



ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ
ΣΧΟΛΗ ΔΙΟΙΚΗΣΗΣ ΚΑΙ ΟΙΚΟΝΟΜΙΑΣ
ΤΜΗΜΑ ΔΙΟΙΚΗΣΗ ΕΠΙΧΕΙΡΗΣΕΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ
ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΥΛΟΠΟΙΗΣΗ
ΕΚΠΑΙΔΕΥΤΙΚΗΣ ΕΦΑΡΜΟΓΗΣ ΓΙΑ ΚΙΝΗΤΕΣ
ΣΥΣΚΕΥΕΣ ΠΛΑΤΦΟΡΜΑΣ IOS

ΣΚΟΥΠΑΣ ΣΠΥΡΙΔΩΝ
ΜΠΑΣΙΝΑΣ ΑΝΔΡΕΑΣ
ΜΟΥΣΤΑΚΗ ΜΑΡΙΑ
ΕΠΟΠΤΕΥΩΝ ΚΑΘΗΓΗΤΗΣ:ΧΑΛΚΙΟΠΟΥΛΟΣ ΚΩΝΣΤΑΝΤΙΝΟΣ

ΠΑΤΡΑ,2017

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΕΡΙΕΧΟΜΕΝΑ	1
ΠΡΟΛΟΓΟΣ	2
ΠΕΡΙΛΗΨΗ	3
ABSTRACT	4
ΕΙΣΑΓΩΓΗ	5
ΚΕΦΑΛΑΙΟ 1 – ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ	6
ii) Λογισμικό ανοιχτού κώδικα	8
iv) Φυσική διεπαφή χρήστη (NUI)	14
v) Visual Studio	16
vi) App Inventor	17
vii) Λειτουργικό σύστημα iOS	21
viii) Xcode	21
x) Τι είναι το GNU	24
ΚΕΦΑΛΑΙΟ 2 - ΣΥΣΤΗΜΑΤΑ ΠΛΗΡΟΦΟΡΗΣΗΣ ΣΠΟΥΔΑΣΤΩΝ	25
ΚΕΦΑΛΑΙΟ 3-ΒΗΜΑΤΙΚΗ ΕΠΕΞΗΓΗΣΗ ΕΥΡΕΣΗ ΙΔΕΑΣ ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΥΛΟΠΟΙΗΣΗΣ ΜΕ PROTOTYPE	29
i) ΑΝΑΚΑΛΥΨΗ ΤΗΣ ΙΔΕΑΣ.....	30
ii) ΠΡΟΣΔΙΟΡΙΣΜΟΣ ΤΗΣ ΙΔΕΑΣ	31
iii) ΣΧΕΔΙΑΣΜΟΣ.....	33
iv) Human Interface Guidelines Ios	34
v) 10 κανόνες ευχρηστίας (10 usability Heuristics for user interface design by Nilsen Norman)	35
vi) ΜΕΛΕΤΗ ΣΧΕΔΙΑΣΤΩΝ ΚΑΙ ΕΥΡΕΣΗ ΕΙΚΟΝΩΝ.....	37
vii) ΔΟΜΗ ΚΑΙ USER FLOW ΕΦΑΡΜΟΓΗΣ	38
viii) ΥΛΟΠΟΙΗΣΗ ΣΤΟ SKETCH	40
ix) ΤΙ ΕΙΝΑΙ ΤΟ PROTOTYPE	50
x) ΤΙ ΕΙΝΑΙ ΤΟ InVision.....	50
xi) ΜΕΛΛΟΝΤΙΚΕΣ ΚΙΝΗΣΕΙΣ	56
ΚΕΦΑΛΑΙΟ 4 – ΕΦΑΡΜΟΓΗ	57
ΣΥΜΠΕΡΑΣΜΑΤΑ	79
ΒΙΒΛΙΟΓΡΑΦΙΑ	80

ΠΡΟΛΟΓΟΣ

Η Παρούσα πτυχιακή εργασία με τίτλο <<ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΥΛΟΠΟΙΗΣΗ ΕΚΠΑΙΔΕΥΤΙΚΗΣ ΕΦΑΡΜΟΓΗΣ ΓΙΑ ΚΙΝΗΤΕΣ ΣΥΣΚΕΥΕΣ iOS >> έχει σκοπό να πληροφορήσει και να εξηγήσει στον αναγνώστη της, το τρόπο που σχεδιάζονται και το μέσω που δημιουργούνται οι εφαρμογές που χρησιμοποιεί σήμερα στις κινητές του συσκευές. Είναι πλέον κοινά αποδεχτό ότι έχει φτάσει η εποχή που όλοι μας χρησιμοποιούμε την τεχνολογία είτε για γνώση, είτε για δουλειά είτε για διασκέδαση, είτε για εκπαίδευση. Η χρήση της τεχνολογίας καλύπτει μεγάλο μέρος του χρόνου από την καθημερινότητάς μας και γι' αυτό τον λόγο η παρούσα πτυχιακή εργασία ευελπιστούμε, να ενημερώσει και να εκπαιδεύσει σε γενικές γραμμές τον αναγνώστη σε αυτό το πολύ σημαντικό κομμάτι που ονομάζεται τεχνολογία και να του παρουσιάσει την εφαρμογή που σχεδιάσαμε με τέτοιο τρόπο, ώστε να την κατανοήσει και να την χρησιμοποιήσει.

ΠΕΡΙΛΗΨΗ

Η παρούσα πτυχιακή εργασία αποτέλεσε το αντικείμενο μελέτης μας εδώ και αρκετά μεγάλο διάστημα και θεωρούμε ότι λόγω της σπουδαιότητας του θέματος θα αποτελέσει ένα πολύ σημαντικό εφαλτήριο για την μετέπειτα πορεία μας στον κλάδο. Θέμα της συγκεκριμένης πτυχιακής εργασίας είναι ο σχεδιασμός και η παρουσίαση μιας εφαρμογής με την οποία ένας φοιτητής ή ένας οποιοσδήποτε χρήστης θα μπορεί να εισάγει τα υλικά που διαθέτει, το χρόνο που έχει στην διάθεσή του και διάφορες άλλες παραμέτρους με σκοπό να πραγματοποιήσει την επιθυμία του να μαγειρέψει μονός του το φαγητό του και να μάθει από αυτό. Πριν φτάσουμε όμως στο τμήμα της εργασίας που αφορά στην πρακτική μελέτη, προηγήθηκε η βιβλιογραφική ανάλυση και η συλλογή υλικού για την παρουσίαση της βιβλιογραφικής επισκόπησης που συνολικά δομείται σε τέσσερα κεφάλαια, το πρώτο εκ των οποίων αναφέρεται στο θεωρητικό υπόβαθρο, που περιλαμβάνει το λειτουργικό σύστημα, το λογισμικό ανοιχτού κώδικα, τη φυσική διεπαφή χρήστη (Natural User Interface), τη γενική δημόσια άδεια (GNU), και το λειτουργικό σύστημα Ios και τα περιβάλλοντα εφαρμογών Visual Studio, App Inventor και Xcode καθώς και πληροφορίες για την γλώσσα προγραμματισμού Swift.

Στο δεύτερο κεφάλαιο πραγματοποιείται ανάλυση των συστημάτων πληροφόρησης σπουδαστών, που χρησιμοποιούν αρκετά εκπαιδευτικά ιδρύματα. Θεωρήθηκε σκόπιμο αν και δεν σχετίζεται άμεσα με το θέμα, να πραγματοποιηθεί αναφορά στα συγκεκριμένα συστήματα, καθώς η εφαρμογή που παρουσιάζεται στην παρούσα εργασία, έχει σκοπό την βοήθεια και την ενημέρωση των φοιτητών, αλλά όχι από εκπαιδευτικής άποψης.

Κλείνοντας, στο τρίτο κεφάλαιο πραγματοποιείται βηματική επεξήγηση για την εύρεση της ιδέας τον σχεδιασμό και την υλοποίηση με prototype. Τα εργαλεία που χρησιμοποιήθηκαν ώστε να σχεδιαστεί η εφαρμογή, ο τρόπος που φτάσαμε στο τελικό αποτέλεσμα, η παρουσίαση της διαδικασίας σχεδιασμού εφαρμογής, η έννοια του prototyping καθώς και μελλοντικές πιθανές κινήσεις.

Στο τέταρτο κεφάλαιο της παρούσας μελέτης πραγματοποιείται λεπτομερώς παρουσίαση της εφαρμογής με εικόνες και επεξήγηση σε βήματα για τις δυνατότητες που προσφέρει στον χρήστη και τον τρόπο λειτουργίας της. Ευελπιστούμε η παρούσα πτυχιακή εργασία να αποτελέσει χρήσιμο εγχειρίδιο για μελλοντικούς προγραμματιστές και επενδυτές για την ολοκληρωτική υλοποίηση της ιδέας μας με σκοπό την χρήση της εφαρμογής.

ABSTRACT

This project was our study object for too long and we feel that due to the importance of the issue will be a major springboard for our future course of the industry. The subject of this thesis is the design and presentation of an application by which a student or any other user will be able to import the products that he has, the time and other filters to achieve his desire to cook his own food and learn from that. Before we get to the part of the work on the practical study preceded the bibliographic analysis and collection of material for the presentation of literature review that whole structured into three chapters, the first of which refers to the theoretical background, which includes the operating system, open source software, natural user interface (NUI), the general public license (GNU), operating system iOS and application environments Visual Studio, app Inventor and Xcode and some informations about programming code Swift. In the second chapter an analysis of student information systems, using several educational institutions. It considered appropriate, although not directly related to the subject, to be held reference to specific systems, as the embodiment shown in this paper, aims to help and information for students, but not from the educational point of view.

Closing the third chapter a step a by step explanation is being used about the adoption of the idea, planning and implementation using Prototype. Furthermore, there is an extend report of the tools that have been used to design the application, a presentation of the designing procedure, and possible future extensions and additional uses.

In the fourth chapter of the present study, a detailed explanation is made, using pictures and step by step analysis of the opportunities that are offered to the possible user and the function mode as well. we hope this study will be a useful handbook to the future programmers and investors to fulfill our idea for the use of this application.

ΕΙΣΑΓΩΓΗ

Κατά την ανάπτυξη του .NET Framework, οι βιβλιοθήκες τάξεων έγιναν αρχικά με τη χρήση ενός συστήματος μεταγλωττιστή διαχειριζόμενου κώδικα που ονομάζεται Simple Managed C (SMC). Τον Ιανουάριο του 1999, ο Anders Hejlsberg σχημάτισε μια ομάδα για να δημιουργήσει μια νέα γλώσσα την εποχή εκείνη που ονομάζεται Cool, η οποία έμοιαζε με την "C-like Object Oriented Language". Η Microsoft είχε θεωρήσει το όνομα "Cool" ως το τελικό όνομα της γλώσσας, αλλά επέλεξε να μην το κάνει για λόγους εμπορικού σήματος. Μέχρι τη στιγμή που το έργο .NET ανακοινώθηκε δημοσίως στο συνέδριο επαγγελματιών προγραμματιστών του Ιουλίου 2000, η γλώσσα μετονομάστηκε σε C #, ενώ οι βιβλιοθήκες κλάσης και το ASP.NET runtime είχαν μεταφερθεί στο C #.

Ο κύριος σχεδιαστής και επικεφαλής αρχιτέκτονας της C # στη Microsoft είναι ο Anders Hejlsberg, ο οποίος ασχολήθηκε προηγουμένως με το σχεδιασμό των Turbo Pascal, Embarcadero Delphi (πρώην CodeGear Delphi, Inprise Delphi και Borland Delphi) και Visual J ++. Σε συνεντεύξεις και τεχνικές εργασίες, έχει δηλώσει ότι οι ελλείψεις στις περισσότερες μεγάλες γλώσσες προγραμματισμού (π.χ. C ++, Java, Delphi και Smalltalk) οδήγησαν τα βασικά στοιχεία του Common Language Runtime (CLR), τα οποία με τη σειρά τους οδήγησαν στη σχεδίαση Της ίδιας της γλώσσας C #.

Ο James Gosling, ο οποίος δημιούργησε τη γλώσσα προγραμματισμού Java το 1994, και ο Bill Joy, συνιδρυτής της Sun Microsystems, ο δημιουργός της Java, ονομάζεται C # μια "απομίμηση" της Java. Ο Gosling είπε επίσης ότι «το C # είναι είδος Java με διαγραφή την αξιοπιστία, την παραγωγικότητα και την ασφάλεια». Ο Klaus Krefit και η Angelika Langer (συγγραφείς ενός βιβλίου ροών C ++) Είναι σχεδόν πανομοιότυπες γλώσσες προγραμματισμού. Μόνο επανάληψη που στερείται καινοτομίας » « Ο καθένας δεν θα ισχυριστεί ότι η Java ή η C # είναι επαναστατικές γλώσσες προγραμματισμού που άλλαξαν τον τρόπο που γράφουμε τα προγράμματα » και « ο C # δανείστηκε πολύ από την Java και αντίστροφα . Τώρα που η C # υποστηρίζει την πυγμαχία και την αποσυμφόρηση, θα έχουμε ένα πολύ παρόμοιο χαρακτηριστικό στην Java. Τον Ιούλιο του 2000, ο Anders Hejlsberg είπε ότι ο C # δεν είναι " κλώνος Java " και είναι " πολύ πιο κοντά στην C ++.

Μετά την κυκλοφορία του C # 2.0 το Νοέμβριο του 2005, οι γλώσσες C # και Java εξελίχθηκαν σε ολοένα και πιο αποκλίνουσες τροχιές, κάνοντας κάπως λιγότερο παρόμοιες. Μια από τις πρώτες μεγάλες αναχωρήσεις ήρθε με την προσθήκη γενικών φαρμάκων και στις δύο γλώσσες, με πολύ διαφορετικές εφαρμογές. Το C # χρησιμοποιεί την πληροφορία για την παροχή γενικών αντικειμένων "πρώτης τάξης" που μπορούν να χρησιμοποιηθούν όπως οποιαδήποτε άλλη κλάση, με την παραγωγή κώδικα να εκτελείται σε χρόνο φόρτωσης τάξης. Επιπλέον, η C # έχει προσθέσει αρκετά σημαντικά χαρακτηριστικά για να φιλοξενήσει προγραμματισμό λειτουργικού στυλ, με αποκορύφωμα τις επεκτάσεις LINQ που κυκλοφορούν με το C # 3.0 και το υποστηρικτικό του πλαίσιο εκφράσεων λάμδα, μεθόδων επέκτασης και ανώνυμων τύπων. Αυτά τα χαρακτηριστικά επιτρέπουν στους προγραμματιστές C # να χρησιμοποιούν λειτουργικές τεχνικές προγραμματισμού, όπως κλεισίματα, όταν είναι συμφέρουσα για την εφαρμογή τους. Οι επεκτάσεις LINQ και οι λειτουργικές εισαγωγές βοηθούν τους προγραμματιστές να μειώσουν το ποσό του κώδικα "boilerplate" που περιλαμβάνεται σε συνήθεις εργασίες όπως η αναζήτηση βάσης δεδομένων,

η ανάλυση ενός αρχείου xhtml ή η αναζήτηση σε μια δομή δεδομένων, μετατοπίζοντας την έμφαση στην πραγματική λογική του προγράμματος, Βελτιώστε την αναγνωσιμότητα και τη συντήρηση αυτής.

ΚΕΦΑΛΑΙΟ 1 – ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ

i) Τι είναι λειτουργικό σύστημα

Ένα λειτουργικό σύστημα (OS) είναι λογισμικό συστήματος που διαχειρίζεται πόρους υλικού και λογισμικού υπολογιστών και παρέχει κοινές υπηρεσίες για προγράμματα υπολογιστών. Όλα τα προγράμματα υπολογιστών, εκτός του υλικολογισμικού, απαιτούν λειτουργία ενός λειτουργικού συστήματος.

Τα λειτουργικά συστήματα χρονομεριστικής κατανομής απαιτούν αποτελεσματική χρήση του συστήματος και μπορεί επίσης να περιλαμβάνουν λογισμικό λογιστικής για την κατανομή του χρόνου επεξεργαστή, τη μαζική αποθήκευση, την εκτύπωση και άλλους πόρους.

Για λειτουργίες υλικού όπως είσοδο και έξοδο και κατανομή μνήμης, το λειτουργικό σύστημα λειτουργεί ως ενδιάμεσος μεταξύ των προγραμμάτων και του υλικού του υπολογιστή, αν και ο κώδικας εφαρμογής εκτελείται συνήθως απευθείας από το υλικό και συχνά κάνει κλήσεις συστήματος σε OS ή διακόπτεται από αυτήν. Τα λειτουργικά συστήματα βρίσκονται σε πολλές συσκευές που περιέχουν έναν υπολογιστή - από κινητά τηλέφωνα και κονσόλες βιντεοπαιχνιδιών έως διακομιστές web και υπερυπολογιστές.

Το κυρίαρχο λειτουργικό σύστημα επιτραπέζιων υπολογιστών είναι τα Microsoft Windows με μερίδιο αγοράς περίπου 83,3%. Το MacOS από την Apple Inc. βρίσκεται στη δεύτερη θέση (11,2%) και οι ποικιλίες του Linux βρίσκονται στην τρίτη θέση (1,55%). Στον τομέα της κινητής τηλεφωνίας (συνδυασμός smartphone και tablet), σύμφωνα με τα στοιχεία του τρίτου τριμήνου του 2016, η Android από την Google κυριαρχεί με 87,5% και ρυθμό ανάπτυξης 10,3% ετησίως, ακολουθούμενη από την iOS της Apple με 12,1% Μερίδιο 5,2%, ενώ άλλα λειτουργικά συστήματα ανέρχονται μόλις στο 0,3%. Οι διανομές του Linux κυριαρχούν στους τομείς διακομιστών και υπερυπολογιστών. Άλλες εξειδικευμένες κατηγορίες λειτουργικών συστημάτων, όπως ενσωματωμένα συστήματα και συστήματα σε πραγματικό χρόνο, υπάρχουν για πολλές εφαρμογές.

Το Unix ήταν αρχικά γραμμένο σε γλώσσα συναρμολόγησης. Ο Ken Thompson έγραψε τη Β, βασισμένη κυρίως στην BCPL, με βάση την εμπειρία του στο έργο MULTICS. Το Β αντικαταστάθηκε από το C και το Unix, που ξαναγράφηκε στο C, αναπτύχθηκε σε μια μεγάλη, περίπλοκη οικογένεια αλληλοσυνδεδεμένων λειτουργικών συστημάτων που έχουν επιρροή σε κάθε σύγχρονο λειτουργικό σύστημα.

Η οικογένεια που μοιάζει με Unix είναι μια διαφορετική ομάδα λειτουργικών συστημάτων, με αρκετές σημαντικές υποκατηγορίες όπως το σύστημα V, το BSD και το Linux. Το όνομα "UNIX" είναι εμπορικό σήμα του The Open Group, το οποίο το εκχωρεί για χρήση με οποιοδήποτε λειτουργικό σύστημα που έχει αποδειχθεί ότι συμμορφώνεται με τους ορισμούς του. Το "UNIX-like" χρησιμοποιείται συνήθως για να αναφερθεί στο μεγάλο σύνολο λειτουργικών συστημάτων που μοιάζουν με το αρχικό UNIX.

Τα συστήματα τύπου Unix τρέχουν σε μια ευρεία ποικιλία αρχιτεκτονικών υπολογιστών. Χρησιμοποιούνται σε μεγάλο βαθμό για εξυπηρετητές σε επιχειρήσεις, καθώς και σταθμούς εργασίας σε ακαδημαϊκά και μηχανολογικά περιβάλλοντα. Οι ελεύθερες παραλλαγές UNIX, όπως το Linux και το BSD, είναι δημοφιλείς σε αυτούς τους τομείς.

Τέσσερα λειτουργικά συστήματα είναι πιστοποιημένα από το The Open Group (κάτοχος του εμπορικού σήματος Unix) ως Unix. Το HP-UX της HP και το AIX της IBM είναι και οι δύο απόγονοι του αρχικού συστήματος V Unix και έχουν σχεδιαστεί για να λειτουργούν μόνο στο υλικό του πωλητή τους. Αντίθετα, το Solaris της Sun Microsystems μπορεί να λειτουργεί σε πολλαπλούς τύπους υλικού, συμπεριλαμβανομένων των διακομιστών x86 και Sparc, καθώς και των υπολογιστών. Το macOS της Apple, που αποτελεί αντικατάσταση του παλαιότερου (μη Unix) Mac OS της Apple, είναι μια υβριδική βασισμένη στον πυρήνα έκδοση BSD που προέρχεται από τους NeXTSTEP, Mach και FreeBSD.

Η διαλειτουργικότητα Unix επιδιώχθηκε με την καθιέρωση του προτύπου POSIX. Το πρότυπο POSIX μπορεί να εφαρμοστεί σε οποιοδήποτε λειτουργικό σύστημα, αν και δημιουργήθηκε αρχικά για διάφορες παραλλαγές Unix.

Το macOS είναι μια σειρά γραφικών λειτουργικών συστημάτων ανοιχτού πυρήνα που αναπτύσσονται, πωλούνται και πωλούνται από την Apple Inc., η τελευταία εκ των οποίων φορτώνεται σε όλους τους υπολογιστές Macintosh. Το macOS είναι ο διάδοχος του αρχικού κλασικού Mac OS, το οποίο ήταν το βασικό λειτουργικό σύστημα της Apple από το 1984. Σε αντίθεση με τον προκάτοχό του, το macOS είναι ένα λειτουργικό σύστημα UNIX βασισμένο στην τεχνολογία που αναπτύχθηκε στο NeXT μέσα στο δεύτερο μισό της δεκαετίας του 1980. Η Apple αγόρασε την εταιρεία στις αρχές του 1997.

Το λειτουργικό σύστημα κυκλοφόρησε για πρώτη φορά το 1999 ως Mac OS X Server 1.0 και ακολούθησε το Μάρτιο του 2001 μια έκδοση προγράμματος-πελάτη (Mac OS X v10.0 "Cheetah"). Από τότε, έχουν κυκλοφορήσει έξι ξεχωριστές εκδόσεις του macOS "client" και "server" του macOS, έως ότου οι δύο συγχωνευτούν στο OS X 10.7 "Lion".

Πριν από τη συγχώνευσή του με το macOS, η έκδοση του server - macOS Server - ήταν αρχιτεκτονικά πανομοιότυπη με την αντίστοιχη επιφάνεια εργασίας του και συνήθως έτρεξε στη γραμμή του Apple για Macintosh. Ο διακομιστής macOS περιλαμβάνει εργαλεία διαχείρισης και διαχείρισης λογισμικού ομάδας εργασίας που παρέχουν απλοποιημένη πρόσβαση σε βασικές υπηρεσίες δικτύου, όπως ένας πράκτορας μεταφοράς αλληλογραφίας, ένας διακομιστής Samba, ένας διακομιστής LDAP, ένας διακομιστής ονομάτων τομέα και άλλοι. Με το Mac OS X v10.7 Lion, όλες οι πτυχές διακομιστών του Mac OS X Server έχουν ενσωματωθεί στην έκδοση προγράμματος-πελάτη και το προϊόν έχει αναδημιουργηθεί ως "OS X" (αφήνοντας "Mac" από το όνομα). Τα εργαλεία διακομιστή προσφέρονται τώρα ως εφαρμογή.

Τα Microsoft Windows είναι μια οικογένεια ιδιόκτητων λειτουργικών συστημάτων σχεδιασμένων από την Microsoft Corporation και απευθύνονται κυρίως σε υπολογιστές που βασίζονται στην αρχιτεκτονική της Intel, με εκτιμώμενο συνολικό ποσοστό χρήσης 88,9% σε υπολογιστές συνδεδεμένους στο Web. Η τελευταία έκδοση είναι τα Windows 10. Το 2011, τα Windows 7 ξεπέρασαν τα Windows XP ως την πιο κοινή έκδοση που χρησιμοποιείται.

Τα Microsoft Windows κυκλοφόρησαν για πρώτη φορά το 1985, ως περιβάλλον λειτουργίας που λειτουργούσε πάνω από το MS-DOS, το οποίο ήταν το τυπικό λειτουργικό σύστημα που κυκλοφόρησε στους περισσότερους προσωπικούς υπολογιστές αρχιτεκτονικής της Intel την εποχή εκείνη. Το 1995 κυκλοφόρησαν τα Windows 95, τα οποία χρησιμοποίησαν μόνο το

MS-DOS ως bootstrap. Για συμβατότητα προς τα πίσω, το Win9x θα μπορούσε να εκτελεί προγράμματα MS-DOS σε πραγματικό τρόπο και προγράμματα οδήγησης Windows 16 bit. Τα Windows ME, κυκλοφόρησε το 2000, ήταν η τελευταία έκδοση της οικογένειας Win9x. Οι μεταγενέστερες εκδόσεις βασίστηκαν στον πυρήνα των Windows NT. Οι τρέχουσες εκδόσεις πελατών των Windows εκτελούνται σε IA-32, x86-64 και 32-bit ARM μικροεπεξεργαστές. Επιπλέον, το Itanium εξακολουθεί να υποστηρίζεται σε παλαιότερη έκδοση διακομιστή Windows Server 2008 R2. Στο παρελθόν, τα Windows NT υποστηρίζουν πρόσθετες αρχιτεκτονικές. Οι εκδόσεις διακομιστών των Windows χρησιμοποιούνται ευρέως. Τα τελευταία χρόνια, η Microsoft έχει δαπανήσει σημαντικά κεφάλαια σε μια προσπάθεια να προωθήσει τη χρήση των Windows ως λειτουργικό σύστημα διακομιστή. Ωστόσο, η χρήση των Windows στους διακομιστές δεν είναι τόσο διαδεδομένη όσο στους προσωπικούς υπολογιστές, καθώς τα Windows ανταγωνίζονται το Linux και το BSD για μερίδιο αγοράς διακομιστή.

Το ReactOS είναι ένα εναλλακτικό λειτουργικό σύστημα των Windows, το οποίο αναπτύσσεται με βάση τις αρχές των Windows - χωρίς να χρησιμοποιεί κανέναν από τους κώδικες της Microsoft.

ii) Λογισμικό ανοιχτού κώδικα

Το λογισμικό ανοιχτού κώδικα (OSS) είναι λογισμικό υπολογιστή που διανέμεται με τον πηγαίο κώδικα του διαθέσιμο για τροποποίηση. Το λογισμικό περιλαμβάνει συνήθως μια άδεια για προγραμματιστές να αλλάζουν το λογισμικό με οποιονδήποτε τρόπο επιλέγουν. Μπορούν να διορθώσουν σφάλματα, να βελτιώσουν λειτουργίες ή να προσαρμόσουν το λογισμικό ώστε να ταιριάζουν στις δικές τους ανάγκες. Η Πρωτοβουλία Ανοικτού Κώδικα (OSI) είναι η κορυφαία αρχή στο OSS. Ο ορισμός τους λογισμικού ανοιχτού κώδικα πληροί τους κανόνες κάτω από 10 κριτήρια.

Αυτά περιλαμβάνουν:

- Αναδιανομή λογισμικού
- Διαθεσιμότητα πηγαίου κώδικα
- Διανομή αδειών
- Ιδιότητες άδειας χρήσης
- Καταπολέμηση των διακρίσεων

Μια λεπτομερής επεξήγηση των 10 διατάξεων μπορεί να βρεθεί στην ιστοσελίδα τους.

Διαφορετικές άδειες επιτρέπουν στους προγραμματιστές να τροποποιούν το λογισμικό με διάφορες συνημμένες συνθήκες. Το OSI εγκρίνει άδειες που συμμορφώνονται με τον ορισμό του λογισμικού ανοιχτού κώδικα. Οι 5 πιο δημοφιλείς άδειες σύμφωνα με τη Γνωσιακή Βάση του Black Duck είναι:

- Άδεια MIT
- Γενική δημόσια άδεια GNU (GPL) 2.0
- Άδεια Apache 2.0
- Γενική δημόσια άδεια GNU (GPL) 3.0

- Άδεια BSD 2.0 (3-ρήτρα, νέα ή αναθεωρημένη)

Όταν αλλάζετε τον πηγαίο κώδικα, μία απαίτηση του OSS είναι η συμπερίληψη των αλλαγών που έχετε αλλάξει καθώς και των μεθόδων σας. Το λογισμικό που δημιουργήθηκε μετά από τροποποιήσεις κώδικα μπορεί ή όχι να μην είναι διαθέσιμο δωρεάν.

- Η διαφορά μεταξύ λογισμικού ανοιχτού κώδικα και εμπορικού λογισμικού:
- Το εμπορικά διαθέσιμο λογισμικό ή το ιδιόκτητο λογισμικό δεν δίνει πρόσβαση στον πηγαίο κώδικα του, επειδή το λογισμικό είναι πνευματική ιδιοκτησία κάποιου άλλου.
- Ως αποτέλεσμα, οι χρήστες συχνά πληρώνουν για αυτό.

Το OSS, από την άλλη πλευρά, είναι μια προσπάθεια συνεργασίας - το λογισμικό είναι κοινόχρηστη πνευματική ιδιοκτησία μεταξύ όλων εκείνων που βοήθησαν στην ανάπτυξη ή την αλλαγή του.

Το Open Source συνεπάγεται την ελεύθερη διαθεσιμότητα πηγαίου κώδικα και διανομής. Το ελεύθερο λογισμικό, με τον ίδιο τρόπο, περιλαμβάνει τροποποίηση κώδικα, αλλά δίνει έμφαση στις ελευθερίες που απολαμβάνουν οι χρήστες να κάνουν ό, τι τους αρέσει με το λογισμικό. Το Ίδρυμα Ελεύθερου Λογισμικού περιγράφει 4 προϋποθέσεις για να θεωρηθεί εντελώς δωρεάν το λογισμικό.

Επιπλέον, το λογισμικό μπορεί να χαρακτηριστεί ως δωρεάν λογισμικό. Αυτό σημαίνει ότι οι χρήστες κατεβάζουν το λογισμικό χωρίς κόστος, αλλά δεν μπορούν να κάνουν αλλαγές στον πηγαίο κώδικα.

Ενώ το κόστος είναι ένας παράγοντας που οδηγεί, το OSS έχει πολλά επιπλέον πλεονεκτήματα:

Τα αποτελέσματα υψηλής ποιότητας όταν ο πηγαίος κώδικας περνά γύρω, δοκιμάζεται και σταθεροποιείται. Είναι μια πολύτιμη ευκαιρία μάθησης για τους προγραμματιστές. Μπορούν να μάθουν και να εφαρμόσουν δεξιότητες στα πιο δημοφιλή προγράμματα που είναι διαθέσιμα σήμερα. Πολλοί θεωρούν το λογισμικό ανοιχτού κώδικα πιο ασφαλές από το ιδιόκτητο λογισμικό, επειδή τα σφάλματα εντοπίζονται και διορθώνονται γρήγορα.

Δεδομένου ότι το λογισμικό ανοιχτού κώδικα είναι δημόσιος τομέας, υπάρχουν ελάχιστες πιθανότητες να μην είναι διαθέσιμες. Αυτό είναι σημαντικό για μακροπρόθεσμα έργα που βασίζονται σε αυτά τα εργαλεία για τη διάρκεια του έργου.

Το μεγαλύτερο μέρος του λογισμικού είναι δωρεάν. Ωστόσο, ενδέχεται να προκύψουν δαπάνες αργότερα, όπως συνδρομές ή τέλη υποστήριξης.

Οι τεχνολογίες ανοιχτού κώδικα βοήθησαν στη δημιουργία ενός μεγάλου μέρους του διαδικτύου.

Επιπλέον, πολλά από τα προγράμματα που χρησιμοποιείτε καθημερινά βασίζονται σε τεχνολογίες ανοιχτού κώδικα. Για παράδειγμα, το λειτουργικό σύστημα Android OS και το OS X της Apple βασίζονται στις τεχνολογίες ανοιχτού κώδικα του πυρήνα και του Unix / BSD, αντίστοιχα.

Αλλα δημοφιλή λογισμικά ανοιχτού κώδικα είναι:

- Το πρόγραμμα περιήγησης ιστού του Firefox του Mozilla

- Πελάτη ηλεκτρονικού ταχυδρομείου Thunderbird
- PHP scripting γλώσσα
- Γλώσσα προγραμματισμού Python
- Apache HTTP διακομιστή ιστού
- Συστήματος βάσεων δεδομένων

Τα έργα OSS είναι ευκαιρίες συνεργασίας που βελτιώνουν τις δεξιότητες και δημιουργούν συνδέσεις στον τομέα. Οι προγραμματιστές πρέπει να είναι ικανοί με τα τυπικά εργαλεία ανάπτυξης ανοιχτού κώδικα.

- Εργαλεία επικοινωνίας - Τα μηνύματα ηλεκτρονικού ταχυδρομείου, τα μηνύματα σε πραγματικό χρόνο, τα φόρουμ και τα wikis βοηθούν τους προγραμματιστές να βρουν λύσεις ή να ξεπερνούν τις ιδέες.
- Κατανεμημένα συστήματα ελέγχου αναθεώρησης - Όταν πολλαπλοί προγραμματιστές σε διαφορετικές γεωγραφικές τοποθεσίες τροποποιούν δεδομένα και αρχεία, τα συστήματα αυτά διαχειρίζονται τις διάφορες εκδόσεις και ενημερώσεις.
- Παρακολούθηση σφαλμάτων και λίστες εργασιών - Επιτρέπει σε έργα μεγάλης κλίμακας να παρακολουθούν τα θέματα και να παρακολουθούν τις διορθώσεις τους.
- Εργαλεία δοκιμής και εντοπισμού σφαλμάτων – Αυτοματοποιεί τις δοκιμές κατά την ολοκλήρωση του συστήματος και διορθώστε τα άλλα προγράμματα.

