  
setProperty (hour, _rotation, e*30);  
setProperty (min, _rotation, f*6);  
setProperty (sec, _rotation, g*6);  
Επιλέγοντας το δεύτερο frame και  
πατώντας F6 για την δημιουργία  
keyframe εισάγεται ο κώδικας:  
  
gotoAndPlay (1);
```

Το ρολόι έχει ολοκληρωθεί, απομένει να τεθεί σε λειτουργία, αυτό επιτυγχάνεται πατώντας CONTROL+ENTER εντός του flash player. Παρακάτω φαίνεται το ρολόι με μηχανικούς δείκτες σε λειτουργία, βάσει της σωστής ώρας που αντλεί από τα windows (Εικόνα 54):

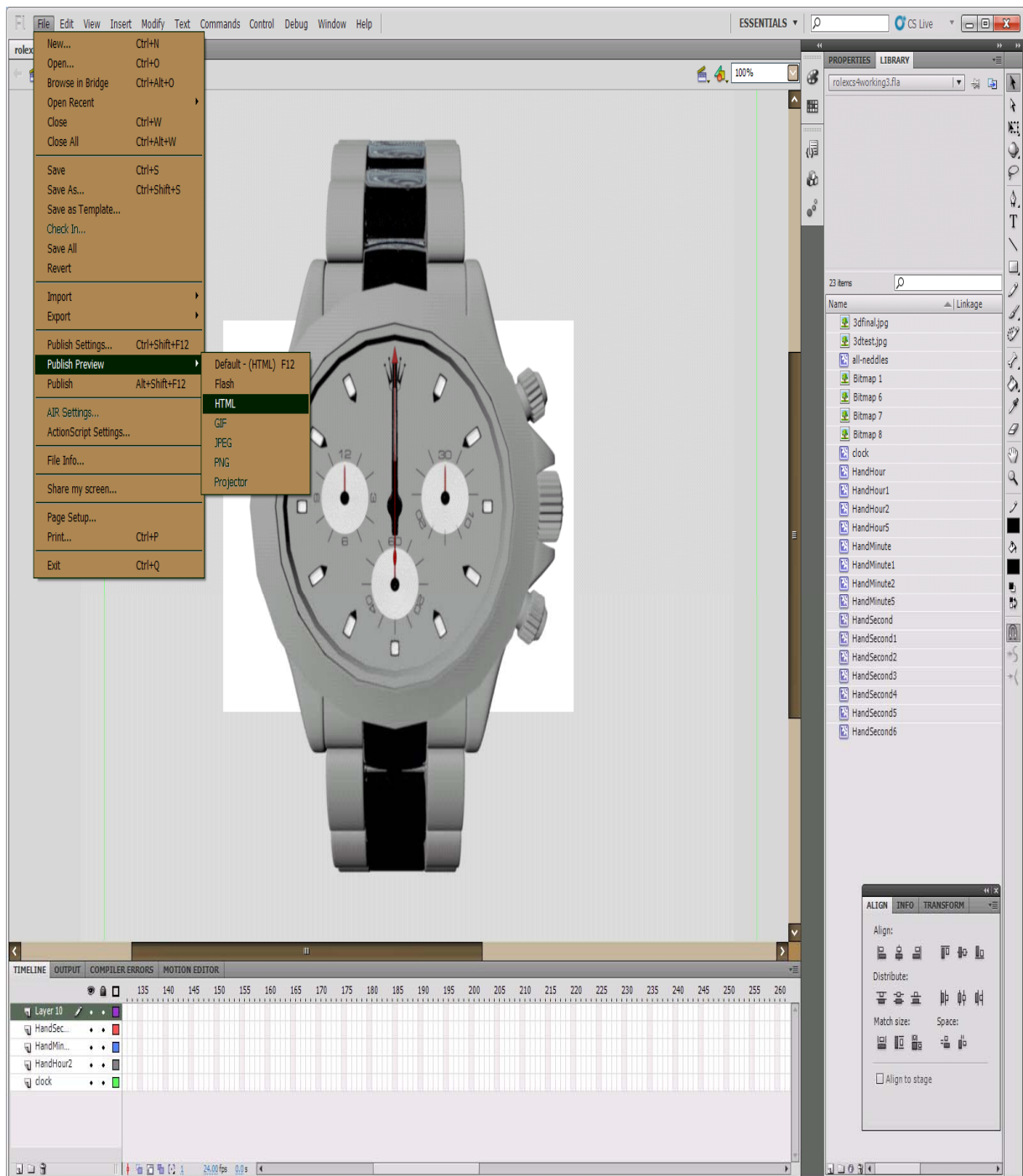


Εικόνα 54

Εν τέλει, περιγράφεται η διαδικασία εξαγωγής του Flash movie στο διαδίκτυο: Για την προβολή μιας εφαρμογής Flash στο διαδίκτυο είναι απαραίτητη η διάθεση μιας ιστοσελίδας (Html), στην οποία θα «στεγαστεί» η εφαρμογή, το είδος της εφαρμογής (βίντεο, ήχος, εικόνα, animation κτλ) καθώς και όλοι οι τύποι αρχείων που χρησιμοποιεί ο Flash Player. Εντός του προγράμματος, η διαδικασία αυτή επιτυγχάνεται με τις εντολές File→Publish (Εικόνα 55), με τις οποίες ο flash δημιουργεί κάποιους τύπους αρχείων. Στην προκειμένη εφαρμογή, δημιουργούνται δύο τύποι αρχείων; 1) .swf, ο οποίος περιέχει το animation του ρολογιού σε λειτουργία βάσει της σωστής ώρας και 2) .html ο οποίος εισαγάγει το animation στην συγκεκριμένη ιστοσελίδα. Για την προβολή του αποτελέσματος, επιλέγεται: File→Publish Preview→Html (Εικόνα 56).



Εικόνα 55



Εικόνα 56

Ρολόι δύο διαστάσεων με δείκτες ταλάντωσης ανά δευτερόλεπτο

Στην εφαρμογή αυτή, χρησιμοποιήθηκε η ίδια μεθοδολογία με το παράδειγμα ρολογιού με μηχανικούς δείκτες, μέσω των ίδιων προγραμμάτων adobe illustrator και adobe flash. Η μόνη διαφορά έγκειται στην αλλαγή του κώδικα, ο οποίος διαμορφώνεται ως εξής:

```
this.onEnterFrame = function()  
{  
var date = new Date();  
var hours = date.getHours();  
var minutes = date.getMinutes();  
var seconds = date.getSeconds();  
min._rotation = 360 / 60 * minutes;  
hour._rotation = (hours*30) +  
(minutes/2);  
sec._rotation = 360 / 60 * seconds; }
```

4.2.3 Ρολόι Rolex τρισδιάστατης γεωμετρίας δισδιάστατης απεικόνισης με μηχανικούς δείκτες.

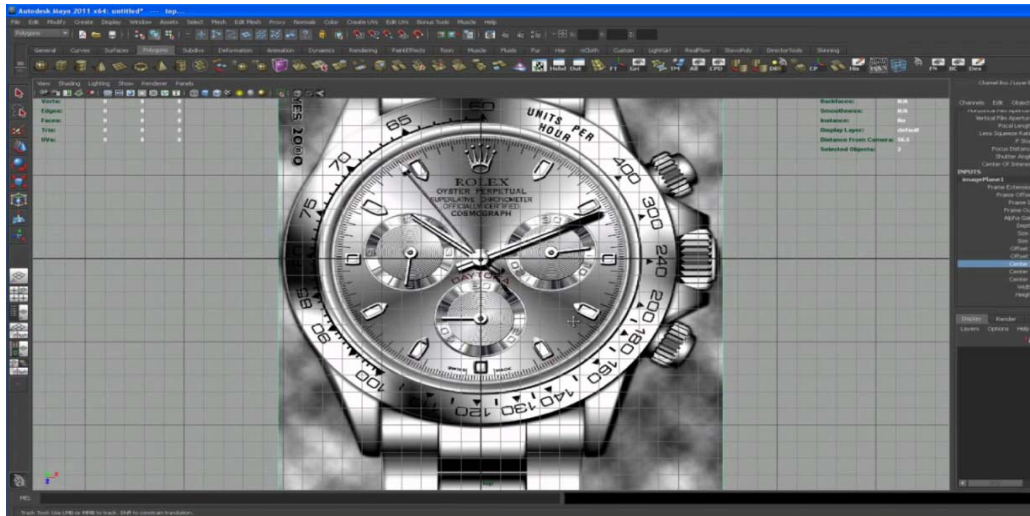
Στην εφαρμογή αυτή, πραγματοποιήθηκε η λειτουργία ενός ρολογιού Rolex τρισδιάστατης γεωμετρίας, σε δισδιάστατη απεικόνιση μέσω των εργαλείων του προγράμματος Adobe flash, όπου εφαρμόστηκε η ίδια μεθοδολογία με τα προηγούμενα παραδείγματα ρολογιών, με μόνη διαφορά ότι ο σχεδιασμός του ρολογιού πραγματοποιήθηκε στο Autodesk Maya (3D λογισμικό), έγινε εξαγωγή του αρχείου σε μορφή εικόνας (.jpg) και εισαγωγή του αρχείου του ρολογιού και των δεικτών του (έναν προς έναν ξεχωριστά) στο Adobe Flash. Εκεί, εφαρμόστηκαν οι δείκτες στο ρολόι με τα εργαλεία του προγράμματος (Εικόνα 57) :



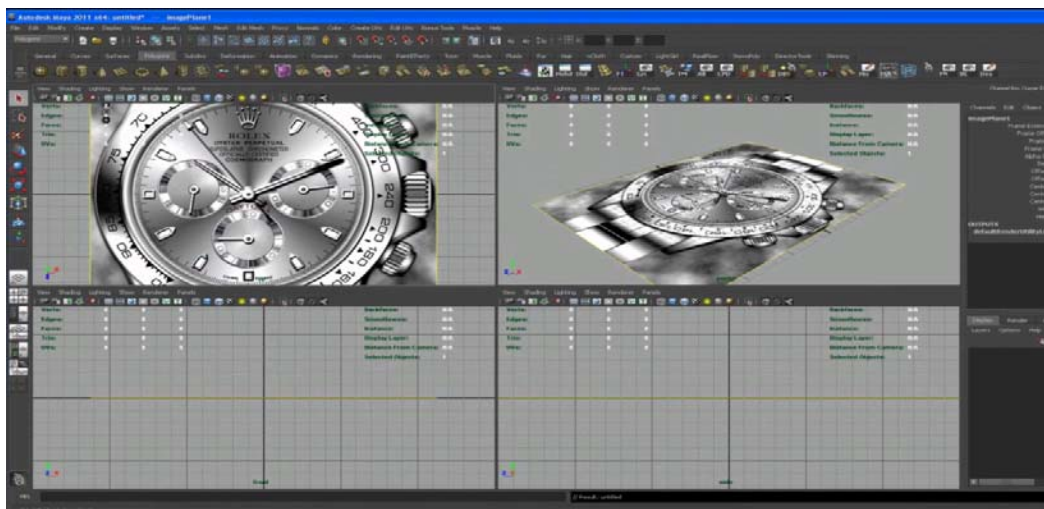
Εικόνα 57

4.3 Εφαρμογή σε Autodesk Maya

Το λογισμικό Autodesk Maya, χρησιμοποιήθηκε για τον σχεδιασμό της τρισδιάστατης γεωμετρίας του ρολογιού Rolex Daytona. Στην συνέχεια, περιγράφονται τα βασικά βήματα σχεδιασμού του ρολογιού και παρατίθενται κάποιες εικόνες μέσα από το πρόγραμμα οι οποίες απεικονίζουν το «χτίσιμο» των ενδιάμεσων σταδίων κατασκευής της γεωμετρίας του ρολογιού. Αρχικά έγινε εισαγωγή μια ρεαλιστικής φωτογραφίας του Rolex σε πρόσοψη, η οποία θα βοηθήσει στην σωστή τοποθέτηση, τον καθορισμό του μεγέθους και γενικότερα την απόδοση ρεαλιστικών προδιαγραφών του σχεδιασμού του ρολογιού: (Εικόνα 58) (Εικόνα 59).



Εικόνα 58



Εικόνα 59

Παρακάτω περιγράφονται τα βασικά βήματα που απαιτούνται για τον σχεδιασμό του ρολογιού:

ΒΗΜΑ 1: Εισαγωγή και στήσιμο της σκηνής.

Αρχικά, εξασφαλίζονται κάποια αρχεία απ' το διαδίκτυο, τα οποία είναι σημαντικά για την εφαρμογή:

- Το αρχείο microgramma D Medium extended font, το οποίο αποτελεί ένα φόντο για την δημιουργία γραμμάτων.
- Το αρχείο Baskerville old face, το οποίο αποτελεί ένα φόντο για την δημιουργία γραμμάτων και αριθμών.
- Μια εικόνα για το Rolex Daytona σε πρόσοψη, για τον ακριβή σχεδιασμό του ρολογιού. Η εικόνα αποθηκεύεται σε ειδικό φάκελο του προγράμματος με τίτλο SourcelImages folder. Για τον σχεδιασμό ορισμένων εξαρτημάτων του ρολογιού χρησιμοποιήθηκαν εικόνες διαφορετικών όψεων από το διαδίκτυο.
- Το αρχείο UV layout, για την δημιουργία υφών σε κάποια σημεία της γεωμετρίας του ρολογιού.

