



ΧΑΡΟΚΟΠΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΜΑΤΙΚΗΣ

Πτυχιακή εργασία

Ανάλυση μηχανισμών διαχείρισης ροών εργασίας διαδικτυακών υπηρεσιών

Όλγα-Αγγελική Κυριακούλη

Επιβλέπων Καθηγητής:

Τσερπές Κωνσταντίνος

Μέλη:

Νικολαΐδου Μάρα

Βαρλάμης Ηρακλής

Καλλιθέα, Ιούλιος 2013

ΠΕΡΙΛΗΨΗ

Η παρούσα διπλωματική εργασία έχει ως στόχο να καταδείξει τις δυνατότητες που διαθέτει ο συνδυασμός των μέσων κοινωνικής δικτύωσης με τη χρήση του Διαδικτύου ως πλατφόρμα εφαρμογών. Η εξέλιξη των τεχνολογιών που βρίσκουν εφαρμογή στο Διαδίκτυο έχει αλλάξει τον τρόπο που αντιμετωπίζονται οι χρήστες. Από παθητικοί αποδέκτες της πληροφορίας έχουν μετατραπεί πλέον σε κοινωνούς και δημιουργούς της πληροφορίας με αποτέλεσμα την πλήρη αναμόρφωση του τοπίου και του περιεχομένου του Διαδικτύου. Στη διαμόρφωση των αλλαγών αυτών έπαιξαν σημαντικό ρόλο οι εξελίξεις της τεχνολογίας. Το web 2.0 και ο προσανατολισμός του σχεδιασμού προς το περιεχόμενο που παρέχεται από τους χρήστες (User Generated Content), ήταν οι πιο καθοριστικοί παράγοντες.

Στην εργασία αυτή παρουσιάζουμε τις τεχνολογίες που αποτελούν το Web 2.0 εστιάζοντας στην μελέτη ενός εκ των δημοφιλέστερων εκπροσώπων του, των Mashups αναλύοντας τις τεχνολογίες τους καθώς επίσης και γνωστές εφαρμογές τους. Στη συνέχεια υλοποιούμε μια λύση για ένα πραγματικό σενάριο αναζήτησης συγκεκριμένων πληροφοριών από social networking sites, μέσα από την οποία προβάλλουμε τα εργαλεία και τις εφαρμογές των παραπάνω τεχνολογιών, επιδεικνύοντας το πώς μπορούν να προσφέρουν στη νέα μορφή περιεχομένου. Τέλος, αξιολογούμε την προσπάθειά μας, καταγράφουμε τα αποτελέσματα και τους στόχους που υλοποιήθηκαν και παραθέτουμε τα συμπεράσματα και τις προτάσεις μας.

Abstract

This thesis aims to illustrate the possibilities offered by the combination of social media by using the Internet as a platform for applications. The evolution of technologies on the Internet has changed the way users are treated. From passive recipients of information have now become communicants and creators of information resulting in a complete overhaul of the landscape and Internet content. In shaping these changes played an important role developments in technology. The web 2.0 and the orientation of the design to the content provided by users (User Generated Content), were the most decisive factors.

In this paper we present the technologies that make up Web 2.0 focusing on the study of one of the most popular representatives of it, mashups analyzing their technologies as well-known applications. Then we implement a solution of a real scenario of searching information from social networking sites introducing tools and applications of these technologies and demonstrating how they can offer to the new format content. Finally, we evaluate our efforts, we record the results and goals that have been implemented and we present the conclusions and suggestions.

Περιεχόμενα

1	ΕΙΣΑΓΩΓΗ	1
2	ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ	5
2.1	Web 2.0.....	5
2.1.1	Ορισμός.....	5
2.2	Εφαρμογές του Web 2.0	6
2.2.1	Ιστολόγια (Blogs).....	7
2.2.2	Wikis	7
2.2.3	Κοινωνική Σήμανση (Social Tagging).....	8
2.2.4	Ιστότοποι Κοινωνικής Δικτύωσης (Social Networking Sites).....	8
2.2.5	RSS (Real Simple Syndication) feeds	9
2.2.6	Mashups	9
2.2.7	Άλλες Web 2.0 εφαρμογές	9
2.3	Τεχνολογίες του Web 2.0	10
2.4	MashUps.....	12
2.4.1	Ορισμός.....	12
2.4.2	Η αρχιτεκτονική των mashups	13
2.4.3	Είδη Mashups	17
2.4.4	Mashups tools.....	19
3	ΠΕΙΡΑΜΑΤΙΚΗ ΠΡΟΣΕΓΓΙΣΗ.....	22
3.1	Περιγραφή της mashup εφαρμογής.....	22
3.2	Υλοποίηση με το Yahoo! Pipes.....	26
3.2.1	Περιγραφή πλατφόρμας	26
3.2.2	Περιγραφή υλοποίησης	32
3.3	Υλοποίηση με το WebHookit.....	38
3.3.1	Περιγραφή πλατφόρμας	38
3.3.2	Περιγραφή υλοποίησης	43
4	ΣΥΓΚΡΙΤΙΚΗ ΑΞΙΟΛΟΓΗΣΗ	49

4.1	Σύγκριση Yahoo! Pipes & WebHookIt	49
4.1.1	Κριτήρια	49
4.1.2	Αξιολόγηση	50
4.2	Σύγκριση υλοποίησης mashup με τη βοήθεια κάποιου εργαλείου ανάπτυξης mashup σε σχέση με την ανάπτυξη «με το χέρι»	54
5	ΣΥΜΠΕΡΑΣΜΑΤΑ & ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ.....	56
5.1	ΣΥΜΠΕΡΑΣΜΑΤΑ	56
5.2	ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ	57
6	Βιβλιογραφία	58

Πίνακας Εικόνων και Πινάκων

Εικόνα 2-2-1: Έννοιες συσχετιζόμενες με το Web 2.	5
Εικόνα 2-2: Κατηγορίες και εκπρόσωποι εφαρμογών Web 2.	10
Εικόνα 2-3: Κατηγοριοποιημένες τεχνολογίες χρησιμοποιούμενες από Mashups.....	14
Εικόνα 2-4: Κατανομή των πιο δημοφιλών web APIs.....	14
Εικόνα 2-5: Βασικά πρότυπα ανάπτυξης mashup, (α) το slave-slave, (b) το master-slave, και (c) το master-master μοτίβο.....	17
Εικόνα 2-6: Κατανομή των διάφορων ειδών mashups τις τελευταίες 14 ημέρες.....	18
Εικόνα 2-7: Γενική κατανομή των διάφορων ειδών mashups	19
Εικόνα 3-1: Η επιφάνεια επεξεργασίας του Yahoo! Pipes αποτελείται από τέσσερις περιοχές: την γραμμή πλοήγησης, την εργαλειοθήκη, το καμβά και το πάνελ εντοπισμού σφαλμάτων και παραγωγής εξόδου	27
Πίνακας 3-1: operators της ροής δεδομένων που προσφέρονται από το Yahoo Pipes	30
Πίνακας 3-2: operators της ροής δεδομένων που προσφέρονται από το Yahoo Pipes	31
Εικόνα 3-2: Υλοποίηση της εφαρμογής στο Yahoo! Pipes	33
Εικόνα 3-3: Αποτελέσματα της εκτέλεσης της εφαρμογής	33
Εικόνα 3-4: Υλοποίηση του Instagram. Μέρος της εφαρμογής στο Yahoo! Pipes	35
Εικόνα 3-5: Υλοποίηση του Picasa μέρους της εφαρμογής στο Yahoo! Pipes	36
Εικόνα 3-6: Υλοποίηση του Flickr μέρους της εφαρμογής στο Yahoo! Pipes.....	37
Εικόνα 3-7: Το περιβάλλον του editor του WebHookIt.....	40
Πίνακας 3-3: Modules δημιουργίας mashup εφαρμογών του WebHookIt.....	41

Εικόνα 3-8: Διαχείριση πακέτων της εφαρμογής.....	42
Εικόνα 3-9: Εκτελέσεις δημιουργημένων εφαρμογών του WebHookIt	43
Εικόνα 3-10: Κατάλογος υπαρχόντων εφαρμογών για περαιτέρω επεξεργασία και εκτέλεσή τους.....	43
Εικόνα 3-11: Η υλοποίηση της εφαρμογής στο WebHookIt	45
Εικόνα 3-12: Η σελίδα εκτέλεσης της εφαρμογής στο WebHookIt	45
Εικόνα 3-13: Αποτελέσματα εκτέλεσης της εφαρμογής στο WebHookIt	46
Εικόνα 3-14: Υλοποίηση του picasa module-κομματιού της εφαρμογής στο WebHookIt	47
Εικόνα 3-15: Υλοποίηση του flickr module-κομματιού της εφαρμογής στο WebHookIt.....	48
Πίνακας 4-1: Υποστηριζόμενες τεχνολογίες των Yahoo! Pipes και WebHookIt....	511
Πίνακας 4-2Σύγκριση Yahoo! Pipes και WebHookIt.....	533

1 ΕΙΣΑΓΩΓΗ

Στα χρόνια που ακολούθησαν την έκρηξη της φούσκας του Ίντερνετ το 2001, το διαδίκτυο μετατράπηκε σε ένα κατά πολύ πιο κοινωνικά προσανατολισμένο μέρος. Μια νέα κουλτούρα του Διαδικτύου, συχνά αναφερόμενη ως «Web 2.0», έχει αναδειχθεί ως ένα νέο πρότυπο. Στο Web 2.0, το διαδίκτυο λειτουργεί ως μια πλατφόρμα λογισμικού για την ανοικτή ανταλλαγή πληροφοριών[1]. Αντί για δικτυακούς τόπους όπως ο Britannica Online[2], η νέα τάση της «σοφίας του πλήθους» προωθεί την ανάπτυξη των sites, όπως η Wikipedia[3]. Επιπλέον, η εισαγωγή των προσανατολισμένων στην παροχή υπηρεσιών διαδικτυακών επιχειρήσεων προωθεί συμπεριφορές, όπως το blogging και την ενεργή συμμετοχή του χρήστη του διαδικτύου. Σε αυτή τη νέα μορφή του διαδικτύου, ο καθένας μπορεί και πρέπει να δημιουργήσει online περιεχόμενο. Με τη βοήθεια των προγραμματιστών λογισμικού και σχεδιαστών οι οποίοι έχουν δημιουργήσει εργαλεία για την υποστήριξη αυτών των νέων τάσεων, δραστηριότητες που απαιτούσαν προηγουμένως γνώσεις προγραμματισμού μπορούν τώρα να ολοκληρωθούν με μικρή προσπάθεια. Για παράδειγμα, οι υπηρεσίες της Google[4] επιτρέπουν στους χρήστες να δημιουργήσουν μια δυναμική ιστοσελίδα μέσω ενός απλού interface, και παρέχουν free hosting για τις παραγόμενες ιστοσελίδες. Ενώ η δημιουργία web σελίδων είναι ένα σημαντικό βήμα, είναι μόνο ένα μικρό βήμα προς τη αναμόχλευση πόρων που έχουν γίνει διαθέσιμοι ως μέρος του Web 2.0. Η αυξανόμενη έμφαση σε online δημιουργία περιεχομένου έχει εμπνεύσει εταιρείες όπως η Google, Yahoo και Amazon να ανοίξουν τα συστήματα δεδομένων τους σε δημόσια χρήση, μέσω της εφαρμογή των διεπαφών προγραμματισμού εφαρμογών (APIs). Ένας χρήστης με δεξιότητες προγραμματισμού μπορεί να αξιοποιήσει αυτά τα APIs και άλλες πηγές δεδομένων για τη δημιουργία νέων εφαρμογών web, γνωστών ως mashups. Ένα mashup είναι ένας συνδυασμός των διαφορετικών API και δεδομένων σε μια ενιαία οπτική διεπαφή. Σύμφωνα με ProgrammableWeb, μια ομάδα που παρακολουθεί web-based APIs και mashups, στις 17 Μαΐου του 2013, ήταν διαθέσιμα 9141 API και 7061 mashups με αυτούς τους αριθμούς να αυξάνονται κατά 2 και κατά 20 ανά ημέρα, αντίστοιχα[5]. Αυτές οι στατιστικές δείχνουν ένα σαφές και αυξανόμενο ενδιαφέρον για τα web mashups. Ωστόσο, γνωρίζουμε πολύ λίγα προς το παρόν για την δυναμική των mashups ως στοιχείο του web προγραμματισμού.

