



Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης

Σχολή Τεχνολογικών Εφαρμογών

Τμήμα Μηχανικών Πληροφορικής



Πτυχιακή Εργασία

Τίτλος: Ανάπτυξη οπτικοακουστικής εφαρμογής σε HTML5 και σύγκριση απόδοσης σε λειτουργικά Android, Firefox OS, Tizen

Κοκολάκη Μαρία Παναγιώτα AM 2386

Επιβλέπων Καθηγητής: Σπυρίδων Παναγιωτάκης

Ευχαριστίες

Θα ήθελα να ευχαριστήσω ιδιαίτερα τον επιβλέποντα καθηγητή μου, κ. Σπύρο Παναγιωτάκη για το ενδιαφέρον, και την αμέριστη βοήθεια που μου προσέφερε σε όλη τη διάρκεια εκπόνησης της πτυχιακής μου.

Περίληψη

Σκοπός της παρούσας πτυχιακής εργασίας είναι η ανάπτυξη μιας εφαρμογής επεξεργασίας οπτικού περιεχομένου (εικόνας) σε 3 διαφορετικές πλατφόρμες για κινητές συσκευές : το πρωτοεμφανιζόμενο λειτουργικό σύστημα Tizen, το επίσης νέο FirefoxOs και ένα τρίτο κάπως παλαιότερο και ήδη επιτυχημένο το Android. Παράλληλα, παρουσιάζονται τα FirefoxOs και Tizen καθώς και οι δυνατότητες της HTML5 γενικά αλλά και ειδικά ως εργαλείο ανάπτυξης εφαρμογών για κινητά περιβάλλοντα. Απώτερος στόχος είναι η σύγκριση των 3 αυτών συστημάτων ως προς τις δυνατότητες που παρέχει για την ίδια την ανάπτυξη της εφαρμογής καθώς και η σύγκριση των τριών αυτών λειτουργικών.

Abstract

This thesis aims at developing an application of image processing in 3 different mobile environments: newly entered Tizen, FirefoxOs, and last but not least Android. At the same time, HTML5, Tizen, and Firefox OS are being introduced in general, but also in particular as development tools for mobile platforms. Ultimate goal is to compare these environments in their own capabilities and also in particular as far as the image processing application is concerned.

ΠΕΡΙΕΧΟΜΕΝΑ

ΕΥΧΑΡΙΣΤΙΕΣ	II
ΠΕΡΙΛΗΨΗ	III
ABSTRACT	IV
ΠΕΡΙΕΧΟΜΕΝΑ	V
Πίνακας εικόνων.....	vii
ΠΙΝΑΚΑΣ ΠΙΝΑΚΩΝ	IX
ΚΕΦΑΛΑΙΟ 10-ΚΙΝΗΤΗ ΕΠΟΧΗ	1
1.1 ΕΙΣΑΓΩΓΗ.....	1
1.2 ΚΑΤΗΓΟΡΙΕΣ ΕΦΑΡΜΟΓΩΝ ΦΟΡΗΤΩΝ ΣΥΣΚΕΥΩΝ	2
ΚΕΦΑΛΑΙΟ 20-ΛΕΙΤΟΥΡΓΙΚΑ ΣΥΣΤΗΜΑΤΑ ΓΙΑ ΚΙΝΗΤΑ ΠΕΡΙΒΑΛΛΟΝΤΑ	5
2.1 ΕΙΣΑΓΩΓΗ.....	5
2.2 ΛΕΙΤΟΥΡΓΙΚΑ ΣΥΣΤΗΜΑΤΑ ΠΟΥ ΧΑΡΑΚΤΗΡΙΖΟΝΤΑΙ ΩΣ NATIVE :	5
2.2 ΣΥΣΤΗΜΑΤΑ ΠΟΥ ΧΑΡΑΚΤΗΡΙΖΟΝΤΑΙ ΩΣ WEB :	12
2.3 ΥΒΡΙΔΙΚΑ ΣΥΣΤΗΜΑΤΑ (HYBRID) :	13
ΚΕΦΑΛΑΙΟ 30 -Η HTML5- ΤΙ ΠΑΡΑΠΑΝΩ ΕΧΕΙ ΝΑ ΠΡΟΣΦΕΡΕΙ	17
3.1 ΕΙΣΑΓΩΓΗ.....	17
3.2 Η ΕΤΙΚΕΤΑ <CANVAS > ΩΣ ΣΤΟΙΧΕΙΟ ΤΗΣ HTML5 :	20
3.3 ΟΙ ΕΤΙΚΕΤΕΣ < VIDEO > ΚΑΙ <AUDIO>	20
3.4 Η ΤΟΠΙΚΗ ΑΠΟΘΗΚΕΥΣΗ ΔΕΔΟΜΕΝΩΝ ΣΤΗΝ HTML5 (WEB STORAGE).....	21
3.5 Ο ΓΕΩΓΡΑΦΙΚΟΣ ΕΝΤΟΠΙΣΜΟΣ ΤΗΣ ΘΕΣΗΣ ΤΟΥ ΧΡΗΣΤΗ ΩΣ ΕΡΓΑΛΕΙΟ ΤΗΣ HTML5	22
3.6 Η ΕΞΕΛΙΞΗ ΤΩΝ ΦΟΡΜΩΝ ΣΤΗΝ HTML5	24
3.6.1 Κείμενο μέσα σε πεδίο φόρμας.....	24
3.6.2 Αυτόματη εστίαση στο εκάστοτε επιλεγμένο πεδίο	25
3.6.3 Πρόσθετες ιδιότητες σε ετικέτες φόρμας	26
3.6.4 Επικύρωση φόρμας	27
3.6.5 Απαιτούμενα πεδία	28
ΚΕΦΑΛΑΙΟ 40-ΤΟ ΤΙΖΕΝ	30
4.1 ΕΙΣΑΓΩΓΗ.....	30
4.2 ΤΙ ΕΙΝΑΙ ΤΟ ΤΙΖΕΝ ;	30
4.3 ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΤΟΥ ΤΙΖΕΝ	31
4.3.1 Πλαίσιο Εφαρμογής.....	32
4.3.2 Η Βάση (Base)	33
4.3.3 Πυρήνας του Linux(Linux kernel)	33
4.4 Η ΕΓΚΑΤΑΣΤΑΣΗ ΤΗΣ ΠΛΑΤΦΟΡΜΑΣ	33
4.5 ΔΗΜΙΟΥΡΓΙΑ ΝΕΑΣ ΕΦΑΡΜΟΓΗΣ ΣΤΟ ΤΙΖΕΝ IDE	34
ΚΕΦΑΛΑΙΟ 50-ΤΟ FIREFOX OS	37
5.1 ΕΙΣΑΓΩΓΗ.....	38
5.2 Η ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΤΟΥ FIREFOX OS	38
5.3 ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΩΝ ΓΙΑ FIREFOX OS.....	40
5.4 ΕΓΚΑΤΑΣΤΑΣΗ ΠΡΟΣΟΜΟΙΩΣΗΣ ΣΕ Η/Υ	41
5.5 FIREFOX ΓΙΑ ANDROID	43
	v

ΚΕΦΑΛΑΙΟ 6Ο : ANDROID.....	45
6.1 ΕΙΣΑΓΩΓΗ.....	45
6.2 ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΤΟΥ ANDROID.....	46
6.2.1 Πυρήνας Linux (Kernel).....	47
6.2.2 Εγγενείς βιβλιοθήκες (Native Libraries).....	48
6.2.3 Χρόνος εκτέλεσης.....	48
6.2.4 Η εικονική μηχανή Dalvik (VM –Virtual Machine).....	48
6.2.5 Πλαίσιο εφαρμογής (Application Framework).....	48
6.3 ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΩΝ ΓΙΑ ANDROID.....	49
ΚΕΦΑΛΑΙΟ 7Ο:ΤΟ PHONEGAP.....	51
7.1 ΕΙΣΑΓΩΓΗ.....	51
7.2 ΧΡΗΣΗ ΤΟΥ PHONEGAP.....	52
7.2.1 Απευθείας εγκατάσταση και τοπική χρήση του PhoneGap για μετατροπή εφαρμογής.....	52
7.2.2.Μετατροπή μέσω PhoneGap Build.....	54
ΚΕΦΑΛΑΙΟ 8Ο :ΜΙΑ ΕΦΑΡΜΟΓΗ ΕΠΕΞΕΡΓΑΣΙΑΣ ΕΙΚΟΝΑΣ ΣΕ HTML5 /JAVASCRIPT.....	57
8.1 ΕΙΣΑΓΩΓΗ.....	57
8.2 Η ΒΑΣΙΚΗ ΥΛΟΠΟΙΗΣΗ ΣΕ HTML5/JAVASCRIPT.....	57
8.3 ΠΑΡΑΜΕΤΡΟΠΟΙΗΣΗ ΤΗΣ ΒΑΣΙΚΗΣ ΕΦΑΡΜΟΓΗΣ ΓΙΑ ΠΕΡΙΒΑΛΛΟΝ ΤΙΖΕΝ.....	64
8.4 ΠΑΡΑΜΕΤΡΟΠΟΙΗΣΗ ΤΗΣ ΒΑΣΙΚΗΣ ΕΦΑΡΜΟΓΗΣ ΓΙΑ FIREFOX OS.....	71
8.5 ΠΑΡΑΜΕΤΡΟΠΟΙΗΣΗ ΤΗΣ ΕΦΑΡΜΟΓΗΣ ΓΙΑ ANDROID/PHONEGAP BUILD.....	73
ΚΕΦΑΛΑΙΟ 9Ο: ΣΥΓΚΡΙΣΗ ΥΛΟΠΟΙΗΣΕΩΝ.....	82
9.1 ΕΙΣΑΓΩΓΗ.....	82
9.2 ΠΟΙΟΤΙΚΗ ΣΥΓΚΡΙΣΗ.....	82
9.3 ΠΟΣΟΤΙΚΗ ΕΚΤΙΜΗΣΗ.....	83
ΚΕΦΑΛΑΙΟ 10Ο :ΣΥΜΠΕΡΑΣΜΑΤΑ.....	86
ΒΙΒΛΙΟΓΡΑΦΙΑ.....	87

Πίνακας εικόνων

Εικόνα 1: Στατιστικά στοιχεία χρήσης φορητών συσκευών	2
Εικόνα 2: Τα κυριότερα λειτουργικά για κινητά περιβάλλοντα	4
Εικόνα 3: Η έναρξη των Windows Phone	6
Εικόνα 4: Τρεις συσκευές με Windows Phone	7
Εικόνα 5: Η αρχική εικόνα του BlackBerry Os	8
Εικόνα 6: Μοντέλα Blackberry με το αντίστοιχο λειτουργικό	9
Εικόνα 7: Η home screen του iOS 7.	10
Εικόνα 8: Η εξέλιξη του iOS αλλά και του iPhone από το 2007	11
Εικόνα 9: Η Home screen του Jolla με το Sailfish	12
Εικόνα 10: Το λογότυπο του Ubuntu	13
Εικόνα 11: Η Home screen του Ubuntu Touch	14
Εικόνα 12: Η Home screen του Chrome Os	15
Εικόνα 13: Το λογότυπο του Chrome Os	16
Εικόνα 14: Η HTML5 και οι δυνατότητές της	17
Εικόνα 15: Μερίδιο αγοράς	19
Εικόνα 16 :Κώδικας της HTML5 για την ενσωμάτωση video	21
Εικόνα 17 :Info-bar για την ενημέρωση και αδειοδότηση από το χρήστη	23
Εικόνα 18: Εκδόσεις φυλλομετρητών που υποστηρίζουν τον γεωγραφικό εντοπισμό	24
Εικόνα 19 : Το πεδίο εισαγωγής της φόρμας πριν το κλικ του χρήστη	25
Εικόνα 20: Το πεδίο εισαγωγής της φόρμας μετά το κλικ του χρήστη	25
Εικόνα 21 : αυτόματη επικύρωση φόρμας ,χωρίς να προηγηθεί «Submit»	28
Εικόνα 22: άρνηση του φυλλομετρητή να δεχθεί την τιμή που εισάγεται	28
Εικόνα 23: Ο φυλλομετρητής δεν δέχεται τα υποχρεωτικά πεδία να υποβληθούν κενά	29
Εικόνα 24 : Αίωρηση του δείκτη του ποντικιού μπορεί να εμφανίσει πληροφορία για απαιτούμενα πεδία	29
Εικόνα 25: Λογότυπο του Tizen	30
Εικόνα 26: Στοιβα λογισμικού αρχιτεκτονικής του Tizen	32
Εικόνα 27: Εγκατάσταση Tizen IDE	34
Εικόνα 28: Δημιουργία νέου project σε περιβάλλον Tizen	35
Εικόνα 29 :Επιλογή του είδους της εφαρμογής ως προς την πολυπλοκότητά της.	35
Εικόνα 30: Τελευταίο στάδιο της δημιουργίας του νέου project	36
Εικόνα 31: Η πρώτη επαφή με τον κώδικα του project που μόλις δημιουργήσαμε	36
Εικόνα 32: Η πρώτη επαφή με την εφαρμογή	37
Εικόνα 33 : Το λογότυπο του Firefox OS	38
Εικόνα 34 : Η αρχιτεκτονική του Firefox OS σχηματικά	40
Εικόνα 35: Το Add-on του Mozilla Firefox που λειτουργεί ως προσομοιωτής για τις εφαρμογές του FirefoxOS	42
Εικόνα 36: Η Home screen του Firefox OS	43
Εικόνα 37: Το λογότυπο του Android	45
Εικόνα 38: Η Home screen του Android σε διαφορετικές συσκευές .Από αριστερά Samsung, Samsung, HTC	46
Εικόνα 39 : Η στοιβα λογισμικού της αρχιτεκτονικής του Android	47
Εικόνα 40 : Το λογότυπο του PhoneGap	51
Εικόνα 41: Η εγκατάσταση του PhoneGap	53
Εικόνα 42: Η σελίδα του PhoneGap όπου μπορούμε να ανεβάσουμε την εφαρμογή μας προς μετατροπή μέσω PhoneGap Build.	55
Εικόνα 43 : Το αποσυμπιεσμένο αρχείο τοπικής εγκατάστασης του PhoneGap	56
Εικόνα 44 : Σχηματική απεικόνιση της διεργασίας που επιτελεί το PhoneGap Build.	56
Εικόνα 45: Ο κώδικας HTML που χρησιμοποιήθηκε για την εφαρμογή (1)	58
Εικόνα 46: Ο κώδικας HTML που χρησιμοποιήθηκε για την εφαρμογή (2)	58
Εικόνα 47 : Ο κώδικας HTML που χρησιμοποιήθηκε για την εφαρμογή (3)	59

Εικόνα 48 : Ο κώδικας Javascript που χρησιμοποιήθηκε για την εφαρμογή (1).	59
Εικόνα 49: Ο κώδικας Javascript που χρησιμοποιήθηκε για την εφαρμογή (2)	60
Εικόνα 50 : Ο κώδικας Javascript που χρησιμοποιήθηκε για την εφαρμογή των φίλτρων (3).	60
Εικόνα 51 : Ο κώδικας Javascript που χρησιμοποιήθηκε για την εφαρμογή των φίλτρων (4)	61
Εικόνα 52 : Στιγμιότυπο από τη φόρτωση της εφαρμογής μέσω wamp server	61
Εικόνα 53 :Στιγμιότυπο από τη φόρτωση εικόνας στην εφαρμογή	62
Εικόνα 54: Εφαρμογή του greyscale φίλτρου στην φορτωθείσα εικόνα	62
Εικόνα 55:Εφαρμογή του red φίλτρου στην φορτωθείσα εικόνα	62
Εικόνα 56: Εφαρμογή του sepia φίλτρου στην εικόνα	63
Εικόνα 57 :Στιγμιότυπο της εφαρμογής με εφαρμοσμένο το φίλτρο brighter 2 φορές	63
Εικόνα 58 :Στιγμιότυπο της εφαρμογής με εφαρμοσμένο το φίλτρο darker 2 φορές	63
Εικόνα 59: Το αρχείο index.html της εφαρμογής.	64
Εικόνα 60:Το αρχείο της main σε javascript.	65
Εικόνα 61:Το αρχείο filters.js	66
Εικόνα 62:Η εφαρμογή τρέχει μέσω προσομοιωτή.	66
Εικόνα 63:Η αρχική διεπαφή της εφαρμογής διαμορφωμένη μέσω divs .	67
Εικόνα 64:Ανοιγμα παραθύρου για επιλογή εικόνας μέσω windows.	67
Εικόνα 65: Η αρχική εικόνα που φορτώνουμε στην εφαρμογή	68
Εικόνα 66:Η εφαρμογή του πρώτου φίλτρου (greyscale)	68
Εικόνα 67:Η εφαρμογή του δεύτερου φίλτρου (sepia)	69
Εικόνα 68 : Η εφαρμογή του τρίτου φίλτρου (red)	69
Εικόνα 69: Η εφαρμογή του τέταρτου φίλτρου (φωτεινότητα-brighter) εφαρμοσμένο 2 φορές.	70
Εικόνα 70: Η εφαρμογή του πέμπτου φίλτρου (darker) εφαρμοσμένο 2 φορές.	70
Εικόνα 71: Το αρχείο manifest.webapp της εφαρμογής	72
Εικόνα 72: Ο Dashboard αφού έχει ελέγξει το αρχείο manifest για την εγκυρότητά του.	72
Εικόνα 73: Η αρχική σελίδα της εφαρμογής μας σε περιβάλλον FirefoxOS	73
Εικόνα 74: Στιγμιότυπο από την εγκατάσταση του BlueStacks (1)	74
Εικόνα 75 : Στιγμιότυπο από την εγκατάσταση του BlueStacks (2)	74
Εικόνα 76: Στιγμιότυπο από την εγκατάσταση του BlueStacks (3)	75
Εικόνα 77: Στιγμιότυπο από την εγκατάσταση του BlueStacks (4)	75
Εικόνα 78:Το φόρτωμα του Bluestacks μετά την εγκατάσταση.	76
Εικόνα 79: Αυτόματη εγκατάσταση του ark αρχείου της εφαρμογής, μόνο με διπλό κλικ πάνω στο αρχείο	76
Εικόνα 80:Η εφαρμογή μας εγκατεστημένη έτοιμη στο Bluestacks.	77
Εικόνα 81 : Φόρτωση της εφαρμογής μας σε περιβάλλον Android με προσομοίωση Bluestacks	77
Εικόνα 82:Η εφαρμογή μας σε περιβάλλον Android	78
Εικόνα 83:Επιλογή αρχείου μέσω file selector του Android	78
Εικόνα 84: Η εφαρμογή επεξεργασίας εικόνας με φορτωμένη την εικόνα χωρίς τα φίλτρα	79
Εικόνα 85:Η εικόνα με εφαρμοσμένο το φίλτρο για ασπρόμαυρη μετατροπή	79
Εικόνα 86:Η εικόνα με εφαρμοσμένο το φίλτρο sepia	80
Εικόνα 87:Η εικόνα με εφαρμοσμένο το φίλτρο red	80
Εικόνα 88 :Η εικόνα με εφαρμοσμένο το φίλτρο φωτεινότητας 2 φορές	81
Εικόνα 89:η εφαρμογή με το φίλτρο ελάττωσης φωτεινότητας εφαρμοσμένο 2 φορές.	81
Εικόνα 90 :Μέτρηση με την αρχική φόρτωση της εφαρμογής στον Chrome	84
Εικόνα 91: Μέτρηση με φόρτωση εικόνας και φίλτρου στην εφαρμογή	84

Πίνακας πινάκων

<i>Πίνακας 1: Η HTML5 και η συμβατότητα με διάφορους φυλλομετρητές σε διαφορετικά περιβάλλοντα</i>	<i>19</i>
<i>Πίνακας 2: οι αντιστοιχίες φυλλομετρητών με τις τεχνολογίες που υποστηρίζουν</i>	<i>21</i>
<i>Πίνακας 3: οι εκδόσεις των δημοφιλέστερων φυλλομετρητών που υποστηρίζουν τοπική αποθήκευση δεδομένων</i>	<i>22</i>

Κεφάλαιο 1ο-Κινητή εποχή

Formatted: Font: (Default) Times New Roman, 24 pt, Bold, Font color: Blue, Greek, Kern at 18 pt

1.1 Εισαγωγή

Στην εποχή που διανύουμε η κινητή τηλεφωνία σε συνδυασμό με το ασύρματο ίντερνετ έχουν μπει για τα καλά στη ζωή μας. Έχουν διαδοθεί ευρέως και έχει φέρει το ίντερνετ στο κινητό με τόσο γρήγορη σύνδεση όσο και στο σπίτι από τη σταθερή σύνδεσή μας. Αυτό έχει σαν επακόλουθο την ανάγκη για εφαρμογές που θα εξυπηρετήσουν όποιον βρίσκεται εν κινήσει –για παράδειγμα για πλοήγηση ή για εύρεση διανυκτερευόντων καταστημάτων. Στην πορεία όταν η βασική ανάγκη αυτή καλύφθηκε, δημιουργήθηκαν άλλες, αυτή της κοινωνικής δικτύωσης και της ψυχαγωγίας **εν κινήσει** γενικότερα, οπότε επεκτάθηκε και η αγορά της κινητής τηλεφωνίας ως προς τις συσκευές που μπορούν να ανταποκριθούν σε αυτές τις απαιτήσεις-τα λεγόμενα smartphones.

Σύμφωνα με στατιστικές, από τα 4 δις. κινητά που υπάρχουν στον πλανήτη, τα 1,08 δις. είναι smartphones, τα οποία μάλιστα πραγματοποιούν τις μισές από το σύνολο των αναζητήσεων σύμφωνα με τις ίδιες στατιστικές. Ο αριθμός των ατόμων που έχουν πρόσβαση στο Διαδίκτυο από τις κινητές συσκευές αυξάνεται με ταχείς ρυθμούς και αναμένεται μέσα στα επόμενα πέντε χρόνια να ξεπεράσει τον αριθμό των ατόμων που έχουν πρόσβαση από PC, κάνοντας το smartphone το πιο δημοφιλές μέσο πρόσβασης στο Διαδίκτυο. Το 86% των χρηστών χρησιμοποιούν τις κινητές τους συσκευές (smartphone/tablet), σε συνδυασμό με την παρακολούθηση τηλεόρασης. Το 61% των χρηστών χρησιμοποιούν τις κινητές συσκευές τους για να παίξουν παιχνίδια. Ο μέσος όρος των Αμερικάνων ξοδεύει 2,7 ώρες την ημέρα σε "socializing" από μια φορητή συσκευή (π.χ. social media εφαρμογές κλπ.). Στο Σημ 1 παρακάτω **γράφημα-εικόνα** φαίνεται ποσοστιαία για ποιο λόγο χρησιμοποιούν περισσότερο οι χρήστες τις φορητές συσκευές τους.



Εικόνα 1: Στατιστικά στοιχεία χρήσης φορητών συσκευών

1.2 Κατηγορίες εφαρμογών φορητών συσκευών

Τα τελευταία 3 χρόνια έχουν αναπτυχθεί πάνω από 300.000 εφαρμογές. Τις καλύτερες εφαρμογές οι χρήστες τις έχουν κατεβάσει 10,9 δισεκατομμύρια φορές. Οι εφαρμογές μπορούν να χωριστούν γενικά σε 2 κατηγορίες ως προς το σκοπό:

- **Χρηστικές εφαρμογές:** σε αυτές συγκαταλέγονται οι εφαρμογές που προσφέρουν μια πληροφορία ή μια υπηρεσία στο χρήστη, όπως πλοήγηση από ένα σημείο στο άλλο (GPS), ανεύρεση διανυκτερευόντων φαρμακείων ή νοσοκομείων μέσω σύνδεσης σε κάποια βάση δεδομένων, πρόγνωση καιρού, ανεύρεση τραγουδιού μέσω ηχογράφησης του, μηχανές αναζήτησης Ιστού κλπ.
- **Ψυχαγωγικές εφαρμογές:** αυτές μπορεί να είναι παιχνίδια, κοινωνική δικτύωση (όπως το facebook ή το twitter, το instagram), παρακολούθηση βίντεο μουσικής ή άλλου περιεχομένου (όπως το YouTube) κλπ.

Υπάρχουν όμως και εφαρμογές που **επιτελούν και τους 2 προαναφερθέντες σκοπούς** (και ψυχαγωγία και πληροφορία) όπως είναι το **foursquare**, που παρέχει πληροφορίες (όπως διεύθυνση, βαθμολογίες και σχόλια από άλλους χρήστες) για bar, restaurant, καταστήματα και προτείνει κοντινά μέρη ανιχνεύοντας τη θέση του χρήστη μέσω GPS, ενώ συγχρόνως ο χρήστης μπορεί να κοινοποιήσει στους φίλους του το πού βρίσκεται καθώς και να κάνει και σχόλια όσον αφορά την επίσκεψή του για την ενημέρωση των φίλων του.

Στην ίδια φιλοσοφία κινείται και το DogSquare, στο οποίο δημιουργεί ο χρήστης το προφίλ του σκύλου του, μοιράζεται με φίλους τη βόλτα, την επίσκεψη στον κτηνίατρο, την ύπαρξη αδέσποτου σκύλου στη γειτονιά του, ή ακόμα και την απώλεια του αγαπημένου του τετράποδου μαζί με την τοποθεσία που χάθηκε ώστε να βρεθεί ευκολότερα, ακόμα και από αγνώστους χρήστες. Μπορούμε να πούμε ότι υπάρχουν εκατοντάδες εφαρμογές που κινούνται στο ίδιο πλαίσιο δηλαδή αυτό του συνδυασμού της κοινωνικής δικτύωσης, σε συνδυασμό με κάποιο τομέα, κάποιο ενδιαφέρον για το οποίο παρέχουν πληροφορίες ή ακόμα και και μετρήσεις οπότε ο απόλυτος, αυστηρός διαχωρισμός ως προς το σκοπό δεν είναι πολύ εύκολος.

Διαχωρίζοντας τις εφαρμογές ως προς το αν απαιτείται σύνδεση στο Διαδίκτυο μέσω του κινητού ή αν δεν απαιτείται για να έχουμε πρόσβαση σε αυτές (δηλ τοπικές κατά μια έννοια), μπορούμε να διακρίνουμε και έναν άλλο τρόπο διαχωρισμού : Σε native και web εφαρμογές :

- **Native (γηγενείς)** : οι native εφαρμογές μπορούμε να πούμε ότι «ζούν» μέσα στην συσκευή και μπορούμε να τις τρέξουμε μέσω ενός εικονιδίου που βρίσκεται είτε στην οθόνη υποδοχής της συσκευής είτε στο βασικό μενού. Αυτού του είδους τις εφαρμογές τις εγκαθιστά ο χρήστης κατεβάζοντάς τις από κάποιο ηλεκτρονικό κατάστημα, είτε την εγκαθιστά μέσω υπολογιστή. Αναπτύσσονται αποκλειστικά για ένα συγκεκριμένο λειτουργικό σύστημα και μπορούν να αξιοποιήσουν πλήρως τις δυνατότητες της εν λόγω συσκευής. Το βασικό είναι ότι λειτουργούν και χωρίς σύνδεση στο internet και συμπεριλαμβάνονται στις ειδοποιήσεις εφαρμογών της συσκευής.
- **Web (Ιστού)**: οι εφαρμογές Ιστού είναι στην πραγματικότητα ιστοσελίδες, που μοιάζουν με native εφαρμογές αλλά δεν είναι υλοποιημένες ως τέτοιες. «Τρέχουν» μέσω φυλλομετρητών, και κατά κύριο λόγο έχουν αναπτυχθεί με γλώσσα HTML5. Οι χρήστες αρχικά αποκτούν πρόσβαση σε αυτές μέσω ενός URL, και στη συνέχεια μπορούν είτε να τις εγκαταστήσουν όπως όλες τις εφαρμογές, είτε να τις «καρφίτσώσουν» στην επιφάνεια εργασίας τους ως σελιδοδείκτες για ευκολότερη πρόσβαση.

Αυτού του είδους οι εφαρμογές έγιναν πολύ δημοφιλείς όταν εμφανίστηκε στην αγορά η HTML5, και το κοινό κατάλαβε ότι μπορεί να υπάρξει λειτουργικότητα παρόμοια με αυτή των native, στις web εφαρμογές (άρα επί της ουσίας και στο φυλλομετρητή) και έτσι πλέον ο διαχωρισμός μεταξύ native εφαρμογών και ιστοσελίδων έγινε ακόμα δυσκολότερος.

Οι εφαρμογές μπορούν να χωριστούν και αλλιώς, ως προς το οικονομικό τους κομμάτι:

- **Δωρεάν** : Στην περίπτωση των δωρεάν εφαρμογών, ο χρήστης κατεβάζει απλώς την εφαρμογή χωρίς να απαιτείται οικονομική συμμετοχή για την χρήση της .
- **Με οικονομική επιβάρυνση** : Αυτή η κατηγορία των εφαρμογών απαιτούν πληρωμή για να μπορέσει ο χρήστης να τις χρησιμοποιήσει, συνήθως κάποιο πολύ χαμηλό ποσό, ώστε να υπάρχει κάποιο σχετικό κίνητρο από το χρήστη για την απόκτησή τους.

Στην παρούσα πτυχιακή εργασία που πραγματεύεται την σύγκριση και την απόδοση τριών λειτουργικών συστημάτων κινητών συσκευών, παρουσιάζονται τρία λειτουργικά, καθώς και οι δυνατότητες των διαφορετικών πλατφορμών που χρησιμοποιούνται για την ανάπτυξη της εφαρμογής. Τα λειτουργικά αυτά είναι το Tizen, Android, Firefox OS. Τα δύο εκ των τριών (Tizen, Firefox OS) χρησιμοποιούν την HTML5 για την ανάπτυξη των εφαρμογών. Η HTML5 ως εξέλιξη της HTML δίνει παραπάνω δυνατότητες και αυτές πρόκειται να αναδειχθούν στα επόμενα κεφάλαια.



Εικόνα 2: Τα κυριότερα λειτουργικά για κινητά περιβάλλοντα

Κεφάλαιο 2ο-Λειτουργικά συστήματα για κινητά περιβάλλοντα

2.1 Εισαγωγή

Η τεχνολογία των smartphone, καθώς προχωρά , μεγαλώνει και η αγορά των προϊόντων της με τη σειρά της-και δε μιλάμε μόνο για τις εφαρμογές, αλλά και για τα ίδια τα λειτουργικά συστήματα που θα τις υποστηρίζουν. Δεν είναι δυνατό άλλωστε η αγορά των συστημάτων να έμνε πίσω στο κομμάτι της κινητής τηλεφωνίας. Έτσι όλα τα μεγάλα συστήματα (μεγάλα όσον αφορά την ανταπόκριση από το κοινό πάντα) αναπτύχθηκαν και σε εκδόσεις για κινητά τηλέφωνα ώστε να επωφεληθούν οικονομικά αλλά και τεχνολογικά από την αύξηση αυτή στην αγορά του νέου προϊόντος, χωρίς αυτό να σημαίνει ότι δεν δημιουργήθηκαν και νέα προϊόντα στο κομμάτι αυτό. Προϊόντα χωρίς δηλαδή να έχει προηγηθεί κάποιο αντίστοιχο προϊόν για ηλεκτρονικούς υπολογιστές που να δημιουργήσει το έδαφος στην αγορά .

Στην περίπτωση των λειτουργικών συστημάτων, και την ανταπόκριση που θα έχουν στο κοινό, μεγάλο ρόλο παίζει και το περιβάλλον(η γλώσσα) το οποίο απαιτείται για την ανάπτυξη των εφαρμογών, διότι καθορίζει το επίπεδο δυσκολίας στην ανάπτυξη τους αρχικά και το πλήθος των διατιθέμενων εφαρμογών στη συνέχεια. Μπορούμε λοιπόν να τα ταξινομήσουμε με βάση το περιβάλλον ανάπτυξης , καθώς αυτό είναι που καθορίζει την ταυτότητα του κάθε συστήματος και κατ' επέκταση και των εφαρμογών του .

Στην αυστηρή ορολογία των λειτουργικών συστημάτων, native (γηγενή) συστήματα είναι αυτά που αναπτύσσονται και προορίζονται για συγκεκριμένες συσκευές, των οποίων τα χαρακτηριστικά μπορούν να αξιοποιηθούν πλήρως από το λειτουργικό σύστημα. Υπάρχουν αυτά τα συστήματα που το περιβάλλον και η γλώσσα προγραμματισμού που χρησιμοποιείται για την ανάπτυξη των εφαρμογών είναι τέτοιο που προορίζεται για την ανάπτυξη γηγενών εφαρμογών (native όπως λέμε, χρησιμοποιώντας λιγότερο αυστηρούς όρους, και αναφερόμενοι σε εφαρμογές) και συνήθως είναι σε γλώσσα C++ ή Java και σπανιότερα σε γλώσσα Objective C. Από την άλλη πλευρά υπάρχουν και εκείνα τα συστήματα που οι εφαρμογές τους προορίζεται να «τρέχουν» μέσω φυλλομετρητών, και άρα πρόκειται για εφαρμογές Ιστού όπως λέμε, επομένως αναπτύσσονται με Javascript /HTML5. Η διαφορά δεν είναι εμφανής, αλλά ουσιαστική όπως αναλύθηκε παραπάνω, στη συνέχεια θα δούμε ένα πιο γενικό διαχωρισμό των συστημάτων σε native και web .