Στην συνέχεια, εντός του προγράμματος επιλέγεται File→New Project για την δημιουργία νέου σχεδίου και εισάγεται στο πρόγραμμα η εικόνα του ρολογιού επιλέγοντας view→Image Plane →Import Image απ' τον φάκελο SourcelImages .

ΒΗΜΑ 2: Μοντελοποίηση της κύριας επιφάνειας.

Η μοντελοποίηση της κύριας επιφάνειας χωρίζεται σε:

1. Σχεδιασμός των υπό-καντράν και της επιφάνειάς τους

Επιλέγεται ένας κύλινδρος με 20 υποδιαιρέσεις άξονα (subdivision axis=20), διαγράφονται το πάνω και το κάτω μέρος του και εφαρμόζεται σ' ένα από τα καντράν της εικόνας του ρολογιού. Στην συνέχεια, διαμορφώνεται η πάνω εσωτερική επιφάνειά του (Εντολές edge loops,extrude), γίνεται αντιγραφή του κυλίνδρου και περιστροφή του

στον άξονα Ψ ώστε να επικολληθεί στα άλλα 2 υπό-καντράν. Ακολουθεί η ένωση των τριών κυλίνδρων (Εντολή: `append polygon tool`). Τέλος, χτίζεται η υπόλοιπη εσωτερική επιφάνεια δημιουργώντας κύλινδρο με 32 υποδιαιρέσεις ως εξωτερικό περίβλημα των υπό-καντράν, ώστε να ενωθεί με αυτά με το χτίσιμο της μεταξύ τους γεωμετρίας (Εντολή: `Edit Mesh→Bridge`) (Εικόνα 60-Σχεδιασμός των υπό-καντράν), (Εικόνα 61-Δημιουργία της επιφάνειας των υπό-καντράν), (Εικόνα 62- Δημιουργία της επιφάνειας των υπό-καντράν).

2. Σχεδιασμός των δεικτών των υπό-καντράν

Επιλέγεται ένας κύλινδρος με 20 υποδιαιρέσεις αξόνων, διαγράφεται όλο το μέρος του κυλίνδρου πλην της κορυφής του, εφαρμόζεται στο εσωτερικό κύκλο των υπό-καντράν και διαμορφώνεται η επιφάνεια του (Εντολές:control+right click →edge ring utilities→edge ring split, Inputs→split type:multi divisions:26, Extrude) . Γίνεται αντιγραφή, περιστροφή και επικόλληση της επιφάνειας στα άλλα δυο υπό-καντράν. Επιλέγεται νέος κύλινδρος, εφαρμόζεται στο κέντρο ενός υπό-καντράν και γίνεται η διαμόρφωση και επεξεργασία του δείκτη (Εντολές:Extrude,Edge loop tool,edge ring utilities→edge ring split). Κατόπιν, σχεδιάζεται το εσωτερικό δακτυλίδι στο κέντρο του δείκτη χρησιμοποιώντας μία polygon pipe (Εντολές:edge loop tool,Mesh→fill hole, Edit Mesh→Poke face, control+right click →edge ring utilities→edge ring split) (Εικόνα 63-Σχεδιασμός των δεικτών των υπό-καντρά).

3. Σχεδιασμός των αριθμών

Για τον σχεδιασμό των αριθμών που αναγράφονται στα υπό-καντράν, χρησιμοποιήθηκε το φόντο Microgramma D extended font (Εντολές: Create→Text→text curve options). Γίνονται κάποιες διαμορφώσεις στις επιφάνειες των αριθμών ώστε να είναι πιο ρεαλιστικοί (Εντολές:Edit Mesh→Delete Edge/Vertex). Στην συνέχεια, αφού εφαρμοστεί στην εικόνα του ρολογιού ο πρώτος αριθμός που σχεδιάζεται, σχεδιάζονται και οι υπόλοιποι και τοποθετούνται στις αντίστοιχες θέσεις τους βάσει της εικόνας, με περιστροφή τους σε σχέση με τον πρώτο αριθμό. Τέλος, χρησιμοποιώντας την πλάγια όψη του προγράμματος, οι αριθμοί τοποθετούνται στο ίδιο ύψος με τα εσωτερικά δακτυλίδια των δεικτών των υπό-καντράν (Εικόνα 64-Σχεδιασμός των αριθμών).

4. Σχεδιασμός των 12 κουτιών

Στο σημείο αυτό, περιγράφεται ο σχεδιασμός των σημείων των αριθμών 6,9,3 οι οποίοι αντιπροσωπεύουν τις αντίστοιχες ώρες. Επιλέγεται έναν κύβος, εφαρμόζεται στην αντίστοιχη θέση στην εικόνα του ρολογιού και διαγράφεται η πίσω του πλευρά. Στην συνέχεια διαμορφώνεται η επιφάνεια του, που θα είναι ορατή στο ρολόι (Εντολές:Extrude, edge loop tool, control+right click →edge ring utilities→edge ring split,scale,Mesh→Fill Hole). Για τον σχεδιασμό των υπόλοιπων δύο κουτιών, γίνεται περιστροφή του κουτιού που δημιουργήθηκε (αριθμός 6) κατά 90 μοίρες (αριθμός 3) και κατά -90 μοίρες (αριθμός 9) (Εντολές:Edit→Duplicate Special tool Options). Για τον σχεδιασμό των υπόλοιπων κουτιών (αριθμοί 7,8,10,11,1,2,4,5) τα οποία έχουν το σχήμα βάρκας, γίνεται αντιγραφή ενός απ' τα κουτιά που δημιουργήθηκε παραπάνω και επεξεργασία του, ώστε η γεωμετρία του να συμφωνεί με τα κουτιά τις εικόνες (Εντολές:scale,edge loop tool). Κατόπιν, αφού δημιουργηθεί ένα απ' αυτά, αντιγράφεται και περιστρέφεται στις αντίστοιχες θέσεις με την εικόνα (Εντολές:Edit→Duplicate Special tool Options) (Εικόνα 65-Σχεδιασμός των κουτιών των αριθμών).

5. Σχεδιασμός του λογότυπου (Rolex)

Επιλέγεται μία polygon pipe, εφαρμόζεται στο λογότυπο της εικόνας ώστε να πάρει το σχήμα της (Εντολές:scale,delete edge) και διαγράφεται το μισό του μέρος. Στην συνέχεια, σχεδιάζονται οι ακτίνες του λογότυπου (Extrude,Inputs→Keep faces together→Off, control+right click →edge ring utilities→edge ring split) και τέλος σχεδιάζονται οι μικρές μπάλες πάνω στις ακτίνες, επιλέγοντας μία σφαίρα με 8 υποδιαίρεσεις άξονα. Η μπάλα εφαρμόζεται πάνω στην εικόνα για να πάρει τις διαστάσεις της, γίνεται αντιγραφή της, επικόλληση και περιστροφή της στα υπόλοιπα σημεία (Εντολές:Scale,Control+D,Move,Rotate,Combine). Έπειτα, ενώνονται οι ακτίνες με τις μπάλες (Εντολή:Edit Mesh→Bridge tool). Τέλος, γίνεται αντιγραφή του

μισού κομματιού που έχει δημιουργηθεί και επικόλληση ώστε να ολοκληρωθεί το σχήμα (Εντολές:Control+D,Scale,Merge, control+right click →edge ring utilities→edge ring split) (Εικόνα 66-Σχεδιασμός του λογότυπου).

6. Σχεδιασμός του δευτερολεπτοδείκτη

Επιλέγεται ένας κύλινδρος με 8 αξονικές υποδιαίρεσεις κι εφαρμόζεται στην θέση του δευτερολεπτοδείκτη της εικόνας (Εντολές:Scale,Rotate). Στην συνέχεια επιλέγεται ένας κύβος, εφαρμόζεται στην πάνω πλευρά του κυλίνδρου και στο ίδιο ύψος με αυτόν (Εντολές:Scale) και ενώνονται (Εντολές:Combine>Edit Mesh→Bridge). Κατόπιν, διαμορφώνεται η επιφάνεια του κύβου ώστε να σχηματίσει το επιθυμητό βέλος του δευτερολεπτοδείκτη (control+right click →edge ring utilities→edge ring split,edge loop tool,Scale,Rotate). Για τον σχεδιασμό του πίσω μέρους του δευτερολεπτοδείκτη, επιλέγεται ένας κύλινδρος και εφαρμόζεται στην κατάλληλη θέση στην εικόνα και στο ίδιο ύψος με τον πρώτο κύλινδρο (Εντολές:Move,Rotate,Scale, control+right click →edge ring utilities→edge ring split) και ενώνεται με αυτόν (Εντολές>Edit Mesh→Bridge). Κατόπιν, σχεδιάζεται ο κεντρικός άξονας περιστροφής των δεικτών, επιλέγοντας έναν ακόμη κύλινδρο και εφαρμόζοντάς τον στο κέντρο του ρολογιού και σύμφωνα με την εικόνα (Εντολές: Scale, Move, control+right click →edge ring utilities→edge ring split). Προστίθεται, τέλος, ένα εσωτερικό δαχτυλίδι στο κέντρο του άξονα περιστροφής, επαναλαμβάνοντας την διαδικασία του σχεδιασμού των δεικτών των υπό-καντράν (Εικόνα 67-Σχεδιασμός του δευτερολεπτοδείκτη).

7. Σχεδιασμός του λεπτοδείκτη

Επιλέγεται ένα polygon plane πλάτους=5, υποδιαίρεσης πλάτους=5 και υποδιαίρεσης ύψους=1. Διαγράφεται το μισό μέρος του, διαμορφώνεται έτσι ώστε να συμφωνεί με τον λεπτοδείκτη της εικόνας (Εντολές:Move,Rotate, control+right click →edge ring utilities→edge ring split,Extrude) και εφαρμόζεται στην κατάλληλη θέση ώστε να πάρει

το σχήμα και τις διαστάσεις της εικόνας. Δημιουργείται, στην συνέχεια, ο άξονας περιστροφής του λεπτοδείκτη, επιλέγοντας έναν κύλινδρο με 17 αξονικές υποδιαιρέσεις κι εφαρμόζοντάς τον στην εικόνα. Κατόπιν ενώνεται μέρος του άξονα περιστροφής με τον λεπτοδείκτη και διαγράφεται το περιττό μέρος του κυλίνδρου (Εντολές: Split polygon tool, Split edge ring tool, Merge, Edge loop tool). Κατόπιν, διπλασιάζεται το μισό μέρος του λεπτοδείκτη ώστε να ολοκληρωθεί η γεωμετρία του (Εντολές: Control+D, Rotate, Combine, Merge) (Εικόνα 68-Σχεδιασμός του λεπτοδείκτη).

8. Σχεδιασμός του ωροδείκτη

Ο σχεδιασμός του ωροδείκτη υλοποιήθηκε, ακολουθώντας την ίδια διαδικασία με το σχεδιασμό του λεπτοδείκτη (Εικόνα 69-Σχεδιασμός του ωροδείκτη).

9. Σχεδιασμός των εξωτερικών δακτυλιδιών και της κάσας

Για τον σχεδιασμό του πρώτου εξωτερικού δακτυλιδιού, επιλέγεται μία polygon ripec με 24 αξονικές υποδιαιρέσεις, εφαρμόζεται στην κατάλληλη θέση στην εικόνα (Εντολές: Move, Scale) και διαμορφώνεται ώστε συμφωνεί με το δαχτυλίδι της εικόνας (Εντολές: Extrude, Edge loop tool). Για τον σχεδιασμό του δεύτερου εξωτερικού δακτυλιδιού, επιλέγεται μία polygon ripec 20 αξονικών υποδιαιρέσεων, εφαρμόζεται στο ίδιο ύψος με το πρώτο δαχτυλίδι και διαμορφώνεται αντίστοιχα η γεωμετρία του (Εντολές: Move, Scale, Edge loop tool). Για τον σχεδιασμό της κάσας, επιλέγεται ένας κύλινδρος κι ένας κύβος με 1 υποδιαίρεση ύψους και 28 υποδιαιρέσεις βάθους, οι οποίοι εφαρμόζονται στην εικόνα στην θέση της κάσας του ρολογιού (Εντολές: Move, Scale). Διαγράφονται τα $\frac{3}{4}$ του κύβου, ώστε να σχεδιαστεί το ένα μέρος της κάσας και αφού ολοκληρωθεί να επικολληθεί στα υπόλοιπα και κατόπιν διαμορφώνεται το κομμάτι αυτό του κύβου ώστε να συμφωνήσει με το σχήμα της εικόνας (Εντολές: Edge loop tool, control+right click → edge ring utilities → edge ring split). Επιλέγεται μια καμπυλωτή γραμμή ώστε να σχεδιαστεί το πόδι της κάσας

(Εντολές:Create→CV curve tool). Αφού γίνει επεξεργασία στην καμπυλωτή γραμμή και πάρει το σχήμα του ποδιού διαμορφώνεται βάσει αυτής το κομμάτι του κύβου (Εντολές:Extrude,Append polygon tool, control+right click →edge ring utilities→edge ring split, Delete edge, Animation mode:Create deformers→lattice, Delete history, Edge loop tool). Στην συνέχεια, το ολοκληρωμένο κομμάτι της κάσας αντιγράφεται ώστε να συμπληρωθεί μισό μέρος της κάσας (Εντολές:Control+D, Scale, Combine, Merge). Ακολουθεί ο σχεδιασμός του πλάγιου μέρους της κάσας (Εντολές: Extrude, control+right click →edge ring utilities→edge ring split ,Create deformers→lattice, Move, Delete history). Τέλος, αφού διαμορφωθεί το μισό μέρος της κάσας, αντιγράφεται και διπλασιάζεται ώστε να ολοκληρωθεί η κάσα (Εντολές:Combine,Control+D,Merge,Scale) (Εικόνα 70-Σχεδιασμός των εξωτερικών δακτυλιδιών και της κάσας).