Χιλιάδες έργα ανοιχτού κώδικα αποτελούν εναλλακτικές λύσεις για το ιδιόκτητο λογισμικό. Εργασία για ένα έργο OSS ως ένα μονοπάτι για την οικοδόμηση μιας σταδιοδρομίας στην ανάπτυξη λογισμικού. Επιπλέον, οι προγραμματιστές μπορούν να βελτιώσουν τις δεξιότητές τους εργαζόμενοι στα μεγαλύτερα προγράμματα λογισμικού στον κόσμο. Το Facebook, το Google και το λογισμικό απελευθέρωσης LinkedIn ως Open Source, έτσι ώστε οι προγραμματιστές να μοιράζονται τη γνώση, να καινοτομούν λύσεις και να συμβάλλουν σε σταθερά, λειτουργικά προϊόντα.

iii) Η Γλώσσα C#

Παρά το γεγονός ότι η C # προέρχεται από την γλώσσα προγραμματισμού C, έχει χαρακτηριστικά όπως η συλλογή απορριμμάτων που επιτρέπουν στους αρχάριους να γίνουν ικανοί στην C # πιο γρήγορα από ό, τι στην C ή στην C ++. Ομοίως και με την Java, είναι object-oriented, έρχεται με μια εκτεταμένη βιβλιοθήκη κατηγοριών, και υποστηρίζει το χειρισμό εξαιρέσεων, πολλαπλούς τύπους πολυμορφισμού, και διαχωρισμός των διεπαφών από τις υλοποιήσεις. Τα χαρακτηριστικά αυτά, σε συνδυασμό με τα ισχυρά εργαλεία της ανάπτυξης, την πολυ-πλατφόρμα υποστήριξης, και τις κοινόχρηστες ονομασίες, καθιστούν την C # μια καλή επιλογή για πολλούς τύπους έργων ανάπτυξης λογισμικού: έργα ταχείας ανάπτυξης εφαρμογών, έργα που υλοποιούνται από μεμονωμένα άτομα ή μικρές ή μεγάλες ομάδες, εφαρμογές Διαδικτύου, και έργα με αυστηρές απαιτήσεις αξιοπιστίας. Τα πλαίσια δοκιμών, όπως τα NUnit κάνουν την C # να επιδέχεται δοκιμή με γνώμονα την ανάπτυξη και, συνεπώς, μια καλή γλώσσα για χρήση για ακραίο προγραμματισμό (XP). Η ισχυρή πληκτρολόγηση του βοηθά στην πρόληψη πολλών λαθών προγραμματισμού που είναι κοινά σε ασθενώς δακτυλογραφημένες γλώσσες. Εξαιτίας αυτών των ομοιοτήτων με άλλες

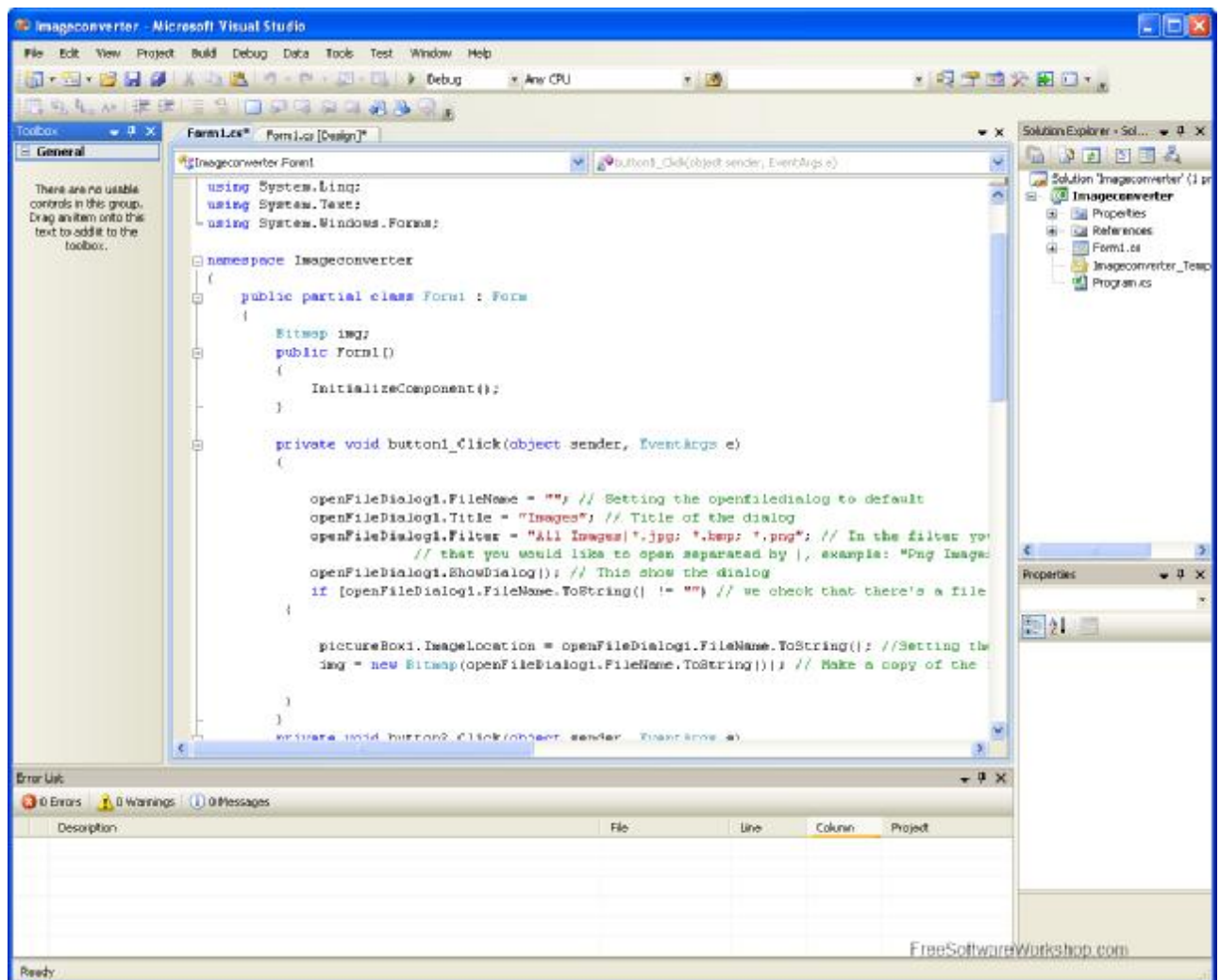
γλώσσες, είναι δυνατόν να εισαχθεί η C # ως γλώσσα με χαρακτηριστικά της C ++ που έχει το ύφος προγραμματισμού της Java και το μοντέλο ταχείας εφαρμογής του BASIC.

Ένα μεγάλο μέρος της δύναμης της C # (όπως με άλλες .NET γλώσσες), έρχεται με το κοινό API του .NET Framework, το οποίο παρέχει ένα μεγάλο σύνολο κατηγοριών, συμπεριλαμβανομένων και εκείνων για την κρυπτογράφηση, το πρωτόκολλο TCP / IP προγραμματισμού υποδοχή, και τα γραφικά. Οι προγραμματιστές μπορούν έτσι να γράψουν μέρος της εφαρμογής σε C # και ένα άλλο μέρος σε άλλη γλώσσα .NET (π.χ. VB.NET), διατηρώντας τα εργαλεία, την βιβλιοθήκη, και το object-oriented μοντέλο ανάπτυξης, ενώ μόνο χρειάζεται να μάθουν τη νέα γλώσσα σύνταξης.

Λόγω των ομοιοτήτων μεταξύ της C # και της οικογένειας των γλωσσών C, καθώς και με την Java, ένας προγραμματιστής, με φόντο τις αντικειμενοστρεφείς γλώσσες όπως η C ++ μπορεί να βρει την δομή και την σύνταξη της C # διαισθητική.

Το .NET Framework είναι ένα κοινό περιβάλλον για την οικοδόμηση, την ανάπτυξη και τη λειτουργία των υπηρεσιών Web, των εφαρμογών Web, των υπηρεσιών των Windows και των εφαρμογών των Windows. Το .NET Framework περιέχει βιβλιοθήκες κοινής κατηγορίας - όπως οι ADO.NET, ASP.NET και οι Windows Forms - για να παρέχει προηγμένου επιπέδου υπηρεσίες που μπορούν να ενσωματωθούν σε μια ποικιλία συστημάτων πληροφορικής.

Τον Ιούνιο του 2000 η Microsoft κυκλοφόρησε τόσο την πλατφόρμα .NET και ένα νέο πρόγραμμα που ονομάζεται γλώσσα C #. Η C # είναι μια γλώσσα OOP γενικής χρήσης που έχει σχεδιαστεί για να δώσει τη βέλτιστη απλότητα, επεκτασιμότητα, και απόδοση. Η σύνταξη της είναι πολύ παρόμοια με την Java, με τη σημαντική διαφορά ότι όλοι οι τύποι μεταβλητών προέρχονται από μια κοινή πρωταρχική τάξη.



Εικόνα 1. Κωδικοποίηση σε γλώσσα C#

Η C # είναι μια γλώσσα από μόνη της. Μπορεί να εκτελεί μαθηματικές και λογικές πράξεις, εκχώρηση μεταβλητής και άλλα αναμενόμενα χαρακτηριστικά μιας γλώσσας προγραμματισμού. Αυτό από μόνο του δεν είναι αρκετά ευέλικτο για πιο σύνθετες εφαρμογές. Σε κάποιο στάδιο, ο κύριος του έργου θα θέλει να αλληλεπιδρά με το σύστημα υποδοχής είτε πρόκειται για την ανάγνωση αρχείων ή την λήψη περιεχομένου από το Διαδίκτυο. Το .NET Framework είναι ένα σύνολο εργαλείων που αναπτύχθηκε για την πλατφόρμα των Windows για να επιτρέψει στον προγραμματιστή να αλληλεπιδρά με το σύστημα υποδοχής ή οποιαδήποτε εξωτερική οντότητα, είτε είναι μια άλλη διαδικασία, είτε ένας άλλος υπολογιστής. Η πλατφόρμα .NET είναι μια συγκεκριμένη πλατφόρμα εφαρμογής των Windows. Άλλα λειτουργικά συστήματα έχουν τις δικές τους υλοποιήσεις λόγω των διαφορών στα λειτουργικά συστήματα διαχείρισης I/O, τα μοντέλα ασφάλειας και τις διασυνδέσεις.

Η βασική μονάδα εκτέλεσης σε ένα C # πρόγραμμα είναι η δήλωση. Μια δήλωση μπορεί να δηλώσει μια μεταβλητή, να ορίσει μια έκφραση, να εκτελέσει μια απλή δράση καλώντας μια μέθοδο, να ελέγξει την ροή της εκτέλεσης των άλλων καταστάσεων, να δημιουργήσει ένα αντικείμενο, ή να ορίσει μια τιμή σε μια μεταβλητή, ιδιότητα, ή πεδίο. Οι δηλώσεις συνήθως

τερματίζονται με ένα ελληνικό ερωτηματικό. Οι δηλώσεις μπορούν να ομαδοποιηθούν σε λίστες δηλώσεων χωρισμένες με κόμμα ή μπλοκ δήλωση περικλεισμένα με αγκύλες.

Παραδείγματα:

```
int sampleVariable;           // declaring a variable
sampleVariable = 5;           // assigning a value
Method();                     // calling an instance method
SampleClass sampleObject = new SampleClass(); // creating a new instance of an object
sampleObject.ObjectMethod();  // calling a member function of an object

// executing a "for" loop with an embedded "if" statement
for (int i = 0; i < upperLimit; i++)
{
    if (SampleClass.SampleStaticMethodReturningBoolean(i))
    {
        sum += sampleObject.SampleMethodReturningInteger(i);
    }
}
```

Μια σειρά από δηλώσεις που περιβάλλονται από αγκύλες σχηματίζουν ένα μπλοκ κώδικα. Μεταξύ άλλων σκοπών, τα μπλοκ κώδικα χρησιμεύουν στο να περιορίσουν το πεδίο ή το εύρος στο οποίο μπορεί να χρησιμοποιηθεί μία μεταβλητή. Μια μεταβλητή είναι προσβάσιμη μόνο στο μπλοκ στο οποίο ορίζεται. Τα μπλοκ κώδικα μπορεί να είναι ένθετα και συχνά εμφανίζονται ως φορείς των μεθόδων.

```
private void MyMethod (int integerValue)
{ // This block of code is the body of "MyMethod()"

    // The 'integerValue' integer parameter is accessible to everything in the method

    int methodLevelVariable; // This variable is accessible to everything in the method

    if (integerValue == 2)
    {
        // methodLevelVariable is still accessible here

        int limitedVariable; // This variable is only accessible to code in the, if block

        DoSomeWork(limitedVariable);
    }

    // limitedVariable is no longer accessible here

} // Here ends the code block for the body of "MyMethod()".
```

Τα σχόλια επιτρέπουν την inline τεκμηρίωση του πηγαίου κώδικα. Η C # compiler αγνοεί σχόλια. Αυτά τα στυλ των επιτρέπονται σχόλια σε C #:

Σχόλια μίας γραμμής

Η // ακολουθία χαρακτήρων σηματοδοτεί το ακόλουθο κείμενο ως σχόλιο μιας γραμμής. Τα σχόλια μιας γραμμής, όπως θα περίμενε κανείς, τελειώνουν στο πρώτο άκρο της γραμμής μετά τον δείκτη σχολίου//.

Σχόλια πολλών γραμμών

Τα σχόλια μπορεί να εκτείνονται σε πολλαπλές γραμμές με τη χρήση του στυλ σχολίου πολλαπλής γραμμής. Οι παρατηρήσεις αυτές ξεκινούν με / * και τελειώνουν με */. Το κείμενο μεταξύ των δεικτών σχολίων πολλαπλών γραμμών είναι το σχόλιο.

XML σχόλια γραμμής τεκμηρίωσης

Τα σχόλια αυτά χρησιμοποιούνται για να παράγουν έγγραφα XML. Μπορεί να χρησιμοποιηθεί ενιαία γραμμή και στυλ πολλών γραμμών. Το στυλ μιας γραμμής, όπου κάθε γραμμή του σχολίου ξεκινά με //, είναι πιο συχνό από ό, τι το στυλ πολλών γραμμών που οριοθετείται από / ** και * /.

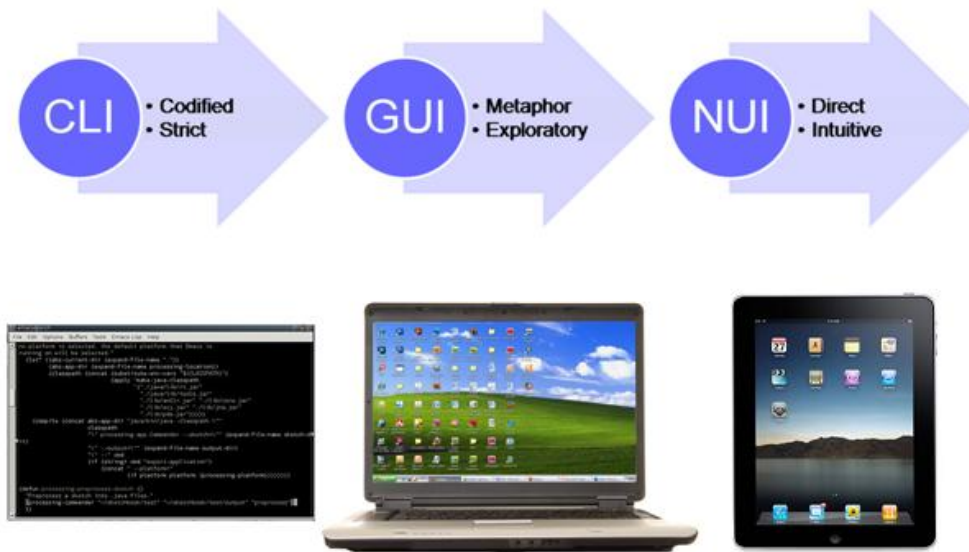
iv) Φυσική διεπαφή χρήστη (NUI)

Στους ηλεκτρονικούς υπολογιστές, ένα φυσικό περιβάλλον χρήστη, ή NUI, ή φυσική Διασύνδεση είναι η καθομιλουμένη που χρησιμοποιείται από τους σχεδιαστές και τους προγραμματιστές των διεπαφών ανθρώπου-μηχανής για να αναφερθούν σε μια διεπαφή χρήστη που είναι ουσιαστικά αόρατη, και παραμένει αόρατη καθώς ο χρήστης μαθαίνει συνεχώς όλο και πιο πολύπλοκες αλληλεπιδράσεις. Η λέξη φυσικό χρησιμοποιείται επειδή οι περισσότερες διεπαφές υπολογιστών χρησιμοποιούν συσκευές τεχνητού ελέγχου των οποίων η λειτουργία πρέπει να μαθευτεί.

Μια φυσική Διασύνδεση στηρίζεται σε έναν χρήστη, στο να είναι σε θέση να περάσει γρήγορα από το στάδιο του αρχάριου μέχρι αυτό του ειδικού. Ενώ η διασύνδεση απαιτεί μάθηση, η μάθηση διευκολύνεται μέσω του σχεδιασμού που δίνει στον χρήστη την αίσθηση ότι είναι άμεσα και συνεχώς επιτυχής. Έτσι, η λέξη «φυσικό» αναφέρεται σε ένα στόχο στην εμπειρία του χρήστη - ότι η αλληλεπίδραση έρχεται φυσικά, ενώ αλληλεπιδρά με την τεχνολογία, και όχι ότι η ίδια η διεπαφή είναι φυσική. Αυτό έρχεται σε αντίθεση με την ιδέα ενός διαισθητικού interface, αναφερόμενοι σε κάτι που μπορεί να χρησιμοποιηθεί χωρίς προηγούμενη μάθηση.

Αρκετές στρατηγικές σχεδιασμού έχουν προταθεί που έχουν εκπληρώσει το στόχο αυτό σε ποικίλους βαθμούς επιτυχίας. Μία στρατηγική είναι η χρήση μιας «διεπαφής πραγματικότητας χρήστη» (RUI), επίσης γνωστή ως μέθοδοι «διασυνδέσεων με βάση την πραγματικότητα» (RBI). Ένα παράδειγμα μιας στρατηγικής RUI είναι να χρησιμοποιήσει κανείς έναν υπολογιστή που φοριέται για να καταστήσει τα πραγματικά αντικείμενα "clickable", δηλαδή έτσι ώστε ο χρήστης να μπορεί να κάνει κλικ σε οποιοδήποτε καθημερινό αντικείμενο, έτσι ώστε να λειτουργεί ως υπερσύνδεσμος, συγχωνεύοντας έτσι τον κυβερνοχώρο και τον πραγματικό κόσμο. Επειδή ο όρος «φυσικό» υπαινίσσεται τον «φυσικό κόσμο», η RBI συχνά συγχέεται με την NUI, όταν στην πραγματικότητα είναι απλώς ένα από

τα μέσα για την επίτευξή της.



Εικόνα 2. Η εξέλιξη των διεπαφών χρήστη

Ένα παράδειγμα μιας στρατηγικής για τον σχεδιασμό μιας NUI που δεν βασίζεται στην RBI είναι ο αυστηρός περιορισμός της λειτουργικότητας και προσαρμογής, έτσι ώστε οι χρήστες να έχουν πολύ λίγα για να μάθουν για τη λειτουργία μιας συσκευής. Υπό την προϋπόθεση ότι οι προεπιλεγμένες δυνατότητες ταιριάζουν με τους στόχους του χρήστη, η διεπαφή είναι εύκολη στη χρήση. Αυτή είναι μια συνολική στρατηγική σχεδιασμού στο iOS της Apple. Επειδή ο σχεδιασμός αυτός συμπίπτει με μια προβολή οθόνης αφής, οι μη-σχεδιαστές συνήθως λάθος αποδίδουν την αβίαστη αλληλεπίδραση με τη συσκευή στην προβολή πολλαπλής αφής, και όχι στον σχεδιασμό του λογισμικού, που ισχύει στην πραγματικότητα.

Αντιληπτικό Pixel

Ένα παράδειγμα είναι το έργο που πραγματοποιήθηκε από τον Jefferson Han σχετικά με τις διεπαφές πολλαπλής αφής. Σε μια επίδειξη στην TED το 2006, έδειξε μια ποικιλία μέσων αλληλεπίδρασης με το περιεχόμενο στην οθόνη, χρησιμοποιώντας τόσο άμεσους χειρισμούς όσο και χειρονομίες. Για παράδειγμα, για να διαμορφώσει μια κολλώδη μάζα στην οθόνη, ο Jefferson κυριολεκτικά τσιμπούσε και έσπρωχνε με τα δάχτυλά του. Σε μια διεπαφή GUI για εφαρμογή σχεδίου για παράδειγμα, ένας χρήστης θα χρησιμοποιήσει την μεταφορά των «εργαλείων» για να το κάνει αυτό, για παράδειγμα, επιλέγοντας ένα εργαλείο σπρωξίματος, ή επιλέγοντας δύο τμήματα της μάζας στα οποία στην συνέχεια θα θελήσει να εφαρμόσει κάποια ένταση. Ο Χαν έδειξε ότι η αλληλεπίδραση του χρήστη θα μπορούσε να είναι πολύ πιο έξυπνη με την κατάργηση των συσκευών αλληλεπίδρασης που έχουμε συνηθίσει και την αντικατάστασή τους με μια οθόνη που είναι ικανή να ανιχνεύσει ένα πολύ ευρύτερο φάσμα ανθρώπινων πράξεων και χειρονομιών. Φυσικά, αυτό επιτρέπει μόνο ένα πολύ περιορισμένο σύνολο αλληλεπιδράσεων που τακτοποιημένα τις χαρτογραφούν σε φυσικό χειρισμό (RBI). Η επέκταση των δυνατοτήτων του λογισμικού πέρα από τις φυσικές ενέργειες απαιτεί σημαντικά περισσότερη μελέτη του έργου.

Η Microsoft PixelSense

Η Microsoft PixelSense παίρνει παρόμοιες ιδέες για το πώς οι χρήστες αλληλεπιδρούν με το περιεχόμενο, αλλά προσθέτει την ικανότητα της συσκευής να αναγνωρίσει οπτικά τα αντικείμενα που τοποθετούνται πάνω της. Με τον τρόπο αυτό, οι χρήστες μπορούν να ενεργοποιήσουν ενέργειες στον υπολογιστή μέσα από τις ίδιες χειρονομίες και κινήσεις, όπως επέτρεπε η οθόνη αφής του Jefferson Han, αλλά αντιτίθεται επίσης στο να γίνει μέρος των μηχανισμών ελέγχου. Έτσι, για παράδειγμα, όταν τοποθετείται ένα ποτήρι κρασί πάνω στο τραπέζι, ο υπολογιστής το αναγνωρίζει ως έχει και εμφανίζει το περιεχόμενο που σχετίζεται με το εν λόγω ποτήρι κρασί. Τοποθετώντας ένα ποτήρι κρασί σε ένα τραπέζι χαρτογραφούνται δράσεις που έχουν αναληφθεί με ποτήρια κρασιού και άλλα τραπέζια, και έτσι χαρτογραφούνται σωστά οι διεπαφές βασίζεται στην πραγματικότητα. Έτσι, θα μπορούσε να θεωρηθεί ως μια εισαγωγή στην εμπειρία NUI.

Μοναδική 3D αφή

Η μοναδική 3D αφή ορίζεται ως η άμεση χειραγώγηση αντικείμενων 3D εικονικού περιβάλλοντος που χρησιμοποιούν hardware απλής επιφάνειας ή επιφάνειας πολλαπλής αφής σε 3D εικονικά περιβάλλοντα πολλαπλών χρηστών. Επινοήθηκε για πρώτη φορά το 2007 για να περιγράψει και να προσδιορίσει τις αρχές μάθησης της 3D διεπαφής φυσικού χρήστη που συσχετίζεται με την Edusim. Η διεπαφή φυσικού χρήστη μοναδικής αφής φαίνεται τώρα να λαμβάνει μια ευρύτερη εστίαση και σημασία, με την ευρύτερη προσαρμογή της επιφάνειας αφής και με γνώμονα το υλικό, όπως το iPhone, το iPod touch, το iPad, και μια αυξανόμενη λίστα άλλου υλικού. Η Apple φαίνεται, επίσης, να λαμβάνει έντονο ενδιαφέρον για την διασυνδέσεις φυσικού χρήστη μοναδικής 3D αφής, κατά την διάρκεια των τελευταίων ετών. Το έργο αυτό βασίζεται πάνω στην ευρύτερη ακαδημαϊκή βάση, η οποία έχει μελετήσει την 3D χειραγώγηση σε περιβάλλοντα εικονικής πραγματικότητας.

v) Visual Studio

Το Microsoft Visual Studio είναι ένα ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) από τη Microsoft. Χρησιμοποιείται για την ανάπτυξη προγραμμάτων ηλεκτρονικών υπολογιστών για τα Microsoft Windows, καθώς και ιστοσελίδων, διαδικτυακών εφαρμογών και υπηρεσιών web. Το Visual Studio χρησιμοποιεί πλατφόρμες ανάπτυξης λογισμικού της Microsoft, όπως το Windows API, τα Windows Forms, το Windows Presentation Foundation, το Windows Store και το Microsoft Silverlight. Μπορεί να παράγει τόσο εγγενή κώδικα όσο και διαχειριζόμενο κώδικα.

Το Visual Studio περιλαμβάνει έναν επεξεργαστή κώδικα με υποστήριξη IntelliSense (συνιστώσα ολοκλήρωσης κώδικα), καθώς και τον κώδικα refactoring. Το ολοκληρωμένο πρόγραμμα εντοπισμού σφαλμάτων λειτουργεί τόσο ως ένα πρόγραμμα εντοπισμού σφαλμάτων πηγής επιπέδου όσο και ως ένα πρόγραμμα εντοπισμού σφαλμάτων επίπεδο μηχανής. Άλλα ενσωματωμένα εργαλεία περιλαμβάνουν μορφές σχεδιαστή για τη δημιουργία εφαρμογών GUI, σχεδιαστή ιστοσελίδων, σχεδιαστή τάξεων, και σχεδιαστή σχήματος βάσης δεδομένων. Δέχεται plug-ins που ενισχύουν τη λειτουργικότητα σχεδόν σε κάθε επίπεδο, συμπεριλαμβανομένης της πρόσθεσης υποστήριξης για συστήματα πηγής ελέγχου (όπως η ανατροπή) και της προσθήκης νέων toolsets όπως εκδότες και visual designers για συγκεκριμένους τομείς ή γλώσσες toolsets για άλλες πτυχές του κύκλου ανάπτυξης λογισμικού (όπως τον client του Team Foundation Server: Team Explorer).

Το Visual Studio υποστηρίζει διαφορετικές γλώσσες προγραμματισμού και επιτρέπει τον επεξεργαστή κώδικα και τον εντοπισμό σφαλμάτων για την υποστήριξη (σε διάφορους βαθμούς) σχεδόν οποιασδήποτε γλώσσα προγραμματισμού, εφόσον υπάρχει μια υπηρεσία εξειδικευμένης γλώσσας. Οι built-in γλώσσες περιλαμβάνουν τις C, C ++ και C ++ / CLI (μέσω της Visual C ++), VB.NET (μέσω της Visual Basic .NET), C# (μέσω Visual C#), και F # (όπως του Visual Studio 2010). Υποστήριξη για άλλες γλώσσες, όπως η M, Python, Ruby και, μεταξύ άλλων, είναι διαθέσιμες μέσω των υπηρεσιών γλώσσας που έχουν εγκατασταθεί ξεχωριστά. Υποστηρίζει, επίσης, XML / XSLT, HTML / XHTML, JavaScript και CSS.

Η Microsoft παρέχει "Express" εκδόσεις του Visual Studio χωρίς κανένα κόστος. Οι εμπορικές εκδόσεις του Visual Studio μαζί με επιλεγμένες εκδόσεις στο παρελθόν είναι διαθέσιμες δωρεάν στους φοιτητές μέσω του προγράμματος DreamSpark της Microsoft.

vi) App Inventor

Το App Inventor για Android είναι ένα περιβάλλον οπτικού προγραμματισμού που παρεχόταν αρχικά από την Google, και τώρα διατίθεται και υποστηρίζεται από το MIT. Το Google App Inventor διατέθηκε στο κοινό για πρώτη φορά το τέλος του 2010 και μετά από ένα χρόνο στις 31 Δεκεμβρίου, 2011 η Google τερμάτισε την υποστήριξη του προγράμματος και το έκανε open source. Το App Inventor ξανά διατέθηκε στο κοινό, πλέον με όνομα MIT App Inventor ως beta service (τον Μάρτιο του 2012), από το MIT Center for Mobile Learning μετά την ανάληψη του έργου από την Google.

Το App Inventor είναι ένα on-line περιβάλλον προγραμματισμού. Πιο συγκεκριμένα για την ανάπτυξη εφαρμογών με το App Inventor είναι απαραίτητη η χρήση ενός φυλλομετρητή (web browser) και ενός συνδεδεμένου τηλεφώνου με Android ή ενός εξομοιωτή τηλεφώνου.

Το περιβάλλον του App Inventor επιτρέπει σε οποιονδήποτε είναι εξοικειωμένος με τον προγραμματισμό ηλεκτρονικών υπολογιστών να δημιουργεί εφαρμογές για λειτουργικό σύστημα Android. Χρησιμοποιεί ένα γραφικό περιβάλλον αλληλεπίδρασης με τον χρήστη, παρόμοιο με του Scratch και της StarLogo TNG, το οποίο επιτρέπει στους χρήστες να σέρνουν και να αφήνουν πλακίδια δημιουργώντας έτσι εφαρμογές που μπορούν να τρέξουν σε Android, το λειτουργικό σύστημα σήμερα πολλών φορητών συσκευών.

Ειδικότερα το γραφικό περιβάλλον προγραμματισμού του App Inventor αποτελείται από:

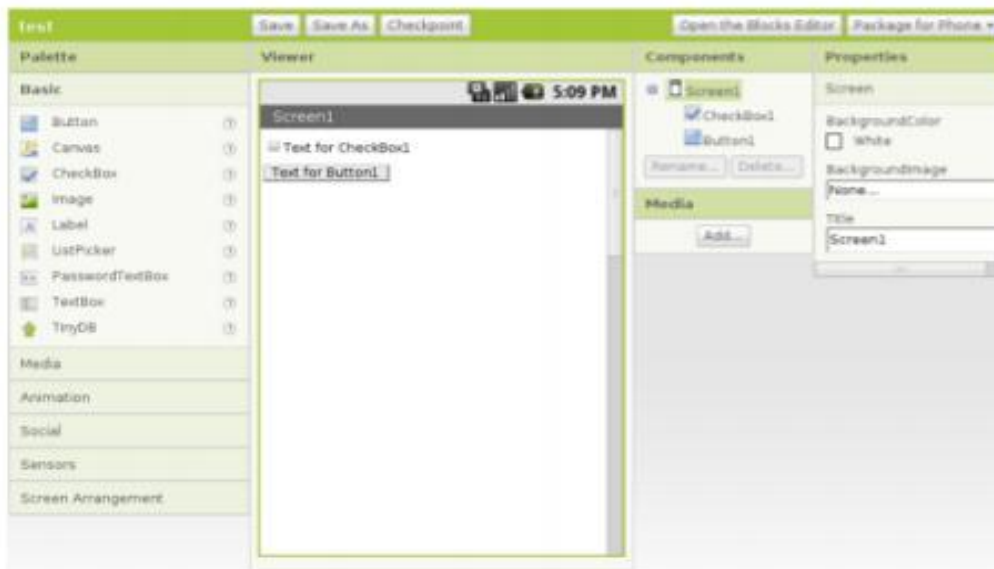
Τον App Inventor Designer, όπου επιλέγονται τα βασικά δομικά στοιχεία της εφαρμογής που θα δημιουργηθεί

Τον App Inventor Blocks Editor, όπου λαμβάνει χώρα ο καθορισμός – προγραμματισμός της συμπεριφοράς των βασικών δομικών στοιχείων της εφαρμογής με σύρσιμο πλακιδίων. Εδώ ο προγραμματισμός συμβαίνει με δόμηση των προγραμμάτων με οπτικό τρόπο, ταιριάζοντας πλακίδια μαζί όπως τα κομμάτια ενός πάζλ.

Η εφαρμογή που δημιουργείται εμφανίζεται στο συνδεδεμένο τηλέφωνο ή στον εξομοιωτή τηλεφώνου, βήμα-προς-βήμα, όπως προστίθενται κομμάτια σε αυτή, ώστε να είναι δυνατός ο έλεγχος της εφαρμογής καθώς αυτή χτίζεται. Με την ολοκλήρωση της εφαρμογής, υπάρχει η δυνατότητα πακεταρίσματος της εφαρμογής έτσι ώστε να είναι δυνατή η εγκατάσταση της σε Android συσκευές.