2.2 Λειτουργικά συστήματα που χαρακτηρίζονται ως native :

Windows phone: είναι η έκδοση του λειτουργικού συστήματος Windows της Microsoft για έξυπνα κινητά τηλέφωνα. Αναπτύχθηκε στις αρχές του 2010, ενώ στις πρώτες εκδόσεις του είχε το όνομα **Windows**

Mobile, παρόλο που στην πορεία η νεότερη έκδοση με την νέα ονομασία δεν είναι συμβατή με κάποια από κείνες τις εκδόσεις. Το Windows phone δεν είναι ανοιχτού κώδικα, που σημαίνει ότι δεν μπορεί κανείς να αναπτύξει και δημοσιεύσει εφαρμογές ούτε να επεξεργαστεί τον υπάρχοντα κώδικα των εφαρμογών ή του ίδιου του λειτουργικού (με σκοπό τη βελτίωση, για παράδειγμα) χωρίς την άδεια της εταιρείας που κατέχει τα δικαιώματα.



Εικόνα 3: Η έναρξη των Windows Phone

Όπως κάθε λειτουργικό σύστημα για smartphones έτσι και τα **Windows Phone** προσφέρουν στους χρήστες μια εύκολη κι έξυπνη χρήση που μπορούν να κάνουν την εμπειρία τους με τα smartphones πιο απλή, όπως απλή μπορούν να κάνουν και την καθημερινότητα τους αφού πολλές λειτουργίες που προσφέρουν οι υπολογιστές έχουν προστεθεί στα Windows Phone. Ο Internet Explorer προσφέρει στους χρήστες την περιήγηση στο διαδίκτυο, ενώ έχει προστεθεί και το Microsoft Office το οποίο προσφέρει εφαρμογές γραφείου όπως το Word, το Power

Point κ.α. Το Windows app store προσφέρει εφαρμογές και παιχνίδια, δωρεάν και επί πληρωμή, και επιπλέον προστέθηκε και το **Xbox** (γνωστή παιχνιδο-κονσόλα της Microsoft) οπου σε αυτό οι χρήστες παίζουν τα παιχνίδια που κατεβάζουν. Επίσης, οι χρήστες μπορούν να δούνε τα email τους εν κινήσει, να βοηθηθούν στην μετακίνηση τους από τους χάρτες, να ενημερώνονται στα κοινωνικά δίκτυα (Facebook, Twitter κ.α) και πολλά άλλα. Οι εφαρμογές, αλλά και το ίδιο το λειτουργικό έχουν αναπτυχθεί σε γλώσσα C++ και C. Το Δεκέμβριο του 2013 κυκλοφόρησε η τελευταία έκδοση με το 3^ο update του Windows phone 8.

Η τελευταία έκδοση αυτή μάλιστα, υποστηρίζει πολλαπλούς επεξεργαστές και οθόνη υψηλής ανάλυσης καθώς και multi-tasking ώστε να μπορεί να ανταγωνιστεί τα Android και iOS που υπερτερούν φυσικά ως εμπορικά προϊόντα. Τον Φεβρουάριο του 2011 η Microsoft ανακοίνωσε την συνεργασία τη με τη Nokia, με σκοπό η τελευταία να κυκλοφορήσει τις επόμενες συσκευές της με νέο λειτουργικό, αντικαθιστώντας το Symbian. Άλλες εταιρείες που κυκλοφορούν μοντέλα με Windows phone είναι η HTC, η Samsung και η Huawei. Τέλος, αξίζει να σημειωθεί ότι από το 2010 μέχρι σήμερα οι πωλήσεις των Windows Phone έχουν αυξηθεί κατά πολύ και έφτασαν στο Νο2 των πωλήσεων στην κατηγορία των smartphones και υπολογίζεται πως μέχρι το 2017 θα αυξηθούν κι άλλο οι πωλήσεις τους.

Στην παρακάτω εικόνα φαίνονται 3 συσκευές με Windows Phone, από αριστερά προς δεξιά HTC 8X, Lumia 920, Lumia 820.



Εικόνα 4: Τρεις συσκευές με Windows Phone

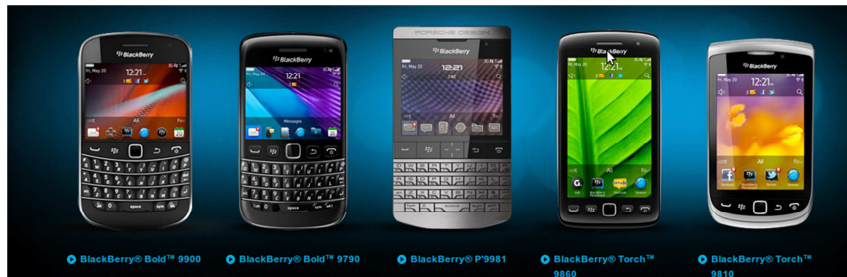
Blackberry: είναι λειτουργικό για κινητά τηλέφωνα ιδιοκτησίας της Blackberry SA από τον Καναδά, και το οποίο αναπτύχθηκε αποκλειστικά για τις συσκευές της (επομένως μπορούμε να πούμε ότι αποτελεί καθαρά native λειτουργικό). Αυτό σημαίνει ότι δεν είναι ανοιχτού κώδικα και άρα δε μπορεί κανείς να επεξεργαστεί τον

υπάρχοντα κώδικα των εφαρμογών ή του ίδιου του λειτουργικού) χωρίς την άδεια της εταιρείας που κατέχει τα δικαιώματα. Οι εφαρμογές για Blackberry αναπτύσσονται σε γλώσσα Java(Micro Edition). Το συγκεκριμένο λειτουργικό υποστηρίζει multi-tasking και εξειδικευμένες συσκευές εισόδου που υιοθετήθηκαν από την Blackberry SA για τις κινητές της συσκευές όπως οι trackwheel(ρολέτα κύλισης), trackball (μπάλα κύλισης), και πιο πρόσφατα trackpad (επιφάνεια αντίχτυσης), καθώς και touchscreen (οθόνη αφής).



Εικόνα 5:Η αρχική εικόνα του BlackBerry Os

Το Blackberry είναι δημοφιλές για την υποστήριξη εταιρικού mail μέσω του MIDP 1.0 (Mobile Information Device Profile- προσδιορισμός που αφορά την χρήση της Java σε ενσωματωμένα συστήματα όπως οι κινητές συσκευές), το οποίο υποστηρίζει πλήρη ασύρματη ενεργοποίηση, σύνδεση και συγχρονισμό με τα Microsoft Exchange, Lotus Domino καθώς και με τις σημειώσεις, τις εργασίες, το ημερολόγιο, τις επαφές όταν χρησιμοποιούνται μαζί με το BlackBerry Enterprise Server. Updates το Blackberry κατεβάζει αυτόματα όταν συνδέεται με ασύρματους φορείς που υποστηρίζουν το σύστημα OTASL(Over The Air Software Loading). Το Blackberry επίσης υποστηρίζει WAP 1.0.



Εικόνα 6: Μοντέλα Blackberry με το αντίστοιχο λειτουργικό

iOS:(προηγούμενος iPhone OS) είναι ένα λογισμικό για κινητά το οποίο αναπτύχθηκε και διανέμεται από την Apple Inc., σε 34 γλώσσες, και έδειξε ουσιαστικά το δρόμο για τη δημιουργία και την εξέλιξη των smartphones. Αρχικά παρουσιάστηκε το 2007 για το iPhone, ενώ υποστηρίζει και άλλες συσκευές της Apple όπως το iPod touch, το iPad και το Apple TV. Αντίθετα από το Windows Phone της Microsoft και το Android της Google, η Apple δεν δίνει την άδεια για την εγκατάσταση του λογισμικού iOS σε συσκευές που δεν είναι κατασκευής Apple(μπορούμε επομένως να πούμε ότι είναι κλειστού κώδικα). Η διεπαφή χρήστη του iOS βασίζεται στη λογική του απευθείας χειρισμού, χρησιμοποιώντας πολλαπλές χειρονομίες. Τα στοιχεία χειρισμού αποτελούνται από κουμπιά, μπάρες, διακόπτες και η αλληλεπίδραση περιλαμβάνει σύρσιμο του δακτύλου, χτύπημα και άνοιγμα των δακτύλων και αντίθετο άνοιγμα, και όλα αυτά έχουν συγκεκριμένη χρήση στο περιβάλλον του iOS. Εσωτερικά ποτενσιόμετρα χρησιμοποιούνται σε ορισμένες εφαρμογές προκειμένου να ανιχνευτεί η κίνηση και η περιστροφή της συσκευής, ούτως ώστε να αλλάξει και η προβολή σε οριζόντια(landscape) από κάθετη και αντίστροφα, ή να επιτευχθεί αναίρεση της τελευταίας ενέργειας.



Εικόνα 7: Η home screen του iOS 7.

Η Apple για τις συσκευές που φορτώνει το λειτουργικό της επιλέγει δυνατές πλατφόρμες 64-bit-ης αρχιτεκτονικής: A4, A5, A6, A7. Τώρα όσον αφορά το ίδιο το λειτουργικό, κυκλοφόρησε πρώτη φορά το 2007 μαζί με το πρώτο iPhone, και έκτοτε κυκλοφορεί κάθε χρόνο μια νέα έκδοση, με την τελευταία, το iOS 7 να έχει κυκλοφορήσει το 2013, το Σεπτέμβριο. Τα updates του iOS κυκλοφορούν μέσω του iTunes, και επίσης μέσω του συστήματος Over-The-Air (που χρησιμοποιεί κυρίως ασύρματη σύνδεση).

Στις Ιανουαρίου 2013, το App Store της Apple περιείχε περισσότερο από 775.000 εφαρμογές iOS, 300.000 από τις οποίες ήταν συμβατές για iPad, κάποιες δωρεάν και κάποιες όχι. Αυτές οι εφαρμογές έχουν μεταφορτωθεί συνολικά περισσότερα από 50 δισεκατομμύρια φορές. Οι εφαρμογές για iOS αναπτύσσονται σε γλώσσες C++, C-objective και C, και προκειμένου να «ανέβουν» στο AppStore απαιτείται πρώτα η έγκριση της Apple. Από τα κέρδη μιας εφαρμογής, το 70% προορίζεται για τον ή τους δημιουργούς, και το 30% για την Apple. Αυτή τη στιγμή, εμπορικά η Apple κατέχει το 60% της αγοράς των εφαρμογών σε προτίμηση του κοινού (έναντι του 40% του Android), παρόλο που το Android παραμένει δημοφιλέστερο γενικά ως λειτουργικό .



Εικόνα 8: Η εξέλιξη του iOS αλλά και του iPhone από το 2007

Android: Το **Android** είναι λειτουργικό σύστημα για συσκευές κινητής τηλεφωνίας το οποίο τρέχει τον πυρήνα του λειτουργικού Linux. Αποτελεί ένα καθαρά native λειτουργικό σύστημα, το οποίο αναπτύχθηκε ταχύτατα μέσα σε λίγα χρόνια, και έχει εξελιχθεί στο Νο1 λειτουργικό για κινητές συσκευές. Σε επόμενο κεφάλαιο θα αναλυθεί εκτενώς, καθώς επίσης θα παρουσιαστεί και μια εφαρμογή σε Android που αναπτύχθηκε στα πλαίσια αυτής της πτυχιακής εργασίας .

Sailfish Os : είναι ένα λειτουργικό για κινητά, με βάση τον πυρήνα του Linux, και αναπτύχθηκε από την Jolla Ltd. σε συνεργασία με το Mer project και την συμμαχία του Sailfish με σκοπό να αναπτυχθεί ένα λειτουργικό για τις επερχόμενες συσκευές Jolla(προφέρεται yolla), αλλά και για άλλες συσκευές που δεν έχουν ανακοινωθεί ακόμα. Η ομάδα της Jolla αποτελείται κυρίως από ανθρώπους της Nokia, και επίσης το Mer project χρησιμοποιήθηκε για την ανάπτυξη και διανομή του MeeGo (που είναι ο προκάτοχος του Tizen, και που προορίζονταν για συσκευές Nokia αρχικά) άρα μπορούμε να πούμε ότι το Sailfish είναι «αδερφάκι» του Tizen(με κάποιες βασικές διαφορές εντούτοις).

Εκτός από τον πυρήνα του Linux, χρησιμοποιήθηκε επίσης το Qt –ένα πλαίσιο εργασίας για ανάπτυξη εφαρμογών με γραφική διεπαφή χρήστη ,(GUI) που χρησιμοποιεί κυρίως γλώσσα C++ και έχει χρησιμοποιηθεί σε

πολλά άλλα λειτουργικά συστήματα για κινητές συσκευές. Αυτό σημαίνει ότι εφαρμογές από άλλα λειτουργικά μπορούν να εγκατασταθούν, και να «τρέξουν» σε Sailfish OS .

Οι εφαρμογές για το Sailfish Os αναπτύσσονται σε γλώσσα C++ καθώς και σε QML (που χρησιμοποιείται κυρίως για τη δημιουργία διεπαφής μέσω Javascript σε λειτουργικά για οθόνες αφής). Μπορούμε επομένως να πούμε ότι πρόκειται για ένα καθαρά native σύστημα από όλες τις απόψεις.



Εικόνα 9: Η Home screen του Jolla με το Sailfish

Το δυνατό σημείο του Jolla, ή αν θέλετε αυτό που διαφημίζουν ως κράχη της συσκευής είναι ότι δεν έχει κανένα πλήκτρο στην εμπρός πλευρά. Αντίθετα ο χειρισμός του γίνεται αποκλειστικά με χειρονομίες. Για παράδειγμα με swiipe από αριστερά προς τα δεξιά ή αντίστροφα πηγαίνεις στο Home, ενώ με swiipe από πάνω προς τα κάτω κλείνεις την τρέχουσα εφαρμογή. Αντίθετα με swiipe από κάτω προς τα πάνω βλέπεις τα μηνύματά σου. Το Jolla κυκλοφόρησε το Νοέμβριο του 2013 στην Φινλανδία και ενσωματώνει την τεχνολογία Here της Nokia για απεικόνιση χαρτών. Στον τομέα του hardware, το Jolla τρέχει σε διύρηνο επεξεργαστή Qualcomm στα 1.4GHz με 1GB RAM και 16GB flash αποθηκευτικού χώρου.

2.2 Συστήματα που χαρακτηρίζονται ως web :

Ως web θεωρούνται με πολύ γενικούς όρους, τα συστήματα και οι εφαρμογές τους κυρίως, που αναπτύσσονται σε γλώσσα υπερκειμένου (βλ HTML 5). Οι εφαρμογές αυτών των συστημάτων χρησιμοποιούν HTML5, Javascript, CSS με σκοπό τη δημιουργία ελαφρύτερων εφαρμογών με περισσότερες δυνατότητες ως προς το πολυμεσικό περιεχόμενο, μιας και κυρίως τέτοιου είδους εφαρμογές παράγονται με γλώσσες υπερκειμένου. Οι εφαρμογές αυτές προτιμώνται χάρη στην απανταχού παρουσία των φυλλομετρητών στα λειτουργικά των κινητών συσκευών, καθώς επίσης και στο γεγονός ότι δε φορτώνουν τον «πελάτη» με την συντήρηση και το επεξεργαστικό βάρος της εφαρμογής, μια κατάσταση γνωστή ως **thin client**, παρά μόνο «φορτώνουν» την εφαρμογή με τα

δεδομένα της στον browser του κινητού για όσο αυτό βρίσκεται συνδεδεμένο με ασύρματη σύνδεση,αποφεύγοντας έτσι updates, και λουπές επεξεργαστικές απαιτήσεις από τη κινητή συσκευή.

Αυτού του είδους ο τρόπος ανάπτυξης των εφαρμογών όσο τα web συστήματα γίνονται πιο διαδεδομένα - και βγαίνουν ολοένα περισσότερες πλατφόρμες ανάπτυξης -τείνει να εκλείψει κάπως, και έτσι έχει απομείνει μόνο το κομμάτι των πλεονεκτημάτων της χρήσης της HTML5. Τέτοια συστήματα είναι το **Tizen**, και το **Firefox OS** που αναλύονται παρακάτω εκτενώς.

2.3 Υβριδικά συστήματα (Hybrid) :

Υβριδικά συστήματα λέγονται εκείνα τα συστήματα που οι εφαρμογές τους μπορούν να αναπτυχθούν είτε σε γλώσσα HTML5/JavaScript (web εφαρμογές), αλλά και σε άλλη γλώσσα (C++ ,C) εξίσου. Δηλαδή μπορούν να υποστηρίξουν και native εφαρμογές,αλλά και web. Αυτά τα συστήματα συνήθως δίνουν τη δυνατότητα της επιλογής πλατφόρμας ανάπτυξης στους developers, μιας και είναι κάπως σπάνιο μια και μόνη πλατφόρμα ανάπτυξης να κάνει compile τόσο διαφορετικές γλώσσες.

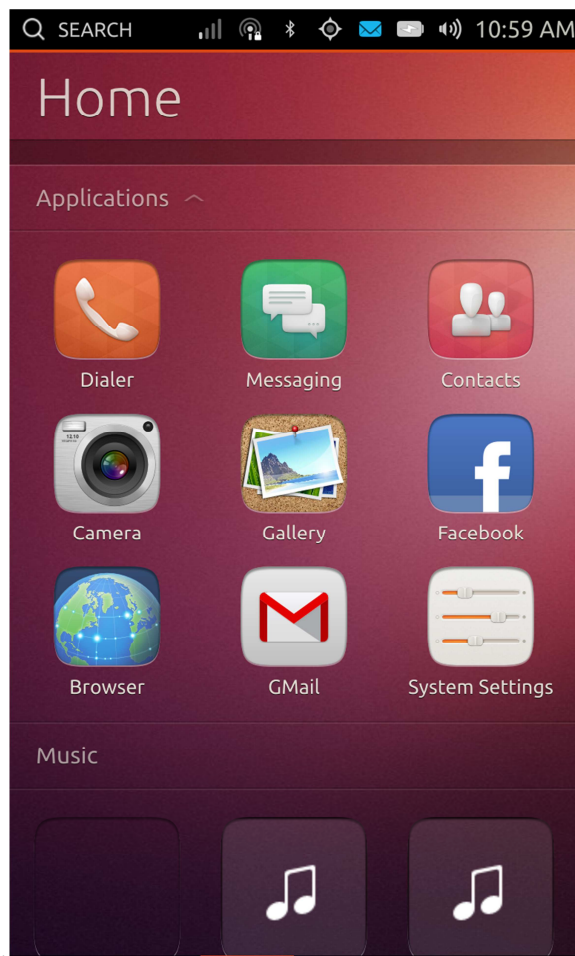
Ubuntu Touch: είναι η έκδοση του **Ubuntu** της Canonical Uk Ltd.που **προορίζεται για κινητές συσκευές δηλ smartphones,και tablets**. Το Ubuntu Touch είναι ανοιχτού κώδικα, το οποίο αρχικά είχε κυκλοφορήσει με το όνομα Ubuntu Mobile το 2008, αλλά η εξέλιξή του στάματησε μετά από απόφαση της Canonical με την υπόσχεση ότι θα αντικαθίσταντο σύντομα από άλλο, βελτιωμένο λειτουργικό. Το πλήρωμα του χρόνου ήρθε για τον Απρίλιο του 2014, οπότε και θα κυκλοφορήσει η κανονική έκδοση, μιας και η πρώτη alpha version κυκλοφόρησε τον Οκτώβριο του 2013.

Το Ubuntu Touch είναι λογισμικό ανοιχτού κώδικα, που χρησιμοποιεί τις ίδιες τεχνολογίες κορμού με το Ubuntu Desktop, επομένως οι εφαρμογές που αναπτύχθηκαν για το δεύτερο «τρέχουν» και στο πρώτο και αντίστροφα.Επιπλέον, το Ubuntu Touch συνοδεύουν εξαρτήματα και δυνατότητες του Ubuntu Desktop, επομένως μπορούμε να συνδέσουμε το πρώτο με ένα εξωτερικό μόνιτορ και να έχουμε μια πλήρη εμπειρία Ubuntu. Μάλιστα, οι συσκευές με το φορητό λειτουργικό, μπορούν αν συνδεθούν με docking station να μετατραπούν σε πλήρες λειτουργικό, και να παίζει ο χρήστης ακόμα και παιχνίδια μέσω αυτής της σύνδεσης .



Εικόνα 10:Το λογότυπο του Ubuntu

Από την έναρξη της λειτουργίας του, οι συσκευές με το Ubuntu Touch, φέρουν εφαρμογές κοινωνικής δικτύωσης και πολυμέσων (όπως το Facebook και το YouTube), αλλά και κάποιες εφαρμογές κορμού όπως κομπιουτεράκι, ξυπνητήρι, διαχείριση αρχείων, e-mail κλπ. Αξίζει να σημειωθεί ότι εφαρμογές για Ubuntu Touch μπορούν να αναπτυχθούν με γλώσσα C++, C αλλά και σε HTML5 και QML οπότε μπορούμε να πούμε ότι το Ubuntu Touch είναι ένα υβριδικό λειτουργικό με δυνατότητα ανάπτυξης web αλλά και native εφαρμογών.



Εικόνα 11: Η Home screen του Ubuntu Touch

Chrome OS: είναι ένα λειτουργικό που αναπτύχθηκε από τη Google, με βάση τον πυρήνα του Linux και με σκοπό να χρησιμοποιηθεί πρωτίστως για web εφαρμογές σε φορητές συσκευές και συγκεκριμένα σε

netbooks και tablets. Από αυτή την άποψη, θεωρείται ως web λειτουργικό σύστημα. Η διεπαφή του ακολουθεί μιμιμαλιστική προσέγγιση και αποτελείται σχεδόν ολοκληρωτικά από τον Google Chrome, και εφόσον σαν λειτουργικό στοχεύει στο κοινό που τον περισσότερο χρόνο παραμονής του όσον αφορά τη χρήση Η/Υ, τον περνά στο Internet και συγκεκριμένα στον Ιστό, οι μόνες γηγενείς εφαρμογές στο Chrome είναι ο ίδιος ο φυλλομετρητής, ο διαχειριστής αρχείων, και ο media player. Μιλάμε επομένως για ένα καθαρό **thin client** σύστημα .

Το λειτουργικό Chrome OS βασίζεται στο Chromium OS(το οποίο είναι η έκδοση του Chrome OS για σταθερούς Η/Υ και το οποίο θεωρείται ως προπομπός), και προορίζεται για να εγκατασταθεί σε συγκεκριμένες συσκευές από κατασκευαστές που συνεργάζεται η Google(και που «φοράνε» πλατφόρμες 32-bit-ες). Οι εφαρμογές για Chrome Os αναπτύσσονται σε γλώσσα HTML5/Javascript. Μπορούμε να πούμε ότι πρόκειται για ένα **υβριδικό** σύστημα από τη στιγμή που αναπτύσσεται για συγκεκριμένες συσκευές (άρα συγχρόνως native), και για συγκεκριμένη χρήση (καθαρά web).



Εικόνα 12: Η Home screen του Chrome Os

Το Chrome Os περιλαμβάνει έναν ενσωματωμένο media player στον chrome browser, ώστε να μπορούν οι χρήστες να αναπαράγουν mp3 αρχεία, καθώς και αρχεία JPEG ακόμα και όντας μη συνδεδεμένοι στο Internet. Επίσης, περιλαμβάνει έναν διαχειριστή αρχείων, που μοιάζει με αυτούς στα άλλα λειτουργικά, για την προσπέλαση των αρχείων και των φακέλων, καθώς και τα περιεχόμενα των φακέλων εφαρμογών όπως Google Docs και Box.net.

Αξίζει να σημειωθεί η σχέση των Chrome OS /Android : προέρχονται και τα 2 προϊόντα από την ίδια εταιρεία, τη Google ενώ προορίζονται για διαφορετικές συσκευές και είναι και τα 2 πετυχημένα open-source λειτουργικά. Η ίδια η Google πάντως μέσω του συν-ιδρυτή της δεν επέκλεισε το ενδεχόμενο συγχώνευσης των 2 λειτουργικών μελλοντικά.



Εικόνα 13:Το λογότυπο του Chrome Os

λειτουργικά συστήματα , όπως και η τεχνολογία χρήσης της γλώσσας υπερκειμένου, και αυτό γιατί είναι νέα και η ίδια η έκδοση της HTML που χρησιμοποιείται, σε συνδυασμό με CSS3 και Javascript, με κάποιες επιπλέον δυνατότητες που φαίνονται πολύ χρήσιμες σε εφαρμογές για κινητά περιβάλλοντα όπως :

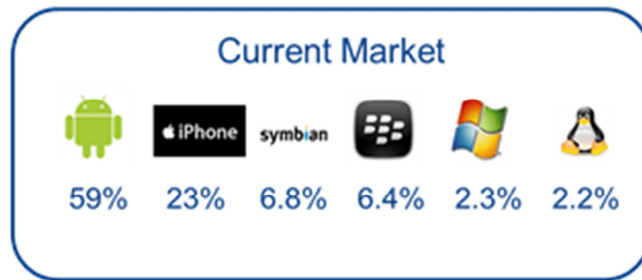
- Ετικέτα <canvas> για ζωγραφική 2-διαστάσεων. Αυτή η προσθήκη στις δυνατότητες της HTML δίνει την δυνατότητα να επεξεργαστούμε έτοιμη εικόνα, αλλά και να ζωγραφίσουμε .
- Ετικέτες <video> ,<audio> για έλεγχο και αναπαραγωγή οπτικοακουστικού υλικού σε σελίδες ή εφαρμογές χωρίς την χρήση Adobe Flash ή Microsoft Silverlight που θα βάραιναν υπερβολικά με την εγκατάστασή τους μια κινητή συσκευή.
- Υποστήριξη τοπικής αποθήκευσης δεδομένων. Αυτή η δυνατότητα είναι πάρα πολύ ενδιαφέρουσα καθώς δίνει τη δυνατότητα σε κινούμενους χρήστες(με διακόπτομενο σήμα σύνδεσης ενδεχομένως) να αξιοποιήσουν δυνατότητες εφαρμογών ακόμα και χωρίς να είναι συνδεδεμένοι.
- Νέες ετικέτες περιεχομένου όπως <article>, <footer>, <header>, <nav>, < section> για μορφοποίηση του κειμένου των εφαρμογών ώστε οι εφαρμογές να είναι εύκολες στην πλοήγηση από κινητή συσκευή, και χρήσιμες ακόμα και σε άτομα με προβλήματα όρασης.
- Έλεγχος σε κάποια στοιχεία που δεν υπήρξε πριν όπως το ημερολόγιο, ώρα, ημερομηνία, αναζήτηση, url, email..
- Εντοπισμός γεωγραφικής θέσης: αποτελεί χρησιμότερο εργαλείο για εφαρμογές πληροφόρησης και πλοήγησης.
- Εξέλιξη των φορμών: Η HTML5 δίνει τη δυνατότητα στους web developers να πάνε τις φόρμες τους ένα βήμα παραπάνω, προσθέτοντας ιδιότητες και στοιχεία που πριν δεν υπήρχαν.

Ο παρακάτω πίνακας δείχνει τις συμβατότητες μεταξύ των δυνατοτήτων της HTML5 και των φυλλομετρητών, μιας και ως νέα τεχνολογία, η HTML5 για φορητές συσκευές, αντιμετωπίζει κι αυτή δυσκολίες στην ενσωμάτωσή της :

Πίνακας 1: Η HTML5 και η συμβατότητα με διάφορους φυλλομετρητές σε διαφορετικά περιβάλλοντα

Feature	Safari on iOS	Android Browser		BlackBerry Browser		Internet Explorer	Opera		Firefox	webOS Browser	Symbian Browser
	iPhone, iPad	Phones (1.0-2.3)	Tablets (3.0+)	Phones	Tablet	Windows Phone	Mobile	Mini	Android		Nokia phones
Version tested											
Minimum version tested	3.2	1.5	3.0	5.0	1.0	9	11	5	6	1.4	^3
Application Cache W3C API Offline package installation	✓	✓ 2.1+	✓	✓ 6.0+	✓		✓		✓	✓	
Web storage W3C API Persistent and session storage	✓	✓ 2.0+	✓	✓ 6.0+	✓	✓	✓		✓	✓	
Web SQL storage W3C API (no active) Persistent SQLite storage	✓	✓ 2.0+	✓	✓ 6.0+	✓		✓			✓	
Geolocation W3C API Geolocation & tracking using GPS, cells or Wi-Fi	✓	✓ 2.0+	✓	✓ 6.0+	✓	✓	✓		✓	✓	
Multimedia W3C API Video & Audio Playback	✓	✓ 2.3+	✓	✓ 7.0+	✓	✓	✓		✓	✓	
Server-Sent Events W3C API EventSource pattern to maintain the connection to the server open	✓ 4.1+						✓		✓		
Web Sockets W3C API Bidirectional protocol over HTTP	✓ 4.2+			✓ 6.1+	✓		✓		✓		

Αυτή τη στιγμή η αγορά της κινητής τηλεφωνίας έχει να επιδείξει μόνο δυο μεγάλα λειτουργικά συστήματα-επιτεύγματα: Android της Google και iOS της Apple. Το μεν Android κατέχει περίπου το 60% της αγοράς, ενώ το iOS μόνο το 25% περίπου όπως φαίνεται και στην εικόνα 2.



Εικόνα 15: Μερίδιο αγοράς

Βλέπουμε και στην εικόνα 15 ότι κυριαρχούν 2 επιλογές, με τη μία να αποτελεί πολύ ακριβή για τη μάζα του κοινού που ενδιαφέρεται για την τεχνολογία, βλ iOS και επιπρόσθετα το λειτουργικό της Apple δε τρέχει σε άλλο hardware πέραν αυτού της Apple, και την άλλη να αποτελεί μονόδρομο, βλ Android και σχεδόν μονοπώλιο, αφού συνεργάζεται σχεδόν με όλες τις μάρκες κινητών τηλεφώνων.

Αυτό που κάνει την HTML5 να ξεχωρίζει, είναι η δυνατότητα που δίνει στους χρήστες να αναπτύξουν τις δικές τους εφαρμογές με μεγαλύτερη ευκολία, αισθητά μειωμένο χρόνο συγγραφής κώδικα σε σχέση με τα ήδη

υπάρχοντα εργαλεία για άλλα λειτουργικά συστήματα για κινητές συσκευές (βλ. Android), το οποίο δίνει τη δυνατότητα σε περισσότερους χρήστες να αναπτύξουν τις δικές τους εφαρμογές με βάση τις ανάγκες τους. Η HTML5 και η Javascript είναι γλώσσες που θεωρούνται πιο εύκολες για το ευρύ κοινό σε σχέση με Java (βλ. Android), C++ κλπ.

3.2 Η ετικέτα <canvas > ως στοιχείο της HTML5 :

Το στοιχείο του καμβά αποτελεί μέρος της HTML5 και επιτρέπει τη δυναμική επεξεργασία και απεικόνιση σχημάτων και εικόνων bitmap 2 διαστάσεων. Ο καμβάς αποτελείται από μια σχεδιάσιμη περιοχή που ορίζεται στον κώδικα της HTML με χαρακτηριστικά το ύψος και το πλάτος. Κώδικας Javascript μπορεί να γραφεί για πρόσβαση στην περιοχή μέσα από ένα πλήρες σύνολο σχεδίασης με λειτουργίες παρόμοιες με εκείνες των άλλων κοινών διαδικαστών APIs, επιτρέποντας έτσι να δημιουργούνται δυναμικά γραφικά. Μερικές προβλεπόμενες χρήσεις του καμβά περιλαμβάνουν παιχνίδια, γραφικά κτιρίων, καθώς και επεξεργασία εικόνων που εισάγονται από το χρήστη είτε μέσω κάμερας είτε από μέσο αποθήκευσης. Ο καμβάς είναι ένα στοιχείο της HTML5 όπως όλα τα άλλα (πχ , <a> κλπ) πράγμα που σημαίνει ότι μπορούμε να έχουμε πάνω από έναν σε μία εφαρμογή ή σελίδα. Αρχικά δεν έχει ούτε περιεχόμενο ούτε περιθώριο, εμείς του δίνουμε αυτές τις ιδιότητες μέσα από την HTML ή τη CSS. Απαιτείται πρώτα να οριστεί ένας καμβάς με συγκεκριμένες διαστάσεις που θα «φιλοξενεί» την εκάστοτε εικόνα που θα επιλέξει ο χρήστης. Σε αυτό τον καμβά θα τοποθετηθεί η εικόνα μέσω κώδικα Javascript που θα την προσαρμόζει μεν στις διαστάσεις του καμβά, χωρίς να την παραμορφώνει δε. Ο καμβάς «χτίζεται» αρχικά σε γλώσσα HTML5 μέσω της ετικέτας <canvas> με ιδιότητες το πλάτος(width), το ύψος (height) και την ταυτότητα (id), στοιχείο απαραίτητο για την διαχείρισή του μέσω της Javascript. Οι default διαστάσεις, δηλαδή αυτές που είναι προ-ρυθμισμένες , είναι 300*150 pixels.

