



# Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης

Σχολή Τεχνολογικών Εφαρμογών  
Τμήμα Εφαρμοσμένης Πληροφορικής & Πολυμέσων



Πτυχιακή Εργασία

Τίτλος: **Ανάπτυξη Γραφικού Παιχνιδιού σε iPhone**

Νυδριώτης Απόστολος (ΑΜ: 1336)

Επιβλέπων Καθηγητής: κ. Τριανταφυλλίδης Γεώργιος

Επιτροπή Αξιολόγησης:

Ημερομηνία Παρουσίασης:

## Ευχαριστίες

Με την ολοκλήρωση της πτυχιακής μου εργασίας θα ήθελα να ευχαριστήσω όλους όσους με υποστήριξαν. Επιπλέον οφείλω ένα μεγάλο ευχαριστώ στον επόπτη-καθηγητή κύριο Τριανταφυλλίδη Γεώργιο για τις πολύτιμες συμβουλές και την καθοδήγησή του κατά την εκπόνηση της συγκεκριμένης εργασίας.

## Abstract

The aim of this report is to record the process of creating an application applied on smartphones, specifically the iPhone in order to understand this remarkable increase in demand in the last years. For purposes of presentation, an app is created (more specifically a game) that runs on the iPhone.

Initially, in the first chapter, there is a reference to smartphones and the need of mobile users for even greater computational power, more features and applications and the essential usability that is requested nowadays. Most of the most common operating systems that run on smartphones are reported, as well as statistics on their sales.

The second chapter describes one of the most popular of them and certainly the most innovative, the iPhone from Apple Inc. Some of its features and its development from the first model -iPhone 2g- are analyzed up until the last model -at the time of writing this text- the iPhone 4g. There is also a thorough explanation of the program used (Flash CS5), as well as the procedure that it takes for someone to become a member of Apple programmers project in order to be able to sell their applications through the app store.

The next chapter is an attempt for users not very familiar with Flash, to understand this very platform and the programming language it uses, by presenting a very simple "hello world" type program.

Subsequently the user familiarizes with the program through a more complex application that makes use of ready-made libraries of Flash for the iPhone and specifically for a new feature, the accelerometer. There is also a detailed analysis done, on both the source code and the steps of converting the application from a file that can only be executed on a computer into a file that will run on the iPhone. Thereafter the file gets transferred to the mobile device and the application gets tested directly by the developer.

Finally the most important parts of the source code of the main application/game are analyzed. The steps taken, the problems encountered and the parts that could take some improvement are described.

## Σύνοψη

Στόχος της παρακάτω εργασίας είναι να καταγραφούν οι διαδικασίες δημιουργίας μιας εφαρμογής των smartphones και συγκεκριμένα του μοντέλου iPhone έτσι ώστε να γίνει κατανοητή αυτή η αξιοσημείωτη αύξηση της ζήτησης τους τα τελευταία χρόνια. Για τις ανάγκες της παρουσίασης, δημιουργείται μία εφαρμογή/παιχνίδι που τρέχει στο iPhone.

Αρχικά, στο πρώτο κεφάλαιο γίνεται μία αναφορά στα smartphones και στην ανάγκη των χρηστών τους για όλο και μεγαλύτερη υπολογιστική ισχύ, περισσότερες δυνατότητες και εφαρμογές και την απαιτούμενη ευχρηστία που οι ανάγκες της εποχής επιβάλλουν. Παρατίθενται τα περισσότερα απ' τα πιο γνωστά λειτουργικά συστήματα τα οποία τρέχουν σε Smartphones, όπως επίσης και στατιστικά στοιχεία με τις πωλήσεις τους.

Στο δεύτερο κεφάλαιο παρουσιάζεται ένα από τα πιο δημοφιλή εξ' αυτών και σίγουρα το πιο καινοτόμο, το iPhone της Apple. Αναλύονται κάποια απ τα χαρακτηριστικά του καθώς και η εξέλιξη του απ το πρώτο μοντέλο, το iPhone 2g, μέχρι και το τελευταίο ως τη στιγμή που γράφεται το παρόν κείμενο, το iPhone 4g.

Στο επόμενο στάδιο αναλύεται λεπτομερώς το πρόγραμμα που χρησιμοποιείται (Flash CS5), καθώς και η διαδικασία που χρειάζεται για να γίνει κάποιος μέλος της Apple έτσι ώστε να μπορεί να πουλήσει την εφαρμογή του μέσω του App Store.

Στο επόμενο κεφάλαιο γίνεται μια προσπάθεια κατανόησης του προγράμματος και της γλώσσας προγραμματισμού που χρησιμοποιεί, από κάποιον όχι ιδιαίτερα εξοικειωμένο χρήστη, με την παρουσίαση ενός πολύ απλού προγράμματος του τύπου "hello world".

Στη συνέχεια ο χρήστης μπαίνει περισσότερο στο κλίμα μέσα από μια πιο πολύπλοκη εφαρμογή που κάνει χρήση έτοιμες βιβλιοθήκες του Flash για το iPhone και συγκεκριμένα για ένα νέο feature, το επιταχυνσιόμετρο. Γίνεται λεπτομερής ανάλυση του κώδικα, όπως και των βημάτων της μετατροπής της εφαρμογής, από αρχείο που τρέχει απλά σε υπολογιστή σε αρχείο που θα τρέχει στο iPhone. Στη συνέχεια το αρχείο περνάει στο κινητό, και η εφαρμογή δοκιμάζεται απευθείας από τον προγραμματιστή.

Τέλος αναλύονται τα πιο σημαντικά σημεία του κώδικα της κύριας εφαρμογής, δηλαδή του παιχνιδιού. Περιγράφονται τα βήματα που ακολουθήθηκαν, τα προβλήματα που προέκυψαν και τα σημεία που μπορούν να βελτιωθούν.

## Περιεχόμενα

Abstract .....	iii
Σύνοψη .....	iv
Εισαγωγή .....	1
Κεφάλαιο 1 - Smartphones .....	3
1.1 Τι είναι τα Smartphones .....	3
1.2 Η ανάγκη για Smartphones .....	3
1.3 Ο "θάνατος" των PCs.....	4
1.4 Ιστορία .....	5
1.4.1 Android και iPhone .....	5
1.5 Ο Νόμος του Android .....	6
1.6 Λειτουργικά Συστήματα .....	7
1.7 Application Stores .....	11
Κεφάλαιο 2 – iPhone.....	12
2.1 Τι είναι το iPhone .....	12
2.2 Τεχνικά Χαρακτηριστικά.....	13
2.2.1 Οθόνη αφής.....	13
2.2.2 Άλλοι τρόποι εισόδου .....	14
2.2.3 Κινητό τηλέφωνο .....	14
2.2.4 Αποθηκευτικός χώρος.....	15
2.2.5 Κάμερα .....	15
2.2.6 Βίντεο .....	15
2.2.7 Μπαταρία.....	15
2.2.8 Διαδίκτυο .....	16
2.2.9 E-mail .....	16
2.2.10 Επιταχυνσιόμετρο (Accelerometer) – Γυροσκόπιο (Gyroscope).....	16
2.2.11 Λειτουργικό σύστημα iOS 4.0.....	17
2.3 Jailbreak.....	17
2.3.1 Τι είναι;.....	17
2.3.2 Τι είναι το Cydia;.....	18
2.3.3 Πως γίνεται το jailbreak? .....	19

2.3.4 Τι είναι τα exploits? .....	19
2.3.5 Σύγκριση iPhone 2G, 3G, 3GS, 4G.....	20
Κεφάλαιο 3 - ADOBE FLASH CS5 .....	22
3.1 Ιστορική Αναδρομή Flash.....	22
3.2 Ιστορία Adobe – Apple.....	22
3.3 Εισαγωγή στο Flash.....	23
3.3.1 Τι είναι.....	23
3.3.2 Που χρησιμοποιείται .....	23
3.3.3 Γενιές Flash μέχρι CS5 – Εξέλιξη Actionscript 2.0 - 3.0 .....	24
3.3.4 Πόσο διαφορετική είναι η ActionScript 3? .....	25
3.3.5 Πως δουλεύει .....	25
3.3.6 Flash Player.....	25
3.3.7 Συμβατότητα .....	26
3.3.8 Κριτική.....	26
3.3.9 Flash και HTML .....	27
3.3.10 HTML5: Τι είναι.....	27
3.3.11 Η HTML5 και το CSS3 φέρνουν το τέλος Flash? .....	27
3.3.12 Πλεονεκτήματα – Μειονεκτήματα .....	28
3.4 Εισαγωγή στο Flash CS5 .....	29
3.4.1 Απαιτήσεις Συστήματος .....	29
3.4.2 Τι καινούριο υπάρχει.....	29
3.4.3 Λειτουργίες που αφαιρέθηκαν .....	30
3.4.4 Γνωστά Προβλήματα (Bugs).....	31
3.4.5 Flash CS5 iPhone Application feature .....	31
3.4.6 Διαμάχη Adobe – Apple .....	31
3.4.7 Πως δουλεύει .....	33
3.4.8 Τεχνική Ανάλυση .....	34
3.4.9 Στάδια Ανάπτυξης Εφαρμογής iPhone .....	35
3.4.10 Βήματα Κλειδιά .....	35
3.4.11 Πακετάρισμα – Τεχνικές Λεπτομέρειες .....	37
3.4.12 Η Αρχιτεκτονική ARM.....	37

3.4.13 Εικονική Μηχανή Χαμηλού Επιπέδου - LLVM (Low Level Virtual Machine).....	38
3.4.14 Σημαντικές Συμβουλές.....	39
3.4.15 Αξιοποίηση Εφαρμογών (App Store).....	40
3.4.16 Απόρριψη των εφαρμογών.....	40
3.4.17 Πλεονεκτήματα – Μειονεκτήματα Flash CS5 έναντι άλλων εργαλείων παραγωγής εφαρμογών (SDKs) για iPhone.....	40
Κεφάλαιο 4 – Εισαγωγή στην Actionscript.....	42
4.1 Hello world!.....	42
Κεφάλαιο 5 – Παράδειγμα Actionscript.....	44
5.1 Ένα απλό παράδειγμα εφαρμογής iPhone.....	44
5.2 Ο κώδικας της εφαρμογής.....	45
5.3 Μετατροπή σε ipa.....	47
5.4 Δοκιμή της εφαρμογής στο iPhone.....	49
Κεφάλαιο 6 – Η κύρια εφαρμογή.....	51
6.1 Ανάλυση και επεξήγηση του κώδικα.....	51
6.2 Προβλήματα που προέκυψαν.....	61
6.3 Σημεία Βελτίωσης.....	62
Βιβλιογραφία.....	63
Προγράμματα που χρησιμοποιήθηκαν.....	63

## Εισαγωγή

Η αναφορά που ακολουθεί αποτελεί λεπτομερή καταγραφή των στόχων, της μεθοδολογίας, των προβλημάτων, των λύσεων αυτών και των αποτελεσμάτων της πτυχιακής εργασίας του συντάκτη αυτού του κειμένου και υποβάλλεται στο Τμήμα Εφαρμοσμένης Πληροφορικής & Πολυμέσων του Α.Τ.Ε.Ι. Κρήτης στα πλαίσια της ολοκλήρωσης μέρους των υποχρεώσεων του σπουδαστή για την απόκτηση του τίτλου σπουδών του τμήματος.

Στην παρούσα εργασία με θέμα **Ανάπτυξη Γραφικού Παιχνιδιού σε iPhone** παρουσιάζεται μία εφαρμογή η οποία στηρίζεται πάνω στο Flash CS5 της Adobe και προορίζεται για χρήση στη συγκεκριμένη συσκευή και πώληση μέσω Internet (μέσω του app store της Apple). Το συγκεκριμένο ηλεκτρονικό κατάστημα παρουσιάζει μεγάλη επισκεψιμότητα, και μέχρι πρότινος, ο μοναδικός τρόπος για έναν προγραμματιστή να δημιουργήσει μία τέτοια εφαρμογή, ήταν αποκλειστικά μέσω λογισμικού της Apple, και μόνο με τη χρήση της γλώσσας προγραμματισμού *objective C*. Επιπλέον, το λογισμικό αυτό τρέχει μόνο στο λειτουργικό σύστημα της εταιρείας, κάτι που καθιστούσε αναγκαία την ύπαρξη υπολογιστή Mac. Η έλλειψη γνώσης πολλών χρηστών στη συγκεκριμένη γλώσσα, αλλά και η μη δυνατότητα πρόσβασης από πολλούς σε υπολογιστή της Apple, οδήγησαν την Adobe στη δημιουργία ενός προγράμματος που θα επέτρεπε στους προγραμματιστές, χρησιμοποιώντας τις ήδη γνώσεις τους πάνω σε μία πιο δημοφιλή γλώσσα, όπως η Actionscript, να δημιουργούν και ταυτόχρονα να ανεβάζουν στο app store εφαρμογές για το επιτυχημένο smartphone.

Πάνω σε αυτήν την πλατφόρμα (Flash CS5), βασίστηκε η εφαρμογή στην οποία αναφέρεται η συγκεκριμένη πτυχιακή εργασία με απώτερο στόχο την ανάδειξη της αλματώδους αύξησης της ζήτησης των εφαρμογών για Smartphones. Πιο συγκεκριμένα η εφαρμογή αυτή υλοποιεί το γνωστό παιχνίδι στο τάβλι, που ονομάζεται **πλακωτό**.

Όσον αφορά τη δομή της συγκεκριμένης εργασίας αρχικά γίνεται μία εκτενής αναφορά στα Smartphones γενικότερα, δηλαδή στη νέα γενιά κινητών τηλεφώνων που ανήκει και το iPhone, και η οποία περιλαμβάνει μεταξύ άλλων αύξηση της υπολογιστικής ισχύος, σύνδεση στο internet, οθόνη αφής κ.α. Στη συνέχεια επιχειρείται μία εκτενέστερη περιγραφή των λειτουργιών του iPhone, καθώς και του Flash CS5, προγράμματος του οποίου χρησιμοποιήθηκε για τη δημιουργία της εφαρμογής. Έπειτα γίνεται αναφορά στον τρόπο λειτουργίας του προγράμματος για την ευκολότερη κατανόηση της γλώσσας προγραμματισμού Actionscript, μέσω της



δημιουργίας ενός πολύ απλού προγράμματος του τύπου "hello world". Ακολουθεί ένα πιο πολύπλοκο πρόγραμμα το οποίο ενσωματώνει έτοιμες βιβλιοθήκες του Flash, και αναδεικνύει μέχρι ενός σημείου τις δυνατότητες που προσφέρει το πρόγραμμα. Στο τελευταίο κεφάλαιο γίνεται λεπτομερής επεξήγηση του κώδικα που υλοποιεί την εφαρμογή. Παρατίθενται επίσης στιγμιότυπα κατά την εξέλιξη της.

# Κεφάλαιο 1 - Smartphones

## 1.1 Τι είναι τα Smartphones

Το Smartphone είναι ένα κινητό τηλέφωνο που προσφέρει πιο αναπτυγμένες υπολογιστικές δυνατότητες από τα υπόλοιπα κινητά τηλέφωνα προηγούμενης γενιάς. Μπορεί γενικά τα κινητά τηλέφωνα να θεωρούνται περιορισμένης υπολογιστικής ισχύς υπολογιστές, ενσωματωμένοι σε μία συσκευή τηλεφώνου, αλλά ενώ τα παλαιότερα κινητά μπορούν να τρέχουν εφαρμογές βασισμένες κυρίως σε πλατφόρμες όπως η Java ME, ένα Smartphone επιτρέπει στο χρήστη να εγκαταστήσει πιο προηγμένες εφαρμογές. Τα Smartphones τρέχουν ολοκληρωμένα λειτουργικά συστήματα παρέχοντας έτσι μία πλατφόρμα για τους προγραμματιστές και ενδιαφέροντα προγράμματα για τους απλούς χρήστες.



Figure 1 - Μερικά από τα πιο δημοφιλή Smartphones

Η αύξηση της ζήτησης για προηγμένες κινητές συσκευές που περιέχουν ισχυρούς επεξεργαστές, άφθονη μνήμη, μεγαλύτερες οθόνες και ανοιχτά λειτουργικά συστήματα, έχει δημιουργήσει αύξηση των πωλήσεων που ξεπερνά τα τελευταία χρόνια την αγορά των υπόλοιπων κινητών τηλεφώνων.

## 1.2 Η ανάγκη για Smartphones

Από την πρώτη στιγμή οι εταιρείες συνειδητοποίησαν ότι ο κόσμος χρειαζόταν κάτι παραπάνω από ένα απλό κινητό για να πραγματοποιεί τις κλήσεις του. Ήθελε να παίζει, να σερφάρει στο Ίντερνετ, να βλέπει το πρόσωπο με το οποίο συνομιλεί στην άλλη άκρη και να τον βλέπει κι εκείνος, να μπορεί να διαβάζει αρχεία διάφορων format όπως pdfs, να κάνει πράξεις

κ.ο.κ. Ήθελε πρακτικά έναν υπολογιστή και μάλιστα στο μέγεθος ενός κινητού τηλεφώνου. Έτσι σιγά σιγά όλες οι εταιρείες άρχισαν να ενσωματώνουν όλο και περισσότερα τεχνικά χαρακτηριστικά και δυνατότητες για να ικανοποιηθούν οι ανάγκες των καταναλωτών. Μάλιστα δεν σταμάτησαν εκεί. Οι εφαρμογές τόσο των ίδιων των εταιρειών αλλά και τρίτων προγραμματιστών για συγκεκριμένα Smartphones σημείωσαν αξιοσημείωτη άνθιση, με αποτέλεσμα τη δημιουργία ειδικών ιστοτόπων για το κατέβασμα αυτών των εφαρμογών και σαν συνέπεια την αναγγελία αυτών των εταιρειών σε παγκόσμιους κολοσσούς.

### 1.3 Ο "θάνατος" των PCs

Κάθε χρόνο ακούμε το κύκνειο άσμα των παλιών καλών επιτραπέζιων υπολογιστών (PCs). Τελευταία ακόμη και το μεγάλο ειδησεογραφικό κανάλι Fox News, ανακοίνωσε ότι... το παλιό κλασικό PC έχει πεθάνει. Το Desktop PC στην εποχή των Smartphones και των Tablets, αποτελεί ένα κατάλοιπο χωρίς χρησιμότητα.



Figure 2 - Ο κλασικός επιτραπέζιος υπολογιστής φαντάζει πλέον τεράστιος

Τόσο καθοδόν όσο και στο σπίτι χρησιμοποιούνται πλέον smartphones, notebooks, netbooks και Tablets. Τα απαιτητικά σε hardware παιχνίδια πλέον μπορούν να εκτελεστούν καλύτερα σε κονσόλες όπως το PlayStation 3 ή το Xbox 360.

Μόνο κάποιες ομάδες όπως οι σχεδιαστές και οι προγραμματιστές χρειάζονται ακόμα τα Desktop PCs σύμφωνα με το Fox News. Για τους υπόλοιπους τα PCs έχουν πεθάνει.

## 1.4 Ιστορία

Το πρώτο smartphone λεγόταν Simon. Σχεδιάστηκε απ' την IBM το 1992 σαν πρωτότυπο και βγήκε στην κυκλοφορία την επόμενη χρονιά από την BellSouth. Εκτός από την υποστάση του σαν κινητό τηλέφωνο περιείχε επίσης ημερολόγιο, βιβλίο διευθύνσεων, ρολόι παγκόσμιας ώρας, υπολογιστή αριθμητικών πράξεων, σημειωματάριο, e-mail, δυνατότητα αποστολής και λήψης fax, καθώς και παιχνίδια. Δεν είχε φυσικά κουμπιά για την πραγματοποίηση κλήσης. Αντ' αυτού ο χρήστης χρησιμοποιούσε οθόνη αφής για να επιλέξει τα νούμερα. Μπορεί με βάση τα στάνταρ του σήμερα αυτά να μην είναι κάτι δύσκολο, αλλά για εκείνη την εποχή ήταν ένα πολύ προηγμένο χαρακτηριστικό.

Τότε αποφασίζει και η Nokia να μπει στην αγορά με το Nokia 9000 που κυκλοφόρησε το 1996. Αυτό το νέου είδους επιπαλάμιου υπολογιστή ήταν το αποτέλεσμα από μία προσπάθεια μίξης, από τη μία του πολύ πετυχημένου και ακριβού PDA που είχε σχεδιάσει η Hewlett Packard και από την άλλη του καλύτερου κινητού της Nokia εκείνη την περίοδο. Το Nokia 9210 ήταν το πρώτο κινητό με έγχρωμη οθόνη και στην ουσία το πρώτο Smartphone που έτρεχε ανοιχτό λειτουργικό σύστημα. Ακολούθησαν πολλές νέες αναβαθμισμένες συσκευές, που έφεραν κάμερες και wi-fi. Το 1997 η Ericsson βγάζει στην κυκλοφορία το GS88, την πρώτη συσκευή με την ετικέτα 'Smartphone'. Το 2000 η Ericson κυκλοφορεί το Smartphone με οθόνη αφής, με κωδικό όνομα R380, την πρώτη συσκευή που χρησιμοποιούσε το λειτουργικό σύστημα Symbian. Ακολουθεί δύο χρόνια μετά το P800 το πρώτο Smartphone με ενσωματωμένη κάμερα. Το 2002 η Handspring κυκλοφορεί το Smartphone Palm OS Treo αξιοποιώντας ένα πλήρες πληκτρολόγιο που συνδυάζει ασύρματη σύνδεση στο Internet, email, ημερολόγιο, οργανωτή επαφών και ενέχει τη δυνατότητα ενσωμάτωσης εφαρμογών τρίτων μέσω υπολογιστή. Το 2002 η RIM κυκλοφορεί το πρώτο BlackBerry, το οποίο ήταν το πρώτο Smartphone που είχε βελτιστοποιηθεί ειδικά για χρήση email, με ασύρματη σύνδεση στο Internet. Το 2007 η Nokia λανσάρει το N95 το οποίο είχε ενσωματωμένα, μία ευρεία γκάμα από χαρακτηριστικά όπως GPS, κάμερα 5MPX, 3G, wi-fi και TV-out. Στα χρόνια που ακολούθησαν αυτά τα χαρακτηριστικά έγιναν πλέον δεδομένα για τα περισσότερα Smartphones. Το 2010 η Nokia κυκλοφορεί το N8, την πρώτη συσκευή που χρησιμοποιούσε το νέο λειτουργικό Symbian<sup>3</sup>. Είχε μία απ τις καλύτερες κάμερες που είχε μέχρι τότε κινητό τηλέφωνο και ένα από τα καλύτερα συστήματα δορυφορικής πλοήγησης.

### 1.4.1 Android και iPhone

Στα χρόνια που ακολουθούν κάνουν την εμφάνισή τους δύο απ τα σημαντικότερα ορόσημα : η συσκευή iPhone της Apple και το λειτουργικό σύστημα για Smartphones της Google. Το 2007 η Apple παρουσιάζει το πρώτο iPhone της. Αρχικά είχε υψηλό κόστος, στην τιμή των \$500 για το φθηνότερο εκ των δύο μοντέλων και με μία διετή σύμβαση. Ήταν ένα από τα πρώτα smartphones που ελεγχόταν κυρίως μέσω της οθόνης αφής του. (Τα άλλα δύο ήταν το LG Prada και το HTC Touch που επίσης κυκλοφόρησαν την ίδια χρονιά). Ήταν το πρώτο κινητό τηλέφωνο που χρησιμοποιούσε χωρητική οθόνη πολυαφής, (δηλαδή μπορούσε να λάβει υπόψη 2 ταυτόχρονα Input του χρήστη, όπως για παράδειγμα στο ζουμάρισμα/ξεζουμάρισμα μιας φωτογραφίας), και περιελάμβανε ένα web browser πολύ ανώτερο κάθε άλλου που προσφερόταν από εκείνους των ανταγωνιστών της. Κατά τη στιγμή του λανσαρίσματος του iPhone ήταν

αμφισβητήσιμο αν στην πραγματικότητα ήταν ένα smartphone καθώς η πρώτη γενιά στερήθηκε τη δυνατότητα να χρησιμοποιεί επίσημα εφαρμογές τρίτων. Μια διαδικασία που ονομάζεται jailbreaking προέκυψε γρήγορα για να παράσχει ανεπίσημες τέτοιες εφαρμογές. Ο Steve Jobs, διευθύνων σύμβουλος της Apple Inc, δήλωσε δημόσια ότι το iPhone στερούνταν την υποστήριξη του 3G λόγω της ανεπαρκούς ανάπτυξης, της κατανάλωσης ισχύος και των φυσικού μεγέθους απαιτήσεων των 3G chipsets εκείνη τη χρονική περίοδο. Ωστόσο έχει φημολογηθεί ότι το δίκτυο παρόχων CDMA2000 (Verizon, Sprint) αρνήθηκε να επιτρέψει το iPhone στο δίκτυο τους, επειδή ο Jobs ήθελε τον απόλυτο έλεγχο του app store που σχετίζεται με το iPhone.

Το λειτουργικό σύστημα Android για smartphones κυκλοφόρησε το 2008. Το Android είναι μια πλατφόρμα ανοικτού κώδικα που υποστηρίζεται από την Google, μαζί με μεγάλους κατασκευαστές υλικού και λογισμικού (όπως Intel, HTC, ARM, Motorola και Samsung), που αποτελούν την Open Handset Alliance. Το πρώτο τηλέφωνο που χρησιμοποίησε το λειτουργικό σύστημα Android ήταν το HTC Dream γνωστό για τη διανομή του από την T-Mobile. Η σουίτα λογισμικού που περιλαμβάνεται στο τηλέφωνο αποτελείται από ενσωματωμένες αποκλειστικές εφαρμογές της Google όπως χάρτες, το ημερολόγιο και το Gmail και ένα πλήρες πρόγραμμα περιήγησης Ιστού σε HTML. Εφαρμογές τρίτων είναι διαθέσιμες μέσω του Android Market (που κυκλοφόρησε τον Οκτώβριο του 2008), συμπεριλαμβανομένων τόσο των δωρεάν όσο και των επί πληρωμή εφαρμογών.

Τον Ιούλιο του 2008 η Apple εισήγαγε το δεύτερης γενιάς iPhone, το οποίο είχε 3G υποστήριξη και χαμηλότερη τιμή καταλόγου. Μαζί με την κυκλοφορία αυτού, η Apple δημιούργησε επίσης το App Store με δωρεάν και επί πληρωμή εφαρμογές. Το App Store μπορεί να παραδώσει τις εφαρμογές smartphone που αναπτύσσονται από τρίτους απευθείας στο iPhone μέσω WiFi ή δικτύου κινητής τηλεφωνίας χωρίς αναγκαία τη χρήση υπολογιστή για τη λήψη αυτής. Το App Store είναι τεράστια επιτυχία για την Apple και μέχρι τον Φεβρουάριο του 2011 φιλοξένησε περισσότερες από 350.000 εφαρμογές. Το App Store έφτασε τα τρία δισεκατομμύρια λήψεις εφαρμογών στις αρχές Ιανουαρίου του 2010 και τα δέκα δισεκατομμύρια ύστερα από μόλις έναν χρόνο.