Όπως αναφέρθηκε και προηγουμένως το App Inventor είναι web εφαρμογή και αυτό είναι ιδιαίτερα σημαντικό αφού παρέχει τρία σημαντικά πλεονεκτήματα. Μπορεί να λειτουργήσει από οποιοδήποτε Η/Υ που έχει πρόσβαση στο διαδίκτυο αρκεί να έχει εγκατεστημένο ένα περιηγητή. Ο περιορισμός που υπάρχει είναι, ότι οι μόνοι περιηγητές που είναι συμβατοί προς το παρόν με το App Inventor, είναι ο Mozilla Firefox και ο Google Chrome. Ο χρήστης δε χρειάζεται να εγκαταστήσει την εφαρμογή στον Η/Υ του. Επίσης, οποτεδήποτε ενσωματώνεται μια αναβάθμιση, δε χρειάζεται ο χρήστης να την κατεβάσει και να την εγκαταστήσει, αφού όλα γίνονται αυτόματα χωρίς εκείνος να επέμβει στη διαδικασία. Όταν θα μπει στο App Inventor θα ενημερωθεί ότι έχει γίνει αναβάθμιση. Ακόμα, όλα τα προγράμματα που γράφει ο χρήστης αποθηκεύονται στο cloud και έτσι μπορεί να έχει πρόσβαση σε αυτά, από οποιαδήποτε συσκευή (υπολογιστής, ταμπλέτα, έξυπνο κινητό τηλέφωνο) αρκεί να συνδέεται με το internet. Παράλληλα η αποθήκευση στο cloud του επιτρέπει το χρήστη να έχει αντίγραφα ασφαλείας των εφαρμογών του.

Το App Inventor 2 έχει αρκετές διαφορές από τον προκάτοχό του. Προστέθηκαν νέες λειτουργίες και βελτιώθηκε το μενού των επιλογών στο οποίο έχει πρόσβαση ο χρήστης. Στη συνέχεια περιγράφονται τα πιο ενδιαφέροντα. Η βασική αλλαγή είναι το web περιβάλλον, σε αντίθεση με την έκδοση 1 που απαιτούσε από το χρήστη την εγκατάσταση της εφαρμογής στον υπολογιστή του. Ο τύπος αρχείου του πηγαίου κώδικα έγινε aia σε αντίθεση με το zip που ίσχυε. Ο χρήστης μπορεί να προσθέσει νέες στο Designer Mode ή στο Blocks Mode (που στην προηγούμενη έκδοση δεν επιτρεπόταν). Στη σελίδα των μπλοκ, η ταξινόμηση τους έχει αλλάξει και τοποθετήθηκαν στα αριστερά της οθόνης. Οι διαδικασίες (μοιάζουν πολύ με τις μεθόδους), απέκτησαν δικιά τους κατηγορία. Επιπλέον μικρές αλλά σημαντικές αλλαγές, που βοηθούν το χρήστη έγιναν στα χρώματα, την προσομοίωση, στις μεταβλητές αλλά και σε ορισμένα μπλοκ.



Εικόνα 3. Περιβάλλον του App Inventor

Το μενού της εφαρμογής, περιλαμβάνει ένα γραφικό περιβάλλον που χωρίζεται σε τρία τμήματα. Στο κεντρικό τμήμα, κυριαρχεί η (υποθετική) οθόνη του smartphone. Στα αριστερά, υπάρχουν οι κατηγορίες με τα αντικείμενα που μπορεί να προσθέσει στην οθόνη, με τη χρήση της μεθόδου drag and drop και στα δεξιά υπάρχουν οι καρτέλες ιδιοτήτων και συστατικών

όπου ο χρήστης μπορεί να ρυθμίσει το κάθε αντικείμενο, που έχει προσθέσει, ξεχωριστά. Η λογική που πρέπει να ακολουθήσει ο χρήστης, είναι να φτιάξει πρώτα το γραφικό περιβάλλον της εφαρμογής του, επιλέγοντας τα χαρακτηριστικά που θέλει και στη συνέχεια να μεταβεί από το χώρο σχεδίασεως στο χώρο των μπλοκ όπου εκεί θα δηλώσει μεταβλητές και λειτουργίες. Στο πάνω μέρος της σελίδας υπάρχει ένα μενού από drop down τα οποία παρέχουν επιπρόσθετες λειτουργίες. Τα μενού αυτά είναι Project από το οποίο δημιουργείται νέο project, εισαγωγή project από τον υπολογιστή του χρήστη, διαγραφή και αποθήκευση. Το μενού Connect επιτρέπει τη δοκιμή της εφαρμογής σε προσομοιωτή, ή σε φορητή συσκευή. Το μενού Build παρέχει στο χρήστη την εφαρμογή σε δύο μορφές. Η πρώτη δείχνει ένα κώδικα QR ώστε ο χρήστης να κατεβάσει την εφαρμογή στη συσκευή (άμεσα) και η δεύτερη παράγει το apk ώστε να μεταφερθεί στη συσκευή έμμεσα, π.χ. μέσω καλωδίου. Το Help εμφανίζει πληροφορίες έκδοσης του App Inventor και συντομεύσεις για τις βιβλιοθήκες, το φόρουμ και τα μαθήματα εκμάθησης. Το My Projects εμφανίζει μια λίστα με όλα τα projects που έχει δημιουργήσει ο χρήστης και εμφανίζει ημερομηνίες δημιουργίας και τελευταίας τροποποίησης. Το Guide ανοίγει ένα νέο παράθυρο με το documentation της εφαρμογής και τέλος, το Report an Issue χρησιμεύει στην εμφάνιση των πιο συχνών προβλημάτων που αντιμετωπίζουν οι χρήστες και στην επικοινωνία με τον κατασκευαστή του App Inventor.

Στα δεξιά της οθόνης υπάρχουν τα Συστατικά, οι Ιδιότητες και τα Μέσα. Τα συστατικά, είναι μια λίστα που περιλαμβάνει όλα τα αντικείμενα που έχει τοποθετήσει ο χρήστης στην οθόνη του. Αυτά, ταξινομούνται ανάλογα με τη θέση τους και είναι δυνατή η αλλαγή της προεπιλεγμένης ονομασίας σε αυτή που επιθυμεί ο χρήστης. Οι ονομασίες αυτές είναι πολύ χρήσιμες γιατί αυτές θα χρησιμοποιηθούν στο προγραμματισμό. Οι ιδιότητες, αναφέρονται σε κάθε ένα από τα αντικείμενα που έχει βάλει στην οθόνη ο χρήστης. Οι ιδιότητες περιλαμβάνουν χρώματα, μεγέθη, αλλαγή διάταξης, βοηθητικά μηνύματα για το χρήστη. Τα Μέσα, περιλαμβάνουν όλες τις εικόνες που χρησιμοποιεί η εφαρμογή. Αυτές τις ανεβάζει ο χρήστης και στη συνέχεια τις τοποθετεί εκεί που επιθυμεί μέσω του προγράμματος. Ο χρήστης σχεδιάζει το περιβάλλον της κάθε οθόνης όπως επιθυμεί με τη χρήση των συστατικών που αναφέρθηκαν παραπάνω. Στη συνέχεια, για την κάθε μια οθόνη μεταβαίνει στην καρτέλα των Blocks.



Εικ

όνα 4. Το περιβάλλον των blocks

Η καρτέλα χωρίζεται σε δύο τμήματα όπου στο αριστερό βρίσκονται τα μπλοκ και στην υπόλοιπη οθόνη είναι το περιβάλλον στο οποίο ο χρήστης τα τοποθετεί. Τα μπλοκ χωρίζονται σε δύο κατηγορίες. Στην πρώτη είναι τα ανώνυμα μπλοκ τα οποία ο χρήστης τα χρησιμοποιεί για να εκτελέσει διάφορες λειτουργίες. Έτσι λοιπόν, αυτά αφορούν τον έλεγχο με τη χρήση της επαναληπτικής δομή while αλλά και των συνθηκών διακλάδωσης if, if-then-else. Επίσης στη σελίδα αυτή, είναι διαθέσιμα μπλοκ όπως το άνοιγμα άλλης οθόνης, κλείσιμο οθόνης και κλείσιμο εφαρμογής. Όλα τα προαναφερθέντα μπλοκ έχουν πορτοκαλί χρώμα. Στη συνέχεια είναι τα μπλοκ των λογικών πράξεων. And, or, not, =, >, <, και οι τιμές true, false. Το χρώμα των μπλοκ είναι πράσινο. Ακολουθούν τα μαθηματικά μπλοκ, με όλες τις πράξεις, τιμές, τυχαίους αριθμούς, στρογγυλοποιήσεις, τριγωνομετρικούς αριθμούς, όπως επίσης και έλεγχος αν το όρισμα είναι αριθμός. Το χρώμα αυτών των μπλοκ είναι μπλε. Στη συνέχεια είναι τα μπλοκ κειμένου. Οι εντολές ελέγχουν για αλφαριθμητικά, μετατρέπουν κεφαλαία σε πεζά και το αντίστροφο, διασπούν ένα αλφαριθμητικό στα κενά του και αντικαθιστούν ένα κείμενο. Το χρώμα που αντιστοιχεί στα μπλοκ αυτά, είναι το φούξια. Ακολουθούν οι λίστες. Ο χρήστης μπορεί να δημιουργήσει κενές λίστες, ή να τοποθετήσει αντικείμενα σ' αυτές εξ αρχής. Μπορεί να γίνει έλεγχος αν ένα αντικείμενο υπάρχει ήδη στη λίστα, όπως επίσης και να μετρηθούν τα αντικείμενα μιας λίστας. Ο χρήστης με κατάλληλες εντολές μπορεί να προσθέσει και να αλλάξει κάποια από τα στοιχεία της λίστας, να αντιγράψει τα στοιχεία μιας λίστας σε μια άλλη και να μεταφερθούν δεδομένα από πίνακες CSV και ακόμα να κάνει αναζήτηση ενός αντικειμένου σε μια λίστα. Το χρώμα των λιστών είναι γαλάζιο. Το μπλοκ των χρωμάτων επιτρέπει την εμφάνιση των 13 πιο γνωστών χρωμάτων. Τα μπλοκ των μεταβλητών, επιτρέπουν τη δημιουργία, αρχικοποίηση, τοπικών και καθολικών μεταβλητών. Το χρώμα των μεταβλητών είναι πορτοκαλί. Τελευταίο μπλοκ είναι οι διαδικασίες. Το χρώμα τους είναι το μωβ.

Η δεύτερη κατηγορία των μπλοκ, σχετίζεται άμεσα με τα αντικείμενα που τοποθετήθηκαν σε μια οθόνη από τον ίδιο το χρήστη. Για το κάθε ένα αντικείμενο της οθόνης, διατίθεται ένα σετ εντολών. Για παράδειγμα, υπάρχουν μπλοκ που αφορούν τη διαχείριση της οθόνης που προβάλλεται στο χρήστη και οι εντολές αφορούν την εμφάνιση μηνυμάτων και την εκτέλεση άλλων διαδικασιών. Τέτοιες είναι οι τίτλοι, αλλαγές χρωμάτων, άνοιγμα και κλείσιμο μια οθόνης και χρήση εικόνας ως φόντο. Η ύπαρξη τουλάχιστον μιας οθόνης σε μια εφαρμογή είναι δεδομένη. Τα μπλοκ που αφορούν την κάθε οθόνη σχετίζονται με το άνοιγμα και κλείσιμο μιας άλλης, ή με λεπτομέρειες που καθορίζουν την ίδια. Το τελευταίο σημαίνει ότι ρυθμίζονται παράμετροι, όπως το φόντο, οι εικόνες, τα χρώματα και η σχεδιοκίνηση. Μπορούν να ρυθμιστούν ο προσανατολισμός, το ύψος και το πλάτος της οθόνης. Για τα αντικείμενα εικόνας, τα μπλοκ αφορούν τη μεταβολή του μεγέθους της, αντικατάστασή της από άλλη και προσθήκη σχεδιοκίνησης. Ο χρήστης επιπλέον, μπορεί να επιλέξει πότε η εικόνα θα είναι ορατή και πότε όχι, όπως επίσης και ποια μηνύματα θα εμφανίζονται στην οθόνη. Τα αντικείμενα διάταξης έχουν μπλοκ που σχετίζονται με τον τρόπο εμφάνισης της πληροφορίας.

Για να δοκιμάσει ο χρήστης την εφαρμογή που έχει φτιάξει, μπορεί να επιλέξει έναν από τους τρεις τρόπους που περιγράφονται. Ο πρώτος από αυτούς είναι η σύνδεση του App Inventor με τη συσκευή μέσω Wi-Fi. Ο χρήστης βλέπει την εκτέλεση της εφαρμογής του και σε πραγματικό χρόνο μπορεί να επέμβει στον κώδικα και να κάνει αλλαγές. Οι αλλαγές περνούν αυτόματα στη συσκευή. Για να λειτουργήσει αυτή η μέθοδος, πρέπει στη συσκευή να εγκατασταθεί το MIT AI2 Companion App που είναι διαθέσιμο στο Google Play. Αμέσως μετά πραγματοποιείται η σύνδεση μεταξύ του App Inventor και της συσκευής μέσω κώδικα QR ή με την εισαγωγή στη συσκευή ενός κωδικού που παρέχει το App Inventor. Αμέσως μετά, η εφαρμογή πρέπει να εκτελεστεί κανονικά και παράλληλα η διεύθυνση IP της

εμφανίζεται και στον υπολογιστή. Σε οποιαδήποτε οθόνη της εφαρμογής και αν βρίσκεται ο χρήστης, ο κώδικας της εμφανίζεται στην οθόνη του υπολογιστή του.

Ο δεύτερος τρόπος δοκιμής είναι η χρήση του προσομοιωτή. Αυτός ο τρόπος επίσης προτείνεται γι' αυτούς που θέλουν να φτιάξουν εφαρμογές χωρίς να διαθέτουν κάποια συσκευή που να εκτελεί το ΛΣ Android. Το μειονέκτημα του προσομοιωτή είναι ότι αργεί να δείξει άλλες οθόνες και πολλές φορές κολλάει. Η δε εκκίνηση λειτουργία του προσομοιωτή χρειάζεται περίπου δύο λεπτά για να εκκινήσει. Για να γίνει η χρήση του προσομοιωτή πρέπει ο χρήστης να εγκαταστήσει στον υπολογιστή του το aiStarter, που είναι διαθέσιμο μόνο για Windows και Linux. Οποιαδήποτε στιγμή, ο χρήστης μπορεί να δημιουργήσει το apk αρχείο και να το μεταφέρει στη συσκευή του για να δοκιμάσει την εφαρμογή. Ο τρίτος τρόπος δοκιμής είναι η σύνδεση της συσκευής μέσω USB με τον υπολογιστή.

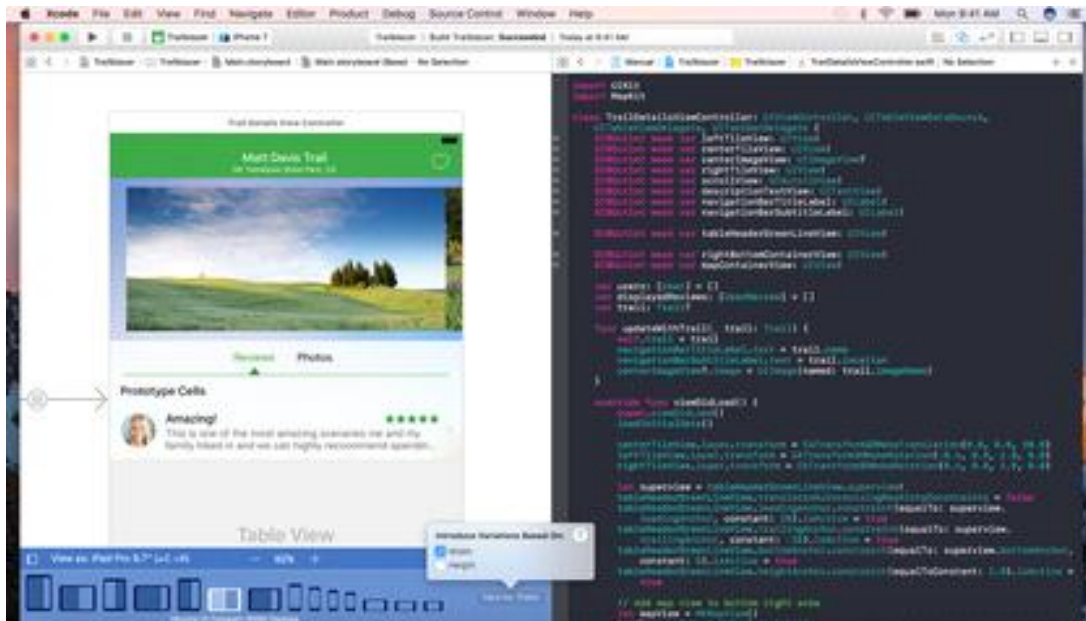
Ο χρήστης πρέπει να κατεβάσει το κατάλληλο πρόγραμμα, ανάλογα με το ΛΣ του υπολογιστή του και στη συνέχεια να ακολουθήσει τα ίδια βήματα με τον προηγούμενο τρόπο. Στο τέλος, πρέπει να δοκιμάσει αν λειτουργεί η σύνδεση. Ο έλεγχος της σύνδεσης γίνεται μέσω του περιηγητή. Ενδεικτική σύνδεση μέσω USB Οι απαιτήσεις που έχει το App Inventor είναι: Λειτουργικό σύστημα windows XP ή ανώτερο, GNU/Linux Ubuntu 8 ή ανώτερο, Debian 5 ή ανώτερο και Mac OS X 10.5 ή ανώτερο. Οι περιηγητές ιστοσελίδων που υποστηρίζονται είναι οι Firefox 3.6, Safari 5.0, Chrome 4.0 και όλες οι μεταγενέστερες εκδόσεις. Ο internet Explorer προς το παρόν δεν υποστηρίζεται. Σχετικά με τη συσκευή (τηλέφωνο ή ταμπλέτα), πρέπει να έχει ΛΣ Android 2.3 ή ανώτερο.

vii) Λειτουργικό σύστημα iOS

Το iOS (προηγουμένως iPhone OS) είναι ένα λογισμικό για κινητά το οποίο αναπτύχθηκε και διανέμεται από την Apple Inc. Αρχικά παρουσιάστηκε το 2007 για το iPhone, ενώ υποστηρίζει και άλλες συσκευές της Apple όπως το iPod touch (Σεπτέμβριος 2007), το iPad (Ιανουάριος 2010) και το Apple TV (δεύτερης γενιάς) (Σεπτέμβριος 2010). 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. Στις Ιανουαρίου 2013, το App Store της Apple περιείχε περισσότερο από 775.000 εφαρμογές iOS, 300.000 από τις οποίες ήταν συμβατές για iPad. Αυτές οι εφαρμογές έχουν μεταφορτωθεί συνολικά περισσότερα από 50 δισεκατομμύρια φορές

viii) Xcode

Το Xcode είναι ένα ολοκληρωμένο περιβάλλον ανάπτυξης για το macOS που περιέχει μια σειρά από εργαλεία ανάπτυξης λογισμικού που αναπτύχθηκαν από την Apple για την ανάπτυξη λογισμικού για macOS, iOS, watchOS και tvOS. Κυκλοφόρησε για πρώτη φορά το 2003.



Εικόνα:Περιβάλλον προγραμματισμού Xcode

Σημαντικά χαρακτηριστικά

Το Xcode υποστηρίζει τον πηγαίο κώδικα για τις γλώσσες προγραμματισμού C, C ++, Objective-C, Objective-C ++, Java, AppleScript, Python, Ruby, ResEdit και Swift με διάφορα μοντέλα προγραμματισμού, Carbon και Java.

Τρίτα μέρη έχουν προσθέσει υποστήριξη για GNU Pascal, Free Pascal, Ada, C #, Perl, και D

Χάρη στην εκτελέσιμη μορφή Mach-O, επιτρέπει σε μεγάλα δυαδικά αρχεία που περιέχουν κώδικα για πολλαπλές αρχιτεκτονικές, το Xcode μπορεί να δημιουργήσει δυαδικά αρχεία γενικής χρήσης, τα οποία επιτρέπουν το λογισμικό να τρέχει σε πλατφόρμες PowerPC και Intel (x86) -bit και κώδικα 64-bit και για τις δύο αρχιτεκτονικές. Χρησιμοποιώντας το iOS SDK, το Xcode μπορεί επίσης να χρησιμοποιηθεί για να μεταγλωττίσει και να εντοπίσει σφάλματα σε εφαρμογές για iOS που εκτελούνται σε επεξεργαστές αρχιτεκτονικής ARM.

Το Xcode περιλαμβάνει το εργαλείο GUI Instruments, το οποίο τρέχει πάνω από ένα δυναμικό πλαίσιο ανίχνευσης, το DTrace, το οποίο δημιουργήθηκε από την Sun Microsystems και κυκλοφόρησε ως μέρος του OpenSolaris.

Παλαιότερα, το Xcode υποστήριζε τη διανομή μιας διαδικασίας δημιουργίας προϊόντων σε πολλά συστήματα. Μία σχετική τεχνολογία ονομαζόταν Shared Workgroup Build, η οποία χρησιμοποίησε το πρωτόκολλο Bonjour για να ανακαλύψει αυτόματα συστήματα που παρέχουν υπηρεσίες μεταγλωττιστή και μια τροποποιημένη έκδοση του προϊόντος distcc του ελεύθερου λογισμικού για να διευκολύνει τη διανομή του φόρτου εργασίας. Οι παλαιότερες εκδόσεις του Xcode παρείχαν ένα σύστημα με το όνομα Dedicated Network Builds. Αυτές οι δυνατότητες δεν υπάρχουν στις υποστηριζόμενες εκδόσεις του Xcode.

Το Xcode περιλαμβάνει επίσης τα εργαλεία και τα πλαίσια του WebObjects της Apple για την κατασκευή εφαρμογών ιστού Java και υπηρεσιών διαδικτύου (που προηγουμένως πωλούνταν ως ξεχωριστό προϊόν).

ix)Swift.

Μια ισχυρή ανοιχτή γλώσσα που επιτρέπει σε όλους να δημιουργούν καταπληκτικές εφαρμογές.

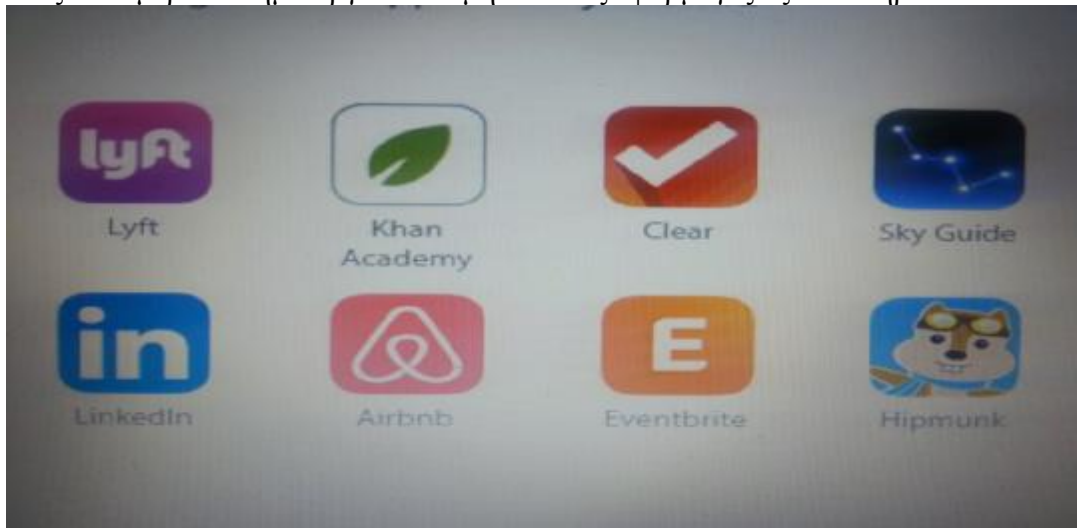
Το Swift είναι μια στιβαρή και διαισθητική γλώσσα προγραμματισμού που δημιουργήθηκε από την Apple για την κατασκευή εφαρμογών για iOS, Mac, Apple TV και Apple Watch. Έχει σχεδιαστεί για να δώσει στους προγραμματιστές περισσότερη ελευθερία από ποτέ. Το Swift είναι εύχρηστο και ανοικτό, οπότε οποιοσδήποτε με μια ιδέα μπορεί να δημιουργήσει κάτι απίστευτο.

Οι προγραμματιστές κάνουν ήδη σπουδαία πράγματα με το Swift.

Το Swift είναι μια γρήγορη και αποτελεσματική γλώσσα που παρέχει ανατροφοδότηση σε πραγματικό χρόνο και μπορεί να ενσωματωθεί άψογα στον υφιστάμενο κώδικα Objective-C code. Έτσι, οι προγραμματιστές είναι σε θέση να γράψουν ασφαλέστερο, πιο αξιόπιστο κώδικα, να εξοικονομήσουν χρόνο και να δημιουργήσουν ακόμα πιο πλούσιες εμπειρίες εφαρμογών.

Εξαιρετικές εφαρμογές που κατασκευάστηκαν με το Swift.

Όλο και περισσότεροι προγραμματιστές ενσωματώνουν τον κώδικα Swift στις εφαρμογές τους. Και μερικοί δημιουργούν ακόμη και νέες εφαρμογές εξ ολοκλήρου στο Swift.



Εικόνα: Εφαρμογές που δημιουργήθηκαν από γλώσσα προγραμματισμού swift

- Η καλύτερη απόδοση ισοδυναμεί με καλύτερες εφαρμογές.

Οι εφαρμογές Swift υπερβαίνουν το όνομα. Για παράδειγμα, ένας κοινός αλγόριθμος αναζήτησης ολοκληρώνεται πολύ πιο γρήγορα χρησιμοποιώντας το Swift.

Μέχρι και
2.6X

πιο γρήγορα από τον Objective-C

Μέχρι και
8.4X

πιο γρήγορα από την Python 2.7

10.000 ακέραιοι που βρέθηκαν σε ένα γράφημα

χρησιμοποιώντας αλγόριθμο αναζήτησης βάθους πρώτου *

- Οι εκπαιδευτικοί προσθέτουν το Swift στο πρόγραμμα σπουδών τους.

Οι προγραμματιστές δεν είναι οι μόνοι που έχουν παρατηρήσει τις δυνατότητες του Swift. Μερικά από τα κορυφαία πανεπιστήμια και ακαδημαϊκά ιδρύματα διδάσκουν το Swift στα μαθήματα προγραμματισμού υπολογιστών τους και προσφέρουν δωρεάν μαθήματα στο iTunes U.

Αυτό σημαίνει ότι η μετάβαση από την εισαγωγική κωδικοποίηση στον επαγγελματικό προγραμματισμό δεν ήταν ποτέ ευκολότερη.

χ) Τι είναι το GNU

Η ανάπτυξη του λειτουργικού συστήματος GNU ξεκίνησε από τον Richard Stallman ενώ εργάστηκε στο εργαστήριο Τεχνητής Νοημοσύνης του Ινστιτούτου Τεχνολογίας του Μασαχουσέτη (MIT). Ονομάστηκε το Έργο GNU και ανακοινώθηκε δημοσίως στις 27 Σεπτεμβρίου 1983 στους net.unix-wizards και στις net.usoft newsgroups του Richard Stallman. Η ανάπτυξη λογισμικού ξεκίνησε στις 5 Ιανουαρίου 1984, όταν ο Stallman εγκατέλειψε τη δουλειά του στο εργαστήριο, ώστε να μην μπορεί να διεκδικήσει την κυριότητα ή να παρεμποδίσει τη διανομή των συστατικών του GNU ως ελεύθερου λογισμικού. Ο Richard Stallman επέλεξε το όνομα χρησιμοποιώντας διάφορα παιχνίδια με λόγια, συμπεριλαμβανομένου του τραγουδιού The Gnu.

Ο στόχος ήταν να δημιουργηθεί ένα πλήρως ελεύθερο λειτουργικό σύστημα λογισμικού. Ο Stallman θέλησε τους χρήστες υπολογιστών να είναι ελεύθεροι να μελετήσουν τον πηγαίο κώδικα του λογισμικού που χρησιμοποιούν, να μοιραστούν λογισμικό με άλλους ανθρώπους, να τροποποιήσουν τη συμπεριφορά του λογισμικού και να δημοσιεύσουν τις δικές τους τροποποιημένες εκδόσεις του λογισμικού. Αυτή η φιλοσοφία δημοσιεύθηκε στη συνέχεια ως Μανιφέστο του GNU τον Μάρτιο του 1985.

Η εμπειρία του Richard Stallman με το ασύμβατο σύστημα χρονομεριστικής μνήμης (ITS), ένα πρώιμο λειτουργικό σύστημα γραμμένο σε γλώσσα συναρμολόγησης που καταργήθηκε λόγω διακοπής του PDP-10, η αρχιτεκτονική του υπολογιστή για την οποία γράφτηκε το ITS οδήγησε σε απόφαση ότι ήταν απαραίτητο ένα φορητό σύστημα. Έτσι, αποφασίστηκε ότι η ανάπτυξη θα ξεκινούσε με τη χρήση γλωσσών προγραμματισμού C και Lisp και ότι η GNU θα ήταν συμβατή με το Unix. Την εποχή εκείνη, το Unix ήταν ήδη ένα δημοφιλές ιδιόκτητο λειτουργικό σύστημα. Ο σχεδιασμός του Unix ήταν αρθρωτός, οπότε θα μπορούσε να ξαναπάρει το κομμάτι.

Μεγάλο μέρος του απαιτούμενου λογισμικού χρειάστηκε να γραφτεί από την αρχή, αλλά χρησιμοποιήθηκαν επίσης υπάρχοντα συμβατά συστατικά του ελεύθερου λογισμικού τρίτων όπως το σύστημα διαλογής TeX, το σύστημα παραθύρων X και το microkernel Mach που αποτελεί τη βάση του πυρήνα του GNU Mach GNU Hurd (ο επίσημος πυρήνας του GNU). Με εξαίρεση τα προαναφερθέντα συστατικά μέρη τρίτου μέρους, το μεγαλύτερο μέρος του GNU έχει γραφτεί από εθελοντές. Μερικοί στον ελεύθερο χρόνο τους, κάποιιοι που πληρώνονται από εταιρείες, εκπαιδευτικά ιδρύματα και άλλους μη κερδοσκοπικούς οργανισμούς. Τον Οκτώβριο του 1985, ο Stallman ίδρυσε το Ίδρυμα Ελεύθερου Λογισμικού (FSF). Στα τέλη της δεκαετίας του '80 και του '90, το FSF προσέλαβε τους προγραμματιστές λογισμικού για να γράψουν το λογισμικό που απαιτείται για το GNU.

Καθώς η GNU κέρδισε, οι ενδιαφερόμενες επιχειρήσεις άρχισαν να συμβάλλουν στην ανάπτυξη ή την πώληση λογισμικού GNU και τεχνικής υποστήριξης. Η πιο εξέχουσα και επιτυχημένη από αυτές ήταν η Cygnus Solutions, τώρα μέρος του Red Hat.

Το έργο GNU Linux δημιουργήθηκε για την ανάπτυξη ενός λειτουργικού συστήματος που μοιάζει με Unix και συνοδεύεται από πηγαίο κώδικα που μπορεί να αντιγραφεί, να

τροποποιηθεί και να αναδιανεμηθεί. Ο Richard Stallman ανακοίνωσε το έργο GNU Linux το 1983 και, μαζί με άλλους, ίδρυσε το Ίδρυμα Ελεύθερου Λογισμικού το 1985.

Ο επίσημος πυρήνας του GNU Project ήταν ο GNU Hurd microkernel. Ωστόσο, από το 2012, ο πυρήνας του Linux έγινε επίσημα μέρος του έργου GNU με τη μορφή Linux-libre, μια παραλλαγή του Linux με όλα τα αποκλειστικά συστατικά που έχουν αφαιρεθεί.

Με την απελευθέρωση του διανέμου του Debian GNU / HURD 2015 από τον Απρίλιο του 2015, το GNU OS παρέχει τώρα τα συστατικά στοιχεία για τη συναρμολόγηση ενός λειτουργικού συστήματος που οι χρήστες μπορούν να εγκαταστήσουν και να χρησιμοποιήσουν σε έναν υπολογιστή. Αυτό περιλαμβάνει τον πυρήνα του GNU Hurd, ο οποίος βρίσκεται σε κατάσταση προπαραγωγής. Η σελίδα κατάστασης Hurd αναφέρει ότι "μπορεί να μην είναι έτοιμη για παραγωγή, καθώς εξακολουθούν να υπάρχουν κάποια σφάλματα και ελλείποντα χαρακτηριστικά. Ωστόσο, θα πρέπει να είναι μια καλή βάση για περαιτέρω ανάπτυξη και μη κρίσιμη χρήση εφαρμογών".

Λόγω του ότι ο Hurd δεν είναι έτοιμος για παραγωγή, στην πράξη αυτά τα λειτουργικά συστήματα είναι διανομές Linux. Περιέχουν τον πυρήνα του Linux, τα στοιχεία GNU και το λογισμικό από πολλά άλλα έργα ελεύθερου λογισμικού. Κοιτάζοντας όλο τον κώδικα του προγράμματος που περιλαμβάνεται στη διανομή του Ubuntu Linux το 2011, το GNU περιλάμβανε 8% και ο πυρήνας του Linux 9%.