10. Σχεδιασμός των κουμπιών

Επιλέγεται ένας κύλινδρος με 24 αξονικές υποδιαιρέσεις και εφαρμόζεται στην κατάλληλη θέση στην εικόνα (Εντολές:Move, Scale). Ακολουθεί η διαμόρφωση της γεωμετρίας του (Εντολές:Extrude, Inputs→Keep face together: off, Scale, Move, control+right click →edge ring utilities→edge ring split, Paint select tool). Προστίθεται στο κουμπί το λογότυπο του Rolex με αντιγραφή του απ' την εσωτερική επιφάνεια του ρολογιού και επικόλλησή του στο κουμπί (Εντολές:Control+D, Scale, Move). Τα υπόλοιπα κουμπιά σχεδιάζονται με παρόμοιο τρόπο, με μικρές διαφορές στο μέγεθος και το σχήμα (Εικόνα 71-Σχεδιασμός των κουμπιών).

11. Σχεδιασμός των δακτυλίων της βάσης

Επιλέγεται μία polygon pipe και εφαρμόζεται κάτω απ' την κάσα του ρολογιού (Εντολές: Move, Scale). Διαγράφεται η πλευρά του δακτυλιδιού που εφαρμόζει στην κάσα με ακρίβεια. Κατόπιν διαμορφώνεται η γεωμετρία του δακτυλιδιού ώστε να συμφωνεί με την εικόνα (Εντολές: Edge loop tool). Για το σχεδιασμό του δεύτερου δακτυλίου της βάσης, επιλέγεται μία polygon pipe, με 140 αξονικές υποδιαιρέσεις και εφαρμόζεται κάτω απ' το πρώτο δακτυλίδι (Εντολές: Scale, Move). Ακολουθεί η διαγραφή της πλευράς που εφαρμόζει στο πρώτο δακτυλίδι και η διαμόρφωσή του (Εντολές: Extrude, Inputs→Keep faces together: off, Scale, control+ right click →edge ring utilities→edge ring split, Pint select tool, Control+ scale(scale proportionally), Extrude) (Εικόνα 72-Σχεδιασμός των δακτυλίων της βάσης).

12. Σχεδιασμός του λουριού

Επιλέγεται μια καμπυλωτή γραμμή ώστε να δοθεί το αρχικό σχήμα το λουριού (Εντολές: Create→CV curve tool, Move, Scale, Surfaces mode→Edit curves→open/close curves). Στην περίπτωση αυτή, όπου δεν υπάρχει ειδική όψη μέσω κάποιας εικόνας, ο σχεδιασμός της καμπυλωτής γραμμής γίνεται περισσότερο με το μάτι, εκτιμώντας τις διαστάσεις του λουριού από κάποια εικόνα άλλης όψης. Ακολουθεί η αντιγραφή και επικόλληση της καμπυλωτής γραμμής σε δύο ακόμη σημεία, δεξιά και αριστερά ώστε να «χτιστεί» το πρώτο μισό κομμάτι του λουριού (Εντολές: Surfaces→Loft). Κατόπιν διαμορφώνεται το πρώτο κομμάτι του λουριού ώστε να εφαρμόζει στο κατάλληλο σημείο της εικόνας, αυτό επιτυγχάνεται με την μετατόπιση της καμπυλωτής γραμμής στα σημεία της εικόνας όπου απεικονίζεται το λουρί και την επεξεργασία της (Εντολές: Control+D, Move, Scale, Control+right click→Control vertex, Surfaces→loft). Κατόπιν, είναι απαραίτητη η μετατροπή του κομματιού του λουριού που έχει δημιουργηθεί μέχρι τώρα, από επιφάνεια (Surface)

σε πολύγωνα (Polygons) (Εντολές: Modify→convert→Nurbs to polygons, Edit mesh→Poke face, Extrude, Scale, edge loop tool, Move, Create deformers→Lattice, Delete history, Scale). Στην συνέχεια, μετά την ολοκλήρωση της γεωμετρίας στο πρώτο μισό κομμάτι του λουριού, γίνεται διπλασιασμός του ώστε να ολοκληρωθεί το πρώτο κομμάτι (Εντολές:Control+D , Scale, Combine, Merge, Edit→Duplicate special tool). Για τον σχεδιασμό του υπόλοιπου λουριού χρησιμοποιείται μία ακόμη καμπυλωτή γραμμή με 16 τομείς (Εντολές: Curves→ Nurbs circle) και διαμορφώνεται ώστε να πάρει το σχήμα του κομματιού (Εντολές: Scale, Move, Control+right click→control vertex, Animation mode: Create deformers→nonlinear deformer→bend, Rotate, Delete history, Move). Αφού ολοκληρωθεί ο σχεδιασμός της επιθυμητής καμπυλωτής γραμμής, ακολουθεί το «χτίσιμο» του δεύτερου κομματιού του λουριού επαναλαμβάνοντας τα βήματα που έγιναν στον σχεδιασμό του πρώτου κομματιού. Έπειτα, γίνεται αντιγραφή του σε τέσσερα ίδια κομμάτια ώστε να συμπληρωθεί ο επιθυμητός αριθμός των έξι κομματιών (Εντολές:Control+D, Edit→Duplicate special tool) και ο διπλασιασμός τους ώστε να προκύψει το ολοκληρωμένο λουρί της μίας πλευράς του ρολογιού (Εντολές:Control+G(group item), Control+D, Scale, Create deformers→nonlinear deformer→bend, Rotate, Move). Τέλος, το ολοκληρωμένο λουρί διπλασιάζεται στην δεύτερη πλευρά του ρολογιού (Εντολές: Control+G, Edit→Duplicate special) (Εικόνα 73-Σχεδιασμός του λουριού).

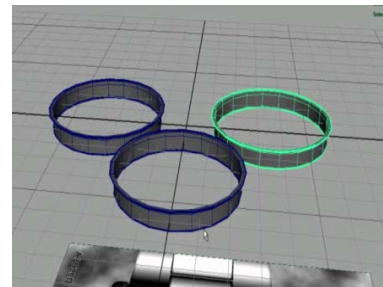
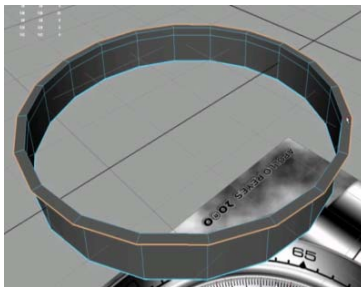
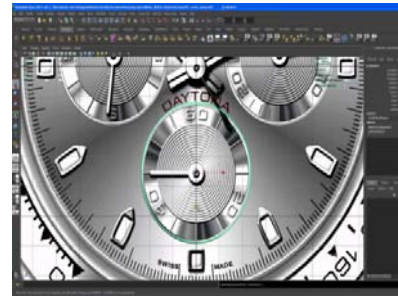
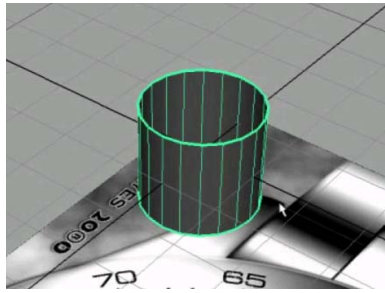
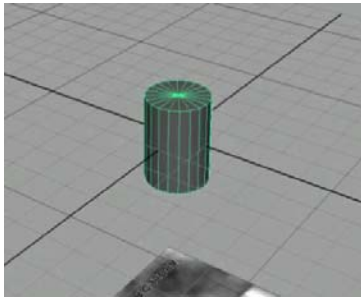
13. Σχεδιασμός της πόρπης ασφαλείας

Επιλέγεται ένας κύβος με 2 υποδιαιρέσεις πλάτους, 1 υποδιαίρεση ύψους και 20 υποδιαιρέσεις βάθους, εφαρμόζεται στο ύψος της βάσης του λουριού (Εντολές: Scale, Move). Ακολουθεί ο σχεδιασμός της μισής πόρπης ασφαλείας και η διαμόρφωση της γεωμετρίας του κύβου. (Εντολές: control+ right click →edge ring utilities→edge ring split, Paint select tool, Extrude, Move, Create deformers→nonlinear deformer→bend, Rotate). Προστίθενται 7 μικρές τρύπες στα πλάγια της πόρπης ασφαλείας, επιλέγοντας έναν κύλινδρο με 8 αξονικές

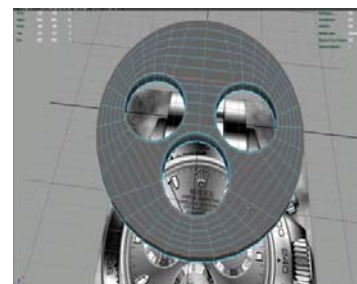
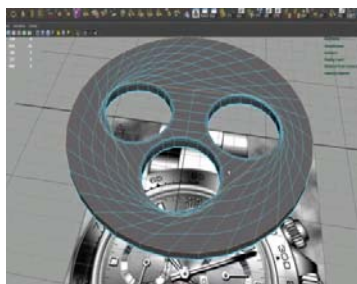
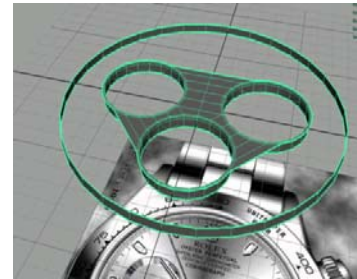
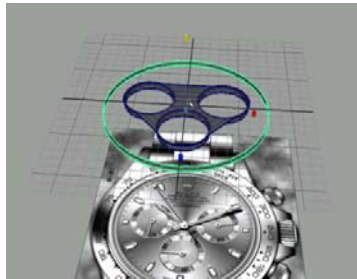
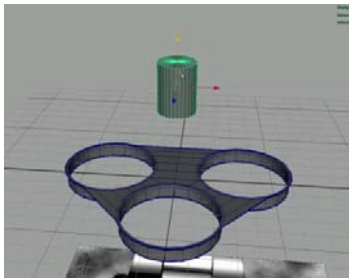
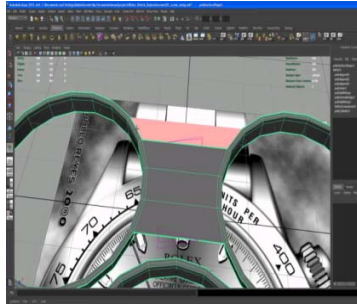
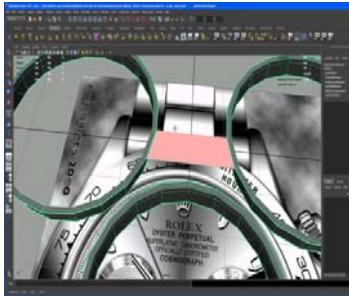
υποδιαιρέσεις και διαγράφοντας τις πλευρές της βάσης και της κορυφής του (Εντολές: Move, Scale, Rotate, Edge loop tool, Combine, Normals→reverse, Edit Mesh→Bridge, control+ right click →edge ring utilities→edge ring split, Edit Duplicate special tool). Διπλασιάζεται το κομμάτι ώστε να προκύψει η ολοκληρωμένη πόρπη ασφαλείας (Εντολές: Control+D, Scale). Κατόπιν, σχεδιάζονται τα πλαϊνά κομμάτια ένωσης της πόρπης με το λουρί, επιλέγοντας έναν κύβο και εφαρμόζοντας τον στην μία άκρη της πόρπης ασφαλείας (Εντολές: Move, Scale, Rotate). Ακολουθεί η διαμόρφωση του κομματιού (Εντολές: Edge loop, Extrude, Control+ right click →edge ring utilities→edge ring split, Rotate, Create deformers→nonlinear deformer→bend, Merge). Σχεδιάζεται οι σύνδεσμοι στην άκρη του λουριού επιλέγοντας έναν κύλινδρο και διαμορφώνοντας τον (Εντολές: Move, Scale, Rotate, Control+ right click →edge ring utilities→edge ring split) και αντιγράφοντας τον ώστε να σχηματιστεί ο δεύτερος σύνδεσμος (Εντολές: Control+D, Move). Για την ένωση των συνδέσμων με την πόρπη, διαγράφονται οι πλευρές εφαρμογής του συνδέσμου και της πόρπης (Εντολές: Edit mesh→Bridge, Control+ right click →edge ring utilities→edge ring split). Ολοκληρώνοντας την πόρπη ασφαλείας, σχεδιάζεται το κούμπωμα του ρολογιού επιλέγοντας έναν κύβο με 20 υποδιαιρέσεις βάθους, εφαρμόζοντάς τον πάνω απ' το κομμάτι της πόρπης και διαμορφώνοντας το σύμφωνα με την εικόνα (Εντολές: Move, Scale, Rotate, Create deformers→nonlinear deformer→bend, Edge loop tool, Paint selection tool, Extrude, Merge) (Εικόνα 74-Σχεδιασμός της πόρπης ασφαλείας).