Ο Προγραμματισμός είναι μια εγγενώς πολύπλοκη εργασία[6]. Η ικανότητα να σχεδιάσει κάποιος και να εφαρμόσει εφαρμογές πληροφορικής βασίζεται στην κατανόηση εννοιών και δεξιοτήτων της μηχανικής του λογισμικού (software engineering), καθώς και την εφαρμογή τους σε συγκεκριμένες γλώσσες προγραμματισμού. Η έρευνα στον τομέα που συχνά αποκαλείται

"End-User Programming" (EUP) στοχεύει στο να καταστεί ο προγραμματισμός περισσότερο προσιτός για άτομα που δεν είναι επαγγελματίες προγραμματιστές ή για άτομα που δεν διαθέτουν τις γνώσεις που απαιτούνται για να επιτευχθεί ένας υπολογιστικός-προγραμματιστικός στόχος. Ενώ πολλοί ερευνητές δουλεύουν ενεργά πάνω στα προβλήματα και τα ζητήματα του EUP, ο «κώδικας» που παράγεται από ανθρώπους οι οποίοι δεν είναι προγραμματιστές (nonprogrammers) εξακολουθεί να είναι συχνά γεμάτος λάθη[7]. Οι ερευνητές έχουν δημιουργήσει εργαλεία για να βοηθήσουν τους τελικούς χρήστες με την απλή ανάπτυξη διαδικτυακών δραστηριοτήτων[8], συμπεριλαμβανομένης της δημιουργίας και επεξεργασίας απλών βάσεων δεδομένων. Ενώ αυτά τα εργαλεία έχουν επιτυχία, το έργο της δημιουργίας ενός mashup είναι πιο περίπλοκο. Οι χρήστες δεν πρέπει μόνο να κατανοήσουν γενικές τεχνικές προγραμματισμού (π.χ., υπό όρους συνθήκες, επαναλήψεις), αλλά και να χειριστούν δεδομένα και να σχεδιάσουν την διεπαφή χρήστη. Ως αποτέλεσμα, ένα μεγάλο μέρος του πληθυσμού των nonprogrammers που θα μπορούσαν να ωφεληθούν από τη δημιουργία mashups αποκλείεται. Το παρακάτω σενάριο δείχνει μια περίπτωση όπου ένα mashup θα ήταν χρήσιμο στον τελικό χρήστη.

Ο Γιάννης είναι ένας φοιτητής στο πρώτο έτος. Έχει πολλούς φίλους από το λύκειο με τους οποίους θέλει να μένει σε επαφή. Ένας τρόπος για να το κάνει αυτό είναι να κάνει chat με αυτούς κάθε εβδομάδα μετά από αθλητικές εκδηλώσεις, ιδίως ποδοσφαιρικούς αγώνες. Για να καταστεί ευκολότερο για τους φίλους του να επικοινωνούν, αποφασίζει να δημιουργήσει ένα mashup που θα επιτρέψει στην παρέα να μιλούν ο ένας στον άλλο, αλλά στο πλαίσιο αθλητικών θεμάτων. Αυτό το mashup περιλαμβάνει feeds ειδήσεων από διαφορετικές αθλητικές πηγές, στατιστικά παικτών, και έναν χάρτη των ομάδων με τα παιχνίδια που παίζονται αυτή την εβδομάδα. Τέλος, προσθέτει, επίσης, ένα chat room, όπου μπορούν να συνομιλούν ζωντανά κατά τη διάρκεια ενός παιχνιδιού, ή απλά να αφήνουν σχόλια ο ένας για τον άλλον[9].

Μέχρι πρότινος, κάποιος μη προγραμματιστής δεν ήταν σε θέση να δημιουργήσει το mashup που περιγράφεται πιο πάνω καθώς η ανάπτυξη του απαιτούσε την εξοικείωση με εργαλεία (π.χ. Dreamweaver, Frontpage) τα οποία απαιτούσαν αναπτυγμένες προγραμματιστικές γνώσεις. Πλέον με την εξέλιξη των εργαλείων για τη δημιουργία mashups ο αποκλεισμός που περιγράψαμε παραπάνω τείνει να εκλείψει προσφέροντας στους χρήστες του διαδικτύου τρόπους πρόσμιξης πληροφοριών και δεδομένων από διάφορες διαδικτυακές πηγές συμπεριλαμβανομένων κατά κύριο όρο των social networking sites, που κυριαρχούν στις μέρες μας διευκολύνοντας τους.

Σκοπός της παρούσας εργασίας είναι να εξεταστεί η έννοια των mashups καθώς και των τεχνολογιών και των εφαρμογών που την περιβάλλουν καθώς επίσης να υπάρξει και μια πιο πρακτική εξέταση του θέματος μέσω πειραματισμού μας με κάποια εργαλεία ανάπτυξης τους. Πιο συγκεκριμένα θα εξεταστεί ένα ιδιαίτερα δημοφιλές εργαλείο ανάπτυξης mashups, το Yahoo Pipes! εν σύγκριση με ένα open-source project ονόματι webhookit ώστε να εξεταστούν τα πλεονεκτήματα και τα μειονεκτήματα της κάθε υλοποίησης οδηγώντας μας στην διαπίστωση του κατά πόσο μπορούν τα mashups να προσφέρουν συναθροίσεις μέσων και να μας προσφέρουν καλύτερη διαχείριση του διαδικτυακού περιεχομένου.

Κατά την επίλυση του προβλήματος αρχικά μελετήθηκε όπως προαναφέρθηκε η θεωρητική οπτική των mashups και όσων αυτά περιλαμβάνουν ώστε στη συνέχεια να υπάρχουν οι απαραίτητες γνώσεις για τον πειραματισμό με τα εργαλεία ανάπτυξης της πειραματικής μας εφαρμογής και να είναι δυνατή αργότερα η σύγκρισή τους καθώς επίσης να καταστεί δυνατό να δούμε κατά πόσο τα mashups είναι όντως απαραίτητα για την ορθή διαχείριση του περιεχομένου του διαδικτύου.

Η παρούσα πτυχιακή εργασία έχει οργανωθεί σε 5 κεφάλαια, τα οποία παρουσιάζονται συνοπτικά παρακάτω:

Το 1^ο κεφάλαιο είναι εισαγωγικό και παρουσιάζει το σύγχρονο περιβάλλον του διαδικτύου στα πλαίσια του οποίου αναπτύχθηκε η παρούσα διπλωματική, καθώς και το αντικείμενο το οποίο πραγματεύεται η διπλωματική αυτή.

Το 2^ο κεφάλαιο παρουσιάζει το θεωρητικό υπόβαθρο το οποίο μελετήθηκε στα πλαίσια αυτής της διπλωματικής. Εξετάζει τι είναι το Web 2.0 από ποιες τεχνολογίες αποτελείται και ποιες κατηγορίες εργαλείων αυτό περιλαμβάνει, εξετάζει τι είναι τα mashups και από ποιες τεχνολογίες αποτελούνται καθώς επίσης και αναφέρεται σε γνωστές πλατφόρμες-εργαλεία δημιουργίας mashups.

Το 3^ο κεφάλαιο παρουσιάζει την υλοποίηση της πειραματικής εφαρμογής mashup στα δυο επιλεγμένα εργαλεία. Αρχικά, εισάγει τον αναγνώστη στο τεχνολογικό υπόβαθρο που μελετήθηκε πριν ξεκινήσει η υλοποίηση αυτής της εφαρμογής, αναλύει τις αποφάσεις που

πήραμε για τα εργαλεία που χρησιμοποιούμε μετά από κατάλληλες συγκρίσεις και δίνει περαιτέρω λεπτομέρειες που αφορούν την υλοποίηση της εφαρμογής μας.

Το 4^ο κεφάλαιο αποτελείται από τα αποτελέσματα της σύγκρισης των δυο εργαλείων καθώς και της σύγκρισης της δημιουργίας ενός Mashup μέσω ενός εργαλείου ή δημιουργώντας το εξολοκλήρου «με το χέρι» ενώ τέλος, στο Κεφάλαιο 5 βρίσκονται τα συμπεράσματα στα οποία καταλήγει η μελέτη που διενεργήθηκε, ενώ προτείνονται και μελλοντικές προτάσεις για την επίλυση προκλήσεων στο μέλλον.

Στην προσπάθεια να οριοθετήσει τον όρο Web 2.0 ο Tim O'Reilly θεωρεί ότι ένας δικτυακός τύπος ή μια γενικότερη διαδικτυακή λύση για να χαρακτηρίζεται ως Web 2.0[10], θα πρέπει:

1. Να περιλαμβάνει υπηρεσίες με δυνατότητες επέκτασης, οι οποίες να είναι προσβάσιμες μέσω φυλλομετρητών ιστού (web browser) ανεξαρτήτως συσκευής πρόσβασης (π.χ. H/Y, PDA, κινητό τηλέφωνο) και λειτουργικού συστήματος. Η ύπαρξη σύνδεσης στο διαδίκτυο αποτελεί τη μόνη προϋπόθεση.
2. Οι υπηρεσίες που χρησιμοποιεί να μπορούν να ενσωματωθούν σε άλλες ιστοσελίδες ή να διαμοιραστούν σε σελίδες κοινωνικής δικτύωσης.
3. Να αποτελείται από «ελαφριά» τεχνολογία όσον αφορά τα πρωτόκολλα και τις γλώσσες προγραμματισμού και από φιλικές προς τον χρήστη εφαρμογές με τις οποίες να μπορεί ο τελευταίος να αλληλοεπιδρά. Επίσης, να προωθεί τον δημοκρατικό χαρακτήρα του διαδικτύου, δίνοντας στον χρήστη τον πρωταγωνιστικό ρόλο.
4. Να αξιοποιεί τη συλλογική νοημοσύνη με σκοπό τη συνεχή και άμεση ανανέωση των δεδομένων και του λογισμικού καθώς και την αποκέντρωση υπηρεσιών και προτύπων. Για παράδειγμα, μπορεί να δίνει την δυνατότητα σε κάθε χρήστη να επιλέξει υπερσυνδέσμους ή ετικέτες (tags) σε νέα περιεχόμενα ή νέους δικτυακούς τόπους, οι οποίοι θα εμφανίζονται και στους επόμενους χρήστες.
5. Να «εμπιστεύεται» τους χρήστες ως συν-δημιουργούς, δηλαδή ο κάθε χρήστης να μπορεί να προσθέτει πληροφορίες και καταχωρήσεις στον δικτυακό τόπο προκειμένου το υλικό του δικτυακού τόπου να ανανεώνεται και να εμπλουτίζεται όχι μόνο από τον δημιουργό αλλά και από τους απλούς χρήστες (συνεργατικότητα).
6. Αμφίδρομη επικοινωνία του χρήστη με επιχειρήσεις ή οργανισμούς που μπορεί να έχει σαν αποτέλεσμα την επίδρασή του στη υιοθέτηση κατευθύνσεων και λήψη αποφάσεων.

2.2 Εφαρμογές του Web 2.0

Ο όρος Web 2.0 οφείλει την ύπαρξή του σε εφαρμογές, υπηρεσίες, εργαλεία και λειτουργίες που παρουσιάζουν καινοτομίες και ευκολίες που αναζητούσαν οι χρήστες, για αυτό το λόγο και όταν υλοποιήθηκαν έτυχαν ευρείας αποδοχής και διάδοσης. Παρακάτω, περιγράφονται μερικές από τις κυριότερες εφαρμογές του:

2.2.1 Ιστολόγια (Blogs)

Η ετυμολογία του όρου "blog" υποδηλώνει μεταξύ άλλων την ίδια την έννοια της εφαρμογής. Προερχόμενος από σύντμηση των λέξεων "web log", δηλαδή «δικτυακή εγγραφή», επικράτησε σαν "blog" όταν ο Peter Merholz με διάθεση αστεϊσμού διέσπασε τη λέξη weblog δημιουργώντας τη φράση "we blog" (δηλαδή εμείς blogάρουμε- διατηρούμε ή καταχωρούμε επαφές στα blogs). Προκάτοχοι των blogs ήταν οι λίστες ηλεκτρονικού ταχυδρομείου, τα Usenet, τα bulletin board συστήματα και οι εμπορικές δικτυακές υπηρεσίες. Το πρώτο γνωστό blog δημιουργήθηκε και λειτούργησε το 1992 από τον πρωτοπόρο κι εμπνευστή του Παγκόσμιου Ιστού, τον Tim Berners-Lee, με το όνομα "What's New Page".