Από τη στιγμή που δίνουμε αυτές τις ιδιότητες στο στοιχείο αυτό της html, με σημαντικότερη όλων την “id”, έχουμε στη συνέχεια τη δυνατότητα να του «φορτώσουμε» εικόνα (προεπιλεγμένη ή δυναμικά επιλεγμένη από το χρήστη) ή να τη συνδέσουμε με κάποιο εξωτερικό μέσο εισαγωγής εικόνας όπως κάμερα ή να ζωγραφίσουμε μέσα, και όλα αυτά μέσω κώδικα Javascript και των βιβλιοθηκών της. Χωρίς τη χρήση Javascript, η HTML5 από μόνη της δε μας δίνει τη δυνατότητα να διαχειριστούμε τον καμβά με οποιοδήποτε τρόπο.

3.3 Οι ετικέτες < video > και <audio>

Μέχρι πριν την εμφάνιση της HTML5, για να αναπαραχθεί βίντεο σε μια σελίδα Ιστού, χρειαζόταν απαραίτητως πρόσθετη εγκατάσταση στον φυλλομετρητή (δηλ κάποιο plug-in). Μάλιστα, διαφορετικοί φυλλομετρητές, υποστήριζαν την αναπαραγωγή βίντεο με διαφορετικά plug-ins. Αυτό σημαίνει πως κάποιος που χρησιμοποιεί διαφορετικούς φυλλομετρητές, να πρέπει να εγκαταστήσει και διαφορετικά plug-ins για κάθε φυλλομετρητή πράγμα που θα αποτελούσε επιπλέον επιβάρυνση για τον Η/Υ του, αλλά και πρόσθετη δυσκολία σε περίπτωση που είχε κάποια έκδοση παλαιότερη εγκατεστημένη. Η HTML5 δίνει τη δυνατότητα μέσω του στοιχείου <video> να προκαθορίσει έναν τρόπο για να ενσωματώσει ένα video σε μια ιστοσελίδα χωρίς να χρειάζεται η εγκατάσταση plug-in.

Η ετικέτα <video> προσθέτει στοιχεία ελέγχου στο βίντεο, όπως παύση, αναπαραγωγή, ένταση ήχου. Κρίνεται σκόπιμο επίσης να καθορίζονται οι διαστάσεις του βίντεο, για να είναι καθορισμένες κατά τη φόρτωση της σελίδας. Ειδάλλως, χωρίς αυτές τις διαστάσεις, ο φυλλομετρητής δεν γνωρίζει το μέγεθος του βίντεο προς αναπαραγωγή, κι έτσι δε γνωρίζει πόσο χώρο να κρατήσει γι' αυτό το σκοπό. Αυτό μπορεί να έχει σαν αποτέλεσμα να αλλάξει όλη η διάταξη των στοιχείων της σελίδας κατά τη φόρτωση του βίντεο. Τα στοιχεία αυτά, δίνονται με την ετικέτα, και έχει την εξής μορφή:








```
<video width="320" height="240" controls>
  <source src="movie.mp4" type="video/mp4">
  <source src="movie.ogg" type="video/ogg">
  Your browser does not support the video tag.
</video>
```

Εικόνα 16 :Κώδικας της HTML5 για την ενσωμάτωση video

Υπάρχουν 3 τύποι αρχείων βίντεο που υποστηρίζει η HTML5 :

- MP4 :MPEG 4 αρχεία με κωδικοποίηση ήχου AAC
- WebM: WebM αρχεία με VP8 κωδικοποίηση βίντεο και κωδικοποίηση ήχου Vorbis
- Ogg :Ogg αρχεία με Theora κωδικοποίηση βίντεο και κωδικοποίηση ήχου Vorbis

Πίνακας 2: οι αντιστοιχίες φυλλομετρητών με τις τεχνολογίες που υποστηρίζουν

CODEC/CONTAINER							
Theora+Vorbis+Ogg	.	3.5+	.	5.0+	10.5+	.	.
H.264+AAC+MP4	.	.	3.0+	5.0+	.	3.0+	2.0+
WebM	.	.	.	6.0+	10.6+	.	.

3.4 Η τοπική αποθήκευση δεδομένων στην HTML5 (Web Storage)

Στην HTML5 είναι δυνατόν να αποθηκευτούν δεδομένα τοπικά μέσω του φυλλομετρητή του χρήστη. Μέχρι πρότινος, αυτό ήταν δυνατό μέσω των cookies. Παρ'όλα αυτά, η Web Storage τεχνολογία είναι πιο ασφαλής και πιο γρήγορη από τα cookies. Αυτό συμβαίνει γιατί τα δεδομένα δεν περιλαμβάνονται με κάθε αίτημα προς τον server, αλλά αποστέλλονται από το server μόνο όταν αιτηθούν. Είναι δυνατόν να αποθηκευτούν μεγάλες ποσότητες δεδομένων χωρίς να επηρεαστεί η απόδοση της ιστοσελίδας. Τα δεδομένα αποθηκεύονται σε ζεύγη

ονόματος/τιμής, και μια ιστοσελίδα μπορεί να έχει πρόσβαση σε δεδομένα που έχει αποθηκεύσει η ίδια και μόνο. Σε αντίθεση με την τεχνολογία των cookies, το όριο αποθήκευσης δεδομένων είναι πολύ μεγαλύτερο, και οι πληροφορίες ποτέ δεν μεταφέρονται πίσω προς το server, πράγμα που θα επιβάρυνε την επικοινωνία με άχρηστη πληροφορία που αποστέλλεται με κάθε μήνυμα. Η τεχνολογία Web Storage υποστηρίζεται από τους εξής φυλλομετρητές: Internet Explorer (από την έκδοση 8 και μετά), Firefox, Opera, Chrome, and Safari.










Η τοπική αποθήκευση μπορεί να επιτευχθεί προγραμματιστικά μέσω 2 νέων αντικειμένων :

- window.localStorage- για αποθήκευση χωρίς ημερομηνία λήξης
- code.sessionStorage- για αποθήκευση δεδομένων όσο διαρκεί η σύννοδος (δηλ μέχρι να κλείσει η καρτέλα του φυλλομετρητή).

Πρόκειται για χρησιμότητα μέσο ανάπτυξης αλληλεπιδραστικών εφαρμογών, και για εφαρμογές κινητές όπου μπορεί να χρειάζεται να δουλέψει κανείς offline λόγω χαμηλού σήματος ή λόγω συνεχούς κίνησης. Απαιτείται βεβαίως πρώτα κάποιος έλεγχος προκειμένου να σιγουρευτούμε ότι ο εκάστοτε φυλλομετρητής υποστηρίζει τοπική αποθήκευση δεδομένων.

Πίνακας 3: οι εκδόσεις των δημοφιλέστερων φυλλομετρητών που υποστηρίζουν τοπική αποθήκευση δεδομένων

Browser Support

									
Web Storage -	4+	3.5+	4+	10.5+	8+	3.2+	2.1+	—	11.5+

3.5 Ο γεωγραφικός εντοπισμός της θέσης του χρήστη ως εργαλείο της HTML5

Η HTML5 δίνει τη δυνατότητα μέσω της μεθόδου `getCurrentPosition()` της Javascript, και κατόπιν άδειας του χρήστη (ώστε να μη γίνεται παραβίαση της ιδιωτικότητας), να γίνει εντοπισμός της θέσης του χρήστη. Η πιο απλή χρήση του Geolocation API μοιάζει κάπως έτσι :

```
function get_location() {
    navigator.geolocation.getCurrentPosition(show_map);
}
```

Εδώ δεν έχουμε βέβαια ούτε ανίχνευση θέσης, ούτε διαχείριση σφάλματος, ούτε ιδιαίτερες επιλογές. Εξαρτάται από το browser του χρήστη, στις περισσότερες περιπτώσεις όμως, ο φυλλομετρητής δίνει τη δυνατότητα ενημέρωσης περί την ταυτότητα της ιστοσελίδας που ζητά τον εντοπισμό της θέσης, καθώς επίσης δίνει τη δυνατότητα αποθήκευσης της προτίμησης του χρήστη για την αποδοχή ή όχι του εντοπισμού της θέσης. Στην παρακάτω εικόνα υπάρχει ένα ενδεικτικό info-bar της ερώτησης που εμφανίζεται στο χρήστη :








Εικόνα 17 :Info-bar για την ενημέρωση και αδειοδότηση από το χρήστη

Αυτό το info-bar έχει κάποιες ιδιότητες συγκεκριμένες που προστατεύουν αλλά και διευκολύνουν το χρήστη :

- μπλοκάρει την ιστοσελίδα ώστε να μη μπορεί να κάνει εντοπισμό μέχρι ο χρήστης να επιλέξει με την απάντησή του αν επιθυμεί ή όχι εντοπισμό,
- παρέχει άνευ- όρων δυνατότητα προσωρινού αποκλεισμού προς τη μεριά της ιστοσελίδας, έτσι ώστε αυτή να μην μπορεί να τον προσπεράσει με κανένα τρόπο, κανένα προγραμματιστικό κόλπο για παράδειγμα.
- Δεν εμποδίζει το χρήστη να μεταβεί σε άλλη καρτέλα, για όσο εκείνος επιθυμεί, και σε όλο αυτό το διάστημα η μπάρα αναμένει εμφανιζόμενη μόνο στην καρτέλα που «τρέχει» τον εντοπισμό και όχι σε όλες.

Όταν υπάρχει δε και ενσωματωμένο GPS στη συσκευή που «τρέχουμε» την εφαρμογή, τότε ο εντοπισμός θέσης είναι ακόμη ακριβέστερος, σε σχέση με τον εντοπισμό μέσω συσκευών που δεν έχουν ενσωματωμένο GPS. Στην παρακάτω εικόνα φαίνεται μέσα από ένα πλήθος δημοφιλών φυλλομετρητών, ποιοι υποστηρίζουν και ποιοι όχι αυτή την δυνατότητα της HTML5 :

Desktop Support		Mobile Support	
Browser	Supported Version	Browser	Supported Version
	9.0+	Android	2.0+
	5.0+	iPhone	3.0+
	10.60+	Symbian	S60v3,s60v5
	3.5+		
	5.0+		

Εικόνα 18: Εκδόσεις φυλλομετρητών που υποστηρίζουν τον γεωγραφικό εντοπισμό

Αφού ο χρήστης επιλέξει να μοιραστεί τη θέση του, στην πορεία μπορεί να γίνει εμφάνιση μέσω χάρτη των συντεταγμένων του χρήστη, αλλά γενικά είναι μια πολύπλοκη διαδικασία που μεσολαμβάνει διάφορες διεργασίες όπως επανάκληση συναρτήσεων, επιστροφή συντεταγμένων με τη μορφή τιμών, σύνδεση με δορυφόρο σε περίπτωση συσκευής με ενσωματωμένο GPS κλπ και γι' αυτό το λόγο προγραμματιστικά χρησιμοποιούνται τεχνικές για διαχείριση σφάλματος μαζί με τον εντοπισμό θέσης.

3.6 Η εξέλιξη των φορμών στην HTML5

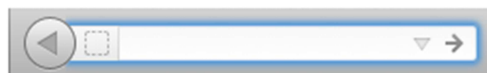
Οι φόρμες στην HTML5 έχουν πολλές πρόσθετες και χρήσιμες δυνατότητες όπως για παράδειγμα αυτήν να επικυρώνονται αυτόματα από το φυλλομετρητή, ώστε να αποφεύγονται οι πολλαπλές επικυρώσεις δεδομένων προκειμένου να υποβληθεί μια φόρμα με στοιχεία χρήστη, αλλά και άλλες. Παρακάτω αναλύονται όλες μία προς μία :

3.6.1 Κείμενο μέσα σε πεδίο φόρμας

Η πρώτη βελτίωση που προσέφερε η HTML5 στις φόρμες είναι η δυνατότητα τοποθέτησης κειμένου μέσα στο πεδίο εισαγωγής που φτιάχνουμε με την ετικέτα <input>. Το κείμενο εμφανίζεται μέσα στο πεδίο εισαγωγής, για όσο διάστημα το πεδίο εισαγωγής είναι κενό (δηλαδή όσο ο χρήστης δεν πληκτρολογεί κείμενο μέσα σε αυτό). Όταν κάνει κλικ μέσα σε αυτό για να εισάγει τιμές, τότε το κείμενο εξαφανίζεται. Στην παρακάτω εικόνα φαίνεται το πεδίο μιας φόρμας με κείμενο πριν το κλικ του χρήστη, και το ίδιο πεδίο μετά το κλικ :



Εικόνα 19 :Το πεδίο εισαγωγής της φόρμας πριν το κλικ του χρήστη



Εικόνα 20:Το πεδίο εισαγωγής της φόρμας μετά το κλικ του χρήστη

Για να εισάγουμε κείμενο μέσα στα πεδία της φόρμας, αρκεί μια έξτρα ιδιότητα στην ετικέτα `<input>`:

```
<form>
```

```
<input name="q" placeholder="Go to a Website">
```

```
<input type="submit" value="Search">
```

```
</form>
```

Υπάρχουν φυλλομετρητές που δεν υποστηρίζουν την ιδιότητα αυτή. Σε αυτή την περίπτωση, απλώς θα αγνοήσουν αυτή την έξτρα ιδιότητα που βάλουμε στην ετικέτα, χωρίς άλλα ιδιαίτερα προβλήματα. Να σημειωθεί ότι δε μπορούμε να εισάγουμε άλλου είδους στοιχεία στα πεδία, πέραν του κειμένου (όπως εικόνα για παράδειγμα). Παρ' όλα αυτά, μπορούμε να μορφοποιήσουμε το κείμενο, μέσω ιδιοτήτων της CSS.

3.6.2 Αυτόματη εστίαση στο εκάστοτε επιλεγμένο πεδίο

Η δεύτερη βελτίωση που προσέφερε η HTML5, όσον αφορά τις φόρμες, είναι η ιδιότητα της αυτόματης εστίασης. Μέχρι πρότινος, όταν η σελίδα εστίαζε αυτόματα στο πρώτο πεδίο της φόρμας, η πληκτρολόγηση γίνεται αυτόματα σε εκείνο το πεδίο, όπως συμβαίνει για παράδειγμα στην σελίδα του Google.com. Αυτό μπορεί για κάποιους χρήστες να είναι εξυπηρετικό, όπως στην περίπτωση του Google.com, γιατί η αναζήτηση γίνεται πιο γρήγορα, με τη φόρτωση της σελίδας πληκτρολογούν την αναζήτηση που τους ενδιαφέρει, και όλα αυτά γίνονταν μέσω Javascript, JQuery, και Flash. Η HTML5 λοιπόν, εισάγει μέσω μόνο μιας έξτρα ιδιότητας τη δυνατότητα να προσθέσουμε αυτόματη εστίαση, χωρίς κλήσεις συναρτήσεων. Αυτή η ιδιότητα είναι η `autofocus`, και μπορεί να μπει σε ετικέτες όπως η `<input>`, `<textarea>`, `<button>`. Για παράδειγμα :

```
<input autofocus="autofocus" />
```

```
<button autofocus="autofocus"></button>
```

```
<textarea autofocus="autofocus"></textarea>
```

3.6.3 Πρόσθετες ιδιότητες σε ετικέτες φόρμας

Μέχρι πρότινος, η HTML λειτουργούσε με ορισμένες μόνο ιδιότητες στην ετικέτα <input> όπως radio, checkbox, password κ.ά. Τώρα έχουν προστεθεί άλλες 13 νέες ιδιότητες, που αφορούν την τιμή που θα πληκτρολογηθεί μέσα στα πεδία που θα προστεθεί. Αυτές είναι

- **Color** : Δέχεται ως τιμή ένα χρώμα που το επιλέγει ο χρήστης από ένα μενού-χρωματολόγιο, και λαμβάνει προς αποστολή στο server την δεκαεξαδική τιμή που αντιστοιχεί σ' αυτό το χρώμα. Υποστηρίζεται από Google Chrome, Opera browsers.
- **Date** :Δέχεται ως τιμή μια ημερομηνία που επιλέγει ο χρήστης από ένα αναδυόμενο ημερολόγιο, και λαμβάνει προς αποστολή στο server την τιμή που επιλέχθηκε με τη μορφή έτος-μήνας-ημέρα. Υποστηρίζεται από Google Chrome, Opera, Safari browsers.
- **Datetime** :Όπως και η παραπάνω ιδιότητα, δέχεται ως τιμή την ημερομηνία και τη λαμβάνει προς αποστολή στο server με την ίδια μορφή, μόνο που μαζί αποστέλλει και την ώρα. Υποστηρίζεται από Opera, Safari browsers.
- **Datetime-local**: :Όπως και η παραπάνω ιδιότητα, δέχεται και λαμβάνει προς αποστολή στο server ώρα και ημερομηνία συν το προ και μετά μεσημβρίας. Υποστηρίζεται από Google Chrome, Opera, Safari browsers.
- **Email** :Δέχεται και λαμβάνει προς αποστολή στο server ως τιμή ένα email. Υποστηρίζεται από Internet Explorer, Mozilla Firefox, Google Chrome, Opera browsers. Ο Safari μάλιστα σε iPhone, αναγνωρίζει το είδος της τιμής που θα εισαχθεί αυτόματα(δηλ ανεξάρτητα από την έκδοση της HTML που έχει συνταχθεί η φόρμα), με αποτέλεσμα να αλλάζει το πληκτρολόγιο εισάγοντας σε νέα πλήκτρα το @ και την τελεία και μικραίνοντας το space.
- **Month** : Δέχεται ως είσοδο και λαμβάνει προς αποστολή στο server μήνα και έτος. Υποστηρίζεται από Opera, Google Chrome, Safari browsers.
- **Number**: Δέχεται ως είσοδο και λαμβάνει προς αποστολή στο server έναν αριθμό. Παίρνει ως πρόσθετες ιδιότητες τα min και max .Υποστηρίζεται από Internet Explorer, Google Chrome, Opera, Safari browsers.
- **Range**: Δέχεται ως είσοδο ένα αριθμό από ένα εύρος τιμών, και με ένα βήμα συγκεκριμένο ενδεχομένως, που καθορίζεται από τις πρόσθετες ιδιότητες min /max και step που θα ορίσουμε στην ετικέτα <input>και αποστέλλει αυτόν τον αριθμό στο server. Εδώ να διευκρινίσουμε ότι στη φόρμα εμφανίζεται μπάρα κύλισης οριζόντια για να απεικονίσει το εύρος. Υποστηρίζεται από Internet Explorer, Google Chrome, Mozilla Firefox, Opera, Safari browsers.
- **Search** :Δέχεται ως είσοδο λέξεις-κλειδιά προς αποστολή στο server και αναζήτηση, αλλά μόνο αν αυτά είναι σε κωδικοποίηση Unicode-8 (δηλαδή δεν δέχεται ελληνικούς χαρακτήρες).Υποστηρίζεται από Google Chrome, Safari browsers.

- **Tel:** Δέχεται ως είσοδο αριθμό τηλεφώνου, μόνο που αυτή την ιδιότητα, δε την υποστηρίζει ακόμη κανείς browser.
- **Time :** Δέχεται και λαμβάνει προς αποστολή στο server ώρα και ημερομηνία συν το προ και μετά μεσημβρίας όπως και η Datetime πιο πάνω. Υποστηρίζεται από Google Chrome, Safari, Opera browsers.
- **url :** Δέχεται ως είσοδο στη φόρμα μια διεύθυνση Ιστού. Η διεύθυνση αυτή, που αποτελεί και την τιμή που αποστέλλεται στο server, επικυρώνεται αυτόματα από την ίδια την σελίδα που βρίσκεται μέσω της HTML5, για να αποφεύγονται περιττά μηνύματα και επιβεβαιώσεις ,που περιέχουν επαναλαμβανόμενο περιεχόμενο. Υποστηρίζεται από Internet Explorer, Mozilla Firefox, Google Chrome, Opera.Αξίζει να σημειωθεί και εδώ ότι ο Safari browser αναγνωρίζει ανεξάρτητα από την έκδοση της HTML το είδος της τιμής που θα εισαχθεί, και αλλάζει το πληκτρολόγιο , προσθέτοντας ένα πλήκτρο .com .
- **Week :**Δέχεται ως είσοδο εβδομάδα και έτος, από ένα αναδυόμενο μενού πάνω στο πεδίο της φόρμας. Υποστηρίζεται από Google Chrome, Safari, Opera browsers.

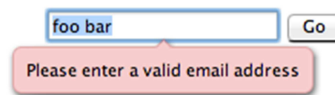
Αξίζει να σημειωθεί ότι στο πεδίο της φόρμας, σε όλα τα παραπάνω πεδία, όταν πληκτρολογεί ο χρήστης, εμφανίζεται δεξιά στο πεδίο ένα x που αποσκοπεί στην διαγραφή των λέξεων που πληκτρολογούνται με μία κίνηση ,και αυτό χάρη στην HTML5.

3.6.4 Επικύρωση φόρμας

Κατά την συμπλήρωση μιας φόρμας, απαιτείται αυτόματη επικύρωση των εισαγομένων δεδομένων. Μπορεί να είναι μέσω Javascript, είτε μέσω PHP server, και επικύρωση από το server σίγουρα. Μπορεί και όλες αυτές οι επικυρώσεις να είναι απαραίτητες. Η HTML5 μπορεί να μη μπορεί να αντικαταστήσει την επικύρωση από το server, αλλά υπάρχει η δυνατότητα να αντικαταστήσει μελλοντικά την επικύρωση από τη Javascript. Και αυτό, γιατί η επικύρωση όσον αφορά την Javascript εγκυμονεί 2 κινδύνους :

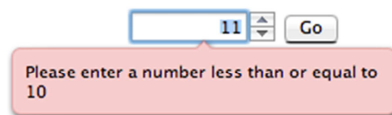
- Μεγάλος αριθμός των επισκεπτών της εκάστοτε σελίδας να μην έχουν ενεργοποιημένη την Javascript – ακούγεται απίθανο,αλλά κυμαίνεται στο 10%
- Να παρουσιαστεί σφάλμα κατά τον έλεγχο. Αυτό είναι ένα πολύ πιθανό σενάριο, δεδομένου ότι όλες οι συμβολοσειρές που μπορεί να εισαχθούν σε μια φόρμα είναι εν δυνάμει διευθύνσεις ηλ.ταχυδρομείου, ή αντίθετα κάθε τιμή που εισάγεται σαν e-mail, μπορεί να είναι μια οποιαδήποτε συμβολοσειρά, και ειδικά δε η επικύρωση της ως έγκυρη, είναι πολύ πολύπλοκη διαδικασία.

Παρακάτω φαίνεται ένα στιγμιότυπο από αυτόματη επικύρωση φόρμας e-mail, δανεισμένο από Opera browser, συγκεκριμένα την 11, παρόλο που η δυνατότητα αυτή είναι παρούσα από την 9^η έκδοση ακόμα. Το μόνο που απαιτείται είναι η ιδιότητα type='email' στην ετικέτα <input>. Ακόμη και στην περίπτωση που ο χρήστης δεν έχει ενεργοποιημένα script από Javascript, η αυτόματη επικύρωση από την HTML5 θα λειτουργήσει.



Εικόνα 21 : αυτόματη επικύρωση φόρμας ,χωρίς να προηγηθεί «Submit»

Το ίδιο ισχύει με την αυτόματη επικύρωση διεύθυνσης Ιστού, δηλαδή με την ιδιότητα `type='url'`, καθώς επίσης και με τους αριθμούς και την ιδιότητα `type='number'`. Γίνεται αυτόματη επικύρωση της διεύθυνσης, και μάλιστα στην περίπτωση των αριθμών, στην επικύρωση λαμβάνονται υπόψη και οι ιδιότητες `min/ max`, οπότε ο φυλλομετρητής δε θα κάνει επικύρωση της φόρμας αν εισαχθεί αριθμός με μεγαλύτερη τιμή από αυτήν που έχει δηλωθεί ως `max`. Παρακάτω φαίνεται στην εικόνα η άρνηση του φυλλομετρητή να δεχθεί την τιμή που εισάγεται, καθώς αυτή ξεπερνά την `max` ιδιότητα που έχει οριστεί.



Εικόνα 22: άρνηση του φυλλομετρητή να δεχθεί την τιμή που εισάγεται

Η ενεργοποίηση της δυνατότητας αυτής της HTML5, δηλαδή της αυτόματης επικύρωσης φόρμας, δεν απαιτεί κάποια πρόσθετη ετικέτα, είναι προκαθορισμένη και λειτουργεί εξ'ορισμού. Αν πάλι θέλουμε να μην λειτουργεί, τότε μπορούμε να την απενεργοποιήσουμε, με την ιδιότητα `novalidate` όπως φαίνεται παρακάτω :

```
<form novalidate>
  <input type="email" id="addr">
  <input type="submit" value="Subscribe">
</form>
```

3.6.5 Απαιτούμενα πεδία

Η επικύρωση των φορμών στην HTML5 δεν περιορίζεται στους τύπους των πεδίων, αλλά αφορά και το κάθε πεδίο ξεχωριστά. Δηλαδή μπορούμε για κάθε πεδίο της φόρμας ξεχωριστά, να ορίσουμε ποια πεδία είναι υποχρεωτικά με την προσθήκη μιας ιδιότητας μόνο, η οποία σε κάθε πεδίο χρησιμοποιείται χωριστά:

```
<form>
  <input id="q" required>
  <input type="submit" value="Search">
```

</form>

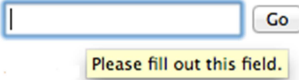
Αν επιχειρηθεί η επικύρωση της φόρμας χωρίς να έχουν συμπληρωθεί αυτά τα πεδία, τότε ο φυλλομετρητής θα εμφανίσει ένα αναδυόμενο παράθυρο (pop-up) όπως φαίνεται στην εικόνα :



The image shows a browser validation error dialog box. At the top, it says "HTML5 required form fields not completed" in blue text. Below this, there are two input fields. The first is labeled "Name:" and is empty. The second is labeled "Address:" and contains the text "Please fill out this field." in a tooltip. At the bottom of the dialog is a blue "Submit" button.

Εικόνα 23:Ο φυλλομετρητής δεν δέχεται τα υποχρεωτικά πεδία να υποβληθούν κενά

Επιπρόσθετα, κάποιοι φυλλομετρητές, μόνο με αιώρηση του δείκτη του ποντικιού πάνω από το πεδίο, μπορεί να εμφανίσουν την πληροφορία ότι το συγκεκριμένο πεδίο είναι υποχρεωτικό, για παράδειγμα η παρακάτω εικόνα που δείχνει αυτό ακριβώς, είναι «δανεισμένη» από τον Mozilla Firefox :



The image shows a single input field with a blue border. To its right is a "Go" button. Below the input field, a yellow tooltip box contains the text "Please fill out this field."

Εικόνα 24 :Αιώρηση του δείκτη του ποντικιού μπορεί να εμφανίσει πληροφορία για απαιτούμενα πεδία

Κεφάλαιο 4ο-Το Tizen

4.1 Εισαγωγή

Το νέο λειτουργικό σύστημα **Tizen** προορίζεται για κινητές συσκευές, καθώς και η αρχιτεκτονική του. Επίσης αναπτύσσεται μια εφαρμογή σε περιβάλλον πλατφόρμας Tizen IDE με γενικότερο στόχο την ομαλή λειτουργία του σε συσκευή με το προαναφερθέν λειτουργικό σύστημα. Όπως φαίνεται στην εικόνα το Tizen έχει ως λογότυπο έναν τροχό. Αυτό το απλό, παγκοσμίως γνωστό αντικείμενο συμβολίζει όλα όσα το Tizen εκπροσωπεί : την απλότητα στη χρήση, την ανθρώπινη δημιουργία μιας και ο οποιοσδήποτε μπορεί να το φτιάξει με λίγο χαρτί (τονίζεται με αυτό η ιδιότητα του, του ανοιχτού κώδικα), και την δυνατότητα να λειτουργήσει με την πιο απλή ανθρώπινη αλληλεπίδραση .



Εικόνα 25: Λογότυπο του Tizen

Το Tizen παρουσιάστηκε για πρώτη φορά το Σεπτέμβριο του 2012 στο στάδιο της έκδοσης 2.0 με το όνομα Manoglia. Η ευρεία κυκλοφορία της κανονικής έκδοσης αναμένονταν μέσα στο 2013 σε συσκευές Samsung με χρήση υλικού από την Intel.

4.2 Τι είναι το Tizen ;

Είναι ένα νέο λειτουργικό σύστημα για κινητά τηλέφωνα, ταμπλέτες, έξυπνες τηλεοράσεις, netbooks και συσκευές για ψυχαγωγία στο αυτοκίνητο: συσκευές δηλαδή με χαμηλές ενεργειακές δυνατότητες. Είναι ανοιχτού κώδικα, που τρέχει τον πυρήνα του **Linux** και τη μηχανή του **WebKit**. Δημιουργημένο από τον ίδιο φορέα που αναπτύσσει, προωθεί και συντηρεί το Linux. Αυτή τη στιγμή λανσάρεται «απέναντι» στα ήδη δημοφιλή Android της Google και iOS της Apple. Δημιουργήθηκε στη θέση του λειτουργικού **MeeGo**, και πάλι από τους ίδιους ανθρώπους, το οποίο είχε κυκλοφορήσει από τη Nokia στις συσκευές της –σε συνεργασία με την Intel- στα μέσα του 2010 αλλά απεσύρθη τον Σεπτέμβριο του 2011 γιατί δεν είχε την αναμενόμενη απήχηση.

Το Tizen αναπτύσσεται από μια συνεργασία των εταιρειών **Samsung, Intel, Linux Foundation**, και τη **Sprint-Nextel**. Η Sprint-Nextel είναι ένας κολοσσός των τηλεπικοινωνιών, η οποία αναπτύσσει καινοτομικές τεχνολογίες, και παρέχει υπηρεσίες ασύρματες και ενσύρματες σε επιχειρήσεις, ιδιώτες και κυβερνήσεις, και η οποία ανακοίνωσε την πρόθεσή της να συμμετέχει στο συνεταιρισμό της δημιουργίας του Tizen κυκλοφορώντας συσκευές με το λειτουργικό αυτό στις μελλοντικές τις σειρές προϊόντων.

Οι εφαρμογές που προορίζονται για το Tizen αναπτύσσονται μέσω μιας πλατφόρμας που διατίθεται, και χρησιμοποιείται γλώσσα προγραμματισμού **HTML5, jQuery, jQuery Mobile, CSS**, και **Javascript**. Να σημειωθεί ότι οι εφαρμογές που έχουν αναπτυχθεί για συσκευές Android μπορούν να τρέξουν σε συσκευές Tizen.

Η Ένωση **Tizen** (Samsung, Intel, Sprint-Nextel, Huawei, Fujitsu, NEC, Panasonic, SK Telecom, Orange, Vodafone) σχηματίστηκε τον Ιανουάριο του 2012 με σκοπό να κατευθύνει τον τεχνολογικό ρόλο του Tizen, συμπεριλαμβανομένου της συγκέντρωσης των απαιτήσεων, ταυτοποίησης και δρομολόγησης των μοντέλων, της επίβλεψης του marketing και της εκπαίδευσης. Η Ένωση αυτή λοιπόν, συνεργάζεται στενά με το **Linux Foundation** για την υλοποίηση της γενικής ιδέας ανοιχτού κώδικα που δίδει το Tizen.