Τον Ιανουάριο του 2010 η Google εγκαινίασε το smartphone 'Nexus One' που χρησιμοποιεί το Android λειτουργικό σύστημα. Παρά το γεγονός ότι το Android έχει multi-touch δυνατότητες, η Google αρχικά αφαίρεσε αυτό το χαρακτηριστικό από το Nexus One, αλλά προστέθηκε μέσω ενός firmware update στις 2 Φεβρουαρίου 2010.

Σύμφωνα με την Gartner στην έκθεσή τους το Νοέμβριο του 2010, οι συνολικές πωλήσεις smartphones διπλασιάστηκαν μέσα σε ένα έτος και τώρα πλέον τα smartphones αντιπροσωπεύουν το 19,3 τοις εκατό των συνολικών πωλήσεων κινητών τηλεφώνων. Κατά τα τέλη του 2009 και 2010 το μερίδιο αγοράς Android smartphone έχει αυξηθεί με ταχείς ρυθμούς.

## 1.5 Ο Νόμος του Android

Βέβαια το να αγοράζει κανείς smartphone αποδεικνύεται δίκικο μαχαίρι. Ναι μεν ο χρήστης μπορεί να έχει το τελευταίας γενιάς κινητό αλλά σε λιγότερο από μισό χρόνο θα βγει το νέο σαφώς ανώτερο μοντέλο. Αυτό το φαινόμενο έχει βαπτιστεί ως "Ο νόμος του Android".

Η αγορά των smartphones ως γνωστόν, παραμένει από τις πιο ανταγωνιστικές με τις συσκευές να αντικαθιστούν η μία την άλλη σε χρόνο μηδέν.

*«Μετά από τις συνεχείς αλλαγές στην εξουσία κατέφτασε από την εξορία ο "επαναστάτης" Samsung Galaxy S ο οποίος πολέμησε το "στέμμα" και πήρε την εξουσία στα χέρια του ως νέος Ναπολέων του smartphone-κόσμου. Βέβαια με την πάροδο της σύντομης κυριαρχίας του Galaxy S, το Nexus S ήρθε για να θρονιαστεί στις Βερσαλλίες του παλατιού της Google. Τα αυλικά παιχνίδια, όμως είναι πολλά, και αυτός ο πρίγκιπας δεν θα αργήσει να εκθρονιστεί από τον επόμενο, ίσως το iPhone 5».*<sup>[1]</sup>

Μερικά χρόνια πριν τα κινητά ήταν σχεδιασμένα εξ' ολοκλήρου από τις κατασκευάστριες εταιρίες όπως οι LG, HTC, Motorola και Samsung. Σαν συνέπεια, οι νέες συσκευές που δημιουργούσαν έκαναν αρκετό καιρό για να ολοκληρωθούν και να εισαχθούν στην αγορά. Επιπλέον οι συσκευές έπρεπε να μείνουν αρκετό χρόνο στην αγορά έτσι ώστε να μεγιστοποιηθούν οι πωλήσεις και η επένδυση των εταιριών να τους αποφέρει κέρδος.

Τώρα πια το Android, με το λειτουργικό ανοιχτού κώδικα έχει αλλάξει τα δεδομένα και το φαινόμενο του "Νόμου του Android" δεν φαίνεται πως θα αλλάξει σύντομα.

Τελικά τα κινητά τηλέφωνα μπήκαν στο χορό της καλπάζουσας τεχνολογίας. Στην τελική επιλογή όμως οι χρήστες είναι αυτοί που καλούνται να κάνουν σωστή και σοφή επιλογή σύμφωνα πάντα με τις ανάγκες και τις επιθυμίες τους.

## 1.6 Λειτουργικά Συστήματα

Τις περισσότερες φορές όταν αναφερόμαστε σε ένα Λειτουργικό Σύστημα (OS – Operating System), εννοούμε το λογισμικό που ελέγχει κάποιον επιτραπέζιο ή φορητό υπολογιστή (με τα Windows, Mac OS και Linux να είναι τα πιο διαδεδομένα). Ωστόσο στους περισσότερους ανθρώπους, κυρίως σε εκείνους που δεν έχουν ιδιαίτερες επαφές με την τεχνολογία, διαφεύγει το γεγονός ότι όλες οι ηλεκτρονικές συσκευές χρειάζονται ένα λειτουργικό σύστημα για να μπορούν να λειτουργήσουν. Στο εξής, όταν θα αναφερόμαστε στην έννοια Λειτουργικό Σύστημα, θα εννοούμε το λογισμικό που είναι υπεύθυνο για τη σωστή λειτουργία των Smartphones.

<sup>[1]</sup> <http://24wro.blogspot.com/2011/02/dt.html>

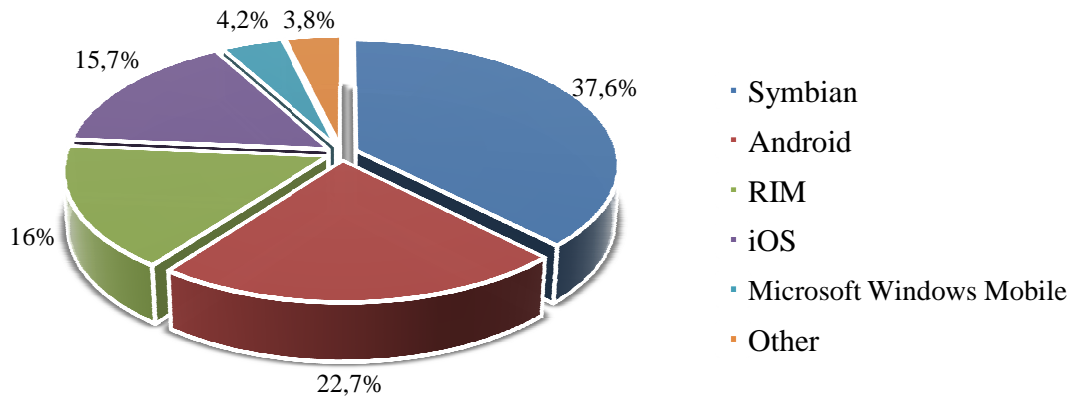


Figure 3 - Ανάλυση της Gartner πάνω στις πωλήσεις Smartphones, Q4 2010.

Η αυξανόμενη σημασία των φορητών συσκευών έχει προκαλέσει έντονο ανταγωνισμό ανάμεσα στους τεχνολογικούς κολοσσούς, όπως η Google, η Microsoft, η Apple και η Nokia σε μια προσπάθεια να καταλάβουν το μεγαλύτερο μερίδιο της αγοράς. Οι Palm, Research In Motion και Ericsson είναι επίσης σημαντικές επιχειρήσεις στον τομέα της φορητής πλατφόρμας. Τον Νοέμβριο του 2007 η Google δημιούργησε μια συμμαχία ανοικτού κώδικα (open source) βασισμένη σε Linux με σκοπό να διεισδύσει στην αγορά της φορητής πλατφόρμας, παρακινώντας τους αγοραστές σε μια ραγδαία αυξανόμενη κατανάλωση της φορητής πλατφόρμας.

Λειτουργικά συστήματα που συναντώνται σε smartphones περιλαμβάνουν το Symbian της Nokia, το Android της Google, το iOS της Apple, το BlackBerry OS της RIM, το Windows Phone της Microsoft, το Linux, το WebOS της Palm/HP, το Bada της Samsung, τα Maemo και Meego της Nokia καθώς και πολλά άλλα. Τα Android, Bada, WebOS και Maemo έχουν υλοποιηθεί σε Linux, και το iOS προέρχεται από τα λειτουργικά συστήματα BSD και NeXTSTEP τα οποία σχετίζονται όλα με το Unix.

Τα πιο κοινά Λειτουργικά Συστήματα που χρησιμοποιήθηκαν από smartphones ως τρίτο τετράμηνο του 2010 σύμφωνα με τις πωλήσεις είναι τα παρακάτω:

- **Symbian OS** από την Symbian Foundation.

Η Symbian έχει το μεγαλύτερο μερίδιο στις περισσότερες αγορές παγκοσμίως αλλά υστερεί σε σχέση με άλλες εταιρείες, στην σχετικά μικρή αλλά αρκετά σημαντική αγορά της βόρειου Αμερικής. Παρομοιάζεται με την επιτυχία του μεγαλύτερου μετόχου και πελάτη της, της

Nokia, σε όλες τις αγορές εκτός της Ιαπωνίας. Στην Ιαπωνία η Symbian είναι ισχυρή λόγω της σχέσης της με την NTT DoCoMo, με μόνο μία από τις 44 χειροσυσσκευές της Symbian να έχει κυκλοφορήσει στην Ιαπωνία από την Nokia. Έχει χρησιμοποιηθεί από πολλούς σημαντικούς κατασκευαστές χειροσυσσκευών, συμπεριλαμβανομένου των BenQ, Fujitsu, LG, Mitsubishi, Motorola, Nokia, Samsung, Sharp, και Sony Ericsson. Μέχρι πριν το 2009 η Symbian υποστήριζε διεπαφές πολλαπλού χρήστη (multiple user interfaces) όπως για παράδειγμα στα UIQ της UIQ Technologies, S60 της Nokia, και στο MOAP της NTT DOCOMO. Ως μέρος του σχηματισμού της Symbian Platform το 2009, οι τρεις αυτές διεπαφές (UI – user interfaces) συμπύχθηκαν σε μια μόνο πλατφόρμα η οποία είναι τώρα ανοικτού κώδικα.

- **Android** από την Google Inc. (ανοικτού κώδικα, Apache)

Το Android υλοποιήθηκε από μια μικρή αρχικά εταιρεία που αγοράστηκε από την Google Inc., και η Google συνεχίζει να εκσυγχρονίζει το λογισμικό. Το Android είναι μία ανοικτού κώδικα, παράγωγη του Linux πλατφόρμα που υποστηρίζεται από τη Google, σε συνεργασία με τους μεγαλύτερους κατασκευαστές υλικού και λογισμικού (όπως οι Intel, HTC, ARM, Samsung, Motorola, eBay), που απαρτίζουν την Open Handset Alliance. Έχουν κυκλοφορήσει επτά εκδοχές του Android – Android 1.0, 1.5, 1.6, 2.0, 2.1, 2.2 και 2.3. Όλες τους επονομαζόμενες από κάποιο επιδόρπιο όπως Cupcake (μικρό κέικ) ή Frozen Yogurt (παγωμένο γιαούρτι). Οι περισσότεροι μεγάλοι πάροχοι υπηρεσιών κινητής τηλεφωνίας χρησιμοποιούν κάποια συσκευή Android.

- **iOS** από την Apple Inc. (κλειστού κώδικα, ιδιόκτητο)

Τα Apple iPhone, iPod Touch και iPad χρησιμοποιούν ένα λειτουργικό σύστημα που ονομάζεται iOS, το οποίο είναι παράγωγο του Mac OS X που υπάρχει στα Mac. Εφαρμογές τρίτων δεν υποστηρίζονταν επίσημα μέχρι την κυκλοφορία του iOS 2.0 τον Ιούλιο του 2008. Μέχρι τότε το “jailbreaking” επέτρεπε σε εφαρμογές τρίτων να εγκατασταθούν και αυτή η μέθοδος είναι ακόμη διαθέσιμη. Σήμερα όλες οι iOS συσκευές αναπτύσσονται από την Apple και υλοποιούνται από την Foxconn ή άλλους συνεργάτες της Apple.

- **BlackBerry OS** από την RIM. (κλειστού κώδικα, ιδιόκτητο)

Αυτό το λειτουργικό σύστημα εστιάζει στον εύκολο χειρισμό και είχε σχεδιαστεί αρχικά για επαγγελματική χρήση. Πρόσφατα έχει παρατηρηθεί ένα κύμα εφαρμογών τρίτων και έχει βελτιωθεί για να προσφέρει πλήρη υποστήριξη στα πολυμέσα. Σήμερα η App World του BlackBerry έχει πάνω από 15.000 εφαρμογές διαθέσιμες για download.

- **Windows Mobile** από την Microsoft. (κλειστού κώδικα, ιδιόκτητο)

Το λειτουργικό σύστημα Windows CE και το middleware Windows Mobile είναι ευρέως διαδεδομένα στην Ασία. Οι δύο βελτιωμένες παραλλαγές αυτού του λειτουργικού συστήματος, το Windows Mobile 6 Professional (για συσκευές με οθόνη αφής) και το Windows Mobile 6 Standard κυκλοφόρησαν τον Φεβρουάριο 2007. Κατακρίθηκε για την παροχή μιας διεπαφής που δεν ήταν βελτιστοποιημένη για εισαγωγή πληροφορίας με το άγγιγμα των δακτύλων, αντί αυτού,



μπορεί να χρησιμοποιηθεί καλύτερα με γραφίδα. Ωστόσο σε αντίθεση με το λειτουργικό σύστημα του iPhone, υποστηρίζει ταυτόχρονα και ρυθμίσεις οθόνης αφής και πληκτρολογίου.

Το μερίδιο αγοράς του Windows Mobile μειώθηκε απότομα τα τελευταία χρόνια σε μόλις 5% το δεύτερο τρίμηνο του 2010. Η Microsoft καταργεί σταδιακά το Windows Mobile OS σε ειδικευμένες αγορές και εστιάζει στη νέα της πλατφόρμα, το Windows Phone.

- **Windows Phone** από την Microsoft. (κλειστού κώδικα, ιδιόκτητο)

Το Φεβρουάριο του 2010, η Microsoft, κυκλοφόρησε το λειτουργικό σύστημα κινητού τηλεφώνου επόμενης γενιάς, το Windows Phone 7. Το νέο λειτουργικό σύστημα κινητού τηλεφώνου περιλαμβάνει μία εντελώς νέα διεπαφή χρήστη (UI) εμπνευσμένη από την "Metro Design Language". Περιλαμβάνει πλήρως ολοκληρωμένες εφαρμογές της Microsoft, όπως τα Windows Live, Zune, Xbox Live και Bing, αλλά ενσωματώνεται και με άλλες μη-Microsoft υπηρεσίες όπως Facebook και Google accounts. Η νέα πλατφόρμα λειτουργικού συστήματος έχει λάβει κάποια θετική υποδοχή από τον τεχνολογικό τύπο. Σαν νέα πλατφόρμα το Windows Phone 7 δεν έχει συμβατότητα προς τα πίσω με Windows Mobile εφαρμογές και μερικά χαρακτηριστικά power-user που υπήρχαν στα Windows Mobile δεν θα είναι παρόντα παρά μόνο σε μελλοντικές ενημερώσεις.

Πρόσφατα μάλιστα ανακοινώθηκε ότι Microsoft και Nokia θα συνεργαστούν για να λανσάρουν το νέο λειτουργικό σύστημα. Η Nokia θα ασχολείται με τα λειτουργικά συστήματα ενώ η μηχανή αναζήτησης Bing και το σύστημα διαφήμισης adCenter θα παρέχουν τις αντίστοιχες υπηρεσίες στις συσκευές της. Όσον αφορά την εφαρμογή Nokia Maps, θα ενσωματωθεί στις υπηρεσίες χαρτογράφησης της Microsoft και στη μηχανή αναζήτησης Bing, ενώ και το κατάστημα εφαρμογών της Nokia θα ενσωματωθεί στο Microsoft Marketplace!

- Το λειτουργικό σύστημα **Linux** (ανοικτού κώδικα, GPL)

Το Linux είναι ισχυρότερο στην Κίνα όπου χρησιμοποιείται από την Motorola και στην Ιαπωνία που χρησιμοποιείται από την DoCoMo. Αντί να αποτελεί πλατφόρμα από μόνο του το Linux χρησιμοποιείται σαν βάση για ένα πλήθος από διαφορετικές πλατφόρμες που αναπτύχθηκαν από διάφορους πωλητές, συμπεριλαμβανομένου τους Android, LiMo, Maemo, Openmoko και Qt Extended, οι οποίοι είναι κατά μεγάλο ποσοστό μη συμβατοί. Η PalmSource (πλέον Access) κινείται προς μία διεπαφή που τρέχει σε Linux. Άλλη πλατφόρμα βασισμένη στα Linux αναπτύσσεται από τις Motorola, NEC, NTT DoCoMo, Panasonic, Samsung και Vodafone.

- **Palm webOS** από την HP (με κάποια μέρη ανοιχτού κώδικα) και το **Palm OS/Garnet OS** από την Access Co. (κλειστού κώδικα, ιδιόκτητο)

Το Palm webOS είναι το λειτουργικό σύστημα επόμενης γενιάς της Palm. Το PalmSource παραδοσιακά χρησιμοποιούσε τη δική του πλατφόρμα που ανέπτυξε η Palm Inc. Η Access Linux Platform (ALP) είναι μια βελτίωση που σχεδιάστηκε να κυκλοφορήσει το πρώτο μισό του 2007. Θα χρησιμοποιούσε τεχνικές προδιαγραφές από το Linux Phone Standards Forum. Το Access Linux Platform θα περιελάμβανε ένα επίπεδο προσομοίωσης για να υποστηρίζει εφαρμογές ανεπτυγμένες από Palm-based συσκευές.

- **Bada** από την Samsung Electronics (κλειστού κώδικα, ιδιόκτητο)

Αυτό είναι ένα λειτουργικό σύστημα που δημιουργήθηκε από την Samsung Electronics. Η Samsung ισχυρίζεται ότι το bada γρήγορα θα αντικαταστήσει την ιδιόκτητη feature phone πλατφόρμα, μετατρέποντας feature phones σε smartphones. Το όνομα bada προέρχεται από την κορεάτικη λέξη  $\text{바다}$  που σημαίνει ωκεανός ή θάλασσα. Η πρώτη συσκευή που "έτρεξε" το bada ονομάζεται "Wave" και αποκαλύφθηκε στο κοινό στο παγκόσμιο συνέδριο κινητών τηλεφώνων το 2010. Το Wave είναι ένα κινητό με αποκλειστικά οθόνη αφής που λειτουργεί με το νέο αυτό λειτουργικό σύστημα. Με το τηλέφωνο, η Samsung επίσης εισήγαγε ένα app store, με το όνομα Samsung Apps στην αγορά. Έχει γύρω στις 3000 εφαρμογές.

Η Samsung ισχυρίζεται ότι το Bada δεν είναι μια smartphone πλατφόρμα, αλλά μια πλατφόρμα με διαμορφώσιμο αρχιτεκτονικό πυρήνα, ο οποίος επιτρέπει τη χρήση είτε ιδιόκτητων Real-time διαχειριστικών συστημάτων (RTOS) πυρήνα, ή τον πυρήνα Linux. Παρόλο που η Samsung σχεδιάζει να εγκαταστήσει το bada σε πολλά τηλέφωνα, η εταιρία έχει ακόμη ένα μεγάλο αριθμό Android τηλεφώνων που περιμένουν.

Η κουλτούρα ανοιχτού κώδικα έχει εισβάλει στην αγορά των smartphones με ποικίλους τρόπους. Έχουν γίνει προσπάθειες ανοιχτού κώδικα και στο hardware και στο software των smartphones. Το Google Android OS είναι ένα πολύ δημοφιλές λειτουργικό σύστημα ανοιχτού κώδικα. Η Nokia έχει πρωτοβουλία γύρω από το Symbian, και χρησιμοποιεί τον ανοιχτό κώδικα σε όλα τα Symbian smartphones. Στο παρελθόν η Nokia ανέπτυξε τον GNU/Linux-based σύστημα ανοιχτού κώδικα Maemo. Το Maemo από τότε συγχωνεύτηκε με το πρότζεκτ της Intel, το Moblin, για να δημιουργήσουν το MeeGo σύστημα διαχείρισης.

## 1.7 Application Stores

Τα application stores (app stores) ή online καταστήματα, είναι ιστοσελίδες από τις οποίες ο χρήστης μπορεί να κατεβάσει διάφορες εφαρμογές, δωρεάν ή επί πληρωμή, για το smartphone του. Το πιο γνωστό σήμερα είναι το app store της **Apple**, με πάνω από 350.000 εφαρμογές και πάνω από 10 δισεκατομμύρια downloads. Από τον συγκεκριμένο ιστότοπο, ο χρήστης μπορεί να κατεβάσει την εφαρμογή που θέλει είτε από τον υπολογιστή του και έπειτα να την περάσει στο iPhone, είτε κατευθείαν από τη συσκευή μέσω του δικτύου του. Μάλιστα η Apple διεκδικεί τη λέξη "app" σαν trademark. Ωστόσο και άλλες εταιρείες πλέον έχουν το δικό τους ηλεκτρονικό κατάστημα όπως η **Google** που τον Οκτώβριο του 2008 άνοιξε το Android Market, η **RIM** που άνοιξε το δικό της app store τον Απρίλιο του 2009 με την ονομασία Blackberry App World, η **Nokia** με το Ovi store το Μάιο του ίδιου χρόνου, η **Palm** με το Palm App Catalog, η **Microsoft** με το Windows Marketplace for Mobile και η Samsung με το Samsung Apps.

## Κεφάλαιο 2 – iPhone

### 2.1 Τι είναι το iPhone

Το iPhone συγκαταλέγεται στην κατηγορία των smartphones, συσκευών που συνδυάζουν πρόγραμμα αναπαραγωγής πολυμέσων, φωτογραφική μηχανή/βιντεοκάμερα και κινητό τηλέφωνο με ασύρματη πρόσβαση στο Διαδίκτυο. Το iPhone είναι κατασκευή της εταιρείας Apple Inc, η οποία το κυκλοφόρησε στην αγορά των ΗΠΑ στις 29 Ιουνίου 2007. Έχει κυκλοφορήσει στις περισσότερες χώρες της Ευρώπης και στην Ιαπωνία. Διαθέτει χωρητική οθόνη πολυαφής και ελάχιστο υλικό διεπαφής (σχεδόν καθόλου φυσικό πληκτρολόγιο). Οι παρεχόμενες υπηρεσίες διαδικτύου περιλαμβάνουν ηλεκτρονικό ταχυδρομείο και φυλλομετρητή ιστοσελίδων με δυνατότητα σύνδεσης σε ασύρματο δίκτυο Wi-Fi.



Figure 4 - Κάτοψη του iPhone

Η πρώτη έκδοση κινητού τηλεφώνου iPhone κυκλοφόρησε τον Ιούνιο του 2007, το οποίο διέθετε τετραπλή ζώνη (quad band) με τεχνολογία EDGE. Οι χώρες που το διέθεσαν στην αγορά τους μετά τις ΗΠΑ ήταν το Ηνωμένο Βασίλειο, η Γαλλία και η Γερμανία το Νοέμβριο του 2007 ενώ η Ιρλανδία και η Αυστρία το διέθεσαν την άνοιξη του επόμενου έτους. Έναν χρόνο μετά την κυκλοφορία του, τον Ιούλιο του 2008, κυκλοφόρησε το iPhone δεύτερης γενιάς, το iPhone 3G, που περιελάμβανε την υποστήριξη κινητής τηλεφωνίας 3G σε 22 χώρες μέσα σε αυτές και οι έξι πρώτες ενώ αργότερα ξεπέρασε τις 80. Ακόμα μία εξέλιξη του iPhone, το iPhone 3GS, κυκλοφόρησε τον Ιούνιο του 2009 σε Αμερική και Ευρώπη. Στις 24 Ιουλίου 2010 ο CEO της Apple ανακοίνωσε την τέταρτη γενιά του iPhone με ονομασία iPhone 4. Το καινούργιο iPhone

έχει μια σειρά από καινούργια χαρακτηριστικά όπως μεγαλύτερη ανάλυση οθόνης, εμπρόσθια κάμερα για βίντεο κλήσεις, κάμερα 5 megapixel με φλας και νέο σχεδιασμό υψηλής ποιότητας. Το iPhone χρησιμοποιεί τον μικροεπεξεργαστή ARM 1176.

## 2.2 Τεχνικά Χαρακτηριστικά

Το iPhone είχε τόση μεγάλη επιτυχία για 2 κύριους λόγους. Ο πρώτος ήταν ότι μπορεί σαν συσκευή να μην είχε τις λειτουργίες που είχαν άλλα κινητά της εποχής του, ωστόσο αυτά που μπορούσε να κάνει τα έκανε άψογα. Οι τεχνικοί της Apple δεν προσπάθησαν να ενσωματώσουν πάρα πολλές λειτουργίες που όμως θα είναι μέτριες, αλλά πολύ σωστά αφοσιώθηκαν στις λίγες που είχε το πρώτο μοντέλο, και τις έκαναν φοβερές. Για παράδειγμα η πρώτη γενιά iPhone δεν είχε εγγραφή βίντεο, όμως το επιταχυνσιόμετρο και η οθόνη αφής, έκαναν τις άλλες εταιρίες να παγώσουν την εξέλιξη των δικών τους συσκευών και να αντιγράψουν/ενσωματώσουν τα χαρακτηριστικά του iPhone. Ο δεύτερος λόγος που το iPhone έγινε τόσο δημοφιλές, ήταν η προσοχή στην κομψότητα αλλά και τη λειτουργικότητα. Ήταν το πρώτο κινητό με τόσο μεγάλη οθόνη αφής ενώ παράλληλα δεν είχε καθόλου κουμπιά, εκτός από ένα (δεν λαμβάνονται υπόψη τα πλαϊνά). Ας δούμε κάποια απ τα τεχνικά χαρακτηριστικά του τελευταίου μοντέλου αναλυτικότερα.

*Είναι χαρακτηριστική η εικόνα που κυκλοφόρησε λίγο μετά την κυκλοφορία του πρώτου μοντέλου, και αναδείκνυε με γλαφυρό τρόπο τις σημαντικές ελλείψεις του.*



	Stone (40,000 B.C.)	iPhone 3G (2008)
SMS	X	X
Video recording	X	X
Videocall	X	X
Changeable memory cards	X	X
Touchscreen	X	✓

Figure 5 - Γραφική σύγκριση iPhone και αρχαίας λίθου

### 2.2.1 Οθόνη αφής

Η οθόνη αφής είναι μία 9 εκατοστών οθόνη υγρών κρυστάλλων με ανθεκτικό στις γρατζουνιές τζάμι. Η χωρητική οθόνη αφής που δουλεύει μόνο με το γυμνό δάχτυλο είναι σχεδιασμένη και για πολλαπλή αφή. Το τελευταίο μοντέλο έχει ανάλυση 640x960 στα 326 ppi (pixel ανά ίντσα), δηλαδή σχεδόν το διπλάσιο απ' ό,τι τα προηγούμενα μοντέλα. Για την εισαγωγή κειμένου η συσκευή διαθέτει εικονικό πληκτρολόγιο. Έχει αυτόματο διορθωτή και πρόβλεψη λέξεων, καθώς επίσης και δυναμικό λεξικό προσθήκης νέων λέξεων.