2.2.2 Wikis

Τα Wikis είναι δικτυακοί τόποι οι οποίοι δίνουν τη δυνατότητα στον επισκέπτη να παρεμβαίνει στο περιεχόμενο τους. Ανάλογα με την πολιτική του wiki, οι χρήστες έχουν άμεση πρόσβαση στην τροποποίηση του περιεχομένου ή αποκτούν αυτό το δικαίωμα μετά τη δημιουργία και την έγκριση ενός προσωπικού λογαριασμού. Της δυνατότητας επεξεργασίας και τροποποίησης του περιεχομένου στερούνται οι χρήστες που έχουν ενταχθεί στις λίστες αποκλεισμού του δικτυακού τόπου. Μέσα από τη διαδικασία συντήρησης ενός τέτοιου δικτυακού τόπου, προκύπτει μία αξιόπιστη βάση περιεχομένου, η οποία ανανεώνεται ανά τακτά χρονικά διαστήματα και αποτελεί ένα συλλογικό έργο της κοινότητας των χρηστών. Τα wikis βρίσκουν εφαρμογή στην κατασκευή συνεργατικών ιστοσελίδων και ιστοσελίδων διαδικτυακής κοινότητας. Το πιο χαρακτηριστικό παράδειγμα των wikis είναι η Wikipedia, η ελεύθερη διαδικτυακή εγκυκλοπαίδεια που αναπτύχθηκε το 2001, και υποστηρίζεται από το μη κερδοσκοπικό οργανισμό Wikimedia. Αρκετές εφαρμογές των wikis συναντούμε επίσης σε επιχειρησιακές εφαρμογές για intranets καθώς και σε συστήματα γνωσιακών βάσεων (knowledge bases). Αξίζει να σημειωθεί ότι η πρώτη εφαρμογή wiki ήταν το "WikiWikiWeb" (γνωστό και ως "Words Web") που αναπτύχθηκε το 1994 από τον αμερικανό προγραμματιστή Ward Cunningham. Σκοπός του WikiWikiWeb ήταν η εύκολη ανταλλαγή ιδεών μεταξύ προγραμματιστών. Τα wikis πήραν την ονομασία τους από τη χαβανέζικη λέξη "wiki" που σημαίνει γρήγορα. Αργότερα, προτάθηκε η χρήση της φράσης "What I Know Is" ως επέκταση των αρχικών w.i.k.i.

2.2.3 Κοινωνική Σήμανση (Social Tagging)

Κοινωνική Σήμανση (Social Tagging) ονομάζουμε τη σύγχρονη πρακτική δημιουργίας και διαχείρισης ετικετών (tags) για την επισήμανση και κατηγοριοποίηση περιεχομένου στον Παγκόσμιο Ιστό. Η έννοια της ετικέτας αναφέρεται σε μια λέξη, όρο ή φράση-κλειδί, χωρίς ιεραρχική σημασία. Οι ετικέτες εκχωρούνται σε τμήματα ή στο σύνολο κάποιας πληροφορίας κειμένου, φωτογραφίας ή βίντεο. Άλλοι παρεμφερείς όροι είναι η «συνεργατική σήμανση» (collaborative tagging) - γνωστή και ως ανθρώπινη ταξινόμηση, η «κοινωνική κατηγοριοποίηση» (social classification) καθώς και η «κοινωνική ευρετηριοποίηση» (social indexing) ή «κοινωνική σήμανση» (social tagging).

2.2.4 Ιστότοποι Κοινωνικής Δικτύωσης (Social Networking Sites)

Οι Ιστότοποι Κοινωνικής Δικτύωσης στοχεύουν στη δημιουργία διαδικτυακών κοινοτήτων (online communities), που αποτελούνται από ανθρώπους-μέλη, οι οποίοι επιθυμούν να έρθουν σε επαφή με άλλα μέλη της κοινότητας, με σκοπό την επικοινωνία και το διαμοιρασμό των ενδιαφερόντων και των δραστηριοτήτων τους. Πρωτοεμφανίστηκαν το 1994, και σήμερα γνωρίζουν πολύ μεγάλη αποδοχή από τους χρήστες του διαδικτύου. Στις μέρες μας, συναντάμε ένα ευρύ φάσμα ιστότοπων κοινωνικής δικτύωσης, διαφόρων τύπων και στόχευσης. Ενδεικτικά αναφέρουμε της υπηρεσίες κοινωνικής δικτύωσης με τη μορφή ευρετηρίων (επαγγελματικές επαφές, παλιοί συμμαθητές, μουσικόφιλοι κ.α.) αλλά και άλλων μέσων διασύνδεσης και επικοινωνίας. Η συμμετοχή των χρηστών σε τέτοιου είδους υπηρεσίες πραγματοποιείται συνήθως, με τη δημιουργία ενός δημόσιου ατομικού ή συλλογικού προφίλ, το οποίο δίνει τη δυνατότητα προβολής σε όλα ή σε επιλεγμένα μέλη και ομάδες της κοινότητας. Οι ιστότοποι κοινωνικής δικτύωσης διακρίνονται με βάση το είδος των υπηρεσιών που προσφέρουν και το κοινό στο οποίο απευθύνονται.

Έτσι προκύπτουν κατηγορίες όπως οι παρακάτω που αναφέρουμε ενδεικτικά:

- «Δίκτυα Κοινωνικοποίησης και Επικοινωνίας» με στόχευση στις προσωπικές σχέσεις (person-oriented). Χαρακτηριστικά παραδείγματα: το Facebook, το Friendster, το Windows Live Space, το MySpace, το Hi5, το Bebo και το Orkut.
- «Επαγγελματικές Κοινότητες» (Professional Communities) που στοχεύουν στις επιχειρήσεις και τους επαγγελματίες. Χαρακτηριστικά παραδείγματα: το LinkedIn και το itLinkz.
- «Δίκτυα Οπτικοακουστικού Υλικού» (media-oriented) με έμφαση στο διαμοιρασμό οπτικοακουστικού (multimedia) περιεχομένου όπως εικόνες, ήχους και βίντεο. Χαρακτηριστικά παραδείγματα: το YouTube και το Flickr.

- «Ιστότοποι μικρο-ιστολογίων» (microblogging sites) που αποτελούν τη νέα τάση στην κοινωνική δικτύωση, ιδιαίτερα σήμερα με την ολοένα αυξανόμενη διείσδυση του διαδικτύου στα κινητά τηλέφωνα και τις έξυπνες συσκευές. Χαρακτηριστικότερο παράδειγμα αυτής της κατηγορίας αποτελεί το Twitter.

2.2.5 RSS (Real Simple Syndication) feeds

Τα RSS (Real Simple Syndication) feeds, προσφέρουν τη δυνατότητα στους χρήστες να λαμβάνουν νέες πληροφορίες από διάφορες ιστοσελίδες, τη στιγμή που δημοσιεύονται, χωρίς να χρειάζεται να τις επισκεφθούν. Η ενημέρωση αυτή μπορεί να γίνεται π.χ. στον browser του Η/Υ χρήστη ή και σε κινητές συσκευές (κινητό τηλέφωνο, PDA, κλπ.). Με αυτό τον τρόπο η σχέση με το διαδίκτυο γίνεται αμεσότερη.

2.2.6 Mashups

Ο συνδυασμός και η χρήση δεδομένων και εφαρμογών από διαφορετικές ιστοσελίδες σε μία, έχει γίνει γνωστός ως mashup. Αυτά υλοποιούνται μέσω ανοιχτών interfaces προγραμματισμού εφαρμογών (open APIs'- Application Programming Interfaces) και συνεισφέρουν στη βελτίωση της λειτουργικότητας των ιστοσελίδων. Για παράδειγμα, σε ιστοσελίδες ενοικίασης σπιτιού, με την ενσωμάτωση χαρτών από άλλη υπηρεσία (π.χ Google maps) παρουσιάζεται ακριβώς που βρίσκονται τα σπίτια και έτσι παρέχεται πληρέστερη πληροφόρηση. Παρακάτω, θα γίνει εκτενή αναφορά για τη συγκεκριμένη εφαρμογή μιας και αποτελεί το κορμό της παρούσας εργασίας.

2.2.7 Άλλες Web 2.0 εφαρμογές

Εκτός από τους τύπους Web 2.0 εφαρμογών που προαναφέρθηκαν συναντιούνται επίσης και οι ακόλουθοι:

- Podcasts και Videocasts που αναφέρονται σε ψηφιακές εγγραφές ήχου και βίντεο, αρχειοθετημένων στα πρότυπα των καταχωρήσεων των ιστολογίων. Τα Podcasts και τα Videocasts δίνουν τη δυνατότητα στους χρήστες να τα αναπαράγουν κατ' απαίτηση (on-demand playback).
- «Εικονικοί κόσμοι» (Virtual Worlds). Ξεκίνησαν με ψυχαγωγικό χαρακτήρα αλλά πλέον έχουν απήχηση σε εφαρμογές επιχειρησιακού και εκπαιδευτικού σκοπού (π.χ. Ψηφιακές Τάξεις).
- «Ιστότοποι συγκέντρωσης και ανταλλαγής φωτογραφιών» (Photo Sharing sites). Μία πρώιμη μορφή τέτοιου ιστότοπου πρωτοεμφανίστηκε στα μέσα της δεκαετίας του '90 με δυνατότητα εκτύπωσης ψηφιακών φωτογραφιών.

■ «Διαδικτυακές συνεδριάσεις» (Web conferences). Αποτελούν ένα συνεργατικό δικτυακό τρόπο επικοινωνίας σε πραγματικό χρόνο (peer-to-peer meetings). Οι περισσότερες εφαρμογές αυτού του είδους διαθέτουν δυνατότητες όπως η ανταλλαγή γραπτών ή φωνητικών μηνυμάτων, βιντεοκλήσεις, διαμοιρασμό αρχείων και εφαρμογών καθώς και δυνατότητα για κοινή χρήση οθόνης μεταξύ των συνομιλητών.



Εικόνα 2-2: Κατηγορίες και εκπρόσωποι εφαρμογών Web 2.0 (πηγή: <http://conem.wordpress.com/2008/08/19/social-media-has-created-a-new-layer-of-influencers/>)

2.3 Τεχνολογίες του Web 2.0

Παρακάτω παρουσιάζονται συνοπτικά οι σημαντικότερες τεχνολογίες που χρησιμοποιούνται από το Web 2.0 και το διαφοροποιούν ως προς τον τρόπο λειτουργίας και παρουσίασης των ιστοσελίδων σε σχέση με το παραδοσιακό Web καθώς και επεξηγηματικά κομμάτια των τεχνολογιών αυτών :

- Χρήση πλούσιων και διαδραστικών interfaces χρηστών (Rich Internet Applications-RIA) που χρησιμοποιούν την Javascript και την Ajax, που αντιπροσωπεύει την τάση του WEB 2.0 για όσο το δυνατόν καλύτερη εκμετάλλευση του δικτύου. Αντί να φορτώνεται ολόκληρη η σελίδα, ανανεώνονται μόνο τα δεδομένα που αλλάζουν όσο ο χρήστης βρίσκεται ή επανέρχεται σε αυτή (π.χ. Στο Gmail ο υπολογισμός του διαθέσιμου

αποθηκευτικού χώρου ανανεώνεται σε πραγματικό χρόνο και από όλη τη σελίδα αλλάζει μόνο αυτός). Πιο συγκεκριμένα:

- Η JavaScript είναι γλώσσα προγραμματισμού η οποία έχει σαν σκοπό την παραγωγή δυναμικού περιεχομένου σε ιστοσελίδες. Έχει τις ρίζες της στην ECMAScript της οποίας ουσιαστικά αποτελεί επέκταση με μερικές πρόσθετες δυνατότητες, έχει βασιστεί όσον αφορά τον τρόπο σύνταξης του κώδικά της στη γλώσσα προγραμματισμού C, με την οποία παρουσιάζει πολλές ομοιότητες και αποτελεί μια client side γλώσσα προγραμματισμού. Αυτό σημαίνει ότι η επεξεργασία του κώδικα Javascript και η παραγωγή του τελικού περιεχομένου HTML δεν πραγματοποιείται στον server, αλλά στο πρόγραμμα περιήγησης των επισκεπτών. Αυτό έχει και πλεονεκτήματα και μειονεκτήματα καθώς η Javascript δεν έχει καμία απαίτηση από πλευράς δυνατοτήτων του server για να εκτελεστεί (επεξεργαστική ισχύ, συμβατό λογισμικό διακομιστή), αλλά βασίζεται στις δυνατότητες του browser των επισκεπτών. Επίσης μπορεί να ενσωματωθεί σε στατικές σελίδες HTML αλλά δεν παρέχει συνδεσιμότητα με βάσεις δεδομένων. Η Javascript δεν θα πρέπει να συγχέεται με τη Java, που είναι διαφορετική γλώσσα προγραμματισμού και με διαφορετικές εφαρμογές. Τονίζεται ότι ο σωστός τρόπος γραφής της είναι "Javascript" και όχι 'Java script' σαν δύο λέξεις, όπως λανθασμένα γράφεται ορισμένες φορές. Ο κώδικας Javascript μιας σελίδας περικλείεται από τα HTML tags `<script language=javascript>` και `</script>` ενώ την απεικόνιση κειμένου αναλαμβάνει η εντολή `document.write`.
- Το AJAX τώρα περιλαμβάνει διάφορες τεχνολογίες που στρέφονται γύρω από την ασύγχρονη φόρτωση και παρουσίαση περιεχομένου από το διαδίκτυο. Τα XHTML και CSS για στιλιστική διαμόρφωση. Το Document Object Model (DOM) που χρησιμοποιείται από το φυλλομετρητή για δυναμική παρουσίαση, αλληλεπίδραση και ασύγχρονη ανταλλαγή δεδομένων, τα οποία συνήθως είναι XML δεδομένα και browser side scripting ως επί το πλείστον με Javascript. Όταν χρησιμοποιούνται μαζί, ο στόχος αυτών των τεχνολογιών είναι να δημιουργηθεί μια ομαλή, συνεκτική διαδικτυακή εμπειρία για το χρήστη με την ανταλλαγή μικρών ποσοτήτων δεδομένων με τους διακομιστές περιεχομένου αντί για την επαναφόρτωση και επανυποβολή ολόκληρης της σελίδας μετά από κάποια ενέργεια του χρήστη. Η κατασκευή AJAX μηχανών για mashups μπορεί να γίνει από διάφορα AJAX toolkits και βιβλιοθήκες συνήθως υλοποιημένα σε Javascript.