Στις 25 Σεπτεμβρίου του 2012 κυκλοφόρησε την έκδοση **2.0 alpha** με το όνομα **Manoglia**. Προσέφερε ένα βελτιωμένο πλαίσιο εργασίας με περισσότερα χαρακτηριστικά, καλύτερη υποστήριξη πλατφόρμας HTML5/W3C, πολυ-διεργασίες και χρόνο εκτέλεσης Ιστού βασισμένη σε WebKit2, καθώς και καλύτερη ασφάλεια για τις εφαρμογές Ιστού. Η υποστήριξη του **OPENGL¹** έχει αναβαθμιστεί. Νέα πλατφόρμα SDK έχει προστεθεί για να βοηθήσει με την ανάπτυξη εφαρμογών σε Open Build Service (OBS). Στις 18 Φεβρουαρίου 2013 κυκλοφόρησε την έκδοση **2.0** με το όνομα **Manoglia**. Εκτός από τις περαιτέρω βελτιώσεις σε πλαίσια εργασίας Ιστού και διεπαφές προγραμματιστικού περιβάλλοντος, στο εγγενές πλαίσιο εφαρμογής με το ενσωματωμένο πλαίσιο ανάπτυξης εφαρμογών και τα σχετικά εργαλεία έχουν προστεθεί κάποια πρόσθετα χαρακτηριστικά όπως εφαρμογές παρασκήνιου, **πρόωθηση IP**, καθώς και το σύστημα **Text-To –Speech** το οποίο μετατρέπει κείμενο σε φωνή. Συμπερίληψη αυτού του πλαισίου είναι ένα φαινόμενο συγχώνευσης του πλαισίου εργασίας και της διεπαφής του λειτουργικού συστήματος για κινητά Bada με την πλατφόρμα του Tizen.

Τον Απρίλιο του 2013 ανακοινώνεται η Port-a-thon ενοποίηση των Samsung –Tizen. Αυτή η καμπάνια υποστηρίζει την αρχή της εισόδου των προγραμματιστών Bada εφαρμογών στην αγορά Tizen παρέχοντας τεχνική υποστήριξη και κίνητρα. Στις 17 Μαΐου 2013 κυκλοφορεί η έκδοση **2.1** με το όνομα **Nectarine**. Τον Ιούλιο του 2013 ανακοινώνεται από τη Samsung η εφαρμογή **Tizen Challenge** με κέρδη 4 εκατ.δολαρίων σε βραβεία. Στις 22 Ιουλίου κυκλοφόρησε η έκδοση 2.2 του Tizen .

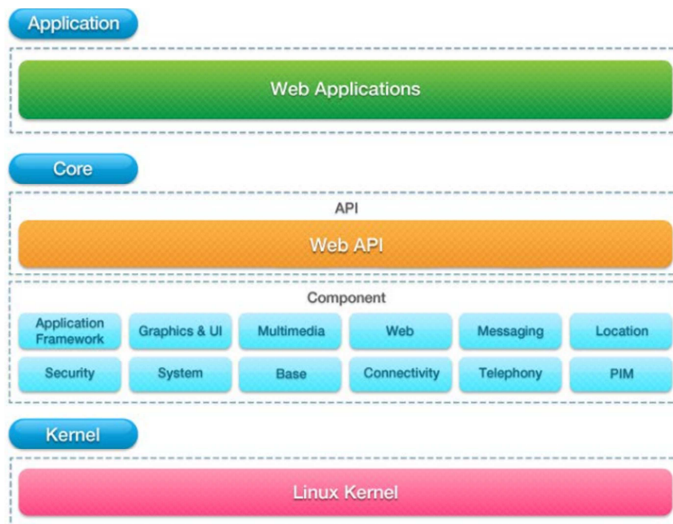
4.3 Αρχιτεκτονική του Tizen

¹ **OPENGL(Open Graphics Library)** είναι μια δια-γλωσσική διεπαφή χρήση για πλατφόρμες εφαρμογών πολύ-διεργασίας που προορίζεται για τη δημιουργία και ενοποίηση γραφικών δύο και τριών διαστάσεων. Αν ο αναγνώστης επιθυμεί να μάθει περισσότερες πληροφορίες μπορεί να ανατρέξει στην διεύθυνση <http://en.wikipedia.org/wiki/OpenGL>

Η αρχιτεκτονική του Tizen βασίζεται σε μια στοίβα λογισμικού . Όπως είπαμε και πιο πάνω, η πλατφόρμα ανάπτυξης χρησιμοποιεί HTML5, CSS, Javascript. Μια εφαρμογή είναι μια σύνθεση αυτών των στοιχείων συνδυασμένων σε ένα πακέτο (package) , με βάση τον προσδιορισμό της W3C, και εγκατεστημένων στη συσκευή. Σ' αυτό το σημείο δε θα επεκταθούμε άλλο στην προτυποποίηση αυτή. Αν ο αναγνώστης το επιθυμεί, μπορεί να ανατρέξει στην ιστοσελίδα του οργανισμού για περισσότερες πληροφορίες:

<http://www.w3.org/TR/widgets/> .

Το πακέτο αυτό έχει πρόσβαση στις λειτουργίες της συσκευής χρησιμοποιώντας το Web Runtime, το οποίο είχε αναπτυχθεί και χρησιμοποιηθεί από το προκάτοχο του Tizen MeeGo. Είναι ένα λογισμικό το οποίο όντας προεγκατεστημένο στη συσκευή, αυτόνομα στην ουσία, τρέχει τις εφαρμογές εκτός φυλλομετρητή, δεδομένου ότι χρησιμοποιείται γλώσσα υπερκειμένου για την ανάπτυξη των εφαρμογών. Στην εικόνα 25 φαίνεται η στοίβα λογισμικού του Tizen με τη μορφή διαγράμματος.



Εικόνα 26:Στοιβά λογισμικού αρχιτεκτονικής του Tizen

4.3.1 Πλαίσιο Εφαρμογής

Το πλαίσιο εφαρμογής (Application Framework) παρέχει τη λειτουργικότητα για το packaging, την εκκίνηση και τη διαχείριση μιας εφαρμογής. Κατά το πλαίσιο εφαρμογής, στην εκκίνηση συμπεριλαμβάνεται η εκκίνηση και άλλων εφαρμογών χρησιμοποιώντας το όνομα πακέτου, URI (Uniform Resource Identifier-Ενιαίο Αναγνωριστικό Πόρου-δηλ μια συμβολοσειρά χαρακτήρων για την αναγνώριση της πηγής), ή ένα MIME type (το οποίο λειτουργεί και αυτό σαν αναγνωριστικό για τον τύπο αρχείου στο Internet.). Επίσης εκκινεί προκαθορισμένες υπηρεσίες, όπως είναι η εφαρμογή συστήματος κλήσης. Το πλαίσιο εφαρμογής επίσης ειδοποιεί

εφαρμογές για συνήθη γεγονότα, όπως χαμηλή μπαταρία, χαμηλή μνήμη, αλλαγές στον προσανατολισμό της οθόνης και συμβάντα όπως πάτημα πλήκτρων. Επιπρόσθετα, παρέχονται και βάση δεδομένων και υποστήριξη ρυθμίσεων.

4.3.2 Η Βάση (Base)

Η Βάση περιέχει θεμελιώδεις βιβλιοθήκες συστήματος βασισμένες στο Linux οι οποίες παρέχουν χαρακτηριστικά-κλειδιά. Η Βάση ορίζεται ως αυτόνομη, με πακέτα που επιτρέπουν στο σύστημα να εκκινείται και να εισάγεται αυτόματα. Αποτελείται από τον πυρήνα των εφαρμογών με τα δομικά στοιχεία τους, και τον Πυρήνα του Linux. Επίσης περιέχει υποστήριξη βάσης δεδομένων, διεθνοποίηση και XML τεμαχισμό (parsing).

4.3.3 Πυρήνας του Linux(Linux kernel)

Το Tizen είναι βασισμένο στα γερά θεμέλια του **Linux**. Ο πυρήνας Linux είναι ανοιχτού κώδικα, δοκιμασμένος, σταθερός και μπορεί να βρεθεί παντού από κινητές συσκευές μέχρι υπερυπολογιστές. Ο πυρήνας του Linux είναι μια πρωτότυπη υλοποίηση πυρήνα λειτουργικού συστήματος. Σήμερα το Linux διαθέτει όλα όσα είναι αναγκαία για ένα σύγχρονο πυρήνα λειτουργικού:

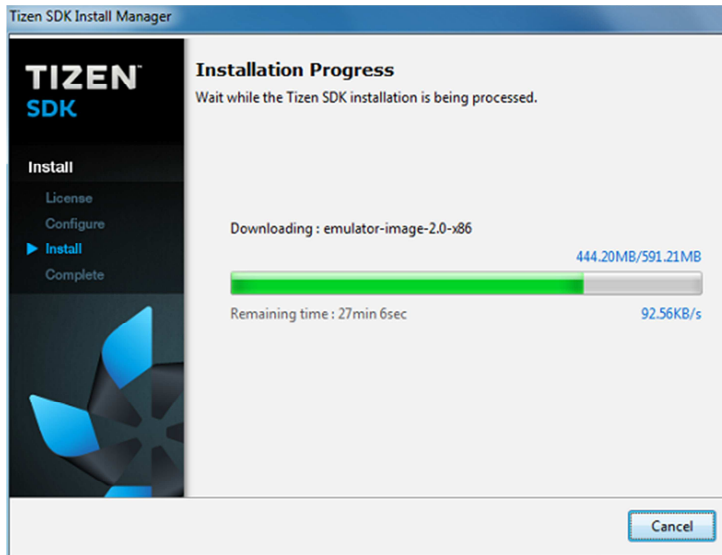
- Υποστήριξη πολύ-επεξεργαστικών συστημάτων.
- Πραγματική πολυδιεργασία
- Εικονική μνήμη
- Διαμοιραζόμενες βιβλιοθήκες
- Σωστή διαχείριση μνήμης
- Δικτύωση μέσω TCP/IP

Στον πυρήνα του **Linux** συμπεριλαμβάνεται και το **Σύστημα Παραθύρων X11 (X Window System)**, το οποίο είναι πρωτόκολλο απεικόνισης το οποίο χρησιμοποιείται για τη δημιουργία του γραφικού περιβάλλοντος χρήστη σε λειτουργικά συστήματα και το οποίο πρωτόκολλο χρησιμοποιεί και το Tizen. Αν ο αναγνώστης ενδιαφέρεται για περισσότερες λεπτομέρειες μπορεί να επισκεφτεί τη διεύθυνση http://en.wikipedia.org/wiki/X_Window_System.

4.4 Η εγκατάσταση της πλατφόρμας

Για να αναπτυχθεί εφαρμογή σε περιβάλλον Tizen SDK, θα πρέπει πρώτα να εγκατασταθεί το κατάλληλο λογισμικό για τη μεταγλώττιση του κώδικα Javascript/HTML5 και την προσομοίωση της λειτουργίας του λειτουργικού συστήματος Tizen σε Η/Υ. Αυτό είναι το Tizen IDE (Integrated Development Environment-Ολοκληρωμένο Περιβάλλον Ανάπτυξης), το οποίο είναι βασισμένο στο δημοφιλές **Eclipse**, μια εφαρμογή που συνδυάζει δυνατότητες μεταγλώττισης, αποσφαλμάτωσης, αυτόματης παραγωγής κώδικα, καθώς και προσομοίωσης περιβάλλοντος λογισμικού Tizen για κινητά τηλέφωνα σε Η/Υ. Αυτό μπορεί να το κατεβάσει ο

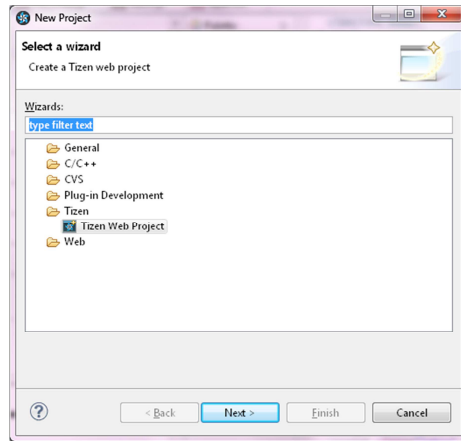
χρήστης και εν δυνάμει προγραμματιστής από τον ιστότοπο του Tizen <https://www.tizen.org/>. Η εγκατάσταση είναι πολύ απλή διαδικασία, το μόνο που απαιτεί είναι σύνδεση στο Internet, καθώς δεν είναι μια διαδικασία που γίνεται τοπικά στο άκρο του χρήστη.



Εικόνα 27: Εγκατάσταση Tizen IDE

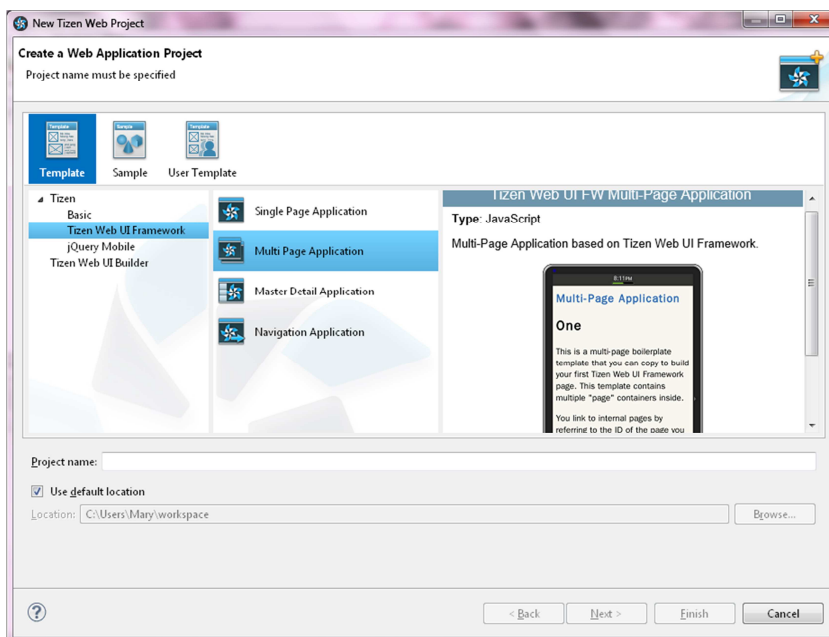
4.5 Δημιουργία νέας εφαρμογής στο Tizen IDE

Από το μενού File(Αρχείο)>New>Project δημιουργούμε νέο project στο φάκελο workspace του Tizen μέσα στο φάκελο του χρήστη στο C: (για τα Windows).Υπάρχουν κάποιες επιλογές που έχει προ-ρυθμισμένες το IDE και αφορούν την εφαρμογή και το πώς είναι διαμορφωμένη ως προς τις σελίδες και τα μενού της, αν θα είναι δηλαδή μιας σελίδας, πολλών ή αν θα έχει μενού.

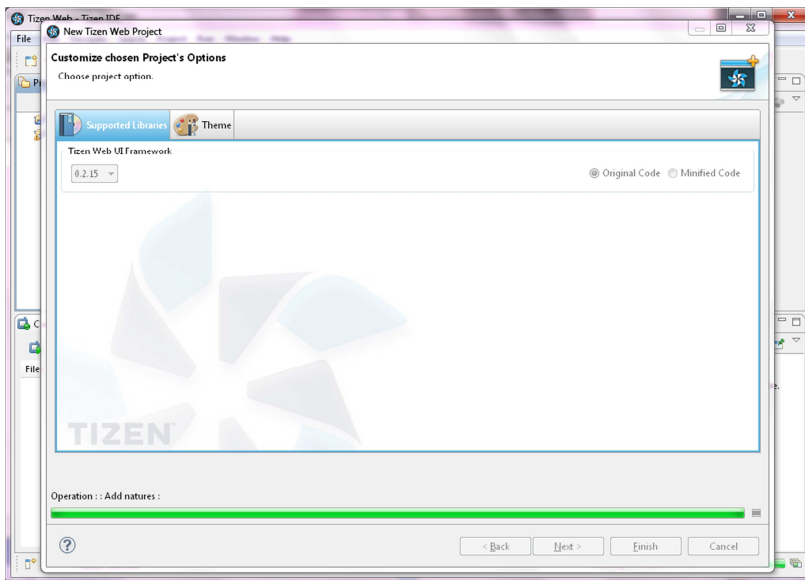


Εικόνα 28: Δημιουργία νέου project σε περιβάλλον Tizen

Σε αυτό το βήμα δίνουμε με την πληκτρολόγηση το όνομα που θέλουμε στην εφαρμογή μας και πατάμε «Next» (Επόμενο) και μετά Finish(Τέλος):

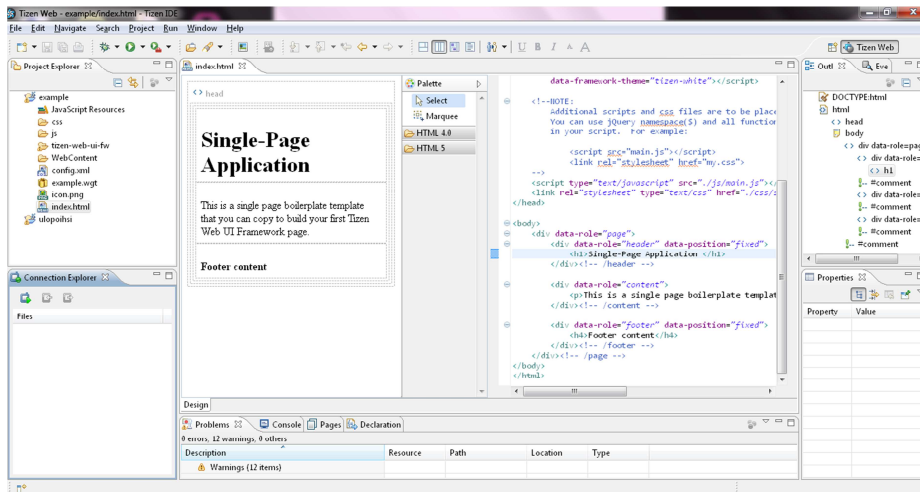


Εικόνα 29 :Επιλογή του είδους της εφαρμογής ως προς την πολυπλοκότητά της.



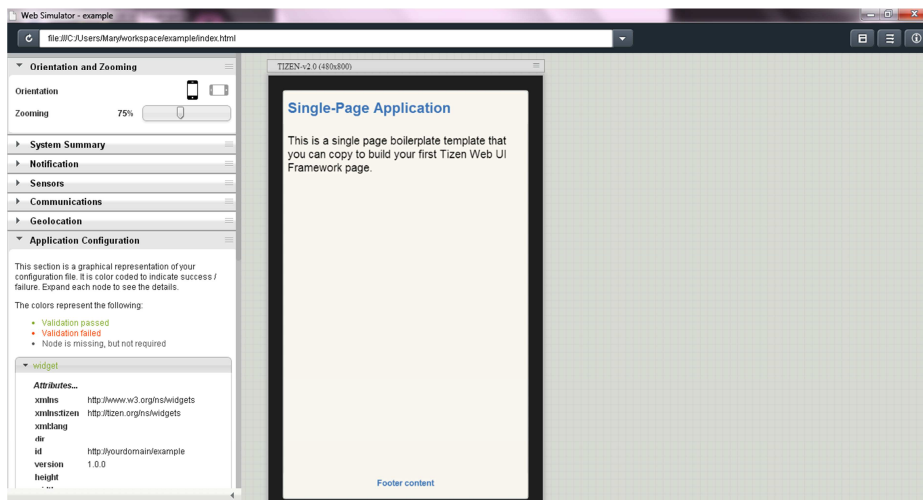
Εικόνα 30: Τελευταίο στάδιο της δημιουργίας του νέου project

Κατά την επιλογή και τη δημιουργία νέου project, το IDE δίνει κάποια στοιχεία του κώδικα έτοιμα, εισάγει δηλαδή κώδικα για την διευκόλυνση του χρήστη-developer. Αυτά αφορούν κυρίως τη μορφοποίηση της εφαρμογής, ανάλογα το πόσο πολύπλοκη επιλέξαμε προηγουμένως να είναι η εφαρμογή μας.



Εικόνα 31: Η πρώτη επαφή με τον κώδικα του project που μόλις δημιουργήσαμε

Αυτά αφορούν κυρίως το διαχωρισμό της οποιαδήποτε εφαρμογής σε σελίδες (πρώτη-δεύτερη κλπ) με divs της HTML5 -ανάλογα βέβαια και με το είδος της εφαρμογής που έχουμε επιλέξει- καθώς και βιβλιοθήκες της Javascript για γραφικό περιβάλλον και jQuery. Αυτό που προκύπτει από τον έτοιμο κώδικα του Tizen είναι ένα γραφικό περιβάλλον μιας εφαρμογής με κείμενο δοκιμαστικό που σκοπό έχει να δώσει στο χρήστη και εν δυνάμει developer μια ιδέα για το πώς διαμορφώνεται μια εφαρμογή .



Εικόνα 32:Η πρώτη επαφή με την εφαρμογή

Όσον αφορά το κομμάτι του κώδικα και το τι «κρύβεται» πίσω από την εφαρμογή που τρέχει ο προσομοιωτής, το Tizen IDE χρησιμοποιεί ένα σύστημα αρχείων με οργάνωση δέντρου χωρίζοντας τα με βάση την επέκταση για το διαχωρισμό της Javascript, του Css και των βιβλιοθηκών για την διεπαφή χρήστη από την html. Αρχικά υπάρχει ένα αρχείο με .js κατάληξη για την javascript, με το όνομα main.js μέσα σε ένα φάκελο με το όνομα js. Επίσης ένα αρχείο με .css κατάληξη για τα φύλλα css με το όνομα my μέσα σε ένα φάκελο css .Αυτό σημαίνει ότι ο χρήστης και εν δυνάμει προγραμματιστής μπορεί να χρησιμοποιήσει τα υπάρχοντα αρχεία για την δημιουργία της εφαρμογής του συμπληρώνοντας κώδικα, καθώς και να προσθέσει επιπλέον αρχεία στους φακέλους αυτών, αν αυτό χρειάζεται. Τώρα όσον αφορά το βασικό αρχείο HTML, αυτό βρίσκεται μόνο του στο βασικό φάκελο της εφαρμογής που έχει το όνομα που του δώσαμε κατά τη δημιουργία της , μαζί με κάποια άλλα αρχεία τα οποία ο χρήστης δε χρειάζεται να τροποποιήσει με κάποιο τρόπο για τη δημιουργία της εφαρμογής του.

Κεφάλαιο 5ο-To Firefox OS

5.1 Εισαγωγή

Το **Firefox OS** είναι ένα web λειτουργικό σύστημα ανοιχτού κώδικα για κινητά τηλέφωνα. Αρχικά ήταν γνωστό σαν B2G - Boot to Gecko. Είναι βασισμένο στον πυρήνα του Linux και η ανάπτυξή του υποστηρίζεται από τη Mozilla, ένα μη κερδοσκοπικό οργανισμό, που είναι γνωστός από το πασίγνωστο φυλλομετρητή Mozilla Firefox. Το Firefox OS έχει σχεδιαστεί να παρέχει ένα νέο λειτουργικό σύστημα για φορητές συσκευές, χρησιμοποιώντας ανοιχτά πρότυπα και προσεγγίσεις όπως εφαρμογές HTML5, Javascript και ανοιχτά APIs για την επικοινωνία με το υλικό των συσκευών, καθώς και με το ηλεκτρονικό «κατάστημα» των εφαρμογών.

Αντίστοιχα, ανταγωνίζεται τα Android /Chrome OS της Google, και τα Sailfish της Jolla, και Windows phone της Microsoft. Το Firefox OS παρουσιάστηκε επίσημα το Φεβρουάριο του 2012, «φορεμένο» σε συσκευή που είναι συμβατή με Android. Τον Ιανουάριο του 2013 κυκλοφόρησε και επίσημα σε συσκευή της ZTE, και μελλοντικά αναμένονται και άλλα μοντέλα καθώς έχουν ανακοινωθεί επίσημα συνεργασίες με κορυφαίους κατασκευαστές όπως η Alcatel, η LG, η Sprint και άλλες εταιρείες από όλον τον κόσμο.



Firefox OS

Εικόνα 33 : Το λογότυπο του Firefox OS

5.2 Η αρχιτεκτονική του Firefox OS

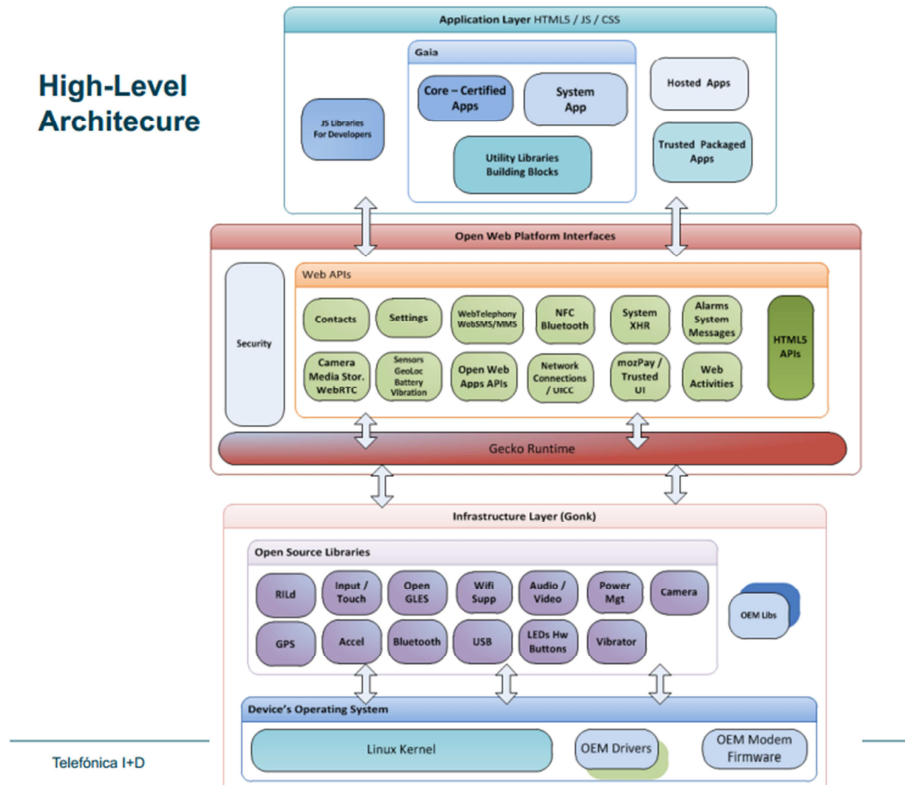
Η αρχιτεκτονική του Firefox OS αποτελείται από τρία επιμέρους στρώματα :

1. **Gonk** : ο πυρήνας του λειτουργικού που είναι βασισμένος στον πυρήνα του Linux, και που είναι ανεξάρτητος από τον τύπο του hardware (αυτό στην ορολογία της ανάπτυξης λογισμικού λέγεται

HAL -Hardware Abstraction Layer- Στρώμα αφαίρεσης υλικού). Κάποια κομμάτια της HAL που χρησιμοποιήθηκαν για την ανάπτυξη του Firefox OS, είναι κοινά με αυτά του Android (που χρησιμοποιεί επίσης τον πυρήνα του Linux) : όπως το GPS, η κάμερα κ.ά. Το Gonk είναι απλή διανομή Linux.

2. **Gecko** : είναι η runtime εφαρμογή του Firefox OS, που σημαίνει ότι το Gecko υλοποιεί τις εφαρμογές που τρέχουν στο Firefox OS, είναι δηλαδή ο μεσολαβητής μεταξύ του πυρήνα και της διεπαφής. Είναι μια απλή μηχανή εμφάνισης ιστοσελίδων, η οποία εφαρμόζει τα περισσότερα από τα σημερινά πρότυπα του παγκόσμιου ιστού για HTML,CSS και Javascript. Περιλαμβάνει δικτυακές λειτουργίες, γραφικές, εικονική μηχανή Javascript, μηχανή σχεδίου και porting layers, είναι το λογισμικό που εμφανίζει όλα τα γραφικά στοιχεία του τηλεφώνου, πρόκειται για το ίδιο λογισμικό που χρησιμοποιεί ο περιηγητής firefox για να απεικονίζει τον ιστό. Το Gecko εφαρμόζει ανοιχτά standards για την HTML5, Javascript, CSS και περιλαμβάνει στοίβα διαδικτύου, στοίβα γραφικών, διάταξη μηχανής, και στρώματα θυρών.
3. **Gaia** : είναι η διεπαφή του Firefox OS και ελέγχει όλα όσα γίνονται στην οθόνη. Το Gaia περιλαμβάνει εξ'ορισμού κάποιες εκτελέσεις για κλείδωμα οθόνης, για αρχική οθόνη, πληκτρολόγιο αριθμών και εφαρμογή επαφών, καθώς επίσης και πληκτρολόγιο μηνυμάτων κειμένου και εφαρμογή κάμερας. Αυτά επιπλέον στις κλασικές εφαρμογές ημερολογίου, υπολογιστή, ταχυδρομείου και καταστήματος εφαρμογών. Το Gaia έχει γραφτεί εξ ολοκλήρου σε HTML, CSS και Javascript. Αλληλεπιδρά με το λειτουργικό σύστημα μέσω ανοιχτών Web APIs που εκτελούνται από το Gecko. Διασυνδέεται με το σύστημα μέσω ανοιχτών δικτυακών API τα οποία υλοποιούνται από το Gecko την μηχανή εμφάνισης ιστοσελίδων. Ακριβώς επειδή χρησιμοποιεί μόνο κάποιες standard web APIs, μπορεί να δουλέψει με διάφορα λειτουργικά και διαφορετικούς φυλλομετρητές.

Η παρακάτω εικόνα δείχνει την ιεραρχία στα στρώματα καθώς και τα κομμάτια της αρχιτεκτονικής του FirefoxOS καθώς και το πώς αλληλεπιδρούν μεταξύ τους :



Εικόνα 34 : Η αρχιτεκτονική του Firefox OS σχηματικά

5.3 Ανάπτυξη εφαρμογών για Firefox OS

Στην περίπτωση του Firefox OS, δεν υπάρχει κάποιο middleware (δηλ SDK) που χρησιμοποιείται αποκλειστικά για ανάπτυξη εφαρμογών για Firefox OS, αλλά τα αρχεία που περιέχουν τον κώδικα της HTML5 / Javascript, μπορούν να συγγραφούν σε κάποιο αρχείο κειμένου και είτε να συμπειστούν μαζί με το manifest αρχείο και οπότε έχουμε την **packaged** εφαρμογή μας, είτε να ανεβάσουμε τα αρχεία σε κάποιον server, όπως θα κάναμε για μια απλή ιστοσελίδα, και έτσι έχουμε μια **hosted** εφαρμογή.

Υπάρχουν δύο τύποι web εφαρμογών ανοιχτού κώδικα : packaged και hosted. Όταν μιλάμε για packaged εφαρμογές, μιλάμε στην ουσία για αρχεία zip που περιλαμβάνουν όλα τα απαιτούμενα κομμάτια της εφαρμογής :HTML, CSS, Javascript, εικόνες, manifest κλπ. συμπεισμένα σε ένα αρχείο. Από την άλλη οι hosted, «τρέχουν» από ένα server μέσω μιας διεύθυνσης συγκεκριμένης, όπως θα ήταν μια οποιαδήποτε ιστοσελίδα. Και οι δύο τύποι

εφαρμογών απαιτούν ένα αρχείο manifest έγκυρο, το οποίο θα πρέπει να βρίσκεται στον root φάκελο της εφαρμογής με το όνομα manifest.webapp.. Αυτό το αρχείο περιλαμβάνει σημαντικές πληροφορίες για την εφαρμογή όπως είναι η έκδοση, το όνομα, η περιγραφή, τις διαδικτυακές διευθύνσεις που μπορεί κανείς να κατεβάσει την εφαρμογή. Από αυτά, μόνο το όνομα και η περιγραφή είναι υποχρεωτικά, και στις packaged εφαρμογές το πεδίο path στο manifest είναι υποχρεωτικό.

Όταν έρθει η ώρα να ανεβεί η εφαρμογή στο Firefox Marketplace (το διαδικτυακό κατάστημα για εφαρμογές), είτε θα ανεβεί το αρχείο zip, είτε μόνο το URL του server που φιλοξενείται η εφαρμογή (ανάλογα δηλαδή με το είδος της εφαρμογής-hosted ή packaged). Τώρα, όσον αφορά τα κυρίως αρχεία της εφαρμογής, αυτά πρέπει να είναι γραμμένα σε HTML5, Javascript και CSS μέσω οποιουδήποτε editor, είτε αυτός είναι το σημειωματάριο των Windows, είτε κάποιο ολοκληρωμένο software όπως το Dreamweaver της Adobe. Αυτά τα αρχεία πρέπει να συμπεστούν όλα μαζί σε ένα φάκελο, μαζί με το manifest.webapp. Αυτό το αρχείο είναι το βασικότερο πράγμα που χρειαζόμαστε για να «τρέξουμε» την οποιαδήποτε εφαρμογή έχουμε φτιάξει είτε σε κάποια συσκευή smartphone, είτε σε προσομοίωση .