### 2.2.2 Άλλοι τρόποι εισόδου

Η οθόνη αποκρίνεται με τρεις τρόπους/αισθητήρες: ένας αισθητήρας προσέγγισης που κλείνει την οθόνη όταν το iPhone πλησιάσει στο πρόσωπο, ένας αισθητήρας περιβάλλοντος φωτισμού, ο οποίος ρυθμίζει την φωτεινότητα της οθόνης και ένα επιταχυνσιόμετρο που αντιλαμβάνεται τον προσανατολισμό της συσκευής.

Ένα και μοναδικό πλήκτρο στο εμπρόσθιο μέρος εμφανίζει το κεντρικό μενού. Οι επιλογές του μενού γίνονται μέσω της οθόνης αφής. Το iPhone χρησιμοποιεί οθόνη "πλήρους σελίδας", με υπομενού εξαρτώμενα από την εκάστοτε λειτουργία, τα οποία εμφανίζονται στο άνω ή κάτω τμήμα της οθόνης, ανάλογα με τον προσανατολισμό της. Οι σελίδες με λεπτομέρειες εμφανίζουν το αντίστοιχο ενός πλήκτρου επιστροφής κατά ένα βήμα πίσω στα μενού.

Το iPhone διαθέτει μόνον τέσσερις διακόπτες στα πλευρά που ρυθμίζουν παραμέτρους όπως την ένταση του ήχου κ.α. Όλες οι υπόλοιπες λειτουργίες εκτελούνται μέσω της οθόνης αφής.

### 2.2.3 Κινητό τηλέφωνο

Το iPhone διαθέτει λειτουργίες σύσκεψης, αναμονής κλήσεων, συγχώνευσης κλήσεων, αναγνώρισης κλήσεως και διασύνδεσης με εφαρμογές άλλων κυψελοειδών δικτύων και άλλες λειτουργίες του iPhone. Παραδείγματος χάριν, εάν κατά τη διάρκεια αναπαραγωγής ενός τραγουδιού δεχθεί μία εξερχόμενη κλήση, το κομμάτι θα σταματήσει σταδιακά. Αφού τερματιστεί η κλήση, η μουσική θα επιστρέψει πάλι σταδιακά.

Το iPhone επίσης περιλαμβάνει τη λειτουργία Οπτικού Φωνητικού Ταχυδρομείου, παράλληλα με την Cingular, η οποία προσφέρει στο χρήστη μία λίστα πρόσφατων φωνητικών μηνυμάτων στην οθόνη, έτσι ώστε να μη χρειάζεται να καλέσει την φωνητική υπηρεσία. Αντίθετα με τις περισσότερες άλλες εταιρείες, τα μηνύματα μπορούν να ακουστούν χωρίς απαραίτητα χρονολογική σειρά, επιλέγοντάς τα από μια λίστα. Η Cingular ανανέωσε πλήρως την υποδομή του φωνητικού της ταχυδρομείου για να υποδεχτεί τη νέα αυτή λειτουργία σχεδιασμένη από την Apple.

Τα ηλεκτρονικά μηνύματα (e-mail) παρουσιάζονται με χρονολογική σειρά στα εισερχόμενα, παρόμοια με την εφαρμογή Mail, η οποία τοποθετεί τα κείμενα όλων των

παραληπτών μαζί με τις απαντήσεις. Τα μηνύματα κειμένου απεικονίζονται σε φούσκες ομιλίας (όπως και στο iChat) κάτω από το όνομα κάθε παραλήπτη.

#### **2.2.4 Αποθηκευτικός χώρος**

Το iPhone αρχικά κυκλοφόρησε με 2 επιλογές όσον αφορά την εσωτερική αποθήκευση. 4GB και 8GB. Το τελευταίο μοντέλο είναι διαθέσιμο με 16 ή 32 GB μνήμης. Όλα τα δεδομένα αποθηκεύονται στον εσωτερικό σκληρό δίσκο μιας και το iPhone δε διαθέτει υποστήριξη επέκτασης του αποθηκευτικού χώρου μέσω memory card ή κάρτας SIM.

#### **2.2.5 Κάμερα**

Το iPhone 4 καθιέρωσε μια κάμερα 5.0 megapixel (2592x1936 pixels), που επίσης βρίσκεται στο πίσω μέρος, η οποία είναι εξοπλισμένη με έναν φωτεινό αισθητήρα ικανό να συλλάβει εικόνες σε συνθήκες χαμηλού φωτισμού, καθώς και ένα LED φλας ικανό να παραμένει αναμμένο για εγγραφή βίντεο σε ανάλυση 720p, που θεωρείται υψηλή ευκρίνεια. Επιπρόσθετα το iPhone 4 έχει και μια δευτέρα κάμερα στην μπροστινή πλευρά ικανή για λήψη φωτογραφιών VGA και εγγραφή βίντεο SD.

#### **2.2.6 Βίντεο**

Το iPhone μπορεί να αναπαράγει βίντεο επιτρέποντας στους χρήστες να παρακολουθήσουν τηλεοπτικά προγράμματα και ταινίες. Σε αντίθεση με κάθε άλλο περιεχόμενο σχετικό με εικόνα, τα βίντεο στο iPhone αναπαράγονται μόνο στον κατακόρυφο προσανατολισμό, με το τηλέφωνο στραμμένο πλάγια. Ο κτύπος δύο δακτύλων ταυτόχρονα εναλλάσσει την προβολή του βίντεο από πλήρη οθόνη σε ευρεία (16:9) και αντίστροφα.

#### **2.2.7 Μπαταρία**

Η Apple κάνει διάφορα τεστ κατά την εξέλιξη και παραγωγή του iPhone για να εξακριβώσει τη διάρκεια της μπαταρίας του. Φυσικά αυτή εξαρτάται από τη χρήση της, καθώς και από κάποιες άλλες παραμέτρους, η επεξήγηση των οποίων ξεφεύγει από το στόχο αυτού του κειμένου. Παρόλα αυτά η εταιρεία έχει βγάλει στη δημοσιότητα κάποια στατιστικά στοιχεία. Στην ιστοσελίδα της αναφέρει ότι η μπαταρία μπορεί να διατηρήσει μέχρι και το 80% της αρχικής της χωρητικότητας μετά από 400 φορτίσεις-αποφορτίσεις. Το πρώτο μοντέλο, μπορούσε να προσφέρει μέχρι και 7 ώρες συνεχούς αναπαραγωγής βίντεο, 8 ώρες ομιλίας, 24 ώρες αναπαραγωγής μουσικής ή 250 ώρες αναμονής πριν κλείσει από μπαταρία, αλλά η αντοχή της μπαταρίας του καινούριου μοντέλου, του iPhone 4 έχει εκτοξευθεί στις 900 ώρες αναμονής.

### 2.2.8 Διαδίκτυο

Το τηλέφωνο διαθέτει ενσωματωμένο WiFi, το οποίο δίνει τη δυνατότητα πρόσβασης στο διαδίκτυο (μέσω ασύρματου δικτύου) με τον φυλλομετρητή (browser) Safari. Το iPhone επίσης είναι δυνατόν να συνδεθεί στο ίντερνετ με το δίκτυο Cingular's EDGE, χωρίς όμως τη δυνατότητα αξιοποίησης του δικτύου τρίτης γενιάς Cingular's 3G/HSDPA. Ο φυλλομετρητής προβάλλει πλήρεις ιστοσελίδες σε αντίθεση με τις απλοποιημένες σελίδες των περισσότερων κινητών. Οι ιστοσελίδες έχουν τη δυνατότητα οριζόντιας και κατακόρυφης προβολής και υποστηρίζουν αυτόματη μεγέθυνση και σμίκρυνση (ζουμ), με κτύπο και διπλό κτύπο αντίστοιχα πάνω σε εικόνες ή κείμενο. Το iPhone επιπλέον ενσωματώνει bluetooth και χρησιμεύει στη σύνδεση με ασύρματα αξεσουάρ με τεχνολογία Bluetooth 2.0.

Μια συμφωνία μεταξύ της Apple και της Google, διασφάλισε πρόσβαση σε μια ειδικά τροποποιημένη έκδοση των Google Maps — σε μορφή χάρτη και δορυφορικής εικόνας, βελτιστοποιημένη για το iPhone. Κατά το λανσάρισμα του προϊόντος έγινε επίδειξη της λειτουργίας αναζήτησης των κοντινότερων καφετεριών και πραγματοποίησης κλήσης με έναν απλό κτύπο.

### 2.2.9 E-mail

Το iPhone επίσης διαθέτει ένα HTML πρόγραμμα ηλεκτρονικού ταχυδρομείου το οποίο επιτρέπει στο χρήστη να ενσωματώνει φωτογραφίες σε ένα e-mail. Η Yahoo θα παρέχει μια δωρεάν Push-IMAP υπηρεσία για αποστολή e-mail παρόμοια με εκείνη ενός BlackBerry. Τα IMAP και POP3 standards ηλεκτρονικού ταχυδρομείου επίσης υποστηρίζονται, συμπεριλαμβανομένου και του Microsoft Exchange. Το πρόγραμμα Outlook για Windows δεν μπορεί να συγχρονιστεί με το iPhone προς το παρόν. Και τέλος δεν υπάρχει ούτε συνδεσιμότητα enterprise email διαφορετική των IMAP και POP3.

### 2.2.10 Επιταχυνσιόμετρο (Accelerometer) – Γυροσκόπιο (Gyroscope)

Το επιταχυνσιόμετρο είναι ένας ανιχνευτής που μετράει την φυσική επιτάχυνση της συσκευής σε ένα πλαίσιο αναφοράς σχετικό με τη βαρύτητα. Μπορεί να μετρήσει την ένταση και την κατεύθυνση της επιτάχυνσης κάτι το οποίο χρησιμοποιείται για να ανιχνεύει τον προσανατολισμό της συσκευής. Το iPhone χρησιμοποιεί το τσιπ LIS331DL της STMicroelectronics το οποίο είναι ικανό να παίρνει είσοδο και στους 3 άξονες.

Το γυροσκόπιο χρησιμοποιείται είτε για να μετράει είτε για να διατηρεί τον προσανατολισμό της συσκευής. Αντίθετα απ το επιταχυνσιόμετρο το οποίο μετράει τη γραμμική επιτάχυνση του κινητού, το γυροσκόπιο μετράει απευθείας τον προσανατολισμό. Οι

περισσότεροι άνθρωποι, όταν αναφέρεται ένα γυροσκόπιο, σκέφτονται ένα μηχανικό γυροσκόπιο, αλλά στην πραγματικότητα το γυροσκόπιο μέσα στο iPhone 4 είναι βασισμένο στην τεχνολογία με μικρο-σένσορες.

Το επιταχυνσίμετρο μπορεί να αναφέρει την κίνηση και τον βαθμό κλίσης της συσκευής, αλλά όχι την κίνηση δεξιά-αριστερά (όταν αυτό παρεκκλίνει). Σε συνδυασμό με το γυροσκόπιο, το iPhone μπορεί να ανιχνεύσει κίνηση σε 6 άξονες : πάνω-κάτω, δεξιά-αριστερά και μπρος-πίσω και αναλόγως να αλλάζει την οθόνη σε κατάσταση πορτρέτου (portrait) ή φόντου (landscape).

Πιο συγκεκριμένα, όταν το iPhone είναι ακίνητο έχει πάντα συνισταμένη σταθερή και δείχνει την επιτάχυνση της βαρύτητας. Το γυροσκόπιο είδαμε ότι μετράει τις επιταχύνσεις γύρω από έναν άξονα, δηλαδή όταν η συσκευή είναι ακίνητη, θα έχει έξοδο 0. Το γυροσκόπιο δίνει τις γωνιακές επιταχύνσεις, οι οποίες με ολοκλήρωση δίνουν τη γωνιακή ταχύτητα, η οποία με ολοκλήρωση δίνει τη γωνία. Επειδή όμως στην ολοκλήρωση πρέπει να υπάρχει και μια σταθερά C, αυτή τη σταθερά τη δίνει το επιταχυνσίμετρο. Οπότε με τη χρήση των δύο αυτών αισθητήρων μπορούμε να βρούμε την ακριβή θέση ενός αντικειμένου, καθώς και τον προσανατολισμό του, εφόσον γνωρίζουμε την αρχική του θέση. Αυτόν τον υπολογισμό τον κάνει ο αλγόριθμος Kalman.<sup>[2]</sup>

### 2.2.11 Λειτουργικό σύστημα iOS 4.0

Το λειτουργικό σύστημα (operating system) iOS 4.0 διατίθεται για συσκευές iPhone 3GS 3ης γενιάς καθώς και σε iPhone 3G γενιάς με περιορισμένες λειτουργίες. Οι νέες λειτουργίες που περιλαμβάνει είναι: η πολυδιεργασία (multitasking), η οποία δίνει τη δυνατότητα χρήσης πολλών εφαρμογών μαζί, η ταξινόμηση εφαρμογών σε φακέλους, η επιλογή διαφορετικών φόντων μεταξύ βασικής οθόνης και οθόνης κλειδώματος και η εμφάνιση των εισερχομένων e-mail όλων των λογαριασμών μαζί. Το καινούριο αυτό λειτουργικό κάνει πολύ καλύτερη χρήση του hardware κάνοντας το όλο πακέτο πολύ πιο γρήγορο.

## 2.3 Jailbreak

### 2.3.1 Τι είναι;

Η διαδικασία του jailbreak δεν είναι τίποτα άλλο από ένα ξεκλείδωμα του συστήματος αρχείων της συσκευής ώστε να μπορεί ο χρήστης να έχει δικαίωμα εγκατάστασης εφαρμογών και βελτιώσεων που δεν είναι διαθέσιμες στο App Store (3rd party apps) ή να πραγματοποιήσει κάποιες άλλες τροποποιήσεις στη συσκευή όπως για παράδειγμα να περάσει ένα theme.

<sup>[2]</sup>[http://en.wikipedia.org/wiki/Kalman\\_filter](http://en.wikipedia.org/wiki/Kalman_filter)



Η Apple όταν δίνει το sdk (πλατφόρμα ανάπτυξης εφαρμογών) στους developers έχει κλειδωμένα κάποια αρχεία μην επιτρέποντας την ανάπτυξη εφαρμογών που παρεμβαίνουν σε σημεία της συσκευής που δεν επιθυμεί. Δεν υπάρχει για παράδειγμα στο App Store εφαρμογή που να παρεμβαίνει στην εφαρμογή του ipod. Το jailbreak λοιπόν έρχεται να απελευθερώσει τη συσκευή από αυτά τα κλειδώματα.

Το Jailbreak δεν είναι απαραίτητο. Όμως τις περισσότερες φορές οι χρήστες δε μένουν ικανοποιημένοι απ τις εφαρμογές που προσφέρει η Apple, και θέλουν περισσότερες λειτουργίες.

Κάποιοι θα πουν ότι το jailbreak γίνεται για να μπορεί κάποιος να περάσει σπασμένες εφαρμογές. Αυτή η αντίληψη είναι εντελώς λανθασμένη. Αν και το jailbreak με το ξεκλείδωμα που κάνει δίνει και αυτή τη δυνατότητα, ο λόγος που γίνεται δεν είναι αυτός αλλά το να γίνει η συσκευή ακόμα καλύτερη και να έρθει στα μέτρα του χρήστη.

Το jailbreak αυτό καθαυτό δεν προκαλεί κάποιο πρόβλημα στη συσκευή. Προβλήματα μπορεί να δημιουργηθούν μόνο με το πέραςμα κάποιων 3rd party εφαρμογών ή να διαπιστωθούν ασυμβατότητες μεταξύ τους. Αν και αυτό συμβαίνει σπάνια, γεγονός παραμένει ότι οποιαδήποτε μη επίσημη εφαρμογή που περνάει μέσα από το Cydia (θα δούμε παρακάτω για αυτό) δεν έχει ελεγχθεί από την Apple, με αποτέλεσμα να υπάρχει μια μικρή πιθανότητα να δημιουργήσει πρόβλημα στη συσκευή και να χρειάζεται Επαναφορά Συστήματος. Όσο όμως οι εφαρμογές που περνάνε από το Cydia έχουν περαστεί απροβλημάτιστα από προηγούμενους χρήστες και δεν γίνονται αχρείαστοι πειραματισμοί δεν θα υπάρξει πρόβλημα.

Το jailbreak καθιστά άκυρη την εγγύηση της συσκευής. Παρόλα αυτά, με το Restore της συσκευής μέσα από το iTunes (πατώντας "set up as new phone") χωρίς να γίνει επαναφορά backup μετά απ αυτό, δε μένει κανένα ίχνος του jailbreak και η συσκευή μπορεί κάλλιστα να σταλεί στο service άφοβα χωρίς τον κίνδυνο να χαθεί η εγγύηση. Εάν όμως γίνει το backup θα επανέλθουν στη συσκευή log files τα οποία θα μπορούν να προδώσουν την προηγούμενη ύπαρξη jailbreak.

### 2.3.2 Τι είναι το Cydia;

Το Cydia υπάρχει εγκατεστημένο στη συσκευή αμέσως μετά το jailbreak (ή σε κάποιες περιπτώσεις, όπως για παράδειγμα μετά από jailbreak με το blackra1n πρέπει περαστεί από το χρήστη μέσα από το εικονίδιο του blackra1n που εμφανίστηκε μετά το jailbreak).

Τι είναι όμως αυτό το περίφημο Cydia; Το Cydia δεν είναι τίποτα άλλο από ένα εναλλακτικό App Store. Εφαρμογές, βελτιώσεις, θέματα καθώς και οτιδήποτε άλλο είναι διαθέσιμο μετά το jailbreak υπάρχει εκεί. Στο Cydia υπάρχουν όλες εκείνες τις εφαρμογές που προσθέτουν και απελευθερώνουν λειτουργίες, δίνοντας πρόσθετες δυνατότητες στο iPhone, themes, ringtones tweaks κλπ.



Figure 6 - Το εικονίδιο που εμφανίζεται μετά από jailbreak

### 2.3.3 Πως γίνεται το jailbreak?

Ένας από τους πιο συνηθισμένους τρόπους με τους οποίους ένας hacker/cracker προσπαθεί να αποκτήσει πρόσβαση σε πληροφορίες / δεδομένα που δεν του επιτρέπεται (π.χ. μια ιστοσελίδα, ένα λειτουργικό σύστημα, μια βάση δεδομένων) είναι τα exploits. Πιο αναλυτικά, η ανάπτυξη ιστοσελίδων, προγραμμάτων και λειτουργικών συστημάτων (όπως το iOS στην περίπτωση μας) αφήνει πάντα κενά ασφαλείας. Το iOS είναι το λειτουργικό σύστημα του iPhone / iPad. Το iOS βασίζεται στο λειτουργικό σύστημα UNIX που σημαίνει ότι ο κώδικας του είναι διαθέσιμος σε μορφή open-source. Οποιοσδήποτε θέλει λοιπόν να ασχοληθεί με τον κώδικα έχει τη δυνατότητα να τον διαβάσει χωρίς κανέναν περιορισμό. Έτσι λοιπόν προγραμματιστές με εμπειρία (comex, musclenerd, planetbeing) που γνωρίζουν καλά την γλώσσα C εντοπίζουν τα κενά ασφαλείας που έχει το iOS και βρίσκουν τρόπους να τα εκμεταλλευτούν. Η εκμετάλλευση των κενών ασφαλείας του iOS πραγματοποιείται με τα exploits.

### 2.3.4 Τι είναι τα exploits?

Το exploit είναι ένα πρόγραμμα –ή ακόμα και μια απλή εντολή– το οποίο εκμεταλλεύεται μια αδυναμία του συστήματος (κενά ασφαλείας) στο οποίο τρέχει και μας δίνει κάποια δικαιώματα που κανονικά δεν θα έπρεπε να πάρουμε. Τα περισσότερα exploits είναι γραμμένα στην γλώσσα C παρόλα αυτά χρησιμοποιούνται και άλλες γλώσσες για την ανάπτυξη

τους όπως η Perl, η assembly κτλ. Αποτελέσματα ενός exploit μπορεί να είναι από το να δούμε τους κωδικούς και τα passwords των χρηστών ενός forum, μέχρι την κατάρρευση ολόκληρου server ή τη διαγραφή όλων των κρίσιμων αρχείων! Στην περίπτωση του Apple iOS τα exploit μας δίνουν δικαιώματα διαχειριστή (root) ξεκλειδώνοντας το file system της συσκευής ώστε να γίνει εφικτή η εγκατάσταση προγραμμάτων και γενικότερη παραμετροποίηση του λειτουργικού συστήματος.

Η μέθοδος του jailbreakme.com (PDF Stack Overflow Exploit) χρησιμοποιεί ένα κενό ασφαλείας στο PDF viewer του Safari μέσω του οποίου μπορεί κάποιος να αποκτήσει πρόσβαση στην συσκευή. Ο comex και η παρέα του έφτιαξαν ένα exploit όπου φορτώνεται ένα αρχείο pdf (ξεχωριστό για κάθε διαφορετική έκδοση καθώς το exploit υπάρχει σε όλες τις εκδόσεις του iOS όχι μόνο στην 4.0.1) όταν ο χρήστης εισέρχεται στον ιστότοπο του jailbreakme.com. Χρησιμοποιώντας ένα κατεστραμμένο είδος γραμματοσειράς μέσα στο pdf αρχείο, το pdf viewer "κρυσάρει" με αποτέλεσμα να επιτρέπει την πρόσβαση στο λειτουργικό σύστημα.

Σημ. Εδώ θα ήθελα να επισημάνω πως η Apple είχε ανακοινώσει πως θα κλείσει τις τρύπες στο iOS 4 με την νέα έκδοση του iOS 4.0.2. Πρακτικά αυτό σήμαινε ότι το buffer overflow στο κομμάτι που αφορά τα PDF αρχεία και το integer overflow που αφορά το IOSurface είχαν πλέον ελεγχθεί και ενημερωθεί με κώδικα έτσι ώστε το jailbreak να αποτελεί όνειρο θερινής νυχτός. Παρόλα αυτά έχει βρεθεί και πάλι τρόπος για να σπάσει το νέο λειτουργικό. Μένει τώρα να μάθουμε την απάντηση της Apple.

### 2.3.5 Σύγκριση iPhone 2G, 3G, 3GS, 4G

Σε γενικές γραμμές η Apple έχει κρατήσει την ίδια σχεδιαστική γραμμή στα 4 μοντέλα που έχει κυκλοφορήσει. Οι μεγαλύτερες αλλαγές έγκεινται στο εσωτερικό τους. Τα πρώτα δύο μοντέλα που βγήκαν στην κυκλοφορία ήταν ολόιδια με το δεύτερο, το iPhone 3g να είναι ελάχιστα πιο παχύ μειώνοντας πάντως το βάρος (μάλλον λόγω του πλαστικού πίσω καλύμματος). Η κύρια διαφορά ήταν ότι πλέον προσφερόταν και σε λευκό. Εσωτερικά οι κύριες λειτουργίες ή καλύτερα οι απουσίες σημαντικών λειτουργιών παρέμειναν. Το 3g δεν είχε, όπως και το προηγούμενο μοντέλο λειτουργία βίντεο στην κάμερα, η οποία συνέχιζε να είναι 2 Megapixel, το ίδιο και η ανάλυση της οθόνης (στα 320×480 pixels με 160 ppi). Επίσης εξακολουθούσε να μην υπάρχει καμία αναφορά για ενσωμάτωση του flash στον web browser όπως επίσης και καμία επιπλέον υποστήριξη codec όπως avi/divx/xvid. Στα συν ήταν ο πιο δυνατός επεξεργαστής και η αύξηση στη χωρητικότητα μνήμης.



Figure 7 - Σύγκριση μεταξύ των μοντέλων του iPhone με χρονολογική σειρά

Η εξέλιξη σε 3GS ήταν πιο ραγδαία, με το S να αναπαριστά τη λέξη 'Speed'. Το νέο iPhone έχει πλέον επεξεργαστή στα 600Mhz από 400 και δυνατότητα μέχρι 32GB μνήμης. Επίσης έχει καλύτερη κάμερα αναβαθμισμένη στα 3Megaapixel, δυνατότητα εγγραφής βίντεο, μεγαλύτερη διάρκεια ζωής μπαταρίας, αισθητήρα κίνησης, υποστήριξη Wi-Fi 802.11b/g και μέθοδο multi-touch εισόδου.

Η μεγάλη διαφορά γίνεται με την εξέλιξη σε 4G. Το τελευταίο (μέχρι τώρα) μοντέλο της Apple έχει πλέον επεξεργαστή στο 1GHz, κάμερα 5Megaapixel, μπροστινή κάμερα με ανάλυση VGA, γυροσκόπιο 3 αξόνων, εγγραφή βίντεο σε HD, ανάλυση 960x640, ακόμα μεγαλύτερη διάρκεια ζωής μπαταρίας, νέο λειτουργικό που κάνει πολύ καλύτερη χρήση του hardware και υποστήριξη Wi-Fi 802.11n εκτός του b/g, ενώ και οι αλλαγές στην αισθητική είναι εμφανείς, με το κινητό να παίρνει μία πιο τετραγωνισμένη μορφή. Θα πρέπει να αναφερθεί επίσης ότι οι διαφορές στην τιμή του κάθε μοντέλου είναι κάτι παραπάνω από μεγάλες, αναλόγως κυρίως την χωρητικότητα μνήμης.

Καινοτομίες όπως το γυροσκόπιο, η μνήμη SSD και οι multi-touch οθόνες που έχει παρουσιάσει η Apple, έχουν κάνει τις υπόλοιπες εταιρείες να "κολλήσουν" και να αναγκαστούν να ενσωματώσουν κι αυτές στα μοντέλα τους τις παραπάνω τεχνολογίες προκειμένου να ανταπεξέλθουν στις απαιτήσεις των αγοραστών.