- Χρήση CSS (Cascading Style Sheets) για να διαχωρίζονται τα δεδομένα καθαρής πληροφορίας από τα δεδομένα μορφοποίησης σε μια ιστοσελίδα. Αυτό, πέρα από την οικονομία στο εύρος ζώνης του δικτύου, προσφέρει και ευελιξία στον τρόπο παρουσίασης των δεδομένων, αφού ο χρήστης βλέπει τα δεδομένα ανάλογα με το CSS που ο ίδιος έχει (π.χ. τα ίδια δεδομένα ανάλογα με το CSS μπορούν να παρουσιαστούν σε οθόνη υπολογιστή, κατευθείαν σε εκτυπωτή, σε μορφή ανάγνωσης για τυφλούς ή και να μετατραπούν σε φωνή και με χρήση κατάλληλου λογισμικού)
- Χρήση RSS feeds ή και Atom . Το RSS (Really Simple Syndication) και το ATOM (Atom Syndication Format) είναι δύο τεχνολογίες του διαδικτύου (Internet) που χρησιμοποιούνται για την ενημέρωση των χρηστών σχετικά με προσθήκες ή μεταβολές στο περιεχόμενο στοχευμένων ιστοχώρων (web sites). Σχεδόν το σύνολο των ιστοχώρων, που ακολουθούν τις προδιαγραφές του Web 2.0, παρέχουν διαρκή ενημέρωση μέσω Web Feeds.
- Χρήση ανοικτού λογισμικού (π.χ. Linux σαν λειτουργικό σύστημα, Apache σαν Web server, MySQL σαν βάση δεδομένων και PHP, Pearl, Python, σαν γλώσσες προγραμματισμού).
- «Ελαφρά» πρωτόκολλα δικτύου REST και SOAP που χρησιμοποιούν απλές HTTP εντολές (get, post, put, κλπ) για ανάκτηση δεδομένων από τους servers.
- Αρχιτεκτονικές SOA (Service Oriented Architecture) που επιτρέπουν το διαμοιρασμό και την επαναχρησιμοποίηση υπηρεσιών-εφαρμογών από διαφορετικά προγράμματα λογισμικού και SaaS (Software as a Service) όπου οι εφαρμογές είναι εγκατεστημένες σε κεντρικό server στο δίκτυο και οι χρήστες τις χρησιμοποιούν μέσω browser ανεξαρτήτως H/Y, τόπου, και χρονικής στιγμής.

2.4 MashUps

2.4.1 Ορισμός

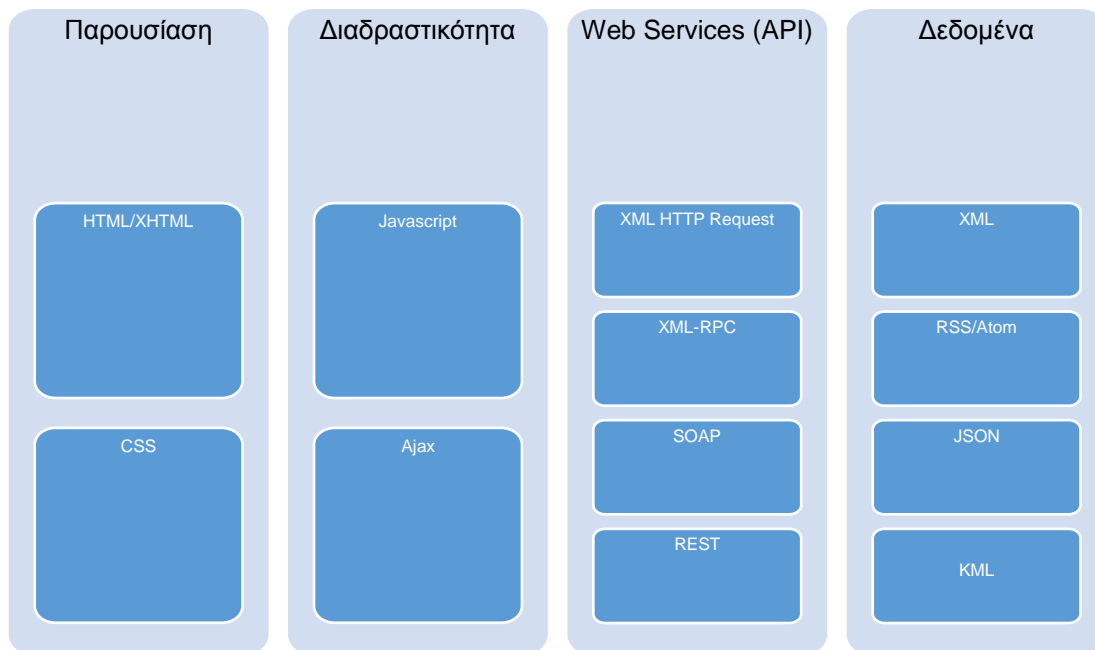
Ένα mashup είναι μια σχετικά νέα μέθοδος ανάπτυξης εφαρμογών που επιτρέπει την ενσωμάτωση προϊόντων και υπηρεσιών από διάφορες ανταγωνιστικές επιχειρήσεις όπως η Microsoft, η Google, η Amazon και το Yahoo με σκοπό την δημιουργία νέων, υπηρεσιών και προϊόντων. Ο όρος «mashup» είναι δανεισμένος από τη μουσική βιομηχανία, όπου χρησιμοποιείται για να περιγράψει τον μη εγκεκριμένο συνδυασμό των φωνητικών ενός τραγουδιού, με τη μουσική υπόκρουση ενός άλλου. Το ίδιο περίπου συμβαίνει και με το mashup των ιστοσελίδων. Συνδυάζουν ιστοσελίδες για να δημιουργήσουν χρήσιμες υβριδικές ιστοσελίδες και να προωθήσουν, έτσι, τη φιλοσοφία του διαδικτύου[11]. Το mashup δηλαδή

αποτελεί μια δικτυακή εφαρμογή που συνδυάζει δεδομένα από περισσότερες από μια διαφορετικές πηγές σε ένα ενιαίο εργαλείο, τα επεξεργάζεται και τα παρουσιάζει με διαφορετικούς τρόπους. Γενικά μπορούμε να πούμε ότι αποτελεί ένα συνδυασμό δύο ή περισσότερων υπηρεσιών σε μία εφαρμογή.

2.4.2 Η αρχιτεκτονική των mashups

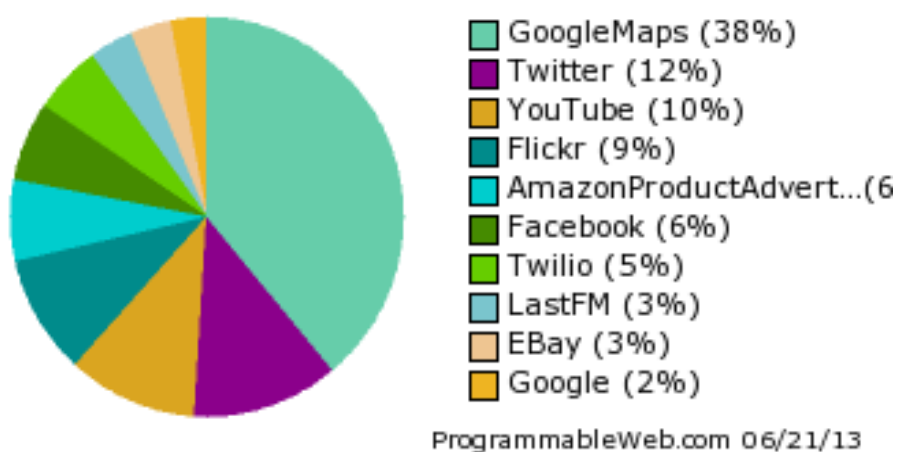
Οι υβριδικές εφαρμογές ιστού, που καλούνται mashups, συνδυάζουν, όπως αναφέρθηκε και παραπάνω, δεδομένα από διάφορους ιστοτόπους με σκοπό τη δημιουργία ενός ενοποιημένου και εμπλουτισμένου αποτελέσματος. Οι βασικότερες μέθοδοι άντλησης περιεχομένου για την δημιουργία των mashups είναι οι ακόλουθες:

1. Web APIs (Web Application Programming Interface) : Το Web API είναι διεπαφή που ένα υπολογιστικό σύστημα, βιβλιοθήκη ή μία υπηρεσία διαδικτύου (Web Services) παρέχει, προκειμένου να επιτρέψει να γίνουν προς αυτό αιτήσεις από άλλα προγράμματα για ανταλλαγή δεδομένων. Πιο απλά, ένα Web API είναι ένα «κανάλι επικοινωνίας» μιας υπηρεσίας διαδικτύου με μία εξωτερική πηγή (π.χ. ιστοσελίδα, web εφαρμογή) μέσω συγκεκριμένων προγραμματιστικών εντολών, που ορίζονται από έναν οδηγό εντολών και χρήσης του API. Τα Web APIs(Application Programming Interface) βασίζονται κατά κύριο λόγο στα τρία πρωτόκολλα που χρησιμοποιούν το μεγαλύτερο μέρος των διαδικτυακών υπηρεσιών και εφαρμογών τα οποία είναι τα XML_RPC(Extensible Markup Language-remote procedure call),REST(Representational State Transfer) και SOAP(Simple Object Access Protocol), μέσω των οποίων γίνεται δυνατή η πρόσβαση σε δεδομένα, υπηρεσίες, πηγές και συστατικά διεπαφών. Τα APIs διαχειρίζονται διάφορα δεδομένα, τα οποία, όπως φαίνεται στο παρακάτω σχήμα μπορούν να είναι σε απλή μορφή XML, σε RSS/Atom που περιεγράφηκε παραπάνω, αλλά και σε JSON και KLM. Το JavaScript Object Notation (JSON) είναι μία μορφή κειμένου, κατανοητή από τον άνθρωπο, για την απεικόνιση αντικειμένων και άλλων δομών δεδομένων και χρησιμοποιείται κυρίως για τη μεταφορά αυτών μέσω μιας σύνδεσης δικτύου. Από την άλλη πλευρά η Keyhole Markup Language (KLM) είναι μία γλώσσα βασισμένη στην XML για τη διαχείριση της απεικόνισης του τρισδιάστατου χώρου σε web εφαρμογές χαρτογράφησης, όπως είναι το Google Maps και το Microsoft Virtual Earth.



Εικόνα 2-3: Κατηγοριοποιημένες τεχνολογίες χρησιμοποιούμενες από Mashups

Τα Web 2.0 APIs προσφέρουν τρία πολύ σημαντικά πλεονεκτήματα. Πρώτον δίνουν πρόσβαση σε εύρωστες τεχνολογίες υψηλής ανάπτυξης, τις οποίες μόνο μεγάλοι οργανισμοί με εξειδικευμένους προγραμματιστές θα μπορούσαν να δημιουργήσουν, δεύτερον παρέχουν προσπέλαση σε τεράστιες ποσότητες περιεχομένου που κανένας ιδιώτης δεν θα μπορούσε να κατέχει από μόνος του και να τις διαχειρίζεται και τρίτον επιτρέπουν την εύκολη ανάπτυξη καινοτόμων εφαρμογών με πανίσχυρες τεχνολογίες[12]. Η ακόλουθη εικόνα δείχνει την κατανομή των πιο δημοφιλών Web APIs που χρησιμοποιούνται για τη δημιουργία Mashups.



Εικόνα 2-4: Κατανομή των πιο δημοφιλών web APIs(Πηγή: <http://www.programmableweb.com/apis>)

2. Web Feeds (RSS/Atom) : Τα Web Feeds είναι ένας άμεσος τρόπος για να ενημερώνεται κάποιος συνοπτικά για τις προσθήκες πληροφοριών στους ιστοχώρους που τον ενδιαφέρουν.

3. Screen Scraping : Τεχνική με την οποία ένα πρόγραμμα εξάγει δεδομένα από την οπτική έξοδο που παράγει κάποιο άλλο πρόγραμμα.