Άλλοι πυρήνες όπως ο πυρήνας του FreeBSD συνεργάζονται επίσης με το λογισμικό GNU για να διαμορφώσουν ένα λειτουργικό σύστημα. Το FSF υποστηρίζει ότι ο πυρήνας του Linux, όταν χρησιμοποιείται με εργαλεία και βοηθητικά προγράμματα GNU, θα πρέπει να θεωρείται παραλλαγή του GNU και προωθεί τον όρο GNU / Linux για τέτοια συστήματα (οδηγώντας στη διαμάχη για το όνομα του GNU / Linux). Το έργο GNU έχει εγκρίνει τις διανομές Linux, όπως το gNewSense, το Trisquel και το Parabola GNU / Linux-libre. Άλλες παραλλαγές του GNU που δεν χρησιμοποιούν το Hurd ως πυρήνα περιλαμβάνουν το Debian GNU / kFreeBSD και το Debian GNU / NetBSD, φέρνοντας το αρχικό σχέδιο του GNU σε έναν πυρήνα BSD.

ΚΕΦΑΛΑΙΟ 2 - ΣΥΣΤΗΜΑΤΑ ΠΛΗΡΟΦΟΡΗΣΗΣ ΣΠΟΥΔΑΣΤΩΝ

Ένα σύστημα ενημέρωσης των φοιτητών (SIS) είναι μια εφαρμογή λογισμικού για τα εκπαιδευτικά ιδρύματα ώστε να διαχειρίζονται τα δεδομένα των φοιτητών. Τα συστήματα ενημέρωσης φοιτητών (συντομογραφία ως συστήματα SIS) παρέχουν δυνατότητες για την εισαγωγή των μαθητών σε εξετάσεις και άλλα αποτελέσματα αξιολόγησης, τη δημιουργία χρονοδιαγραμμάτων φοιτητή, παρακολούθηση της συμμετοχής των φοιτητών, και διαχείριση πολλών άλλων σχολικών αναγκών σχετικά με δεδομένα που σχετίζονται με τους μαθητές (Morgan, 2003).

Το SIS είναι ισοδύναμο με ένα Enterprise Resource Planning ή ERP σύστημα για τον εταιρικό πελάτη. Ως εκ τούτου, πολλά από τα ζητήματα με την Μεθοδολογία Επιλογής Συστήματος ERP, την υλοποίηση και τη λειτουργία ενός συστήματος ERP εφαρμόζονται πάρα πολύ.

Τα συστήματα αυτά διαφέρουν ως προς το μέγεθος, την έκταση και την ικανότητα, από τα πακέτα που υλοποιούνται από σχετικά μικρές οργανώσεις για την κάλυψη των εγγραφών σπουδαστών και μόνο, έως και λύσεις για ολόκληρη την επιχείρηση που έχουν ως στόχο να καλύψουν τις περισσότερες πτυχές της λειτουργίας μεγάλων πολυεθνικών οργανισμών εκπαίδευσης με σημαντική τοπική αρμοδιότητα. Πολλά συστήματα μπορούν να προσαρμοστούν σε διαφορετικά επίπεδα λειτουργικότητας με την αγορά add-on (πρόσθετων) modules (Paulsen, 2003).

Μέχρι πρόσφατα, οι κοινές λειτουργίες του συστήματος αρχείων των φοιτητών είναι να υποστηρίζουν τη διατήρηση των προσωπικών πληροφοριών και πληροφοριών μελέτης που αφορούν (Langsdale, 2014):

Να επαναφέρουν προκαθορισμένα δεδομένα του φοιτητή

Χειρισμός αιτήσεων από τους υποψήφιους φοιτητές

Χειρισμός της διαδικασίας εισδοχής

Εγγραφή νέων φοιτητών και αποθήκευση επιλογών διδασκαλίας

Αυτόματη δημιουργία προγραμμάτων τάξεων και δασκάλων

Χειρισμός αρχείων εξετάσεων, αξιολογήσεων, βαθμών και ακαδημαϊκής εξέλιξης

Τήρηση αρχείων των απουσιών και παρουσιών

Καταγραφή των επικοινωνιών με τους μαθητές

Διατήρηση αρχείων πειθαρχίας

Παροχή στατιστικών εκθέσεων

Συντήρηση λεπτομερειών διαχείρισης φοιτητικής λέσχης και κοιτώνων

Επικοινωνία στοιχείων των φοιτητών στους γονείς μέσω μιας πύλης γονέων

Υπηρεσίες Ειδικής Αγωγής/ Εξατομικευμένου Εκπαιδευτικού Προγράμματος (ΕΕΠ)

Υπηρεσίες ανθρώπινων πόρων

Λογιστική και προϋπολογισμός των υπηρεσιών

Μητρώα υγείας Φοιτητών

Διαχείριση κυλικείου

Διαχείριση Μεταφορών

Έξοδα Διαχείρισης

Απογραφή και Περιουσιακά Στοιχεία του σχολείου

Επεξεργασία Μισθοδοσίας για το προσωπικό στο σχολείο

Σε μεγαλύτερες επιχειρηματικές λύσεις που έχουν τα δεδομένα φοιτητής στον πυρήνα τους, περαιτέρω λειτουργίες περιλαμβάνουν την διαχείριση χρηματοπιστωτικής βοήθειας στον φοιτητή και περισσότερες μπορούν να προσαρμοστούν από τον κύριο του έργου. Όπου υπάρχουν εθνικά ή κρατικά συστήματα για τη χρηματοδότηση των σπουδών ή για στατιστικούς σκοπούς, το σύστημα αρχείων των φοιτητών συχνά παρέχει λειτουργικότητα που φροντίζει για αυτό, μέσω των ενοτήτων ή των βασικών στοιχείων που χειρίζονται την παραγωγή των απαιτούμενων αρχείων, ή ασχολούνται με την διαμορφωμένη μεταφορά πληροφοριών. Παραδείγματα είναι η διαδικασία FAFSA (δωρεάν εφαρμογή για την ομοσπονδιακή ενίσχυση σπουδαστών) στις Ηνωμένες Πολιτείες, οι διαδικασίες Εταιρείας Φοιτητικών Δανείων του Ηνωμένου Βασιλείου (επεξεργασία αρχείων SSAR, SSAC και ATFEE), το UCAS (Υπηρεσία Εισόδων σε Πανεπιστήμια και Κολέγια) στο Ηνωμένο Βασίλειο, ή οι στατιστικές καταγραφές φοιτητών EXAE και HESES στο Ηνωμένο Βασίλειο (Ellis & Calvo, 2007).

Στο παρελθόν, τα πανεπιστήμια και οι μεγάλες σχολικές περιοχές, ιδίως έχουν δημιουργήσει τα δικά τους συστήματα καταγραφής των φοιτητών. Ένα τέτοιο παράδειγμα είναι το σύστημα Αποθετηρίου Πληροφοριών των σπουδαστών (ROSI) στο Πανεπιστήμιο του Τορόντο. Με την αυξανόμενη πολυπλοκότητα στην επιχείρηση των εκπαιδευτικών ιδρυμάτων, οι περισσότεροι οργανισμοί επιλέγουν πλέον να αγοράσουν προσαρμόσιμο λογισμικό, και ολοένα και περισσότεροι αγοράζουν το λογισμικό ως υπηρεσία (SaaS). Τα περισσότερα συστήματα ενημέρωσης των φοιτητών σε χρήση σήμερα είναι server-based, με την εφαρμογή να φιλοξενείται σε ένα κεντρικό server υπολογιστών, και την πρόσβαση από τις εφαρμογές πελάτη σε διάφορα σημεία εντός και εκτός από το σχολείο. Αλλά τα συστήματα ενημέρωσης των φοιτητών κινούνται στο διαδίκτυο από τα τέλη της δεκαετίας του 1990 και η τάση αυτή επιταχύνεται, καθώς τα ιδρύματα αντικαθιστούν τα παλαιότερα συστήματα (Rankin, 2013).

Φυσικά, καθώς όλο και περισσότερα κολέγια, πανεπιστήμια και σχολεία δείχνουν ενδιαφέρον για τις cloud-based επιλογές του SIS, το θέμα της ασφάλειας των πληροφοριών έχει μετατραπεί σε μια σημαντική ανησυχία. Σε κάθε περίπτωση, τα πανεπιστήμια των ΗΠΑ φέρουν μεγαλύτερο κίνδυνο για παραβιάσεις της ασφάλειας σε σχέση με εταιρείες λιανικής πώλησης ή παρόχους υγειονομικής περίθαλψης. Οι προμηθευτές λογισμικού εκπαιδευτικής τεχνολογίας θα πρέπει να είναι σε πλήρη επίγνωση των βέλτιστων πρακτικών για τη

διασφάλιση της ενημέρωσης των φοιτητών (συμπεριλαμβανομένων των πληροφοριών για την κοινωνική ασφάλιση, των οικονομικών πληροφοριών, κλπ) και των θεσμικών οργάνων που έχουν την ευθύνη να αναζητήσουν προμηθευτές που εφαρμόζουν τα καλύτερα στην κατηγορία τους μέτρα ασφαλείας (Rankin, 2013).



Εικόνα 5. Τυπική Εικόνα Συστήματος Πληροφοριών Σπουδαστών

Σε αντίθεση με μια αναβάθμιση σε ένα πρόγραμμα περιήγησης στο Web ή έναν επεξεργαστή κειμένου, αλλαγές και αναβαθμίσεις σε αυτά τα συστήματα τείνουν να έχουν σημαντικές επιπτώσεις στις λειτουργίες των εργαζομένων του κάθε σχολείου κάθε ημέρα. Τα συστήματα αυτά συνήθως αγγίζουν κάθε πτυχή της λειτουργίας του σχολείου, ακόμη και όταν μόνο οι βασικές μονάδες χρησιμοποιούνται. Για τους λόγους αυτούς, θα πρέπει να ληφθεί μέριμνα ώστε να εξεταστεί ο αντίκτυπος στα ακόλουθα στοιχεία (Seale, 2009):

Ροή εργασίας: Δεδομένου ότι τα προγράμματα αυτά είναι στενά δεμένα με την επιχειρηματική ροή εργασίας και τις διαδικασίες ενός σχολείου, μια αλλαγή σε ένα σύστημα SIS μπορεί να επιφέρει αλλαγές στη ροή εργασιών. Αυτό μπορεί να έχει σημαντικές επιπτώσεις στην καθημερινή λειτουργία αν δεν εξεταστεί προσεκτικά πριν από την εφαρμογή.

Μετατροπή των δεδομένων: Η μετατροπή των δεδομένων των ιστορικών δεδομένων (μεταγραφές, συμμετοχή, ιατρικά αρχεία, κ.λπ.) τόσο για τους τρέχοντες όσο και για τους παλαιότερους φοιτητές μπορεί επίσης να είναι ένα σημαντικό ζήτημα για τη μετάβαση σε ένα νέο SIS. Δεδομένου ότι τα περισσότερα σχολεία υποχρεούνται να τηρούν τα ιστορικά στοιχεία για το παρελθόν των φοιτητών, θα πρέπει να δοθεί προσοχή και στο ποιες πληροφορίες θα μετατραπούν και τι θα πρέπει να αρχειοθετηθεί.

Προσαρμοσμένες Αναφορές: Δεδομένου ότι υπάρχει μικρή τυποποίηση στο τι και το πώς αποθηκεύονται οι πληροφορίες του φοιτητή, τα περισσότερα σχολεία έχουν τις δικές τους μεθόδους και διαδικασίες (π.χ. μορφοποίηση και διάταξη των εκθέσεων δεδομένων) για εκτυπώσεις βαθμών φοιτητή ή παρουσιολογίου. Δεδομένου ότι τα περισσότερα SIS δεν είναι απόλυτα συμβατά με το προηγούμενο SIS, η αναβάθμιση μπορεί να είναι μια μακρά και επίπονη διαδικασία.

Εκπαίδευση: Μερικά νέα προγράμματα SIS έχουν την τάση να περιλαμβάνουν μερικά περιττά χαρακτηριστικά, και έτσι η εκπαίδευση των εργαζομένων στην χρήση του νέου προγράμματος SIS πιθανότατα να είναι μια δαπανηρή και χρονοβόρα διαδικασία.

Όπως και με ένα σύστημα ERP, τα σχολεία θα πρέπει να εξετάσουν τις διεργασίες που παρόμοιες με την μεθοδολογία επιλογής ERP συστήματος κατά την επιλογή ενός συστήματος SIS (Seale, 2009).

Σε μια μελέτη στην οποία συμμετείχαν περιοχές γνωστές για την έντονη χρήση των δεδομένων, το 48% των εκπαιδευτικών είχαν δυσκολία θέτοντας ερωτήματα που προέκυπταν από τα δεδομένα, το 36% δεν είχε κατανοήσει τα δεδομένα στοιχεία, και το 52% ερμήνευσε εσφαλμένα τα δεδομένα. Για να καταπολεμηθεί αυτό, κάποια συστήματα SIS παρουσιάζουν τα δεδομένα να εκπαιδευτικούς με μια πιο κατανοητή μορφή (ενσωμάτωση ετικετών, συμπληρωματική τεκμηρίωση και βοήθεια του συστήματος και λήψη σημαντικών αποφάσεων πακέτου / οθόνης και περιεχομένου) ώστε να βελτιώσουν την ανάλυση των δεδομένων από τους εκπαιδευτικούς κατά τη χρήση ενός SIS (Seale, 2009).

ΚΕΦΑΛΑΙΟ 3-ΒΗΜΑΤΙΚΗ ΕΠΕΞΗΓΗΣΗ ΕΥΡΕΣΗ ΙΔΕΑΣ ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΥΛΟΠΟΙΗΣΗΣ ΜΕ PROTOTYPE

Σαν ομάδα χρειάστηκαν ώρες συζητήσεων μέχρι να βρεθεί η ιδέα και να προχωρήσουμε στην υλοποίηση της.

Για να μπορέσουμε όμως να φτάσουμε στο τελικό αποτέλεσμα που θα σας παρουσιάσουμε στην συνέχεια δηλαδή στον σχεδιασμό και την υλοποίηση μιας εκπαιδευτικής εφαρμογής για κινητές συσκευές πλατφόρμας Ios χρειάστηκαν κάποιες γνώσεις που αποκτήθηκαν σταδιακά και κάποιες ενέργειες που ήταν αναγκαίες να γίνουν.

Για την καλύτερη κατανόηση της ιδέας και τον τρόπο που την υλοποιήσαμε θα την αναλύσουμε σε βήματα.

Αρχικά πολλές ιδέες εμφανίστηκαν και συζητήθηκαν αλλά δεν ήταν ικανές να μας πείσουν για την επιτυχία τους. Γι αυτό τον λόγο έπρεπε να θέσουμε κάποιες ερωτήσεις και να δώσουμε τις κατάλληλες απαντήσεις που θα μας εμφάνιζαν την σωστή και ιδανική κατεύθυνση ώστε να συνεχίσουμε προς την υλοποίηση.

Αρχικά θα έπρεπε να ορίσουμε τους στόχους της ομάδας, έπρεπε να τους ορίσουμε όσο το δυνατόν πιο συγκεκριμένα και στην συνέχεια να βρούμε πως θα πραγματοποιήσουμε αυτούς τους στόχους.

ι) ΑΝΑΚΑΛΥΨΗ ΤΗΣ ΙΔΕΑΣ

Σαν φοιτητές έχουμε περάσει πολλές ώρες από την φοιτητική μας ζωή μαζί με άλλους φοιτητές και ξέρουμε τα προβλήματα που έχουν πέρα από τις δυσκολίες φοίτησης, ένα κοινό πρόβλημα που αντιμετωπίζουν σχεδόν όλοι οι φοιτητές είναι το πρόβλημα της διατροφής.

Μια σύντομη έρευνα μας αποκάλυψε μερικά δεδομένα που ίσχυαν και σε μας ατομικά χωρίς να γνωρίζουμε ότι αυτά τα δεδομένα αποτελούν κοινό γνώρισμα μεταξύ πολλών συμφοιτητών μας.

Από όλους τους φοιτητές που ρωτήσαμε η πλειοψηφία είχε βασιστεί για την διατροφή της σε πρόχειρο φαγητό. Αυτό συνέβαινε είτε για ευκολία είτε για μικρότερο κόστος είτε εξαιτίας της έλλειψης γνώσης πάνω στην μαγειρική.

Μεγάλο μέρος την πλειοψηφίας είχε πρόσβαση σε σπιτικό μαγειρεμένο φαγητό που όμως τις περισσότερες δεν καταναλωνόταν είτε γιατί το θεωρούσαν μαγαιάτικο όπως ανέφεραν είτε γιατί το ξεχνούσαν και αλλοιωνόταν στο ψυγείο τους.

Επίσης παρατηρήσαμε ότι όλο και πιο πολλοί φοιτητές παρόλο που δεν ήταν ευχαριστημένοι από τις γεύσεις και την ποιότητα πολλές φορές, επέλεγαν ως λύση την σίτιση που πρόσφερε το Τ.Ε.Ι ή Α.Ε.Ι ανάλογα τον φοιτητή που ρωτούσαμε.

Σε μια άλλη ερώτηση που θέσαμε στους φοιτητές που ρωτήσαμε δηλαδή γιατί δεν μαγειρεύουν μόνοι τους το φαγητό τους σχεδόν όλοι απάντησαν ότι δεν ξέρουν να μαγειρεύουν καθόλου ή ήξεραν ελάχιστα και αυτό περιελάμβανε απλές συνταγές (π.χ μακαρόνια, αυγά και τηγανιτά κυρίως φαγητά) και όσοι προσπάθησαν να κάνουν κάτι παραπάνω απέτυχαν γιατί δεν ήξεραν πως να τα καταφέρουν και τα παράτησαν.

Ένα άλλο θέμα που μας απασχολούσε ήταν το ημερήσιο κόστος διατροφής και το πόσο υγιεινό και θρεπτικό ήταν το κάθε γεύμα.

Κατά μέσο όρο λοιπόν κάθε μέρα ο φοιτητής ξοδεύει για να τραφεί για κυρίως γεύματα (μεσημεριανό και βραδινό) από οχτώ (8) έως είκοσι (20) ευρώ καθημερινά και πολλές φορές μείωνε είτε την ποσότητα είτε την ποιότητα των γευμάτων του ή πολλές φορές αφαιρούσε ένα κυρίως γεύμα λόγω χρημάτων.

Όλοι σαν φοιτητές είχαμε και έχουμε περιορισμένο χρόνο που περνάμε στο φοιτητικό μας σπίτι αλλά διαπιστώσαμε ότι σε αυτό τον λίγο χρόνο ένα μεγάλο μέρος του 30-45 λεπτά τουλάχιστον σχεδόν κάθε φορά τον σπαταλάμε περιμένοντας το γρήγορο φαγητό που έχουμε παραγγείλει.

Ακόμα ένα κοινό γνώρισμα σχεδόν όλων των φοιτητών ήταν ότι είχαν στην κατοχή τους σε ντουλάπια ψυγεία και λοιπούς αποθηκευτικούς χώρους αρκετά μεγάλη ποικιλία από πρώτες ύλες (λάδι, μπαχαρικά, λαχανικά, κρεατικά κλπ) που θα μπορούσαν να χρησιμοποιήσουν και να αξιοποιήσουν στην παρασκευή του γεύματος τους όμως μένανε αχρησιμοποίητες. και αυτές, βεβαία δεν είχαν επιβαρύνει τον φοιτητή στην αγορά τους διότι κατά κύριο λόγο ήταν 'προσφορά' από τις οικογένειες τους με σκοπό την χρήση τους.

Ακόμα σε όλα τα φοιτητικά σπίτια υπήρχαν μαγειρικά σκεύη που έμεναν αχρησιμοποιήτα ή ήταν ελαφρώς χρησιμοποιημένα.

Όλες αυτές οι λεπτομέρειες μας αποκάλυψαν την ιδέα. θα πρέπει δηλαδή να φτιάξουμε μια εκπαιδευτική εφαρμογή που ο φοιτητής θα μπορούσε να την χρησιμοποιήσει ώστε να τον βοηθήσει να μαγειρέψει με τα υλικά που κατέχει, με τα σκεύη που διαθέτει ένα γεύμα με την βοήθεια συνταγών που θα του εμφανίζονταν σύμφωνα με τα υλικά που διαθέτει.

ii) ΠΡΟΣΔΙΟΡΙΣΜΟΣ ΤΗΣ ΙΔΕΑΣ

Τώρα αφού είχαμε ανακαλύψει την ιδέα, θα έπρεπε να βρούμε πως θα την προσδιορίσουμε καλύτερα για να προχωρήσουμε στο επόμενο βήμα.

Θα έπρεπε να κατανοήσουμε όλες αυτές τις πληροφορίες που είχαμε συλλέξει και να τις αξιοποιήσουμε με τέτοιο τρόπο ώστε να μας φέρει μια επιτυχημένη εφαρμογή.

Έπρεπε να φτιάξουμε μια εκπαιδευτική εφαρμογή που να εκπαιδεύει του φοιτητές και γενικά όποιον την χρησιμοποιεί και να τους εξοικειώνει με την μαγειρική και να τους προσφέρει λύσεις.Οπότε έπρεπε να συγκεντρώσουμε τα δεδομένα για να μπορέσουμε να τα επεξεργαστούμε καλύτερα.

Δηλαδή, τι ήταν αυτό που μας οδήγησε σε αυτή την ιδέα;

Αρχικά ότι οι περισσότεροι φοιτητές ξοδεύουν χρήματα σε πρόχειρο φαγητό.

Ότι συνήθως υπάρχει η εντύπωση ότι το γρήγορο φαγητό σου γλιτώνει χρόνο ενώ τις περισσότερες φορές αυτό δεν ισχύει.

Ότι η διατροφή δεν είναι σωστή πράγμα που σημαίνει ότι η διατροφική αξία των γευμάτων είναι χαμηλή.

Ότι μένουν πάρα πολλές ή πετιούνται πρώτες ύλες που θα μπορούσαν να χρησιμοποιηθούν.

Και τέλος ότι πολλοί φοιτητές δεν γνωρίζουν καθόλου ή γνωρίζουν ελάχιστα να μαγειρεύουν.

Με αυτά τα δεδομένα θα έπρεπε να σχεδιάσουμε μια τέτοια εφαρμογή που θα έχει και εκπαιδευτικό χαρακτήρα.

Σχεδιάσαμε κάποια πρότυπα και τα μοιράσαμε σε φοιτητές και τις οικογένειές τους που ήταν στο κοντινό μας οικογενειακό και φιλικό περιβάλλον.

Κάποιες ενδεικτικές ερωτήσεις ήταν:

- Πείτε μας ένα φαγητό που μαγειρέψατε την προηγούμενη εβδομάδα;
- Ποια από αυτά τα υλικά βρίσκονταν στο ψυγείο/ντουλάπια σας;
- Αν βρείτε στο Internet 2-3 συνταγές με τα υλικά που έχετε στο ψυγείο σας, ποιοι παράγοντες επηρεάζουν συνήθως περισσότερο την επιλογή σας;

Οι βασικοί μας στόχοι ήταν:

1. Εκμάθηση του μη εξοικειωμένου χρήστη στην μαγειρική
2. Μηδενικό ή χαμηλό κόστος για τον χρήστη στην επιλογή υλικών
3. Εξοικονόμηση χρόνου για τον χρήστη
4. Επίλυση καθημερινών αναγκών

Άρα το project ήταν η επιλογή από διαθέσιμα υλικά που είχαμε μέσα στο ψυγείο, ντουλάπια κλπ εξοικονομώντας χρήματα.

Εύρεση συνταγών γρήγορα εξοικονομώντας χρόνο.

Εμφάνιση συνταγής για να την πραγματοποιήσει ο χρήστης.

Τα σχόλια των φοιτητών και των οικογενειών τους μας δώσανε μια άλλη οπτική που και μας βοήθησε να κατανοήσουμε τις ανάγκες τους και να καλυτερεύσουμε την αρχική μας ιδέα και να την βελτιώσουμε σε τέτοιο βαθμό που η εφαρμογή δεν περιοριζόταν πλέον στον φοιτητή-χρήστη.

Με την ανάλυση των απαντήσεων του δείγματος, καταλήξαμε στις βασικές ανάγκες των χρηστών που θα εξυπηρετεί η εφαρμογή μας. Οι ανάγκες αυτές μετατράπηκαν σε συγκεκριμένες λειτουργικότητες.

Ενδεικτικά:

- Εύρεση συνταγής με τα πράγματα που έχω στο ψυγείο/ντουλάπια μου

- Λίστα με κατηγορίες υλικών
- Τις περισσότερες φορές έχω λίγο χρόνο διαθέσιμο
 - Φίλτρα επιλογής χρονικού διαστήματος
- Θέλω να φτιάξω τραπέζι για 5 άτομα
 - Επιλογή αριθμού μερίδας
- Θέλω να συγκεντρώνω εύκολα και γρήγορα τα υλικά που επιλέγω και να μπορώ να τα ελέγχω
 - POT (κατσαρόλα)
- Θέλω να συγκεντρώνω εύκολα και γρήγορα κάπου τα υλικά που μπορεί να λείπουν από μια συνταγή
 - Λίστα για ψώνια

Όλες οι απαντήσεις που λάβαμε χρησιμοποιήθηκαν από την ομάδα μας και σχεδιάσαμε μια εφαρμογή με πολλές δυνατότητες

1. Εξοικείωση του χρηστή με την μαγειρική
2. Εύκολη και γρήγορη επιλογή διαθέσιμων υλικών
3. Δυνατότητα επιλογής μη διαθέσιμων υλικών
4. Επιλογή συνταγής από μια τεραστία γκάμα συνταγών βάση επιθυμίας του χρηστή
5. Λεπτομερείς παρουσίαση βημάτων εκτέλεσης συνταγής
6. Απευθύνεται και σε άτομα με ειδικό πρόγραμμα διατροφής (gluten free, vegetarian, vegan) και χρήσιμα χαρακτηριστικά και τον τρόπο λειτουργίας τους που θα σας παρουσιάσουμε στο 4^ο κεφάλαιο.

iii) ΣΧΕΔΙΑΣΜΟΣ

Στη φάση του σχεδιασμού, οι λειτουργικότητες που καταγράφηκαν στην προηγούμενη φάση παίρνουν σάρκα και οστά, θα πρέπει να βρούμε τον τρόπο ώστε να τις αποτυπώσουμε όσο το δυνατόν καλύτερα για να ανταποκρίνονται όσο πιο κοντά στην ιδέα μας και στην ανάλυση που είχαμε κάνει.

Καταρχάς θα έπρεπε να βρούμε την βάση για τον σχεδιασμό αυτής της εφαρμογής που είχαμε στο μυαλό μας και είχαμε αποτυπώσει προφορικά και γραπτά σε στόχους και σε λειτουργικότητες.

Η εφαρμογή μας θα ήταν σχεδιασμένη για κινητές πλατφόρμες συσκευών Ios. Αυτό σημαίνει ότι ήταν αναγκαίο να γνωρίσουμε καλύτερα το συγκεκριμένο λογισμικό, τα χαρακτηριστικά του, τις ιδιαιτερότητές του και γενικά τους κανόνες που υπάρχουν για να μπορέσουμε να ξεκινήσουμε τον σχεδιασμό.

Βασική προϋπόθεση σαν σχεδιαστές ήταν να βρούμε και να μελετήσουμε τις οδηγίες που δίνονται από την apple σχετικά με την αλληλεπίδραση του χρήστη-ανθρώπου πάνω σε αυτό που θέλουμε να σχεδιάσουμε και ποιοί κανόνες υπάρχουν. Αρχικά έπρεπε να επιλέξουμε σε τι συσκευή θα ήταν σχεδιασμένη η εφαρμογή μας για να μπορέσουμε να μελετήσουμε συγκεκριμένες οδηγίες και κανόνες. Αποφασίσαμε να επιλέξουμε σαν πρότυπο το iPhone μιας και οι διαστάσεις του εμφανίζονται σε πολλά iPhone και αυτές τις διαστάσεις τις χρησιμοποιούν οι περισσότεροι χρήστες.

iv) Human Interface Guidelines Ios

Οι οδηγίες ανθρώπινης διεπαφής (HIG) είναι έγγραφα ανάπτυξης λογισμικού που παρέχουν στους προγραμματιστές εφαρμογών ένα σύνολο συστάσεων. Σκοπός τους είναι να βελτιώσουν την εμπειρία για τους χρήστες, καθιστώντας τις διεπαφές εφαρμογών πιο διαισθητικές, κατανοητές και συνεπείς. Οι περισσότεροι οδηγοί περιορίζονται στον ορισμό μιας κοινής εμφάνισης και αίσθησης για εφαρμογές σε ένα συγκεκριμένο περιβάλλον εργασίας. Οι οδηγοί απαριθμούν συγκεκριμένες πολιτικές. Οι πολιτικές βασίζονται ενίοτε σε μελέτες αλληλεπίδρασης ανθρώπου-υπολογιστή (αποκαλούμενες μελέτες χρηστικότητας), αλλά οι περισσότερες βασίζονται σε αυθαίρετες συμβάσεις που έχουν επιλεγεί από τους προγραμματιστές της πλατφόρμας.

Ο κεντρικός στόχος ενός HIG είναι να δημιουργήσει μια συνεπή εμπειρία σε όλο το περιβάλλον (γενικά ένα λειτουργικό σύστημα ή περιβάλλον εργασίας), συμπεριλαμβανομένων των εφαρμογών και άλλων εργαλείων που χρησιμοποιούνται. Αυτό σημαίνει τόσο την εφαρμογή του ίδιου οπτικού σχεδιασμού όσο και τη συνεπή πρόσβαση και συμπεριφορά των κοινών στοιχείων της διασύνδεσης - από απλά όπως κουμπιά και εικονίδια μέχρι πιο περίπλοκες κατασκευές, όπως κουτιά διαλόγου.

HIGs είναι συστάσεις και συμβουλές που στοχεύουν να βοηθήσουν τους σχεδιαστές και σκοπίμως επιλέγουν να τις σπάσουν αν πιστεύουν ότι οι οδηγίες δεν ταιριάζουν στην αίτησή τους ή οι δοκιμές ευχρηστίας αποκαλύπτουν ένα πλεονέκτημα. Αλλά με τη σειρά του, ο οργανισμός που δημοσίευσε το HIG ενδέχεται να αρνηθεί την επικύρωση της αίτησης. Η διεπαφή χρήστη του Mozilla Firefox, για παράδειγμα, έρχεται σε αντίθεση με το HIG του GNOME του έργου, το οποίο είναι ένα από τα κύρια επιχειρήματα για την ενσωμάτωση των Eriphany αντί του Firefox στη διανομή του GNOME

Εμείς σαν ομάδα που ασχολείται με την δημιουργία εφαρμογής για κινητές πλατφόρμες iOS έπρεπε να μελετήσουμε τις HIG (Human Interface Guidelines) της Apple.

Βασικές κατηγορίες :

- Overview

- App Architecture
- User Interaction
- System Capabilities
- Visual Design
- Icons and Images
- Bars
- Views
- Controls
- Extensions
- Technologies
- Resources

Εμείς όμως σαν σχεδιαστές της εφαρμογής δώσαμε περισσότερο βάση σε κάποιες από αυτές τις κατηγορίες, ασχοληθήκαμε περισσότερο με το μέγεθος που θα επιλέγαμε για την κινητή συσκευή που θα χρησιμοποιούσαμε, για τα κουμπιά και τις μπάρες αναζήτησης, τις μπάρες εργαλείων, τα pixels, τα εικονίδια, τις εικόνες, τα κενά μεταξύ οθόνης και τηλεφώνου και λοιπά.

Κάθε φορά που τελειώναμε μια οθόνη με την επιλογή mirror του sketch ελέγχαμε αν ήταν όλες οι ενέργειες σωστές και αν είχε την μορφή που θέλαμε. ασχοληθήκαμε με το χρώμα το όνομα και την μορφή της εφαρμογής. Όλη αυτή η μελέτη μας βοήθησε να σχεδιάσουμε τις οθόνες σωστά για το κινητό που είχαμε επιλέξει (iPhone 6)

v)10 κανόνες ευχρηστίας (10 usability Heuristics for user interface design by Nilsen Norman)

Το επόμενο στάδιο ήταν να μελετήσουμε τους κανόνες ευχρηστίας, αποτελούνται από δέκα (10) βασικές αρχές (10 Usability Heuristics for User Interface Design) και ονομάζονται "Heuristics" από την ελληνική λέξη εφευρίσκω και αυτό γιατί σχετίζεται με την διαδικασία απόκτησης γνώσης ή κάποιου επιθυμητού αποτελέσματος από έξυπνες ικανότητες προτιμώντας αυτό παρά να ακολουθηθούν κάποιοι προκαθορισμένοι τύποι.

Ο χρήστης θα πρέπει να γνωρίζει τι συμβαίνει στο εσωτερικό του συστήματος, πρέπει δηλαδή να του δώσουμε μια ανατροφοδότηση της δράσης του μέσα σε εύλογο χρονικό διάστημα να του "δείξουμε" δηλαδή ότι κάτι συμβαίνει μετά από μια ενεργεία του. Αυτή η ανατροφοδότηση συνδέεται κανονικά με σημεία δράσης και μπορεί να παρασχεθεί χρησιμοποιώντας μια αλλαγή χρώματος, φορτωτή, γραφικά αριστερά ώρας κ.λπ.

Π.χ η εμφάνιση μπάρας γεμίσματος και ο χρόνος και το ποσοστό όταν κάνει upload στο facebook όπως και ο ήχος ολοκλήρωσης όταν ολοκληρωθεί το ανέβασμα.