## Κεφάλαιο 3 - ADOBE FLASH CS5

### 3.1 Ιστορική Αναδρομή Flash

Αρχικά σχεδιασμένο απ τη Macromedia, το Flash εισήχθη στην αγορά το 1996 και πλέον εξελίσσεται και διανέμεται από την Adobe Systems. Ο πρόδρομος του Flash ήταν το SmartSketch, μία εφαρμογή σχεδίασης για υπολογιστές που χρησιμοποιούσαν ηλεκτρονικά στυλό σαν είσοδο. Όταν το λειτουργικό σύστημα που χρησιμοποιούσαν, το PenPoint OS, απέτυχε στην αγορά το SmartSketch έγινε συμβατό με τα Microsoft Windows και το Mac OS. Με το Ίντερνετ να γίνεται όλο και πιο δημοφιλές, το SmartSketch ξανά εκδόθηκε με το όνομα FutureSplash, μία εφαρμογή Web βασισμένη στα διανύσματα και υπό τον έλεγχο της Macromedia Shockwave. Το 1995 το SmartSketch τροποποιήθηκε περαιτέρω με εισαγωγή λειτουργιών "καρέ-καρέ" και επανεκδόθηκε σαν FutureSplash Animator συμβατό με πάρα πολλές πλατφόρμες. Το προϊόν τότε προσφέρθηκε στην Adobe και χρησιμοποιήθηκε από τη Microsoft στις πρώτες δουλειές με το Ίντερνετ (MSN). Το 1996 το FutureSplash αποκτήθηκε από τη Macromedia και εκδόθηκε με την ονομασία Flash, το οποίο και διατηρήθηκε μέχρι τις μέρες μας, με τη διαφορά ότι πλέον ανήκει και πάλι στην Adobe.

### 3.2 Ιστορία Adobe – Apple

Η Adobe έχει μία μακροχρόνια σχέση με την Apple. Η τελευταία ήταν ο πρώτος τους μεγάλος πελάτης, υιοθετώντας την Postscript γλώσσα της Adobe για της καινούριας γενιάς εκτυπωτές laser. Η Apple επένδυσε στην Adobe και είχε για πολλά χρόνια περίπου το 20% των μετοχών της εταιρείας. Οι δύο εταιρείες συνεργάστηκαν στενά για να πρωτοπορήσουν στον τομέα των υπολογιστών γραφείου και υπήρξαν πολλές καλές στιγμές. Από εκείνη τη χρυσή εποχή κι έπειτα οι δύο εταιρείες έχουν πάρει ξεχωριστό δρόμο. Η Apple κόντεψε να εξαφανιστεί πριν μονοπωλήσει την αγορά, αρχικά με τα Mac και στη συνέχεια με τα iPhone, iPod και iPad και η Adobe πέρασε στην εταιρική αγορά με τα προϊόντα του Acrobat. Και οι δύο εταιρείες έχουν επωφεληθεί τα μέγιστα η μία από την άλλη, και κυρίως η Apple, οι χρήστες της οποίας είναι κατά κύριο λόγο σχεδιαστές, και αγοράζουν τα μηχανήματά της κυρίως για χρήση προϊόντων της Adobe όπως το Photoshop.

Σήμερα οι δύο εταιρείες συνεχίζουν να συνεργάζονται για τους κοινούς τους δημιουργικούς πελάτες – Οι χρήστες Mac αγοράζουν περίπου τα μισά προϊόντα της συλλογής της Adobe (CS) – αλλά πέραν τούτου υπάρχουν ελάχιστα κοινά ενδιαφέροντα.

## 3.3 Εισαγωγή στο Flash

### 3.3.1 Τι είναι

Το Adobe Flash (πρώην Macromedia Flash) είναι μία πλατφόρμα πολυμεσικών εφαρμογών, που χρησιμοποιείται για την προσθήκη κίνησης, βίντεο και διαδραστικότητας σε ιστοσελίδες. Το Flash χειρίζεται διανύσματα (vectors) και καμβά με κουκίδες (raster) για να παράγει γραφικά, και να παρέχει κίνηση με κείμενο, σχέδιο και εικόνες. Υποστηρίζει είσοδο απ το ποντίκι, το πληκτρολόγιο, το μικρόφωνο και την κάμερα. Περιέχει επίσης μία αντικειμενοστρεφή γλώσσα προγραμματισμού, την ActionScript.

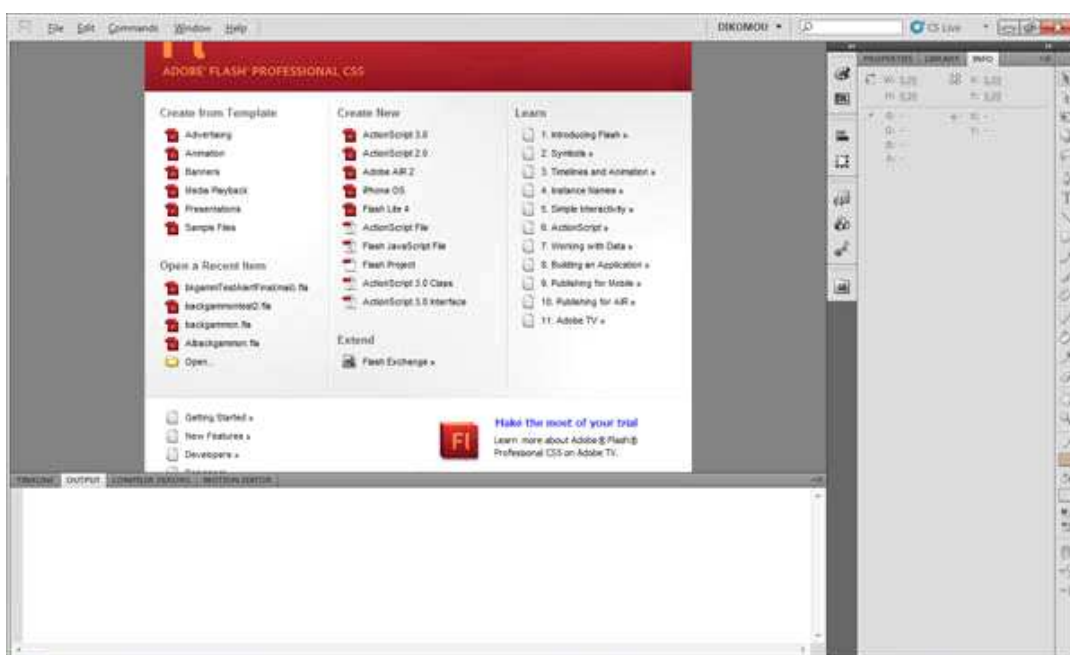


Figure 8 - Η αρχική σελίδα του Flash CS5

Το περιεχόμενο σε Flash μπορεί να προβληθεί σε πολλά διαφορετικά λειτουργικά συστήματα και συσκευές, χρησιμοποιώντας τον Flash Player, ο οποίος διανέμεται δωρεάν από την ιστοσελίδα της Adobe, και εγκαθίσταται σαν plug-in σε όλους τους κοινούς Web browsers.

### 3.3.2 Που χρησιμοποιείται

Το Flash συχνά χρησιμοποιείται ευρέως για την προβολή διαφημίσεων, για παιχνίδια, αλλά και για την προβολή βίντεο στο Διαδίκτυο. Το Youtube που περιέχει περίπου το 40% των συνολικών βίντεο στο Ίντερνετ αλλά και οι περισσότεροι και μεγαλύτεροι ιστότοποι (gmail,

google maps, Vimeo, Facebook κ.α.) χρησιμοποιούν όλοι τον Flash Player. Προσφάτως έχει τοποθετηθεί σαν ένα εργαλείο για δημιουργία "Rich Internet Applications" ("RIAs").

### 3.3.3 Γενιές Flash μέχρι CS5 – Εξέλιξη Actionscript 2.0 - 3.0

Η αρχική σύνταξη της Actionscript 1 με το Flash 3 ήταν απλή και "φλύαρη" και η λειτουργικότητά της ήταν βασισμένη κυρίως γύρω από την εύρεση του εκάστοτε frame και την πλοήγηση της σκηνής σε αυτό αλλά και βασικές αλληλεπιδράσεις με το ποντίκι. Αυτό έμεινε περίπου το ίδιο μέχρι το Flash 5, όταν και η Actionscript υιοθέτησε μία φόρμα πιο κοντά στη Javascript. Με αυτόν τον τρόπο παρείχε μεγαλύτερη λειτουργικότητα και σύνταξη με τελείες (dot-syntax) για την πρόσβαση σε μεταβλητές και μεθόδους. Η Actionscript έγινε επίσης η πρώτη γλώσσα που επέτρεπε απλή αντικειμενοστραφούς προγραμματισμού λειτουργία (OOP), παρόμοια με εκείνη που ήταν διαθέσιμη στη Javascript. Αυτό έγινε με την επόμενη έκδοση του Flash, το Flash MX (6).

Το Flash MX 2004 (7) εισήγαγε την Actionscript 2.0. Υπήρχαν 2 μεγάλες βελτιώσεις σε σχέση με την Actionscript 1: ο τρόπος που δηλώνονταν οι μεταβλητές και μία καινούρια σύνταξη για κλάσεις. Στην πρώτη περίπτωση, η γλώσσα επέτρεπε τη διευκρίνιση του τύπου δεδομένων που θα κρατούσε η κάθε μεταβλητή, αν θα ήταν νούμερο, κείμενο ή κάποιας άλλης μορφής αντικείμενο. Αυτού του είδους η σύνταξη ήταν αυστηρά για compile-time debugging. Αυτό σήμαινε ότι οτιδήποτε συσχετιζόταν με μια μεταβλητή, αυτό θα ελεγχόταν για λάθη μόνο όταν το πρόγραμμα γινόταν compile ή published. Αν δεν υπήρχαν διενέξεις κατά τη διάρκεια του compile, το SWF θα δημιουργούνταν με όλες τις πληροφορίες του κώδικα όσον αφορά τους τύπους των μεταβλητών να έχουν αφαιρεθεί απ' αυτό και η Actionscript θα έτρεχε χωρίς να καταλαβαίνει τι τύπου είναι η κάθε μεταβλητή που χρησιμοποίησε ο προγραμματιστής στον αρχικό κώδικα. Αν και αυτό δεν έδινε κανένα πλεονέκτημα στον Flash Player κατά τη διάρκεια αναπαραγωγής, παρόλα αυτά παρείχε στους προγραμματιστές ένα χρήσιμο εργαλείο για debugging πιο περίπλοκων εφαρμογών.

Η νέα σύνταξη των κλάσεων που χρησιμοποιήθηκε στην Actionscript 2 ήταν παρόμοια με εκείνη που όριζαν γλώσσες όπως η Java. Παρόλο που το Flash ακόμα δεν υποστηρίζει κλάσεις πέρα από εκείνες της μορφής των πρωτότυπων (στο τέλος η Actionscript 2 ακόμα βασικά γινόταν compiled σε Actionscript 1), αυτή η νέα σύνταξη παρείχε ένα οικείο στυλ, χρήσιμο για ανθρώπους που "μετανάστευαν" από άλλες γλώσσες. Παρείχε επίσης έναν πιο "συμπαγή" τρόπο οργάνωσης των ορισμών των κλάσεων μέσα σε μεμονωμένα αρχεία AS και packages. Στην επόμενη έκδοση, στο Flash 8, δεν άλλαξαν πολλά όσον αφορά την Actionscript.

Έτσι φτάνουμε στο Flash CS3 (δηλαδή στο Flash 9) με την Actionscript 3.0. Τώρα έχουμε μία εντελώς καινούρια Actionscript γλώσσα που δεν είχε μόνο καινούριο αριθμό διαφορετικής έκδοσης, αλλά επίσης μία καινούρια εικονική μηχανή (Virtual machine) δηλαδή το θεμελιώδες λειτουργικό μέσα στον Flash Player που εκτελεί την Actionscript κατά τη διάρκεια της αναπαραγωγής. Και η Actionscript 1 και η Actionscript 2 χρησιμοποιούσαν το AVM1 (Actionscript Virtual Machine 1) καθώς και οι δύο σε τελική ανάλυση ήταν ίδιες όσον αφορά την

αναπαραγωγή. Όπως ανέφερα και παραπάνω η Actionscript 2 το μόνο που πρόσθεσε κυρίως ήταν ο έλεγχος λαθών κατά τη διάρκεια του compile και μία καινούρια σύνταξη που στο compile γινόταν Actionscript 1. Η Actionscript 3 τρέχει σε μία εντελώς καινούρια εικονική μηχανή που λέγεται AVM2 (ή και AVM+) που σχεδιάστηκε από το μηδέν, ειδικά για να εκτελεί κώδικα Actionscript 3 και μόνο. Σαν αποτέλεσμα όχι μόνο δε μπορούν οι εφαρμογές σε Actionscript 3 να επικοινωνήσουν με τις δύο προηγούμενες εκδόσεις, αλλά οι αλλαγές στην Actionscript 3 είναι ακόμα πιο εμβριθείς απ ότι έγινε από την μετάβαση από την Actionscript 1 στη 2.

### 3.3.4 Πόσο διαφορετική είναι η ActionScript 3?

Η Actionscript 3 είναι τόσο διαφορετική που στην πραγματικότητα απαιτεί μία εντελώς νέα εικονική μηχανή για να δουλέψει. Αλλά στον πυρήνα της είναι ακόμα Actionscript και σαν Actionscript θα παρατηρήσει κανείς ότι πολλές απ τις εντολές και τις προγραμματιστικές ιδέες που εφαρμόστηκαν στην Actionscript 1 και την Actionscript 2 ακόμα εφαρμόζονται στην Actionscript 3. Πολύ λίγα συντακτικά τουλάχιστον έχουν αλλάξει. Και μερικές φορές ο αντίστοιχος κώδικας της Actionscript 3 είναι παρόμοιος με εκείνον της Actionscript 2, αν όχι εντελώς ίδιος. Παρόλα αυτά υπάρχουν αρκετές αλλαγές μεταξύ τους για να καθιστούν μία "συνομιλία" μεταξύ Actionscript 2 και 3 πάρα πολύ δύσκολη.

Αυτές οι αλλαγές ήταν αναγκαίες. Η Actionscript 3 είναι σχεδιασμένη για αποδοτικότητα και παραγωγικότητα. Και αποδοτική όχι μόνο κατά την ανάπτυξη της εφαρμογής (μιλώντας πάντα για μεγαλύτερα προγράμματα) αλλά και κατά την αναπαραγωγή. Ο κώδικας στην Actionscript 3 μπορεί να τρέξει μέχρι και 10 φορές πιο γρήγορα στον Flash Player, από ότι η Actionscript 2. Εντάξει, είναι σίγουρα πολλές απ τις απλές εφαρμογές που είναι φτιαγμένες σε Actionscript 1 και 2 μπορεί να πάρουν το διπλάσιο χρόνο για να κωδικοποιηθούν σε Actionscript 3, αλλά στο τέλος αυτό που μετράει είναι η έξτρα ταχύτητα και λειτουργικότητα.

### 3.3.5 Πως δουλεύει

Το Flash δημιουργεί διαδραστικές παρουσιάσεις που λέγονται "ταινίες" (movies). Οι ταινίες αυτές εμπεδώνονται σε μία ιστοσελίδα, περίπου όπως και μία εικόνα. Δεν είναι φτιαγμένες σε HTML παρόλο που μία σελίδα HTML χρησιμοποιείται για να τις κρατήσει. Για να μπορεί ο χρήστης να παρακολουθήσει μία τέτοια ταινία πρέπει να έχει εγκατεστημένο ένα έξτρα πρόγραμμα που διανέμεται δωρεάν απ την Adobe, τον Flash Player.

### 3.3.6 Flash Player

Ο Flash Player είναι στην ουσία η εικονική μηχανή (virtual machine) που χρησιμοποιείται για να τρέξει τα flash αρχεία. Τα αρχεία αυτά έχουν κατάληξη SWF και μπορούν να δημιουργηθούν από τα εργαλεία Flash και Flex της Adobe ή από διάφορα άλλα "third party" προγράμματα. Ο Flash Player έχει υποστήριξη για την ενσωματωμένη scripting



γλώσσα που υπάρχει στο Flash, την ActionScript (AS). Αρχικά ήταν σχεδιασμένος για προβολή δισδιάστατων διανυσματικών γραφικών αλλά έκτοτε έχει γίνει κατάλληλος για δημιουργία RIAs "rich Internet applications" αλλά και streaming βίντεο και ήχου. Χρησιμοποιεί διανύσματα για να προβάλει τα γραφικά ελαχιστοποιώντας έτσι το μέγεθος των αρχείων και δημιουργεί αρχεία για εξοικονόμηση bandwidth και χρόνου φόρτωσης "loading". Το Flash είναι ένα συνηθισμένο format για παιχνίδια, animations, και διεπαφές (GUIs) ενσωματωμένες σε ιστοσελίδες.

Ο Flash Player υποστηρίζει τοπική αποθήκευση δεδομένων (persistent local storage), τα οποία μπορούν να χρησιμοποιηθούν παρόμοια με τα cookies του πρωτοκόλλου HTTP. Η τοπική αποθήκευση στον Flash Player επιτρέπει σε ιστοσελίδες να αποθηκεύουν μη εκτελέσιμα δεδομένα στον υπολογιστή όπως πληροφορίες πιστοποίησης, high scores παιχνιδιών, προτιμήσεις ιστοσελίδων, προσωρινά αρχεία (temporary files) κ.α. Ο Flash Player δεν θα επιτρέψει στη συνέχεια πρόσβαση από διαφορετικά domain ιστοσελίδων. Βέβαια η δυνατότητα να κατασκευαστεί ένα προφίλ βασισμένο σε στατιστικά του χρήστη, θεωρείται από πολλούς πιθανός κίνδυνος εκμετάλλευσης του προσωπικού απορρήτου. Ο χρήστης μπορεί να απενεργοποιήσει τη χρήση της τοπικής αποθήκευσης του Flash Player ή να αλλάξει απλά τις ρυθμίσεις μέσω μιας "web-based" σελίδας. Αυτές οι ρυθμίσεις μπορούν να προσπελαστούν απ το site της Adobe ή με δεξί κλικ σε οποιοδήποτε περιεχόμενο φτιαγμένο σε flash κι έπειτα επιλέγοντας "γενικές ρυθμίσεις" (Global Settings). Κάποιοι web browsers όπως οι Chrome, Firefox, Internet Explorer, Safari κ.α., έχουν μία ειδική λειτουργία την "ιδιωτική κατάσταση" (privacy mode), κατά την οποία κανένα δεδομένο δεν αποθηκεύεται τοπικά στον υπολογιστή του χρήστη.

### 3.3.7 Συμβατότητα

Ο Flash Player είναι διαθέσιμος σαν plug-in για τις σύγχρονες εκδόσεις των web browsers (όπως των Mozilla Firefox, Opera, Safari κ.α.). Το plug-in δεν είναι διαθέσιμο πλέον από τον Chrome μιας και η Google ενσωμάτωσε υποστήριξη για Flash. Η Adobe υποστηρίζει ότι κάθε έκδοση του plug-in είναι συμβατή με τις παλαιότερες εκδόσεις με εξαίρεση τις αλλαγές ασφαλείας που εισήχθησαν στην έκδοση 10.

Ο Flash Player είναι συμβατός με τα μεγαλύτερα λειτουργικά συστήματα όπως Windows, Linux, και Mac OS καθώς πλέον και για τις εκδόσεις των 64 bits αλλά και με μικρότερης εμβέλειας λειτουργικά όπως το Solaris και το Symbian OS.

### 3.3.8 Κριτική

Τον Απρίλιο του 2010 ο Γενικός Διευθυντής της Apple εξέδωσε μια ανακοίνωση στην οποία εξηγούσε γιατί η εταιρία του δε θα επέτρεπε το Flash στα iPhone, iPod touch και iPad. Ένας από τους πολλούς λόγους που αναφέρθηκαν ήταν θέματα ασφαλείας. Η Adobe απάντησε καταδεικνύοντας την αναφορά της Symantec, που αφορούσε στις παγκόσμιες Διαδικτυακές απειλές για το 2009 και στην οποία ο Flash Player κατείχε τη δεύτερη θέση σε θέματα ασφαλείας

μεταξύ των plug-ins για browsers αφήνοντας τεχνολογίες όπως η Java και το QuickTime (το οποίο ανήκει στην Apple) με πολλά περισσότερα τρωτά σημεία.

### 3.3.9 Flash και HTML

Για πολλά χρόνια το Flash και η HTML συνυπάρχουν (όχι και τόσο αρμονικά), με το πρώτο στην ουσία να συμπληρώνει το δεύτερο. Η HTML είναι μία πολύ παλιά γλώσσα και σίγουρα δε μπορεί να συναγωνιστεί το Flash σε όρους κινούμενων γραφικών 2 αλλά και 3 διαστάσεων. Θα έλεγε κανείς, ότι ο λόγος που υπάρχει ακόμα το Flash παρόλα τα προβλήματά του, είναι ότι δεν υπάρχει κάποιο άλλο πρόγραμμα αντάξιο του στην αγορά. Τουλάχιστον όχι μέχρι τώρα.

### 3.3.10 HTML5: Τι είναι

Το HTML5 είναι μία υπό ανάπτυξη γλώσσα markup για το Παγκόσμιο Ιστό που όταν ετοιμαστεί θα είναι η επόμενη μεγάλη έκδοση της HTML και πρόκειται να αντικαταστήσει την ήδη υπάρχουσα HTML 4. Έχει μία γλώσσα προγραμματισμού (Javascript), έχει όπως και το Flash μία σειρά από βιβλιοθήκες (APIs)- αξιοσημείωτα επεκτεινόμενες-, έχει τρόπους να σχεδιάζει κανείς γραφικά (SVG, canvas) και ακόμα μπορεί να παίζει βίντεο και ήχο. Αντίθετα με το Flash, οι εφαρμογές για HTML5 επεξεργάζονται απ τον browser και όχι από ένα κατάλληλο plug-in του browser.

Η υλοποίησή της ξεκίνησε τον Ιούνιο του 2004 με το όνομα Web Applications 1.0 και σκοπός της είναι η εξάλειψη της ανάγκης για ιδιόκτητα plug-in αλλά και η δυνατότητα ανάπτυξης πλούσιων διαδικτυακών εφαρμογών (RIAs) κάτι που μέχρι τώρα γινόταν με εργαλεία όπως το Adobe Flash, το Microsoft Silverlight κ.α.

### 3.3.11 Η HTML5 και το CSS3 φέρνουν το τέλος Flash?

Πολλά έχουν ακουστεί για το αν η νέα mark-up γλώσσα, η HTML5, που στην ουσία είναι ένα πακέτο αποτελούμενο από την HTML5, το CSS3 και τη Javascript, είναι ικανά να εξαλείψουν το Flash. Έχουν όμως κάποια βάση;

Παρά την ενθουσιώδη υποδοχή που έχουν επιφυλάξει στο HTML5 εταιρείες όπως η Apple και η Microsoft, η παγκόσμια οργάνωση του Παγκόσμιου Ιστού, το World Wide Web Consortium (W3C) προειδοποιεί ότι ο ανταγωνιστής του Flash δεν είναι ακόμη έτοιμος για ευρεία χρήση. Ο Philippe Le Hégarret, εκ των κορυφαίων στελεχών της Κοινοπραξίας ήταν ξεκάθαρος για το ρόλο που μπορεί να διαδραματίσει αυτή τη στιγμή το HTML5. Συγκεκριμένα αναφέρει ότι είναι εντάξει να πειραματίζεται κανείς με το HTML5 και τις υπάρχουσες υλοποιήσεις, αλλά είναι σχεδόν απίθανο να υπάρχει σταθερότητα, ακυρώνοντας

έτσι την επιχειρηματολογία όσων υποστηρίζουν ότι το HTML5 είναι έτοιμο για ευρεία υιοθέτηση δίχως πάντως να πει συγκεκριμένα γιατί το HTML5 είναι ακόμη ακατάλληλο. Βέβαια μια γνώμη προερχόμενη από το W3C έχει βαρύνουσα σημασία. Αν ο Le Hégarret έχει δίκιο, τότε η Adobe που αναπτύσσει την τεχνολογία Flash έχει κάθε λόγο να πανηγυρίζει. Αντίθετα δεν έχει κανένα λόγο να πανηγυρίζει η Apple του Steve Jobs, που έχει επενδύσει πολλά στο HTML5 ως αντικαταστάτη του "μισητού" Flash.

Υπάρχουν βέβαια και πιο προφανείς λόγοι που το πακέτο HTML5, δηλαδή HTML - CSS3 - JS, δε μπορεί να είναι άμεσος αντικαταστάτης του Flash, τουλάχιστον όχι όπως θέλουν κάποιοι να διαδίδουν. Υπάρχουν εφαρμογές που είναι εξαιρετικά δύσκολο, αν όχι αδύνατο, να κατορθώσει κανείς να τις υλοποιήσει με Javascript και CSS. Για βασικά animations και λειτουργίες που θυμίζουν JQuery, σίγουρα το HTML5 μπορεί να λειτουργήσει καλύτερα αντικαθιστώντας το Flash. Αλλά για πιο περίπλοκα παιχνίδια και εφαρμογές RIA απλά δε γίνεται. Και σίγουρα κανείς δεν είναι διατεθειμένος, όσο κι αν μισεί το Flash, να σπαταλά το διπλάσιο τουλάχιστον χρόνο για κάτι που μπορεί να το τελειώσει πολύ, πολύ πιο γρήγορα. Η Actionscript έχει εξελιχθεί σε μία δυνατή, γεμάτη λειτουργικότητας γλώσσα.

### 3.3.12 Πλεονεκτήματα – Μειονεκτήματα

Η HTML5 δεν είναι δυνατό να αντικαταστήσει το Flash, τουλάχιστον στο άμεσο μέλλον, γιατί υπάρχουν αρκετοί τομείς στους οποίους το τελευταίο υπερτερεί. Καταρχήν ο τομέας του HTML5 είναι καθαρά το web, ενώ το Flash έχει ήδη αναπτυχθεί εκτός του Διαδικτύου, όπως στις συσκευές κινητών τηλεφώνων. Το HTML5 δεν είναι ακόμα έτοιμο ενώ το Flash χρησιμοποιείται ήδη από δεκάδες ίσως και εκατοντάδες χιλιάδες προγραμματιστές που δεν είναι διατεθειμένοι να εγκαταλείψουν τους κόπους τους και να αναπρογραμματίσουν ό,τι έχουν φτιάξει μέχρι τώρα σε μια άλλη γλώσσα.

Επίσης, μετά από επαναλαμβανόμενα τεστ, η Actionscript 3 στους περισσότερους browsers είναι πολύ πιο γρήγορη απ τη JavaScript, ενώ και σε εκείνους στους οποίους υστερεί, η διαφορά είναι πολύ μικρή. Ένα από τα πιο βασικά θέματα είναι ότι ο Flash Player αναβαθμίζεται περιοδικά, για να διευθετήσει τυχόν προβλήματα που μπορεί να υπάρξουν. Πώς όμως θα αναβαθμίσουν/βελτιώσουν την HTML5 αν αυτή γίνει το μοντέλο και το Flash εξαφανιστεί; Δεν θα μπορεί να γίνει κάτι τέτοιο. Θα πρέπει να περιμένουμε για την HTML6 και όλους τους browsers να την υποστηρίξουν. Το οποίο μπορεί να σημαίνει μερικά χρόνια αναμονής. Ήδη αυτή τη στιγμή που μιλάμε, και ενώ οι περισσότεροι browsers υποστηρίζουν την HTML5 δεν την υποστηρίζουν συνολικά. Κανένας γνωστός browser δεν υποστηρίζει όλα τα κομμάτια της HTML5, του CSS3 και των codecs βίντεο και ήχου.