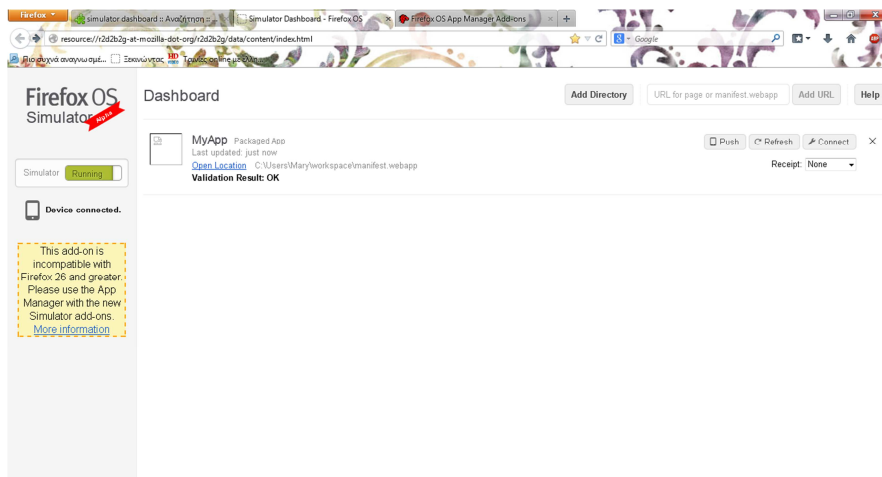
Υπάρχει όμως και η δυνατότητα της **αυτό-δημοσίευσης** (auto-publish). Σε hosted εφαρμογές, το μόνο που απαιτείται είναι το «ανέβασμα» σε ένα server όπως περιγράφηκε παραπάνω. Σε packaged εφαρμογές, αυτό μπορεί να γίνει ανεβάζοντάς την σε ένα server, μαζί με ένα mini-manifest αρχείο στον ίδιο φάκελο το οποίο θα ταυτοποιεί την εφαρμογή, και θα χρησιμοποιηθεί κατά την διαδικασία εγκατάστασης. Σε αυτή την περίπτωση (της αυτό-δημοσίευσης), το zip αρχείο της packaged εφαρμογής (που περιέχει και το manifest αρχείο), πρέπει να μετονομαστεί σε package.zip. Στη συνέχεια πρέπει να δημιουργηθεί ένα δεύτερο, νέο manifest αρχείο με τίτλο package.manifest. Αυτό θα περιέχει κάποια στοιχεία όπως όνομα, μονοπάτι, έκδοση, δημιουργός κ.ά. Αξίζει να σημειωθεί ότι αυτό δεν είναι το βασικό manifest αρχείο της εφαρμογής, εκείνο που βρίσκεται μέσα στο συμπεσμένο αρχείο είναι το βασικό.

Επίσης, χρειάζεται ένα ακόμη αρχείο, index.html το οποίο θα περιέχει κώδικα Javascript που θα καλεί την ίδια την packaged εφαρμογή, καθώς και κώδικα για την επανάκληση για την επιτυχία και την αποτυχία κοινοποίησης. Αυτά τα 3 αρχεία, δηλαδή τα manifest, package και index, στη συνέχεια πρέπει να τοποθετηθούν στον αρχικό φάκελο της εφαρμογής. Τώρα, από κει και πέρα χρειάζεται μια συσκευή συμβατή με Firefox Os για να γίνει περιήγηση στον server, εκεί όπου «ανέβηκαν» τα αρχεία, και να επιβεβαιωθεί μέσω του κώδικα Javascript η προτροπή να εγκατασταθεί η εφαρμογή.

5.4 Εγκατάσταση προσομοίωσης σε Η/Υ

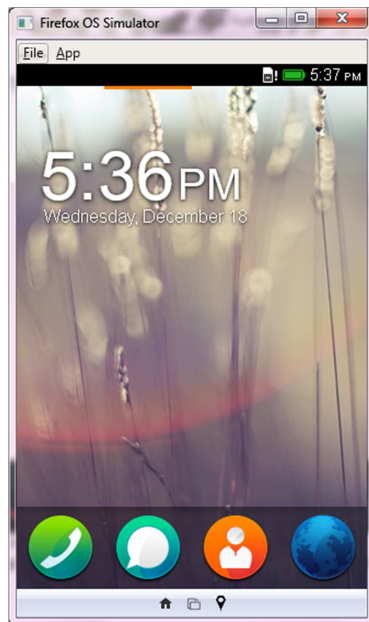
Όπως αναφέρθηκε και παραπάνω, δεν υπάρχει κάποιο middleware που θα λειτουργήσει ως προσομοιωτής και συντάκτης σε γλώσσα HTML5/Javascript για τις εφαρμογές που θα φτιάξει ο εν δυνάμει developer. Αυτό που έχει αναπτύξει η Mozilla γι' αυτό το σκοπό, είναι ένα add-on, δηλαδή ένα πρόσθετο για τον Firefox browser, που θα λειτουργήσει σαν προσομοιωτής για όσο διαρκέσει η ανάπτυξη της εφαρμογής μας ώστε να μη χρειάζεται να

φορτώνουμε την εφαρμογή ολόκληρη σε κάθε βήμα της ανάπτυξης αυτής στη συσκευή μας smartphone. Ονομάζεται **Firefox OS Simulator** και εγκαθίσταται χωρίς να χρειάζεται επανεκκίνηση του Η/Υ. Μπορεί κανείς να το βρει στη σελίδα με τα πρόσθετα του Mozilla Firefox : <https://addons.mozilla.org/el/firefox/addon/firefox-os-simulator/> ,είναι ένα αρχείο κάποιου μεγέθους, οπότε χρειάζεται κάποιος χρόνος για να «κατεβεί» αρχικά, αλλά δεν βαραίνει το σύστημα ιδιαίτερα κατά τη χρήση. Όταν εγκατασταθεί σαν πρόσθετο του Mozilla Firefox browser,τότε το μόνο που μένει είναι να «φορτωθεί» η εφαρμογή μέσω του αρχείου manifest.webapp στο πεδίο Add directory.



Εικόνα 35:Το Add-on του Mozilla Firefox που λειτουργεί ως προσομοιωτής για τις εφαρμογές του FirefoxOS

Αμέσως μετά, κάνοντας «Connect», θα δούμε την αρχική οθόνη και το μενού του Firefox OS. Όπως φαίνεται και στην παραπάνω εικόνα, το dashboard είναι σε alpha έκδοση ακόμα, οπότε κάποιες δυσκολίες είναι αναμενόμενες κατά την προσομοίωση. Πατώντας το κουμπί «Connect» μπορούμε να «τρέξουμε» το λειτουργικό με όποιες εφαρμογές μπορεί να έχουμε φορτώσει .Στην επόμενη εικόνα φαίνεται η αρχική οθόνη (home screen) του FirefoxOS:



Εικόνα 36: Η Home screen του Firefox OS

5.5 Firefox για Android

Η Mozilla, που είναι η υπεύθυνη εταιρία για το Firefox OS καθώς και για τον ίδιο το Firefox browser, ανακοίνωσε την δημιουργία του Firefox για Android. Αυτό σημαίνει ότι ανέπτυξε έκδοση του δημοφιλούς φυλλομετρητή για κινητά τηλέφωνα με λειτουργικό Android. Μπορεί με το πρώτο άκουσμα κανείς να θεωρήσει αυτή την κίνηση λίγο περίεργη, δεδομένου ότι από τη στιγμή που αναπτύχθηκε εξ ολοκλήρου νέο λειτουργικό Firefox θα έπρεπε να προτρέπει η εταιρία τους χρήστες να προτιμήσουν το «πρωτότυπο» και ολοκληρωμένο δημιούργημα, αλλά αν το σκεφτεί κανείς καλύτερα, διαφαίνεται μια προοπτική διαφορετική μέσα από αυτή την κίνηση: Να βρεθεί ένα προϊόν Mozilla σε κάθε κινητή συσκευή smartphone, ό,τι λειτουργικό και να «φοράει» αυτή, και όχι με αντίτιμο και σκοπό το οικονομικό κέρδος, μια και η Mozilla είναι μια μη-κερδοσκοπική εταιρία.

Ο Firefox για κινητά Android δίνει πολλές δυνατότητες νέες στους χρήστες όπως:

- Γρήγορη περιήγηση εφάμιλλη με αυτή του Firefox που προορίζεται για υπολογιστές.
- Πρόσθετα (add-ons) : μπλοκάρισμα παραθύρων και διαφημίσεων, διαχείριση συνθηματικών (passwords) .
- Δυνατότητα διαχείρισης καρτελών, για διευκόλυνση της εμπειρίας σερφαρίσματος από τη μικρή οθόνη του κινητού.
- Αναπαραγωγή βίντεο μέσω φυλλομετρητή του κινητού, και μάλιστα η συγκεκριμένη έκδοση που προορίζεται για Android υποστηρίζει μια ευρεία γκάμα κωδικοποιήσεων για video όπως το h264.

- Ανάγνωση κειμένου μέσω του φυλλομετρητή: προσαρμόζει το κείμενο στο μέγεθος της σελίδας για καλύτερη εμπειρία ανάγνωσης
- Υποστήριξη της HTML5 και όλων των νέων δυνατοτήτων που αυτή προσφέρει.
- Ασφάλεια :Η Mozilla έχει προνοήσει για την ασφάλεια των χρηστών που χρησιμοποιούν τον Firefox για Android, με το να παρέχει προστασία από κακόβουλα λογισμικά .Λειτουργεί με το να ελέγχει τις ιστοσελίδες που επισκέπτεται ο χρήστης μέσα από λίστες που έχουν δηλωμένες ιστοσελίδες με κακόβουλα λογισμικά. Αυτές οι λίστες κατεβαίνουν και ανανεώνονται αυτόματα κάθε 30 λεπτά περίπου αν το πρόγραμμα της ασφάλειας από κακόβουλα λογισμικά είναι ενεργοποιημένο.
- Συγχρονισμός :μπορεί να γίνει συγχρονισμός των συσκευών του χρήστη όπως tablets, υπολογιστές, smartphones κλπ για διαμοιρασμό ιστορικού περιήγησης, σελιδοδεικτών, συνθηματικών κλπ

Κεφάλαιο 6ο : Android

6.1 Εισαγωγή

Το Android αρχικά αναπτύχθηκε από την Google και αργότερα από την Open Handset Alliance. Επιτρέπει στους κατασκευαστές λογισμικού να συνθέτουν κώδικα με την χρήση της γλώσσας προγραμματισμού Java, ελέγχοντας την συσκευή μέσω βιβλιοθηκών λογισμικού ανεπτυγμένων από την Google. Αυτό σημαίνει ότι είναι ένα λογισμικό ανοιχτού κώδικα με ιδιότητα σημεία. Το λογότυπο για το λειτουργικό σύστημα Android είναι ένα ρομπότ σε χρώμα πράσινου μήλου και σχεδιάστηκε από τη γραφίστρια Irina Blok.



Εικόνα 37:Το λογότυπο του Android

Η πλατφόρμα είναι προσαρμόσιμη σε μεγαλύτερη ανάλυση (VGA), διδιάστατες ψηφιακές γραφικές βιβλιοθήκες, τρισδιάστατα γραφικά βασισμένα στην OpenGL ES 1.0 έκδοση χαρακτηριστικών, καθώς και παραδοσιακές απεικονίσεις οθόνης smartphone. Το Android υποστηρίζει τεχνολογίες συνδεσιμότητας συμπεριλαμβανομένου GSM/EDGE, CDMA, UMTS, Bluetooth, και Wi-Fi. Αυτές οι δυνατότητες αναπτύχθηκαν σιγά σιγά μέσα από τις διάφορες εκδόσεις που κυκλοφόρησαν από το 2009.

Αρχικά η έκδοση 1.1 το 2007 προοριζόταν για συσκευές απλές, χωρίς οθόνες αφής, οπότε δεν υποστήριζε πολλές δυνατότητες. Η πρώτη έκδοση για smartphones, το 2009, ήταν η 1.5 cupcake. Στη συνέχεια το 1.6 Donut στην ίδια χρονιά, το 2.0 Éclair μέσα στο 2009 και αυτό, το 2010 τα: 2.2 Froyo, 2.3 Gingerbread, το 2011 τα: 3.0 Honeycomb, 4.0 Icecream Sandwich, το 2012 το Jellybean ως έκδοση 4.1, 4.2, 4.3 σταδιακά με updates και το τελευταίο που κυκλοφόρησε το 2013 το 4.4 KitKat. Παρατηρείται εκτός από μια γρήγορη παραγωγικότητα και εξέλιξη, και μια ευρηματικότητα στα ονόματα !

Παρόμοια με το App Store του iPhone OS, το Android Market είναι ένας κατάλογος εφαρμογών που μπορούν να μεταφορτωθούν και εγκατασταθούν στην συσκευή άμεσα μέσω ασύρματων καναλιών, χωρίς την

χρήση υπολογιστή. Το λειτουργικό Android είναι το δημοφιλέστερο για smartphones καταλαμβάνοντας το 48% επί των κατόχων κινητών στην Ευρώπη. Όλα τα applications διατίθενται από το Android Market της Google και το σύνολο αυτών ξεπερνάει τις 450.000 με συνολικά 10 εκατομμύρια downloads. Η πληθώρα των συσκευών και οι χαμηλές τιμές τους κάνουν την μαζικότερη σε λειτουργία χρήση του. Εταιρείες όπως η Samsung, Sony Ericsson, LG, HTC, ZTE, Huawei, Motorola, αλλά και Sharp, Asus, Acer, DROID, που δε κυκλοφορούν στην ελληνική αγορά, φέρουν λειτουργικό Android.



Εικόνα 38: Η Home screen του Android σε διαφορετικές συσκευές .Από αριστερά Samsung, Samsung, HTC

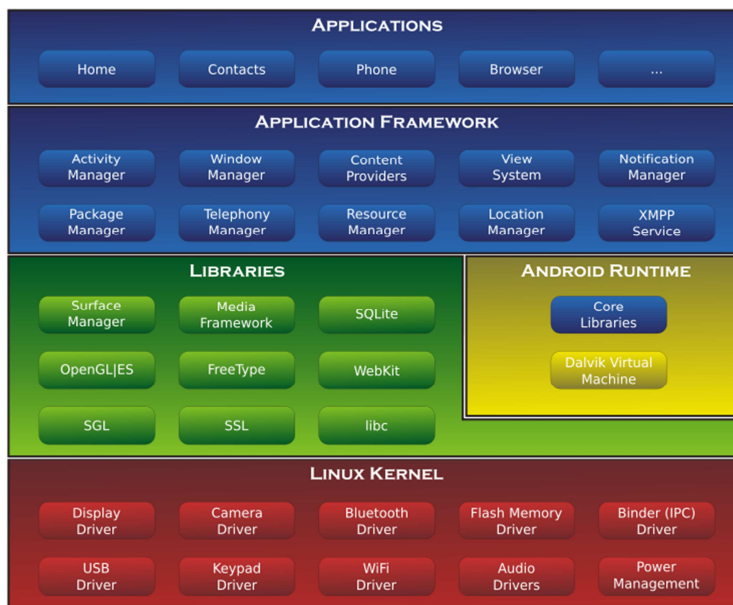
6.2 Αρχιτεκτονική του Android

Το Android είναι μια στοίβα λογισμικού. Η λογική πίσω από αυτήν την έκφραση και σε όλη την φιλοσοφία του Android, κρύβεται στο ακόλουθο διάγραμμα με τα βασικά συστατικά του. Η αρχιτεκτονική του Android ενθαρρύνει την έννοια επαναχρησιμοποίησης κώδικα, που επιτρέπει να δημοσιευτούν και να μοιραστούν δραστηριότητες, υπηρεσίες και δεδομένα με άλλες εφαρμογές με πρόσβαση που ελέγχεται από τους περιορισμούς ασφάλειας που έχουν τεθεί. Έτσι γίνεται δυνατή η επέκταση και βελτιστοποίηση υπαρχόντων εφαρμογών ή δημιουργίας καινούργιων, χρησιμοποιώντας κώδικα από αυτές. Οι ακόλουθες υπηρεσίες εφαρμογών είναι οι ακρογωνιαίοι λίθοι της αρχιτεκτονικής όλων των εφαρμογών Android, παρέχοντας το πλαίσιο που θα χρησιμοποιηθεί για τη δημιουργία του λογισμικού :

- **Διαχειριστής Δραστηριοτήτων (Activity Manager)** -Ελέγχει τον κύκλο ζωής των δραστηριοτήτων,συμπεριλαμβανομένης της διαχείρισης του σωρού δραστηριοτήτων.
- **Όψεις (Views)** -Χρησιμοποιούνται για να κατασκευάσουν διεπαφές χρήστη (user interface) για τις δραστηριότητες (activities).

- **Διαχειριστή Ειδοποιήσεων (Notification Manager)** - Παρέχει ένα συνεπή και αποτελεσματικό μηχανισμό για να προειδοποιεί τον χρήστη
- **Παροχέας Περιεχομένου (Content Providers)** -Επιτρέπουν στις εφαρμογές να μοιραστούν δεδομένα μεταξύ τους.
- **Διαχειριστής Πόρων (Resource Manager)** -Υποστηρίζει πόρους εκτός κώδικα όπως συμβολοσειρές και γραφικά.

Στην στοίβα του Android που φαίνεται στην παρακάτω εικόνα., παρατηρούμε 4 επίπεδα. Ακολούθως θα περιγράψουμε συνοπτικά τα βασικά αυτά επίπεδα χωρίς να μπούμε σε λεπτομέρειες για όλα τα περιεχόμενα του κάθε επιπέδου. Αν ο αναγνώστης επιθυμεί να μάθει περισσότερα, μπορεί να επισκεφθεί την επίσημη ιστοσελίδα του Android για κατασκευαστές (<http://developer.android.com>). Κάθε επίπεδο στην αρχιτεκτονική αυτή, χρησιμοποιεί τις υπηρεσίες που του προσφέρονται από τα πιο πάνω επίπεδα. Παρακάτω θα δούμε αυτά τα επίπεδα ξεκινώντας από το πιο χαμηλό.



Εικόνα 39 : Η στοίβα λογισμικού της αρχιτεκτονικής του Android

6.2.1 Πυρήνας Linux (Kernel)

Το Android είναι βασισμένο στα γερά θεμέλια του Linux. Ο πυρήνας Linux είναι δοκιμασμένος, σταθερός και πετυχημένος και μπορεί να βρεθεί παντού, από ρολόγια χειρός μέχρι

υπερπολογιστές. Το Linux παρέχει στο Android το αφαιρετικό επίπεδο υλικού, επιτρέποντας του να μπορεί να χρησιμοποιηθεί σε μεγάλη ποικιλία πλατφορμών στο μέλλον. Ειδικότερα, το Android χρησιμοποιεί τον πυρήνα Linux για την διαχείριση μνήμης, την διαχείριση διεργασιών, την δικτύωση και άλλες υπηρεσίες του λειτουργικού συστήματος.

6.2.2 Εγγενείς βιβλιοθήκες (Native Libraries)

Στο αμέσως υψηλότερο επίπεδο βρίσκουμε τις Native Libraries – Εγγενείς Βιβλιοθήκες. Όλες αυτές είναι γραμμένες στην γλώσσα προγραμματισμού C και C++ και μεταγλωττίστηκαν για την συγκεκριμένη αρχιτεκτονική υλικού που χρησιμοποιείται από το τηλέφωνο. Οι βιβλιοθήκες αυτές δεν είναι εφαρμογές που μπορούν να στηθούν από μόνες τους. Υπάρχουν για να μπορούν να κληθούν από προγράμματα υψηλότερου επιπέδου. Από την έκδοση Donut του Android και μετά, οι κατασκευαστές μπορούν να γράφουν τις δικές τους τέτοιες βιβλιοθήκες με την χρήση της Εργαλειοθήκης NDK (Native Development Kit).

6.2.3 Χρόνος εκτέλεσης

Στο ίδιο επίπεδο με τις εγγενείς βιβλιοθήκες, βρίσκουμε και τον χρόνο εκτέλεσης Android. Εδώ υπάρχουν οι βασικές βιβλιοθήκες της Java και η εικονική μηχανή Dalvik. Η Dalvik είναι μια βελτιστοποιημένη υλοποίηση μιας εικονικής μηχανής Java για φορητές συσκευές από την Google. Η Dalvik τρέχει .dex αρχεία, τα οποία είναι bytecodes που προέρχονται από αρχεία .class και .jar. Εν αντιθέσει όμως με τα .class αρχεία, τα .dex είναι πολύ πιο συμπαγή και αποδοτικά, γεγονός σημαντικό για συσκευές με περιορισμένη μνήμη και μπαταρία. Το Android περιλαμβάνει ένα σύνολο βασικών βιβλιοθηκών που παρέχουν τις περισσότερες από τις διαθέσιμες λειτουργίες των βασικών βιβλιοθηκών της Java. Κάποια πακέτα και κλάσεις υπάρχουν και στο Android, κάποια άλλα δεν υποστηρίζονται καθόλου, ενώ ταυτόχρονα το Android παρέχει και επιπρόσθετα προσαρμοσμένα στις δικές του ανάγκες.

6.2.4 Η εικονική μηχανή Dalvik (VM –Virtual Machine)

Ένα από τα στοιχεία κλειδιά του Android είναι η εικονική μηχανή Dalvik. Το Android χρησιμοποιεί την δικιά του εικονική μηχανή και όχι μια παραδοσιακή, με σκοπό να εξασφαλίσει ότι πολλαπλά στιγμιότυπα τρέχουν αποτελεσματικά σε μια ενιαία συσκευή. Η Dalvik VM (Virtual Machine) χρησιμοποιεί τον πυρήνα Linux της συσκευής για να χειριστεί τις χαμηλού επιπέδου λειτουργίες που περιλαμβάνουν την ασφάλεια, τον πολυνηματισμό και τη διαχείριση διαδικασιών και μνήμης. Είναι επίσης δυνατό να γραφτούν εφαρμογές C/C++ που τρέχουν άμεσα στο εσωτερικό του λειτουργικού Linux. Αν και μπορεί να γίνει αυτό, στις περισσότερες περιπτώσεις δεν υπάρχει κανένας λόγος.

6.2.5 Πλαίσιο εφαρμογής (Application Framework)

Πάνω από τις εγγενείς βιβλιοθήκες και το χρόνο εκτέλεσης Android, είναι το πλαίσιο εφαρμογής. Αυτό το επίπεδο παρέχει υψηλού επιπέδου δομικές μονάδες τις οποίες μπορούμε να χρησιμοποιούμε για την κατασκευή των εφαρμογών. Αυτό το πλαίσιο είναι προεγκατεστημένο στο Android, αλλά είναι επεκτάσιμο, αφού ο κάθε

κατασκευαστής μπορεί να το συμπληρώσει με δικά του κομμάτια. Τα σημαντικότερα δομικά στοιχεία του πλαισίου αυτού είναι:

- **Διαχειριστής Δραστηριοτήτων Activity Manager:** Υπεύθυνος για τον έλεγχο του χρόνου ζωής των εφαρμογών και για την διατήρηση μιας στοίβας που επιτρέπει την πλοήγηση του χρήστη σε προηγούμενες οθόνες.
- **Παροχέας Περιεχομένου -Content Providers:** Αυτά τα αντικείμενα περιέχουν δεδομένα που μπορούν να διαμοιραστούν μεταξύ εφαρμογών.
- **Διαχειριστής Πόρων-Resource Manager:** Οι πόροι είναι οτιδήποτε υπάρχει σε ένα πρόγραμμα και δεν είναι κώδικας. Για παράδειγμα μπορεί να είναι κωδικοί χρωμάτων, αλφαριθμητικοί χαρακτήρες ή ακόμα και έτοιμα σχεδιαγράμματα οθονών φτιαγμένα σε XML, τα οποία μπορεί το πρόγραμμα να καλεί.
- **Διαχειριστής Τοποθεσίας - Location Manager:** Χρησιμοποιείται για να μπορεί να ξέρει το τηλέφωνο που βρίσκεται ανά πάσα στιγμή.
- **Διαχειριστής Κοινοποιήσεων -Notification Manager:** Ιδανικός τρόπος για την ενημέρωση του χρήστη για γεγονότα που συμβαίνουν, διακριτικά χωρίς να διακόπτεται η εργασία του.

6.3 Ανάπτυξη εφαρμογών για Android

Όπως αναφέρθηκε και παραπάνω, το Android είναι ένα native λειτουργικό, το οποίο για τις εφαρμογές του χρησιμοποιεί την πλατφόρμα Android SDK. Το Android SDK παρέχει τα απαραίτητα εργαλεία για την ανάπτυξη προγραμμάτων χρησιμοποιώντας την γλώσσα προγραμματισμού Java. Τα χαρακτηριστικά του παρέχουν τεράστια ευελιξία και δυνατότητα ανάπτυξης έξυπνων εφαρμογών για κινητά τηλέφωνα, παρέχοντας δυνατότητες ανάπτυξης σε επιχειρηματίες όλων των κλάδων, είτε πρόκειται για αυτοματοποίηση πωλήσεων, ψυχαγωγία, παιχνίδια αλλά και κάθε άλλου είδους επιχειρήσεις. Ακόμα, παρέχει APIs για την χρήση web browser, εμφάνιση δισδιάστατων και τρισδιάστατων γραφικών, δομημένη αποθήκευση δεδομένων σε βάση δεδομένων, εμφάνιση πολυμεσικού υλικού (ήχος, βίντεο, εικόνες), χρήση των τεχνολογιών GSM, Bluetooth, EDGE, 3G και WiFi, χρήση συσκευών όπως φωτογραφική μηχανή, GPS, πυξίδα, επιταχυνσιόμετρο.

Ένα σημαντικό γεγονός που ευνοεί την ανάπτυξη εφαρμογών είναι πως το πακέτο Android SDK συνεργάζεται με το Eclipse και συνεπώς ο προγραμματιστής μπορεί εύκολα και γρήγορα να βλέπει τις αλλαγές του κώδικα στον προσομοιωτή που του παρέχει το Android SDK, χωρίς να χρειάζεται να εξάγει κάθε φορά την εφαρμογή και να την εγκαθιστά σε κινητό. Επίσης, ο προσομοιωτής είναι πολύ αξιόπιστος, καθώς έχει ακριβώς την ίδια συμπεριφορά με αυτή που θα είχε η εφαρμογή εάν είχε εγκατασταθεί σε ένα κινητό τηλέφωνο Android.

Ένα άλλο σημαντικό πλεονέκτημα είναι το γεγονός πως σε αναβαθμίσεις του λογισμικού, η εφαρμογή εξακολουθεί να δουλεύει χωρίς την ανάγκη επανασχεδιασμού κάποιων σημαντικών κομματιών του κώδικα, που αφορούν την αλληλεπίδραση της εφαρμογής με τα δομικά μέρη-hardware- του κινητού τηλεφώνου. Το Eclipse ως λογισμικό παρέχει δυνατότητες αυτόματης συμπλήρωσης, διόρθωσης του κώδικα γλώσσας Java που συντάσσει ο χρήστης και που προορίζεται για εφαρμογή Android. Μαζί με την εφαρμογή του Eclipse, για να μπορέσει ο

developer να «τρέξει» την προσομοίωση, θα πρέπει να κατεβάσει και το ADT Plugin του Eclipse IDE. Είναι αυτό που συμπληρώνει το IDE, για πλήρη προσομοίωση της χρήσης του κινητού Android. Για την εγκατάσταση του Eclipse, αρκεί να επισκεφτεί κανείς την παρακάτω διεύθυνση <http://developer.android.com/sdk/index.html> απ' όπου μπορεί ο εν δυνάμει developer να κατεβάσει τα απαραίτητα εργαλεία .

Κεφάλαιο 7ο: Το PhoneGap

7.1 Εισαγωγή

Το PhoneGap είναι ένα πλαίσιο εφαρμογής για κινητές συσκευές, που δημιουργήθηκε από την Adobe. Δίνει τη δυνατότητα σε δημιουργούς λογισμικού να φτιάχνουν εφαρμογές για κινητές συσκευές με χρήση Javascript, HTML5, CSS3, αντί για χρήση γλωσσών όπως η Objective C για παράδειγμα. Το αποτέλεσμα είναι υβριδικές εφαρμογές, υπό την έννοια ότι δεν είναι ούτε πλήρως native -μια και η διεπαφή τους είναι φτιαγμένη με τεχνικές που χρησιμοποιούνται στο web, αντί για το native πλαίσιο που παρέχει το PhoneGap, αλλά ούτε και αμιγώς βασισμένο σε web τεχνικές δεδομένου ότι πρόκειται για εφαρμογές που προορίζονται για δημοσίευση και διανομή, και έχουν πρόσβαση σε APIs γηγενών εφαρμογών. Το λογισμικό που υποβόσκει κάτω από το PhoneGap είναι το Apache Cordova το οποίο είναι ανοιχτού κώδικα, επομένως και το PhoneGap είναι και αυτό **ανοιχτού κώδικα**. Αν ο αναγνώστης το επιθυμεί μπορεί να διαβάσει περισσότερα ιστορικά και άλλα στοιχεία στο <http://en.wikipedia.org/wiki/PhoneGap>.



Εικόνα 40 : Το λογότυπο του PhoneGap

Το μεγάλο πλεονέκτημα της χρήσης PhoneGap είναι η δυνατότητα που παρέχει στους χρήστες να δημιουργήσουν εφαρμογές για πολλά και διαφορετικά κινητά περιβάλλοντα, που ενδεχομένως να απαιτούν γνώση χειρισμού διαφορετικών γλωσσών προγραμματισμού, χωρίς απαραίτητα να τη διαθέτουν. Αρκεί η γνώση HTML5/Javascript /CSS3 και στη συνέχεια, το **PhoneGap Build** αναλαμβάνει να επιτύχει την προσαρμογή σε διαφορετικά περιβάλλοντα (“in the cloud” όπως λέμε). Η διαφορά του PhoneGap Build είναι ότι είναι μια **cloud-based υπηρεσία** φτιαγμένη και δομημένη με βάση το πλαίσιο του PhoneGap, ενώ PhoneGap είναι το **γενικότερο** πλαίσιο ανάπτυξης κινητών εφαρμογών βασισμένο στο Apache Cordova και είναι αυτό που παρέχει τις προδιαγραφές για την ανάπτυξη των εφαρμογών σε Javascript/HTML5, και επιτρέπει την μετατροπή σε μια ευρεία γκάμα περιβαλλόντων (μεταξύ άλλων iOS, Android, Blackberry, Bada, Windows Phone, WebOS) χωρίς να χαθούν τα χαρακτηριστικά και οι δυνατότητες μιας native εφαρμογής. Στην παρακάτω εικόνα (βλ εικόνα 43) φαίνεται σχηματικά η μετατροπή αυτή.

Ειδικά κατά τη μετατροπή, δεδομένου ότι φτιάχνοντας μια εφαρμογή με μια web γλώσσα όπως η HTML5, επιχειρώντας να την μετατρέψουμε σε μια native εφαρμογή για άλλο περιβάλλον, είναι πιθανό να χαθούν κάποιες από τις δυνατότητες αυτές, καθώς δεν είναι όμοια πράγματα αυτά που εμπλέκονται στη μετατροπή. Στον παρακάτω πίνακα φαίνονται οι δυνατότητες που είναι δυνατό να διατηρηθούν και ποιες όχι κατά τη μετατροπή αναλόγως το λειτουργικό .

	iPhone / iPhone 3G	iPhone 3GS and newer	Android	Blackberry OS 6.0+	Blackberry 10	WebOS	Windows Phone 7 + 8	Symbian	Bada
Accelerometer	✓	✓	✓	✓	✓	✓	✓	✓	✓
Camera	✓	✓	✓	✓	✓	✓	✓	✓	✓
Compass	X	✓	✓	X	✓	✓	✓	X	✓
Contacts	✓	✓	✓	✓	✓	X	✓	✓	✓
File	✓	✓	✓	✓	✓	X	✓	X	X
Geolocation	✓	✓	✓	✓	✓	✓	✓	✓	✓
Media	✓	✓	✓	X	✓	X	✓	X	X
Network	✓	✓	✓	✓	✓	✓	✓	✓	✓
Notification (Alert)	✓	✓	✓	✓	✓	✓	✓	✓	✓
Notification (Sound)	✓	✓	✓	✓	✓	✓	✓	✓	✓
Notification (Vibration)	✓	✓	✓	✓	✓	✓	✓	✓	✓
Storage	✓	✓	✓	✓	✓	✓	✓	X	X

Πίνακας 3 : Δυνατότητες της HTML5 που είναι δυνατό να διατηρηθούν κατά τη μετατροπή μέσω PhoneGap

7.2 Χρήση του PhoneGap

Υπάρχουν 2 τρόποι χρήσης του PhoneGap, όπως αναφέρθηκε και παραπάνω , ανάλογα με το είδος της ανάπτυξης που ακολουθείται .