Ταίριασμα του συστήματος με τον πραγματικό κόσμο, έπρεπε να αναρωτηθούμε αν υπάρχει κάτι στην εφαρμογή που να μην μπορεί να καταλάβει ο χρήστης γι αυτό θα πρέπει η εφαρμογή να ‘μιλάει’ την γλώσσα του χρήστη.

Π.χ ενώ θα μπορούσε να εμφανίζει ένα απλό κουμπί sing in σε μια σελίδα διδασκαλίας μαθημάτων από απόσταση εμφανίζει ένα κουμπί που αναγράφει ‘‘ναι, επιθυμώ να εγγραφώ στα μαθήματα’’

Θα πρέπει ο χρήστης να έχει έλεγχο και ελευθερία. να προσφέρει την παροχή στον χρήστη της ελευθερίας να πλοηγείται και να εκτελεί ενέργειες και να τις αναιρεί αν επιθυμεί. Αυτή την αρχή του ελέγχου και της ελευθερίας μπορούμε να την κατανοήσουμε καλύτερα με το flash μήνυμα του Gmail με αναίρεση δράσης όταν διαγράψουμε τυχαία ένα μήνυμα ηλεκτρονικού ταχυδρομείου.

Θα πρέπει να υπάρχει συνέπεια και πρότυπα, η συνέπεια είναι το κλειδί. Ένα κουμπί υποβολής σε μια σελίδα θα πρέπει να φαίνεται το ίδιο σε ολόκληρη την ιστοσελίδα σε οποιαδήποτε σελίδα. Εάν παρουσιάσουμε τα δεδομένα σε μια συγκεκριμένη μορφή πίνακα σε μία σελίδα, θα πρέπει να φαίνεται η ίδια την επόμενη φορά που τα δεδομένα εμφανίζονται σε μορφή πίνακα. Εάν η κεφαλίδα εμφανίζεται με συγκεκριμένο τρόπο στις δημόσιες σελίδες, θα πρέπει να είναι η ίδια όταν συνδέεται.

Πρέπει να υπάρχει πρόληψη σφάλματος ώστε να προλαβαίνει η εφαρμογή τυχόν λάθη του χρήστη της.

Π.χ όταν μας ζητάει να εισάγουμε έναν κωδικό ασφαλείας και όσο πληκτρολογούμε μας εμφανίζει αν είναι αδύναμος, δυνατός κλπ.

Μια εξίσου σημαντική αρχή είναι και η αναγνώριση και όχι ανάκληση, δηλαδή είναι πάντα καλύτερο να προτείνουμε στο χρήστη ένα σύνολο επιλογών παρά να τον αφήσει να θυμηθεί και να πληκτρολογήσει ολόκληρο αυτό που θέλει. ο στόχος είναι να ελαχιστοποιηθεί η εφαρμογή της μνήμης χρήστη.

Πχ. Στο google η εμφάνιση προτάσεων για το τί θέλει να αναζητήσει ο χρήστης

Η διεπαφή πρέπει να είναι ευέλικτη και να μεταμορφώνεται μεταξύ ενός αρχάριου χρήστη και ενός προχωρημένου χρήστη

Π.χ όταν γίνεται ενημέρωση ενός λογισμικού υπάρχει η επιλογή που ρωτά αν ο χρήστης θέλει να προχωρήσει με προεπιλεγμένη εγκατάσταση ή προσαρμοσμένη εγκατάσταση. Ένας προηγμένος χρήστης επιλέγει μια προσαρμοσμένη εγκατάσταση για να κόψει τις περιττές υπηρεσίες. αυτό ικανοποιεί την αρχή της ευελιξίας και αποδοτικότητας χρήσης

Εξίσου σημαντική αρχή είναι και ο αισθητικός και μινιμαλιστικός σχεδιασμός, Για τον σχεδιαστή ή τον προγραμματιστή, όλες οι πληροφορίες που παρουσιάζονται στη σελίδα είναι

σχετικές. Ο διαχειριστής προϊόντος (δηλαδή εμείς) πρέπει να ρωτήσει τον τελικό χρήστη είναι απαραίτητες και χρήσιμες όλες οι πληροφορίες που εμφανίζονται στη διεπαφή. Οι διεπαφές πρέπει να απαλειφθούν από περιττά στοιχεία και περιεχόμενο που δεν υποστηρίζουν τους στόχους και τις εργασίες της σελίδας

Π.χ Η Apple παρέχει μόνο τις βασικές πληροφορίες για τη δυνατότητα απόκρυψης πρόσθετων πληροφοριών κάτω από το στοιχείο "Μάθετε περισσότερα"

Ο χρήστης της εφαρμογής μας όπως και σε κάθε εφαρμογή κάνει αρκετές φορές ακούσια σφάλματα γι αυτό και πρέπει να ελέγχουμε αν τα λάθη αυτά εξηγούνται στον χρήστη σε κατανοητή γλώσσα.

Π.χ ενημέρωση του χρήστη ότι το username που προσπαθεί να χρησιμοποιήσει δεν είναι διαθέσιμο ή ο κωδικός που έχει εισάγει είναι λανθασμένος (εμφανίζει την επιλογή στον χρήστη να επανακτήσει τον κωδικό του αν τον έχει ξεχάσει)

Έτσι ικανοποιείτε η αρχή που αναφέρει ότι πρέπει να βοηθάμε τον χρήστη να αναγνωρίζει, να συμπεραίνει και να διορθώνει τα λάθη του.

Τελευταία αρχή αλλά εξίσου σημαντική είναι και αυτή της βοήθειας και τεκμηρίωση (documentation). Αν και θα ήταν καλύτερα αν το σύστημα μπορούσε να χρησιμοποιείτε χωρίς set από έγγραφα, είναι απαραίτητο κάθε τέτοια πληροφορία να πρέπει να είναι εύκολα προσβάσιμη, εστιασμένη στις ανάγκες του χρήστη να είναι προσεκτικά σχεδιασμένη για να βγάζει νόημα και όχι μεγάλη σε μέγεθος.

Το συμπέρασμα που οδηγηθήκαμε από την μελέτη των δέκα (10) αυτών αρχών είναι ότι αυτές οι οδηγίες είναι γενικοί κανόνες και θα ισχύουν ως επί το πλείστον για οποιαδήποτε εφαρμογή Ιστού & Κινητού με ορισμένες εξαιρέσεις.

Και πρέπει πάντα να χρησιμοποιούνται κατά την κρίση μας για την εφαρμογή αυτών των αρχών ή οποιωνδήποτε άλλων πρακτικών

vi) ΜΕΛΕΤΗ ΣΧΕΔΙΑΣΤΩΝ ΚΑΙ ΕΥΡΕΣΗ ΕΙΚΟΝΩΝ

Μετά από όλη αυτή την μελέτη και την διατριβή πάνω στις οδηγίες και τις αρχές που ισχύουν για την σωστή σχεδίαση μιας εφαρμογής τώρα θα έπρεπε να μελετήσουμε άλλους σχεδιαστές (disigners) και να εμπνευστούμε από την δουλειά τους για να πραγματοποιήσουμε την δικιά μας.

Κάποιοι disigners που θεωρήσαμε ότι μπορούν να μας βοηθήσουν και βρήκαμε δουλειές τους στο ίντερνετ ήταν οι Behance, Dribbble, Awwwards. Αυτοί οι disigners μας βοήθησαν να αποφασίσουμε και να πάρουμε ιδέες για σχεδιασμό της αλληλεπίδρασης των οθονών που είχαμε σκοπό να σχεδιάσουμε αναζήτηση εικόνων σχετικών με μαγειρική και φαγητών. Οι εικόνες που έχουμε χρησιμοποιήσει είναι από το διαδίκτυο και κύρια πηγή έμπνευσής μας ήταν του Jamie Oliver Recipes και kitchen storie και δεν επιτρέπεται η χρήση του χωρίς την άδεια τους αλλά μπορεί όποιος θέλει να μπει στο shutterstock και να αγοράσει φωτογραφίες από διάφορους σχεδιαστές.

vii) ΔΟΜΗ ΚΑΙ USER FLOW ΕΦΑΡΜΟΓΗΣ

Φτάνοντας προς την ολοκλήρωση του σχεδιασμού έπρεπε να σχεδιάσουμε την δομή της εφαρμογής.

Αρχικά θα έπρεπε να αποφασίσουμε πως θα είναι η δομή της.

Σχεδιάσαμε μια εφαρμογή με βασικές κατηγορίες και υποκατηγορίες που ο χρήστης θα μπορούσε να περιηγηθεί. Αρχικά ορίσαμε τις βασικές κατηγορίες

Και μέσα σε κάθε βασική κατηγορία ορίσαμε κάποια υλικά

- Π.χ ορίσαμε βασικές κατηγορίες όπως
- Κρέας
- Αλκοολούχα ποτά
- Ρύζι
- Μακαρόνια
- Κλπ
- Και στην κάθε κατηγορία ενδεικτικά προσθέσαμε και μερικά υλικά

Π.χ

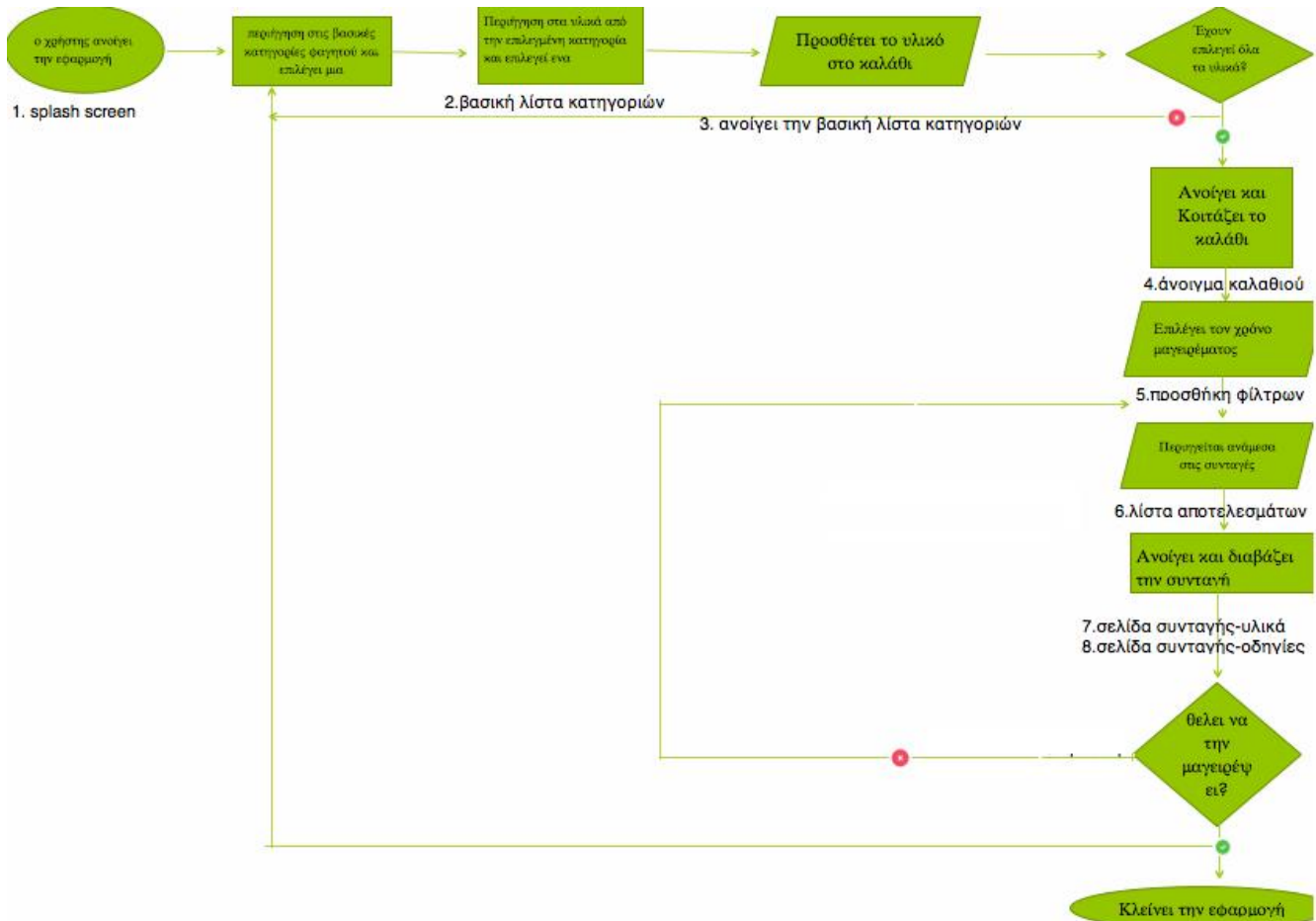
- Κρέας
 - Μοσχάρι
 - Χοιρινό

Ύστερα μερικά υλικά τα χωρίσαμε σε υποκατηγορίες

π.χ

- Μοσχάρι
 - Κιμάς
 - Ψαχνό

Εφόσον πλέον είχαμε την βασική δομή της εφαρμογής το επόμενο στάδιο που είναι ο σχεδιασμός του user flow



Εικόνα 1:User flow εφαρμογής

Επεξήγηση του user flow

Ο χρήστης ανοίγει την εφαρμογή του εμφανίζεται η αρχική εικόνα περιηγείται στην βασική κατηγορία,

ύστερα επιλέγει τα υλικά που έχει διαθέσιμα από τις βασικές λίστες κατηγοριών.

Σε κάθε μία λίστα επιλέγει τα υλικά που έχει διαθέσιμα αν έχει επιλέξει όλα τα υλικά ανοίγει το καλάθι αν όχι γυρνάει στις βασικές κατηγορίες φαγητού.

Αν έχει επιλέξει όλα τα υλικά συνεχίζει στις συνταγές που του εμφανίζει αν επιθυμεί επιλέγει το χρόνο την δυσκολία και το είδος γεύματος (χρήση φίλτρων) αν όχι συνεχίζει στα αποτελέσματα (συνταγές).

Επιλέγει μια συνταγή ανοίγει την σελίδα της συνταγής με τα υλικά και τις οδηγίες αν θέλει τότε συνεχίζει αν δεν θέλει ξαναγυρνάει στην περιήγηση των συνταγών αν δεν τον ικανοποιεί καμία μπορεί να γυρίσει στις βασικές κατηγορίες φαγητών ή να τερματίσει την εφαρμογή.

Τελευταία μας ενέργεια ήταν η εύρεση ονομασίας για την εφαρμογή μας. Ύστερα από σκέψη καταλήξαμε στην ονομασία «INSTACOOKIT» η οποία αναλύεται στο 4^ο κεφάλαιο πώς δημιουργήθηκε και τι ακριβώς σημαίνει.

.

viii) ΥΛΟΠΟΙΗΣΗ ΣΤΟ SKETCH

Υπάρχουν αρκετές επιλογές για σχεδιαστικά εργαλεία. μερικά από αυτά είναι γνωστά εκτός από του designers και στον απλό κόσμο. Ένα τέτοιο εργαλείο είναι το ευρέως γνωστό photoshop.

Ενδεικτικά κάποια σχεδιαστικά εργαλεία που βρήκαμε:

- Framer.js
- Flinto
- Invision
- Hammer
- Icon slate

Αλλά το εργαλείο που μας προτάθηκε από διάφορους designers ήταν το sketch. Οι λόγοι που κάναμε αυτή την επιλογή είναι οι εξής:

1. Εύκολο στη χρήση για σχεδιασμό ψηφιακών εφαρμογών.
2. Ελεύθερη βιβλιοθήκη με icons και components
3. Mirror σε κινητά Apple: Μπορούσαμε να βλέπουμε live την εφαρμογή που σχεδιάσαμε στο Apple κινητό μας
4. Είναι το εργαλείο που χρησιμοποιούνε πλέον οι περισσότεροι designers

Ξεκινήσαμε λοιπόν την δημιουργία οθονών. Όμως για να μπορέσουμε να τις δημιουργήσουμε καλύτερα έπρεπε να αποφασίσουμε και να προχωρήσουμε στην δημιουργία βασικών

κατηγοριών και τι υλικά θα έχουμε σε κάθε κατηγορία. Ύστερα θα πρέπει να επιλέξουμε τι υλικά θα έχει η κάθε κατηγορία και ποιές υποκατηγορίες υπάρχουν.

Οι βασικές κατηγορίες που επιλέξαμε ήταν:

1. Κρέας
2. Σάλτσες
3. Ζυμαρικά
4. Λαχανικά
5. Τυροκομικά
6. Μυρωδικά & Βότανα
7. Κονσέρβες
8. Αλκοολούχα
9. Πουλερικά
10. Ψάρια & θαλασσινά
11. Είδη ρυζιού
12. Όσπρια
13. Μπαχαρικά
14. Βούτυρο
15. Λάδι

Το επόμενο βήμα ήταν να εισάγουμε υλικά σε κάθε κατηγορία και να ορίσουμε υποκατηγορίες όπου χρειαζόμαστε.

- Κρέας
 - ΜΟΣΧΑΡΙ
 - § ΚΙΜΑΣ
 - § ΨΑΧΝΟ
 - ΧΟΙΡΝΟ
 - § ΚΙΜΑΣ
 - § ΨΑΧΝΟ
- Σάλτσες
 - ΝΤΟΜΑΤΟ ΠΕΛΤΕΣ
 - ΚΡΕΜΑ ΓΑΛΑΚΤΟΣ
- Ζυμαρικά
 - ΜΑΚΡΑ ΖΥΜΑΡΙΚΑ(ΜΑΚΑΡΟΝΙΑ Η ΣΠΑΓΓΕΤΙ)
 - § ΣΠΑΓΓΕΤΙ
 - § ΡΙΓΚΑΤΙ
 - § ΛΙΓΚΟΥΙΝΙ
 - ΚΟΦΤΑ ΖΥΜΑΡΙΚΑ
 - § ΜΑΚΑΡΟΝΙ ΧΟΝΤΡΟ ΚΟΦΤΟ ΚΑΙ ΡΙΓΕ
 - § ΠΕΝΕΣ ΑΠΛΕΣ
 - § ΠΕΝΕΣ ΡΙΓΕ ΤΡΙΚΟΛΟΡ
 - § ΡΙΓΓΑΤΟΝΙ
 - § ΚΟΧΥΛΙ
 - § ΦΙΟΓΚΑΚΙΑ
 - § ΜΑΚΑΡΟΝΑΚΙ ΚΟΦΤΟ
 - ΕΙΔΙΚΑ ΖΥΜΑΡΙΚΑ ΚΑΙ ΦΩΛΙΕΣ
 - § ΚΑΝΝΕΛΟΝΙΑ
 - § ΛΑΖΑΝΙΑ
 - § ΠΑΠΑΡΔΕΛΕΣ
 - § ΤΑΛΙΑΤΕΛΕΣ

- § ΦΙΔΕΣ
 - § ΨΙΛΑ ΖΥΜΑΡΙΚΑ
 - § ΚΡΙΘΑΡΑΚΙ ΧΟΝΤΡΟ
 - § ΚΡΙΘΑΡΑΚΙ ΜΕΤΡΙΟ
 - § ΡΥΖΟΝΙ
 - § ΡΥΖΑΚΙ
 - § ΠΕΠΟΝΑΚΙ
 - ΠΑΡΑΔΟΣΙΑΛΑ ΖΥΜΑΡΙΚΑ
 - § ΤΡΑΧΑΝΑΣ
 - § ΧΥΛΟΠΙΤΕΣ
 - § ΧΥΛΟΠΙΤΑΚΙ
 - § ΛΑΖΑΝΙΑ
 - § ΚΡΙΘΑΡΑΚΙ
 - ΓΕΜΙΣΤΑ ΖΥΜΑΡΙΚΑ
 - § ΤΟΡΤΕΛΙΝΙ
 - § ΡΑΒΙΟΛΙ
- Λαχανικά
 - ΑΓΓΟΥΡΙ
 - ΑΓΚΙΝΑΡΑ
 - ΑΡΑΚΑΣ
 - ΚΑΡΟΤΟ
 - ΚΟΛΟΚΥΘΑΚΙ
 - ΚΟΥΝΟΥΠΙΔΙ
 - ΚΕΜΜΥΔΙ
 - ΛΑΧΑΝΟ
 - ΜΑΡΟΥΔΙ
 - ΜΕΛΙΤΖΑΝΑ
 - ΜΠΑΝΙΑ
 - ΜΠΡΟΚΟΛΟ
 - ΝΤΟΜΑΤΑ
 - ΠΑΝΤΖΑΡΙ
 - ΠΑΤΑΤΑ
 - ΠΙΠΕΡΙΑ
 - ΠΡΑΣΟ
 - ΣΕΛΙΝΟ
 - ΣΚΟΡΔΟ
 - ΣΠΑΝΑΚΙ
 - ΣΠΑΡΑΓΓΙ
 - ΦΑΣΟΛΑΚΙ
- Τυροκομικά
 - ΦΕΤΑ
 - ΓΡΑΒΙΕΡΑ
 - ΚΕΦΑΛΟΤΥΡΙ
 - ΜΥΖΗΘΡΑ
 - ΚΑΣΕΡΙ
 - ΜΕΤΣΟΒΟΝΕ
 - ΠΑΡΜΕΖΑΝΑ
 - ΧΑΛΟΥΜΙ
 - ΑΝΘΟΤΥΡΟ

- Μυρωδικά & Βότανα
 - ΑΝΗΘΟΣ
 - ΒΑΣΙΛΙΚΟΣ
 - ΔΑΦΝΗ
 - ΔΕΝΤΡΟΛΙΒΑΝΟ
 - ΔΥΟΣΜΟΣ
 - ΕΣΤΡΑΓΚΟΝ
 - ΘΥΜΑΡΙ
 - ΚΟΛΛΙΝΑΤΡΟΣ
 - ΜΑΙΝΤΑΝΟΣ
 - ΜΑΤΖΟΥΡΑΝΑ
 - ΜΑΡΑΘΟΣ
 - ΡΙΓΑΝΗ
 - ΣΑΦΡΑΝ
 - ΣΧΙΝΟΠΡΑΣΟ
- Κονσέρβες
 - ΤΟΜΑΤΑΚΙ
 - ΜΑΝΙΤΑΡΙΑ
 - ΚΑΛΑΜΠΟΚΙ
 - ΦΑΣΟΛΙ ΚΟΚΚΙΝΟ
- Αλκοολούχα
 - ΚΡΑΣΙ
 - § ΚΟΚΚΙΝΟ
 - § ΛΕΥΚΟ
 - § ΡΟΖΕ
- Πουλερικά
- Ψάρια & θαλασσινά
- Είδη ρυζιού
- Όσπρια
- Μπαχαρικά
- Βούτυρο
- Λάδι

Αυτές Έπειτα βρήκαμε (2) συνταγές και σημειώσαμε τα υλικά που χρειάζονται για να πραγματοποιηθούν. Αυτές οι συνταγές ήταν:

ΣΥΝΤΑΓΗ 1 (ΜΟΣΧΑΡΑΚΙ ΚΟΚΚΙΝΙΣΤΟ)

- 1 ποτηράκι κρασί
- 1 κουταλιά της σούπας πελτέ
- 1 κονσέρβα τοματάκι
- 1 κρεμμύδι ψιλοκομμένο
- 1/2 φλυτζάνι βούτυρο
- αλάτι, πιπέρι

- 1 κιλό κρέας από μοσχάρι
- 2 σκελίδες σκόρδο
- λίγο μπαχάρι

ΣΥΝΤΑΓΗ 2 (ΚΑΝΕΛΛΟΝΙΑ ΜΕ ΚΙΜΑ)

- 1/2 φλυτζάνι κρασί
- αλάτι και πιπέρι
- 4 σκελίδες σκόρδο ψιλοκομμένο
- ρίγανη
- 2 μεγάλα κρεμμύδια ψιλοκομμένα
- 3 φλυτζάνια κεφαλοτύρι τριμμένο
- 1 φλυτζάνι λάδι
- 1/2 κιλό κιμά
- 2 κουταλιές πελτέ
- 1 πακέτο κανελόνια
- 1/2 ματσάκι μαϊντανό
- 2 κονσέρβες τοματάκι

Ύστερα σημειώσαμε τα υλικά από την κάθε συνταγή αντίστοιχα για να ξέρουμε ποία υλικά θα χρησιμοποιήσουμε για να τα “ενεργοποιήσουμε”.

Βρήκαμε τα βήματα εκτέλεσης για κάθε μια από τις δυο συνταγές που είχαμε επιλέξει ώστε να τα χρησιμοποιήσουμε όταν ο χρήστης ήθελε να δει την συνταγή και να προχωρήσει στην εκτέλεσή της.

ΕΚΤΕΛΕΣΗ ΣΥΝΤΑΓΗΣ 1

- Βάζετε σε μία κατσαρόλα το βούτυρο να κάψει
- Προσθέτετε το κρεμμύδι και το τσιγαρίζετε
- Προσθέτετε το κρέας και το τσιγαρίζετε
- Σβήνετε με κρασί
- Προσθέτετε την ντομάτα και τον πελτέ
- Ρίχνετε το σκόρδο και το μπαχάρι
- Αλατοπιπερώνετε
- Προσθέτετε 1 ποτήρι νερό

- Χαμηλώνετε τη φωτιά και αφήνετε να σιγοβράσει μέχρι να μείνει με τη σάλτσα του

ΕΚΤΕΛΕΣΗ ΣΥΝΤΑΓΗΣ 2

- Σε μία κατσαρόλα βάζετε το μισό λάδι να κάψει
- Προσθέτετε το 1 κρεμμύδι και το τσιγαρίζετε
- Προσθέτετε την 1 κονσέρβα τοματάκι και την 1 κουταλιά πελτέ
- Αλατοπιπερώνετε
- Προσθέτετε τη ρίγανη και τις 2 σκελίδες σκόρδο
- Αφήνετε να σιγοβράσει για 15 λεπτά
- Σε μία άλλη κατσαρόλα ζεσταίνετε το υπόλοιπο λάδι και τσιγαρίζετε το υπόλοιπο κρεμμύδι
- Προσθέτετε τον κιμά και τσιγαρίζετε ανακατεύοντας με πιρούνι για να μην σβολιάσει
- Προσθέτετε τις άλλες 2 σκελίδες σκόρδο
- Αλατοπιπερώνετε
- Σβήνετε με κρασί
- Προσθέτετε τον μαϊντανό
- Ρίχνετε την υπόλοιπη τομάτα και τον πελτέ και ανακατεύετε
- Ψήνετε μέχρι να απορροφηθούν όλα τα υγρά και να μείνει με τη σάλτσα του
- Γεμίζετε ένα-ένα κανελόνι με τον κιμά προσεκτικά και τα τοποθετείτε το ένα δίπλα στο άλλο σε ένα ταψί τα περιχύνετε με την σάλτσα που φτιάξατε
- Ψήνετε σε μέτριο φούρνο

Το επόμενο βήμα ήταν να πλαισιώσουμε την κάθε συνταγή με δυο τρεις συνταγές ακόμα. Κάνοντας όλη αυτή την διαδικασία σχεδιάζαμε πόσες οθόνες και τι θα έχει η κάθε μια από αυτές τις οθόνες για να τις δώσουμε την μορφή και την υπόσταση που θέλαμε με το sketch. Ύστερα σχεδιάσαμε τα φίλτρα έτσι όπως τα οραματιστήκαμε από τα σχόλια των χρηστών που ρωτήσαμε. Αυτή η επιλογή έδινε στον χρήστη την δυνατότητα να επιλέξει τον χρόνο που χρειάζεται το είδος του γεύματος και τον βαθμό δυσκολίας.

Θέλαμε να συγκεντρώνονται κάπου τα υλικά που έχει επιλέξει ο χρήστης αλλά σε όλες σχεδόν τις εφαρμογές το σημείο που συγκέντρωναν τις επιλογές του χρήστη ονομάζονταν καλάθι. Εμείς επειδή σχεδιάζαμε μια εφαρμογή για συνταγές σκεφτήκαμε να ορίσουμε ως καλάθι μια κατσαρόλα με το σκεπτικό ότι όλα τα υλικά θα καταλήξουν σε μαγειρικό σκεύος που το πιο κοινό είναι η κατσαρόλα.

Μια άλλη δυνατότητα που είχε η εφαρμογή ήταν να μπορεί ο χρήστης να προσθέσει υλικά που λείπουν από συνταγές που θα ήθελε να εκτελέσει.

Τώρα που είχαμε τελειώσει πλέον με τον σχεδιασμό το περιεχόμενο και την δομή των οθονών έπρεπε να βρούμε τις εικόνες που θα θέλαμε να χρησιμοποιήσουμε και τα εικονίδια. Τα εικονίδια που χρησιμοποιήσαμε ήταν από το web. (www.thenounproject.com)

Το μεγαλύτερο μέρος της εφαρμογής το είχαμε καλύψει, τώρα θα έπρεπε όλα όσα είχαμε μελετήσει να τα μεταφέρουμε στο σχεδιαστικό πρόγραμμα του sketch.

Το sketch είναι ένας ιδιόκτητος επεξεργαστής διανυσματικών γραφικών για το macOS της Apple, που αναπτύχθηκε από την ολλανδική εταιρεία Bohemian Coding. Κέρδισε βραβείο σχεδίου Apple το 2012. Το sketch κυκλοφόρησε για πρώτη φορά στις 7 Σεπτεμβρίου 2010. Λίγα λόγια για το sketch

Για να μπορέσουμε να ολοκληρώσουμε τον σχεδιασμό και να δώσουμε ζωή στην ιδέα μας θα πρέπει να πάρουμε όλη την έρευνα και να την μετατρέψουμε σε οθόνες
Αυτό κάναμε και εμείς σχεδιάσαμε μία-μία την κάθε οθόνη ξεχωριστά

ΑΡΧΙΚΑ:

Σχεδιάσαμε την πρώτη οθόνη που εμφανίζεται στον χρήστη όταν ανοίγει την εφαρμογή μας
ΒΗΜΑ 1

Έπρεπε να φτιάξουμε μια οθόνη για την βασική κατηγορία φαγητών, και μία οθόνη για κάθε μια κατηγορία ξεχωριστά με τα υλικά της. Γι αυτό τον λόγο δεν φτιάξαμε οθόνες για κάθε κατηγορία αλλά μόνο σε όσες ήταν χρήσιμες.