Έρχεται έτσι ένα από τα μεγαλύτερα μειονεκτήματα του Flash ή τουλάχιστον κάτι για το οποίο έχει δεχτεί παρά πολύ κριτική μέχρι τώρα, δηλαδή το ότι είναι κλειστό λειτουργικό ελεγχόμενο μόνο από μία εταιρία, να αποδειχτεί ένα από τα μεγαλύτερα πλεονεκτήματά του. Άρα όσο κι αν τα "ανοιχτά στάνταρ" είναι αξιόπαινα προς το παρόν, ας μείνουμε στο Flash.

Τέλος και σημαντικότερον, ακόμα κι αν η HTML5 αντικαταστήσει τελικά το Flash, η Adobe έχει προβλέψει και γι' αυτό, και στην καινούρια σουίτα της ο χρήστης μπορεί να κάνει export την εφαρμογή του και σε HTML5, το οποίο είναι ένα πολύ σημαντικό εργαλείο.

## 3.4 Εισαγωγή στο Flash CS5

### 3.4.1 Απαιτήσεις Συστήματος

Το νέο Flash CS5 της Adobe είναι πολύ καλύτερο και με περισσότερες προσθήκες από όλες τις προηγούμενες εκδόσεις μέχρι τώρα. Αυτό συνεπάγεται κατανάλωση πολύ περισσότερων πόρων συστήματος. Για να λειτουργήσει λοιπόν το αδηφάγο αυτό πρόγραμμα σύμφωνα με τα επίσημα στοιχεία της εταιρείας, χρειάζεται έναν πολύ ισχυρό επεξεργαστή όχι χειρότερο από Intel Pentium 4, αν και από προσωπική πείρα, επεξεργαστής με 2 πυρήνες είναι το λιγότερο που θα χρειαστεί. Απαιτείται επίσης τουλάχιστον 1 GB μνήμης καθώς και κάρτα γραφικών όχι χειρότερη από μέτριων δυνατοτήτων.

### 3.4.2 Τι καινούριο υπάρχει

Στο Flash CS5 έχουν προστεθεί πάρα πολλά νέα χαρακτηριστικά (features). Ενδεικτικά αναφέρω μερικά:

- Νέα διεπαφή. Οι αλλαγές εδώ δεν είναι ιδιαίτερα μεγάλες, αξίζει όμως να αναφερθεί η διαφορετική εισαγωγική σελίδα με αρκετές καινούριες επιλογές καθώς και η μεγαλύτερη ευελιξία στην αλλαγή του περιβάλλοντος, όπως αυτό αρέσει στον κάθε χρήστη.
- Νέα μηχανή κειμένου (text engine). Το κείμενο μπορεί τώρα να μεταχειριστεί με έναν βελτιωμένο και καλύτερο τρόπο και με εισαγωγή περισσότερων χαρακτηριστικών ελέγχου. Για να αναφέρω μερικά σημαντικά, έχουν εισαχθεί: αναπτυγμένος τρόπος σχεδίασης και επιλογές διάταξης, στήλες κ.α.
- Πίνακας για έτοιμα κομμάτια κώδικα (panel for code snippets). Γρήγορη πρόσβαση σε κοινή Actionscript 3.0 είναι πλέον δυνατή και μπορεί να γίνει τμήμα του κώδικα γρήγορα και εύκολα. Ο χρήστης μπορεί επίσης να δημιουργήσει το δικό του κομμάτι κώδικα και να το αποθηκεύσει σαν snippet. Αυτό όμως δουλεύει μόνο με Actionscript 3.
- Δυνατότητες iPhone (iPhone Capabilities). Ο χρήστης μπορεί πλέον να εξάγει εφαρμογές που θα τρέχουν και σε iPhone.
- Βελτιωμένο Αντικείμενο Αναπαραγωγής Βίντεο FLV (Improved FLVPlayback component). Εξοπλισμένο με νέα όψη και επιλογή για ζωντανή επίδειξη στη σκηνή.

- Σημεία Επεξεργασίας για βίντεο (cue points). Για πρώτη φορά υπάρχει διεπαφή για την επεξεργασία βίντεο και όχι μόνο με κώδικα.
- Νέο εργαλείο σχεδίασης (Deco tool). Στην ουσία αυτό το εργαλείο υπήρχε και στην προηγούμενη έκδοση, αλλά πλέον έχει απίστευτα πολλές νέες δυνατότητες όπως σχεδίαση 3D, σχεδίαση ρεαλιστικής φωτιάς κ.α.
- Πλήρης συμβατότητα του Flash με άλλα προγράμματα της Adobe. Το Flash πλέον είναι απολύτως συμβατό Photoshop, το Illustrator, το Fireworks, το Catalyst και το Flash Builder και μπορεί να υποστηρίξει FXG αρχεία.
- Νέες ιδιότητες για το εργαλείο Bone. Ιδιότητες όπως φυσική ελατηρίου (αναπήδησης) έχουν προστεθεί στο IK bones για να δημιουργήσουν ρεαλιστικά φυσικά εφέ.
- Νέες βιβλιοθήκες για motion tweens (APIs). Εκτός από τις νέες βιβλιοθήκες που έχουν προστεθεί, έχουν υπάρξει βελτιώσεις σε κώδικα για συνηθισμένες (έτοιμες) κλάσεις, καθώς και βελτιώσεις στην κωδικοποίηση και τη διάταξη κώδικα (οι αγκύλες κλείνουν αυτόματα όσο ο χρήστης γράφει κώδικα κ.α.).
- Ενσωμάτωση γραμματοσειρών (embedding fonts). Η ενσωμάτωση γραμματοσειρών είναι πλέον απλή και γρήγορη. Οι γραμματοσειρές μπορούν πλέον να είναι βασισμένες σε fla αντί να είναι βασισμένες σε κείμενο. Ένα φιλικό προς τον χρήστη παράθυρο παρέχει όλες τις λειτουργίες για την ενσωμάτωση γραμματοσειρών.
- Πρόσβαση σε Live υπηρεσίες.
- Πρόσβαση στην κοινότητα βοήθειας της Adobe.
- κ.α.

### 3.4.3 Λειτουργίες που αφαιρέθηκαν

Υπήρξαν βέβαια και πολλά χαρακτηριστικά που δεν συμβαδίζουν με το καινούριο Flash και έπρεπε να αφαιρεθούν.

- Screen-based αρχεία πλέον δε θα μπορούν να ανοίξουν μιας και όλες οι βιβλιοθήκες που έχουν να κάνουν με την οθόνη, έχουν διαγραφεί.
- Η έκδοση Cue (για την επεξεργασία βίντεο) δουλεύει μόνο με ορισμένο σεντ αρχείων.
- Στοιχεία δέσμευσης Δεδομένων (Data Binding Components) της Actionscript 2.0 δεν είναι πλέον διαθέσιμα στον Πίνακα Στοιχείων (Components Panel).
- Μη υποστηριζόμενα εισαγόμενα μορμάτ (Unsupported import formats): Το Flash Professional CS5 δεν θα υποστηρίξει αρχεία FreeHand, PICT, PNTG, SGI και TGA.

- Μη υποστηριζόμενα εξαγόμενα φορμάτ (Unsupported export formats): Το Flash CS5 δεν θα μπορεί πλέον να εξαγάγει EMF και WMF αρχεία, ακολουθίες εικόνων WFM, καθώς και ακολουθίες BMP και TGA.
- Η λειτουργία Αποθήκευση και Συμπίεση (Save and Compact) δεν είναι πλέον αναγκαία λόγω της προεπιλεγμένης εσωτερικής διάταξης XFL των αρχείων FLA. *Συγκεκριμένα από προεπιλογή πλέον, τα αρχεία Flash θα σώζονται σε διάταξη XFL εσωτερικά σαν ένα XML αρχείο, κάτι που βελτιώσει την ανταλλαγή δεδομένων με τα υπόλοιπα προγράμματα της Adobe.*

### 3.4.4 Γνωστά Προβλήματα (Bugs)

Το Flash CS5 έχει και αρκετά προβλήματα (bugs) τα οποία η Adobe προσπαθεί να επιλύσει. Μέχρι αυτή τη στιγμή που γράφεται το κείμενο, τα πιο χτυπητά από αυτά συνοψίζονται παρακάτω.

- Η καινούρια μηχανή κειμένου TLF δεν λειτουργεί πάντα όπως θα έπρεπε. Για την ακρίβεια, πολλές φορές, αν το κείμενο TLF έχει κάποιο motion tween, τότε μπορεί να μην είναι δυνατό να γίνει αποθήκευση του fla αρχείου. Επίσης το undo/redo δε δουλεύει πολύ καλά και δεν υπάρχει ακόμα scroll bar για αυτού του είδους το κείμενο.
- Τα περισσότερα έτοιμα κομμάτια κώδικα βρίσκονται ακόμα υπό ανάπτυξη.
- Η αποθήκευση ενός αρχείου που περιέχει IK (δηλαδή πληροφορίες δημιουργημένες με το Bone tool) μπορεί να μην είναι επιτυχημένη.
- Η αποθήκευση αρχείων με μη-λατινικούς χαρακτήρες ή σε διαδρομή (path) που περιέχει μη λατινικούς χαρακτήρες, μπορεί να προκαλέσει απώλεια του αρχείου.

### 3.4.5 Flash CS5 iPhone Application feature

Για τους περισσότερους χρήστες του Flash, η αλλαγή που ξεχωρίζει στο καινούριο Flash CS5 είναι η δυνατότητα δημιουργίας εφαρμογών για iPhone. Αυτό που επιτρέπει το Flash στον χρήστη, είναι ότι αφού φτιάξει το αρχείο χρησιμοποιώντας και Actionscript, μπορεί να το εκδώσει σαν native εφαρμογή για iPhone, το οποίο σημαίνει ότι αυτό θα έχει κατάληξη .ipa και όχι .swf και θα δουλεύει μόνο στο iPhone. Επίσης θα μπορεί υπό προϋποθέσεις βέβαια, να το ανεβάσει στο App Store, την επίσημη ιστοσελίδα της Apple για το iPhone.

### 3.4.6 Διαμάχη Adobe – Apple

Μέχρι όμως η Apple να δώσει την έγκριση της και η Adobe να εισάγει αυτό το νέο feature στην πλατφόρμα της, μεσολάβησαν πάρα πολλά. Ας πάρουμε τα πράγματα απ' την αρχή.

( σημ. Εδώ θα ήθελα να παραθέσω μια πληροφορία που αφορά μια απόφαση που έλαβε επίσημα η Adobe στη δεκαετία του 90 όπου ανακοίνωσε ότι η επίσημη πλατφόρμα ανάπτυξης της σουίτας της, θα είναι τα Windows, ενώ τα Mac θα είναι σε δεύτερη μοίρα. Προφανώς η Apple ποτέ δεν το ζέχασε αυτό.)



Figure 9 - Η διάσημη πλέον κόντρα μεταξύ Apple και Adobe

Για πολλά χρόνια η Apple αρνείται να δεχτεί υποστήριξη του Flash στις συσκευές της. Μάλιστα τον Απρίλιο 2010 ο Steve Jobs, γενικός διευθυντής της εταιρείας, δημοσιεύει μία ανοιχτή επιστολή με τίτλο "Συλλογισμός γύρω από το Flash", στο οποίο αναφέρει και αναλύει τους λόγους που η εταιρεία του δεν θέλει το iPhone, το iPad και το iPod Touch να το υποστηρίζουν.

Ξεκινώντας την επιχειρηματολογία του, ο Steve Jobs αναφέρει ότι το Flash είναι ένα κλειστό σύστημα στο οποίο η Adobe έχει τον πλήρη έλεγχο κάτι το οποίο είναι αντίθετο με τη πολιτική της Apple η οποία υποστηρίζει τα «ανοιχτά» πρότυπα HTML5, CSS και JavaScript. Συνεχίζοντας, φέρνει στο προσκήνιο τους όρους αξιοπιστία, ασφάλεια και επιδόσεις, τονίζοντας πως το Flash είναι ο κύριος λόγος για τα κρασαρίσματα στους Mac υπολογιστές. Άλλα θέματα που αναφέρει είναι η αυτονομία η οποία μειώνεται στο μισό με τη χρήση του Flash καθώς και το γεγονός ότι χάρη πλέον στο format H.264 το οποίο οι συσκευές της Apple μπορούν να αναπαράγουν, οι χρήστες της εταιρείας μπορούν να περιηγηθούν στο πλήρες web και να δουν ένα μεγάλο ποσοστό των βίντεο που υπάρχουν εκεί. Αλλά ο σημαντικότερος λόγος, σύμφωνα πάντα με τον ίδιο τον διευθυντή της εταιρείας είναι ότι η πολιτική της Adobe είναι να δημιουργεί

εργαλεία για ανάπτυξη εφαρμογών που θα τρέχουν σε διαφορετικές πλατφόρμες με τον ίδιο κώδικα και όχι η δημιουργία καλύτερων εφαρμογών για το iPhone/iPod/iPad.

Με το πέρασμα των μηνών η Apple παρόλα αυτά δείχνει να αλλάζει ρότα και οι δύο κολοσσοί φαίνεται πως συμφωνούν για την ενσωμάτωση στο νέο Flash της δυνατότητας δημιουργίας εφαρμογών για iPhones. Και ενώ όλα δείχνουν να κυλούν ομαλά, 2 μέρες πριν η Adobe παρουσιάσει την καινούρια σουίτα της το CS5, η Apple αλλάζει τους όρους χρήσης που έχει συντάξει η ίδια, με τις προδιαγραφές που θα πρέπει να πληρούν οι εφαρμογές προκειμένου να τεθούν σε κυκλοφορία. Η διαδικασία για να κυκλοφορήσει μια εφαρμογή στο App Store περιλαμβάνει αρκετά αυστηρά κριτήρια και κάθε νέα εφαρμογή υποβάλλεται σε αυστηρό έλεγχο από το αρμόδιο προσωπικό για να λάβει το "OK". Αν όλα πάνε καλά, οι εφαρμογές παίρνουν το δρόμο για το App Store. Αν όχι, τότε η Apple στέλνει πίσω την εφαρμογή, με σημειώσεις για το τι πρέπει να διορθωθεί για να μπορέσει να προκριθεί. Αυτό που άλλαξε λοιπόν η Apple ήταν ο όρος χρήσης 3.3.1 στο SDK 4.0, στον οποίο προστέθηκε ότι οι εφαρμογές θα πρέπει να είναι αρχικά γραμμένες σε Objective-C, C, C++ ή Javascript και να μην κατεβάζουν βιβλιοθήκες μέσω ενδιάμεσων στρωμάτων.

Πολλοί αναλυτές συνειδητοποιούν ότι αυτές οι απαγορεύσεις έχουν σαν κύριο στόχο το Adobe Flash CS5, μαζί με ένα σωρό άλλα μικρότερα προγράμματα και απώτερος στόχος είναι να προστατέψει το ίδιο το App Store, καθότι έχει το μονοπώλιο στην πλέον επιτυχημένη πλατφόρμα ψηφιακής διανομής εφαρμογών, παιχνιδιών και υλικού γενικότερα με πάνω από 10 δισεκατομμύρια downloads παγκοσμίως.

Όλα δείχνουν πως η Apple δεν είναι διατεθειμένη να υποχωρήσει. Όμως ο Steve Jobs έχει ήδη πέσει σε μία παγίδα τον Απρίλιο στην επίσημη παρουσίαση του iPhone OS 4.0. Συγκεκριμένα παρουσιάζει την εφαρμογή Tap Tap Revenge 3, η οποία με τους νέους κανόνες δεν είναι συμβατή, καθώς έχει δημιουργηθεί με non-Objective C γλώσσα (για την ακρίβεια βασίζεται στη scripting γλώσσα LUA). Κανείς δεν ξέρει εάν αυτό το συμβάν έπαιξε καθόλου ρόλο αλλά γεγονός είναι ότι από τις 9 Σεπτεμβρίου 2010, η ιστορία αλλάζει και πάλι. Η Apple ανακοινώνει ότι άρει του περιορισμούς που ίσχυαν για τους third-party προγραμματιστές και το νέο feature ενσωματώνεται με επιτυχία στην νέα πλατφόρμα.

### 3.4.7 Πως δουλεύει

Όταν δημιουργηθεί η εφαρμογή και ο χρήστης την κάνει publish για iPhone, το Flash μεταγλωττίζει την Actionscript σε κώδικα ARM assembly. Το ARM είναι το κύκλωμα (chipset) που υπάρχει στο iPhone, κατασκευασμένο απ τη Samsung και είναι το ντε φάκτο στάνταρ για το πως δουλεύουν τα κινητά. (Παρακάτω υπάρχει λεπτομερέστερη ανάλυση γύρω απ την αρχιτεκτονική ARM)



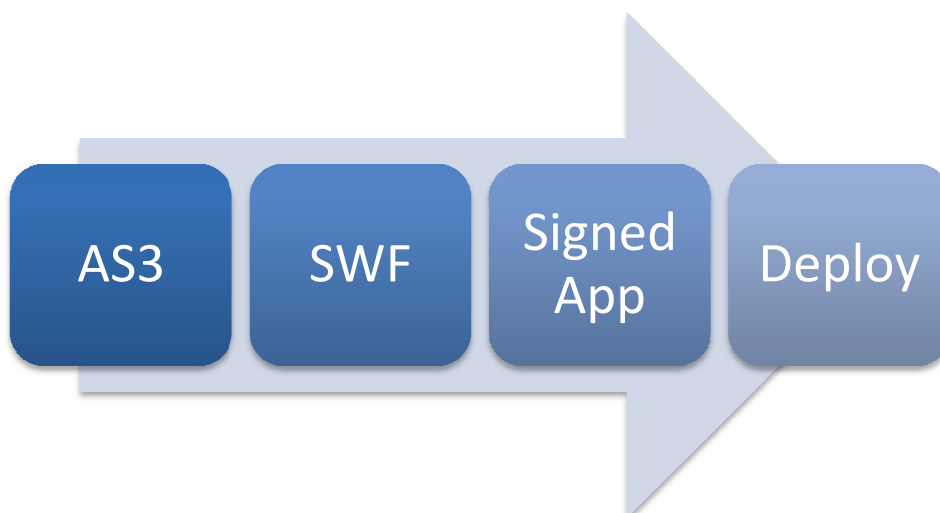


Figure 10 - Στάδια διαδικασίας ανάπτυξης της εφαρμογής.

### 3.4.8 Τεχνική Ανάλυση

Ο χρήστης φτιάχνει κανονικά το αρχείο του, όπως θα έκανε εάν αυτό προοριζόταν για κανονικό υπολογιστή. Γράφει την κώδικα του σε Actionscript, μπορεί να χρησιμοποιεί την timeline ή και να έχει διάφορα σύμβολα η αντικείμενα στη βιβλιοθήκη ή στη σκηνή. Και έπειτα κάνει το τελικό του αρχείο compile σε SWF. Το SWF περιέχει ένα κομμάτι κώδικα που λέγεται ABC (Actionscript Byte Code), οπότε συγκεκριμένα η Actionscript 3 γίνεται compile σε Actionscript Byte Code, και αυτά τα ABC αρχεία απλά "κάθονται" μέσα στο αρχείο SWF. Όταν ο χρήστης κάνει publish το αρχείο, τότε το Flash πηγαίνει μέσα στο SWF, βρίσκει το συγκεκριμένο κομμάτι κώδικα, το πετάει έξω, και το κάνει compile σε ARM assembly χρησιμοποιώντας την υποδομή του LLVM compiler (Low Level Virtual Machine). Το LLVM είναι ένα open source ερευνητικό πρόγραμμα εξαιρετικής τεχνολογίας που χρησιμοποιείται από πάρα πολλούς ανθρώπους και ο λόγος που επιλέχτηκε, είναι η ευελιξία του. Επιτρέπει τη δημιουργία ενός front-end αποτελούμενο από υπομονάδες έτσι ώστε ο χρήστης να μπορεί να πάρει ό,τι είδους είσοδο θέλει. Έπειτα τη δημιουργία ενός back-end αποτελούμενου επίσης από υπομονάδες, οπότε ο χρήστης μπορεί να στοχεύσει σε ARM ή σε XAD6 ή άλλα ενσωματωμένα λειτουργικά συστήματα και στην ουσία τη σύνδεση αυτών των συστατικών μεταξύ τους για να παίξουν. Οπότε μπορούμε να πάρουμε ό,τι υπάρχει στη μία μεριά, σε αυτή την περίπτωση Actionscript Byte Code, και ό,τι υπάρχει στην άλλη μεριά, σε αυτήν την περίπτωση αυτούσια ARM assembly και να δημιουργηθεί έτσι η εφαρμογή. (Στη συνέχεια παρατίθεται λεπτομερέστερη ανάλυση της υποδομής LLVM)

Στη συνέχεια, μόλις ο αυτούσιος κώδικας δημιουργηθεί, πρέπει να γίνει εγγραφή για την εφαρμογή στην Apple, για διανομή μέσω του App Store, το οποίο γίνεται μέσω του iTunes μέσα από έναν μηχανισμό της Apple που θα αναλυθεί εκτενέστερα παρακάτω.

### 3.4.9 Στάδια Ανάπτυξης Εφαρμογής iPhone

Υπάρχουν 4 βήματα, που ακολουθώντας τα θα βοηθήσουν τον χρήστη να δημιουργήσει μία εφαρμογή για iPhone. Δηλαδή ξεκινώντας από το μηδέν, και φτάνοντας στο σημείο να τρέχει την εφαρμογή του στο iPhone.



Figure 11 - Τα στάδια ανάπτυξης μιας εφαρμογής για iPhone.

### 3.4.10 Βήματα Κλειδιά

Αρχικά η **ανάπτυξη (developing)**. Η διαδικασία ανάπτυξης έχει να κάνει με το στήσιμο του όλου αρχείου μέσα από την πλατφόρμα του Flash. Αυτό μπορεί να έχει να κάνει με το στήσιμο των αντικειμένων στη σκηνή, το χειρισμό των αντικειμένων στη βιβλιοθήκη, το χειρισμό της timeline, τον ορισμό των συμβόλων και φυσικά τον κώδικα (Actionscript).

Έπειτα είναι ο **έλεγχος (testing)** για σφάλματα. Ο έλεγχος έχει να κάνει με δύο μέρη.

- Πρώτον εάν η εφαρμογή τρέχει κανονικά χωρίς σφάλματα, όπως ακριβώς θα θέλαμε.
- Δεύτερον έχει να κάνει με το ίδιο το iPhone. Όσο καλό κινητό κι αν είναι, όσο κι αν θυμίζει κανονικό προσωπικό υπολογιστή, παραμένει ένα κινητό με περιορισμένες δυνατότητες -σε σχέση πάντα με έναν ηλεκτρονικό υπολογιστή- τόσο σε υπολογιστική ισχύ, όσο και σε μνήμη. Άρα πρέπει να γίνει καλός

έλεγχος για να σιγουρευτεί κανείς ότι η εφαρμογή δεν καταναλώνει πολύ μνήμη. Το iPhone OS είναι πολύ καλό λειτουργικό σύστημα ενάντια σε κακόβουλες εφαρμογές. Και η λέξη κακόβουλες σημαίνει, ότι εάν μία εφαρμογή αρχίσει και μεγαλώνει όσο τρέχει και χρειάζεται όλο και περισσότερη μνήμη, την οποία το iPhone δε μπορεί να αντέξει, τότε θα κλείσει και θα επιστρέψει στην αρχική οθόνη.

Τρίτο βήμα είναι το **πακετάρισμα (packaging)**. Δηλαδή η μετατροπή της εφαρμογής από flash εφαρμογή .fla σε εφαρμογή για το iPhone, δηλαδή σε native .ipa εφαρμογή, μαζί με όλα τα παρελκόμενα όπως ψηφιακές υπογραφές, ψηφιακά πιστοποιητικά κ.α.).

Τέλος είναι η **εγκατάσταση (deploying)**. Η Εγκατάσταση γίνεται μέσω iTunes. Αφού δημιουργηθεί η εφαρμογή με την κατάληξη .ipa, ο χρήστης μπορεί πολύ απλά με drag-n-drop να τοποθετήσει το αρχείο μέσα στο iTunes, και με το πάτημα ενός κουμπιού το αρχείο αυτό μπαίνει στο iPhone, εάν και εφόσον φυσικά είναι συνδεδεμένο στο pc.

Στο διάγραμμα που ακολουθεί, φαίνεται καλύτερα η διαφορά έκδοσης μίας απλής εφαρμογής για οποιαδήποτε άλλη συσκευή (πχ android), και μίας native εφαρμογής για iPhone, ανεβασμένη στο App Store.

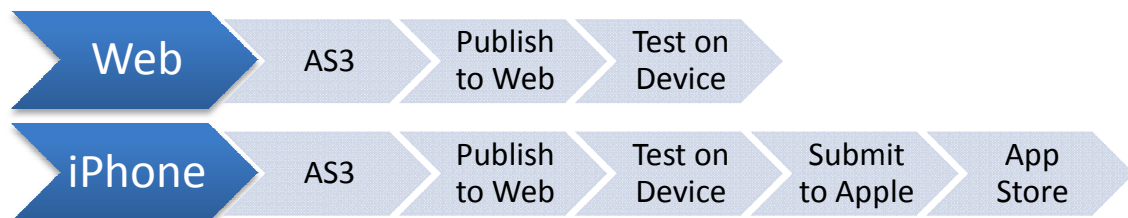


Figure 12 - Οι διαφορές ανάμεσα στα στάδια ανάπτυξης εφαρμογής για Web και iPhone.

### 3.4.11 Πακετάρισμα – Τεχνικές Λεπτομέρειες



Figure 13 - Ανάλυση των σταδίων του πακεταρίσματος (publishing).

### 3.4.12 Η Αρχιτεκτονική ARM

Το ARM είναι ένα 32-bit πρότυπο αρχιτεκτονικής RISC (reduced instruction set computer) ISA (instruction set architecture), που αναπτύχθηκε από τον ARM Holdings. Πιο παλιά ήταν γνωστό σαν Προηγμένη Μηχανή RISC (Advanced RISC MACHINE). Η αρχική σύλληψη ήταν για επεξεργαστές από κανονικούς desktop υπολογιστές, μία αγορά στην οποία τώρα κυριαρχούν οι x86 family της IBM. Η σχετική μη περιπλοκότητα των επεξεργαστών ARM τους έκανε κατάλληλους για χαμηλών απαιτήσεων εφαρμογές. Αυτό τους έκανε κυρίαρχους στην βιομηχανία των κινητών τηλεφώνων, και των ενσωματωμένων ηλεκτρονικών όπως στους μικρού μεγέθους και χαμηλού κόστους μικροεπεξεργαστές (microprocessors) και μικροελεγκτές (microcontrollers).