Πολλοί άνθρωποι πειραματίζονται με τα mashups χρησιμοποιώντας γνωστά sites που παρέχουν υπηρεσίες που μπορούν να χρησιμοποιηθούν όπως το Amazon, το eBay, το Flickr, το Google, η Microsoft, το Yahoo αλλά και τα YouTube APIs , το οποίο έχει οδηγήσει στη δημιουργία ενός συντάκτη για mashups (Mashup editor) .

Η αρχιτεκτονική των εφαρμογών ιστού, Mashups, αποτελείται πάντα από τρία μέρη:

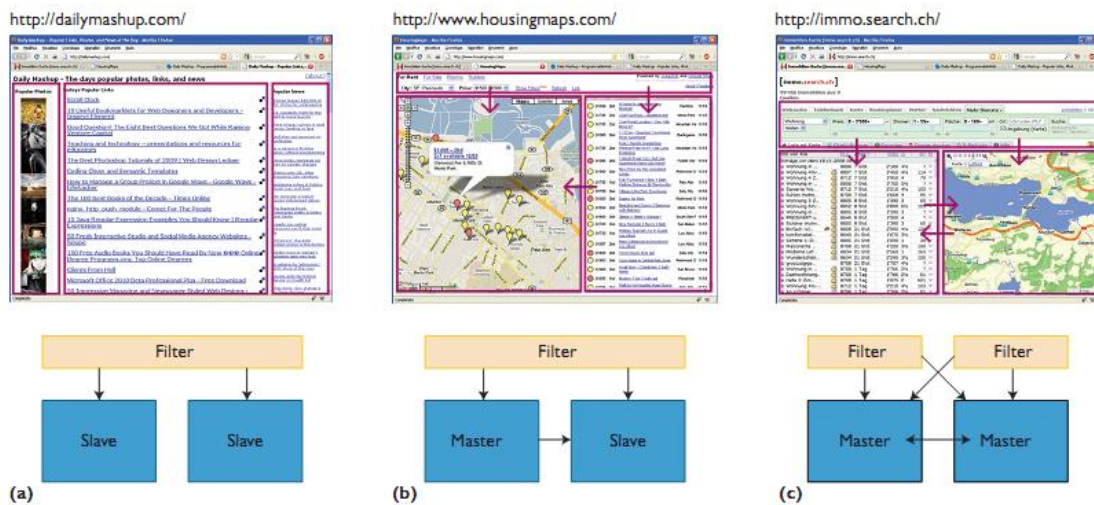
1. Τους παροχείς περιεχομένου : Αποτελεί την πηγή των δεδομένων. Τα δεδομένα παρέχονται χρησιμοποιώντας τις τεχνικές που αναφέρθηκαν παραπάνω.
2. Την ιστοσελίδα Mashup: είναι η διαδικτυακή εφαρμογή η οποία παρέχει τη νέα υπηρεσία. Τα Mashups μπορούν να υλοποιηθούν με τη χρήση παραδοσιακών server-side τεχνολογιών, όπως Java servlets, CGI, PHP ή ASP ή εναλλακτικά, το ενοποιημένο περιεχόμενο μπορεί να παράγεται απευθείας στον browser του πελάτη μέσω client-side scripting (JavaScript) ή applets. Τα Mashups που χρησιμοποιούν client-side λογική μπορούν να χαρακτηριστούν Rich Internet Applications (RIAs), πράγμα που σημαίνει ότι είναι πολύ προσανατολισμένα προς τη διαδραστική εμπειρία του χρήστη η οποία αποτελεί έναν από τους στόχους του web 2.0. Μερικές φορές τα mashups χρησιμοποιούν ένα συνδυασμό της server-side και client-side λογικής για την επίτευξη συνάθροισης των δεδομένων τους.
3. Τον φυλλομετρητή σελίδων (web-browser) του χρήστη : Στο browser του χρήστη παρουσιάζεται η εφαρμογή γραφικά και είναι το σημείο όπου υπάρχει αλληλεπίδραση του χρήστη με αυτή. . Όπως περιγράφεται παραπάνω, τα mashups συχνά χρησιμοποιούν client-side λογική για να συγκεντρώσουν και να συνθέσουν το ενοποιημένο περιεχόμενο. Αυτή η αρχιτεκτονική ακούγεται παρόμοια με ένα portal οπότε ποια είναι η διαφορά; Ο σχεδιασμός του portal είναι πιο συγκεντρωτικός με κάθε μέρος να σχεδιάζεται ως ένα widget με τη πρόθεση να συγκεντρώνονται σε μια ή περισσότερα portal οθόνες. Επίσης, ένα widget είναι λίγο GUI-data πακέτο, αποτελείται από ένα μεγάλο κομμάτι επαναχρησιμοποιήσιμων δεδομένων και τα μέσα για να εμφανιστούν αυτά στο portal. Τα

Mashups αντιθέτως είναι API-driven, έτσι ώστε η πηγή παρέχει μόνο τα δεδομένα, δεν είναι ένα μίνι-GUI που εμφανίζει τα δεδομένα.

Τα συστατικά ενός Mashup μπορούν να είναι UI widgets, πηγές δεδομένων και υπολογιστικές υπηρεσίες. Μερικά είναι ορατά στο mashup, ενώ άλλα μη ορατά. Κρυμμένα στοιχεία (πηγές δεδομένων, όπως RSS feeds) απαιτούν ένα άλλο στοιχείο για να καθιστούν δεδομένα. Για παράδειγμα, μπορούμε να χρησιμοποιήσουμε ένα RSS reader για να εμφανιστούν τα στοιχεία RSS feed έτσι ώστε ο χρήστης να μπορεί να επιθεωρεί και να περιηγηθεί μέσα σε αυτά. Εμφανή στοιχεία μπορούν να παίξουν διαφορετικούς ρόλους που επηρεάζουν την αντίληψη του χρήστη για την ποιότητα του τελικού συνταιριάσματος και θα πρέπει, συνεπώς, να ληφθούν προσεκτικά υπόψη. Αναλύοντας τα πιο δημοφιλή mashups που δημοσιεύθηκαν στο programmableweb.com, έχουμε προσδιορίσει τους εξής τυπικούς ρόλους συστατικών στοιχείων τους:

- **Master**. Ακόμη και αν ένα mashup ενσωματώνει πολλαπλές συνιστώσες σε μία σελίδα, στις περισσότερες περιπτώσεις, μία συνιστώσα είναι πιο σημαντική από ό, τι οι άλλες. Αυτή είναι και η συνιστώσα που οι χρήστες αλληλεπιδρούν περισσότερο. Είναι συνήθως το σημείο εκκίνησης για την αλληλεπίδραση με τον χρήστη που προκαλεί τις άλλες συνιστώσες να αντιδράσουν και να συγχρονιστούν ανάλογα.
- **Slave**. Η συμπεριφορά μιας συνιστώσας slave εξαρτάται από μία άλλη συνιστώσα: η κατάσταση της κυρίως τροποποιήθηκε από γεγονότα που προέρχονται από άλλη (master) συνιστώσα. Πολλά mashups, επίσης, επιτρέπουν στους χρήστες να αλληλεπιδρούν με τις slave συνιστώσες. Ωστόσο, τα στοιχεία περιεχομένου που οι slave συνιστώσες προβάλλουν επιλέγονται μέσω της αλληλεπίδρασης του χρήστη με την κατασκευαστική master συνιστώσα και με την αυτόματη διάδοση και συγχρονισμό πληροφοριών από την master στις slave συνιστώσες.
- **Filter**. Οι Filter συνιστώσες επιτρέπουν στους χρήστες να καθορίζουν τους όρους για το περιεχόμενο που οι άλλες συνιστώσες δείχνουν. Παρέχουν (ενδεχομένως ιεραρχικά) μηχανισμούς πρόσβασης που επιτρέπουν στους χρήστες σταδιακά να επιλέξουν ποιο περιεχόμενο θέλουν να δουν. Τα Filter επίσης μειώνουν το μέγεθος των συνόλων δεδομένων άλλων συνιστωσών, συμβάλλοντας στη βελτίωση της κατανόησης του mashup και στην ευκολία χρήσης του. Στις περισσότερες περιπτώσεις, οι χρήστες καθορίζουν συνθήκες filter πάνω σε δεδομένα master συνιστωσών ενώ συγχρονίζονται με τις slave, έτσι ώστε η λογική της ολοκλήρωσης να φιλτράρει αυτόματα το περιεχόμενο των slave. Με λίγα λόγια, μια filter συνιστώσα επιτρέπει στους χρήστες να επιλέγουν ομάδες ή σύνολα στοιχείων δεδομένων, ενώ η master συνιστώσα επιτρέπει

στους χρήστες να επιλέξουν μεμονωμένα στοιχεία που οι slave συνιστώσες συμπληρώνουν με επιπρόσθετα στοιχεία. Οι master και slave συνιστώσες συνήθως προέρχονται από το διαδίκτυο, ο συνθέτης του mashup αναπτύσσει επι το πλείστον τις filter συνιστώσες.



Εικόνα 2-5: Βασικά πρότυπα ανάπτυξης mashup, (α) το slave-slave, (b) το master-slave, και (c) το master-master μοτίβο(πηγή: Cinzia Cappiello, Florian Daniel, Maristella Matera, Cesare Pautasso, “Information Quality in Mashups”, IEEE Internet Computing, 1089-7801/2010, 2010, pp. 30-38)

2.4.3 Είδη Mashups

Τα Mashups έρχονται αυτήν την περίοδο σε δυο γενικές μορφές τα *mashups καταναλωτή* (consumer mashups) και τα *mashups δεδομένων* (data mashups).

- Τα mashups καταναλωτή συνδυάζουν στοιχεία δεδομένων από πολλαπλές πηγές και τα καμουφλάρουν πίσω από μια απλή ενοποιημένη γραφική διεπαφή.
- Τα «*mashups δεδομένων*» αναμειγνύουν δεδομένα παρόμοιων τύπων από διαφορετικές πηγές (π.χ. τα Yahoo Pipes), όπως παραδείγματος χάριν συνδυάζοντας τα στοιχεία από πολλαπλά RSS feeds σε ένα ενιαίο feed με ένα γραφικό περιβάλλον σαν interface.

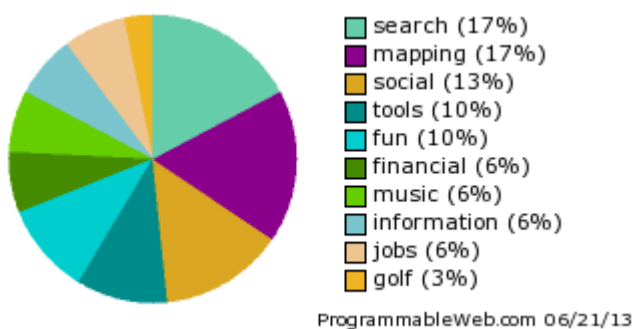
Επιπλέον, τα mashups μπορούν να χωριστούν σε τέσσερις κύριες κατηγορίες με βάση το περιεχόμενο που διαχειρίζονται:

- **Mapping mashups** (π.χ. Chicago Crime) : Πρόκειται για τις πιο γνωστές εφαρμογές, οι οποίες χρησιμοποιούν χάρτες. Στην περίπτωση αυτή οι άνθρωποι συγκεντρώνουν περιεχόμενο για πράγματα ή δραστηριότητες το οποίο μπορεί να συνδυαστεί με

τοποθεσίες πάνω σε ένα χάρτη. Οι εφαρμογές αυτές χρησιμοποιούν μια υπηρεσία παροχής χαρτών (π.χ. Google, Yahoo) συνδυάζοντας δεδομένα με σημεία πάνω στο χάρτη.