ΒΗΜΑ 2

Έπρεπε να σχεδιάσουμε μια οθόνη με το «καλάθι» μας

ΒΗΜΑ 3

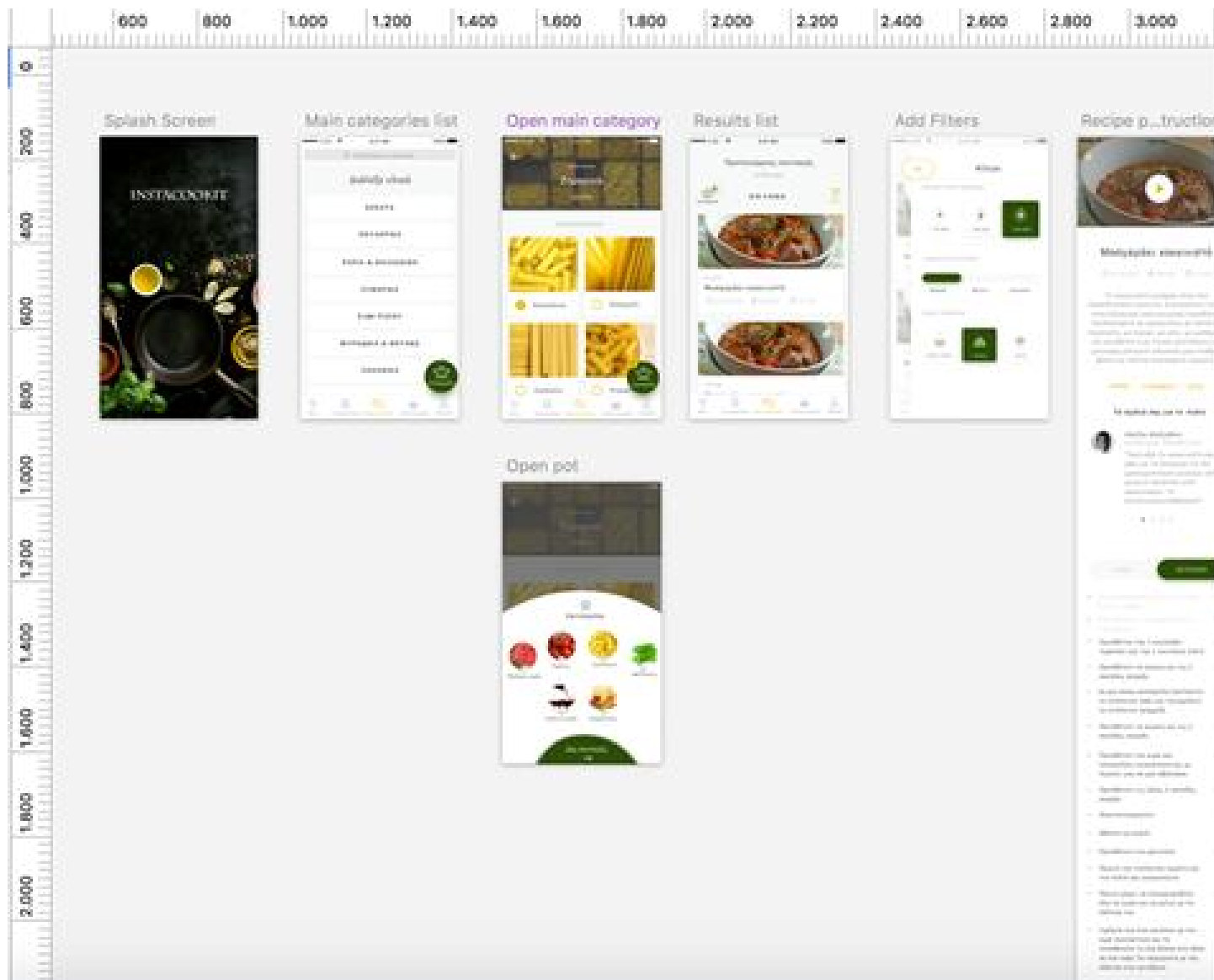
Να φτιάξουμε δύο οθόνες για τα αποτελέσματα των συνταγών, μία για κάθε συνταγή

ΒΗΜΑ 4

Δημιουργία μια οθόνης με τα φίλτρα που είχαμε σχεδιάσει

ΒΗΜΑ 5

Να φτιάξουμε τέσσερις (4) οθόνες, δυο οθόνες για κάθε συνταγή από τις δυο που είχαμε επιλέξει. Η μία οθόνη θα ήταν τα υλικά και η άλλη οι οδηγίες

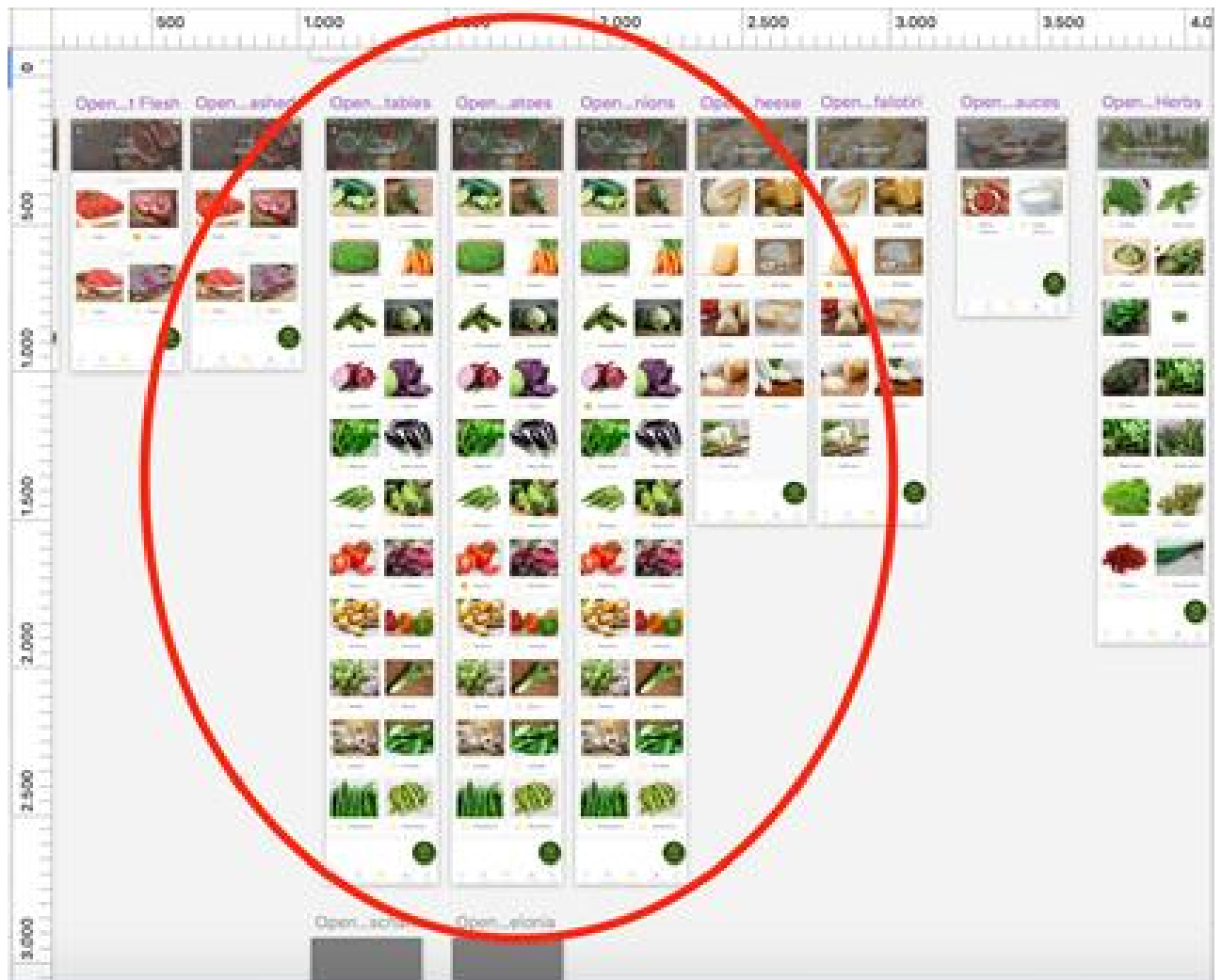


Εικόνα 1: print screen από τις οθόνες που δημιουργήσαμε στο sketch

Μόλις τελείωσε η δημιουργία των βασικών οθονών έπρεπε να δημιουργήσουμε αντίγραφα των οθονών ώστε να μπορούμε να επιλέξουμε τα υλικά της κάθε συνταγής

Και κάθε το αντίγραφο της αρχικής οθόνης να έχει και μία αλλαγή.

Π.χ για να μπορέσουμε να κλικάρουμε το μοσχάρι σαν υλικό θα έπρεπε να φτιάξουμε μία οθόνη με το μοσχάρι χωρίς να είναι επιλεγμένο και μία οθόνη με το μοσχάρι να έχει κλικαριστεί. Γιαυτό το λόγο το να κάνουμε όλες τις κατηγορίες και τα υλικά επιλέξιμα δεν θα είχε νόημα και θα μας έπαιρνε πολυ χρόνο.



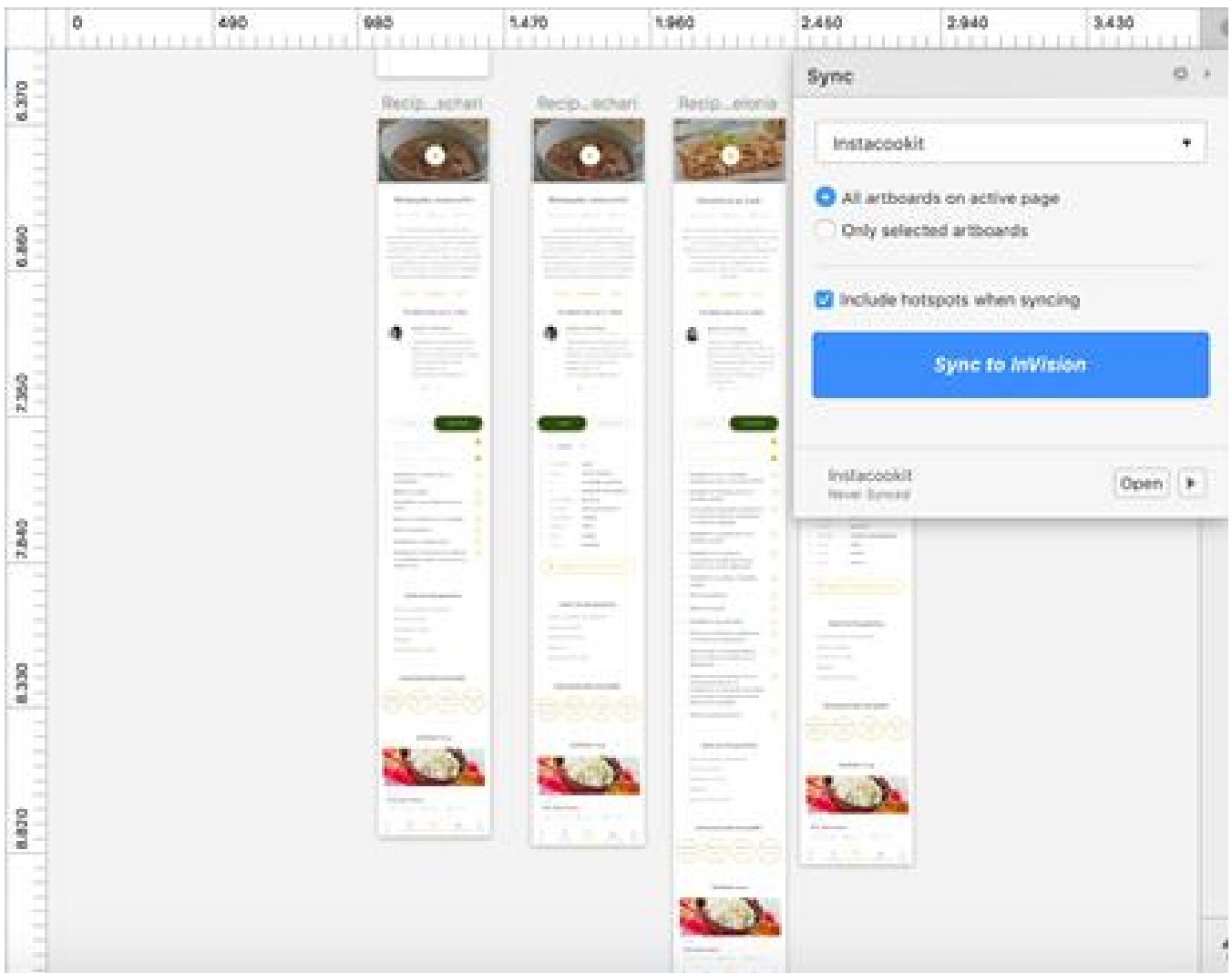
Εικόνα 2: print screen από τα αντίγραφα οθονών που δημιουργήσαμε στο sketch

Στην εικόνα βλέπετε μερικά αντίγραφα οθονών και στο κυκλωμένο σημείο βλέπετε ένα μέρος από τη διαδικασία που χρειάστηκε για να κάνουμε επιλέξιμα υλικά το κρεμμύδι, το τυρί κλπ.

Μία ακόμα λειτουργία του sketch που μας βοήθησε να ελέγχουμε την εφαρμογή στο κινητό μας ήταν το Mirror σε κινητά Apple: Μπορούσαμε να βλέπουμε live την εφαρμογή που σχεδιάσαμε στο Apple κινητό μας

Μόλις τελειώσαμε από το sketch θα πρέπει να συνεχίσουμε στο InVision για την ολοκλήρωση της εφαρμογής που σχεδιάσαμε.

Μεταφορά όλων των οθονών από το sketch στο Invision.



Εικόνα 3:Print screen με τον συγχρονισμό και μεταφορά εικόνων από το sketch στο invision.

ix)ΤΙ ΕΙΝΑΙ ΤΟ PROTOTYPE

Στην ανάπτυξη λογισμικού, ένα πρωτότυπο είναι ένα στοιχειώδες μοντέλο εργασίας ενός προϊόντος ή συστήματος πληροφοριών, συνήθως χτισμένο για σκοπούς επίδειξης ή ως μέρος της αναπτυξιακής διαδικασίας. Στο μοντέλο πρωτότυπου ανάπτυξης κύκλου ζωής (SDLC), βασική έκδοση του συστήματος κατασκευάζεται, δοκιμάζεται και στη συνέχεια επεξεργάζεται αναλόγως, μέχρις ότου επιτευχθεί τελικά ένα αποδεκτό πρωτότυπο από το οποίο μπορεί να αναπτυχθεί το πλήρες σύστημα ή προϊόν. Στο πρωτότυπο προγραμματισμό, ένα πρωτότυπο είναι ένα πρωτότυπο αντικείμενο, δημιουργούνται νέα αντικείμενα με την αντιγραφή του πρωτοτύπου. Στο σχεδιασμό υλικού, ένα πρωτότυπο είναι ένα μοντέλο που αντιπροσωπεύει ένα κατασκευασμένο (εύκολα αναπαραγόμενο) προϊόν επαρκώς για τους σχεδιαστές να οπτικοποιήσουν και να δοκιμάσουν το σχέδιο.

Το *πρωτότυπο* λέξη προέρχεται από τις λατινικές λέξεις *πρώτο*, που σημαίνει *πρωτότυπο*, και *τύπος*, που σημαίνει *μορφή ή μοντέλο*. Σε ένα μη τεχνικό πλαίσιο, ένα πρωτότυπο είναι ένα ιδιαίτερα αντιπροσωπευτικό παράδειγμα μιας δεδομένης κατηγορίας.

x) ΤΙ ΕΙΝΑΙ ΤΟ InVision

Το InVision είναι ένας απλός τρόπος για να σχεδιάσετε, να δημιουργήσετε και να δοκιμάσετε πρωτότυπα, επιχειρηματικά μοντέλα και σχέδια για οποιαδήποτε συσκευή. Περισσότεροι από 1,5 εκατομμύρια σχεδιαστές χρησιμοποιούν το InVision για να τροφοδοτήσουν τη διαδικασία σχεδιασμού τους.

Με το InVision, οι χρήστες μπορούν να τραβήξουν οποιαδήποτε εικόνα, σκίτσο ή σχεδίαση από διάφορες πηγές, συμπεριλαμβανομένου του PhotoShop, και να τα μετατρέψουν σε ένα διαδραστικό σχέδιο που μπορεί να εφαρμοστεί και να δοκιμαστεί σε οποιαδήποτε συσκευή. Οι χρήστες μπορούν εύκολα να μεταφέρουν την εικόνα στο σύστημα InVision με τη λειτουργία "drag and drop" ή να χρησιμοποιήσουν ενοποιησεις με το Dropbox, Box και το Google Drive για τον ομαλό συγχρονισμό των αρχείων. Εκτός από τη δημιουργία πρωτότυπων που μοιάζουν με το πραγματικό, το InVision παρέχει επίσης εργαλεία για συνεργασία, ανατροφοδότηση, ροή εργασιών και δοκιμές χρηστών.

Επισκόπηση των πλεονεκτημάτων InVision

Το InVision κάνει τη διαδικασία σχεδιασμού και πρωτοτύπου εύκολη και ανώδυνη για τις εταιρείες και τους σχεδιαστές τους. Αυτή η συγκεκριμένη λύση παρέχει στους χρήστες πρωτοποριακά εργαλεία και χαρακτηριστικά που τους επιτρέπουν να δημιουργήσουν καλύτερα, λειτουργικά πρωτότυπα που μπορούν να παρουσιάσουν στους πελάτες τους. Με ένα πρωτότυπο με δυνατότητα κλικ, καθιστά εύκολο για τους σχεδιαστές και τους πελάτες να βρίσκονται στην ίδια σελίδα μεταξύ τους, να βλέπουν τις ατέλειες, να καθορίζουν τις απαραίτητες βελτιώσεις κλπ.

Η κατασκευή ενός πρωτοτύπου από το έδαφος είναι σχετικά γρήγορη και εύκολη. Χάρη στη λειτουργία drag and drop της InVision, οι σχεδιαστές μπορούν απλώς να μεταφέρουν σχέδια, εικόνες και σκίτσα απευθείας από την πηγή και στη συνέχεια να τα βάζουν στο σύστημα InVision. Η οι σχεδιαστές μπορούν να επιλέξουν να ανακτήσουν εικόνες από υπηρεσίες online αποθήκευσης, όπως Dropbox, Box και Google Drive.

Η συνεργασία είναι ένα από τα ισχυρότερα χαρακτηριστικά του InVision. Οι σχεδιαστές μπορούν να εργάζονται ξεχωριστά χωρίς κανένα πρόβλημα ή παρενόχληση, αλλά μπορούν εύκολα να επικοινωνούν με τους συναδέλφους σχεδιαστές και τους πελάτες τους για ταχύτερη ανάδραση, προτάσεις, ιδέες και πολλά άλλα. Με το InVision, είναι εύκολο για τους χρήστες να μοιράζονται συνδέσμους για γρήγορη ανατροφοδότηση και να δίνουν άμεσα σχόλια αντί να χρειάζεται να μοιράζονται JPGs ή PDF και να περιμένουν μια απάντηση από το άλλο μέρος.

Καθώς το πρωτότυπο έχει μια πιο ρεαλιστική αίσθηση σε αυτό και λειτουργεί σε ένα πραγματικό πρόγραμμα περιήγησης ή κινητό mockup, οι πελάτες είναι σε θέση να αποκτήσουν περισσότερες πληροφορίες σχετικά με το χρώμα, το μέγεθος και την εμπειρία του προγράμματος περιήγησης.

Επισκόπηση των χαρακτηριστικών του InVision

·	σμός πρωτοτύπων	Σχεδια
·	ση κοινής χρήσης και παρουσίαση	Σχεδια
·	ση σχολίων και απόψεων	Σχεδια
·	ήσεις σχεδίασης σε πραγματικό χρόνο και Whiteboarding	Συναντ
·	σμός, Οργάνωση και Συνεργασία	Σχεδια
·	ιση Έργων για Σχεδιαστές	Διαχείρ
·	ς χρηστών και έρευνα	Δοκιμέ
·	οίηση χρηστών	Ικανοπ

Αντιλαμβανόμαστε ότι όταν αποφασίζετε να αγοράσετε λογισμικό ανάπτυξης λογισμικού, είναι σημαντικό όχι μόνο να δείτε πώς οι εμπειρογνώμονες την αξιολογούν στις αναθεωρήσεις τους αλλά και να μάθετε εάν οι πραγματικοί άνθρωποι και οι εταιρείες που το αγοράζουν ικανοποιούν πραγματικά το προϊόν. Γι 'αυτό δημιουργήσαμε το Customer Satisfaction Algorithm™ με βάση τη συμπεριφορά μας, το οποίο συγκεντρώνει κριτικές πελατών, σχόλια και κριτικές InVision σε ένα ευρύ φάσμα ιστότοπων κοινωνικής δικτύωσης. Στη συνέχεια, τα δεδομένα παρουσιάζονται σε μια εύχρηστη φόρμα που δείχνει πόσοι άνθρωποι είχαν θετική και αρνητική εμπειρία με το InVision. Με αυτές τις πληροφορίες θα πρέπει να είστε έτοιμοι να κάνετε μια ενημερωμένη απόφαση αγοράς.

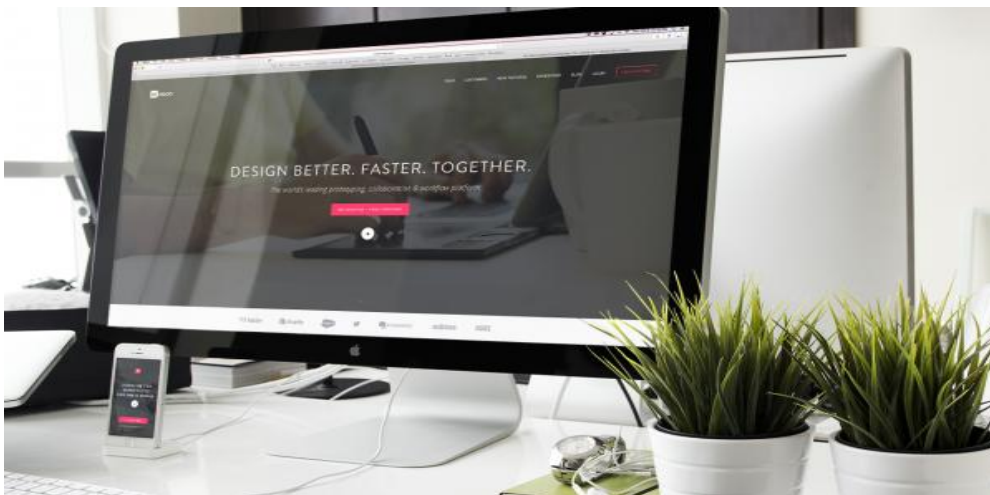
Εφαρμογή InVision: Οι Τελευταίες στη Συνεργατική Σχεδίαση

Υπάρχει μια νέα εφαρμογή που αναλαμβάνει τη συνεργασία σχεδίασης UX. Έχει σχεδόν δύο εκατομμύρια χρήστες παγκοσμίως και ήδη το 70% των εταιρειών του Fortune 100 το χρησιμοποιεί επίσης. Αυτό περιλαμβάνει Disney, IBM, Apple και Verizon για να αναφέρουμε μόνο μερικά. Ο ιστότοπός τους είναι "η κορυφαία πλατφόρμα πρωτοτύπων, συνεργασίας και ροής εργασιών στον κόσμο." Ονομάζεται InVision και αν δεν το χρησιμοποιείτε ακόμα, πρέπει να το δοκιμάσετε.

Σχεδιασμός InVision

Το InVision είναι μια δωρεάν πλατφόρμα πρωτοτύπων για web και κινητά που επιτρέπει στους σχεδιαστές της να αναπτύσσουν και να συνεργάζονται με ευκολία. Είναι τόσο απλό όσο η μεταφόρτωση των αρχείων σας, είτε πρόκειται για έγγραφα του Photoshop, έγγραφα Sketch, GIF, JPG, PNG ή PDF, και προσθέτοντας κινούμενες εικόνες και μεταβάσεις για να μετατρέψετε τις στατικές οθόνες σε διαδραστικά πρωτότυπα με δυνατότητα κλικ.

Αλλά είναι πολύ περισσότερα από αυτό.



Σχεδιασμός InVision

Η ίδια η εταιρεία είναι 100% απομακρυσμένη με 185 μέλη της ομάδας σε 14 διαφορετικές χώρες. Αυτό το μοναδικό set up δίνει στην ομάδα InVision μεγάλη εικόνα για το τι ψάχνουν

ακριβώς οι συνεργατικές ομάδες στις πλατφόρμες σχεδιασμού τους. Ο Διευθύνων Σύμβουλος της InVision, Clark Valberg, μας μίλησε για την υπευθυνότητα και την υπευθυνότητα των εργαζομένων: "Υπάρχουν δύο διαφορετικές διαστάσεις για την κοινωνική συνιστώσα της εργασίας: υπάρχει η συνθετική συνιστώσα και υπάρχει η συνιστώσα της κοινότητας." Σύμφωνα με τον ίδιο, για να τροφοδοτήσει το συνεργατικό στοιχείο, η InVision εξασφαλίζει στους υπαλλήλους τους το βέλτιστο λογισμικό και τις πλατφόρμες που τους διατίθενται - Slack, Trello, Dropbox και άλλα αγαπημένα. Αυτά τα εργαλεία διευκολύνουν τη δημιουργία μιας σταθερής διαδικασίας ροής εργασίας και τη δημιουργία θετικού ρυθμού για τις συναντήσεις της εταιρείας.

Όσον αφορά την πτυχή της "κοινότητας", ο Valberg λέει: "Ακριβώς επειδή εργάζεστε εξ αποστάσεως δεν σημαίνει ότι πρέπει να εργαστείτε μόνος σας". Το InVision έχει φυσικά γραφεία σε μερικές μεγάλες πόλεις και προσφέρει επιδότηση για τους υπαλλήλους που αναζητούν χώρους γραφείου ιδιαίτερη πατρίδα.

Αλλά πώς ακριβώς αντικατοπτρίζεται η κουλτούρα της εταιρείας στην ίδια την εφαρμογή;

Για αρχάριους, το InVision (η εφαρμογή) συνδυάζεται με τις ίδιες συνεργατικές πλατφόρμες που χρησιμοποιεί η εταιρία τους κάθε μέρα. Οι υπάλληλοι της InVision χρησιμοποιούν το Dropbox Business για να παραμείνουν 100% απομακρυσμένες και να παρακολουθούν τα σημαντικά δεδομένα της εταιρείας. Οι χρήστες InVision έχουν τις ίδιες διαθέσιμες επιλογές. Μέσα από την εφαρμογή, οι χρήστες μπορούν να συγχρονίσουν τον λογαριασμό Dropbox με τον λογαριασμό InVision, γεγονός που καθιστά εύκολη την περιήγηση και την κοινή χρήση αρχείων σχεδιασμού.

Επιπλέον, οι χρήστες μπορούν να αντιγράψουν και να επικολλήσουν κώδικες "ζωντανής ενσωμάτωσης" από τα έγγραφα InVision τους απευθείας σε πλατφόρμες επικοινωνίας όπως Trello, Basecamp, Slack και πολλά άλλα. Οι ενσωματώσεις της εφαρμογής "προωθούν αυτόματα τη δραστηριότητα από τα αγαπημένα σας συστήματα κατευθείαν στο InVision".

Σχεδιασμός InVision

Όσον αφορά την κοινότητα, το InVision καθιστά ανώφελη τη συμμετοχή ολόκληρης της ομάδας σας στη δημιουργική διαδικασία δημιουργίας φιλικών προς το χρήστη σχεδίων ιστού. Οι συμπαίκτες μπορούν να σχολιάσουν άμεσα τα σχέδια και να συμμετάσχουν σε συνεργασία σε πραγματικό χρόνο, πραγματοποιώντας συναντήσεις εντός του προγράμματος περιήγησης μέσω της εφαρμογής InVision ή της προσθήκης LiveShare Photoshop. Όλα τα σχόλια και οι αλλαγές σχεδιασμού αποθηκεύονται για να κάνουν την ιστορία αναθεώρησης της προβολής αβίαστη.

Έξω από τις συναντήσεις στο πρόγραμμα περιήγησης υπάρχει ακόμα ένα ισχυρό εργαλείο συνεργασίας που ονομάζεται Boards. Οι πίνακες είναι σαν το Pinterest για τα σχέδια. Παρέχουν ένα χώρο για οδηγούς στυλ, γκαλερί, moodboards και πολλά άλλα. Χρησιμοποιήστε πίνακες για να αναπτύξετε παλέτες χρωμάτων, να επιλέξετε γραμματοσειρές και να δημιουργήσετε οπτική ιεραρχία για μεμονωμένα έργα.

Αν δεν έχετε ερωτευτεί ακόμα την ιδέα του InVision, σίγουρα πρόκειται να το κάνετε. Πέντε λέξεις: δωρεάν και απεριόριστη δοκιμή χρηστών. Απλά στείλτε στο δοκιμαστικό κοινό σας ένα σύνδεσμο με το πρωτότυπο σας. Όταν το κοινό ανοίξει τη σύνδεση χρησιμοποιώντας το iPhone, η κάμερα που βρίσκεται μπροστά στρέφεται για να καταγράψει πώς αλληλεπιδρά ο χρήστης με το πρωτότυπο. Οι χειρονομίες ήχου και οθόνης καταγράφονται επίσης και προσφέρουν ζωτικής σημασίας πληροφορίες σχετικά με το σχεδιασμό. Σύμφωνα με την ιστοσελίδα της InVision, "Τα πρωτότυπα παρουσιάζονται για δοκιμές χρηστών με υψηλή πιστότητα, έτσι ώστε να φαίνονται πραγματικές και να αισθάνονται πραγματικούς χρήστες που επικεντρώνονται στο ίδιο το σχέδιο".



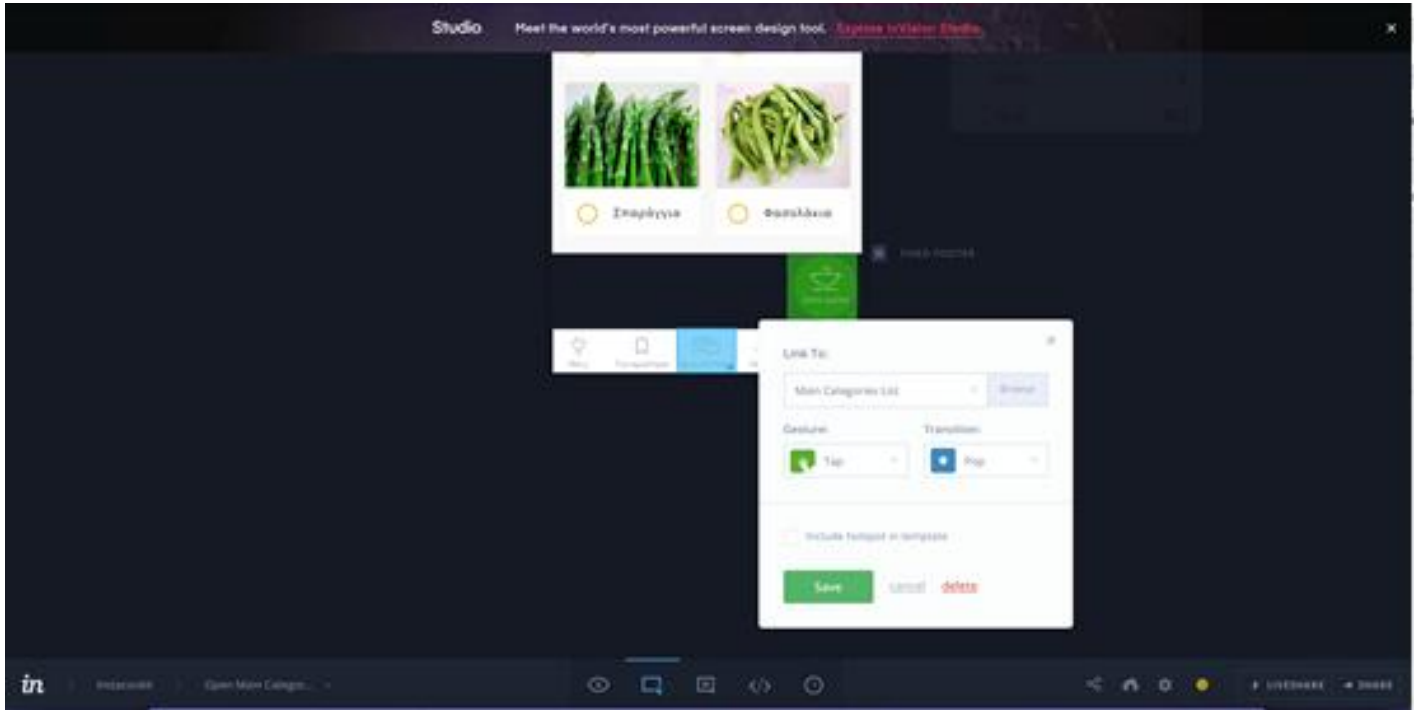
Πρόσφατα η ομάδα InVision ένωσε τις δυνάμεις της με την Silver Flows, μια εφαρμογή που επιτρέπει στους σχεδιαστές να δημιουργούν πρωτότυπα απευθείας στο Sketch. «Οι σχεδιαστές μπορούν να σχεδιάζουν και να κατασκευάζουν πρωτότυπα στον ίδιο χώρο, μπορούν να δοκιμάσουν και να επικυρώσουν πιο γρήγορα την ευρύτερη εμπειρία χρηστών που παρέχουν οι εφαρμογές τους. Μαζί, η InVision και η Silver Flows δημιουργούν ένα δημιουργικό σύννεφο μέσα στο Sketch το οποίο προσφέρει έναν απαράμιλλο βαθμό συνέχειας για τους σχεδιαστές και τις ομάδες τους. Είναι αρκετά προφανές ότι η InVision κάνει κάποιες σημαντικές αλλαγές στον κόσμο του πρωτοτύπου και της συνεργασίας για το σχεδιασμό ιστοσελίδων.