Η αρχιτεκτονική ARM είναι η πιο διαδεδομένη 32-άμπιτη ISA. Οι περισσότεροι 32-bit επεξεργαστές που πουλήθηκαν μέσα στο 2005 χρησιμοποιούν τη γλώσσα ARM assembly. Από το 2007 και έπειτα περίπου το 98% σε σύνολο άνω του ενός δισεκατομμυρίων κινητών τηλεφώνων που πωλούνται κάθε χρόνο, χρησιμοποιεί τουλάχιστον έναν επεξεργαστή ARM. Από το 2009, οι επεξεργαστές ARM αριθμούν περίπου το 90% όλων των ενσωματωμένων 32-άμπιτων RISC επεξεργαστών.

Παρακάτω παραθέτω ένα απλό παράδειγμα της γλώσσας ARM assembly σε αντιπαραβολή με τη γλώσσα C.

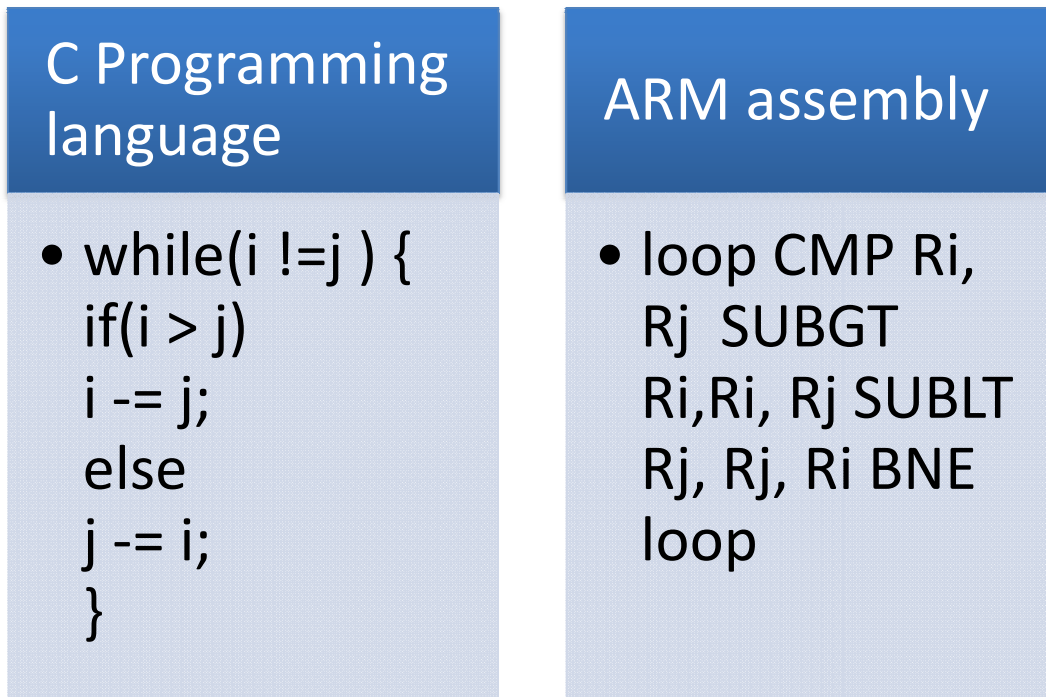


Figure 14 - Παράδειγμα σε γλώσσα C μεταγλωττισμένο σε ARM assembly.

### 3.4.13 Εικονική Μηχανή Χαμηλού Επιπέδου - LLVM (Low Level Virtual Machine)

Η LLVM είναι μία υποδομή μεταγλώττισης γραμμένη σε C++ η οποία είναι σχεδιασμένη για βελτιστοποίηση προγραμμάτων γραμμένων σε αυθαίρετες γλώσσες, σε "compile-time" (δηλαδή τη χρονική περίοδο που μεταγλωττίζονται), "link-time" (δηλαδή τη χρονική περίοδο που συνδέονται μεταξύ τους) και "run-time" (δηλαδή τη χρονική περίοδο που τρέχουν). Αρχικά υλοποιήθηκε για τις γλώσσες C και C++ αλλά από τότε, και λόγω του ανεξαρτητοποιημένου σχεδιασμού της γλώσσας που είναι γραμμένο, έχει γεννήσει μία ποικιλία από "front-ends" συμπεριλαμβανομένων των Actionscript, Objective-C, Fortran, Python και άλλων προγραμματιστικών γλωσσών.

Το όλο εγχείρημα ξεκίνησε το 2000 στο πανεπιστήμιο του Ιλινόις στην Αλαμπάμα υπό τις οδηγίες των Βίκραμ Άντβε (Vikram Adve) και Κρις Λάτνερ (Chris Lattner). Η LLVM αρχικά αναπτύχθηκε σαν υποδομή, μελετώντας δυναμικές τεχνικές μεταγλώττισης για στατικές και δυναμικές γλώσσες. Το 2005 η Apple προσέλαβε τον Λάτνερ να σχηματίσει μία ομάδα για να δουλέψουν πάνω σε διάφορες χρήσεις της για τα συστήματα της εταιρίας με σήμα το δαγκωμένο

μήλο. Η LLVM είναι αναπόσπαστο κομμάτι των πιο σύγχρονων εργαλείων ανάπτυξης της Apple για τα Mac OS και iOS.

#### 3.4.14 Σημαντικές Συμβουλές

- Όταν η εφαρμογή δεν είναι σε full-screen, η “status bar” καταλαμβάνει 20 pixels απ την οθόνη.
- Δεύτερον, το δάχτυλο είναι η συσκευή εισόδου. Μπορούν να χρησιμοποιηθούν mouse events (και touch events) για να εντοπιστεί ο στόχος του χρήστη, αλλά το δάχτυλο είναι μία όχι πολύ ακριβής συσκευή εισόδου. Μερικές φορές το δάχτυλο μπορεί να κινείται στην οθόνη σε περισσότερα από 1 σημεία. Κάποιες συμπεριφορές που συχνά συναντάμε κατά την ανάπτυξη μιας εφαρμογής για υπολογιστές, μπορεί να μην ισχύουν για το iPhone. Επίσης το δάχτυλο καταλαμβάνει στην οθόνη περισσότερα pixels απ’ ότι το ποντίκι (στο iPhone υπολογίζεται ότι είναι περίπου 44 pixels) και άρα κουμπιά και άλλα αντικείμενα στόχων εισόδου (target objects) πρέπει να είναι αρκετά μεγάλα.
- Ρηχές display-lists. Όταν ο χρήστης φτιάχνει την display-list στην Actionscript, είναι εύκολο να προσθέτει συνεχώς αντικείμενα (addChild), καθιστώντας την εξαιρετικά βαθεία και σύνθετη. Όμως θα πρέπει να έχουμε υπόψην μας ότι το iPhone δεν είναι παρά ένα πιο εξελιγμένο κινητό και όχι κανονικός υπολογιστής και άρα η display-list θα πρέπει να διατηρείται όσο πιο ρηχή γίνεται.
- Στο ίδιο μήκος κύματος κυμαίνεται και ο χειρισμός των mouse events. (mouseEnabled, stopImmediatePropagation). Εάν ο χρήστης κάνει κλικ σε ένα αντικείμενο και υπάρχει κάτι βαθεία κάτω σε μια display-list, και θέλει να “πιάσει” το event, το θέμα με το πως δουλεύει η Actionscript και το πως τα events πυροδοτούνται, έχει ως εξής: το Flash στέλνει κάτω το event σε οποιοδήποτε αντικείμενο πρόκειται να το πάρει και έπειτα ξαναστέλνεται πάνω σε οποιοδήποτε άλλο αντικείμενο που το “ακούει”. Άρα ο χρήστης θα πρέπει, μόλις βρει το event που θέλει, να το σταματήσει πριν πάει κάπου αλλού και αρχίζει να πολλαπλασιάζεται. Αυτό σημαίνει, άλλα αντικείμενα που το άκουγαν, δε χρειάζεται να το ακούνε πλέον.
- Άλλο ένα θέμα που ο χρήστης θα πρέπει να λάβει υπόψην, είναι τα fonts. Ανεξάρτητα αν αρέσουν ή όχι, τα Helvetica είναι αυτά που χρησιμοποιούνται κατά κόρον στο iPhone.
- Τελευταίο και ίσως πιο σημαντικό απ’ όλα, αφορά το ίδιο το hardware, και είναι η χρήση της μεθόδου “cacheAsBitmap”. Το cacheAsBitmap είναι μία Boolean συνάρτηση που αν η τιμή της είναι αληθής, τότε ο Flash Player κάνει cache μία εσωτερική αναπαράσταση bitmap του αντικειμένου. Με αυτόν τον τρόπο βελτιώνεται η απόδοση για τα αντικείμενα που περιέχουν σύνθετο διανυσματικό περιεχόμενο. Η ιδιότητα cacheAsBitmap επιτρέπει στο χρήστη να πει στο Flash να πάρει ένα στιγμιότυπο της διανυσματικής εικόνας και να χρησιμοποιήσει αυτό το στιγμιότυπο ως μια επιφάνεια bitmap για το MovieClip. Ο Flash Player μεταφράζει έπειτα το MovieClip σαν να περιείχε ένα bitmap και το αποτέλεσμα θα είναι πιθανώς, ποιο ομαλές κινήσεις.

Από προσωπική εμπειρία, η λειτουργία αυτή δε δουλεύει καθόλου καλά με τα αντικείμενα που παραμένουν σταθερά παρά μόνο με αυτά που κινούνται στο χώρο.

### 3.4.15 Αξιοποίηση Εφαρμογών (App Store)

Το App Store της Apple είναι μία υπηρεσία για τα iPhone, iPod Touch και iPad που προσφέρει η εταιρία και η οποία επιτρέπει στους χρήστες της να κατεβάζουν εφαρμογές και παιχνίδια από το iTunes. Ανάλογα με την εφαρμογή, μπορεί να προσφέρονται είτε δωρεάν είτε έναντι ενός μικρού αντιτίμου εφαρμογές μπορούν να γίνουν download απευθείας από τη συσκευή, ή από έναν υπολογιστή μέσω του iTunes.

Η Apple επιτρέπει το 70% των κερδών της κάθε εφαρμογής που προέρχονται από την ιστοσελίδα, να πηγαίνει κατευθείαν στον κατασκευαστή της εφαρμογής ενώ κρατάει το υπόλοιπο 30% για δικό της συμφέρον. Το App Store άνοιξε για πρώτη φορά στις 10 Ιουλίου 2008. Την επόμενη μέρα η εταιρία λάνσαρε την καινούρια γενιά iPhone 3GS η οποία με το καινούριο λειτουργικό iOS 2.0, είχε υποστήριξη του App Store. Από τότε έχουν "ανέβει" στη σελίδα πάνω από 350.000 εφαρμογές, και έχουν γίνει download πάνω από 10 δισεκατομμύρια φορές.

### 3.4.16 Απόρριψη των εφαρμογών

Οι εφαρμογές είναι υποκείμενες στην έγκριση της Apple, όπως αναφέρεται και στην επίσημη συμφωνία του iPhone SDK, όσον αφορά θεμελιώδη αξιοπιστία κ.α. Μία εφαρμογή που έχει απορριφθεί, μπορεί να διανεμηθεί για συγκεκριμένο σκοπό από τον δημιουργό της εάν αυτός καταθέσει αίτηση στην Apple για άδεια χρήσης για συγκεκριμένες συσκευές, αν και η Apple διατηρεί το δικαίωμα για εκ νέου μελλοντική απόσυρση της. Το περιεχόμενο των ειδοποιήσεων απόρριψης λόγω αρχικής συμφωνίας απαγορεύεται και δεν έχει δει ποτέ το φως της δημοσιότητας. Είναι όμως όρος της συμφωνίας οι εφαρμογές να μην αντιγράφουν τις συναρτήσεις του ίδιου του iPhone ή τη λειτουργικότητα του iTunes. Άλλος λόγος που μπορεί να "κοπεί" μία εφαρμογή είναι σύμφωνα με την Apple "περιορισμένη χρησιμότητα". Κάποιες αναφορές κάνουν λόγο για τοποθέτηση των εργαλειοθηκών (toolbars) μόνο στο κάτω μέρος της οθόνης και η λειτουργία της δόνησης να χρησιμοποιείται μόνο για ειδοποιήσεις.

Φαίνεται λοιπόν ότι η Apple έχει το καρπούζι και το μαχαίρι, κόβοντας και ράβοντας τους όρους στα μέτρα που αυτή θέλει, έχοντας δεχτεί μεγάλη κριτική από τον τύπο αλλά κυρίως από χιλιάδες προγραμματιστές.

### 3.4.17 Πλεονεκτήματα – Μειονεκτήματα Flash CS5 έναντι άλλων εργαλείων παραγωγής εφαρμογών (SDKs) για iPhone

- Ο σημαντικότερος λόγος που το Flash CS5 υπερέχει έναντι άλλων SDK είναι ότι είναι cross-platform, δηλαδή βγαίνει και για Windows. Δεν είναι όλοι υποστηρικτές των

μηχανημάτων Mac, ούτε μπορούν να έχουν όλοι πρόσβαση σε ένα Mac και το Flash λύνει απευθείας αυτό το πρόβλημα.

- Το Flash υπερέχει σε σχέση με τα υπόλοιπα προγράμματα όσον αφορά το γραφικό τμήμα. Η πλατφόρμα παρέχει πάρα πολλά έτοιμα features για αρχάριους χρήστες πολλά περισσότερα από οποιοδήποτε άλλο εξειδικευμένο πρόγραμμα για iPhone.
- Το Flash υπερέχει επίσης όσον αφορά την ενσωματωμένη γλώσσα που έχει. Η Actionscript είναι μία γλώσσα που μοιάζει πολύ με τη Java, και οι περισσότεροι προγραμματιστές έχουν καλύτερες γνώσεις Java απ' ότι Objective-C. Συν το γεγονός ότι η Objective-C είναι μία πολύ παλιά γλώσσα, και άρα στερείται αρκετών feature σε σχέση με τις νεότερες γλώσσες. (Για παράδειγμα είναι πολύ πιθανό ένας νέος προγραμματιστής να πρέπει να μάθει να χειρίζεται δείκτες για το χειρισμό της μνήμης, το οποίο μπορεί να είναι και καλό και κακό).
- Ένα ακόμα χαρακτηριστικό που κάνει το Flash να πλεονεκτεί είναι ότι υποστηρίζεται από μία εταιρία κολοσσό την Adobe. Η Apple έχει τον πλήρη έλεγχο όσον αφορά τις εφαρμογές για το iPhone και κανείς δεν αποκλείει ότι οποιαδήποτε στιγμή θέλει, μπορεί να απαγορεύσει εκ νέου third-party εφαρμογές όπως του Monotouch, ενός απ τα πιο διαδεδομένα προγράμματα για το iPhone σήμερα. Δύσκολα όμως θα απαγορεύσει το Flash.



## Κεφάλαιο 4 – Εισαγωγή στην Actionscript

### 4.1 Hello world!

Είδαμε παραπάνω ότι το Flash έχει μία δική του ενσωματωμένη γλώσσα την Actionscript, με δικές της βιβλιοθήκες/APIs και Compiler. Ας δούμε ένα απλό παράδειγμα. Ανοίγουμε το Flash CS5, δημιουργούμε ένα καινούριο έγγραφο (ctrl+N), και αρκεί να πατήσουμε για συντόμευση το κουμπί "F9", και το Flash ανοίγει ένα παράθυρο όπου μπορούμε να γράψουμε κώδικα. Για να εμφανίσουμε οτιδήποτε αρκεί να γράψουμε την εξής γραμμή:

```
trace("hello world!");
```

Πατώντας τη συντόμευση **alt + Enter**, το Flash, τρέχει τον κώδικά μας, και εμφανίζει ένα παράθυρο στο οποίο υπάρχουν όλα όσα έχουμε βάλει (στην προκειμένη περίπτωση είναι κενό). Βλέπουμε ότι η εντολή αυτή, αν και είναι η αντίστοιχη "printf" της C, στην Actionscript, δεν βάζει τίποτα στο παράθυρο, αλλά το εμφανίζει σαν έξοδο, μόνο στο ίδιο το Flash και συγκεκριμένα στο tab "output" (δίπλα στο tab με τα layers). Αυτό σημαίνει ότι αν κλείσουμε το πρόγραμμα και τρέξουμε μόνο το .swf αρχείο (που είναι το αντίστοιχο .exe), το παράθυρο που θα εμφανιστεί δεν θα εμφανίζει τίποτα.

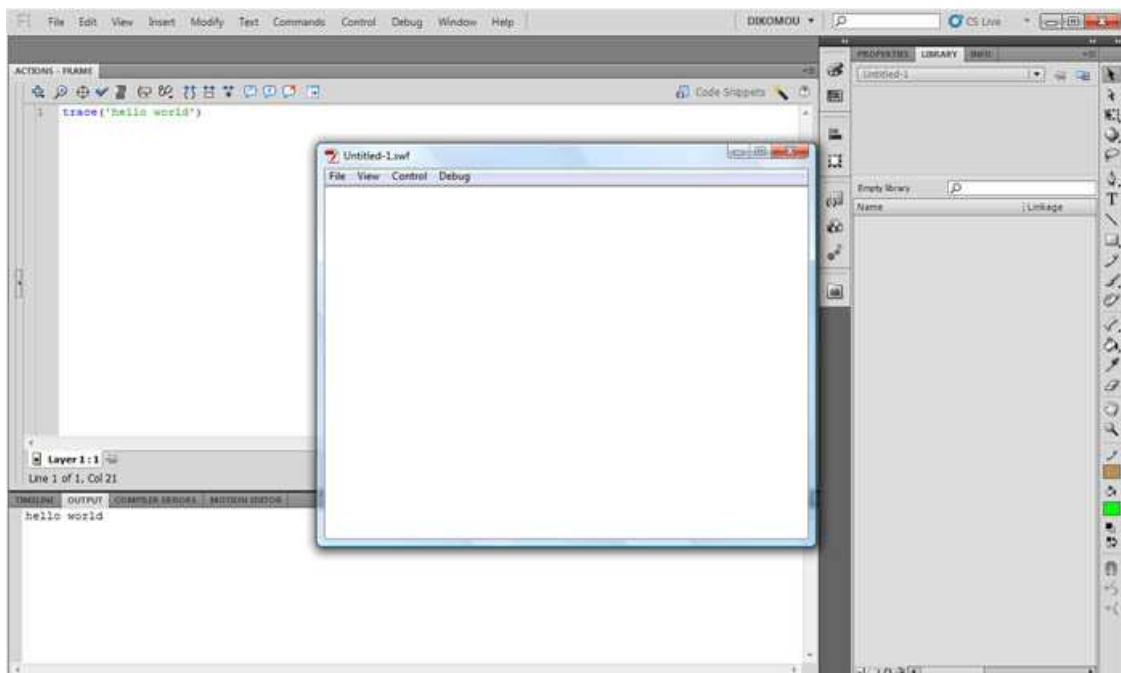


Figure 15 - Το actions panel, το output tab και το παράθυρο που εξάγει το Flash.

Κατά κάποιο τρόπο λοιπόν αυτή η εντολή υπάρχει κυρίως για έλεγχο λαθών, κατά τη δημιουργία κάποιου προγράμματος. Για να εμφανιστεί η πρότασή μας στο swf, υπάρχουν 2 τρόποι. Ο πρώτος να γυρίσουμε πίσω στο πρόγραμμα, πατώντας πάλι τη συντόμευση "F9" να επιλέξουμε απ' τα εργαλεία το κεφαλαίο "T" και να το γράψουμε κατευθείαν στη σκηνή. Έτσι όμως δε χρησιμοποιούμε καθόλου κώδικα, με ότι αυτό συνεπάγεται. Ο δεύτερος τρόπος, είναι αφού γράψουμε στη σκηνή ότι θέλουμε, να το κάνουμε μία κλάση έτσι ώστε κάθε φορά που θα θέλουμε να εμφανίσουμε αυτό το μήνυμα στην οθόνη να δημιουργούμε απλά ένα αντικείμενο της κλάσης αυτής. Η διαδικασία είναι να το επιλέξουμε, να το μετατρέψουμε σε MovieClip (*έχοντας το επιλεγμένο, πατάμε τη συντόμευση F8*) να του προσδώσουμε κάποιο όνομα (*οποιοδήποτε όνομα, στη συγκεκριμένη περίπτωση δεν έχει μεγάλη σημασία αφού δεν το χρησιμοποιούμε πουθενά στον κώδικα*) και στη συνέχεια, αφού δημιουργηθεί στη βιβλιοθήκη μας, δεξί κλικ -> Properties -> Export for Actionscript, και εδώ επιλέγουμε το όνομα της κλάσης, πχ Text. (*Η Actionscript είναι case sensitive, και άρα θέλει προσοχή πως χρησιμοποιούμε τα ονόματα.*) Η συνέχεια είναι πιο απλή. Τώρα μπορούμε να σβήσουμε το αντικείμενό μας απ τη σκηνή, και να το προσθέσουμε με κώδικα. Ανοίγοντας και πάλι το Frame (F9) γράφουμε τα εξής:

```
var myText:Text = new Text();
```

```
addChild(myText);
```

```
myText.x = myText.y = 100;
```

Στην πρώτη σειρά δημιουργούμε ένα αντικείμενο της κλάσης που φτιάξαμε λίγο πιο πάνω με όνομα "**myText**".

Η εντολή "**addChild**" αναλαμβάνει να βάλει το αντικείμενο αυτό στη σκηνή και το κάνει παιδί του αντικειμένου της σκηνής. (*Θα μιλήσω παρακάτω γι αυτήν την λειτουργία, το πιο σημαντικό είναι πως όταν ένα αντικείμενο γίνεται "παιδί" κάποιου άλλου, παίρνει τις ιδιότητες του "πατέρα" του*).

Τέλος μπορούμε να ορίσουμε τη θέση του αντικειμένου ή ένα σωρό άλλες λειτουργίες, όπως το μέγεθος, τη διαφάνεια κ.α.

Βλέπουμε πως σαν πρόγραμμα το Flash δεν είναι το εύρηστο για να το μάθει κανείς, ειδικά αν είναι καινούριος στο χώρο, ωστόσο οι δυνατότητες που προσφέρει είναι τεράστιες.

## Κεφάλαιο 5 – Παράδειγμα Actionscript

### 5.1 Ένα απλό παράδειγμα εφαρμογής iPhone

Ας δούμε τώρα μια πιο πολύπλοκη εφαρμογή η οποία κάνει χρήση ενός χαρακτηριστικού (hardware) που διαθέτει το iPhone, το επιταχυνσιόμετρο (accelerometer). Στο τέλος θα κάνουμε export την εφαρμογή σε .ipa, δηλαδή σε αρχείο που θα παίζει στο iPhone.

Στην αρχική οθόνη του προγράμματος λοιπόν, από τις διάφορες επιλογές που μας δίνει το Flash, επιλέγουμε "iPhone OS". Αυτή είναι μία εκ των καινούριων επιλογών της σουίτας CS5 της Adobe. Πατώντας την επιλογή αυτή, το Flash δημιουργεί ένα καινούριο έγγραφο, το οποίο έχει το ακριβές ύψος και πλάτος που έχει το iPhone, δηλαδή 320 x 480 px. (Αυτό μπορεί να αλλάξει από υπολογιστή σε υπολογιστή ανάλογα με την ανάλυση της οθόνης, σημασία όμως έχει το αρχείο που θα δημιουργηθεί στο τέλος, και αυτό θα είναι πάντα ακριβώς στο μέγεθος της οθόνης του iPhone) . Για το συγκεκριμένο παράδειγμα θα επιλέξουμε μαύρο background, και ρυθμό 30 frames/sec.

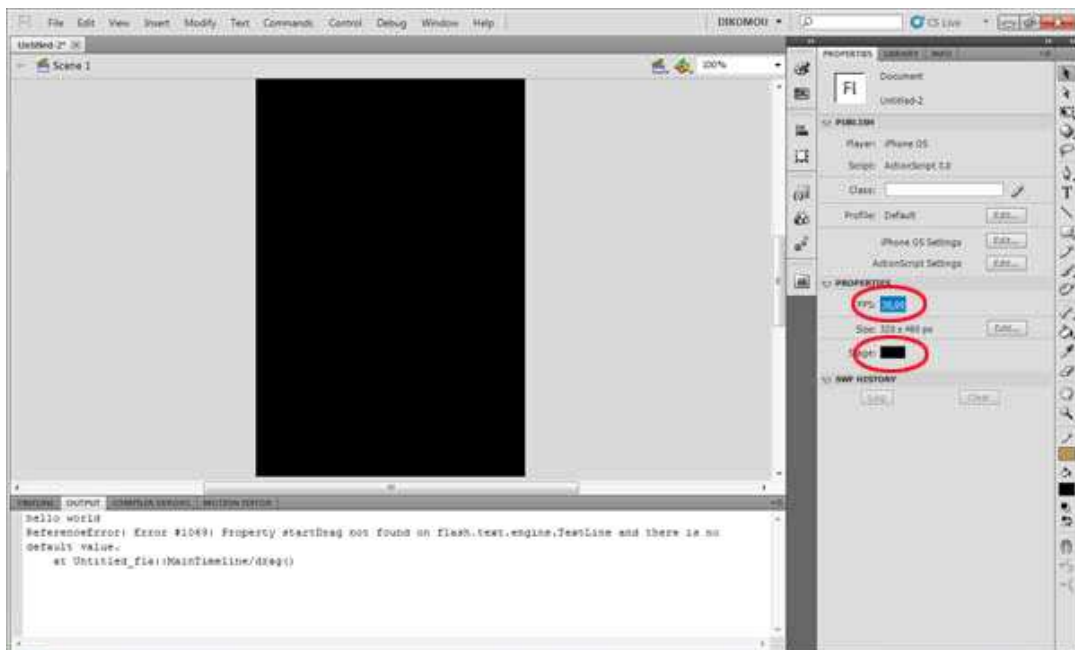


Figure 16 - Επιλογές ρύθμισης των frames/sec και του χρώματος του background πάνω και κάτω αντίστοιχα.