14. Απόδοση σκιάσεων και υφών

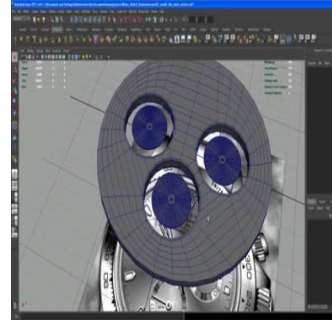
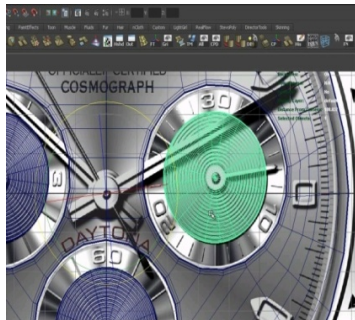
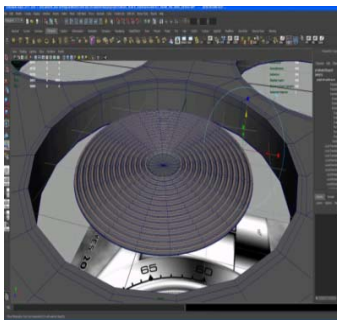
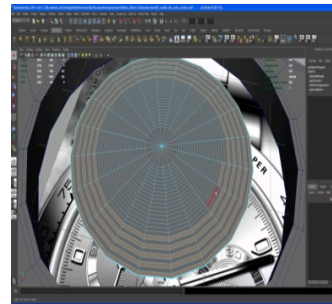
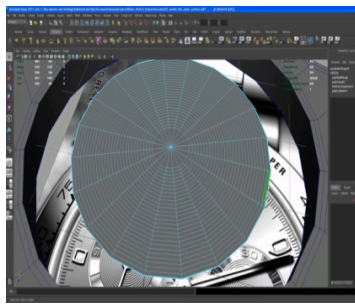
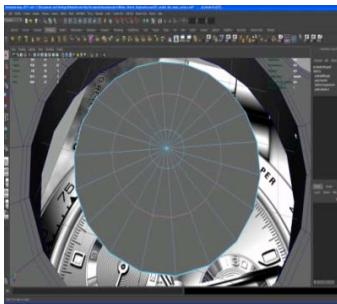
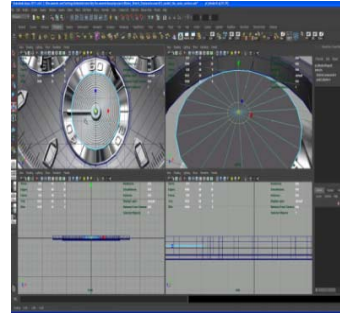
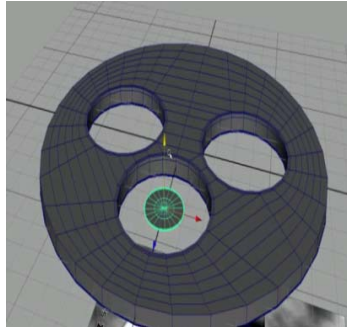
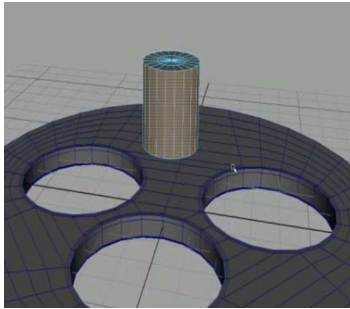
Η απόδοση σκιάσεων και υφών δεν ήταν εφικτό να πραγματοποιηθεί, λόγω του ότι οι απαιτήσεις του λογισμικού σε μνήμη και γραφικά ήταν μεγαλύτερες απ' αυτές που διέθετε ο υπολογιστής στον οποίο υλοποιήθηκε η εφαρμογή (Εικόνα 75-Απόδοση σκιάσεων και υφών).



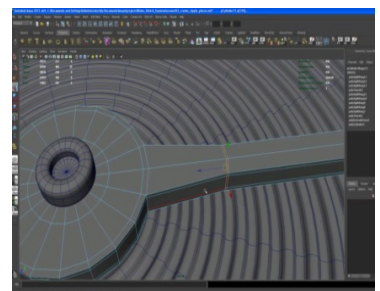
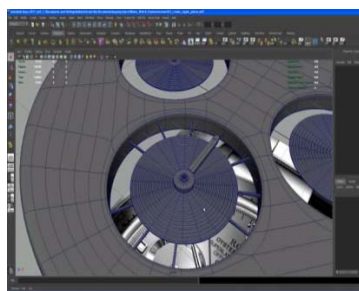
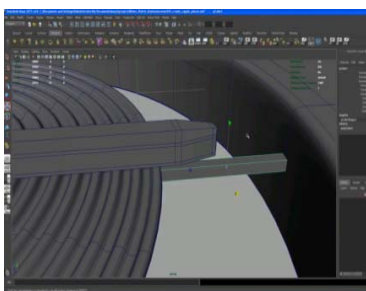
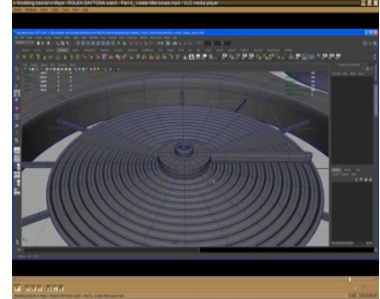
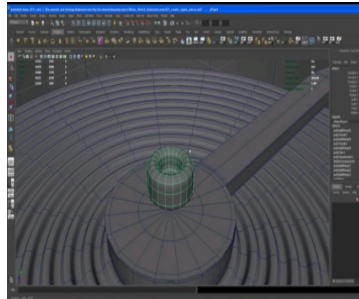
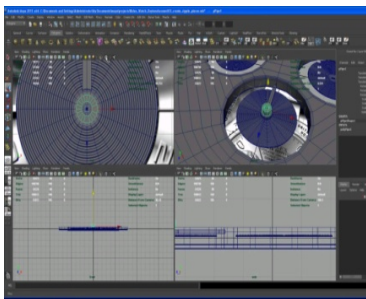
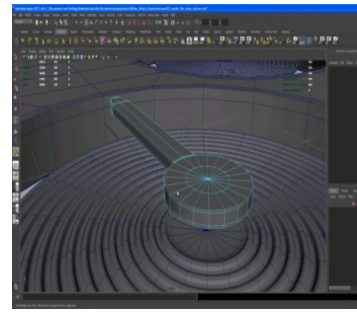
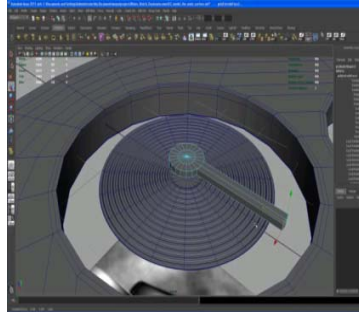
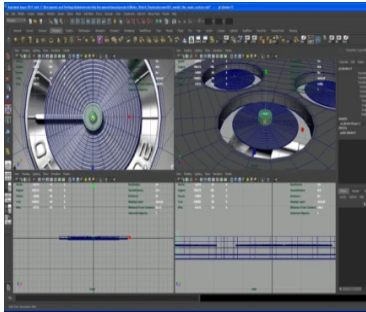
Εικόνα 60-Σχεδιασμός των υπό-καντράν



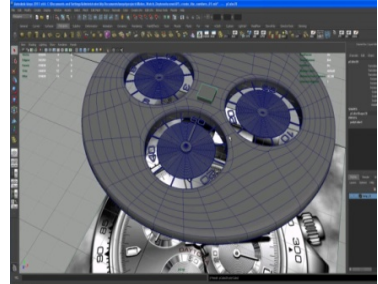
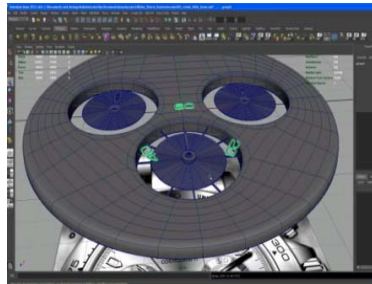
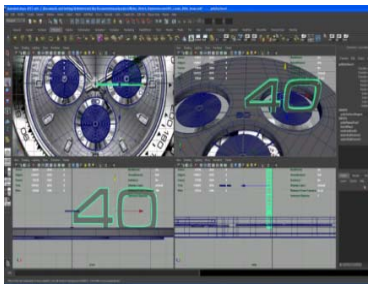
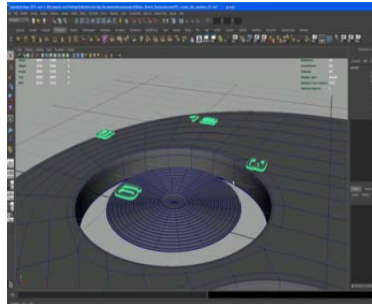
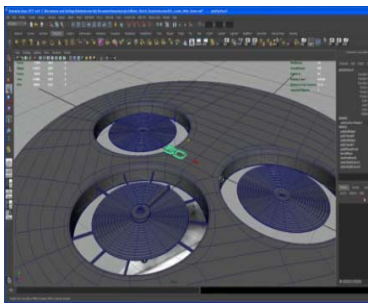
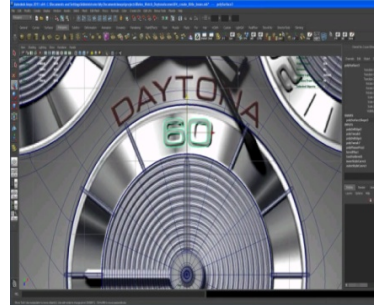
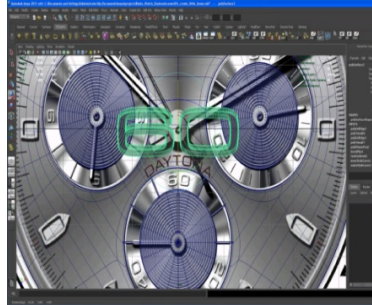
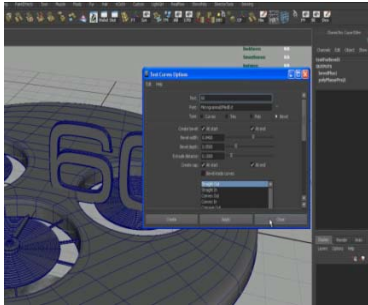
Εικόνα 61-Δημιουργία της επιφάνειας των υπο-
καντράν



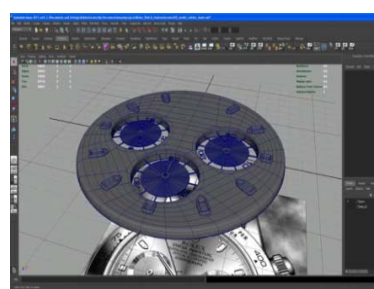
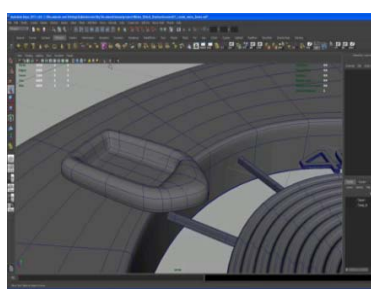
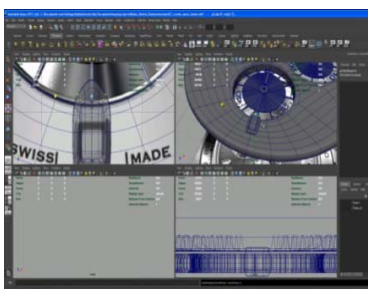
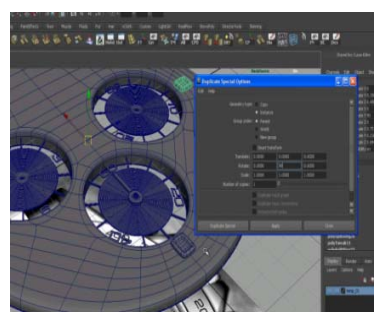
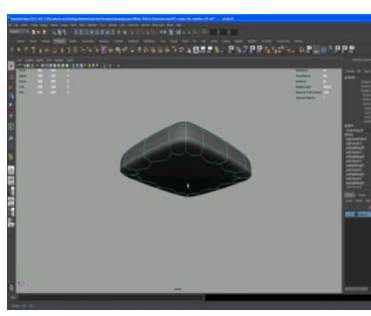
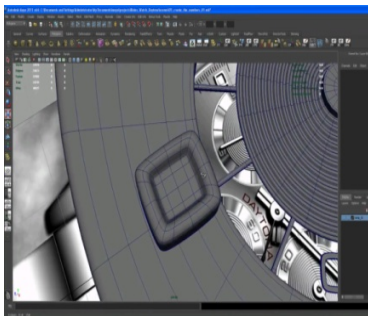
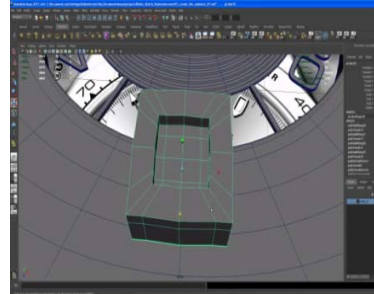
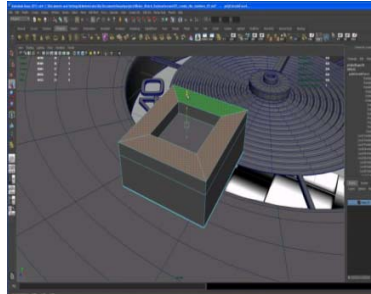
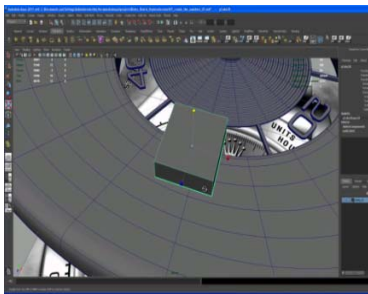
Εικόνα 62- Δημιουργία της επιφάνειας των υπό-
καντράν



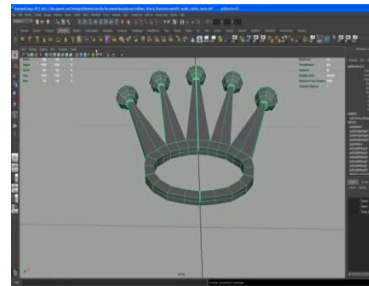
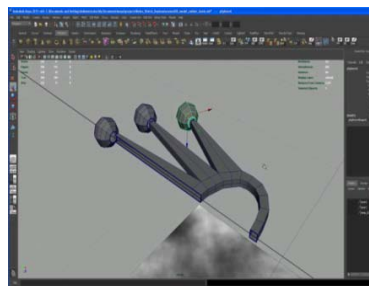
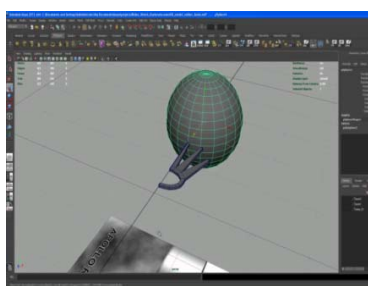
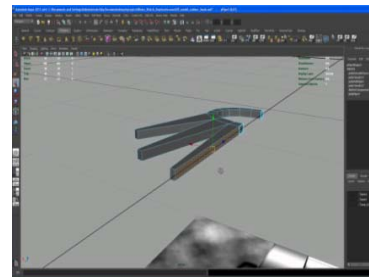
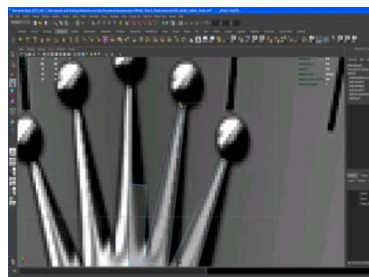
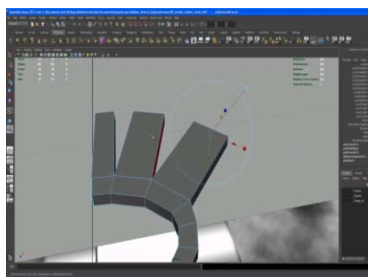
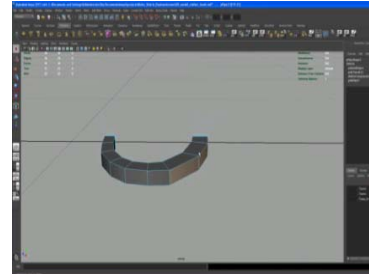
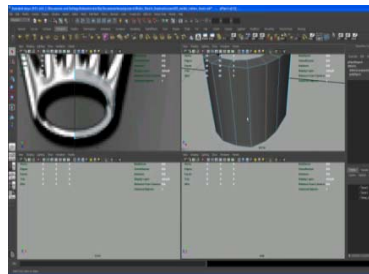
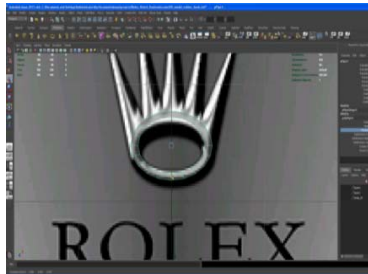
Εικόνα 63-Σχεδιασμός των δεικτών των υπό-καντράν



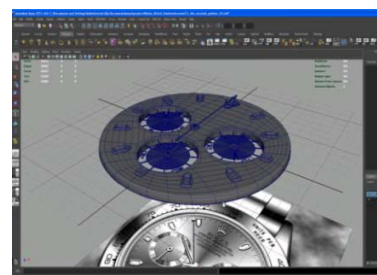
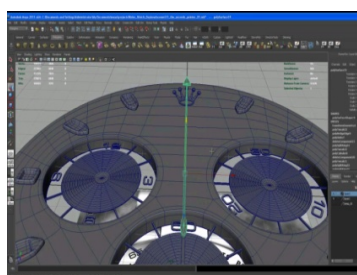
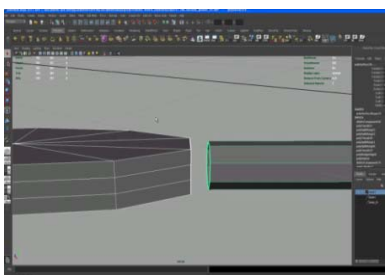
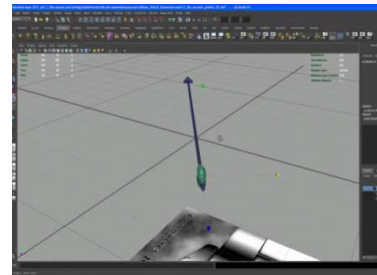
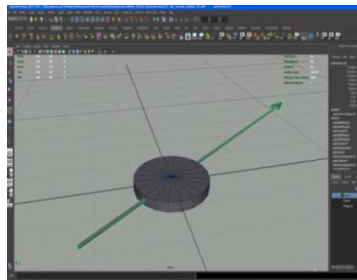
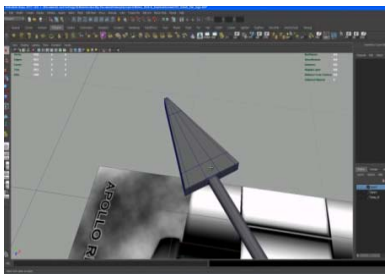
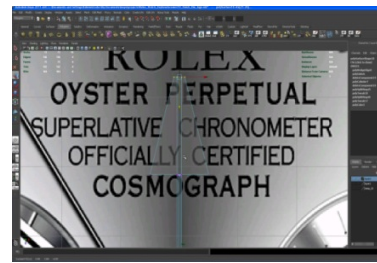
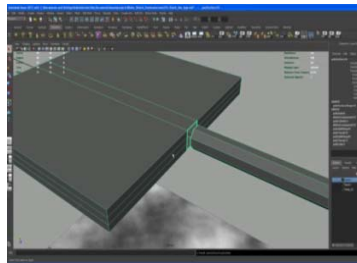
Εικόνα 64-Σχεδιασμός των αριθμών



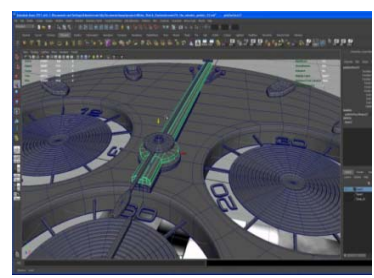
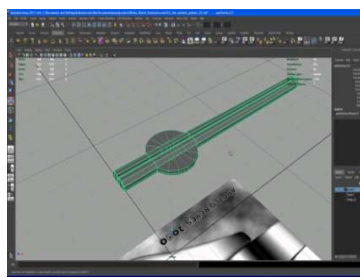