Οι λόγοι που επιλέξαμε το InVision:

- } Online εργαλείο (δεν χρειάζεται εγκατάσταση): Μας βοήθησε να επεξεργαζόμαστε όλοι μαζί το ίδιο project απομακρυσμένα από το σπίτι μας
- } Εύκολο και απλό στη χρήση
- } Ωραίο οπτικό αποτέλεσμα για την παρουσίαση μιας εφαρμογής κινητού σε desktop και σε mobile
- } Plug-in για να επικοινωνεί με το σχεδιαστικό μας εργαλείο: Καθοριστικός παράγοντας επιλογής prototyping εργαλείου

Περάσαμε όλες τις οθόνες που είχαμε δημιουργήσει στο sketch και ύστερα κάναμε τις διασυνδέσεις από οθόνη σε οθόνη. Επιλέγαμε το είδος πατήματος του κάθε κουμπιού

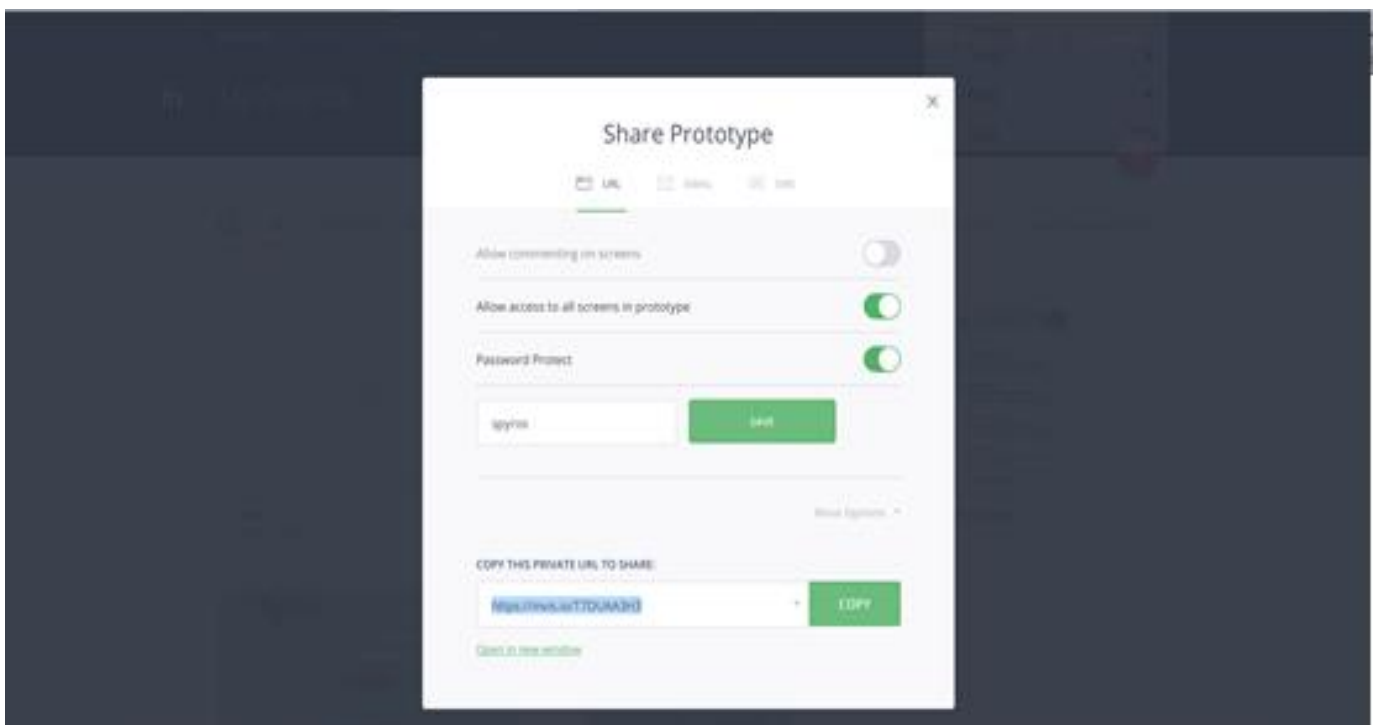
και τρόπο εναλλαγής και που μεταφερόμασταν μόλις πατούσαμε το κάθε κουμπί όπως



φαίνεται στην εικόνα παρακάτω

Εικόνα 2:Print screen από το invision με την επιλογή κλικαρίσματος και εναλλαγής οθονών

Και αφού φτιάξαμε όλες τις κινήσεις τότε αποθηκεύσαμε το prototype και το δώσαμε σε



κάποιους χρήστες ώστε να καταγράψουμε τις εντυπώσεις τους

Εικόνα 3:Print screen από το invision με τον τρόπο διανομής

xi) ΜΕΛΛΟΝΤΙΚΕΣ ΚΙΝΗΣΕΙΣ

Για τον σχεδιασμό και υλοποίηση μιας εφαρμογής ακολουθούνται κάποια βήματα ή καλύτερα θα μπορούσαμε να τα παρουσιάσουμε σαν στάδια, η ομάδα μας ακολούθησε αυτά τα στάδια αυστηρά για να φτάσει έως εδώ.

Αυτά τα στάδια για την δημιουργία μιας εφαρμογής είναι

1. Να ανακαλύψει την ιδέα
2. Να την προσδιορίσει
3. Να την σχεδιάσει, τεστάρει να την μετατρέψει σε prototype (αυτή είναι μια κυκλική διαδικασία που κάναμε μέχρι να βρούμε το τελικό αποτέλεσμα που θα είχε επιτυχία)
4. Να φτιαχτεί και να προγραμματιστεί η εφαρμογή
5. Να μετρηθεί (αναλυθεί και να τελειοποιηθεί)

Σαν φοιτητές του τμήματος ΔΙΟΙΚΗΣΗΣ ΕΠΙΧΕΙΡΗΣΕΩΝ οι γνώσεις και το αντικείμενο μελέτης μας ήταν να φτάσουμε την εφαρμογή μέχρι εκείνο το σημείο που θα ήταν έτοιμη για να παραδοθεί στο τμήμα που χρειάζεται για να ολοκληρωσή της.

Να σχεδιαστεί σύμφωνα με τις απαιτήσεις των χρηστών, σύμφωνα με τις ανάγκες που διαπιστώσαμε ότι καλύπτει και σύμφωνα με τους στόχους που είχαμε ορίσει.

Κάναμε ότι χρειαζόταν για να παραδώσουμε ένα τέλειο και ολοκληρωμένο project μιας εκπαιδευτικής εφαρμογής για κινητές πλατφόρμες iOS.

Πλέον έχουμε στα χέρια μας ένα τέλειο προϊόν σχεδιασμένο και το επόμενο βήμα είναι η εύρεση ενός εξίσου καλά καταρτισμένου τμήματος προγραμματισμού για κινητές εφαρμογές iOS για να μπορέσει να το υλοποιήσει σωστά την εφαρμογή και να βγει στην αγορά. Να χρησιμοποιηθεί από όλους τους χρήστες για το σκοπό που δημιουργήθηκε και να προσφέρει όλα αυτά για τα οποία σχεδιάστηκε έχοντας ως στόχο την αποδοχή από τους χρήστες και την επιτυχία.

Στο επόμενο κεφάλαιο γίνεται επεξηγηματική παρουσίαση της εφαρμογής για τον τρόπο που λειτουργεί και για τις ενέργειες που μπορεί να κάνει ο χρήστης.

ΚΕΦΑΛΑΙΟ 4 – ΕΦΑΡΜΟΓΗ

Ο βασικός στόχος αυτής της εφαρμογής είναι να προσφέρει στον χρήστη την δυνατότητα να επιλέξει από μια πληθώρα συνταγών και να τον εκπαιδεύσει πάνω στην μαγειρική .

Αυτός ο στόχος μας οδήγησε στον σχεδιασμό μιας εφαρμογής και στην εξέλιξη της σύμφωνα με τις ανάγκες και τις απαιτήσεις που έχουν οι χρήστες μιας τέτοιας εφαρμογής.

Έτσι οδηγηθήκαμε στον σχεδιασμό μιας εφαρμογής που ικανοποιεί αυτό το βασικό στόχο και προσφέρει και πάρα πολλές ακόμα δυνατότητες

Κατά τα φοιτητικά μας χρόνια η διατροφή μας αποτελούνταν αποκλειστικά από πρόχειρο φαγητό είτε φαγητό αποθηκευμένο σε καταψύκτες με τις περισσότερες φορές η διαδρομή να είναι φαγητό από το σπίτι-κατάψυξη-σκουπίδια σπαταλώντας έτσι αρκετά χρήματα.

Η αρχική ιδέα ήταν πολύ απλή και σχεδιάστηκε μόνο για τον φοιτητή
Δηλαδή να μπορέσει ο φοιτητής-χρήστης ανοίξει την εφαρμογή και απλά να επιλέξει κάποια καθημερινά υλικά, απλά υλικά που σίγουρα είχε στην κατοχή του μέσα στην κουζίνα του και οι επιλογές του αυτές τον οδηγούσαν σε μια συνταγή και στο πώς θα καταφέρει να την ολοκληρώσει-μαγειρέψει.

Σαν ιδέα ήταν αξιόλογη αλλά ήταν πάρα πολύ απλή και δεν πρόσφερε πολλές δυνατότητες στον φοιτητή-χρήστη γι αυτό και ήταν επιτακτική η ανάγκη για εξέλιξη της υπάρχουσας ιδέα και την τροποποίηση της για προσφέρει όλα όσα ήταν ικανή και να μπορέσει να έχει επιτυχία και χρησιμότητα.

Η αναζήτηση αυτή μας έφερε στην επιφάνεια πολλές δυνατότητες που θα μπορούσε να έχει αυτή η εφαρμογή όχι μόνο στον φοιτητή αλλά και σε οποιονδήποτε άλλον χρήστη και στοχεύοντας και σε ομάδες χρηστών που είχαν κάποια ιδιαίτερα χαρακτηριστικά γνωρίσματα, ένα παράδειγμα είναι οι χορτοφάγοι χρήστες.

Αρχικά ο χρήστης ανοίγοντας της εφαρμογή βλέπει την αρχική εικόνα και το όνομα της εφαρμογής που χρησιμοποιεί.

Το όνομα της εφαρμογής προήλθε από την ένωση τριών (3) λέξεων

Από την λέξη instant που σημαίνει άμεσος την λέξη cook που σημαίνει μαγειρεύω και την λέξη kit που σημαίνει εξοπλισμός, σύνεργα θέλοντας να δηλώσει ότι είναι μια εφαρμογή που σε βοηθάει άμεσα να μαγειρέψεις με τα σύνεργα που διαθέτεις!



Εικόνα 1;SPLASH SCREEN

Αμέσως μετά την αρχική εικόνα ο χρήστης ξεκινάει την χρήση της εφαρμογής.

Του εμφανίζεται η λίστα με τις βασικές κατηγορίες υλικών με σκοπό να περιηγηθεί στην εκάστοτε κατηγορία που έχει επιλέξει.

Η βασικές κατηγορίες είναι:

- Κρέατα
- Σάλτσες
- Ζυμαρικά
- Λαχανικά
- Τυροκομικά
- Μυρωδικά και βότανα
- Κονσέρβες

- Αλκοολούχα ποτά
- Πουλερικά
- Ψάρια και Θαλασσινά
- Είδη ρυζιού
- Όσπρια
- Μπαχαρικά
- Βούτυρο
- Λάδι



Εικόνα 2: Βασική κατηγορία υλικών υλικών



Εικόνα 3 Βασική κατηγορία

Στο πάνω μέρος υπάρχει η αναζήτηση, ενώ στο κάτω μέρος της οθόνης δεξιά υπάρχει το καλάθι (κατσαρόλα) που προστίθονται τα υλικά που επιλέγουμε

Στο κάτω μέρος της οθόνης μια μπάρα εργαλείων (toolbar) και υπάρχουν τα εικονίδια

Ιδέες

Για αργότερα

Βρες την συνταγή

Λίστα αγορών

Προφίλ του χρήστη.



Εικόνα 4: Η μπάρα εργαλείων που έχουμε επιλέξει

Η αναζήτηση και το καλάθι παραμένουν σταθερά μέχρι ο χρήστης να επιλέξει το κουμπί Δες συνταγή ενώ η μπάρα εργαλείων παραμένει σταθερή καθόλη την διάρκεια που ο χρήστης χρησιμοποιεί την εφαρμογή.

Ιδέες: Ο χρήστης μπορεί να καταχωρήσει ιδέες από συνταγές που έχει χρησιμοποιήσει ή να διαβάσει ιδέες από άλλους χρήστες

Για αργότερα: Μπορεί να αποθηκεύσει συνταγές για αργότερα

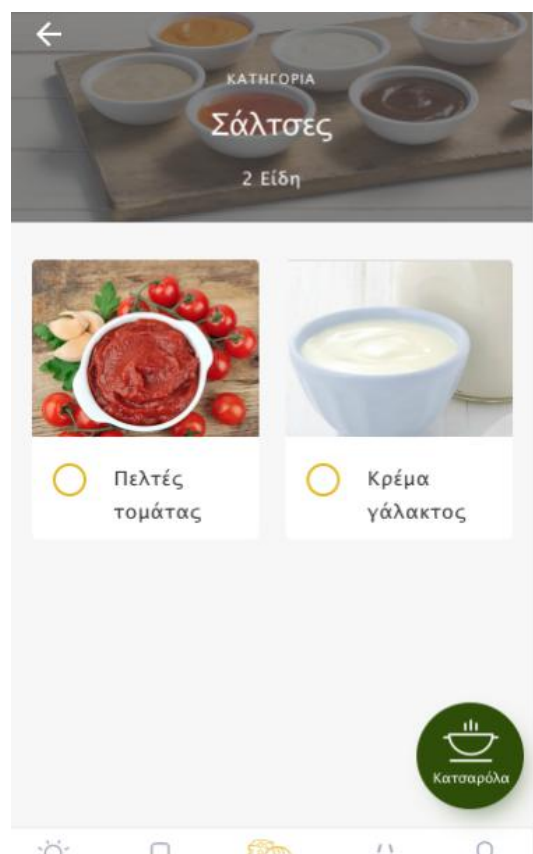
Βρες την συνταγή: Βρίσκει τις συνταγές σύμφωνα με τα υλικά που έχει επιλέξει

Λίστα αγορών: Μπορεί να δει το καλάθι αγορών του

Προφίλ του χρήστη: Μπορεί να επισκευτεί το προφίλ του, εκεί έχει φωτογραφία του και πληροφορίες γενικά για τον χρήστη όπως ημερομηνία γέννησης, τόπος διαμονής, αγαπημένες συνταγές κλπ.

Αφού επιλέξει μια από τις βασικές κατηγορίες του ανοίγει μια νέα λίστα με υλικά της κατηγορίας που έχει ήδη επιλέξει και περιηγείται στην κάθε μια κατηγορία κλικάροντας το

υλικό που έχει στην διάθεση του.



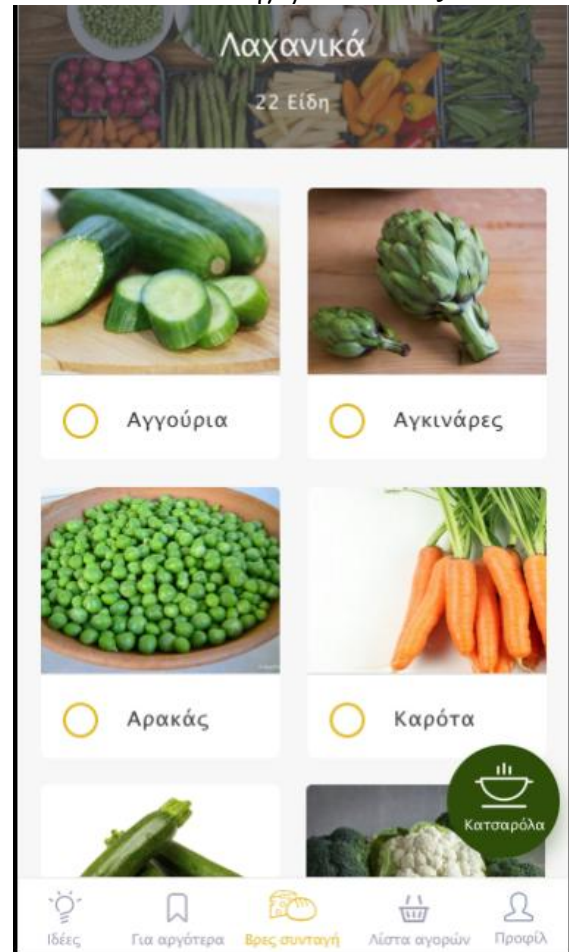
Εικόνα 5 Κατηγορία Κρέατα



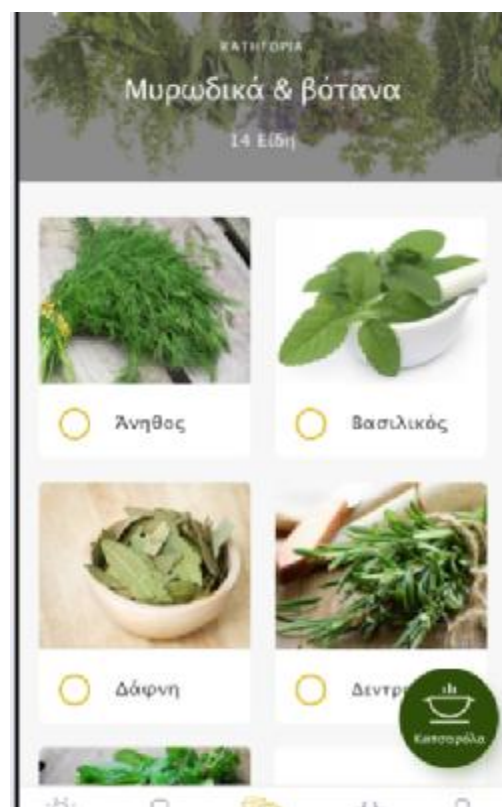
Εικόνα 7: Κατηγορία Ζυμαρικά



Εικόνα 6: Κατηγορία Σάλτσες

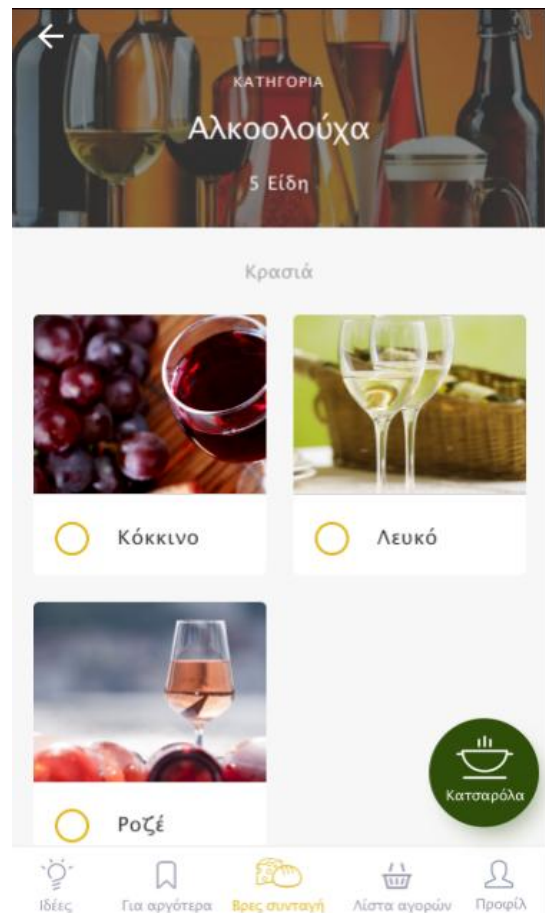
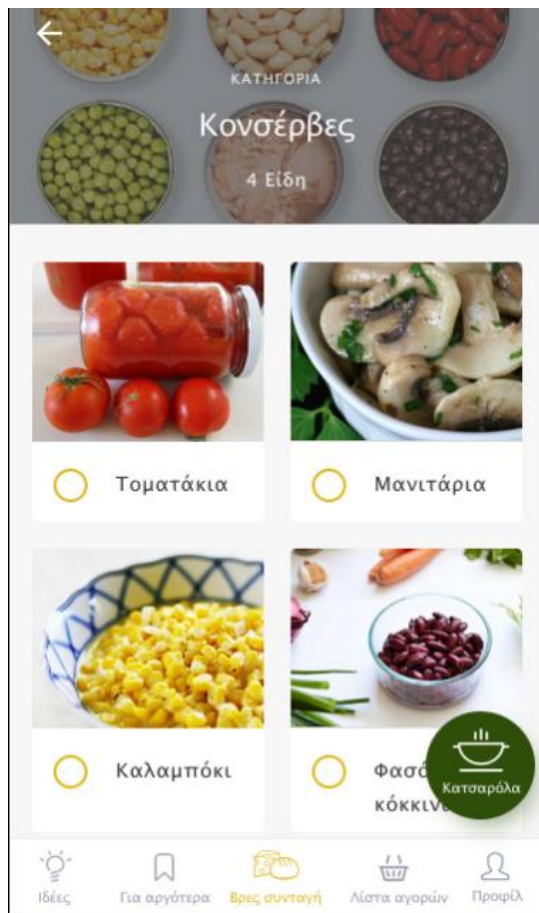


Εικόνα 8: Κατηγορία Λαχανικά



Εικόνα 9:Κατηγορία Τυροκομικά βότανα

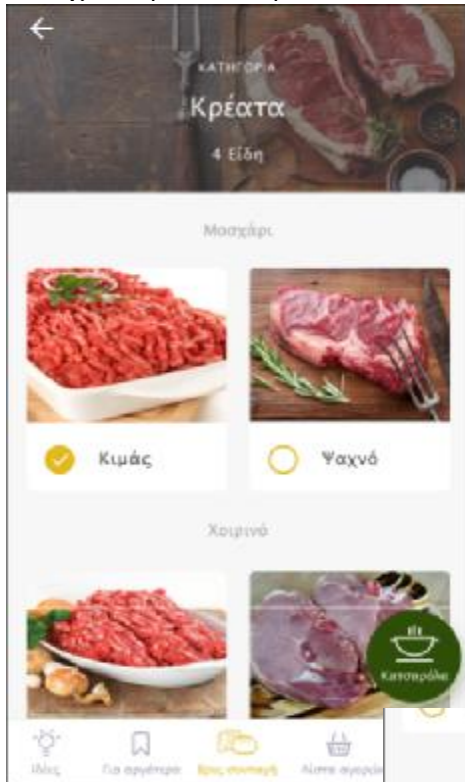
Εικόνα 10:Κατηγορία Μυρωδικά &



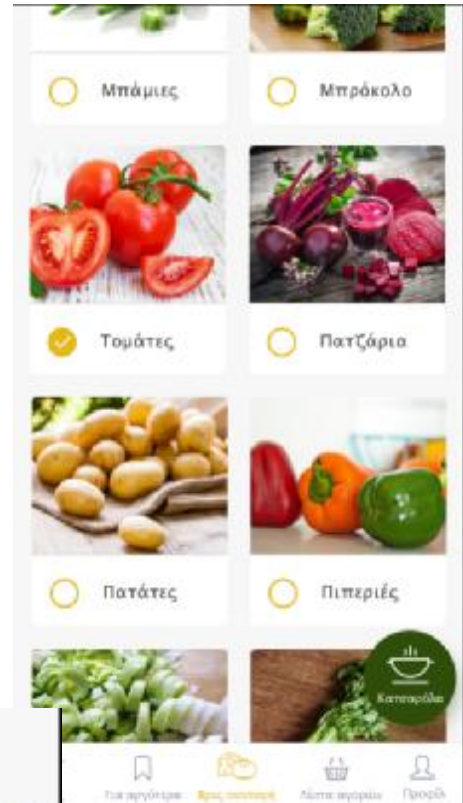
Εικόνα 11:Κατηγορία Κονσέρβες

α 12:Κατηγορία Αλκοολούχα

Ενδεικτικά κάποιες κατηγορίες που μπορεί να περιηγηθεί ο χρήστης και να επιλέξει τα υλικά που έχει στην διάθεσή του



Εικόνα 12:Επιλεγμένο υλικό κιμάς
Εικόνα 13:Επιλεγμένο υλικό Τομάτες



Κανελόνια
Αφού ο χρήστης επιλέξει τα θέλει να χρησιμοποιήσει επισκόπηση στο καλάθι του υλικό.

Εικόνα 14:Επιλεγμένο υλικό υλικά που έχει διαθέσιμα και τότε μπορεί να κάνει μια να δει αν έχει επιλέξει όλα τα

15:Επισκόπηση

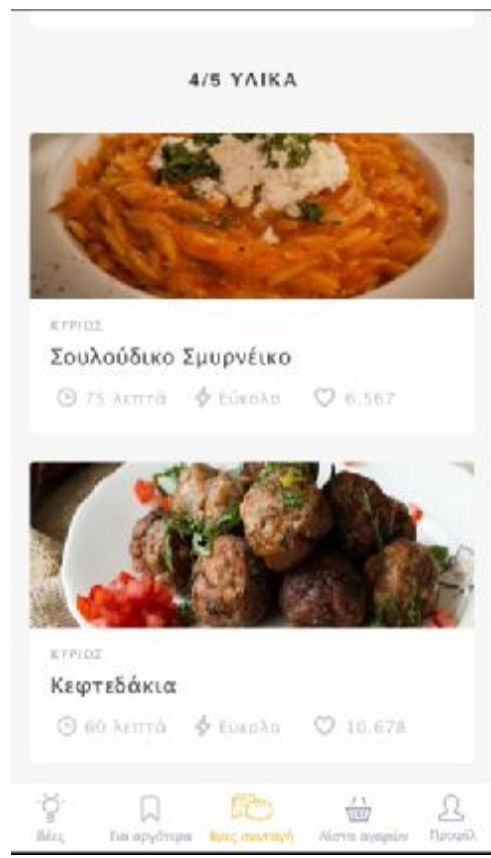
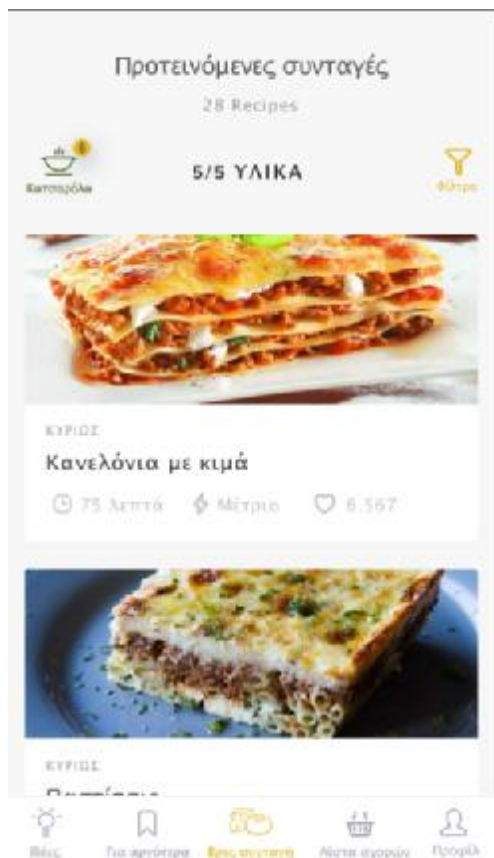
Στην συνέχεια αν υλικά που έχει στη μπορεί να συνεχίσει συνταγών Ο χρήστης μπορεί στις εμφανιζόμενες να επιλέξει την αρέσει.



Εικόνα καλαθιού

έχει επιλέξει όλα τα διάθεσή του τότε στην εύρεση

τώρα να περιηγηθεί συνταγές και τελικά συνταγή που του

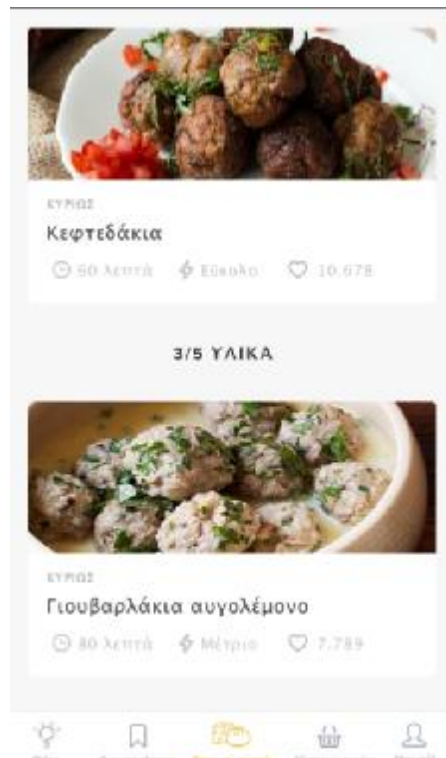


Εικόνα 16: Αποτελέσματα συνταγών

Εικόνα 17: Αποτελέσματα συνταγών

συνταγών

Στην οθόνη με τις
(Αποτελέσματα συνταγών)
Από πάνω από την εικόνα
πόσα υλικά κατέχει ο
χρειάζονται για την
συγκεκριμένης της συνταγής



Εικόνα 18: Αποτελέσματα

προτεινόμενες συνταγές
υπάρχουν κάποια στοιχεία
της συνταγής εμφανίζει
χρήστης από αυτά που
πραγματοποίηση της

19: Τα
λείπουν

Από
την
κάθε
υπάρχει το είδος του γεύματος (ορεκτικό, κυρίως κλπ)



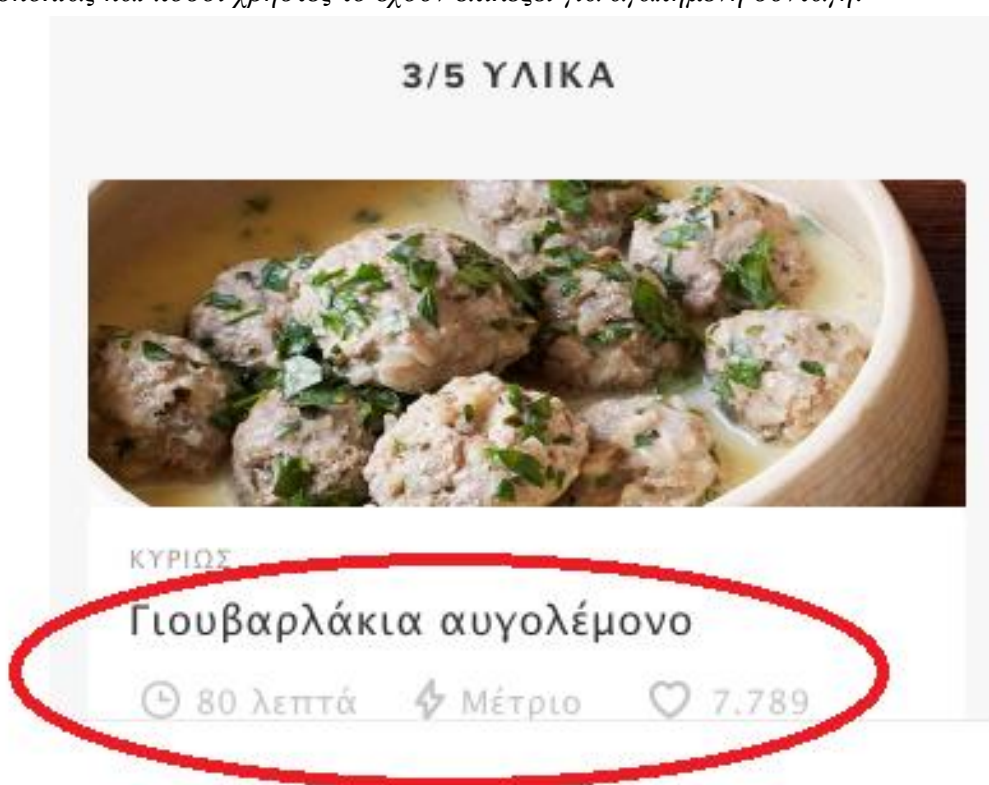
Εικόνα
υλικά που
από την
συνταγή
κάτω από
εικόνα της
συνταγής



Εικόνα
20:Είδο
ς
γέυματο
ς
συνταγή

ς

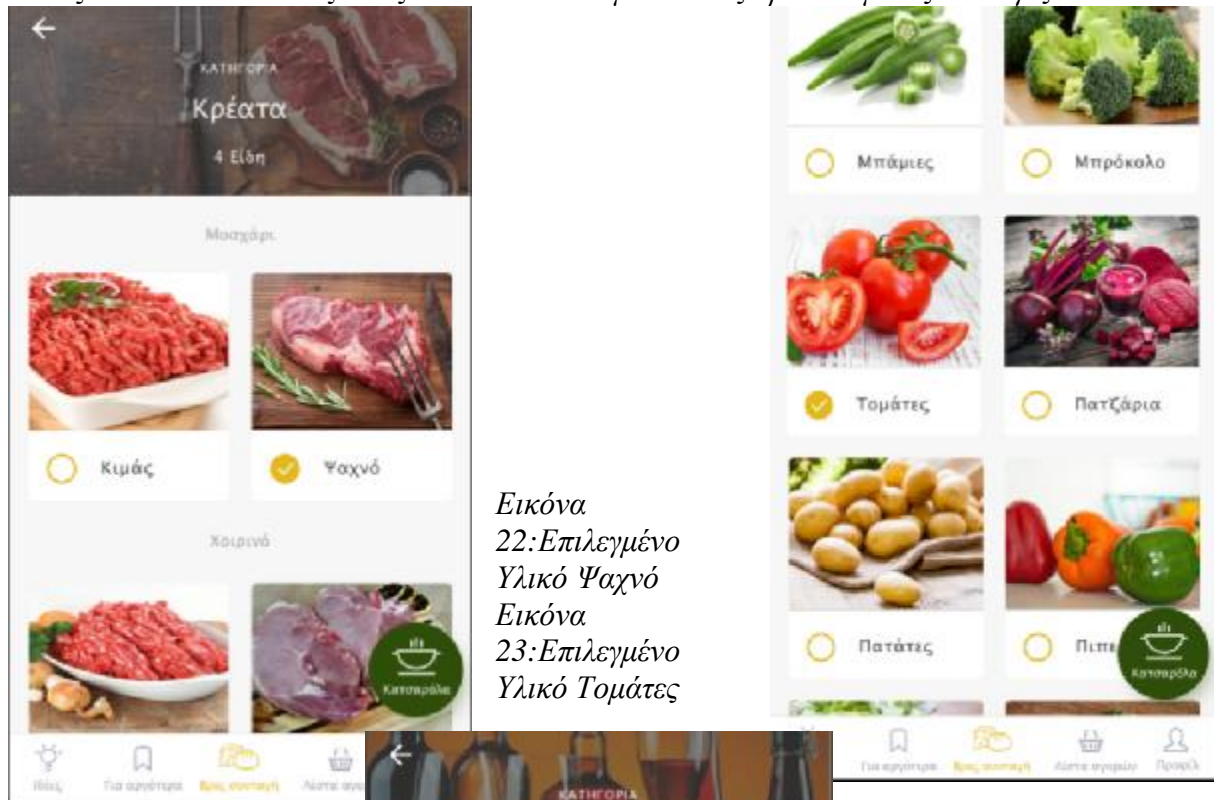
Και τέλος η ονομασία της και κάτω από την ονομασία ο χρόνος μαγειρέματος ο βαθμός δυσκολίας και πόσοι χρήστες το έχουν επιλέξει για αγαπημένη συνταγή.