Με το εργαλείο σχεδίασης φτιάχνουμε έναν κύκλο, τον μετατρέπουμε σε movieclip και του δίνουμε ένα όνομα με το οποίο θα μπορούμε να το χειριστούμε μέσω κώδικα. Έστω "ball" το όνομά του. Επίσης τσεκάρουμε και την επιλογή "Cache as Bitmap" έτσι ώστε το πρόγραμμα να

μην χρειάζεται να ζωγραφίζει vectors ξανά και ξανά, εξοικονομώντας έτσι χώρο στη μνήμη. Τέλος, δημιουργούμε 2 layers, το ένα για να έχουμε τη μπάλα ή οτιδήποτε αντικείμενο θελήσουμε να βάλουμε και το δεύτερο για να κρατάμε τον κώδικα. (Γενικά είναι καλό, πάντα, τον κώδικα να τον κρατάμε σε ξεχωριστό layer, και το layer αυτό να το κλειδώνουμε μόλις το δημιουργούμε, έτσι ώστε να μην βάλουμε κάποιο αντικείμενο κατά λάθος). Φτάνουμε λοιπόν έτσι στο παρακάτω σχήμα.

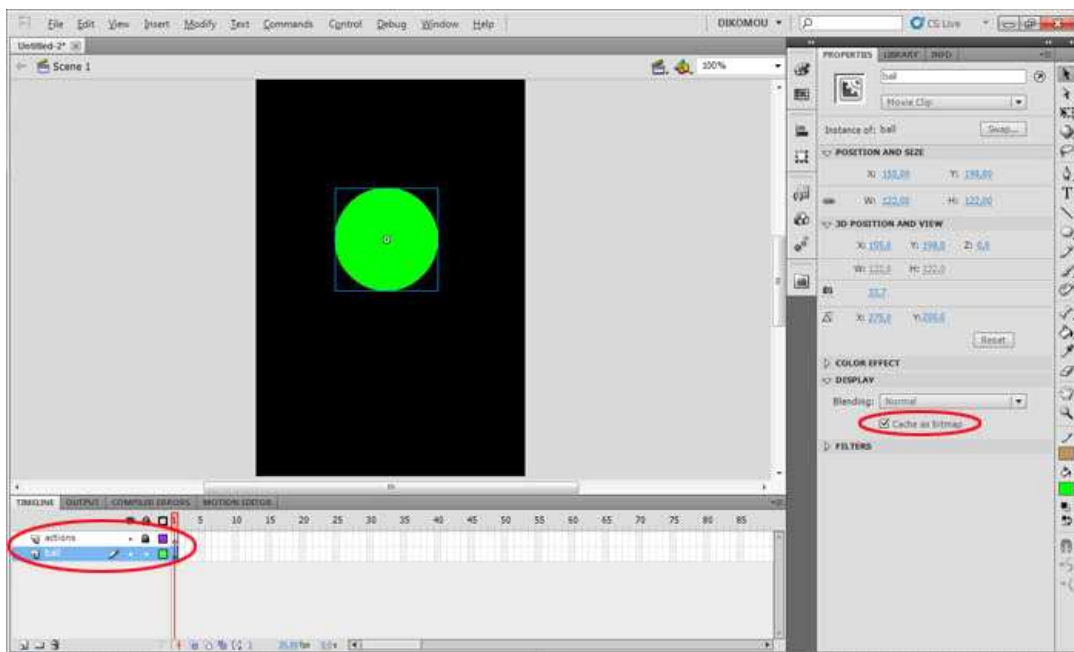


Figure 17 - Στα αριστερά το tab με τα layers, δεξιά η επιλογή cacheAsBitmap.

## 5.2 Ο κώδικας της εφαρμογής

Πάμε τώρα να γράψουμε λίγο κώδικα. Το πρώτο πράγμα που πρέπει να κάνουμε, για να μπορεί η μπάλα να αντιδράει στο επιταχυνσιόμετρο, είναι να φτιάξουμε ένα αντικείμενο της κλάσης του accelerometer. Άρα:

Εισάγουμε το πακέτο που περιέχει το API για το Accelerometer, το οποίο είναι ένα καινούριο πακέτο στο Flash CS5, βασισμένο στον πιο καινούριο flash player (μέχρι τη στιγμή που γράφεται το παρόν κείμενο), τον 10.1.

```
import flash.sensors.Accelerometer;
```

Φτιάχνουμε το αντικείμενο με όνομα "acc"

```
var acc:Accelerometer = new Accelerometer();
```

και το κάνουμε να αντιδράει κάθε φορά που το επιταχυνσιόμετρο κάνει *update*

```
acc.addEventListener(AccelerometerEvent.UPDATE, update);
```

το *Flash* τώρα προσθέτει από μόνο του στην κορυφή, το κατάλληλο πακέτο με τα APIs για τα *events* του *accelerometer*.

```
import flash.events.AccelerometerEvent;
```

Επειδή το *update* του επιταχυνσιόμετρου δε γίνεται αρκετά γρήγορα για το συγκεκριμένο παράδειγμα, θα χρειαστούμε 2 μεταβλητές για να αυξήσουμε λίγο την ένταση.

```
var tx:int = 0;
```

```
var ty:int = 0;
```

Δημιουργούμε τη συνάρτηση *update()*

```
function update(e:AccelerometerEvent):void
```

```
{
```

Παίρνουμε την τιμή επιτάχυνσης του επιταχυνσιόμετρου, και την τοποθετούμε στη μία μεταβλητή, και επειδή όπως προανέφερα η τιμή αυτή είναι πολύ μικρή για να χρησιμοποιηθεί, θα την πολλαπλασιάσουμε με ένα πιο μεγάλο νούμερο. Το ίδιο και για τη δεύτερη τιμή. (Οι τιμές αυτές είναι οι τιμές για τον X και Y άξονα αντίστοιχα. Το επιταχυνσιόμετρο δίνει και τιμή για τον Z άξονα, αλλά στο συγκεκριμένο παράδειγμα δε μας χρειάζεται).

```
tx = e.accelerationX * 100;
```

```
ty = e.accelerationY * 100;
```

```
}
```

Τώρα θα πρέπει να κάνουμε τη μπάλα να κινείται ανάλογα με αυτές τις τιμές.

Στην ουσία δημιουργούμε έναν *Listener* ο οποίος, κάθε φορά που γίνεται *update* το *accelerometer*, εκτελεί τη συνάρτηση *loop*.

```
this.addEventListener(Event.ENTER_FRAME, loop);
```

```
function loop(e:Event):void
```

```
{
```

Η συνάρτηση απλά ανανεώνει κάθε φορά την θέση της μπάλας ανάλογα με τις τιμές που παίρνει από το επιταχυνσιόμετρο.

```
ball.x += ((ball.x + tx) - ball.x) * 0.3;
```

```
ball.y += ((ball.y - ty) - ball.y) * 0.3;
```

```
}
```

### 5.3 Μετατροπή σε ipa

Αυτό ήταν. Φυσικά δεν υπάρχει τρόπος προς το παρόν για να το τεστάρουμε στο pc, οπότε θα πρέπει να το κάνουμε export σε native εφαρμογή για iPhone και να το περάσουμε μέσα στο τηλέφωνο.

Πάμε στις ρυθμίσεις για το iPhone "iPhone OS Settings" και πατάμε "edit".

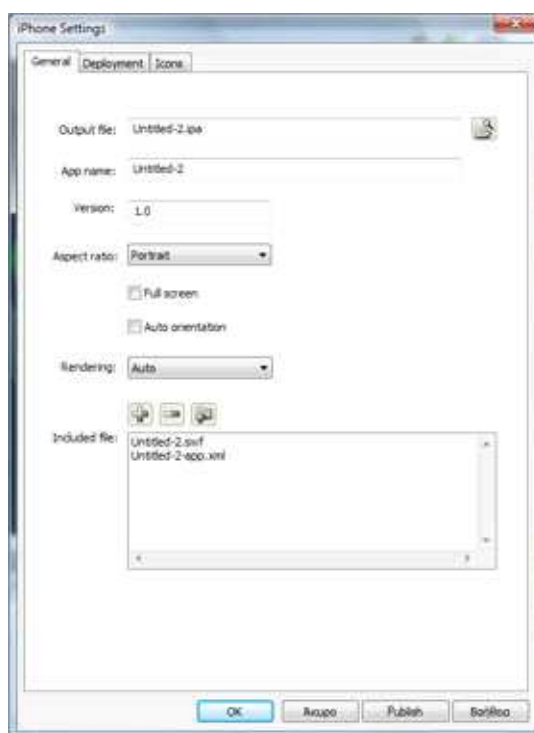


Figure 18 - Η πρώτη καρτέλα της διαδικασίας μετατροπής αρχείου swf σε ipa.

Στην πρώτη καρτέλα επιλέγουμε διάφορα χαρακτηριστικά που θέλουμε να έχει η εφαρμογή μας. Πέρα απ' το όνομα, μπορούμε να ρυθμίσουμε το αν θα εμφανίζεται στο κινητό οριζόντια ή κάθετα, αν θα πιάνει όλη την οθόνη ή θα αφήνει χώρο για τη status bar του iPhone και αν θα αλλάζει οριζόντια ή κάθετα ανάλογα με την κλίση που παίρνει το κινητό.

Στην επόμενη καρτέλα τα πράγματα περιπλέκονται λίγο.



Figure 19 - Η δεύτερη καρτέλα της διαδικασίας μετατροπής αρχείου swf σε ipa.

Για να προχωρήσουμε σε αυτό το βήμα θα πρέπει να είμαστε μέλη του προγράμματος developers της Apple. Για να γίνει αυτό, θα πρέπει να πληρώσουμε 99\$ το χρόνο στην Apple, για να μπορούμε να δημιουργήσουμε εφαρμογές για το iPhone. Για να προχωρήσουμε τη διαδικασία, παράγουμε 2 αρχεία από το site της Apple για να κάνουμε compile το αρχείο. Το ένα είναι το **iPhone certificate**, και το άλλο λέγεται **mobile provisioning profile** το οποίο περιέχει το ID της εφαρμογής. Τέλος θα πρέπει να επιλέξουμε τον τύπο του deployment. Αν θέλουμε να είναι όσο πιο γρήγορα γίνεται για να το τεστάρουμε στο κινητό, τότε διαλέγουμε την πρώτη επιλογή. Αν θέλουμε να κάνουμε debugging, δηλαδή να λάβουμε διάφορα στοιχεία από την εφαρμογή μέσω της εντολής "trace()" που μπορεί να έχουμε βάλει σε διάφορα μέρη του κώδικα, τότε επιλέγουμε τη δεύτερη. Η τρίτη επιλογή είναι για συγκεκριμένο σκοπό, ανάπτυξη της εφαρμογής μέχρι και για 100 διαφορετικές συσκευές, και τέλος και σημαντικότερη η τελευταία επιλογή, αν θέλουμε να ανεβάσουμε την εφαρμογή στο app store.

Τέλος στην τελευταία καρτέλα, θα πρέπει να έχουμε φτιάξει 3 εικονίδια με τα οποία θα εμφανίζεται η εφαρμογή στο κινητό. Τα 3 αυτά εικονίδια είναι διαφορετικού μεγέθους, 29x29, 57x57 και 512x512 px το καθένα.

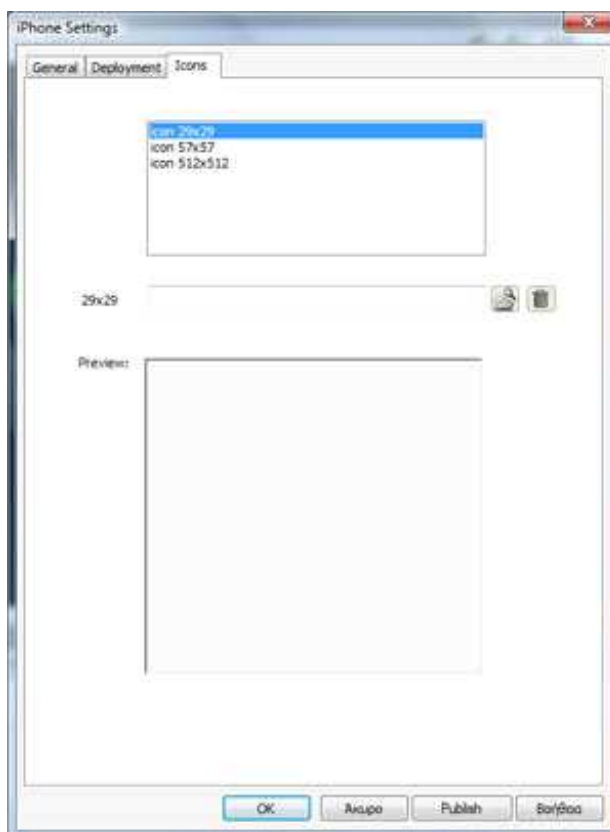


Figure 20 - Η τελευταία καρτέλα της διαδικασίας μετατροπής αρχείου swf σε ipa.

Αυτό ήταν. Πλέον έχουμε δημιουργήσει το αρχείο .ipa. Θα πρέπει εδώ να αναφέρω ότι η διαδικασία αυτή (του compiling), μπορεί να πάρει αρκετή ώρα, από λίγα έως αρκετά λεπτά, ανάλογα με την πολυπλοκότητα του προγράμματος. Επίσης το μέγεθος του εξαγόμενου αρχείου είναι σημαντικά μεγαλύτερο από εκείνο του .swf. Ενδεικτικά να πω ότι στο συγκεκριμένο παράδειγμα, το αρχείο swf ήταν μόλις μερικά kb, ενώ το αρχείο ipa ήταν μερικά MB.

#### 5.4 Δοκιμή της εφαρμογής στο iPhone

Τώρα μένει να βάλουμε το αρχείο στο iPhone και να τρέξουμε την εφαρμογή. Εάν έχουμε windows, τότε θα πρέπει να έχουμε κατεβάσει το πρόγραμμα που δίνει δωρεάν η Apple, το iTunes. Χρησιμοποιείται μεταξύ άλλων για οργάνωση των μουσικών αρχείων και για διάφορα άλλα πράγματα, αλλά είναι απαραίτητο για να περαστεί το .ipa στο iPhone. Εάν το έχουμε εγκαταστημένο, αρκεί να πατήσουμε στο αρχείο διπλό κλικ, το αρχείο τότε περνάει πρώτα στο iTunes, γίνεται "sync" και περνάει στο iPhone.



Για τους χρήστες Mac η διαδικασία είναι πιο απλή, μιας και το iTunes υπάρχει μέσα στο λειτουργικό. Αρκεί ένα drag n drop στο φάκελο "applications", και πατώντας το κουμπί "Sync", η εφαρμογή περνάει στο iPhone.

Αρκεί να την τρέξουμε και θα έχουμε τα παρακάτω αποτελέσματα, δηλαδή ανάλογα με την κλίση και την ταχύτητα κλίσης του κινητού, η μπάλα θα αντιδράει ανάλογα.

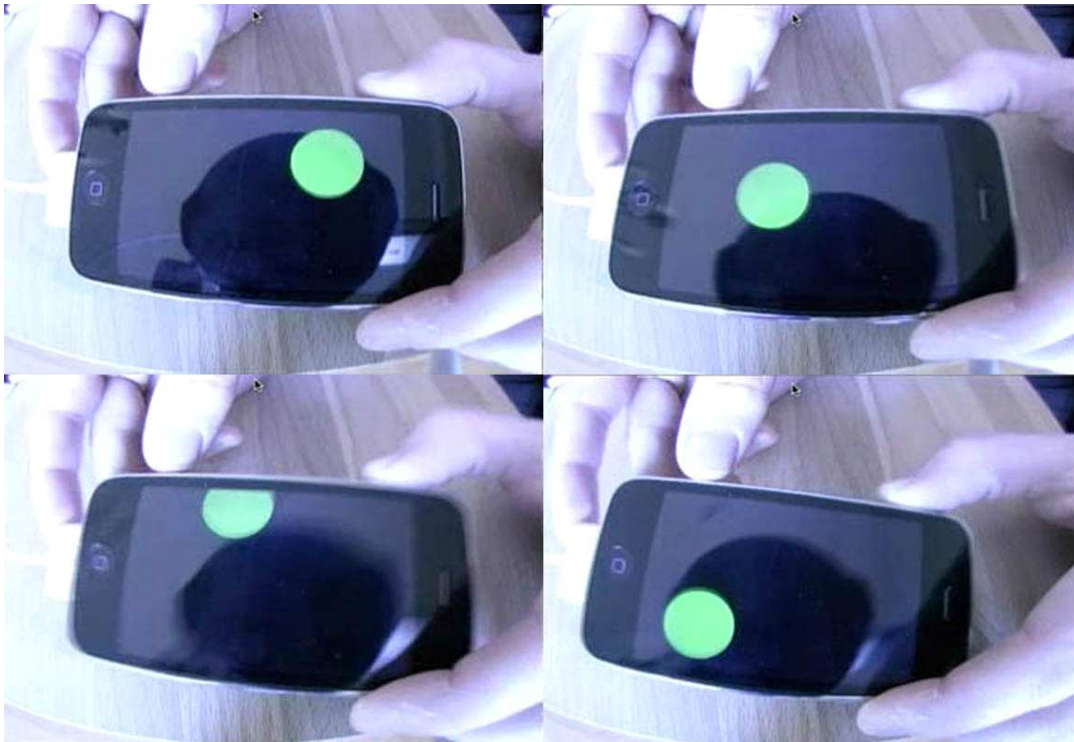


Figure 21 - Τα αποτελέσματα της εφαρμογής.

## Κεφάλαιο 6 – Η κύρια εφαρμογή

### 6.1 Ανάλυση και επεξήγηση του κώδικα

Ας δούμε τώρα το μεγάλο πρόγραμμα. Είναι ένα παιχνίδι για το iPhone, και συγκεκριμένα το γνωστό παιχνίδι στο τάβλι, το "Πλακωτό". Εδώ να αναφερθεί πως το παιχνίδι έχει σχεδιαστεί έτσι ώστε να παίζουν 2 παίκτες, ο ένας απέναντι απ τον άλλον όπως κρατάνε το κινητό.

Στην αρχή δημιουργώ όλα τα αντικείμενα που μου χρειάζονται, όπως τα ζάρια, τα κουμπιά, τα πούλια κλπ. Έτσι η αρχική οθόνη του προγράμματος, για την εφαρμογή μας, έχει αυτή τη μορφή

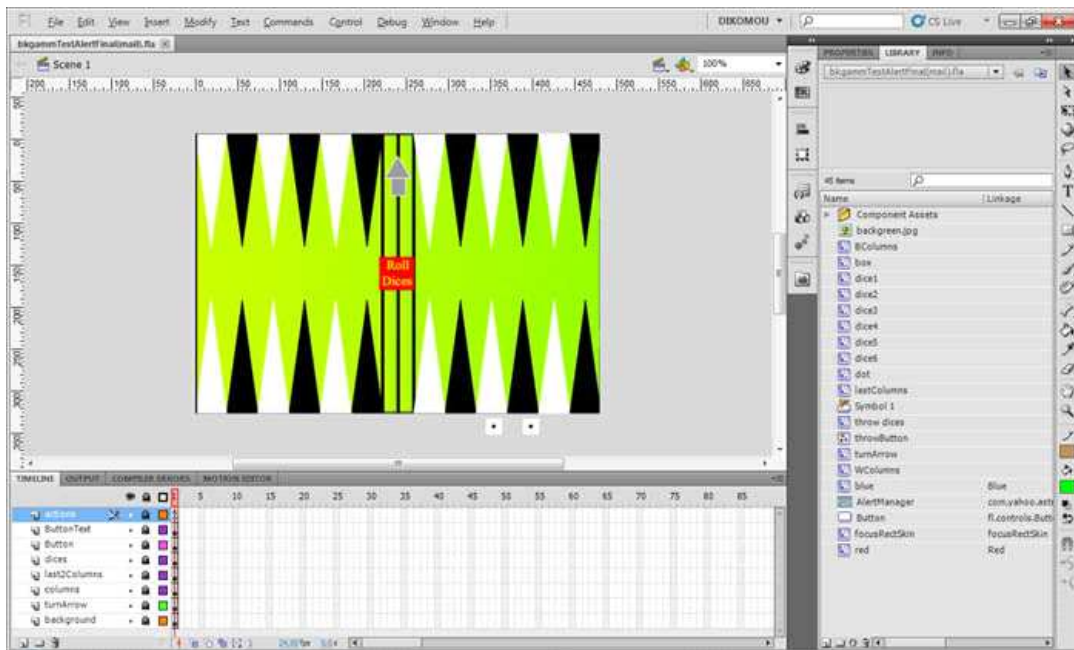


Figure 22 - Έτσι φαίνεται η εφαρμογή μετά τη δημιουργία όλων των αντικειμένων.

Για να επεξηγήσω ενδεικτικά κάποια σημεία, να πω, πως στα δεξιά στη βιβλιοθήκη είναι όλα τα αντικείμενα που έχω φτιάξει και θα χρησιμοποιήσω μετέπειτα. Ανάμεσα σ' αυτά είναι 2 κλάσεις για τα πούλια, (τα οποία δεν τα τοποθετώ εξαρχής, μιας και θα που πάρει περισσότερη ώρα, αλλά τα βάζω στη συνέχεια μέσω κώδικα), ένα movieclip που περιέχει τα ζάρια και ανάλογα με το τι θα φέρουν κάθε φορά, θα εμφανιστεί το αντίστοιχο frame με το κατάλληλο ζάρι, ένα βέλος που θα δείχνει τον παίκτη που είναι η σειρά του να παίξει και 2 κλάσεις για κολόνες, όσες είναι και οι στήλες του background, που χρησιμοποιώ για να μπορώ να χειρίζομαι τα πούλια καλύτερα. Στο κέντρο είναι η σκηνή, εκεί που θα διαδραματιστεί όλο το σκηνικό, και κάτω έχω δημιουργήσει

διάφορα layers στα οποία έχω βάλει τα αντικείμενα που δημιούργησα προηγουμένως. Το πρώτο layer θα περιέχει τον κώδικα και τίποτε άλλο. Πατώντας πάνω σε αυτό το layer, και πατώντας τη συντόμευση F9, εμφανίζεται το παράθυρο που είναι γραμμένος όλος ο κώδικας.

Ας αναλύσουμε τώρα κάποια κομμάτια του κώδικα. Στην αρχή ενσωματώνονται οι κατάλληλες βιβλιοθήκες, και δηλώνονται και αρχικοποιούνται όλες οι μεταβλητές που είναι απαραίτητες για την ομαλή διεξαγωγή του παιχνιδιού. Έπειτα το πρώτο που γίνεται, είναι να μπουν στη θέση τους τα πούλια. Ξεκινώντας με τα μπλε:

*Δημιουργώ έναν πίνακα 15 θέσεων*

```
var blackPieceArray:Array = new Array(15);
```

```
for (var i=0; i<15; i++) {
```

*φτιάχνω 15 αντικείμενα τα οποία τα εκχωρώ στον πίνακα που έφτιαξα παραπάνω*

```
var bpiece:Blue = new Blue();
```

```
blackPieceArray[i] = bpiece;
```

```
blackPieceArray[i].name = "black"
```

*το κάθε πούλι γίνεται cacheAsBitmap για να γλιτώσω χώρο στη μνήμη*

```
blackPieceArray[i].cacheAsBitmap = true;
```

*τα πούλια μπαίνουν στη θέση τους*

```
column1.addChild(blackPieceArray[i]);
```

```
blackPieceArray[i].x = 0;
```

```
blackPieceArray[i].y = topY - i*4;
```

*και τέλος τους προστίθενται 2 listeners, για να μπορούν να γίνονται drag n drop*

```
blackPieceArray[i].addEventListener(MouseEvent.CLICK, drag);
```

```
blackPieceArray[i].addEventListener(MouseEvent.CLICK, drop);
```

```
}
```

*Παρόμοια και για τα κόκκινα πούλια.*

*Σε αυτό το σημείο, όλα τα πούλια είναι τοποθετημένα, αλλά δεν μπορούν να κουνηθούν. Το μόνο που μπορεί να παίξει ο παίκτης, είναι να πατήσει το κουμπί για να ρίξει τα ζάρια.*

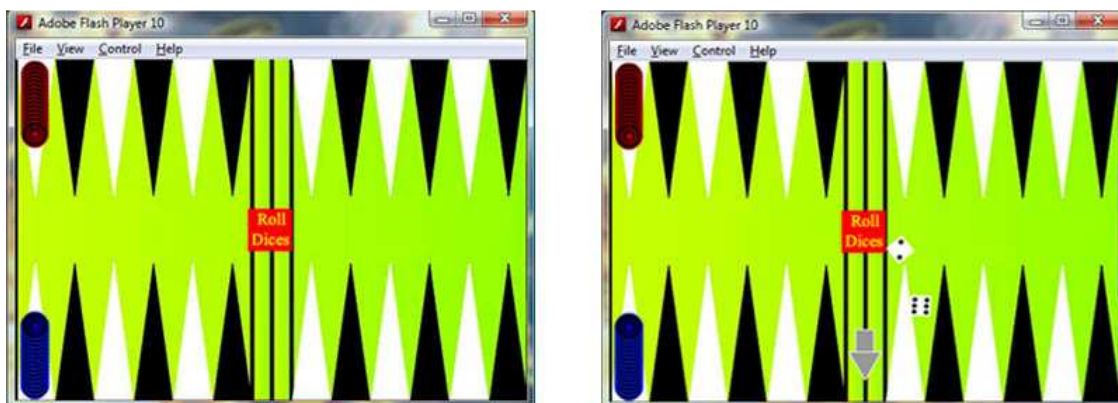


Figure 23 - Μετά το πάτημα του κουμπιού τα ζάρια εμφανίζονται στο τάβλι.