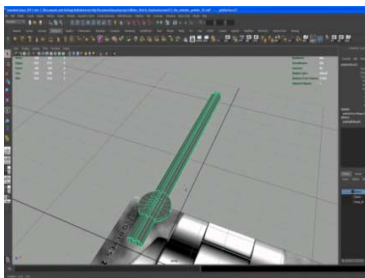
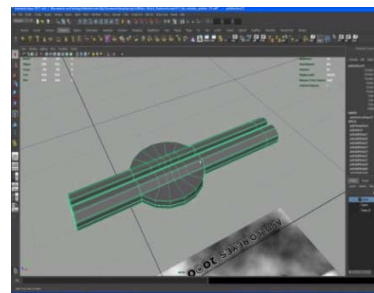
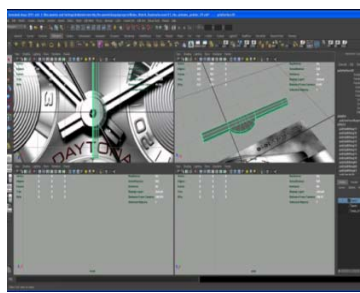
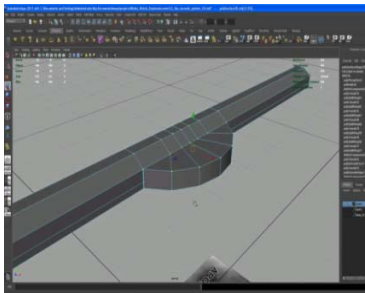
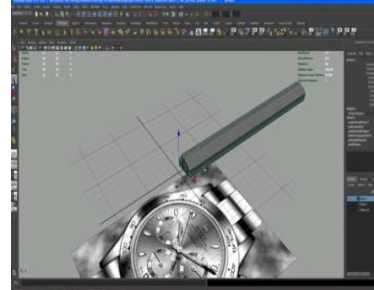
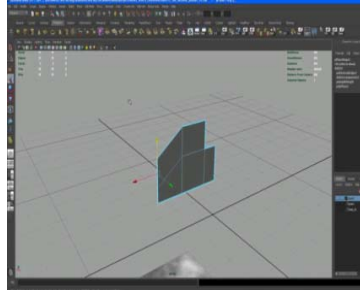
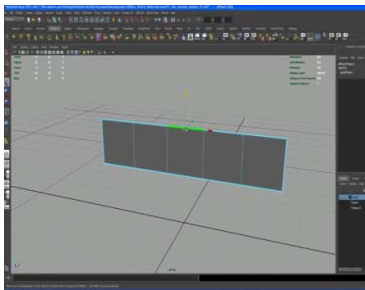
Εικόνα 65-Σχεδιασμός των κουτιών των αριθμών



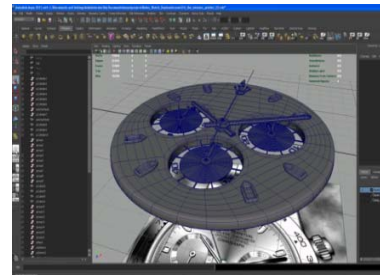
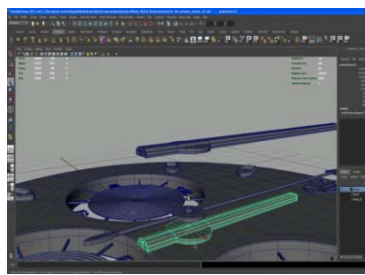
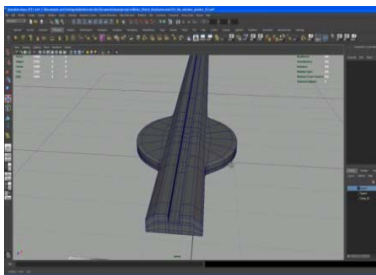
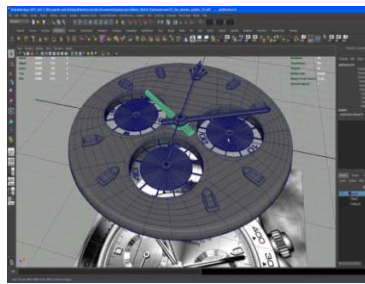
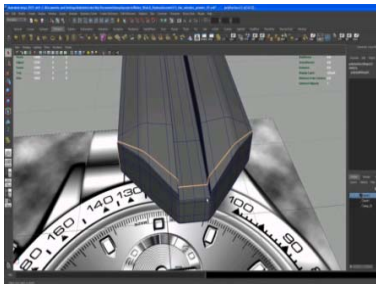
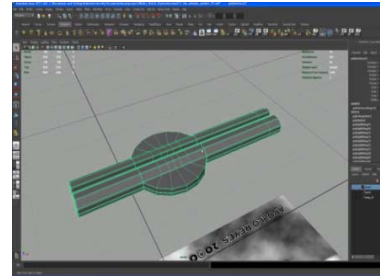
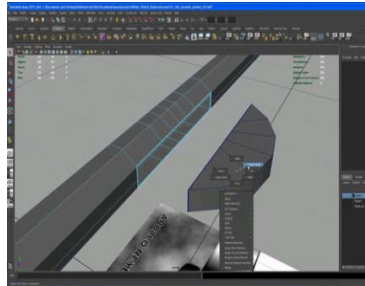
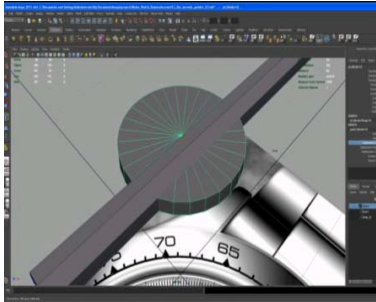
Εικόνα 66-Σχεδιασμός του λογότυπου



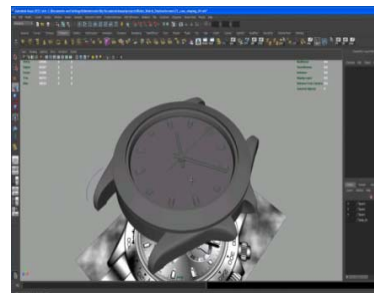
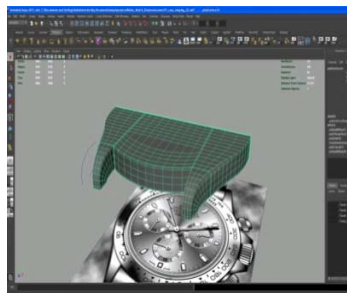
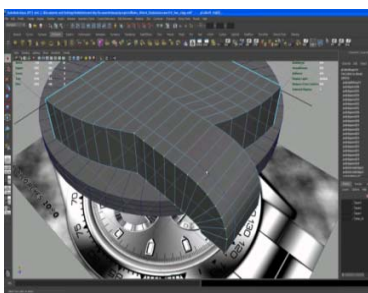
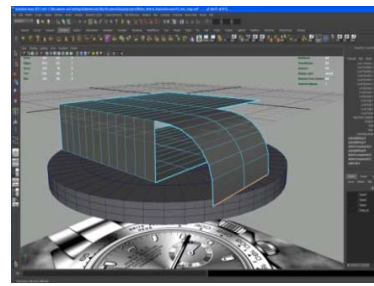
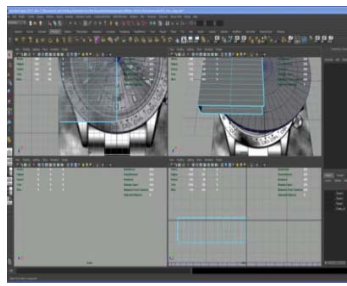
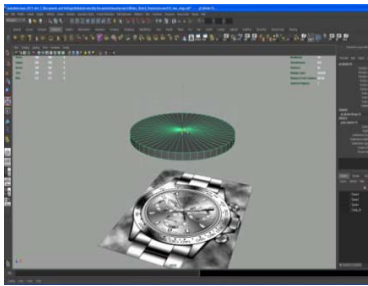
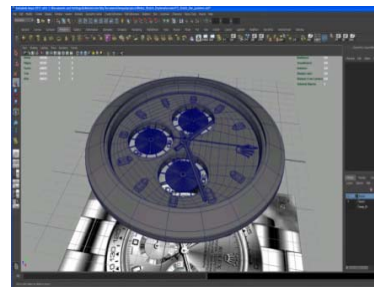
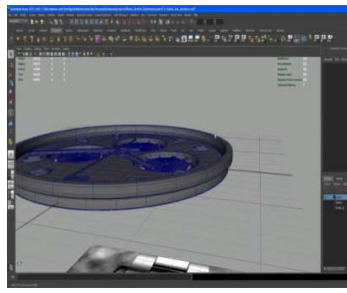
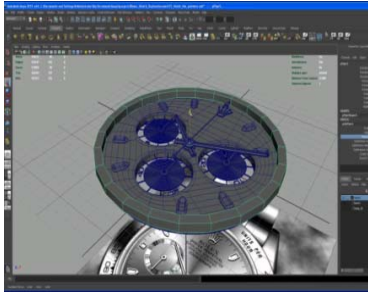
Εικόνα 67-Σχεδιασμός του δευτερολεπτοδείκτη



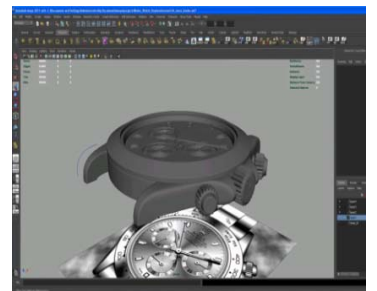
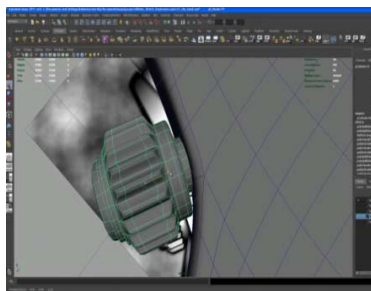
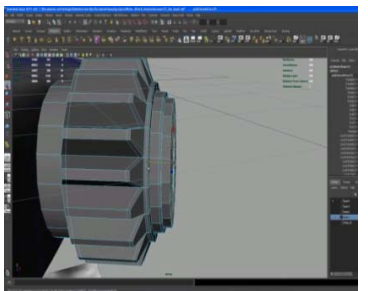
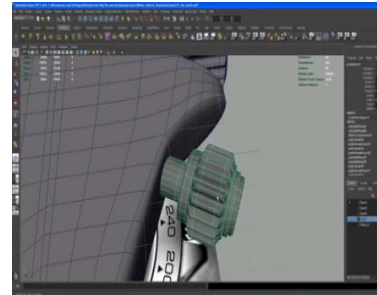
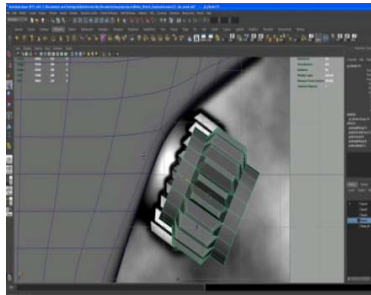
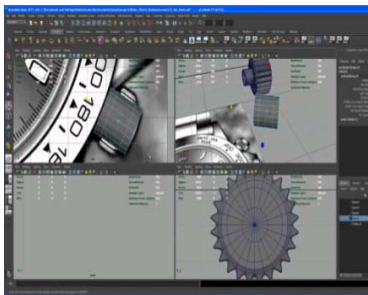
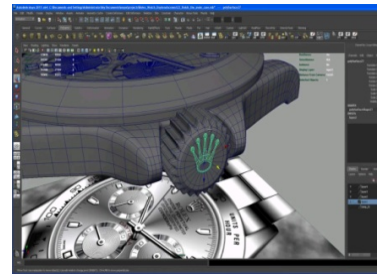
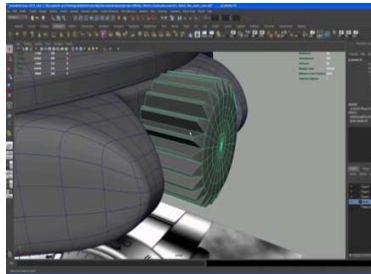
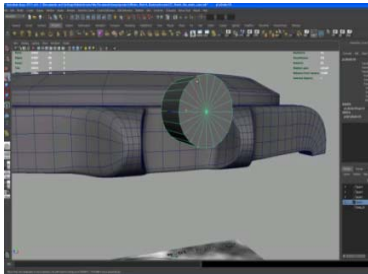
Εικόνα 68-Σχεδιασμός του λεπτοδείκτη



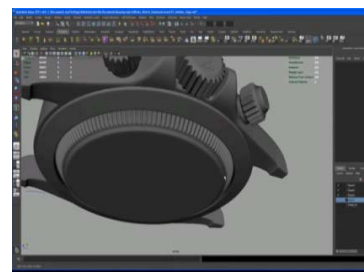
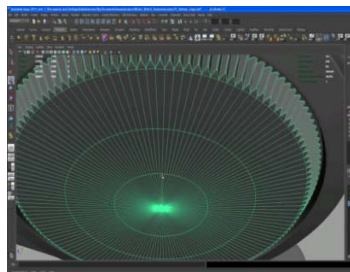
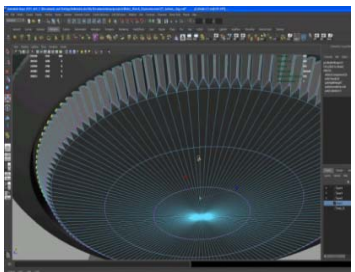
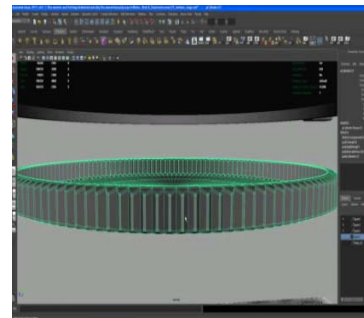
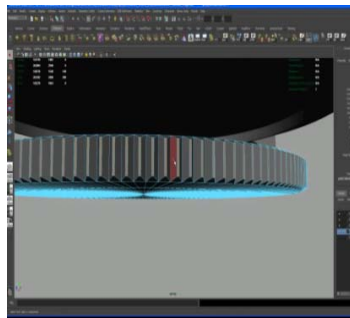
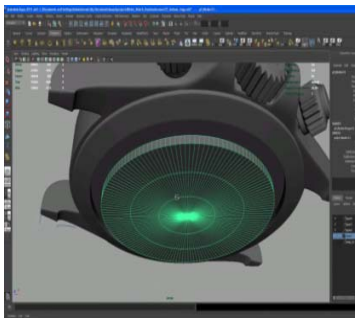
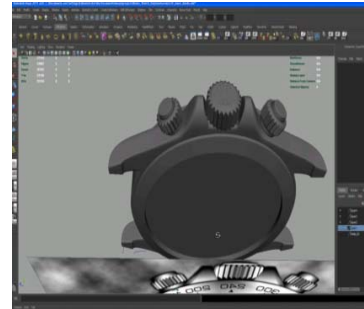
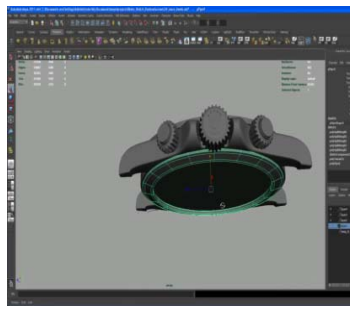
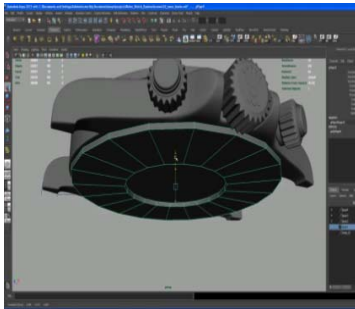
Εικόνα 69-Σχεδιασμός του ωροδείκτη



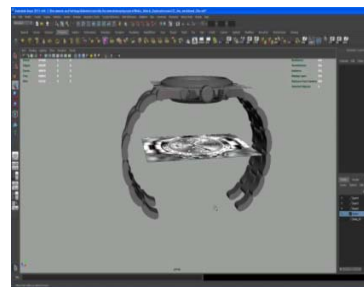
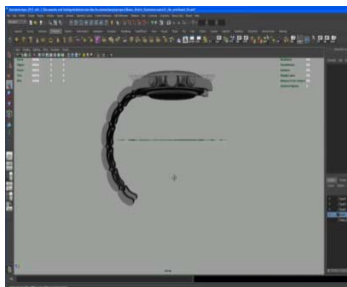
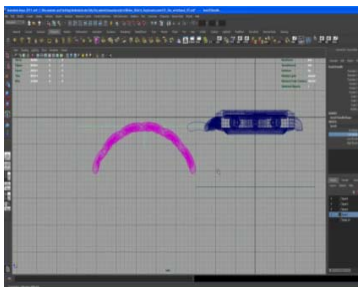
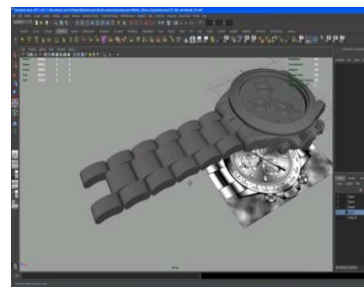
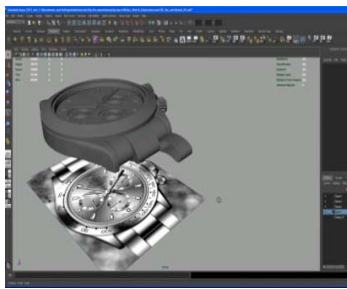
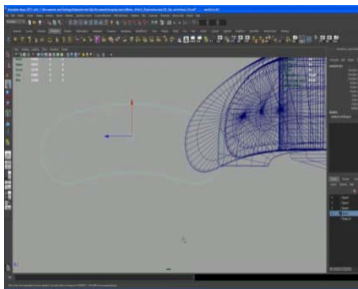
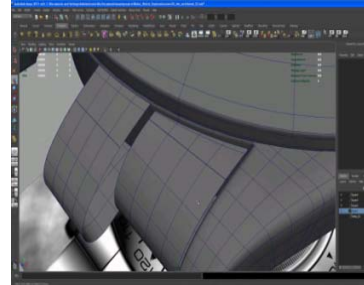
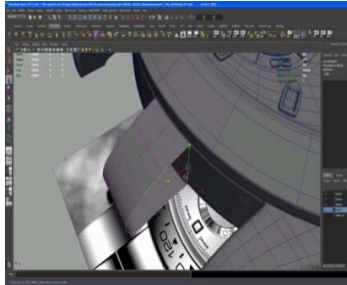
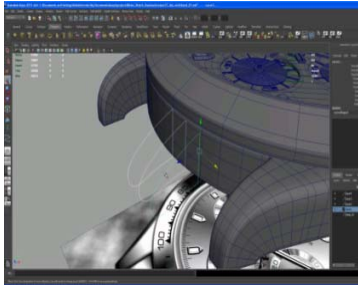
Εικόνα 70-Σχεδιασμός των εξωτερικών δακτυλιδιών και της κάσας



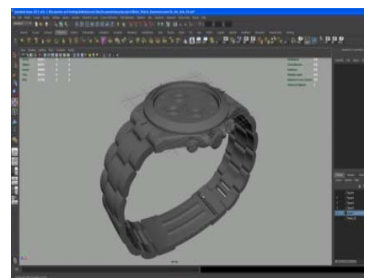
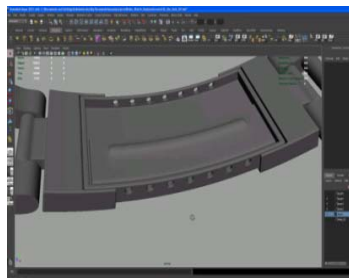
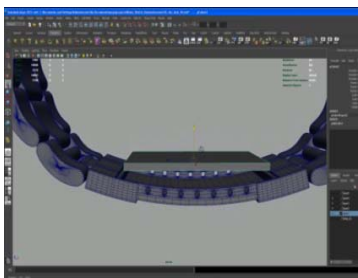
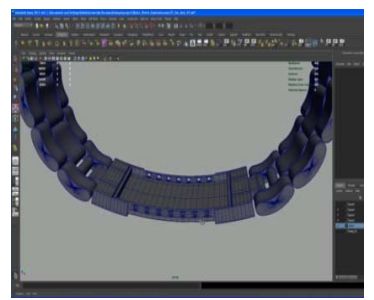
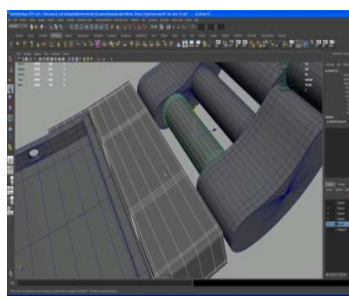
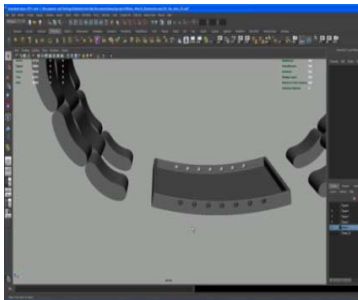
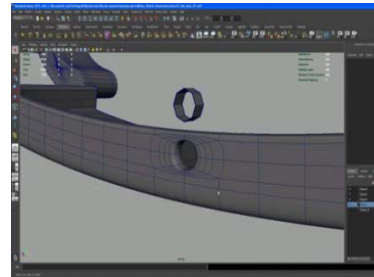
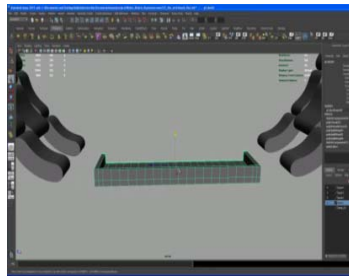
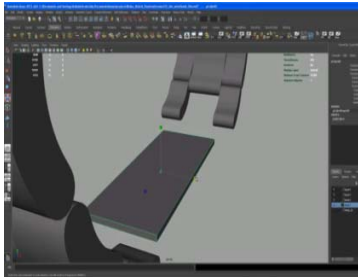
Εικόνα 71-Σχεδιασμός των κουμπιών