Εικόν
α
21:Ον
ομασί
α
Συντα
γής
Χρόνο

ς ,Δυσκολία ,Αγαπημένη συνταγή

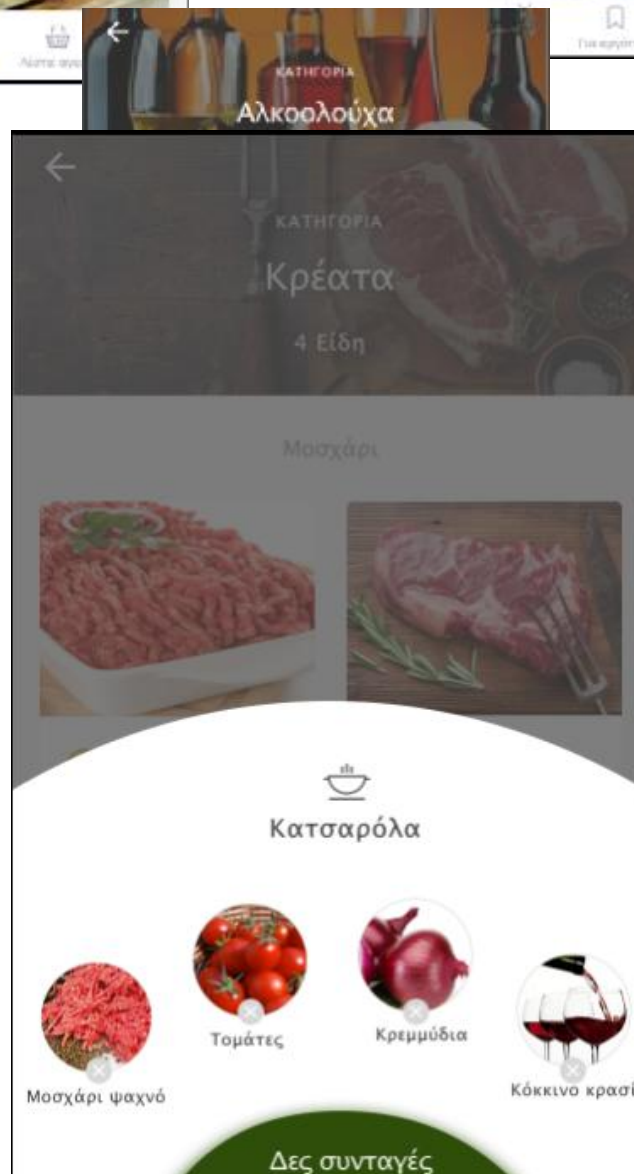
Αν ο χρήστης δεν είναι ικανοποιημένος από τις συνταγές που του έχουν προταθεί με τα υλικά που έχει επιλέξει μπορεί να επιστρέψει στην αρχική βασική κατηγορία υλικών και να επιλέξει άλλα υλικά αλλάζοντας και το αποτέλεσμα από τις προτεινόμενες συνταγές



Εικόνα 22:Επιλεγμένο Υλικό Ψαχνό
Εικόνα 23:Επιλεγμένο Υλικό Τομάτες

24:Επιλεγμένο Κρασί

Τα νέα υλικά που εμφανίσουν άλλες περιηγηθεί. Ελέγχει το αν έχει επιλέξει τα έχει στην διάθεση

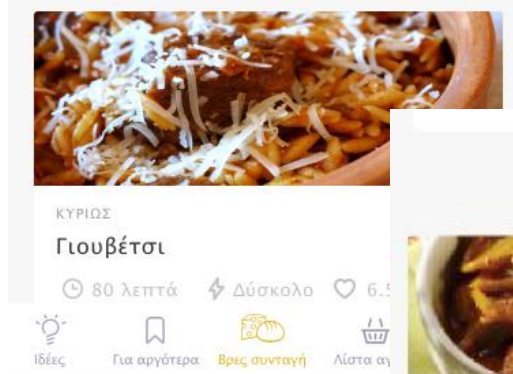
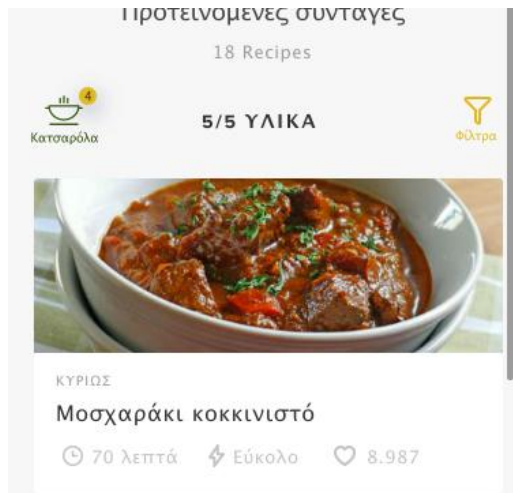


Εικόνα 25:Επιλεγμένο Υλικό Κόκκινο

θα επιλέξει θα του συνταγές για να

καλάθι/κατσαρόλα υλικά που θέλει και του.

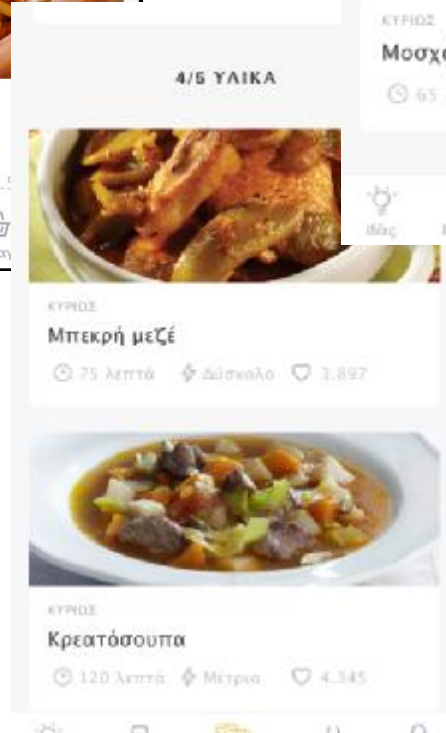
Εικόνα 25 Νέα επισκόπηση κατσαρόλας με νέα υλικά



Εικόνα 27: Νέα



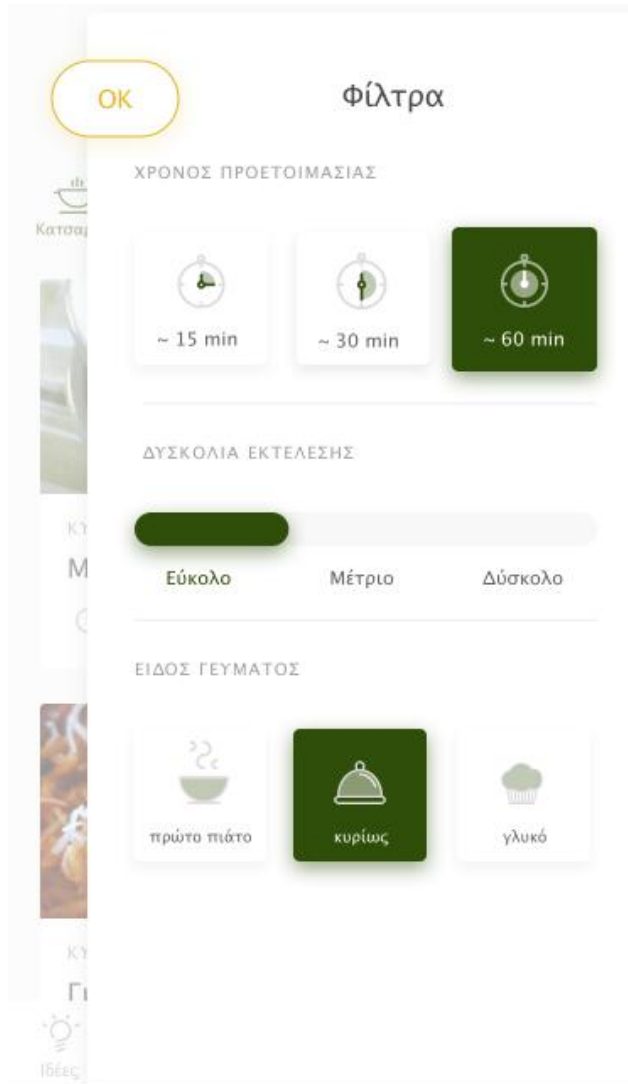
Εικόνα 26: Νέα



αποτελέσματα συνταγών
αποτελέσματα συνταγών

Εικόνα 28: Νέα αποτελέσματα συνταγών

Ύστερα από κάποια σχόλια που λάβαμε υπόψη από τους χρήστες που δοκίμασαν την εφαρμογή αποφασίσαμε να δώσουμε στον χρήστη την δυνατότητα να χρησιμοποιήσει και κάποια φίλτρα που θα τον βοηθήσουν να αποκλείσει μερικές από τις εμφανιζόμενες συνταγές προς διευκόλυνσή του.



Εικόνα 29: Φίλτρα

συνταγών

Αυτά τα με τον: Χρόνο που διάθεση του, εμφανίζονται έχουν τον χρόνο που Βαθμό

φίλτρα έχουν να κάνουν

έχει ο χρήστης στη με αυτόν τον τρόπο ΜΟΝΟ οι συνταγές που επιλεγμένο χρόνο έως και επιλέξαμε δυσκολίας εμφανίζοντας

ΜΟΝΟ τις συνταγές με τον επιλεγμένο βαθμό δυσκολίας Και τέλος με το είδος γεύματος εμφανίζοντας ΜΟΝΟ το είδος γεύματος που έχουμε επιλέξει. Μόλις ο χρήστης κάνει όλες τις επιλογές που επιθυμεί και έχει επιλέξει τα υλικά που έχει διαθέσιμα και έχει περιηγηθεί στις συνταγές τότε μπορεί να επιλέξει αυτή τη συνταγή που θέλει για να συνεχίσει στην επόμενη οθόνη.

επιλεγμένης

Αφού ο χρήστης συνταγή και ανοίξει κάποια στοιχεία. Για να κάνουμε πιο εφαρμογή λεπτομέρειες για να κάποιες ανάγκες που πρέπει να καλυφθούν

σχόλια των χρηστών που την δοκίμασαν.

Στο πάνω μέρος της σελίδας εμφανίζεται ένα βίντεο με το φαγητό σε γρήγορο τέμπο δείχνοντας στον χρήστη την διαδικασία και το τελικό αποτέλεσμα.

Πιο κάτω υπάρχει ένα κείμενο με πληροφορίες σχετικές με την συνταγή που έχουμε επιλέξει, και διάφορες λεπτομέρειες που προσφέρουν κάποιες απλές γνώσεις στον χρήστη.

Από κάτω έχουμε προσθέσει την δυνατότητα στον χρήστη της εφαρμογής μας να επιλέξει αν θέλει να μοιραστεί την συνταγή κάνοντας **SHARE**

Να την σχολιάσει αφήνοντας κάποιο σχόλιο **COMMENT**

Και να την βαθμολογήσει αν θέλει με την επιλογή **RATE**



Μοσχάρι κοκκινιστό

🕒 75 λεπτά ⚡ Μέτριο ❤️ 6.567

Το κοκκινιστό μοσχάρι είναι ένα παραδοσιακά γιορτινό, Κυριακάτικο πιάτο στην Ελληνική γαστρονομική παράδοση. Συνδυασμένο με μακαρόνια, με πατάτες τηγανητές, με πουρέ, με ρύζι, με κριθαράκι ως γιουβέτσι ή με πουρέ μελιτζάνας ως χουνκιάρ μπεγεντί αποτελεί μια σταθερή βάση για πολλά αγαπημένα γεύματα.

SHARE | **COMMENT** | **RATE**



Ιδέες



Για αργότερα



Βρες συνταγή



Λίστα αγορών



Προφίλ

Εικόνα 30:Εικόνα Συνταγής

“κλικάρει” την η σελίδα βλέπουμε

ολοκληρωμένη την προσθέσαμε κάποιες ικανοποιήσουμε παρατηρήσαμε ότι σύμφωνα με τα

Τα σχόλιά σας για το πιάτο



Αλεξία Αλεξιάδου

Καλαμαριά, Θεσσαλονίκη

“Έφτιαξα το κοκκινιστό σας χθες με τη διαφορά ότι δεν χρησιμοποίησα μοσχάρι αλλά χοιρινό (φιλετάκι από ψαρονέφρι). Το καταευχαριστήθηκαμε!!”



ΥΛΙΚΑ

ΕΚΤΕΛΕΣΗ

2 μερίδες

1 κούπα κρασί

ΥΛΙΚΑ

ΕΚΤΕΛΕΣΗ

2 μερίδες

- 1 κούπα κρασί
- 1 κ.σ. πελτέ τομάτας
- 1 κονσέρβα τοματάκι
- 1 κρεμμύδι φιλοκομμένο
- 1/2 κούπα βούτυρο
- 1/2 κιλό κρέας μοσχαρίσιο
- 2 σκελίδες σκόρδο
- 100 γρ. αλάτι
- 2 κ.γ. πιπέρι
- 1 κ.γ. μπαχάρι



Σημείωσε τα υλικά που λείπουν

περιήγηση
Αμέσως πιο
αφήσεις το
διαβάσεις τα
είχαν επιλέξει
συνταγή.
Αφού έχει
συνταγή, έχει
διαβάσει τα
χρηστών και
αν επιθυμεί την
μπορεί να
ολοκλήρωση της.
κουτάκι υλικά

Εικόνα 30: Συνταγή

κάτω υπάρχει ο χώρος να
σχόλιο του και να
σχόλια άλλων χρηστών που
και μαγείρεψαν την

επιλέξει την
σχολιάσει, έχει
σχόλια των άλλων
έχει βαθμολογήσει
συνταγή, τότε
συνεχίσει προς την
Επιλέγοντας το



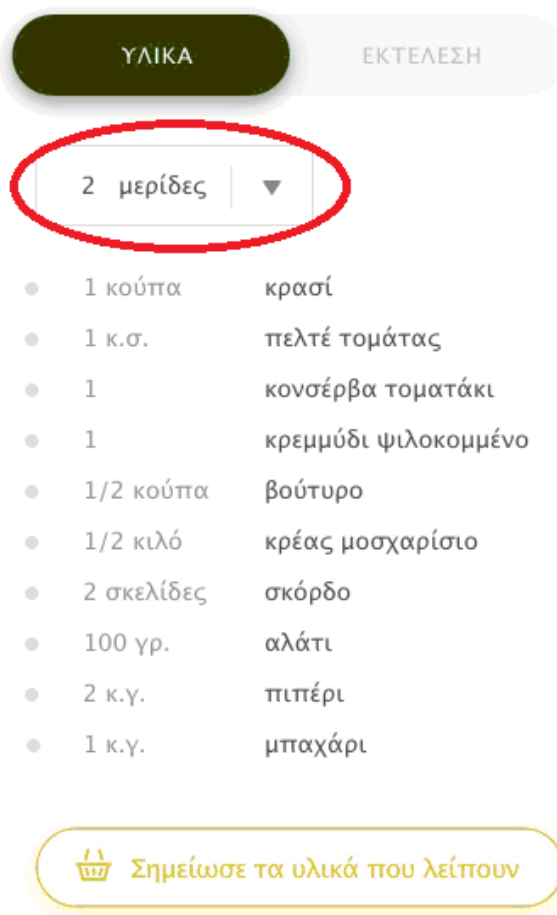
Εικόνα 31:Επιλογή ΥΛΙΚΑ

Του δίνεται αρχικά η επιλογή να επιλέξει για πόσες μερίδες θέλει να μαγειρέψει.

Έτσι ο χρήστης της εφαρμογής ανάλογα με τις μερίδες που επιλέγει του εμφανίζονται και οι κατάλληλες δοσολογίες των υλικών που έχει επιλέξει να χρησιμοποιήσει.

Με αυτόν τον τρόπο δεν χαραμίζονται αχρείαστες ποσότητες υλικών για την ολοκλήρωση της συνταγής και ο χρήστης μαγειρεύει χωρίς να χρειάζεται χωρίς

Παρατηρήσαμε φορές ο χρήστης υλικά για να συνταγή



ΥΛΙΚΑ ΕΚΤΕΛΕΣΗ

2 μερίδες ▼

- 1 κούπα κρασί
- 1 κ.σ. πελτέ τομάτας
- 1 κονσέρβα τοματάκι
- 1 κρεμμύδι φιλοκομμένο
- 1/2 κούπα βούτυρο
- 1/2 κιλό κρέας μοσχαρίσιο
- 2 σκελίδες σκόρδο
- 100 γρ. αλάτι
- 2 κ.γ. πιπέρι
- 1 κ.γ. μπαχάρι

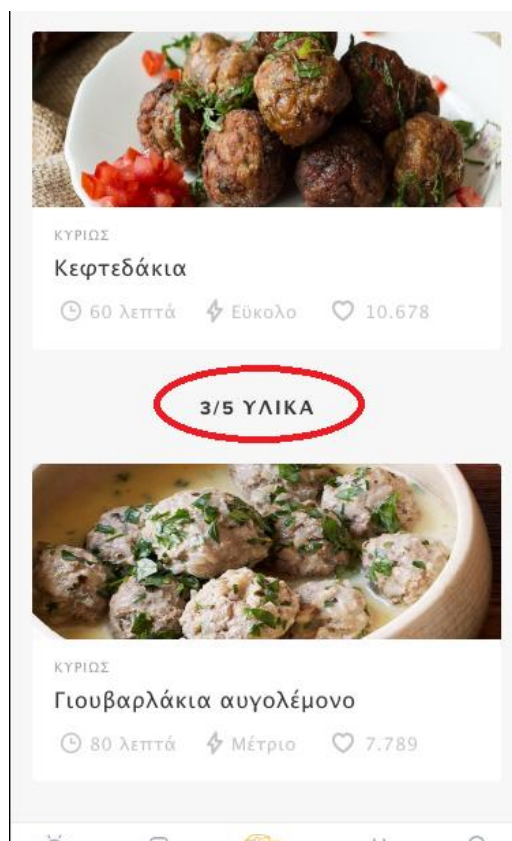
🛒 Σημείωσε τα υλικά που λείπουν

όμως ότι πολλές είχε ελλείψεις στα ολοκληρώσει μια

Εικόνα 32 :Επιλογή μερίδων

Όπως είδαμε στην εικόνα της περιήγησης των συνταγών κάποιες συνταγές είχαν τον αριθμό των υλικών που είχαμε προς το σύνολο των υλικών που χρειαζόμασταν για να ετοιμάσουμε την επιλεγμένη συνταγή.

Έτσι σκεφτήκαμε να προσθέσουμε την επιλογή στον χρήστη να μπορεί να προσθέσει τα υλικά που του λείπουν και να τα σημειώσει για να μπορεί να πραγματοποιήσει όποια συνταγή θέλει χωρίς περιορισμούς



Εικόνα 32:Υλικά που λείπουν από τις συνταγές



Σκεύη που θα χρειαστείς

- Χύτρα ή μεγάλη κατσαρόλα
- Ξύλινη κουτάλα
- Επιφάνεια κοπής
- Μαχαίρι
- Δοσομετρητή υγρών

Διατροφική αξία ανά μερίδα



Σκεύη που θα χρειαστείς

- Χύτρα ή μεγάλη κατσαρόλα
- Ξύλινη κουτάλα
- Επιφάνεια κοπής
- Μαχαίρι
- Δοσομετρητή υγρών

Διατροφική αξία ανά μερίδα



Υλικών που λείπουν

Ύστερα ο χρήστης της πληροφορείτε από την θα χρειαστεί να έχει για την ολοκλήρωση επιλέξει

Εικόνα 33:Σημείωση

εφαρμογής εφαρμογή για τα σκεύη και να χρησιμοποιήσει της συνταγής που έχει

Εικόνα 34:Σκεύη που θα χρειαστούν

μερίδα.Υστερα έχουμε επιλέξει να εμφανίζεται η διατροφική αξία ανά μερίδα

- Θερμίδες
- Πρωτεΐνη
- Λιπαρά
- Υδατάνθρακες

Με αυτόν τον τρόπο ο κάθε φορά τι του που έχει επιλέξει να

Αυτή η επιλογή κυρίως για χρήστες διατροφή τους.

🛒 Σημείωσε τα υλικά που λείπουν

Σκεύη που θα χρειαστείς

Χύτρα ή μεγάλη κατσαρόλα
Ξύλινη κουτάλα
Επιφάνεια κοπής
Μαχαίρι
Δοσομετρητή υγρών

Διατροφική αξία ανά μερίδα

Θερμίδες 642 Kcal	Πρωτεΐνη 39 g	Λιπαρά 24 g	Υδατ/κες 64 g
----------------------	------------------	----------------	------------------

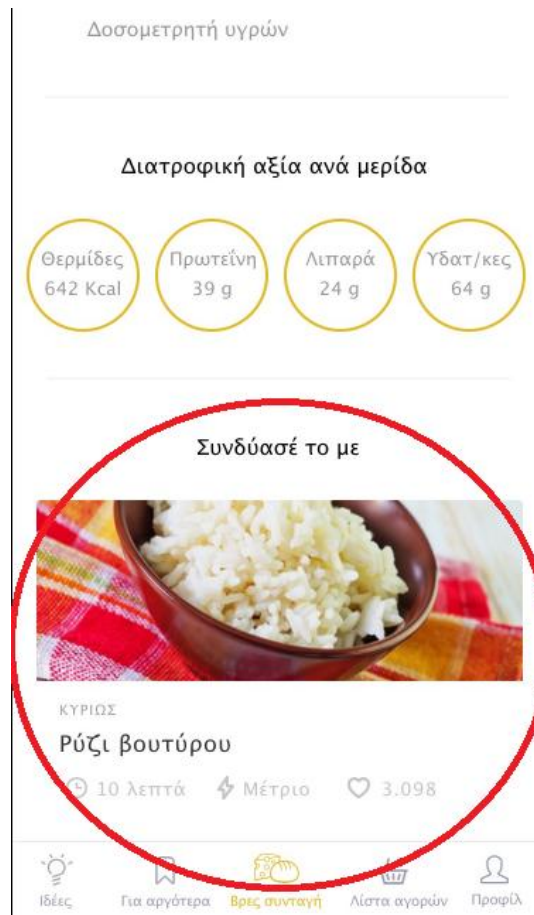
χρήστης γνωρίζει προσφέρει το φαγητό ετοιμάσει.

δημιουργήθηκε που προσέχουν την

Εικόνα 35:Διατροφική αξία ανά

Στο κάτω μέρος της σελίδας έχουμε επιλέξει να εμφανίζονται κάποιες επιλογές με το τί θα μπορούσε να συνοδεύσει ο χρήστης την συνταγή που έχει επιλέξει.

Αν ο χρήστης συμφωνεί με αυτή την πρόταση, μπορεί κλικάροντας πάνω στην προτεινόμενη συνταγή για συνοδευτικό να οδηγήσει αμέσως στην σελίδα της συνταγής του συνοδευτικού με πάλι τις ίδιες δυνατότητες και επιλογές που είχε και στην συνταγή που επιλέγει.

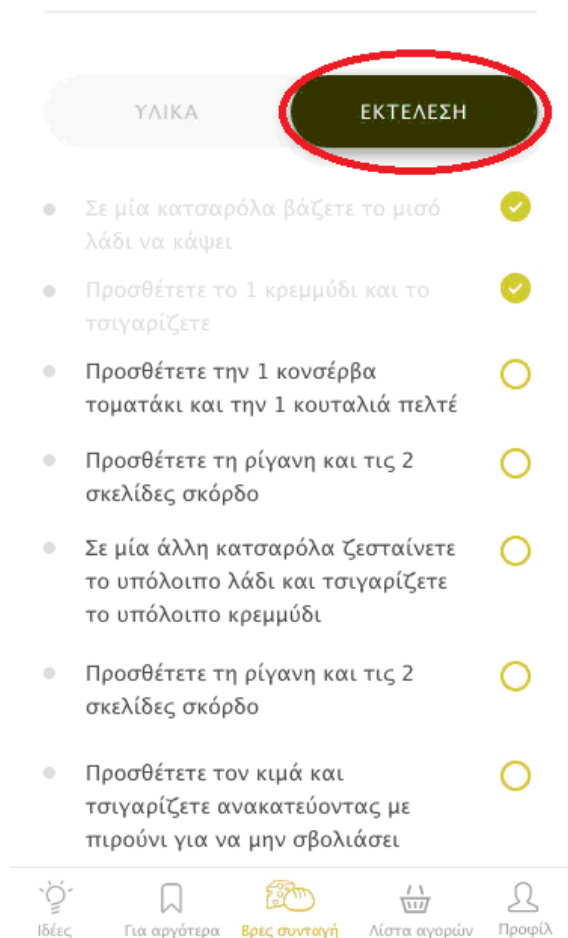


Εικόνα 36:Πρόταση για συνοδευτικό με την συνταγή

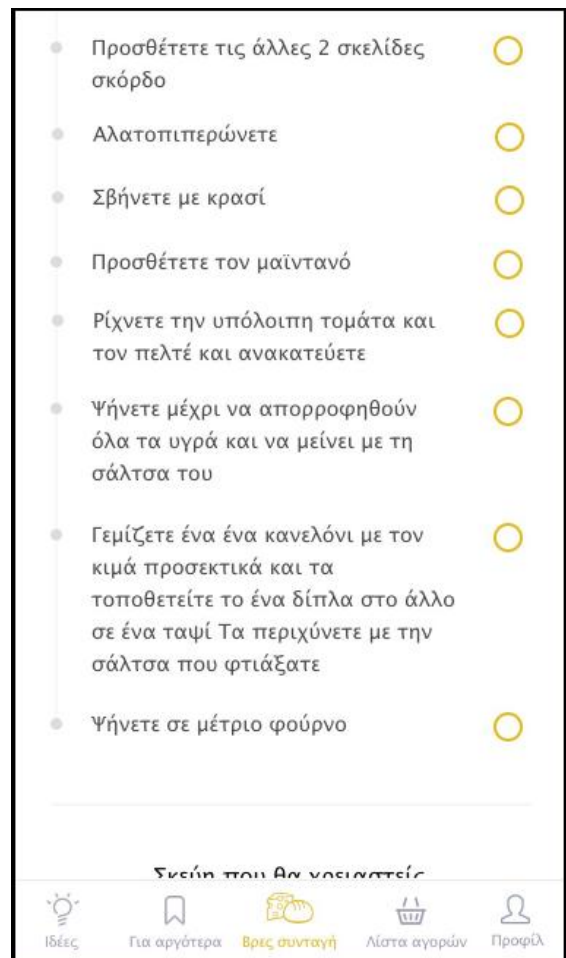
Επειδή η εφαρμογή έχει και εκπαιδευτικό χαρακτήρα θέλαμε να βρούμε έναν τρόπο έτσι ώστε ο χρήστης μέσα από όλη αυτή την διαδικασία να μαθαίνει να μαγειρεύει και να αρχίζει σιγά σιγά να εξοικειώνεται με την μαγειρική

Θεωρήσαμε λοιπόν ότι ο πιο εύκολος τρόπος να γίνει αυτό θα ήταν μέσα από πολύ απλά και κατανοητά βήματα

Έτσι ο χρήστης αφού έχει επιλέξει την συνταγή και είναι έτοιμος να ξεκινήσει κλικάρει την επιλογή εκτέλεση που βρίσκεται δίπλα από τα υλικά.Με αυτόν τον τρόπο ανοίγει ένα άλλο παράθυρο



Εικόνα 38:Εκτέλεση συνταγής σε βήματα



Εικόνα 37:Εκτέλεση συνταγής σε βήματα

Σε αυτή την επιλογή ο χρήστης ακολουθάει αναλυτικά τα βήματα και κάθε φορά που τελειώνει ένα βήμα κλικάρει το κυκλάκι που βρίσκεται δεξιά και το βήμα σταματάει να είναι ενεργό.

Με αυτόν τον τρόπο ο χρήστης δεν χάνει την επαφή με την συνταγή και τα βήματα της και μπορεί ανά πάσα ώρα και στιγμή να βρει σε ποιο βήμα βρίσκεται μέχρι την τελική ολοκλήρωση της συνταγής.

Μόλις ολοκληρώσει τα βήματα και φέρει εις πέρας την συνταγή μπορεί να αφήσει τα σχόλια του, τις εντυπώσεις του, τις ιδέες του και ότι άλλο επιθυμεί στις δυνατότητες που του παρέχονται γενικά από την εφαρμογή μας.

Ύστερα μπορεί ή να τερματίσει την εφαρμογή ή να συνεχίσει σε νέα αναζήτηση.

ΣΥΜΠΕΡΑΣΜΑΤΑ

Η δημιουργία ενός εγχειριδίου χρήσης της εφαρμογής είναι ένα πολύ σημαντικό κομμάτι τόσο για τον χρήστη όσο και για τον προγραμματιστή που θα αναλάβει στην συνέχεια να υλοποιήσει την σχεδιασμένη εφαρμογή ικανοποιώντας τις απαιτήσεις και στόχους που έχουμε θέσει. Όσον αφορά τον χρήστη, βοηθά στην εκμάθηση ενός καλύτερου χειρισμού της εφαρμογής, εξοικειώνεται με τον τρόπο λειτουργίας της εφαρμογής και του προσφέρονται λύσεις σε τυχόν προβλήματα που ενδεχομένως να προκύψουν κατά την χρήση της. Από την μεριά του προγραμματιστή τον βοηθά να αξιολογήσει καλύτερα την λειτουργικότητα και την ευχρηστία της εφαρμογής που έχουμε σχεδιάσει με σκοπό να αποδώσει όσο πιο κοντά γίνεται, αν όχι επακριβώς, την υλοποίηση της εφαρμογής

Το συμπέρασμα από όλη τη διαδικασία αυτή της εύρεσης της ιδέας που θα εξελίξασαμε, της κατανόησης της ιδέας, της μελέτης, της επεξεργασίας απαιτήσεων και τελικώς του σχεδιασμού ήταν ότι δεν είναι καθόλου απλή διαδικασία καθώς απαιτεί εξειδικευμένες γνώσεις πάνω σε θέματα σχεδιασμού και υλοποίησης μιας εκπαιδευτικής εφαρμογής για κινητές συσκευές πλατφόρμας iOS. Για την απόκτηση των γνώσεων αυτών απαιτήθηκε πολύς χρόνος και ταυτόχρονα διαρκής αναζήτηση για την επίλυση διάφορων προβλημάτων. Πιο συγκεκριμένα, κατά την διάρκεια εκπόνησης της πτυχιακής εργασίας παρουσιάζονταν αρκετά λογικά λάθη τα οποία οφείλονταν τόσο στην έλλειψη εμπειρίας όσο και λανθασμένων χειρισμών.

Η εμπειρία που αποκτήθηκε κατά την διάρκεια εκπόνησης της πτυχιακής είναι ένα πολύ σημαντικό εφόδιο για την έναρξη της επαγγελματικής μας σταδιοδρομίας. Η συνεχής ενασχόληση με τα απαραίτητα εργαλεία σχεδιασμού για την δημιουργία της εφαρμογής, κατά την προσωπική μας άποψη μας δικαίωσε διότι έπειτα από έξι μήνες συνεχούς μελέτης, εξάσκησης αλλά και συζήτησης με πιο έμπειρα άτομα μας βοήθησαν ώστε να ολοκληρωθεί με επιτυχία.

Η εκμάθηση των εργαλείων αυτών που χρησιμοποιήσαμε κατά την σχεδίαση και υλοποίηση της εκπαιδευτικής εφαρμογής είτε σε μεγάλο βαθμό είτε σε μικρότερο μας έδωσε την δυνατότητα να ασχοληθούμε με έναν αρκετά ενδιαφέρον τομέα που ανθίζει και μπορεί να μας προσφέρει μελλοντική επαγγελματική αποκατάσταση, τον τομέα του digital design.

BIBΛΙΟΓΡΑΦΙΑ

Michaelis, Spokas, January 2002, Εισαγωγή στον Προγραμματισμό με την C#

Νικόλαος Μισυρλής, Edition 3, 2007, Εισαγωγή στον Προγραμματισμό με την C

Rob Miles, Edition 2.1 January 2011, C# Programming

Microsoft - Wikipedia, the free encyclopedia. (n.d.). Retrieved April 26, 2016, from <http://en.wikipedia.org/wiki/Microsoft>

Microsoft Visual Studio - Wikipedia, the free encyclopedia. (n.d.). Retrieved April 26, 2016, from http://en.wikipedia.org/wiki/Microsoft_Visual_Studio

Visual Studio - Home. (n.d.). Retrieved from <http://www.visualstudio.com/>

Application Development. (n.d.). Retrieved from <http://www.visualstudio.com/engb/explore/application-development-vs>

Microsoft Blend - Wikipedia, the free encyclopedia. (n.d.). Retrieved April 26, 2016, from http://en.wikipedia.org/wiki/Microsoft_Blend

InVision -. Retrieved April 26, 2016, from <http://www.howdesign.com/web-design-resources-technology/ux-and-ui-design/invision-app/>

InVision-Retrieved April 29, 2016, from <https://reviews.financesonline.com/p/invision/>

Visual C# Resources. (n.d.). Retrieved from <http://msdn.microsoft.com/enus/vstudio/hh341490.aspx>

C# Tutorials (C#). (n.d.). Retrieved from [http://msdn.microsoft.com/enus/library/aa288436\(v=vs.71\).aspx](http://msdn.microsoft.com/enus/library/aa288436(v=vs.71).aspx)

Visual Basic - Wikipedia, the free encyclopedia. (n.d.). Retrieved April 26, 2016, from http://en.wikipedia.org/wiki/Visual_Basic

Visual Basic Wiki. (n.d.). Retrieved April 26, 2016, from http://vb.wikia.com/wiki/Main_Page