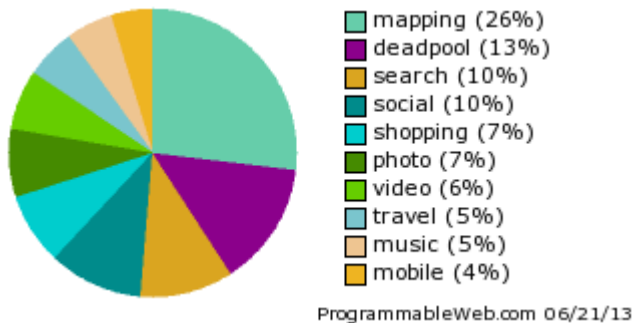
- Video and photo mashups (π.χ. Flickr) : Σήμερα στο web υπάρχουν διάφοροι δικτυακοί τόποι που φιλοξενούν video ή φωτογραφίες για κάποιο χρήστη τα οποία μάλιστα περιέχουν και μεταδεδομένα (ποιος πήρε τη φωτογραφία, πότε και που). Αυτό έχει οδηγήσει στη δημιουργία εφαρμογών mashups οι οποίες συγκεντρώνουν τις φωτογραφίες μαζί με άλλες πληροφορίες και τις συνδυάζουν με τα μεταδεδομένα.
- Search and shopping mashups (π.χ. Travature) : Οι εφαρμογές αυτές αφορούν συγκριτικούς καταλόγους μεταξύ προϊόντων αλλά και αναζήτηση για τους διαθέσιμους προμηθευτές. Οι εφαρμογές αυτές προϋπήρχαν των όρων «mashup» και «Web 2.0» ωστόσο απέκτησαν άλλο νόημα από τότε που το Amazon και το eBay δημιούργησαν τα δικά τους APIs ώστε οποιοσδήποτε χρήστης να έχει πρόσβαση στο περιεχόμενό τους.
- News mashups (π.χ. Digg, iGoogle) : Διάφορες πηγές που δημοσιεύουν νέα έχουν χρησιμοποιήσει τεχνολογίες RSS και ATOM (από το 2002) προκειμένου οι επισκέπτες τους να έχουν τα νέα στον υπολογιστή τους ή στον προσωπικό τους δικτυακό τόπο. Οι εφαρμογές αυτού του είδους επιτρέπουν σε ένα χρήστη να συγκεντρώνει νέα από διαφορετικές πηγές προκειμένου να δημιουργήσει την προσωπική του εφημερίδα στο δίκτυο με βάση τα ενδιαφέροντά του.

Στην ακόλουθη εικόνα, απεικονίζεται η κατανομή των διαφόρων ειδών mashups όπως έχει καταγραφεί από το site www.ProgrammableWeb.com τις τελευταίες 14 ημέρες από την αναγραφόμενη ημερομηνία.



Εικόνα 2-6: Κατανομή των διάφορων ειδών mashups τις τελευταίες 14 ημέρες(Πηγή: <http://www.programmableweb.com/mashups>)

Η αμέσως επόμενη εικόνα παρουσιάζει την γενική κατανομή των διαφορετικών ειδών mashups.



Εικόνα 2-7: Γενική κατανομή των διάφορων ειδών mashups(Πηγή: <http://www.programmableweb.com/mashups>)

Συγκρίνοντας τις δύο παραπάνω εικόνες, συμπεραίνεται ότι τα mapping mashups και τα search mashups διατηρούν τις πρώτες θέσεις στην κατάταξη των ειδών κατά τη διάρκεια της εξέλιξης των ειδών.

2.4.4 Mashups tools

Όπως είναι λογικό, τα mashups διαδίδονται πολύ γρήγορα, καθώς όλο και περισσότεροι χρήστες του διαδικτύου θέλουν να έχουν πρόσβαση σ' ένα δικό τους Mashup που να ανταποκρίνεται στις ανάγκες τους. Για αυτό το λόγο έχουν αναπτυχθεί διαδικτυακές εφαρμογές οι οποίες μέσω γραφικού περιβάλλοντος απλοποιούν την δημιουργία Mashup κάνοντας κάθε χρήστη του διαδικτύου εν δυνάμει προγραμματιστή. Παραδείγματα τέτοιων εφαρμογών είναι : Google Mashup Editor, Microsoft Popfly, Netvibes, Pageflakes, Ubiquity (Firefox), Yahoo! Pipes.

Παρακάτω παρουσιάζονται κάποια παραδείγματα πλατφορμών που έχουν δημιουργηθεί με στόχο την ανάπτυξη mashup εφαρμογών:

- Exhibit[13] είναι ένα πλαίσιο για τη δημιουργία ιστοσελίδων που περιέχουν δυναμικά και πλούσια οπτικά δομημένα δεδομένα. Γενικώς, χρησιμοποιείται σε συνδυασμό με το Babel[14]. Επιτρέπει στους χρήστες να συγκεντρώνουν δεδομένα που λήφθηκαν σε διαφορετικές μορφές, όπως η RDF / XML, N3, Bibtex. Το Exhibit οπτικοποιεί την έξοδο του mashup σε σελίδες HTML, και επιπρόσθετα επιτρέπει στο χρήστη να έχει πρόσβαση

στα δεδομένα εξάγοντας τα σε διάφορες μορφές, συμπεριλαμβανομένων των RDF / XML και Exhibit JSON.

- Apatar[15] είναι ένα Mashup εργαλείο ενοποίησης δεδομένων που βοηθά τους χρήστες να ενσωματώνουν δεδομένα της επιφάνειας εργασίας τους με το διαδίκτυο. Οι χρήστες εγκαθιστούν μια εφαρμογή οπτικού σχεδιασμού εργασιών για να δημιουργήσουν σχήματα ενσωμάτωσης που ονομάζονται DataMaps. Το Apatar στοχεύει κυρίως στην συγκέντρωση και να επεξεργασία δεδομένων που μπορούν να επαναχρησιμοποιηθούν από άλλες εφαρμογές, έτσι πρόσθετα εργαλεία που κατανοούν τις μορφές εξόδου του Apatar να χρησιμοποιούνται για την παρουσίαση των δεδομένων.
- Dapper είναι ένα δωρεάν online εργαλείο για την δημιουργία data wrappers που εξάγουν δεδομένα από καλά δομημένες ιστοσελίδες. Το Dapper βασίζεται σε point and click τεχνική καθιστώντας τον χρήστη σε θέση να επιλέξει τα περιεχόμενα που πρόκειται να εξαχθούν και να αναγάγει σε κανόνες εξαγωγής (π.χ., κανονικές εκφράσεις). Συγκεκριμένα, η άντληση στοιχείων αξιοποιεί τη δομή της HTML μορφοποίησης για να γίνει κατανοητό ποια στοιχεία θέλουμε να εξαγάγουμε (π.χ. τα πρώτα κελιά όλων των γραμμών ενός πίνακα). Μόλις ταυτοποιηθούν σωστά, τα εξαγόμενα πεδία δεδομένων μπορούν να ονομαστούν και να δομηθούν και στη συνέχεια να δημοσιευθούν, για παράδειγμα, ως RSS ή XML υπηρεσίες δεδομένων. Οι δημοσιευμένες υπηρεσίες μπορούν εύκολα να είναι πρόσβασιμες μέσω ενός μοναδικού URL και επαναφορτώνονται κάθε φορά που επισκεπτόμαστε την αντίστοιχη διεύθυνση URL.
- MashMaker[16] είναι ένα web-based εργαλείο από την Intel που επεξεργάζεται, εξετάζει και χειρίζεται δεδομένα του Παγκόσμιου Ιστού. Το MashMaker είναι διαφορετικό από τα άλλα εργαλεία καθώς λειτουργεί άμεσα στις σελίδες web. Στην πραγματικότητα, το MashMaker επιτρέπει στους χρήστες να δημιουργήσουν ένα mashup κάνοντας περιήγηση στο διαδίκτυο και συνδυάζοντας διαφορετικές ιστοσελίδες. Ο τελικός στόχος αυτού του εργαλείου είναι να προτείνει ο χρήστης κάποιες βελτιώσεις (mashups ή widgets), εάν είναι δυνατές, για τις ιστοσελίδες που επισκέπτεται.
- JackBe Presto[17] είναι μια ισχυρή και ολοκληρωμένη πλατφόρμα mashup που παρέχει επιχειρησιακού επίπεδου λύσεις. Το Presto δίνει τη δυνατότητα να παραχθούν εύκολα (σχεδιάζει, δοκιμάζει και αναπτύσσει) mashups συγχώνευσης δεδομένων που

προέρχονται από διαφορετικές πηγές. Ειδικότερα μπορεί να συνδέεται με πηγές δεδομένων πολύ κοινές στον κόσμο των επιχειρήσεων (όπως υπολογιστικά φύλλα του Excel, Oracle λογισμικό, κ.λπ.), δυνατότητα που τα περισσότερα mashup tools δεν παρέχουν. Απλή σύνθεση mashup μπορεί να γίνει, επίσης, από μη-IT χρήστες, μέσω του Presto Wires εργαλείου. Το Wires είναι ένας οπτικός editor που βασίζεται σε μια απλή και διαισθητική προσέγγιση σύνθεσης pipeline δεδομένων επιτρέποντας σε κάποιον να συγχωνεύσει τα δεδομένα που προέρχονται από διαφορετικές εσωτερικές και εξωτερικές πηγές που παράγουν ένα τελικό αποτέλεσμα που μπορεί να εμφανίζεται γραφικά ως mashlet. Τα Mashlets μπορεί να συνδεθούν σε ένα ταμπλό όπως το περιβάλλον εργασίας του χρήστη ή σε ένα portal , ή μπορούν να ενσωματωθούν σε μια κανονική σελίδα web.

- Yahoo! Pipes[18] είναι ένα web-based εργαλείο που παρέχεται από τη Yahoo. Οι χρήστες μπορούν να δημιουργήσουν εφαρμογές mashup με την άθροιση και το χειρισμό δεδομένων από web feeds, ιστοσελίδες, και άλλες υπηρεσίες. Ένα pipe αποτελείται από μία ή περισσότερες ενότητες, κάθε μία από τις οποίες εκτελεί ένα ενιαίο έργο, όπως την ανάκτηση feeds από μια πηγή στο διαδίκτυο, το φιλτράρισμα, την ταξινόμηση ή την συγχώνευση τους. Η έξοδος των Pipes μπορεί να είναι είτε προσβάσιμες από ένα Πελάτη-client μέσω μιας μοναδικής διεύθυνσης URL, όπως RSS ή JSON, ή να οπτικοποιείται στο Yahoo Map. Παρακάτω, θα γίνει εκτενέστερη ανάλυση για το συγκεκριμένο εργαλείο μιας και αποτελεί βασικό κομμάτι της πρακτικής εξέτασης του αντικειμένου της πτυχιακής αυτής.

3 ΠΕΙΡΑΜΑΤΙΚΗ ΠΡΟΣΕΓΓΙΣΗ

Όπως προαναφέρθηκε και στο εισαγωγικό κεφάλαιο πέραν της θεωρητικής εξέτασης του προβλήματος μας υπήρξε και μια πιο πρακτική εξέταση του μέσω πειραματισμού με διαθέσιμα εργαλεία δημιουργίας mashups. Πιο συγκεκριμένα στην προσπάθεια να εξεταστεί κατά πόσο τα mashups μπορούν να συμβάλουν ενεργά στην συνάθροιση μέσων από κοινωνικά δίκτυα πειραματιστήκαμε με το Yahoo!Pipes και το WebhookIt.

3.1 Περιγραφή της mashup εφαρμογής

Ο πειραματισμός μας περιλάμβανε την δημιουργία ενός photo mashup το οποίο πραγματοποιώντας αναζήτηση μέσω μιας λέξη-κλειδιού που δίνεται από τον τελικό χρήστη στα Flickr, Instagram και Picasa του επιστρέφονται δεδομένα των διαδικτυακών αυτών τόπων που συσχετίζονται με την λέξη-κλειδί που δόθηκε. Με άλλα λόγια το mashup που δημιουργήθηκε για τις ανάγκες του περάματος μας ,θα μπορούσε κανείς να πει ότι συνδυάζει τις τρεις αυτές ιστοσελίδες κοινωνικών δικτύων εκκαθαρίζοντας και θέτοντας υπό επεξεργασία το υλικό τους που τον ενδιαφέρει περισσότερο.

Αναλυτικότερα ο χρήστης εισαγάγει σε μία φόρμα την λέξη-κλειδί για την οποία θέλει να του επιστραφεί υλικό. Στη συνέχεια η λέξη-κλειδί μαζί με άλλες απαραίτητες πληροφορίες που παρέχονται αυτόματα από την εφαρμογή , μέσω των APIs του κάθε social network site (δηλαδή των Flickr , Instagram και Picasa) επιστρέφει τα ζητούμενα δεδομένα τα οποία πάλι μέσω της εφαρμογής μας χρησιμοποιώντας κάποιο ταξινομητή-φίλτρο δίδονται στον τελικό χρήστη με σκοπό είτε την απλή θέαση τους είτε την περαιτέρω επεξεργασία τους από άλλες εφαρμογές.