7.2.1 Απευθείας εγκατάσταση και τοπική χρήση του PhoneGap για μετατροπή εφαρμογής

Η εγκατάσταση του PhoneGap τοπικά είναι μια σχετικά απλή διαδικασία. Πηγαίνοντας ο χρήστης στην διεύθυνση www.phonegap.com, βρίσκει πάνω δεξιά στη γωνία ένα πλήκτρο εγκατάστασης (“install”), όπως φαίνεται και στην εικόνα.



Εικόνα 41: Η

εγκατάσταση του PhoneGap

Αυτό το κουμπί ανακατευθύνει σε μια σελίδα ενημέρωσης όπου ο χρήστης πρώτα ενημερώνεται ότι πρέπει να έχει εγκατεστημένο το node.js² για να μπορέσει να συνεχίσει με την εγκατάσταση και χρήση του PhoneGap. Στην ίδια σελίδα βρίσκονται και τα αρχεία εγκατάστασης του PhoneGap συμπιεσμένα σε ένα zip αρχείο. Ο χρήστης πρέπει να τα κατεβάσει στον υπολογιστή του και να τα αποσυμπιέσει για να μπορεί να γίνει χρήση των βιβλιοθηκών σε ανάπτυξη εφαρμογής. Όταν τα αποσυμπιέσει, τα αρχεία εμφανίζονται σε ένα φάκελο με υποφάκελους χωρισμένα (όπως φαίνεται στην παρακάτω εικόνα) και ονοματισμένα με βάση το λειτουργικό σύστημα το οποίο αφορούν, δεδομένου ότι το PhoneGap υποστηρίζει πολλά και διαφορετικά λειτουργικά. Σε κάθε υποφάκελο βρίσκονται τα αρχεία βιβλιοθηκών Javascript για καθένα από αυτά τα λειτουργικά. Αυτά τα αρχεία θα χρησιμοποιηθούν σε συνδυασμό με το εκάστοτε IDE που προορίζεται για κάθε ένα από τα λειτουργικά που υποστηρίζει το PhoneGap (Android, Blackberry, iOS, Windows Phone).

Στην περίπτωση του Android που μας ενδιαφέρει εδώ, υπάρχουν δύο τρόποι που μπορεί να ενεργήσει κανείς, σε 2 διαφορετικές γλώσσες προγραμματισμού: **Ο πρώτος τρόπος, αφορά τη γλώσσα Java.** Πρώτα πρέπει να γίνει η εγκατάσταση του Eclipse IDE ή κάποιου άλλου συντάκτη για κώδικα- το οποίο θα πραγματοποιήσει και την μεταγλώττιση του κώδικα Java που απαιτείται μιας και σ' αυτή την περίπτωση θα πρέπει να γράψουμε εκ νέου την βασική εφαρμογή μας, καθώς επίσης και του ADT (Android Development Tool) plugin για το Eclipse. Επίσης απαραίτητα πρέπει να γίνει η εγκατάσταση του Android SDK, το οποίο θα εκτελέσει την προσομοίωση σε H/Y. Στη συνέχεια, θα πρέπει να συμπεριληφθούν τα αρχεία που έχουμε κατεβάσει και εξάγει από την σελίδα του PhoneGap και που προορίζονται για Android, στο δέντρο των αρχείων της εφαρμογής, στον αρχικό φάκελο όπου τοποθετούνται όλα τα αρχεία από το IDE. Αν ο αναγνώστης επιθυμεί να εμβαθύνει σε αυτή τη διαδικασία, μπορεί να μάθει περισσότερα εδώ : <http://www.adobe.com/devnet/html5/articles/getting-started-with-phonegap-in-eclipse-for-android.html>

Ο δεύτερος τρόπος αφορά κώδικα σε HTML/Javascript. Για να πραγματοποιηθεί αυτή η διαδικασία, χρειάζεται ένας συντάκτης κειμένου ή ένα πρόγραμμα για συγγραφή κώδικα HTML/Javascript όπως για

² Το node.js είναι μια πλατφόρμα για ανάπτυξη λογισμικού που κατασκευάστηκε για να παρέχει ένα εύκολο τρόπο δημιουργίας κλιμακωτών διαδικτυακών εφαρμογών. Στηρίζεται σε ένα μοντέλο ασύγχρονης επικοινωνίας εισόδου /εξόδου. Για περισσότερες λεπτομέρειες ο αναγνώστης μπορεί να ανατρέξει στη σελίδα <http://en.wikipedia.org/wiki/Nodejs>

παράδειγμα το Dreamweaver CS5 ή CS6, το οποίο υποστηρίζει κιάλας το συντακτικό του PhoneGap. Το συντακτικό αυτό λοιπόν, Javascript στην ουσία, χρησιμοποιεί βιβλιοθήκες έτοιμες που συμπεριλαμβάνονται στα αρχεία που εγκαθίστανται κατά το κατέβασμα από τη σελίδα του PhoneGap. Αυτά τα αρχεία περιέχουν τις βιβλιοθήκες που θα συμπεριληφθούν στο αρχείο HTML, και αναλόγως την εφαρμογή που θέλουμε να φτιάξουμε θα χρησιμοποιήσουμε τα ανάλογα εργαλεία από αυτές. Έχουν γραφτεί βιβλία ολόκληρα πάνω στο PhoneGap και τις δυνατότητες που προσφέρει πρόσθετα στις δυνατότητες της Javascript, και που αφορούν την ανάπτυξη εφαρμογών για κινητά περιβάλλοντα. Αν ο αναγνώστης επιθυμεί να μάθει περισσότερα, μπορεί να βρει πληροφορίες εδώ : <http://www.adobe.com/inspire/2012/02/combining-web-skills-phonegap-mobile-apps.html>, και εδώ: <http://pdf.th7.cn/download/files/1312/PhoneGap%20Mobile%20Application%20Development%20Cookbook.pdf>

Είναι σημαντικό να τονιστεί ότι τα αρχεία προέλευσης αυτά είναι διαφορετικά μεταξύ τους, και δε μπορεί να χρησιμοποιηθούν οι βιβλιοθήκες PhoneGap του ενός λειτουργικού για ανάπτυξη εφαρμογής σε κάποιο άλλο, πράγμα που καθιστά την ανάπτυξη εφαρμογών με PhoneGap πολύπλοκη διαδικασία, και έτσι δημιουργείται η ανάγκη για τη χρήση του PhoneGap Build, το οποίο και θα χρησιμοποιήσουμε.

7.2.2.Μετατροπή μέσω PhoneGap Build.

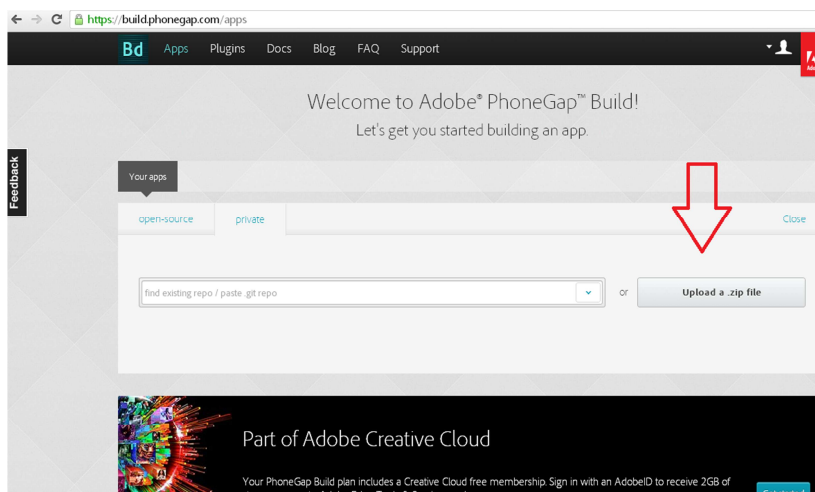
Το PhoneGap δίνει τη δυνατότητα για δωρεάν μετατροπή ιδιωτικής εφαρμογής (δηλ μιας εφαρμογής που δεν έχει ανεβάσει στο **GitHub**³) αλλά μόνο για μια εφαρμογή. Δηλαδή δεν είναι δυνατό κάποιος να χρησιμοποιήσει το PhoneGap Build πάνω από μια φορές δωρεάν, είτε η εφαρμογή του αποσκοπεί σε οικονομικό όφελος είτε χωρίς οικονομικό όφελος. Για να μπορέσει ο χρήστης να χρησιμοποιήσει το PhoneGap Build παραπάνω φορές για ιδιωτική εφαρμογή, πράγμα που σημαίνει ότι ασχολείται επαγγελματικά, θα πρέπει πρώτον, να πληρώσει κάποιο ποσό στο PhoneGap της τάξεως των 10 δολαρίων, καθώς και να πραγματοποιήσει «εγγραφή», και δεύτερον να είναι χρήστης του GitHub. Από κει και πέρα έχει τη δυνατότητα να μετατρέψει με χρήση του PhoneGap Build έως και 25 ιδιωτικές εφαρμογές, και απεριόριστες δημόσιες με χρήση του GitHub. Δηλαδή αφού ο κώδικάς του «ανεβεί» στο GitHub (οπότε και θεωρείται δημόσια η εφαρμογή), τότε με το σύνδεσμο εκείνο, κάνει τη μετατροπή μέσω του PhoneGap.

Η λειτουργία του PhoneGap Build είναι πολύ απλή και γλυτώνει τον developer από εκμάθηση βιβλιοθηκών Javascript και άλλες περίπλοκες διαδικασίες. Αρχικά από την κεντρική σελίδα του PhoneGap χρειάζεται να γίνει εισαγωγή του χρήστη είτε μέσω λογαριασμού του στο GitHub (όπως αναφέρθηκε και παραπάνω) ή μέσω Adobe ID. Αν ο ενδιαφερόμενος developer δεν έχει τίποτα από τα δύο, τότε θα πρέπει να δημιουργήσει, διαφορετικά δε μπορεί να χρησιμοποιήσει την υπηρεσία.

Η υπηρεσία δραστηριοποιείται “in the cloud” δηλαδή με τη χρήση σύνδεσης Internet, που θα συνδέσει το χρήστη με το server, απομακρυσμένα. και όχι τοπικά. Επομένως δε χρειάζεται εγκατάσταση κάποιου λογισμικού,

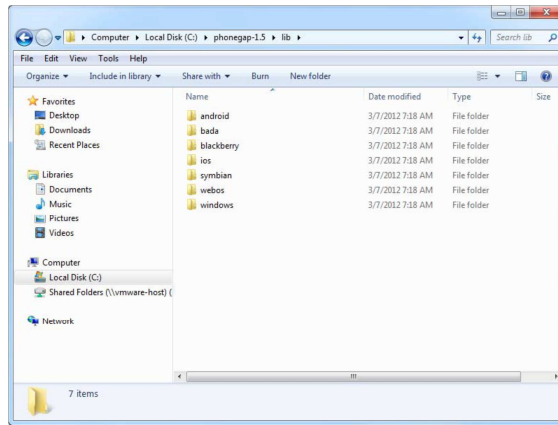
³ Το GitHub είναι ένα εργαλείο κοινωνικής δικτύωσης όπου οι χρήστες μπορούν να δημιουργήσουν το προσωπικό τους προφίλ και να «ανεβάσουν» τον κώδικά τους με σκοπό να το μοιραστούν με διαδικτυακούς φίλους που ενδιαφέρονται για αυτό τον κλάδο, αλλά και να μελετούν τη δουλειά άλλων χρηστών, αντίστοιχα. Για περισσότερες πληροφορίες και λεπτομέρειες, ο αναγνώστης μπορεί να επισκεφθεί τη σελίδα <http://bit.ly/1n5mGNr>

ούτε το «κατέβασμα» αρχείων όπως συμβαίνει με την τοπική χρήση του Phonegap. Η βασικότερη προϋπόθεση όλων των παραπάνω, βασικότερη και από τη δημιουργία λογαριασμού GitHub, είναι να έχει ήδη δημιουργήσει την εφαρμογή του σε γλώσσα HTML5/Javascript. Δηλαδή χρειάζεται να υπάρχει ήδη εφαρμογή με αρχεία .html,.js δημιουργημένη είτε σε κάποιον επεξεργαστή κειμένου, είτε με κάποιο άλλο εργαλείο ανάπτυξης. Στη συνέχεια θα πρέπει αυτά τα αρχεία να είναι είτε συμπιεσμένα σε ένα αρχείο zip για να μπορούν να «ανέβουν» ως ένα αρχείο στο PhoneGap Build, είτε να έχει δημιουργηθεί ήδη ως project στο GitHub, και άρα να «περαστεί» μέσω του GitHub στο PhoneGap Build.

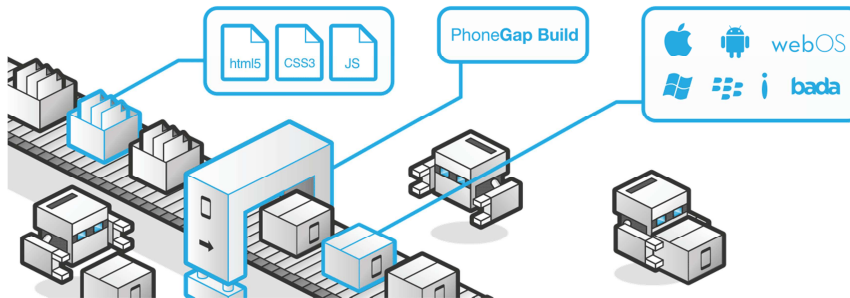


Εικόνα 42: Η σελίδα του PhoneGap όπου μπορούμε να ανεβάσουμε την εφαρμογή μας προς μετατροπή μέσω PhoneGap Build.

Από τη στιγμή που θα περαστεί η εφαρμογή στο PhoneGap Build, το επόμενο βήμα που απαιτείται να γίνει είναι η επιλογή ως προς το είδος του λειτουργικού που θα «τρέχει» η εφαρμογή που θα παρθεί κατά την έξοδο, δεδομένου ότι το PhoneGap Build έχει τη δυνατότητα δημιουργίας εφαρμογών για ένα πλήθος λειτουργικών για κινητά περιβάλλοντα, μεταξύ άλλων και το Android. Αφού γίνει η επιλογή, χρειάζονται μερικά λεπτά για να επιτευχθεί η μετατροπή. Κατά την έξοδο θα παρθεί αρχείο .apk προς εγκατάσταση σε συσκευή smartphone που «φορά» λειτουργικό Android.



Εικόνα 43 : Το αποσυμπερισμένο αρχείο τοπικής εγκατάστασης του PhoneGap



Εικόνα 44 : Σχηματική απεικόνιση της διεργασίας που επιτελεί το PhoneGap Build.

Κεφάλαιο 8ο :Μια εφαρμογή επεξεργασίας εικόνας σε HTML5 /Javascript

8.1 Εισαγωγή

Για να έχουμε πλήρη εικόνα των δυνατοτήτων της HTML5, ειδικά σε κινητές συσκευές, πρέπει να τη χρησιμοποιήσουμε πάνω σε κάτι πολύ συγκεκριμένο, σε μια εφαρμογή για παράδειγμα, που θα έχει ως βασικό εργαλείο υλοποίησης την HTML5 σε συνδυασμό με την Javascript. Επομένως μια εφαρμογή για επεξεργασία εικόνας, με εικόνα επιλεγμένη δυναμικά από το χρήστη, που θα χρησιμοποιούσαμε την ετικέτα <canvas> για την εφαρμογή των φίλτρων, θα αναδείκνυε τις δυνατότητες αυτές πλήρως. Αρχικά, θα δούμε την εφαρμογή ως βασική υλοποίηση σε γλώσσα HTML5 /Javascript και στην πορεία παρακάτω, θα προσαρμόσουμε αυτή την υλοποίηση σε 3 διαφορετικά περιβάλλοντα για κινητές συσκευές ξεχωριστά (Tizen , Firefox, Android).

8.2 Η βασική υλοποίηση σε HTML5/Javascript

Για την υλοποίηση της εφαρμογής μας, το πρώτο που πρέπει να υλοποιηθεί είναι τα divs στο βασικό αρχείο της HTML ώστε να αποκτήσει η εφαρμογή μας την μορφοποίηση που χρειάζεται. Αυτό σημαίνει ότι χρειαζόμαστε έναν αριθμό από divs. Ξεχωριστό για την επικεφαλίδα, ξεχωριστό για το μενού της εφαρμογής, άλλο για να λειτουργήσει ως υποδοχή του καμβά, και ξεχωριστό div ως footer. Μπορεί να φαίνονται πολλά για μια απλή εφαρμογή, αλλά είναι απαραίτητα για την σωστή εμφάνιση των επιμέρους στοιχείων της εφαρμογής. Χρησιμοποιούνται 4 divs συνολικά.

```

<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8"/>

  <title>Image processing Application</title>

  <!--NOTE:
  Additional scripts and css files are to be placed here.
  You can use jQuery namespace($) and all functionalities in jQuery
  in your script. For example:

  <script src="main.js"></script>
  <link rel="stylesheet" href="my.css">
  -->
  <script type="text/javascript" src="./js/my_main.js">
</script>
  <script type="text/javascript" src="./js/filters.js">
</script>
  <link rel="stylesheet" type="text/css" href="./css/style.css"/>
<script class="jsbin" src="http://ajax.googleapis.com/ajax/libs/jquery/1/jquery.min.js"></
<script class="jsbin" src="http://ajax.googleapis.com/ajax/libs/jqueryui/1.8.0/jquery-ui.m
<style>
  article, aside, figure, footer, header, hgroup,
  menu, nav, section { display: block; }
</style>

```

Εικόνα 45:

Ο κώδικας HTML που χρησιμοποιήθηκε για την εφαρμογή (1)

```

</head>
<body >
  <!-- Start of first page: #one -->
  <div data-role="page" id="one" >
    <div id="header" data-position="fixed">
      <center><h2><font color="white">Image.Process.Applications</h2></center>
    </div>

    <!-- /header -->

    <div data-role="content">
      <div id="menu_div" align="left">
        <table cellpadding="1" cellspacing="5" width="100">
          <tr>
            <td> <a href="javascript:setFilter(grayscale);">Grayscale</a></td>

          </tr>
          <tr>
            <td> <a href="javascript:setFilter(sepia);">Sepia</a></td>

          </tr>
          <tr><td> <a href="javascript:setFilter(red);">Red</a></td>
          </tr>
          <tr><td> <a href="javascript:setFilter(brightness(30));">Brighter</a></td></tr>
          <tr><td><a href="javascript:setFilter(brightness(-30));">Darker</a></td>
          </tr>
          <tr><td><a href="javascript:undoDrawOnCanvas();">Undo Image</a></td></tr>

        </table></div>
      <div id="image_placing" position="right" >
        <center>...

```

Εικόνα 46: Ο κώδικας HTML που χρησιμοποιήθηκε για την εφαρμογή (2)

Βλέπουμε ότι για τα φίλτρα χρησιμοποιήθηκαν κλήσεις συναρτήσεων της javascript.

```

<input type="file" id="input" >
<canvas id="canvas" width="300" height="305"></canvas>
</center>

</div>

<!-- /content -->

</div><!-- /page one -->
<div id="footer" bottom="0px;" position="fixed">
<i><font color="white" > This is an application for image processing that has been develop
</div>
</div>

</body>
</html>

```

Εικόνα

47 : Ο κώδικας HTML που χρησιμοποιήθηκε για την εφαρμογή (3)

```

var restorePoints = [];
function saveRestorePoint() {
    var oCanvas = document.getElementById("canvas");
    var imgSrc = oCanvas.toDataURL("image/png");
    restorePoints.push(imgSrc);
}

function undoDrawOnCanvas() {
    if (restorePoints.length > 0) {
        var oImg = new Image();
        oImg.onload = function() {
            var canvasContext = document.getElementById("canvas").getContext("2d");
            canvasContext.drawImage(oImg, 0, 0);
        }
        oImg.src = restorePoints.pop();
    }
}

function handleFiles(e) {
    var can = document.getElementById('canvas');
    var ctx = can.getContext('2d');
    var url = URL.createObjectURL(e.target.files[0]);
    var img = new Image();
    img.onload = function() {

```

Εικόνα 48 : Ο κώδικας Javascript που χρησιμοποιήθηκε για την εφαρμογή (1).

Βλέπουμε επίσης ότι έχει χρησιμοποιηθεί κλήση από στοιβία στην saveRestorePoint() και στην undoDrawOnCanvas() για την υλοποίηση της ανίερσης.


```

img.onload = function() {
img.width = can.width;
img.height = can.height;
ctx.drawImage(img, 0, 0, img.width, img.height);
}
img.src = url;
}

var URL = window.webkitURL || window.URL;

window.onload = function() {
var input = document.getElementById('input');
input.addEventListener('change', handleFiles, false);
}

function recall(){
ctx.restore();
}

//set filter function
function setFilter(filter) {
var canvas= document.getElementById('canvas');
var ctx = canvas.getContext('2d');
if (canvas.width > 0 && canvas.height > 0) {
var imageData = ctx.getImageData(0, 0, canvas.width, canvas.height);
filter(imageData);
ctx.putImageData(imageData, 0, 0);
}
}

```

Εικόνα 49: Ο κώδικας Javascript που χρησιμοποιήθηκε για την εφαρμογή (2)

Βλέπουμε ότι γίνεται φόρτωση της εικόνας από path του χρήστη δυναμικά μέσω eventListener. Επίσης βλέπουμε συνάρτηση για την εφαρμογή φίλτρου στο κάτω μέρος της εικόνας.

```

grayscale = function (pixels, args)
{
var d = pixels.data;

for (var i = 0; i < d.length; i += 4)
{
var r = d[i];
var g = d[i + 1];
var b = d[i + 2];
d[i] = d[i + 1] = d[i + 2] = (r+g+b)/3;
}
saveRestorePoint();
return pixels;
};

sepia = function (pixels, args)
{
var d = pixels.data;

for (var i = 0; i < d.length; i += 4)
{
var r = d[i];
var g = d[i + 1];
var b = d[i + 2];
d[i] = (r * 0.393)+(g * 0.769)+(b * 0.189);
d[i + 1] = (r * 0.349)+(g * 0.686)+(b * 0.168);
d[i + 2] = (r * 0.272)+(g * 0.534)+(b * 0.131);
}
saveRestorePoint();

return pixels;
};

```

Εικόνα 50 : Ο κώδικας Javascript που χρησιμοποιήθηκε για την εφαρμογή των φίλτρων (3).

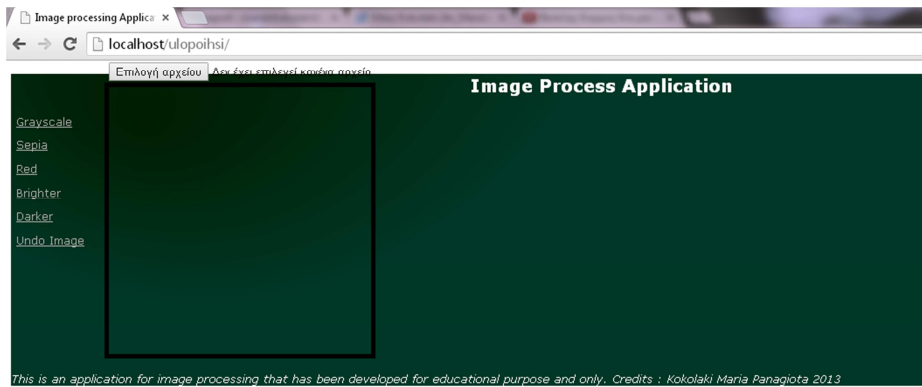
```
red = function (pixels, args)
{
var d = pixels.data;
for (var i = 0; i < d.length; i += 4)
{
var r = d[i];
var g = d[i + 1];
var b = d[i + 2];
d[i] = (r+g+b)/3;
d[i + 1] = d[i + 2] = 0;
}
saveRestorePoint();

return pixels;
};

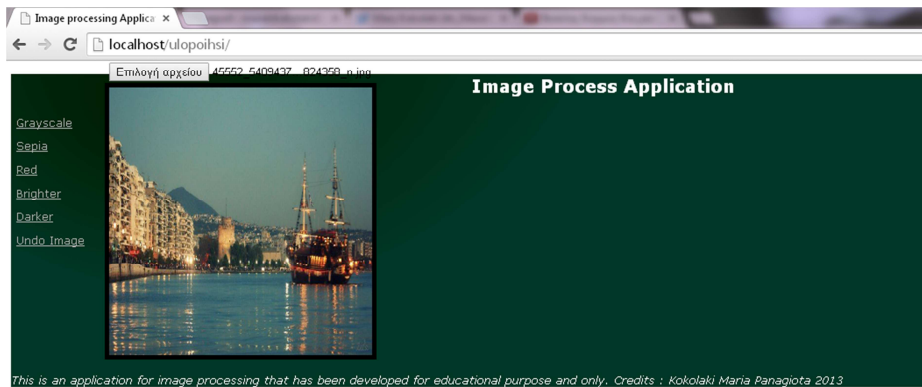
brightness = function(delta)
{
return function (pixels, args)
{
var d = pixels.data;
for (var i = 0; i < d.length; i += 4)
{
d[i] += delta;
d[i + 1] += delta;
d[i + 2] += delta;
}
saveRestorePoint();

return pixels;
};
};
```

Εικόνα 51 : Ο κώδικας Javascript που χρησιμοποιήθηκε για την εφαρμογή των φίτρων (4)



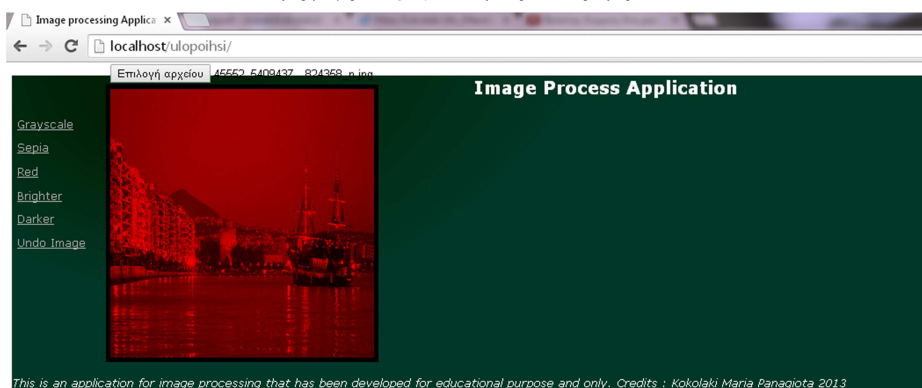
Εικόνα 52 : Στιγμιότυπο από τη φόρτωση της εφαρμογής μέσω wamp server



Εικόνα 53 :Στιγμιότυπο από τη φόρτωση εικόνας στην εφαρμογή



Εικόνα 54: Εφαρμογή του greyscale φίλτρου στην φορτωθείσα εικόνα



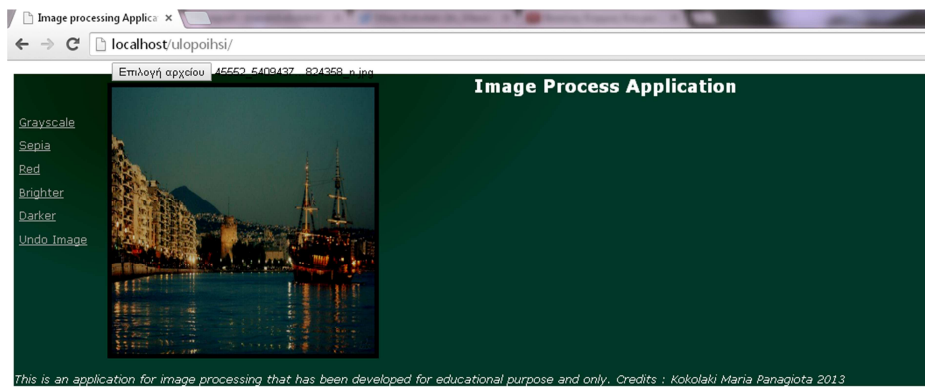
Εικόνα 55:Εφαρμογή του red φίλτρου στην φορτωθείσα εικόνα



Εικόνα 56: Εφαρμογή του sepia φίλτρου στην εικόνα



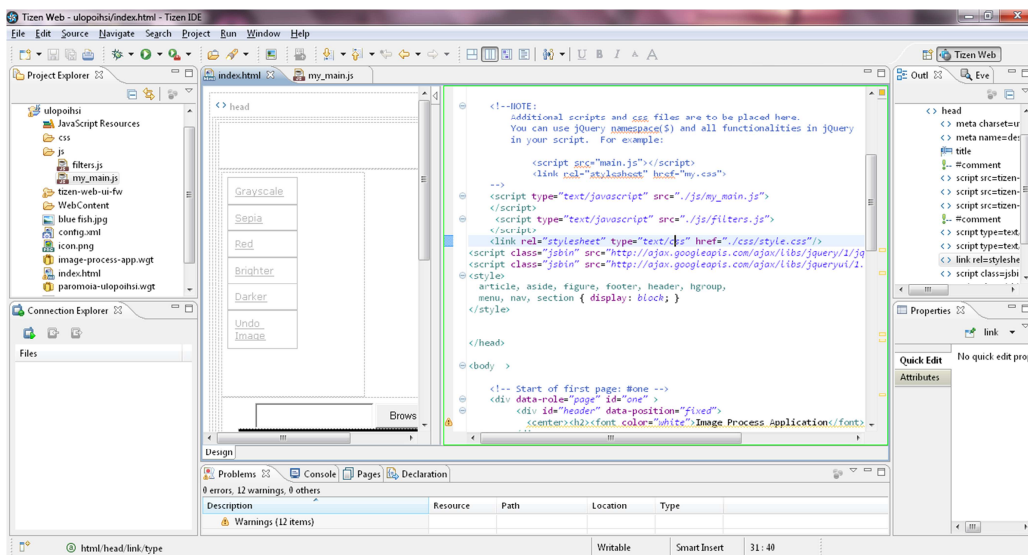
Εικόνα 57 :Στιγμιότυπο της εφαρμογής με εφαρμοσμένο το φίλτρο brighter 2 φορές



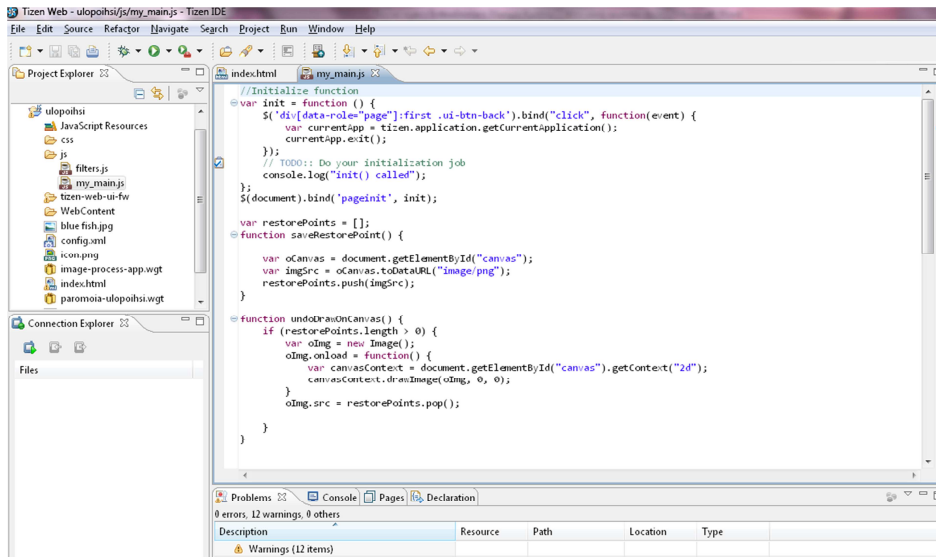
Εικόνα 58 :Στιγμιότυπο της εφαρμογής με εφαρμοσμένο το φίλτρο darker 2 φορές

8.3 Παραμετροποίηση της βασικής εφαρμογής για περιβάλλον Tizen

Αφού είδαμε τη βασική υλοποίηση σε HTML5, για να προσαρμοστεί η εφαρμογή μας στα δεδομένα του Tizen IDE, πρέπει να σβήσουμε τον έτοιμο κώδικα με τα divs και το κείμενο της demo εφαρμογής που μας βάζει το IDE όταν πρωτοφτιάχνουμε το project, και να βάλουμε τον δικό μας κώδικα με τα δικά μας divs πάνω στα divs του Tizen IDE που θα επιλέξουμε να κρατήσουμε. Με άλλα λόγια, να επιλέξουμε ποια κομμάτια του έτοιμου κώδικα θα κρατήσουμε και να σβήσουμε τα περιττά. Αυτό πρέπει να γίνει στο αρχείο με κατάληξη .html, επίσης στα αρχεία .js και .css που θα «τοποθετήσουμε» στο δέντρο των αρχείων του Tizen IDE. Πέραν αυτής της μικρής προσαρμογής, δεν απαιτείται κάποια άλλη προσθήκη αρχείου, ή κάτι άλλο προκειμένου να «τρέξει» η εφαρμογή μας σε περιβάλλον Tizen. Στην παρακάτω εικόνα βλέπουμε το αρχείο index.html με τις προσαρμογές που κάναμε στον κώδικά μας:

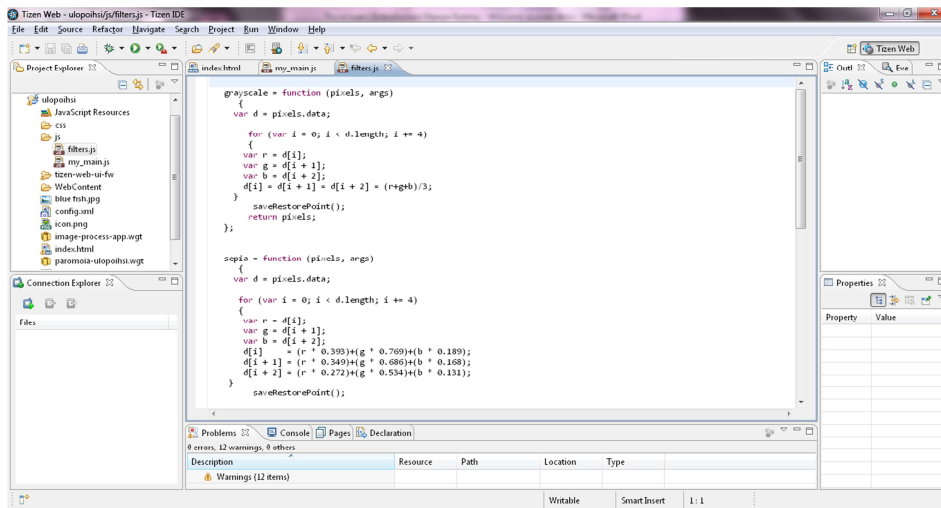


Εικόνα 59: Το αρχείο index.html της εφαρμογής.