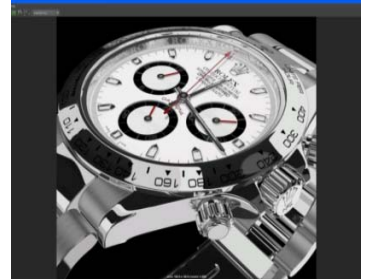
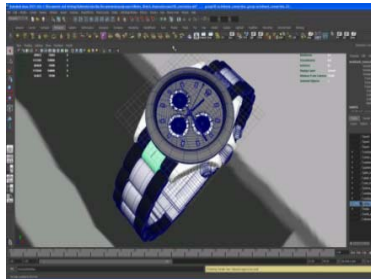
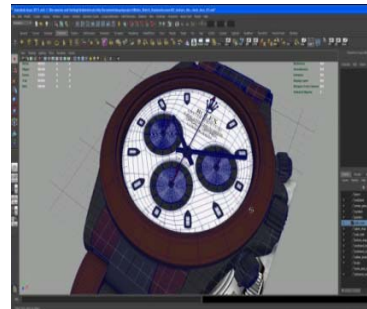
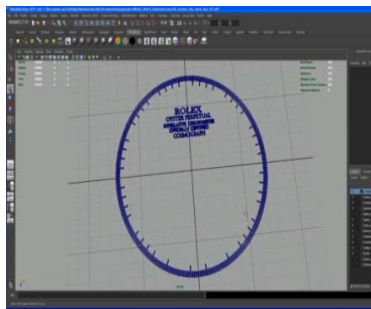
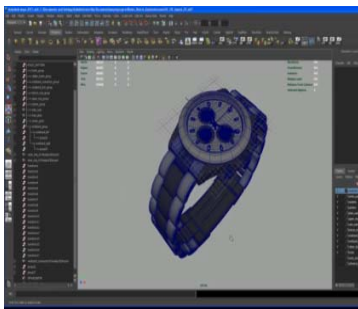
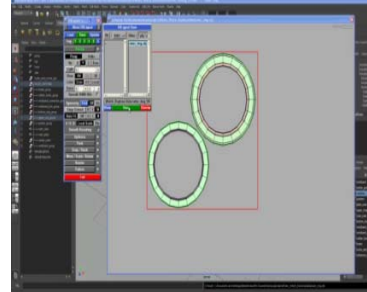
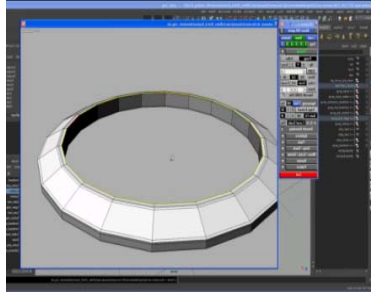
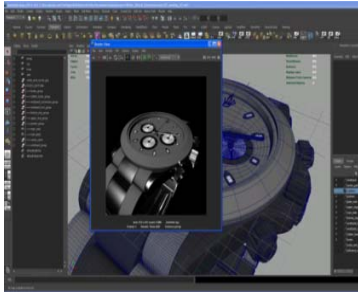
Εικόνα 72-Σχεδιασμός των δακτυλίων της βάσης



Εικόνα 73-Σχεδιασμός του λουριού



Εικόνα 74-Σχεδιασμός της πόρτης ασφαλείας



Εικόνα 75-Απόδοση σκιάσεων και υφών

5 ΣΥΜΠΕΡΑΣΜΑΤΑ-ΠΕΡΙΟΡΙΣΜΟΙ-ΠΡΟΒΛΗΜΑΤΑ

Κατά την διάρκεια της πραγματοποίησης της πτυχιακής εργασίας, παρουσιάστηκαν κάποιοι περιορισμοί /προβλήματα ως προς τα λογισμικά που χρησιμοποιήθηκαν και γενικότερα τις εφαρμογές στις οποίες αφορούσαν.

Εφαρμογή 4.2.3:

ΤΙΤΛΟΣ: Σχεδιασμός της τρισδιάστατης γεωμετρίας ρολογιού

ΛΟΓΙΣΜΙΚΟ: **Autodesk Maya**

Συμπεράσματα:

Για τον σχεδιασμό του ρολογιού, έγινε χρήση του λογισμικού Autodesk Maya έκδοσης 2010, το οποίο χρησιμοποιήθηκε μέσω torrent από το διαδίκτυο. Ο λόγος για τον οποίο επιλέχτηκε το εν λόγω λογισμικό, είναι το γεγονός του ότι αποτελεί ένα από τα κορυφαία λογισμικά δημιουργίας τρισδιάστατων γραφικών και αλληλεπίδρασης τους με το περιβάλλον, λόγος για τον οποίο χρησιμοποιείται κατά κόρον από τις εταιρείες κινηματογραφικών ταινιών, για την προβολή των τρισδιάστατων εφέ και των γραφικών τους. Είναι ένα λογισμικό με αρκετά σύνθετο και πλούσιο μενού, γεγονός το οποίο προϋποθέτει αρκετό χρόνο για την εκμάθηση του τρόπου χειρισμού του. Ωστόσο, για τον σχεδιασμό ενός ρολογιού τριών διαστάσεων, δεν αρκεί μόνο η καλή γνώση ενός λογισμικού, αν είναι επιθυμητό ένα ρεαλιστικό αποτέλεσμα βάσει προδιαγραφών. Για τον λόγο αυτό, αναζητήθηκαν απ' το διαδίκτυο και συγκεντρώθηκαν κάποια βίντεο, το περιεχόμενο των οποίων ήταν ο σχεδιασμός ενός ρολογιού Rolex Daytona στο Autodesk Maya. Τα βίντεο αυτά, συνέβαλαν αρκετά στην πραγματοποίηση του σχεδιασμού ως οδηγοί εντός του προγράμματος και σε μία γενικότερη κατανόηση του για μελλοντικές εφαρμογές.

Περιορισμοί/προβλήματα

Παρουσιάστηκαν, βέβαια, κάποιες δυσκολίες στην ερμηνεία των βίντεο, όσον αφορά στην αναζήτηση και εύρεση κάποιων εντολών εντός του προγράμματος, σε μία μικρή ασυμβατότητα μεταξύ των διαφορετικών εκδόσεων (2010 έναντι 2012), η οποία δεν επέτρεπε την πραγματοποίηση κάποιων αποτελεσμάτων παρότι γινόταν χρήση των ίδιων εντολών. Πέραν της ασυμβατότητας λόγω διαφορετικών εκδόσεων, περιορισμό αποτέλεσε το ότι τα προγράμματα που χρησιμοποιήθηκαν δεν ήταν αυθεντικά, γεγονός που δημιουργούσε περιστασιακά σφάλματα η και τερματισμό της λειτουργίας προγράμματος. Σημαντικό περιορισμό αποτέλεσε το γεγονός ότι δεν ήταν εφικτή η ολοκλήρωση της γεωμετρίας του ρολογιού, λόγω του ότι ο υπολογιστής στον οποίο σχεδιάστηκε το ρολόι, δεν επαρκούσε για τις απαιτήσεις του προγράμματος λόγω μη επαρκούς μνήμης και κάρτας γραφικών, ώστε να προκύψει το επιθυμητό αποτέλεσμα. Συγκεκριμένα, δεν αποδόθηκαν κάποιες υφές (textures) στο ρολόι όπως τα λογότυπα: «made in Swiss», οι αριθμοί και τα διακριτικά της περιστρεφόμενης στεφάνης, η απόδοση σκιάσεων και φωτισμού κτλ.