Η αναλυτική αυτή περιγραφή της εφαρμογής διαχωρίζει επι της ουσίας την εφαρμογή στις εξής διαδικασίες:

- Εισαγωγή της λέξης –κλειδιού στην φόρμα εισαγωγής της με σκοπό την σύνδεση της με τις υπηρεσίες των Flickr, Instagram και Picasa
- Αποστολή της λέξης-κλειδιού και άλλων απαραίτητων δεδομένων στο API του Flickr ώστε να επιτευχθεί η σύνδεση με αυτό για την χρησιμοποίηση των υπηρεσιών του ώστε να δοθούν τα προσδοκώμενα δεδομένα-αποτελέσματα.
- Διακοπή της σύνδεσης με το Flickr API μόλις ληφθούν τα δεδομένα που ζητήσαμε από την υπηρεσία

- Μορφοποίηση των δεδομένων που μας επιστράφηκαν από το Flickr ώστε να είναι δυνατός ο μετέπειτα συνδυασμός τους με τα δεδομένα των άλλων πηγών
- Αποστολή της λέξης-κλειδιού και άλλων απαραίτητων δεδομένων στο API του Instagram ώστε να επιτευχθεί η σύνδεση με αυτό για την χρησιμοποίηση των υπηρεσιών του και να δοθούν τα δεδομένα που χρειαζόμαστε.
- Διακοπή της σύνδεσης με το Instagram API μόλις ληφθούν τα δεδομένα που ζητήσαμε από την υπηρεσία
- Μορφοποίηση των δεδομένων που μας επιστράφηκαν από το Instagram ώστε να είναι δυνατός ο μετέπειτα συνδυασμός τους με τα δεδομένα των άλλων πηγών.
- Αποστολή της λέξης-κλειδιού και άλλων απαραίτητων δεδομένων στο API του Picasa ώστε να επιτευχθεί η σύνδεση με αυτό για την χρησιμοποίηση των υπηρεσιών του και να δοθούν τα δεδομένα που χρειαζόμαστε.
- Διακοπή της σύνδεσης με το Picasa API μόλις ληφθούν τα δεδομένα που ζητήσαμε από την υπηρεσία
- Μορφοποίηση των δεδομένων που μας επιστράφηκαν από το Picasa ώστε να είναι δυνατός μετά ο συνδυασμός τους με τα δεδομένα των άλλων πηγών
- Συνδυασμός και επεξεργασία των δεδομένων και από τις τρεις πηγές με σκοπό την έξοδο τους στον χρήστη.
- Έξοδος στον χρήστη των αποτελεσμάτων με δυνατότητα επιλογής εισαγωγής τους σε άλλη υπηρεσία.

Στο σημείο αυτό και πριν συνεχίσουμε με την αναλυτική περιγραφή της κάθε πλατφόρμας και της υλοποίησης της εφαρμογής τίθεται απαραίτητο να υπάρξει και μια επιγραμματική αναφορά στις 3 εμπλεκόμενες web υπηρεσίες(flickr, picasa και instagram).

- Το Flickr (τυποποιημένο ως flickr και προφερόμενο ως "flicker") αποτελεί ένα δημοφιλή ιστότοπο κοινωνικού χαρακτήρα στον οποίο επιτελείται online διαχείριση εικόνων και εφαρμογών διαμοιρασμού τους. Το Flickr επιτρέπει στους χρήστες να κάνουν upload, να αποθηκεύουν και να μοιράζονται ψηφιακές εικόνες οι οποίες υπάρχει η δυνατότητα να έχουν χαρακτηριστεί από ετικέτες (ελεύθερα επιλεγμένες λέξεις-κλειδιά) προκειμένου

να βοηθήσουν την οργάνωση και την ανάκτηση τους. Τέτοιου είδους εικόνες υπάρχει η δυνατότητα να χαρακτηριστούν ως ιδιωτικές, να είναι ορατές σε επιλεγμένους φίλους και στην οικογένεια μόνο ή μπορούν να είναι ορατές από τη συνολική κοινότητα χρηστών του συστήματος. Εκτός του ότι είναι μια δημοφιλής ιστοσελίδα για τους χρήστες να μοιράζονται και να ενσωματώνουν προσωπικές φωτογραφίες, η υπηρεσία χρησιμοποιείται ευρέως από bloggers για να φιλοξενήσει τις εικόνες που ενσωματώνουν σε blogs και social media[19]. Το Yahoo ανέφερε τον Ιούνιο του 2011 ότι το Flickr είχε συνολικά 51 εκατομμύρια εγγεγραμμένα μέλη και 80 εκατομμύρια μοναδικούς επισκέπτες [20]. Επίσης τον Αύγουστο του 2011, η ιστοσελίδα ανέφερε ότι φιλοξενεί περισσότερα από 6 δισεκατομμύρια εικόνες και ο αριθμός αυτός συνεχίζει να αυξάνεται σταθερά, σύμφωνα με πηγές αναφοράς[21]. Φωτογραφίες και βίντεο μπορούν να προσεγγιστούν από το Flickr χωρίς την ανάγκη να καταχωρηθεί ο χρήστης μέσω λογαριασμού, αλλά ένας λογαριασμός πρέπει να γίνει προκειμένου να ανέβει περιεχόμενο στην ιστοσελίδα. Η δημιουργία ενός λογαριασμού, επίσης, επιτρέπει στους χρήστες να δημιουργήσουν μια σελίδα προφίλ που περιέχει φωτογραφίες και βίντεο που ο χρήστης έχει ανεβάσει και παρέχει τη δυνατότητα προσθήκης ενός άλλου χρήστη Flickr ως επαφή. Για να μπορέσει κάποιος να χρησιμοποιήσει το Flickr API χρειάζεται να διαθέτει ένα κλειδί της εφαρμογής. Αυτό χρησιμοποιείται από τους διαχειριστές του Flickr για να εντοπίζουν τη χρήση του API. Για να γίνει αναζήτηση των φωτογραφιών που πληρούν τα κριτήρια για το δοκιμαστικό mashup που δημιουργήθηκε για τις ανάγκες της πτυχιακής αυτής χρησιμοποιήθηκε η συνάρτηση flickr.photos.search του Flickr API. Η συγκεκριμένη συνάρτηση επιστρέφει μια λίστα φωτογραφιών οι οποίες πληρούν κάποια κριτήρια, επιστρέφοντας μόνο φωτογραφίες στις οποίες ο συγκεκριμένος χρήστης που κάνει την αναζήτηση έχει δικαιώματα πρόσβασης. Παρακάτω θα περιγραφουν κάποια από τα ορίσματα που μπορεί να δεχτεί η συνάρτηση. Ο αριθμός και το είδος των ορισμάτων ποικίλλει αναλόγως με το πόσο εξειδικευμένη αναζήτηση θέλει να κάνει ο συγκεκριμένος χρήστης. Τα ορίσματα που χρησιμοποιούνται συνήθως και χρησιμοποιήθηκαν και για την εφαρμογή είναι τα:

- api_key: Το κλειδί της API εφαρμογής
- tags: Η λίστα των ετικετών. Ο διαχωρισμός μεταξύ τους γίνεται με κόμματα. Θα επιστραφούν φωτογραφίες που έχουν χαρακτηριστεί με τουλάχιστον μία από τις ετικέτες που ανήκουν στη συγκεκριμένη λίστα.
- page: Η σελίδα των αποτελεσμάτων που επιστρέφονται. Εάν το όρισμα αυτό παραλειφθεί τότε η προεπιλεγμένη τιμή είναι 1.

- per_page: Ο αριθμός των φωτογραφιών που επιστρέφονται ανά σελίδα. Εάν αυτό το όρισμα παραλειφθεί έχει ως προεπιλεγμένη τιμή το 100. Η μέγιστη επιτρεπόμενη τιμή είναι το 500.
- Το Instagram είναι μια online υπηρεσία διαμοιρασμού φωτογραφιών και κοινωνικής δικτύωσης που επιτρέπει στους χρήστες της να βγάζουν φωτογραφίες, να εφαρμόζουν σε αυτές ψηφιακά φίλτρα, και να τις μοιράζονται σε μια ποικιλία από υπηρεσίες κοινωνικής δικτύωσης, όπως το Facebook ή το Twitter. Το Instagram δημιουργήθηκε από τον Kevin Systrom και τον Mike Krieger και ξεκίνησε τον Οκτώβριο του 2010[22]. Η υπηρεσία κέρδισε γρήγορα δημοτικότητα, με πάνω από 100 εκατομμύρια ενεργούς χρήστες ως τον Απρίλιο του 2012[23]. Για να μπορέσει κάποιος να χρησιμοποιήσει το Instagram API χρειάζεται να πραγματοποιήσει μια διαδικασία πιστοποίησης ώστε οι διαχειριστές του Instagram να εντοπίζουν τη χρήση του API. Για να γίνει αναζήτηση των φωτογραφιών που πληρούν τα κριτήρια για το δοκιμαστικό mashup που δημιουργήθηκε για τις ανάγκες της πτυχιακής αυτής χρησιμοποιήθηκε το URL-query [https://api.instagram.com/v1/tags/\[TAG\]/media/recent?client_id=\[ClientID\]&access_token=\[CODE\]](https://api.instagram.com/v1/tags/[TAG]/media/recent?client_id=[ClientID]&access_token=[CODE]) του Instagram API που επιστρέφει μια λίστα φωτογραφιών με τις πιο πρόσφατες φωτογραφίες που σχετίζονται με μια συγκεκριμένη λέξη που στο παραπάνω query δένεται ως [TAG] ενώ τα clientID και CODE παρέχονται στον χρήστη του API μέσω της διαδικασίας πιστοποίησης.
- Το Picasa αποτελεί έναν οργανωτή και προβολέα εικόνων για την οργάνωση και επεξεργασία ψηφιακών φωτογραφιών, καθώς και ένα ολοκληρωμένο website διαμοιρασμού φωτογραφιών. Η ονομασία "Picasa" είναι ένα μείγμα από το όνομα του ζωγράφου Pablo Picasso και της Ισπανικής φράση mi casa (που στα Ισπανικά μεταφράζεται ως το σπίτι μου) και του "pic" για τις εικόνες (εξατομικευμένη τέχνη[24]). Το Picasa Web Albums Data API επιτρέπει στις client εφαρμογές για να βλέπουν και να ενημερώνουν το περιεχόμενο των Picasa Web Albums , με τη μορφή του Google data API GData feeds. Η client-εφαρμογή έτσι μπορεί να ζητήσει μια λίστα με άλμπουμ κάποιου χρήστη , φωτογραφίες, σχόλια, ή ετικέτες, να επεξεργαστεί ή να διαγράψει το περιεχόμενο σε μια υπάρχουσα καταχώριση, και να διευρύνει το περιεχόμενου σε μια υπάρχουσα καταχώρηση. Για να γίνει αναζήτηση των φωτογραφιών που πληρούν τα κριτήρια για το δοκιμαστικό mashup που δημιουργήθηκε για τις ανάγκες της πτυχιακής αυτής χρησιμοποιήθηκε το query <http://picasaweb.google.com/data/feed/t/all> του Picasa Web Albums Data API και επιστρέφει μια λίστα φωτογραφιών οι οποίες πληρούν

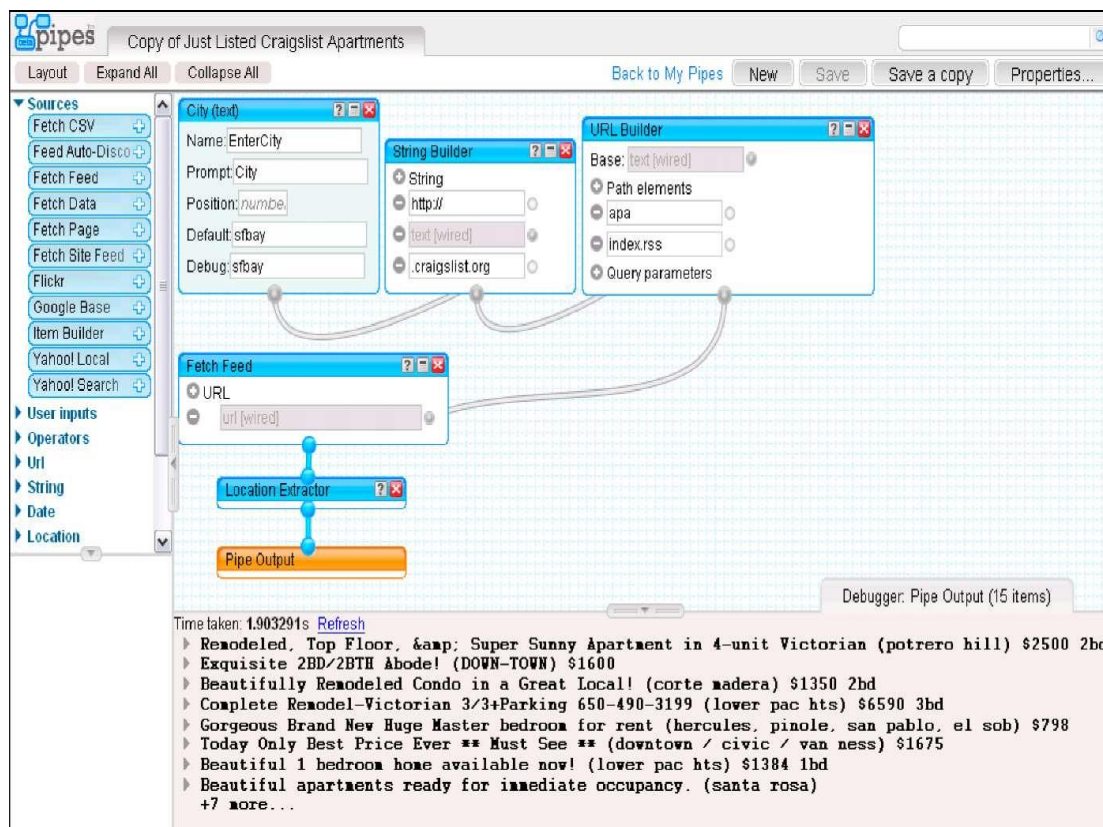
κάποια κριτήρια. Παρακάτω θα περιγραφθούν κάποιοι από τις τις παραμέτρους που μπορεί να δεχτεί το query με τον αριθμό και το είδος των παραμέτρων να ποικίλλει αναλόγως με το πόσο εξειδικευμένη αναζήτηση θέλει να κάνει ο χρήστης. Οι παράμετροι που χρησιμοποιούνται συνήθως και χρησιμοποιήθηκαν και στα πλαίσια της υλοποίησης της εφαρμογής είναι οι:

- kind: επιστρέφει το τύπο της λίστας που θα επιφορτωθεί και παίρνει τις τιμές album , comment , tag , user,photo.
- Max-results:καθορίζει τον μέγιστο αριθμό των αποτελεσμάτων που ανακτώνται.
- q: κάνει αναζήτηση στον τίτλο ,στην λεζάντα και στα tags των δεδομένων σύμφωνα με μια συγκεκριμένη τιμή συμβολοσειράς που δίνει ο χρήστης.