Αφού το κάνει αυτό γίνονται μία σειρά από πράγματα. Υπολογίζονται 6 τυχαίες τιμές. Οι δύο πρώτες είναι τα νούμερα που θα φέρουν τα ζάρια, και είναι μεταξύ 1 και 6,

```
numDice1 = Math.floor(Math.random()*(max-(min-1))+min;
```

```
numDice2 = Math.floor(Math.random()*(max-(min-1))+min;
```

οι επόμενες 2 είναι οι συντεταγμένες του ενός ζαριού, και έπειτα αναλόγως το δεύτερο ζάρι τοποθετείται σε κάποια διπλανή του θέση,

```
var DiceX:int = Math.ceil(Math.random() * 400);
```

```
var DiceY:int = Math.ceil(Math.random() * 200);
```

και τέλος η γωνία που θα έχουν τα ζάρια, ώστε να φαίνεται πιο ρεαλιστικό το αποτέλεσμα.

```
var Dice1Rotation:int = Math.ceil(Math.random() * 50);
```

```
var Dice2Rotation:int = Math.ceil(Math.random() * 50);
```

Στη συνέχεια ελέγχεται ποιανού σειρά είναι να παίξει, και αναλόγως γίνονται τα πούλια του σωστού παίκτη μετακινήσιμα (*draggable*) και τα υπόλοιπα μη μετακινήσιμα (*undraggable*) έτσι ώστε να μην μπορεί ο ένας παίκτης να παίξει τα πούλια του άλλου.

```
if(switchPlayer) {
```

```
    for(i=0;i<15;i++) {
```

```
        if(whitePieceArray[i].parent.name != 'lastColW2') {
```

```
            whitePieceArray[i].addEventListener(MouseEvent.CLICK, drag);
```

```
            whitePieceArray[i].addEventListener(MouseEvent.CLICK, drop); } }
```

```

        blackPieceArray[i].removeEventListener(MouseEvent.MOUSE_DOWN,
drag);

        blackPieceArray[i].removeEventListener(MouseEvent.MOUSE_UP, drop);
    }

    switchPlayer = false; }

```

έπειτα μεταξύ άλλων, τοποθετώ τα ζάρια να φαίνονται πάνω απ τα υπόλοιπα αντικείμενα της σκηνής (των πουλιών δηλαδή) για να φαίνεται καθαρά τι έχουν φέρει.

```
setChildIndex(dice1,numChildren - 1);
```

```
setChildIndex(dice2,numChildren - 1);
```

Τέλος λαμβάνεται υπόψη η περίπτωση τα ζάρια να φέρουν το ίδιο νούμερο, οπότε σε αυτήν την περίπτωση ο παίκτης μπορεί να παίζει 4 φορές.

```

if(posMovArr.length > 2) {
    posMovArr.pop();
    posMovArr.pop(); }

posMovArr[0]=numDice1;
posMovArr[1]=numDice2;

if (numDice1 == numDice2) {
    posMovArr.push(numDice1);
    posMovArr.push(numDice1); }

```

### Οι συναρτήσεις drag() και drop()

```
function drag(event:MouseEvent) {
```

Όταν ένα πούλι γίνεται drag, αρχικά αποθηκεύω τις αρχικές του συντεταγμένες, και την κολόνα στην οποία βρισκόταν έτσι ώστε να μπορεί να επιστρέψει στην αρχική του θέση, σε περίπτωση που ο παίκτης προσπαθήσει να παίζει μία κίνηση που δεν επιτρέπεται.

```

    initPosArray[0]= event.target.x;
    initPosArray[1]= event.target.y;

    initCol = event.target.parent;

```

Στη συνέχεια, ελέγγω το πούλι να μην είναι πλακωμένο

```

if(event.target.name == initCol.getChildAt(1).name) {
    if(initCol.numChildren > 2) {
        if(initCol.getChildAt(2).name.length== initCol.getChildAt(1).name.length) {

```

*εάν πληρούνται όλες οι προδιαγραφές για να κινηθεί το πούλι, τότε αρχικά γίνεται "παιδί" του αντικειμένου της σκηνής (αυτό γίνεται επειδή θέλουμε το συγκεκριμένο πούλι να βρίσκεται πάνω από όλα τα υπόλοιπα αντικείμενα στη σκηνή με την έννοια των layers), και στη συνέχεια γίνεται μετακινήσιμο.*

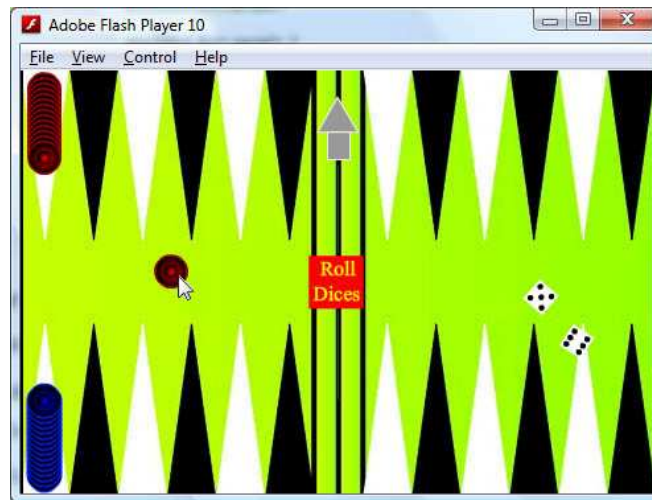


Figure 24 - Το πούλι γίνεται drag.

```

    addChild(event.target as MovieClip);
    event.target.startDrag();
    event.target.x = mouseX;
    event.target.y = mouseY; } }
else {
    addChild(event.target as MovieClip);
    event.target.startDrag();
    event.target.x = mouseX;
    event.target.y = mouseY; } }
else {

```

```
addChild(event.target as MovieClip);  
event.target.startDrag();  
event.target.x = mouseX;  
event.target.y = mouseY; } }
```

Η συνάρτηση **drop()** είναι αρκετά πιο περίπλοκη.

Μόλις ο παίκτης αφήσει το πούλι, αρχικά αυτό σταματάει να ακολουθεί το ποντίκι.

```
e.target.stopDrag();
```

Επειτα ελέγχεται σε ποια στήλη έχει αφήσει ο παίκτης το πούλι.

```
if (e.target.hitTestObject(column1))
```

η παραπάνω γραμμή υπάρχει συνολικά 24 φορές, (else if), όσες δηλαδή είναι και οι στήλες του φόντου. Σε περίπτωση που ο παίκτης δεν έχει αφήσει το πούλι σε κάποια λογική θέση τότε αυτό επιστρέφει εκεί που ήταν στην αρχή. Υπάρχουν άλλες δύο θέσεις (περιπτώσεις) στις οποίες μπορεί ο παίκτης να αφήσει το πούλι, τις οποίες θα δούμε παρακάτω.

Στη συνέχεια γίνεται έλεγχος αν ο παίκτης προσπαθήσει να κλέψει. Αρχικά ελέγχεται αν υπάρχει πόρτα από πούλια του άλλου παίκτη, εκεί που αφέθηκε το πούλι.

```
if(legalMove2(e, column1))
```

Η συνάρτηση είναι σχετικά απλή, δηλαδή αν η στήλη που αφέθηκε το πούλι έχει τουλάχιστον δύο πούλια

```
if(column.numChildren > 2)
```

και ένα απ τα δύο έχει το ίδιο όνομα με το πούλι που αφέθηκε (στην ουσία δηλαδή το ίδιο χρώμα)

```
if(ev.target.name.length==column.getChildAt(column.numChildren-1).name.length)
```

επιστρέφει αληθές

```
return (true)
```

έπειτα ελέγχεται αν η κίνηση του πουλιού συνάβει με τα ζάρια, και αναλόγως, μηδενίζει το αντίστοιχο ζάρι, και το κάνει διάφανο.

```
if (legalMove(1, e))
```

παίρνει σαν ορίσματα την τελική στήλη, και το event, και επιστρέφει αν η κίνηση μπορεί να γίνει ή όχι με έναν boolean.

```
function legalMove(lastCol:int, ev:MouseEvent):Boolean {
```

*μηδενίζεται το αντίστοιχο ζάρι*

```
posMovArr[0] = 0;
```

*και γίνεται διάφανο*

```
dice1.alpha= .5;
```

*σε αρκετές περιπτώσεις χρειάζεται, μέσα στη συνάρτηση `legalmove()`, η χρήση της συνάρτησης `legalmove2()` που είδαμε παραπάνω για την περίπτωση που το πούλι έφτασε στην τελική του θέση, χτυπώντας πάνω σε πόρτες του άλλου παίκτη. (αυτό μπορεί να γίνει επειδή υπάρχει η δυνατότητα ο παίκτης να παίζει είτε ένα ζάρι, είτε το άθροισμα. Το ίδιο και αν έχει φέρει διπλές. Μπορεί να παίζει το άθροισμα 2, 3 ή και όλων των ζαριών.)*

*εάν λοιπόν είχε έστω και μία δίοδο να περάσει από κάποιον, το αποτέλεσμα είναι αληθές.*

```
if(legalMove2(ev, getChildAt(initColNum - posMovArr[0] + 1)) == true || legalMove2(ev, getChildAt(initColNum - posMovArr[1] + 1)) == true)
```

*εάν λοιπόν περάσει και από αυτή τη συνθήκη, τότε το πούλι τοποθετείται στη στήλη και γίνεται παιδί της, υιοθετώντας όλες τις ιδιότητές της.*

```
column1.addChild(e.target as MovieClip);
```

*Στη συνέχεια, σουλουπώνεται η στήλη που αφέθηκε το πούλι. Με αυτόν τον τρόπο ο παίκτης δεν ανησυχεί να αφήσει το πούλι ακριβώς εκεί που πρέπει. Αρκεί το πούλι να ακουμπάει σε οποιοδήποτε σημείο τη στήλη αυτή, και αυτόματα θα μεταφερθεί πάνω απ τα υπόλοιπα.*

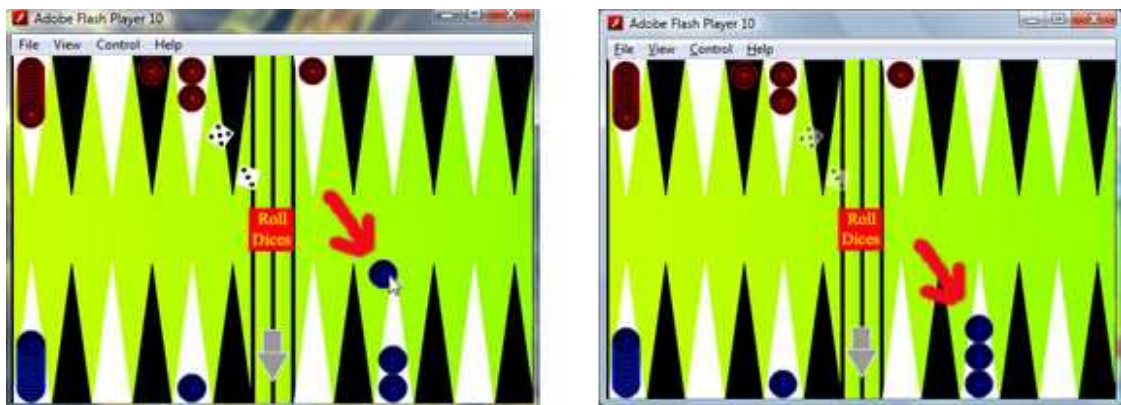


Figure 25 - Μόλις ο παίκτης αφήσει το πούλι σε κάποιο σημείο της κολώνας, αυτό μπαίνει στη θέση του.

```
custLast(column1.numChildren-2, e, false);
```

*έτσι αν η στήλη έχει 2 πούλια*



```
if (numOfChildren == 3)
```

ανάλογα αν η στήλη είναι πάνω ή κάτω

```
if(myColumn)
```

το πούλι θα τοποθετηθεί στην τρίτη θέση πάνω απ τα άλλα 2.

```
e.target.y = -topY + 3*e.target.height;
```

```
else
```

```
e.target.y = topY - 3*e.target.height;
```

τόρα και μόνο για τις αρχικές στήλες, γίνεται έλεγχος για το αν έχει πλακωθεί η "μαμά".

```
if(e.target.name.length > column1.getChildAt(1).name.length)
```

αν ναι, εμφανίζεται το κατάλληλο μήνυμα νίκης (διπλού) για τον κατάλληλο παίκτη, αφαιρούνται όλοι οι listeners μέσω της συνάρτησης `telos()`, και εμφανίζεται ένα κουτί ημιδιάφανο για να σηματοδοτήσει τη λήξη. Το συγκεκριμένο, και επειδή η κατάλληλη βιβλιοθήκη έχει αφαιρεθεί από το CS5, το κατέβασα έτοιμο από το internet, και συγκεκριμένα απ την ιστοσελίδα <http://developer.yahoo.com/flash/astra-flash/alertmanager/>.

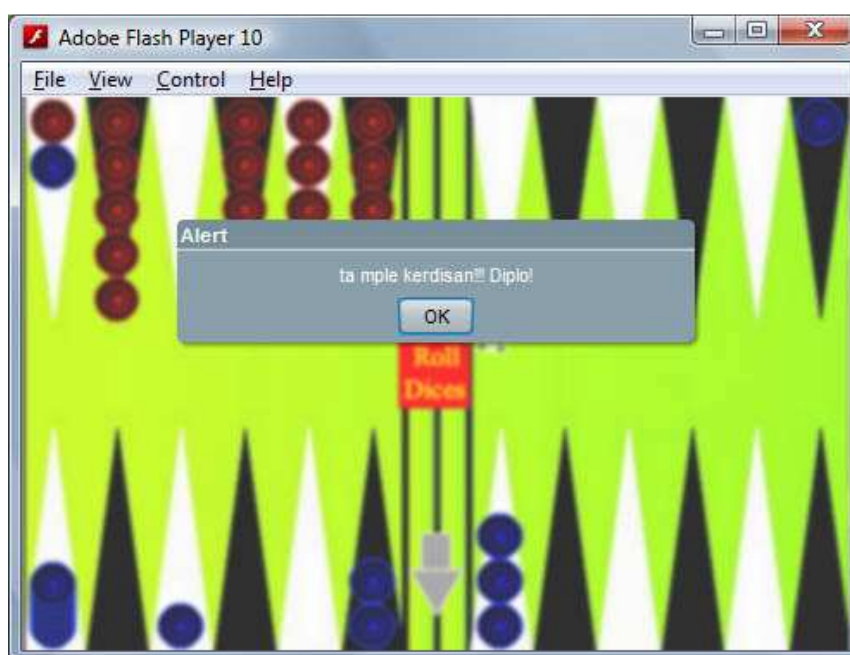


Figure 26 - Το μήνυμα νίκης.

Τέλος μέσω της συνάρτησης `custInit()`, ελέγχεται και σουλουπώνεται η στήλη από την οποία έφυγε το πούλι. Αυτό γίνεται για την περίπτωση που ο παίκτης δεν πήρε το πάνω πάνω στη στήλη, αλλά

ένα τυχαίο από κάτω. Έτσι φαίνεται πιο όμορφο, αλλά κυρίως εξαλείφονται διάφορα bugs που θα υπήρχαν μετά, αν η στήλη έμενε με κενό ενδιαμέσα.

```
custInit(initialCol, custInitBoo);
```

Για να γίνει αυτό, ελέγχεται η απόσταση μεταξύ όλων των συνεχόμενων πουλιών μεταξύ τους, και αν η διαφορά είναι μεγαλύτερη από αυτήν που πρέπει, όλα τα υπόλοιπα πουλία κατεβαίνουν μια θέση.

```
for(var j=0;j<initialCol.numChildren-1;j++)
```

```
    if(initialCol.getChildAt(1).y-initialCol.getChildAt(0).y < topY)
```

```
        for(var i=1;i<initialCol.numChildren;i++)
```

```
            initialCol.getChildAt(i).y += H;
```

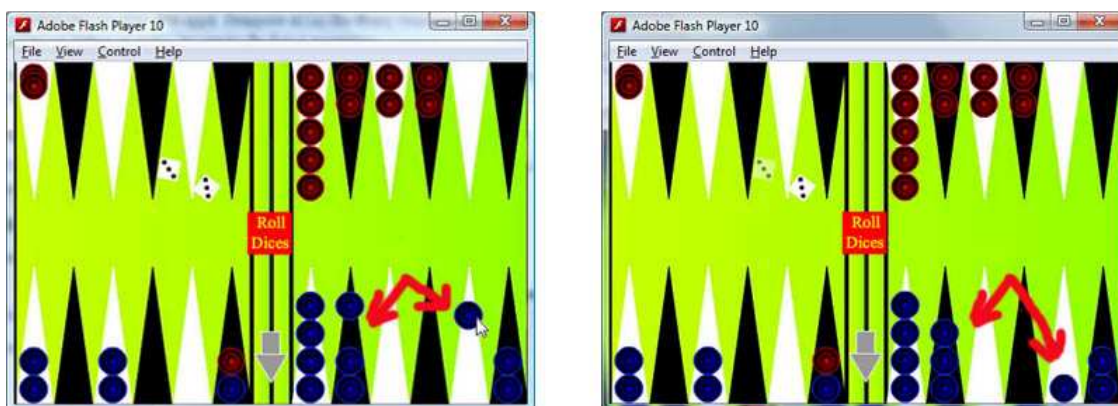


Figure 27 - Η προηγούμενη στήλη που βρίσκεται το πούλι "φτιάζεται" μετά από κάθε κίνηση.

Η διαδικασία διαφέρει λίγο ανάλογα με το αν η στήλη βρίσκεται πάνω ή κάτω, εξού και το δεύτερο όρισμα στην κλήση της.

Εάν σε κάποιο σημείο ο έλεγχος βγει ψευδής, η συνάρτηση *back()* αναλαμβάνει να επιστρέψει το πούλι στην αρχική του θέση και στήλη

```
initCol.addChild(e.target as MovieClip);
```

```
e.target.x = initPosArray[0];
```

```
e.target.y = initPosArray[1];
```

Η συνάρτηση *drop()* για να τελειώσει ελέγχει άλλες 2 περιπτώσεις τις οποίες ανέφερα και παραπάνω. Αυτές είναι οι δύο κολώνες στις οποίες ο παίκτης μπορεί να αφήσει το πούλι όταν

βρίσκεται στη διαδικασία του μαζέματος και βρίσκονται πάνω και κάτω από το κουμπί "roll dices" για τα μπλε και τα κόκκινα πούλια αντίστοιχα.



Figure 28 - Ο μόνος παίκτης που μπορεί αρχίζει να μαζεύει τα πούλια του.

Ο κώδικας έχει ως εξής: Για την πρώτη στήλη

```
if(e.target.hitTestObject(lastColW2))
```

Η συνάρτηση αρχικά ελέγχει αν το πούλι είναι του σωστού χρώματος, έτσι ώστε να μην μπει μπλε στη στήλη για τα κόκκινα και το αντίθετο

```
if(e.target.name.length > 9)
```

Στη συνέχεια ελέγχει αν όλα τα υπόλοιπα πούλια του ίδιου χρώματος, είναι στην περιοχή μαζέματος

```
for(i=0;i<15;i++)
```

```
if(parseInt(whitePieceArray[i].parent.name.substr(6)) < 7)
```

αν αυτό είναι αληθές, τότε η μεταβλητή `checkAreaWBoo` γίνεται αληθής και ο έλεγχος συνεχίζει στο επόμενο στάδιο όπου τρέχει η συνάρτηση

```
function lastColFunc(column:Object, e:MouseEvent,  
j:int,ifBoo:Boolean,zariBoo:Boolean):void
```

(για την περίπτωση που ο παίκτης δε χρειάζεται να σπάσει), η οποία αναλαμβάνει να βάλει το πούλι στη στήλη αυτή, έπειτα ελέγχει αν αυτό ήταν το τελευταίο πούλι, οπότε εμφανίζει μήνυμα νίκης,

```
if(column.numChildren>15)
```

*στη συνέχεια μηδενίζει το εκάστοτε ζάρι το οποίο το κάνει και διάφανο και τέλος αφαιρεί τους listeners από το κάθε πούλι που βάζει στη στήλη αυτή, για να μην μπορεί ο παίκτης να το κουνήσει.*

*Τέλος στην περίπτωση που ο παίκτης οφείλει να σπάσει, αναλαμβάνει η συνάρτηση sprasimo().*

*Η συνάρτηση αυτή ψάχνει όλες τις στήλες από εκείνη που βρίσκεται το πούλι μέχρι τη μεγαλύτερη σε αριθμό στήλη που υπάρχει μέσα στην περιοχή μαζέματος, και αν ο παίκτης πρέπει να σπάσει*

```
for(i=initCol.id+1;i<7;i++)
```

```
if(DisplayObjectContainer(getChildAt(i+1)).numChildren>1) {
```

```
if(DisplayObjectContainer(getChildAt(i+1)).getChildAt(1).name == 'whitepiece') {
```

```
returnBoo=false;
```

```
break; } }
```

*επιστρέφει ανάλογα αληθές ή ψευδές*

```
return (returnBoo)
```

## 6.2 Προβλήματα που προέκυψαν

Κατά τη δημιουργία της εφαρμογής, προέκυψαν διάφορα προβλήματα, τα οποία είτε δεν είχα φανταστεί ότι θα εμφανιστούν, είτε ήμουν σίγουρος ότι ο τρόπος που είχα επιλέξει ήταν ο σωστός. Το πιο σημαντικό ήταν ίσως το πρόβλημα με τα layers. Και εξηγούμαι. Όπως είδαμε και παραπάνω, ο τρόπος που επέλεξα για να "κάθονται" τα πούλια ακριβώς στις στήλες του φόντου, ήταν να φτιάξω κολόνες, οι οποίες ήταν ίδιο χρώμα με το background, δηλαδή άορατες στον κοινό χρήστη, και κάθε φορά να έχω αυτές σαν σημείο αναφοράς. Δηλαδή κάθε φορά, να ελέγχω αν το πούλι εφάπτεται με κάποια απ τις στήλες αυτές. Το πρόβλημα που προέκυψε, ήταν ότι δημιουργώντας τις κλάσεις για να κάνω το πρόγραμμα πιο ευέλικτο και πιο εύκολο για κάποιον να το μελετήσει, ήταν αδύνατο να κάνω το εκάστοτε πούλι που γινόταν drag, να μπαίνει στην κορυφή των layers. Έτσι, δε δούλεψε σωστά η εντολή που χρησιμοποίησα κατά κόρον, η hitTestObject() και δεν μπορούσα να συνεχίσω. Το πρόβλημα λύθηκε βάζοντας όλον τον κώδικα στο ίδιο αρχείο. Έπειτα το μόνο που είχα να κάνω ήταν να βάλω την εντολή addChild() στην συνάρτηση drag(). Η εντολή αυτή, εκτός των άλλων, βάζει το αντικείμενο που παίρνει σαν είσοδο, πάνω πάνω, σε επίπεδο layers.

Ένα άλλο σημαντικό πρόβλημα που μου πήρε αρκετό χρόνο μέχρι να το βρω, ήταν μία πολύ απλή -αφού τη βρήκα- προσθήκη. Συγκεκριμένα, η εντολή αντικείμενο1.hitTestObject(αντικείμενο2) ελέγχει αν το αντικείμενο2 καλύπτει σε οποιοδήποτε σημείο το αντικείμενο 1, και αναλόγως επιστρέφει αληθές ή ψευδές. Στη δική μου περίπτωση, αυτό που ήθελα, ήταν στη θέση του αντικείμενου2 να έχω το αποτέλεσμα από την συνάρτηση drag(), δηλαδή το e.target αφού θα ήταν ανώφελο να έχω 30 if(), κάθε μία για ένα πούλι. Μετά από αρκετό ψάξιμο, κατάφερα να

βρω ότι έπρεπε να προσθέσω απλά `e.target as Movieclip` μιας και η Actionscript θέλει να ξέρει τι είδους αντικείμενο είναι αυτό.

Πιο μικρά προβλήματα που επιλύθηκαν σχετικά γρήγορα ήταν αρχικά ένα αντιαισθητικό υπόλειμμα χρώματος που άφηναν τα πούλια, λόγω της λάθους χρήσης της εντολής `cacheAsBitmap` στις κολώνες οι οποίες παραμένουν σταθερές. Έπειτα η αντισυμβατική μάλλον επιλογή των προγραμματιστών της Adobe να ορίσουν ότι όταν ένα αντικείμενο γίνεται παιδί ενός άλλου, αυτόματα υιοθετεί και όλες τις ιδιότητές του. Λόγω αυτής της ιδιότητας, δε μπορούσα να δημιουργήσω απλά κολώνες με `alpha 0` (`alpha` είναι στη γλώσσα του Flash η διαφάνεια), γιατί μόλις ένα πούλι γινόταν παιδί της κολώνας ...εξαφανιζόταν.

Τέλος, είχα την επιθυμία να βάλω κάποιο μενού με το οποίο ο παίκτης θα μπορεί να κάνει διάφορες επιλογές χρωματισμών, ή να μπορεί να πάρει την κίνησή του πίσω, όμως το γεγονός ότι η εφαρμογή θα τρέχει σε ένα κινητό με την οθόνη να είναι περιορισμένου μεγέθους, και τα πούλια ήταν ήδη μικρά για να γίνονται `drag` εύκολα, στάθηκε τροχοπέδη σε κάτι τέτοιο. Έτσι προσπάθησα απλά να το κάνω όσο πιο απλό γίνεται, με τα πούλια όσο πιο μεγάλα, στα όρια του αισθητικού φυσικά.

### 6.3 Σημεία Βελτίωσης

Αυτά ήταν τα πιο σημαντικά κομμάτια του κώδικα της κύριας εφαρμογής. Βέβαια, υπάρχουν αρκετά περιθώρια βελτίωσης.

- Θα μπορούσαν οι χρήστες στην αρχή να δίνουν κάποιο κωδικό όνομα και να επιλέγουν οι ίδιοι τα χρώματα, τόσο των πουλιών όσο και του `background`.
- Επίσης θα ήταν αρκετά ρεαλιστικό, αν αντί για κουμπί για να ρίχνονται τα ζάρια, να γίνεται χρήση της βιβλιοθήκης που υλοποιεί συναρτήσεις για το επιταχυνσιόμετρο του iPhone, έτσι ώστε η ρίψη των ζαριών να γίνεται με ένα απλό τράνταγμα.
- Ένα άλλο σημείο που η εφαρμογή θα μπορούσε να βελτιωθεί, θα ήταν να κρατάει σκορ ανάμεσα στους δύο παίκτες καθώς στο συγκεκριμένο παράδειγμα, η εφαρμογή κλείνει όταν ένας απ τους δύο κερδίσει.

## Βιβλιογραφία

- [1] O'Reilly Essential Actionscript 3.0. Colin Mook. Actionscript 3.0 Programming Fundamentals, 2007.
- [2] Actionscript™ 3.0 Bible. Roger Braunstein, Mims H. Wright, Joshua J. Noble. Wiley Publishing, Inc, 2008.
- [3] O'Reilly Actionscript 3.0 Design Patterns. William B. Sanders – Chandima Cumaranatunge. Object-Oriented Programming Techniques, 2007.
- [4] <http://en.wikipedia.org>
- [5] <http://labs.adobe.com>
- [6] <http://www.apple.com>
- [7] <http://www.kirupa.org>
- [8] <http://www.byteme.gr>
- [9] <http://answers.oreilly.com>
- [10] <http://news.cnet.com>
- [11] <http://www.gotoandlearnforum.com>
- [12] <http://www.flashmagazine.com>
- [13] <http://www.authorstream.com>
- [14] <http://techcrunch.com>
- [15] <http://www.actionscript.org/>
- [16] <http://www.mikechambers.com>

## Προγράμματα που χρησιμοποιήθηκαν

- Adobe Flash CS5
- Adobe Photoshop CS4
- Microsoft Word 2007