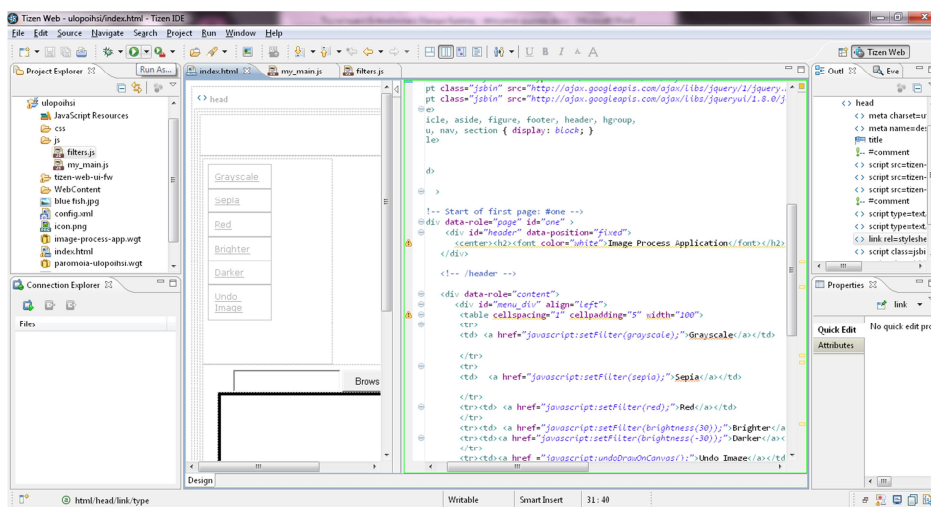


Εικόνα 60:Το αρχείο της main σε javascript.

Στην εφαρμογή αυτή έχουμε ένα αρχείο .js με το όνομα main.js το οποίο περιέχει τον κώδικα με τις συναρτήσεις για την επιλογή και εμφάνιση της εικόνας, την προσαρμογή του μεγέθους της στον καμβά, καθώς επίσης και συναρτήσεις για εφαρμογή φίλτρων και επαναφορά της εικόνας. Επίσης, έχουμε ακόμη ένα αρχείο .js με το όνομα filters.js το οποίο περιέχει τον κώδικα των συναρτήσεων των φίλτρων χωριστά. Αυτό συμβαίνει για να μην έχουμε το φαινόμενο «σεντόνι», δηλαδή κώδικα με μεγάλη έκταση συγκεντρωμένο σε ένα αρχείο, το οποίο είναι πολύ κουραστικό στην ανάγνωση και στη διόρθωση.

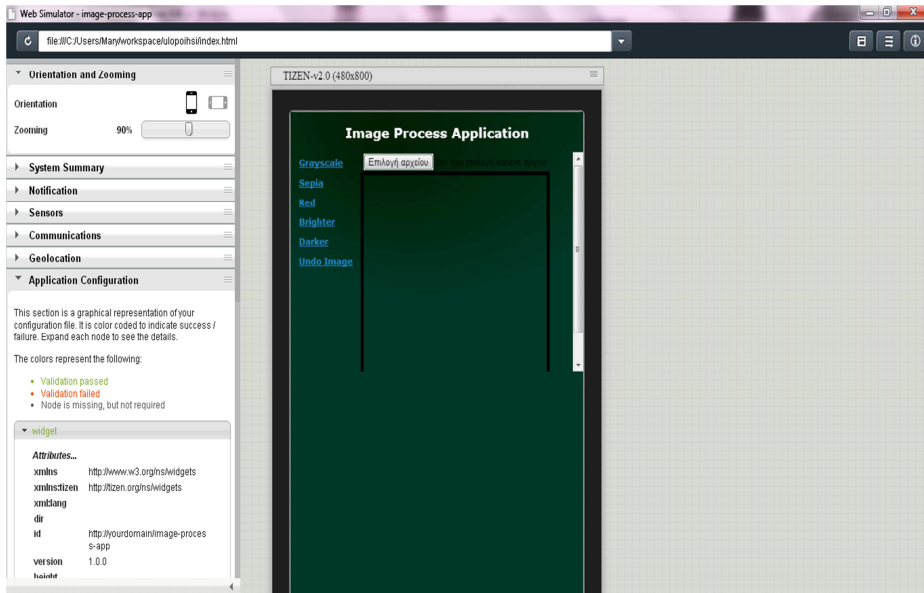


Εικόνα 61:Το αρχείο filters.js

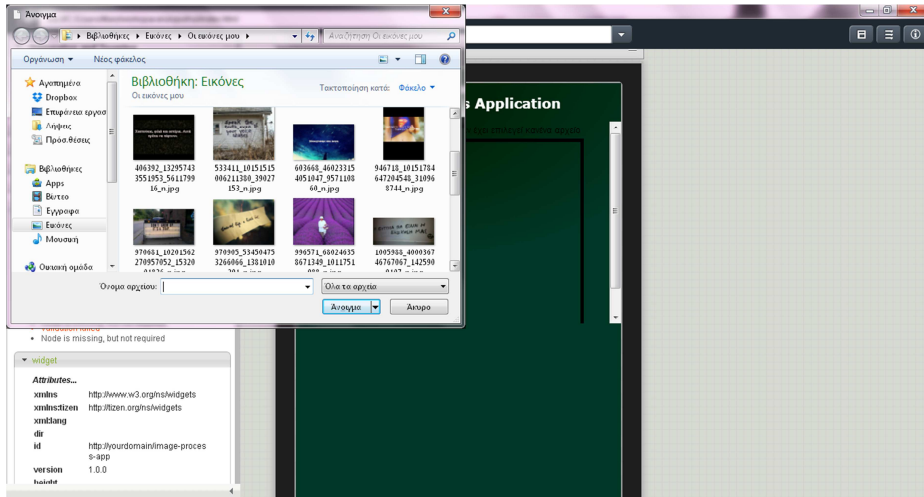


Εικόνα 62:Η εφαρμογή τρέχει μέσω προσομοιωτή.

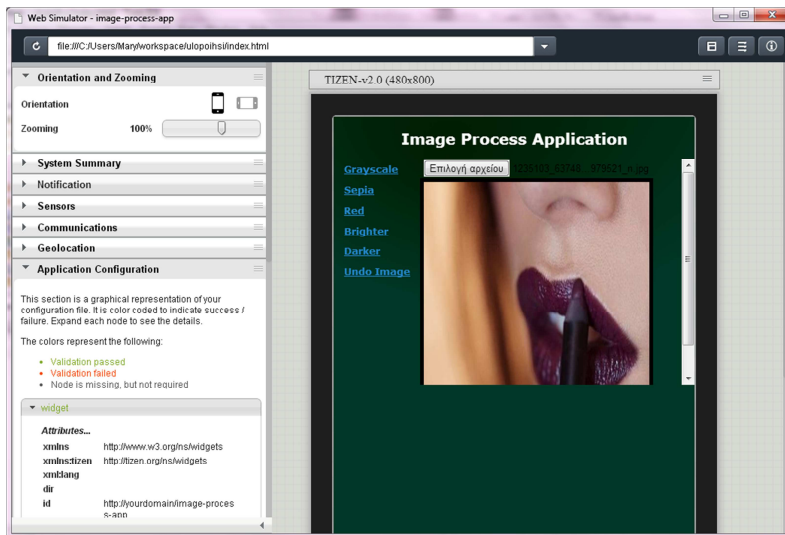
Επιλέγοντας “Run As-> Tizen Web Simulator Application” η εφαρμογή τρέχει μέσω προσομοιωτή.



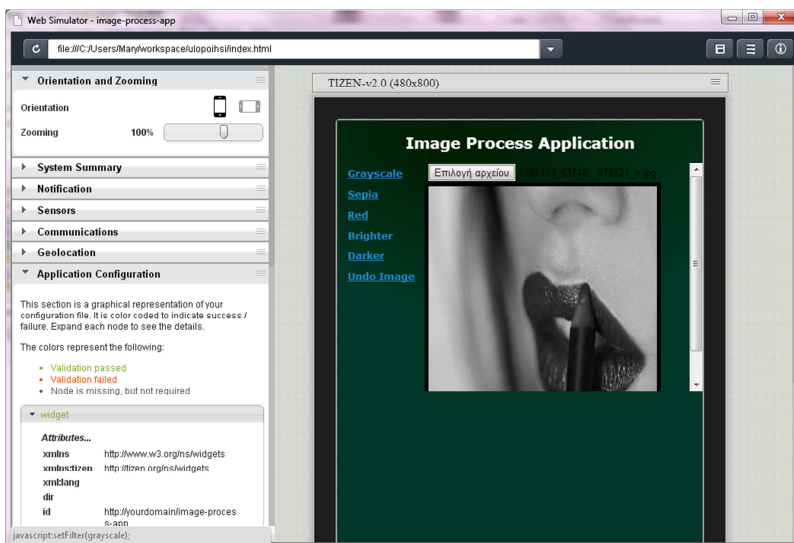
Εικόνα 63: Η αρχική διεπαφή της εφαρμογής διαμορφωμένη μέσω divs .



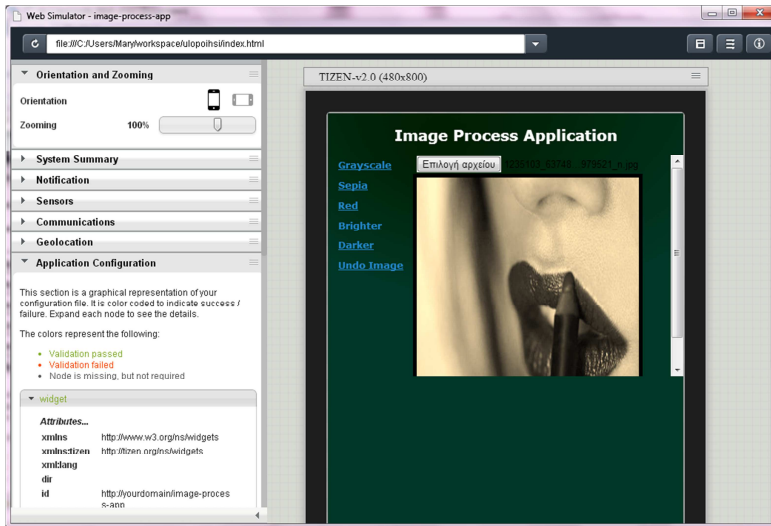
Εικόνα 64: Άνοιγμα παραθύρου για επιλογή εικόνας μέσω windows.



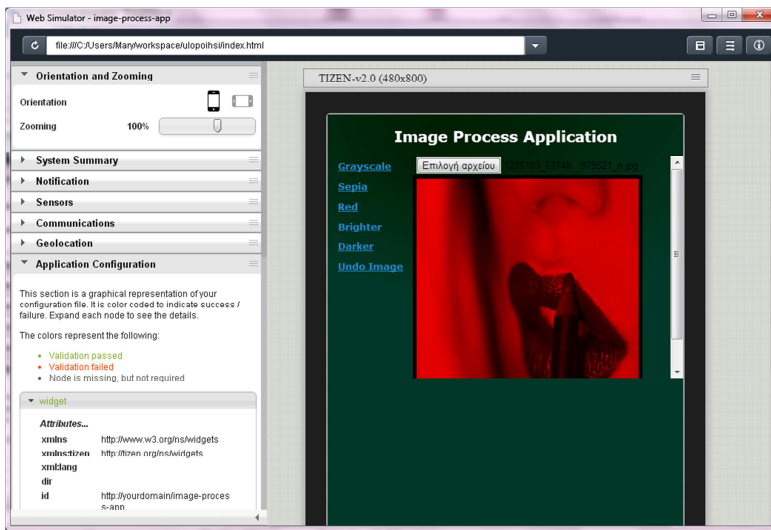
Εικόνα 65: Η αρχική εικόνα που φορτώνουμε στην εφαρμογή



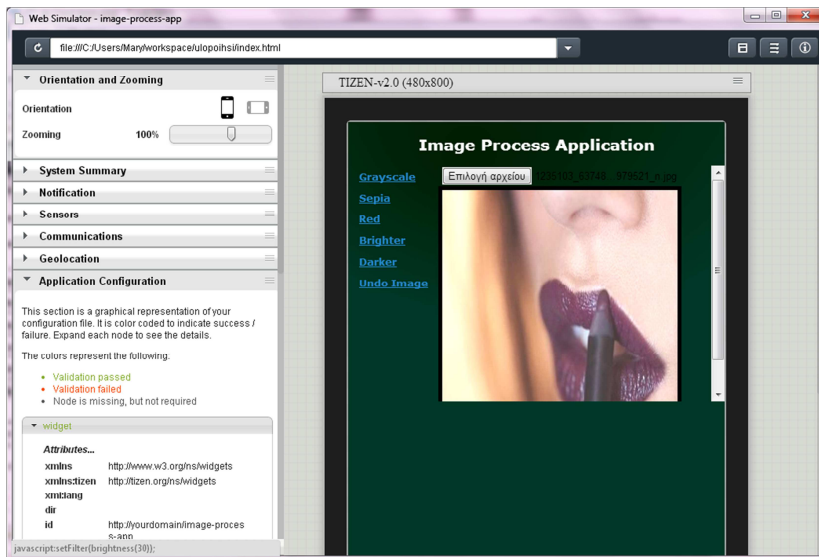
Εικόνα 66: Η εφαρμογή του πρώτου φίλτρου (greyscale)



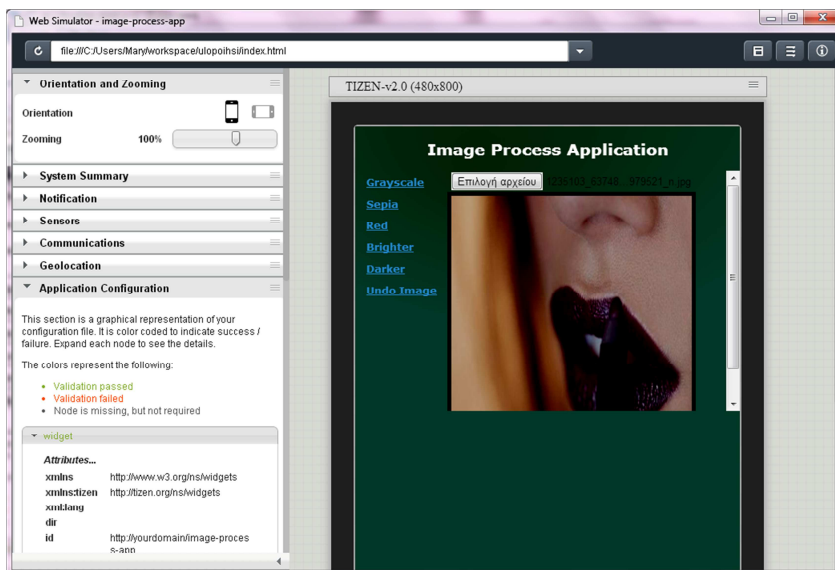
Εικόνα 67: Η εφαρμογή του δεύτερου φίλτρου (sepia)



Εικόνα 68 : Η εφαρμογή του τρίτου φίλτρου (red)



Εικόνα 69: Η εφαρμογή του τέταρτου φίλτρου (φωτεινότητα-brighter) εφαρμοσμένο 2 φορές.

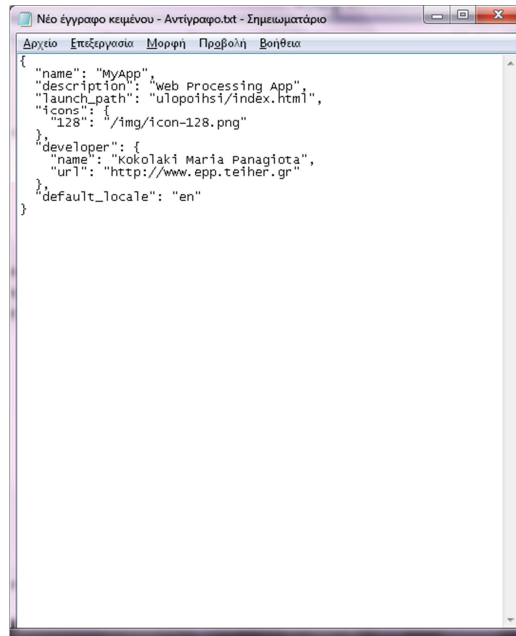


Εικόνα 70: Η εφαρμογή του πέμπτου φίλτρου (darker) εφαρμοσμένο 2 φορές.

8.4 Παραμετροποίηση της βασικής εφαρμογής για Firefox OS

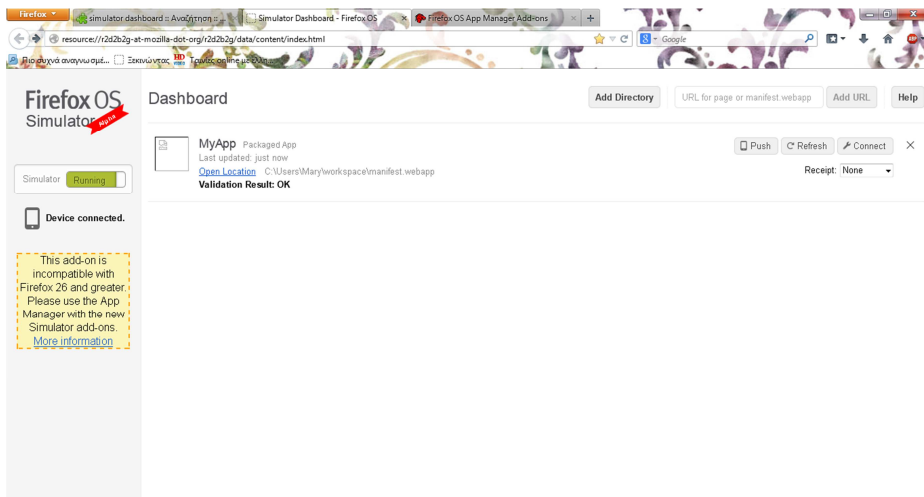
Όπως είδαμε και παραπάνω, οι packaged εφαρμογές για Firefox OS, για να τρέξουν μέσω dashboard, χρειάζονται ένα αρχείο manifest.webapp. Αυτό το αρχείο που περιέχει πληροφορίες για την εφαρμογή, είναι το **μόνο** πράγμα που χρειάζεται να συντάξουμε επιπλέον στα αρχεία .html, .js, .css της packaged εφαρμογής. Στην παρακάτω εικόνα φαίνεται το αρχείο manifest.webapp για την εφαρμογή της επεξεργασίας εικόνας. Αυτό το αρχείο, όταν μετονομαστεί σε manifest.webapp δεν είναι πια προσπελάσιμο μέσω προγράμματος σύνταξης κειμένου. Θα πρέπει να μπει μαζί με το φάκελο που περιέχει τα αρχεία της εφαρμογής και αυτά τα 2 μόνα τους να συμπεστούν. Θα πρέπει όμως αυτό το αρχείο manifest να υπάρχει και στο root directory που βρίσκεται ο συμπεσμένος φάκελος, **ασυμπέστο** εκεί.

Το αρχείο αυτό περιέχει κάποιες βασικές πληροφορίες για την εφαρμογή, όπως το όνομα ης εφαρμογής, μια περιγραφή, το μονοπάτι που βρίσκεται η εφαρμογή, γλώσσα κ.ά. Για **packaged** εφαρμογές τα υποχρεωτικά πεδία είναι το όνομα, η περιγραφή και το μονοπάτι και όλα τα υπόλοιπα είναι προαιρετικά. Συγκεκριμένα για packaged εφαρμογές, το πεδίο μονοπάτι συγκεκριμενοποιεί το σημείο αναφοράς σε σχέση με τα αρχεία περιεχομένου της εφαρμογής (δηλ το συμπεσμένο φάκελο), το οποίο θεωρείται και root directory. Για παράδειγμα αν ο φάκελος με τα αρχεία της εφαρμογής (με το όνομα uloroihsi) βρίσκεται συμπεσμένος μέσα σε ένα άλλο φάκελο workspace,μαζί με το manifest αρχείο, τότε στο πεδίο μονοπάτι (launch_path) θα βάλουμε το πλήρες μονοπάτι και όχι μόνο το όνομα του κεντρικού αρχείου html της εφαρμογής όπως φαίνεται και στην εικόνα. Αφού τα αρχεία συμπεστούν και τοποθετηθούν κατά αυτό τον τρόπο, τότε μέσω του Firefox Dashboard, «τρέχουμε» την εφαρμογή μας. Πατώντας το κουμπί “Add Directory” επιλέγουμε από το παράθυρο που μας ανοίγει το αρχείο manifest της εφαρμογής μας (το ασυμπέστο). Αφού επιλεγεί, ο dashboard κάνει επικύρωση (validation) ώστε να ελεγχθεί η εγκυρότητά του και εμφανίζει από κάτω τα αποτελέσματα του ελέγχου αυτού όπως φαίνεται και στην εικόνα 71.



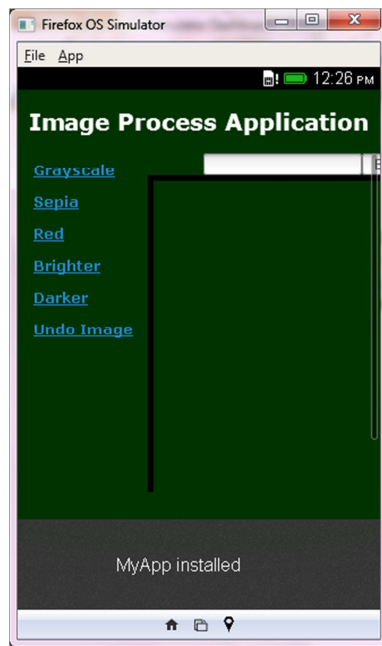
```
Νέο έγγραφο κειμένου - Αντίγραφο.txt - Σημειωματάριο
Αρχείο  Επεξεργασία  Μορφή  Προβολή  Βοήθεια
{
  "name": "MyApp",
  "description": "Web Processing App",
  "launch_path": "υλοποιησι/index.html",
  "icons": {
    "128": "/img/icon-128.png"
  },
  "developer": {
    "name": "Κοκολάκη Μαρία Παναγιώτα",
    "url": "http://www.epp.teiker.gr"
  },
  "default_locale": "en"
}
```

Εικόνα 71: Το αρχείο manifest.webapp της εφαρμογής



Εικόνα 72: Ο Dashboard αφού έχει ελέγξει το αρχείο manifest για την εγκυρότητά του.

Για να «τρέξουμε» την εφαρμογή πατάμε το κουμπί Connect στα δεξιά, και έτσι κάνουμε προσομοίωση της εφαρμογής μας σε περιβάλλον Firefox OS. Το πρώτο πράγμα που θα μας εμφανίσει είναι η αρχική σελίδα της εφαρμογής μας, δηλαδή το index.html αρχείο.

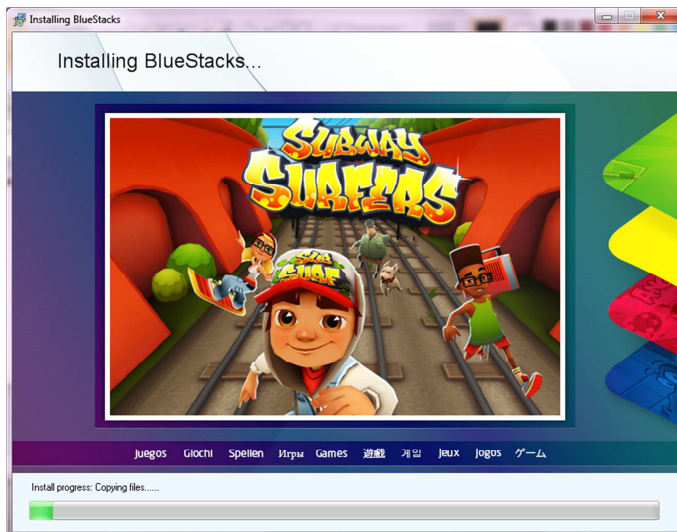


Εικόνα 73: Η αρχική σελίδα της εφαρμογής μας σε περιβάλλον FirefoxOS

8.5 Παραμετροποίηση της εφαρμογής για Android/PhoneGap Build

Όπως είδαμε, για να κάνουμε μετατροπή μιας εφαρμογής HTML5/Javascript μπορούμε για μέγιστη διευκόλυνση αυτής της εργασίας, να χρησιμοποιήσουμε το PhoneGap Build το οποίο θα κάνει τη μετατροπή αυτή για μας. Στην είσοδό του θα βάλουμε τα αρχεία της εφαρμογής (δηλ το html αρχείο ,τα 2 js και το ένα css συμπεριλαμβανόμενα), για να πάρουμε στην έξοδο αρχείο .apk δηλαδή εφαρμογή έτοιμη να εγκατασταθεί σε συσκευή Android. Σε αυτή την εργασία θα χρησιμοποιήσουμε και εδώ προσομοίωση σε Η/Υ μιας και δεν υπάρχει η δυνατότητα να εγκατασταθεί σε συσκευή Android. Για να γίνει αυτό, χρειαζόμαστε μια εφαρμογή για εγκατάσταση του αρχείου apk στον Η/Υ που λέγεται BlueStacks. Το BlueStacks εγκαθιστά στον υπολογιστή μας ο,τι εφαρμογές μπορεί να έχουμε στο Android κινητό μας μεταξύ άλλων και παιχνίδια, και μπορούμε να το

κατεβάσουμε από τη σελίδα <http://www.bluestacks.com/>. Η εγκατάσταση είναι μια πάρα πολύ απλή διαδικασία που δεν περιλαμβάνει ούτε καν Setup Wizard, όμως απαιτεί σύνδεση στο Internet. Παρακάτω φαίνονται εικόνες από την διαδικασία εγκατάστασης του Bluestacks, η οποία γίνεται με χρήση σύνδεσης στο Internet.



Εικόνα 74: Στιγμιότυπο από την εγκατάσταση του BlueStacks (1)



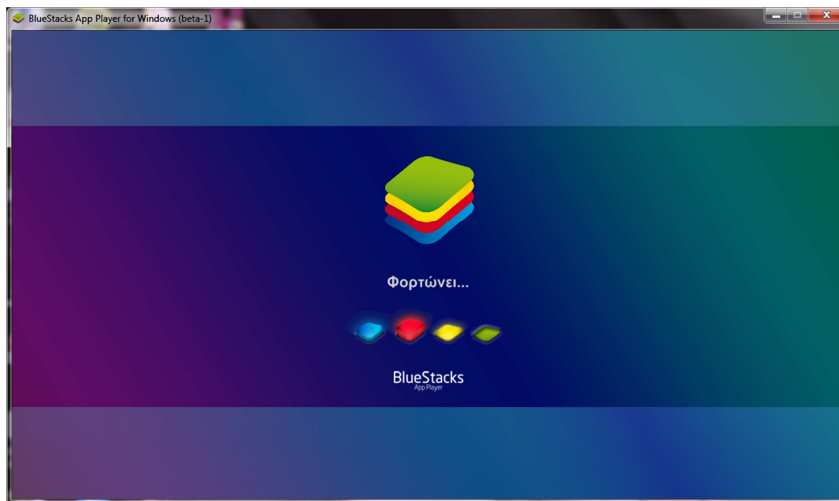
Εικόνα 75 : Στιγμιότυπο από την εγκατάσταση του BlueStacks (2)



Εικόνα 76: Στιγμιότυπο από την εγκατάσταση του BlueStacks (3)



Εικόνα 77: Στιγμιότυπο από την εγκατάσταση του BlueStacks (4)

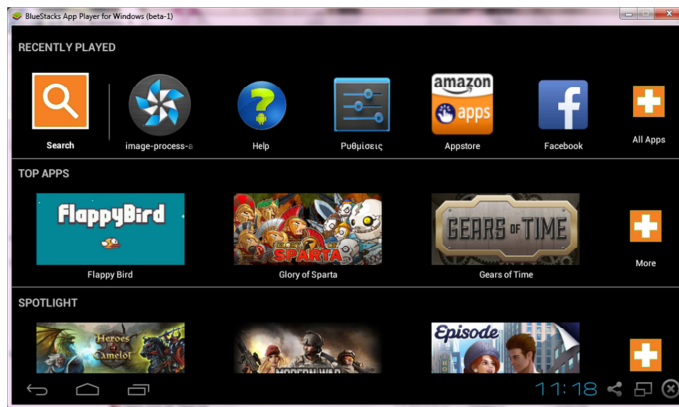


Εικόνα 78:Το φόρτωμα του Bluestacks μετά την εγκατάσταση.

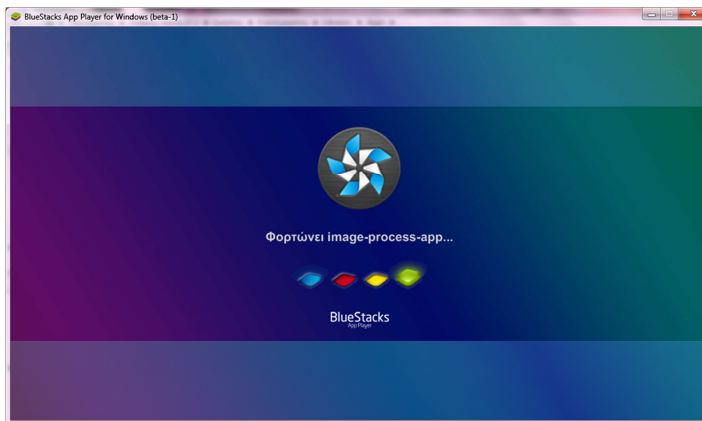
Αφού έχουμε κάνει μετατροπή της εφαρμογής μας, και έχουμε το apk αρχείο(μέσω Phoneyar Build), και αφού έχουμε εγκαταστήσει το Bluestacks, τότε ανοίγοντάς το με διπλό κλικ, αυτό (το apk) εγκαθίσταται αυτόματα! Χωρίς να χρειάζεται να κάνουμε κάποια επιλογή μέσα από το ίδιο το Bluestacks. Με διπλό κλικ πάνω στο apk, ανοίγει το παράθυρο που φαίνεται στην εικόνα. Και έτσι, αυτόματα έχουμε εγκατεστημένη την εφαρμογή μας στο Bluestacks.