Εφαρμογή 4.2.3

ΤΙΤΛΟΣ: **Απόδοση γραφικών στο ρολόι**

ΛΟΓΙΣΜΙΚΟ: **Adobe Flash Player**

Συμπεράσματα:

Σκοπός της δημιουργίας του ρολογιού τριών διαστάσεων, ήταν να χρησιμοποιηθεί η γεωμετρία του στο Adobe Flash, ώστε μέσω των εργαλείων του να κινούνται οι δείκτες του βάσει της σωστής ώρας και να προβληθεί στην συνέχεια διαδικτυακά. Η ιδέα προήλθε από την ιστοσελίδα της Rolex, στην οποία ένα ρολόι προβάλλεται με μία ελαφριά κλίση, μετά από λίγα δευτερόλεπτα οι δείκτες μετατοπίζονται στην σωστή ώρα και από κει και πέρα το ρολόι συνεχίζει να λειτουργεί κανονικά (προβολή γραφικών),

δίνοντας έτσι την αίσθηση ενός τρισδιάστατου αντικειμένου. Η επιλογή του Adobe Flash Player, έγινε για δύο κυρίως λόγους:

- 1) Στην ιστοσελίδα της Rolex, κάνοντας δεξί κλικ σ' ένα ρολόι κατά την λειτουργία του, υπάρχει η επιλογή "About Adobe Flash player", που οδηγεί στο συμπέρασμα ότι έγινε χρήση του Adobe Flash Player.
- 2) Στη δυνατότητα του Adobe Flash, εξαγωγής αρχείων στο διαδίκτυο και προβολής τους ως εφαρμογών κάθε τύπου.

Περιορισμοί/προβλήματα

Το Adobe Flash Player, αποτελεί εν γένει ένα λογισμικό που δημιουργεί, επεξεργάζεται και αλληλεπιδρά με γραφικά δύο διαστάσεων. Επιπλέον διαθέτει κάποιες δυνατότητες επεξεργασίας 3D γραφικών μέσω εξωτερικών μεσαζόντων προγραμμάτων, ωστόσο δεν αφορά σε πραγματικά τρισδιάστατα γραφικά, αλλά σε δισδιάστατα γραφικά που δίνουν την ψευδαίσθηση του τρισδιάστατου. Το Autodesk Maya, όπως και άλλα λογισμικά τα οποία επεξεργάζονται τρισδιάστατα γραφικά, όπως είναι φυσικό, δεν είναι εφικτό να «συνεργαστεί» μ' ένα πρόγραμμα επεξεργασίας δισδιάστατων γραφικών όπως το Adobe Flash Player, με την λογική αποστολής/εξαγωγής αρχείων από 3D σε 2D. Ως εκ τούτου, η αποστολή/εξαγωγή της τρισδιάστατης γεωμετρίας του ρολογιού από το Autodesk Maya και κυρίως η επεξεργασία του στο Adobe Flash, δεν ήταν εφικτή. Αναλυτικότερα, έγιναν αρκετές προσπάθειες ανταλλαγής αρχείων μεταξύ 3D και 2D λογισμικών με διαφορετικά μέσα, είτε χρησιμοποιώντας αποκλειστικά τα εργαλεία των λογισμικών, είτε κάνοντας χρήση εξωτερικών λογισμικών:

1. Μεταφορά αρχείων από Maya σε flash μέσω των πηγαίων τύπων αρχείων που διαθέτουν: Από Maya: (.fbx), (.lges), (.obj), (.stl), σε Flash: (.swf), (.png), (.fxg), (.dxf), (.bmp), (.tiff), αποτέλεσμα των οποίων ήταν είτε αποτυχία μεταφοράς της γεωμετρίας με την μορφή σφάλματος, είτε με μερική μεταφορά γεωμετρίας (σπασίματα ή ελλείψεις στην γεωμετρία), ωστόσο σε μορφή εικόνας(.bmp).