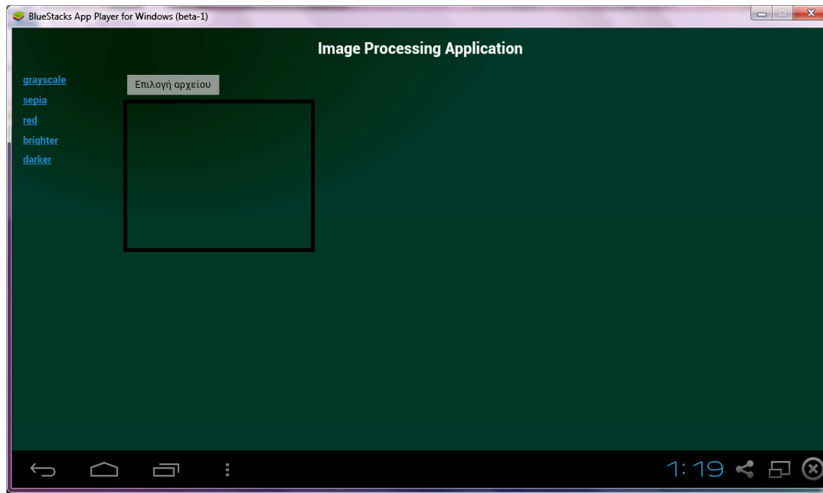
Εικόνα 79: Αυτόματη εγκατάσταση του apk αρχείου της εφαρμογής, μόνο με διπλό κλικ πάνω στο αρχείο



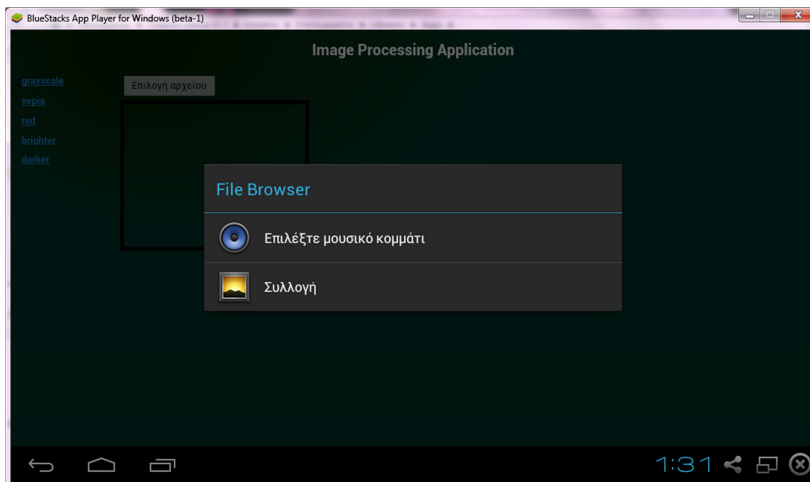
Εικόνα 80: Η εφαρμογή μας εγκατεστημένη έτοιμη στο Bluestacks.



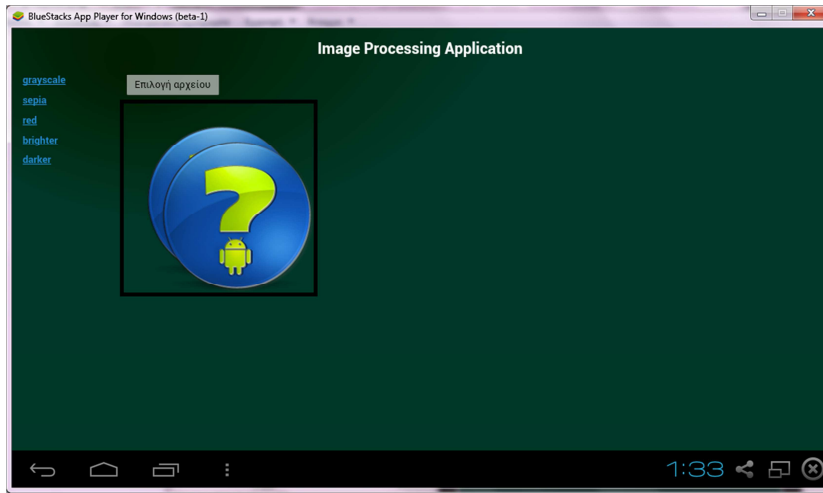
Εικόνα 81 : Φόρτωση της εφαρμογής μας σε περιβάλλον Android με προσομοίωση Bluestacks



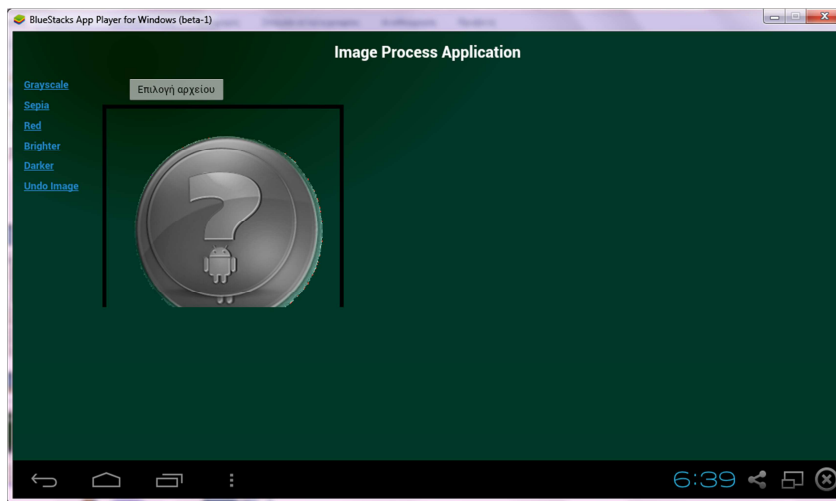
Εικόνα 82: Η εφαρμογή μας σε περιβάλλον Android



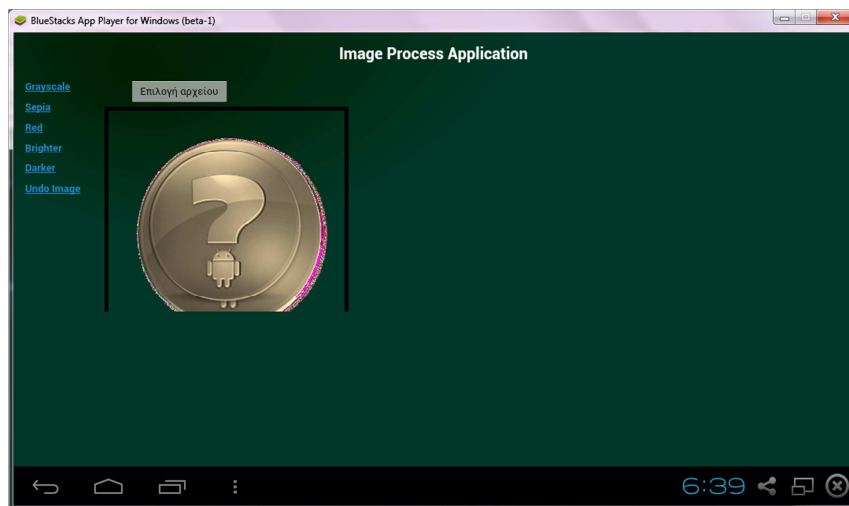
Εικόνα 83: Επιλογή αρχείου μέσω file selector του Android



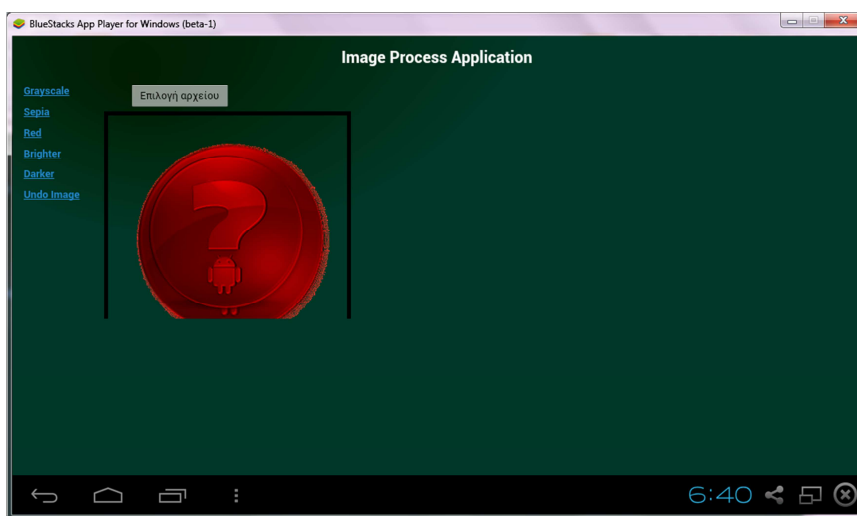
Εικόνα 84: Η εφαρμογή επεξεργασίας εικόνας με φορτωμένη την εικόνα χωρίς τα φίλτρα



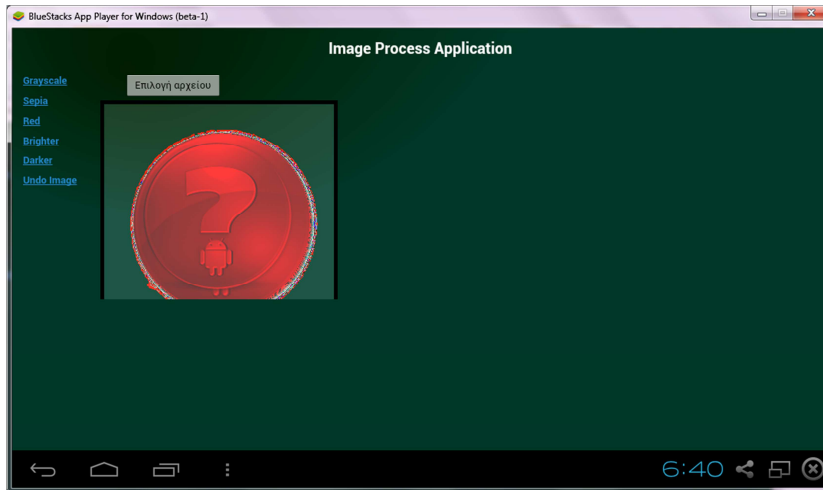
Εικόνα 85: Η εικόνα με εφαρμοσμένο το φίλτρο για ασπρόμαυρη μετατροπή



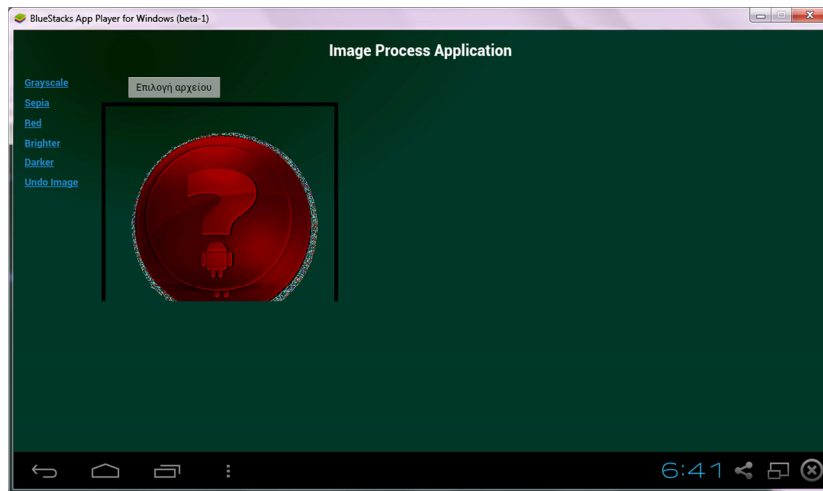
Εικόνα 86:Η εικόνα με εφαρμοσμένο το φίλτρο sepia



Εικόνα 87:Η εικόνα με εφαρμοσμένο το φίλτρο red



Εικόνα 88 :Η εικόνα με εφαρμοσμένο το φίλτρο φωτεινότητας 2 φορές



Εικόνα 89:η εφαρμογή με το φίλτρο ελάττωσης φωτεινότητας εφαρμοσμένο 2 φορές.

Κεφάλαιο 9ο: Σύγκριση υλοποιήσεων

9.1 Εισαγωγή

Τώρα που έχουμε δει την υλοποίηση των δυνατοτήτων της HTML5 σε κινητές συσκευές και σε τρία διαφορετικά περιβάλλοντα, χρήσιμο θα ήταν να γίνει μια σύγκριση ποιοτική, αλλά και να παρθούν μετρήσεις ποσοτικές προκειμένου να εξαχθούν συμπεράσματα για τις δυνατότητες καθενός από τα περιβάλλοντα που συμμετέχουν στη σύγκριση.

9.2 Ποιοτική σύγκριση

Τα τρία περιβάλλοντα που συγκρίνονται, (Tizen, Firefox, Android) υποβάλλονται σε σύγκριση για τη φιλικότητά τους προς τον developer. Θα μπορούσε κανείς να πει ότι η HTML5 κάνει την δουλειά του developer αρκετά ευκολότερη, καθώς και το περιβάλλον του Tizen IDE με το να προσθέτει αυτόματα κώδικα για τη διευκόλυνση του χρήστη, αλλά το γεγονός είναι ότι πρόκειται για μια τεχνολογία ανώριμη ακόμα, και έχει μέλλον ακόμη να διανύσει όσον αφορά την ευκολία ενσωμάτωσης δυνατοτήτων προγραμματιστικά. Δηλαδή, ενώ η JavaScript δίνει δυνατότητες πολλές μέσω βιβλιοθηκών της, προγραμματιστικά, το Tizen IDE δεν τα υποστηρίζει πλήρως.

Γενικά, κατά τη μετατροπή από HTML εφαρμογή σε Tizen app χρειάζεται να δομήσουμε τον υπάρχοντα κώδικα της HTML διαφορετικά (παρ'όλο που θα εξακολουθεί να είναι HTML), δεδομένου ότι οι σελίδες σε μια Tizen app δομούνται μέσω divs, πράγμα που σημαίνει ότι μια σελίδα με πολλούς υπερσυνδέσμους απαιτεί αναπροσαρμογή κατά τη μετατροπή, ώστε οι σελίδες στις οποίες οδηγούν οι υπερσυνδέσμοι να μπουν σε divs και να μικρύνει αισθητά το μέγεθος της κάθε σελίδας ώστε να χωρά σε οθόνη κινητού/tablet πλέον. Όσο και αν στην περιγραφή ακούγεται ως οξύμωρο σχήμα η χρήση των divs να οδηγεί σε μικρότερο μέγεθος σελίδας, στην πράξη κατά την προσομοίωση, το Tizen IDE με τη χρήση των divs και συνδέσμων (δηλαδή ετικέτας <a>) κάνει αυτά να λειτουργήσουν ως υποδοχείς περιεχομένου για την εφαρμογή, απομονώνοντας το περιεχόμενο κατά περίπτωση.

Τώρα για το Firefox OS, το οποίο είναι κι αυτό ακόμα σε πολύ πρώιμο στάδιο (alpha έκδοση) είναι γεγονός ότι πρόκειται για ένα πολύ φιλικό προς το χρήστη περιβάλλον, με πολύ απλή λειτουργία προγραμματιστικά και ωραίο περιβάλλον εφαρμογών. Προκειμένου να επιτευχθεί η μετατροπή από HTML εφαρμογή σε Firefox app, το μόνο που απαιτείται προγραμματιστικά είναι η σύνταξη του ξεχωριστού αρχείου manifest.webapp το οποίο είναι απολύτως απαραίτητο και θα συμπειστεί σε ένα αρχείο zip μαζί με τα αρχεία της εφαρμογής (.html, .js, .css) και θα «ανέβει» στον dashboard ο οποίος λειτουργεί ως simulator. Τίποτε άλλο, προγραμματιστικά το Firefox OS είναι ένα πλήρες σύστημα που είναι έτοιμο να «τρέξει» HTML εφαρμογές με χρήση **μόνο** σύνδεσης Internet και Firefox browser ώστε να γίνει verification της ορθότητας του manifest αρχείου, χωρίς δηλαδή να μεσολαβεί κάποιο IDE ή κάποιο ιδιαίτερο συντακτικό που να πρέπει να συμπεριληφθεί. Αν εξετάσουμε την πιθανότητα μετατροπής εφαρμογής σε Firefox app μιας Tizen app, αν χρησιμοποιήσουμε δηλαδή

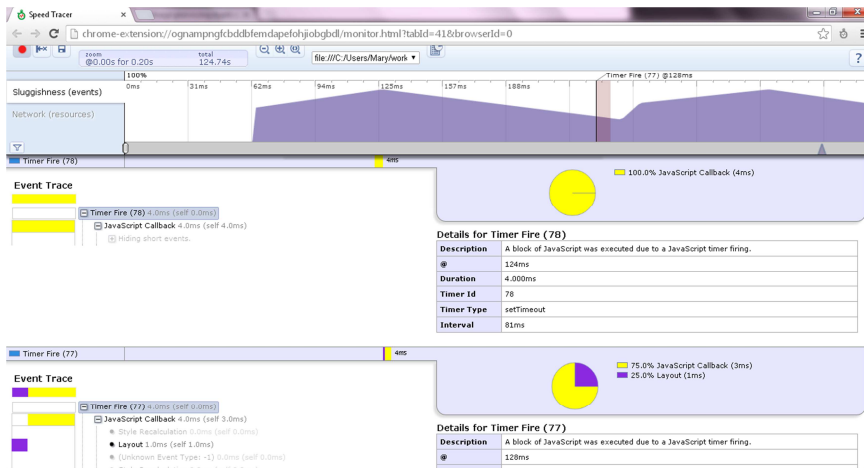
ένα απόφιο index.html αρχείο από Tizen app, τότε βλέπουμε ότι «τρέχει» κανονικά, παρά τις ιδιαίτερες μορφοποιήσεις σελίδων με divs (που σε τελική ανάλυση και αυτά HTML είναι), επίσης παρά και την προσθήκη έτοιμου κώδικα από το IDE (που αφορά κυρίως την χρήση βιβλιοθηκών jquery).

Όσον αφορά την περίπτωση του Android, είναι σαφής η ευκολία ενσωμάτωσης δυνατοτήτων του, για το πόσο εφαρμογές από άλλα περιβάλλοντα «τρέχουν» ικανοποιητικά χωρίς προβλήματα, καθώς και για το πόσο πολλές εφαρμογές έχουν αναπτυχθεί και κυκλοφορούν στο web για να διευκολύνουν τη χρήση και τις μετατροπές από άλλα περιβάλλοντα. Κατά τη μετατροπή της υπάρχουσας εφαρμογής με καθαρό HTML5 κώδικα σε PhoneGap/Android εφαρμογή, και παρά το γεγονός ότι δε χρησιμοποιήθηκε συντακτικό PhoneGap, μιας και η μετατροπή έγινε μέσω PhoneGap Build, δεν παρουσιάστηκε κανένα πρόβλημα στη διαδικασία. Ακόμα όμως και στην περίπτωση που χρησιμοποιούσαμε συντακτικό PhoneGap, δεδομένου του γεγονότος ότι πρόκειται για κώδικα HTML προσαρμοσμένο, και επίσης του ότι χρησιμοποιεί βιβλιοθήκες Javascript, η μετατροπή και πάλι θα ήταν εύκολη σε Android. Δεδομένου ότι εφαρμογές για Android μπορούν να αναπτύσσονται επίσης εκ του μηδενός και με χρήση γλώσσας Java (μέσω του SDK που έχει κυκλοφορήσει από το Eclipse) η οποία είναι μια πιο απαιτητική γλώσσα από την HTML, θα λέγαμε ότι σε εκείνη την περίπτωση ανεβαίνει ο βαθμός δυσκολίας, και το PhoneGap διευκολύνει αρκετά τον εν δυνάμει developer.

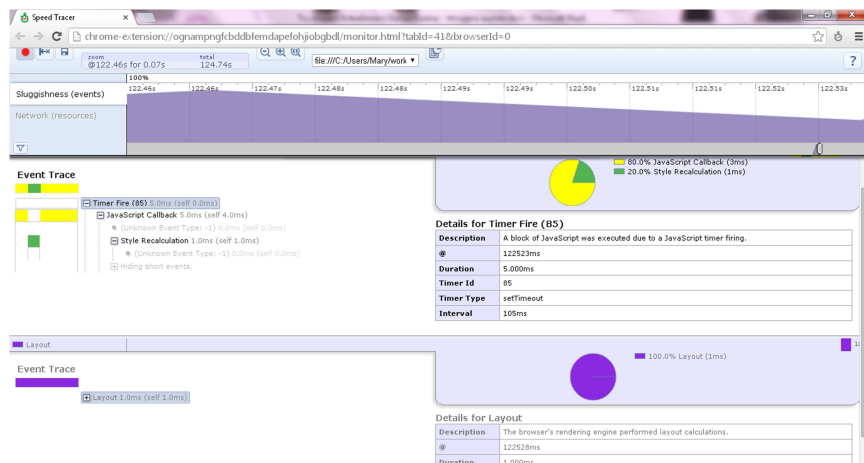
Αν επιχειρηθεί τώρα και μια πιο γενική σύγκριση των τριών αυτών διαδικασιών μετατροπής, σε ευκολία και απλότητα ξεχωρίζει η διαδικασία μετατροπής από HTML σε Firefox OS app. Χωρίς να μεσολαβεί επεξεργασία στον κώδικα της εφαρμογής ούτε καν για να διαμορφωθεί το Layout, η διαδικασία έτσι όπως έχει δομηθεί από τη Mozilla, καθίσταται απλή ακόμα και για τον νέο χρήστη- εν δυνάμει developer.

9.3 Ποσοτική εκτίμηση

Προκειμένου να υπάρχει ολοκληρωμένη εικόνα της υλοποίησης της εφαρμογής, καλό είναι να έχουμε και ποσοτικά στοιχεία της εφαρμογής. Γι' αυτό το σκοπό, θα μας φανεί χρήσιμο το Speed Tracer. Είναι ένα εργαλείο – επέκταση του Google Chrome, το οποίο εγκαθίσταται και «περιμένει» στον browser, να το ανοίξει ο χρήστης στην εκάστοτε καρτέλα (σελίδα ή εφαρμογή) που αποτελεί το αντικείμενο διερεύνησης ως προς τις απαιτήσεις του. Με άλλα λόγια, το Speed Tracer κάνει ανάλυση των απαιτήσεων μιας εφαρμογής σε σχέση με τις μεθόδους που αυτή περιλαμβάνει και πόσο ποσοστό της φόρτωσης καταλαμβάνουν αυτές στον Chrome. Τώρα, για την εφαρμογή μας της επεξεργασίας εικόνας, στις παρακάτω εικόνες φαίνονται τα αποτελέσματα του Speed Tracer μέσω Chrome.



Εικόνα 90 :Μέτρηση με την αρχική φόρτωση της εφαρμογής στον Chrome



Εικόνα 91: Μέτρηση με φόρτωση εικόνας και φίλτρου στην εφαρμογή

Στις μετρήσεις βλέπουμε ότι μεγάλο ποσοστό της φόρτωσης καταλαμβάνουν οι συναρτήσεις της Javascript. Στην πρώτη εικόνα που έχει παρθεί από την φόρτωση της εφαρμογής, βλέπουμε ότι το μεγαλύτερο μέρος της φόρτωσης το καταλαμβάνει η φόρτωση της Javascript, δηλαδή η φόρτωση των συναρτήσεων και των δυνατοτήτων της στον browser. Η πρώτη μέτρηση που φαίνεται στην εικόνα 89, διαδραματίζεται κατά τη φόρτωση της εφαρμογής στον Chrome, όπου και η Javascript καταλαμβάνει το 100% της επεξεργαστικής δυνατότητας του browser, και μεταφράζεται σε χρόνο 4 ms. Η δεύτερη μέτρηση τώρα στην ίδια εικόνα είναι κατά την επιλογή και φόρτωση εικόνας στην εφαρμογή-εξ'ού και η παρουσία του layout στο γράφημα – όπου και βλέπουμε να

μοιράζεται ο προηγούμενος χρόνος φόρτωσης των 4ms σε 75-25 με το layout να καταλαμβάνει το 25% που μεταφράζεται σε 1 ms.

Στη δεύτερη εικόνα βλέπουμε μέτρηση που έχει παρθεί κατά την εφαρμογή φίλτρων σε εικόνα που ήδη έχει φορτωθεί. Σ' αυτό το σημείο πάρθηκε η πρώτη μέτρηση, όπου η JavaScript (δηλαδή η εφαρμογή φίλτρου) καταλαμβάνει και πάλι το μεγαλύτερο κομμάτι της φόρτωσης, η οποία μεταφράζεται σε αριθμούς με μοίρασμα 80-20 με το Style Recalculation του γραφήματος να αντιπροσωπεύει προφανώς την απεικόνιση της εικόνας με το φίλτρο εφαρμοσμένο και το οποίο καταλαμβάνει μόνο ένα μικρό μέρος της φόρτωσης. Η δεύτερη μέτρηση αντιπροσωπεύει την απεικόνιση της εικόνας με τα φίλτρα ήδη εφαρμοσμένα.

Κεφάλαιο 10ο :Συμπεράσματα

Το κύριο συμπέρασμα που προκύπτει από την μελέτη των νέων λειτουργικών για κινητά περιβάλλοντα Tizen / Firefox Os είναι ότι πρόκειται για ιδέες εύκολα προσαρμόσιμες από νέους χρήστες και πρωτοποριακές –η εφαρμογή δηλαδή μιας γλώσσας υπερκειμένου σε άλλου είδους περιβάλλοντα, κινητά δηλαδή , εκτός από αυτό του web, παρόλο που ακόμα δεν έχει εξελιχθεί πλήρως ώστε να φανούν όλες της οι δυνατότητες.

Προγραμματιστικά μιλώντας, από τα τρία περιβάλλοντα που παρουσιάστηκαν σ' αυτή την πτυχιακή ξεχωρίζει για την ευκολία προσαρμογής της HTML το Firefox Os , πράγμα που δεν αποτελεί έκπληξη, αν σκεφτεί κανείς ότι αναπτύχθηκε από τη Mozilla, δηλαδή από μια εταιρία που γνωρίζει το web πολύ καλά. Γενικά μιλώντας, αν εξετάσουμε τα περιβάλλοντα αυτά από άλλη σκοπιά, δηλαδή αυτή του χρήστη, το Android ξεχωρίζει για την αποτελεσματικότητα που τρέχουν οι εφαρμογές και για το πόσο πολλά εργαλεία έχουν ήδη αναπτυχθεί ώστε να ικανοποιήσουν και τον πιο απαιτητικό χρήστη αλλά και τον κάθε εν δυνάμει developer. Τώρα όσον αφορά το Tizen, ως νέα και ανώριμη τεχνολογία, έχει να διανύσει δρόμο και εκδόσεις μέχρι να μπορέσει προγραμματιστικά να συναγωνιστεί από τη μία το μεγαθήριο Mozilla, και από την άλλη το φαινόμενο Android.

Ένα άλλο συμπέρασμα που προκύπτει από αυτή την μελέτη, είναι ότι η τεχνολογία του web χρειάζεται πολλές ή λίγες, πάντως σίγουρα **κάποιες** προσαρμογές προκειμένου να λειτουργήσει σε άλλα, κινητά περιβάλλοντα. Για παράδειγμα στην περίπτωση του Tizen όπου σε αυτή την εφαρμογή δεν υπάρχουν πολλοί υπερσύνδεσμοι που να οδηγούν σε άλλες σελίδες δεν χρειάστηκε αναπροσαρμογή τέτοιου είδους, αλλά το μέγεθος της οθόνης είναι ένας παράγοντας που θα πρέπει να τον λαμβάνει κανείς υπ' όψιν όταν φτιάχνει εφαρμογές για κινητά περιβάλλοντα ώστε να μη κουράζεται το μάτι του χρήστη, και να μη χρειάζονται συνεχή scroll-ups, scroll-downs, και zoom που εκτός από άβολα για το χρήστη είναι και ενεργοβόρα για μια φορητή συσκευή. Καλό είναι να διατηρηθεί το κυρίως περιεχόμενο μιας εφαρμογής στο κέντρο της οθόνης.

Βιβλιογραφία

[1] <http://www.5-one.gr/index.php/arthra/165-mobile-statistika> :

Όπως την επισκέφθηκα 1/2014

[2] <http://www.nngroup.com/articles/mobile-native-apps/> :

Όπως την επισκέφθηκα 2/2014

[3] http://el.wikipedia.org/wiki/Windows_Phone :

Όπως την επισκέφθηκα 1/2014

[4] http://en.wikipedia.org/wiki/Windows_Phone :

Όπως την επισκέφθηκα 1/2014

[5] <http://el.wikipedia.org/wiki/IOS> :

Όπως την επισκέφθηκα 1/2014

[6] <http://el.wikipedia.org/wiki/Android> :

Όπως την επισκέφθηκα 2/2013

[7] <http://dspace.cst.uop.gr:8080/bitstream/handle/123456789/27/glikiotis-thesis.pdf?sequence=1> :

Όπως την επισκέφθηκα 2/2013

[8] http://en.wikipedia.org/wiki/Ubuntu_Phone :

Όπως την επισκέφθηκα 1/2014

[9] http://en.wikipedia.org/wiki/Chrome_OS :

Όπως την επισκέφθηκα 1/2014

[10] http://en.wikipedia.org/wiki/Sailfish_OS :

Όπως την επισκέφθηκα 2/2014

[11] <http://www.linuxinside.gr/news/8978/kykloforise-proto-jolla-smartphone-me-sailfish-os>: Όπως την επισκέφθηκα 2/2014

[12] http://en.wikipedia.org/wiki/HTML5_in_mobile_devices :

Όπως την επισκέφθηκα 3/2014

[13]<http://www.geek.com/mobile/a-one-page-guide-to-html5-support-in-mobile-browsers-1420697/> :

Όπως την επισκέφθηκα 3/2014

[14] http://www.w3schools.com/html/html5_webstorage.asp :

Όπως την επισκέφθηκα 3/2014

[15] http://www.w3schools.com/html/html5_video.asp :

Όπως την επισκέφθηκα 3/2014

[16]<http://imalhasaranga.com/wp-content/uploads/2013/12/browsersupport1.png> : Όπως την

επισκέφθηκα 3/2014

[17] http://www.w3schools.com/html/html5_geolocation.asp :

Όπως την επισκέφθηκα 3/2014

[18] <http://diveintohtml5.info/geolocation.html>:

Όπως την επισκέφθηκα 3/2014

[19]<http://www.c-sharpcorner.com/UploadFile/1e050f/geolocation-api/Images/image3.jpg> :

Όπως την επισκέφθηκα 3/2014

[20] <http://diveintohtml5.info/forms.html> :

Όπως την επισκέφθηκα 3/2014

[21] <http://davidwalsh.name/autofocus> :

Όπως την επισκέφθηκα 3/2014

[22] http://www.w3schools.com/html/html5_form_input_types.asp:

Όπως την επισκέφθηκα 3/2014

[23] http://www.w3schools.com/html/html5_form_attributes.asp:

Όπως την επισκέφθηκα 3/2014

[24] <http://diveintohtml5.info/forms.html#validation> :

Όπως την επισκέφθηκα 3/2014

[25]<http://usability.com.au/wp-content/uploads/2013/05/html5-required-indication.jpg> : Όπως την

επισκέφθηκα 3/2014

[26] <http://en.wikipedia.org/wiki/Tizen> :

Όπως την επισκέφθηκα 2/2013

[27] <https://developer.mozilla.org/en-US/Apps/Developing/Manifest?redirectlocale=en-US&redirectslug=Web%2FApps%2FManifest> : Όπως την επισκέφθηκα 3/2014

[28] https://developer.mozilla.org/en-US/docs/Tools/Firefox_OS_Simulator:

Όπως την επισκέφθηκα 12/2013

[29] <http://www.mozilla.org/el/firefox/partners/> :

Όπως την επισκέφθηκα 4/2014

[30]http://en.wikipedia.org/wiki/Firefox_OS:

Όπως την επισκέφθηκα 12/2013

[31] http://el.wikipedia.org/wiki/Firefox_OS :

Όπως την επισκέφθηκα 12/2013

[32] <https://www.tizen.org/sites/default/files/tizen-architecture-linuxcollab.pdf> :

Όπως την επισκέφθηκα 2/2013

[33]https://developer.mozilla.org/en-US/Marketplace/Publishing/Publish_options :

Όπως την επισκέφθηκα 4/2014

[34]<http://stackoverflow.com/questions/12368910/html-display-image-after-selecting-filename> :

Όπως την επισκέφθηκα 10/2013

[35] http://www.w3schools.com/html/html5_intro.asp :

Όπως την επισκέφθηκα 9/2013

[36] <http://en.wikipedia.org/wiki/HTML5> :

Όπως την επισκέφθηκα 9/2013

[37] <http://diveintohtml5.info/canvas.html> :

Όπως την επισκέφθηκα 10/2013

[38] http://en.wikipedia.org/wiki/Canvas_element:

Όπως την επισκέφθηκα 10/2013

[39]<http://www.mozilla-hispano.org/archivos/docs/foxosappdaysvall/arquitectura.pdf>:

Όπως την επισκέφθηκα 1/2014

[40] <http://hvankov.wordpress.com/2010/12/26/how-to-implement-html5-canvas-undo-function/>:

Όπως την επισκέφθηκα 11/2013

[41] https://developer.mozilla.org/en-US/Apps/Quickstart/Build/Your_first_app :

Όπως την επισκέφθηκα 12/2013