2. Μεταφορά αρχείων από Maya σε flash κάνοντας χρήση εξωτερικών προγραμμάτων ή plug-ins τα οποία λειτουργούν ως μεσάζοντα η συμπληρωματικά προγράμματα και σκοπός των οποίων είναι, ως επί το

πλείστον, η επεξεργασία 3D γραφικών. Τα πιο διαδεδομένα από αυτά τα προγράμματα είναι: Swift3D, PaperVision3D, Alternativa3D. Στην προκειμένη περίπτωση περιορισμό αποτέλεσε το γεγονός ότι σχεδόν όλα τα προγράμματα αυτά δεν διατίθεντο δωρεάν αλλά ούτε σε μη αυθεντικές εκδόσεις που κάποιος θα μπορούσε να προμηθευτεί απ' το διαδίκτυο. Ήταν εφικτή η εύρεση και χρήση του swift3D το οποίο, όπως αναφέρθηκε και παραπάνω, δεν βοήθησε ιδιαίτερα απ' την στιγμή που επεξεργάζεται δισδιάστατα γραφικά που δίνουν την ψευδαίσθηση των τριών διαστάσεων χωρίς να είναι πραγματικά τρισδιάστατα. Εν κατακλείδι, με τις πηγές, τα μέσα, τις γνώσεις καθώς και με την έρευνα και τις δυνατότητες που διατέθηκαν, δεν ήταν εφικτός ο πρωταρχικός στόχος της πτυχιακής, δηλαδή η διαδικτυακή προβολή ρολογιού σε τρεις διαστάσεις. Έναντι, η εφαρμογή υλοποιήθηκε σε δύο διαστάσεις και αναπτύχθηκε ξεχωριστή εφαρμογή σε γλώσσα Vmml η οποία αναλύεται παραπάνω.

Εφαρμογές 4.1.1- 4.1.2:

ΤΙΤΛΟΣ: **Διαδικτυακή προβολή τρισδιάστατων γραφικών με χρήση VRML**

ΛΟΓΙΣΜΙΚΟ: **Cortona , Cosmo player.**

Συμπεράσματα:

Η παρούσα εφαρμογή, έγινε σε ως εναλλακτική προσπάθεια για την πραγματοποίηση της προβολής τρισδιάστατων γραφικών στο διαδίκτυο. Η προσπάθεια αυτή προϋπέθετε την εκμάθηση και κατανόηση των βασικής δομής της γλώσσας Vmml και την εφαρμογή αρκετών απλών παραδειγμάτων, ώστε ν' αποκτηθεί μία στοιχειώδης εμπειρία και να τεθούν τα θεμέλια για την πραγματοποίηση πιο σύνθετων παραδειγμάτων. Η γλώσσα VRML αποτελεί μία πρωτογενή μορφή της HTML, μιας καθολικά διαδεδομένης διαδικτυακής γλώσσας κι ως εκ τούτου η μελέτη της βασικής της δομής αποτέλεσε χρήσιμο εργαλείο και εφόδιο.

Περιορισμοί/προβλήματα

Η εισαγωγή σε οποιαδήποτε καινούργια γλώσσα, ως επί το πλείστον, δεν γίνεται εύκολα και εν προκειμένω ο προγραμματισμός σε VRML και ειδικότερα η πραγματοποίηση της συγκεκριμένης εφαρμογής, χρειάστηκε αρκετή έρευνα και χρόνο. Επιπλέον έγινε μια απόπειρα να εξαχθεί η τρισδιάστατη γεωμετρία του ρολογιού σε VRML, με σκοπό να είναι η εφικτή η κίνηση των δεικτών του και κατά συνέπεια η λειτουργία του ρολογιού και να προβάλλεται διαδικτυακά. Ωστόσο, αν και η εξαγωγή του ρολογιού ήταν εφικτή, η προσπάθεια τροποποίησης του κώδικα ώστε να κινούνται οι δείκτες απέτυχε, ενδεχομένως λόγω μη επαρκούς γνώσης και κατανόησης της γλώσσας VRML.

Εφαρμογή 4.2.1- 4.2.2 - 4.2.3:

ΤΙΤΛΟΣ: **Δημιουργία δισδιάστατου ρολογιού και κίνηση των δεικτών του**

ΛΟΓΙΣΜΙΚΟ: **Adobe Illustrator, Adobe Flash**

Συμπεράσματα:

Η ανάπτυξη της προκείμενης εφαρμογής, είχε 2 αντικειμενικούς στόχους: Α) την μελέτη και γενικότερη κατανόηση του προγράμματος Adobe Flash, Β) την δημιουργία ενός ρολογιού δύο διαστάσεων και επεξεργασία του αποκλειστικά στο flash player, ώστε να γίνει κατανοητή η μέθοδος απόδοσης δεικτών στο ρολόι καθώς και η χρήση κατάλληλου κώδικα ώστε οι δείκτες να λειτουργούν βάσει της σωστής ώρας, προκειμένου να εφαρμοστεί η διαδικασία αυτή στο ρολόι τριών διαστάσεων που σχεδιάστηκε στο Autodesk Maya. Το ρολόι στην παρούσα εφαρμογή σχεδιάστηκε εξ' ολοκλήρου στο Adobe Illustrator, ένα πρόγραμμα που διαθέτει πλούσια εργαλεία και δυνατότητες σχεδίασης και επεξεργασίας γραφικών σε δύο διαστάσεις. Προς το παρόν, με εξωτερικά υποστηρικτικά προγράμματα η μηχανή του Adobe Flash μπορεί να επεξεργάζεται γραφικά που δίνουν την εντύπωση και την ψευδαίσθηση της τρίτης διάστασης. Μελλοντικά, με την συνεχόμενη εξέλιξη του, το εν λόγω πρόγραμμα ενδεχομένως να

εισαγάγει στην μηχανή του τη δυνατότητα αλληλεπίδρασης με γραφικά και στις τρεις διαστάσεις.

Περιορισμοί/προβλήματα

Ο γενικότερος περιορισμός και σε αυτήν την εφαρμογή είναι το ότι και τα δύο προγράμματα που χρησιμοποιήθηκαν σε αυτήν έχουν τη δυνατότητα αλληλεπίδρασης δισδιάστατων γραφικών και όχι πραγματικών τρισδιάστατων. Προς το παρόν με εξωτερικά υποστηρικτικά προγράμματα η μηχανή του Adobe Flash μπορεί να επεξεργάζεται γραφικά που δίνουν την εντύπωση και την ψευδαίσθηση της τρίτης διάστασης. Ένας ακόμη περιορισμός παρουσιάστηκε εντός του προγράμματος Adobe Flash, μεταξύ της λειτουργίας του κώδικα που χρησιμοποιήθηκε για την κίνηση των δεικτών και την επιλογή της γλώσσας προγραμματισμού του flash για την δημιουργία αλληλεπιδράσεων και ειδικών εφέ (ActionScript). Αναλυτικότερα, στο Adobe Flash υπάρχει επιλογή μεταξύ τριών ActionScript τα οποία περιέχουν τα στοιχεία της γλώσσας προγραμματισμού, ηλεκτρονικές βιβλιοθήκες του προγράμματος, εργαλεία, υπηρεσίες κ.α. και οι διαφορές μεταξύ των ActionScript, συνίστανται στις αναβαθμίσεις και τις προσθήκες των παραπάνω περιεχομένων. Αρχικά χρησιμοποιήθηκε το ActionScript 3.0, ως η πιο αναβαθμισμένη έκδοση. Ωστόσο, κατά την εκτέλεση του κώδικα παρουσιάζονταν συνεχώς σφάλματα τα οποία δεν είχαν σχέση με την ορθότητα της δομής του κώδικα. Μετά από κάποιες προσπάθειες ερμηνείας και διόρθωσης του σφάλματος, αποδείχτηκε ότι ο συγκεκριμένος κώδικας δεν ήταν συμβατός με το συγκεκριμένο ActionScript, ενώ δοκιμάστηκε στην αμέσως προηγούμενη έκδοση ActionScript 2.0 στην οποία λειτουργούσε κανονικά.

6 ΕΠΙΛΟΓΟΣ

Σκοπός αυτής της εργασίας, ήταν η απεικόνιση τρισδιάστατων γραφικών με δυνατότητα προβολής τους στο διαδίκτυο. Η εφαρμογή η οποία αναπτύχθηκε, ήταν ο σχεδιασμός ενός τρισδιάστατου ρολογιού, η επεξεργασία του ώστε να κινούνται οι δείκτες του και η διαδικτυακή του προβολή. Τα λογισμικά τα οποία επιλέχθηκαν για τον σκοπό αυτό, ήταν το Autodesk Maya και το Adobe Flash Player. Ωστόσο, λόγω του ότι τα δύο λογισμικά διέθεταν διαφορετική τεχνολογία επεξεργασίας γραφικών, η υλοποίηση της εφαρμογής αυτής δεν ήταν εφικτή. Το συμπέρασμα αυτό, προέκυψε μετά από εκτενή έρευνα και χρήση διάφορων μεθοδολογιών και υποστηρικτικών προγραμμάτων. Έτσι, η εφαρμογή πραγματοποιήθηκε στα πλαίσια της απεικόνισης της λειτουργίας του ρολογιού σε δύο διαστάσεις και της προβολής του στο διαδίκτυο. Η τεχνολογία των παραπάνω λογισμικών αναπτύσσεται συνεχώς και ενδεχομένως στο μέλλον να είναι εφικτή η συνεργασία τους ως προς την επεξεργασία γραφικών και κατ' επέκταση να γίνει εφικτή η υλοποίηση της προκείμενης εφαρμογής σε τρισδιάστατη απεικόνιση. Επίσης, η υλοποίηση της εφαρμογής αυτής, θα μπορούσε να αποτελέσει αντικειμενικό στόχο μίας νέας πτυχιακής εργασίας που θα αναλάμβανε ένας άλλος σπουδαστής. Για την ολοκλήρωση της παρούσας εργασίας και παράλληλα για την επίτευξη του σκοπού της, αναπτύχθηκε μια ξεχωριστή εφαρμογή βάσει τις οποίας προβάλλονται κινούμενα γραφικά τριών διαστάσεων στο διαδίκτυο και για την υλοποίηση της οποίας έγινε χρήση της τεχνολογίας VrmI (Virtual Reality Modeling Language) και του λογισμικού Cortona.

7 ΒΙΒΛΙΟΓΡΑΦΙΑ

VRML

1. <http://sites.google.com/site/multimediamdreamwiki/4--programming-and-vrml>
2. <http://www.agocg.ac.uk/train/vrml2rep/appendix/appxanod.htm>
3. <http://www.lighthouse3d.com/vrml/tutorial/index.shtml?intro>
4. <http://vrmlworks.crispen.org/tutorials/events.html>
5. <http://www.it.uom.gr/project/vrml/vrml01/index.html#node>
6. <http://www.uco.es/grupos/eatco/automatica/ihtm/cursovrml/frmorint.htm>
7. <http://www.agocg.ac.uk/train/vrml2rep/part1/guide3.htm>
8. <http://cic.nist.gov/vrml/vbdetect.html>
9. http://users.sch.gr/nikmavr/wp-content/files/tech_3d_and_examples_for_teaching_with_VRML.pdf
10. <http://vrmlworks.crispen.org/faq/faq4.html>

ADOBE FLASH

1. <http://www.stunningmesh.com/2010/08/smooth-analog-clock-in-flash/>
2. http://www.entheosweb.com/Flash/analog_clock.asp
3. <http://flashexplained.com/design/making-an-antique-analog-clock/>
4. <http://www.freeitsolutions.com/flash-tutorials/viewTutorial.aspx?id=3422>
5. <http://www.slicktutorials.com/3d-lcd-alarm-clock>
6. <http://www.webdesign.org/vector-graphics/adobe-illustrator/how-to-build-a-vector-clock-graphic-in-illustrator.18055.html>
7. <http://designinstruct.com/iconlogo-design/design-a-3d-digital-clock-icon-with-illustrator/>
8. <http://www.codeproject.com/KB/dotnet/vgclock.aspx>

AYTODESK MAYA

1. www.codeproject.com/KB/dotnet/vgclock.aspx<http://www.creativecrash.com/maya/downloads/scene-files/c/>
2. www.codeproject.com/KB/dotnet/vgclock.aspx[http://www.creativecrash.com/maya/downloads/scene-](http://www.creativecrash.com/maya/downloads/scene-files/c/)
[files/c/](http://www.creativecrash.com/maya/downloads/scene-files/c/)<http://www.3dbuzz.com/vbforum/showthread.php?128609-Tutorial-Convert-3ds-max-animations-to-Flash-format-%28.swf%29>
3. <http://www.headus.com/phpbb/viewtopic.php?t=394>
4. <http://www.youtube.com/watch?v=loXpbUEPG8k&feature=related>
5. <http://usa.autodesk.com/maya/>
6. <http://www.extensis.com/en/WebINK/fonts/font.jsp?family=Microgramma%20D>
7. <http://www.broble.com/download-free-font/baskerville-old-face/>