3.2 Υλοποίηση με το Yahoo! Pipes

3.2.1 Περιγραφή πλατφόρμας

Το Yahoo! Pipes όπως αναφέρθηκε και σε προηγούμενο κεφάλαιο είναι μια web-based οπτική γλώσσα προγραμματισμού για την κατασκευή mashups δεδομένων. Το Yahoo! Pipes αρχικά αναπτύχθηκε ως ένα εργαλείο για να κάνει την εξόρυξη, την συγκέντρωση, και την αναδημοσίευση στοιχείων από ολόκληρο το web ευκολότερη. Από την έναρξή του το Φεβρουάριο του 2007, πάνω από 90.000 προγραμματιστές έχουν δημιουργήσει επιμέρους pipes στην πλατφόρμα του Yahoo! Pipes, και τα pipes που εκτελούνται είναι πάνω από 5.000.000 κάθε μέρα. Το παρακάτω Σχήμα δείχνει το περιβάλλον επεξεργασίας του εργαλείου.



Εικόνα 3-1: Η επιφάνεια επεξεργασίας του Yahoo! Pipes αποτελείται από τέσσερις περιοχές: την γραμμή πλοήγησης, την εργαλειοθήκη, το καμβά και το πάνελ εντοπισμού σφαλμάτων και παραγωγής εξόδου

Το περιβάλλον αποτελείται από τέσσερις κύριες περιοχές: την γραμμή πλοήγησης στην κορυφή, την εργαλειοθήκη στα αριστερά, το καμβά εργασίας στο κέντρο, καθώς και το πάνελ εντοπισμού σφαλμάτων και παραγωγής εξόδου στο κάτω μέρος. Η εργαλειοθήκη περιλαμβάνει modules, τα δομικά στοιχεία της οπτική γλώσσας του Yahoo! Pipes. Η λογική του Yahoo! Pipes προέρχεται από την Unix γραμμή εντολών και της εντολής διασωλήνωσης, η οποία επιτρέπει στον χρήστη να συνδέσει μαζί μια σειρά από εντολές, όπου η έξοδος της μιας μεταβιβάζεται ως είσοδος στην επόμενη. Στη γραφική γλώσσα των Yahoo! Pipes, τα modules (operator) ορίζονται σε έναν καμβά σχεδίασης, μπορούν να έχουν μηδέν ή περισσότερες θύρες εισόδου, και όλα έχουν τουλάχιστον μία θύρα εξόδου. Επιπροσθέτως, τα modules μπορούν να έχουν παραμέτρους που μπορούν να ρυθμιστούν από τον προγραμματιστή, ή από την έξοδο άλλων module που ενσωματώνεται σαν παράμετρος, έτσι ώστε η τιμή της παραμέτρου να εξαρτάται από μια runtime τιμή που καθορίζεται αλλού. Οι είσοδοι και εξοδοι ενσωματώνονται μαζί, αντιπροσωπεύοντας τη ροή των δεδομένων της εφαρμογής. Επιλέγοντας μια έξοδο, επιδεικνύονται όλες οι συμβατές είσοδοι που ο χρήστης μπορεί να επιλέξει για να συνδεθεί με

την συγκεκριμένη έξοδο. Υπάρχουν διάφοροι τύποι δεδομένων εντός των Yahoo! Pipes που καθορίζουν ποιες εισοδοί και έξοδοι είναι συμβατές. Σε πιο γενικές γραμμές, υπάρχουν απλές βαθμισμένες τιμές δεδομένων, και items, τα οποία είναι σύνολα αντικειμένων δεδομένων (π.χ., τα items σε ένα RSS feed, ή κόμβοι σε ένα έγγραφο XML). Οι τιμές τους περιλαμβάνουν είδη όπως κείμενο, URLs, περιοχές, αριθμούς, ημερομηνίες και ώρες. Επίσης στο Yahoo! Pipes, οι ροές δεδομένων από το αρχικό module(ή modules), όπου ο χρήστης δίνει δεδομένα εισόδου, ή ανακτώνται τα εξωτερικά δεδομένα, μέσω επόμενων module με το τρόπο και την σειρά που υπογορεύεται από το διάγραμμα συνδεσμολογίας. Όλες οι εφαρμογές έχουν ένα μοναδικό module εξόδου, το οποίο είναι συνδεδεμένο με το τέλος της ακολουθίας εκτέλεσης, και συλλέγει την τελική ροή δεδομένων για τη διανομή τους μέσω RSS, JSON (JavaScript Object Notation), ή μια ποικιλία από άλλες μορφές. Με βάση πάλι την χρησιμοποίηση του Unix παραλληλισμού, η έξοδος είναι ανάλογη με την «τυποποιημένη έξοδο» ή το τερματικό του χρήστη. Σε αντίθεση με τη βασική ιδέα των Unix Pipes, το Yahoo! Pipes επιτρέπει στους χρήστες να καθορίσουν πολύπλοκες δομές διακλάδωσης και επανάληψης, χρησιμοποιούν πολλαπλές πηγές, τα μονοπάτια εκτέλεσης εκτελούνται παράλληλα, και σε γενικές γραμμές, δημιουργούν προγράμματα αυθαίρετης πολυπλοκότητας. Δεν υπάρχει καμία άμεση μέθοδος για τη σύνταξη αναδρομικών συναρτήσεων, και το Yahoo! Pipes δεν επιτρέπει κύκλους στη δομή του προγράμματος (δηλαδή, η έξοδος ενός module δεν επιτρέπεται να ανατροφοδοτείται στην είσοδο περαιτέρω module αν αυτά είναι αναδρομικά). Αυτό επιβάλλει μια κατά κύριο λόγο γραμμική ροή εκτέλεσης των εφαρμογών που οροθετείται από την ποσότητα των δεδομένων που υποβάλλονται σε επεξεργασία. Σε κάθε pipe εφαρμογή δίνεται μια μοναδική ταυτότητα και διεύθυνση URL. Οι χρήστες μπορούν να δημοσιεύουν τα Pipes στο δημόσιο κατάλογο, όπου μπορεί να γίνει αναζήτηση τους, περιήγηση τους, και να προβληθούν από οποιονδήποτε. Έχει λοιπόν ένα πολύ ανοικτό μοντέλο ασφάλειας, επιτρέποντας σε κάθε χρήστη να δει και να τρέξει οποιοδήποτε pipe, εφ' όσον γνωρίζει τη διεύθυνση URL, ακόμη και αν δεν έχει δημοσιευθεί στον κατάλογο. Αυτός ο σχεδιασμός έγινε εκ προθέσεως, καθώς οι προγραμματιστές της Yahoo ήθελαν να προωθήσουν το είδος της μάθησης-από-παράδειγμα.. Έτσι, κάθε pipe που δημιουργείται έχει ένα "edit source" κουμπί που επισυνάπτεται σε αυτό, επιτρέποντας στους χρήστες να ελέγχουν το πώς ένα pipe λειτουργεί. Αυτό επιτρέπει στους χρήστες όχι μόνο να μοιράζονται τα links κατά την εξέλιξη τους, καθώς και αδημοσίευτα Pipes, αλλά να βλέπουν και να τροποποιούν κάθε pipe των άλλων χρηστών. Επιτρέποντας στους χρήστες να κάνουν debug συλλογικά. Όταν ένα pipe τίθενται επεξεργασίας από κάποιον που δεν είναι ο ιδιοκτήτης του, ένα τοπικό αντίγραφο γίνεται στο πρόγραμμα περιήγησης του θεατή. Οι αλλαγές που γίνονται από τον θεατή αποθηκεύονται σε ένα νέο αντίγραφο στο διακομιστή, διατηρώντας το αρχικό pipe. Επιπλέον, κάθε pipe έχει ένα "clone" κουμπί, το οποίο δημιουργεί ένα αντίγραφο

του υπάρχοντος pipe .Το κουμπί αυτό καθιστά δυνατόν τη δημιουργία αντιγράφων των pipe του χρήστη ή και των άλλων ανθρώπων. Εκτός του ότι ολόκληρα τα pipe είναι copyable και τροποποιήσιμα, μπορούν να ενσωματωθούν μέσα σε ένα άλλο, ως «υπο-pipes». Αυτό επιτρέπει στους προγραμματιστές να δημιουργούν και να μοιράζονται επαναχρησιμοποιήσιμα κατασκευαστικά στοιχεία, και να δημιουργούν ενδιάμεσα επίπεδα αφαίρεσης στις εφαρμογές τους. Ένα ενσωματωμένο sub-pipe αναπαρίσταται ως ένα module στο Yahoo! Pipes interface, το οποίο μπορεί να συνδεθεί με άλλα module, ή άλλες sub-pipes.

Αναλύοντας λίγο πιο επισταμένα τι παρέχει βλέπουμε ότι το Yahoo! Pipes υποστηρίζει κυρίως REST υπηρεσίες web, αλλά παρέχει επίσης ειδικές ενότητες για την πρόσβαση σε υπηρεσίες όπως το Flickr για την αναζήτηση φωτογραφιών με λέξεις-κλειδιά και γεωγραφική θέση, το Google Base, που επιτρέπει σε οποιονδήποτε να δημιουργήσει και να δημοσιεύσει στο διαδίκτυο προσβάσιμες πληροφορίες, το Yahoo Local για την αναζήτηση υπηρεσιών σε μια συγκεκριμένη περιοχή, το Yahoo Search για την δημιουργία προσαρμοσμένων αναζητήσεων ως σημείο εκκίνησης για την δημιουργία Pipes και την μετατροπή σε πηγή μιας συγκεκριμένης ιστοσελίδας (Fetch Module Page) ή ενός αρχείου CSV (Fetch CSV Module). Για να συνδυάσει τα feeds δεδομένα, το Yahoo Pipes μετατρέπει τις μορφές των πηγών του (οι οποίες μπορεί να είναι RSS, Atom ή RDF) σε εσωτερικό μοντέλο δεδομένων RSS feed. Το mapping τους είναι ημιαυτόματο, εάν το όνομα των πεδίων εισόδου ενός feed ταιριάζει με το όνομα των πεδίων RSS, η μετατροπή του γίνεται αυτόματα αλλιώς παρέχονται πολλές διαφορετικές διευκολύνσεις για να βοηθήσουν τον χρήστη για το mapping των δεδομένων. Ένα παράδειγμα είναι το «Item Building» module το οποίο χρησιμοποιείται για την αναδιάρθρωση και μετονομασία πολλών στοιχείων του feed, ώστε να μετατρέπεται το εξωτερικό feed στην μορφή του εσωτερικού RSS μοντέλου δεδομένων.

Για να αναδιαρθρώσει το σχήμα των εισερχόμενων δεδομένων, το Pipes παρέχει τρεις φορείς που περιγράφονται στον Πίνακα 3-1. Για την προδιαγραφή της ροής των δεδομένων, το Yahoo Pipes παρέχει μόνο το module «Union» που συνδυάζει μια λίστα στοιχείων σε ένα ενιαίο κατάλογο.

Operator	Description
Regex	επιτρέπει την τροποποίηση των πεδίων σε ένα RSS feed χρησιμοποιώντας κανονικές εκφράσεις.
Rename	χρησιμοποιείται για την μετονομασία των στοιχείων του feed εισόδου και για την προσθήκη νέων στοιχείων στις ροές των feeds. .
Sub-Element	επιτρέπει την εξαγωγή επιμέρους στοιχείων από τα feeds, που είναι “θαμμένα” στην ιεραρχία της.
Union	επιτρέπει τον συνδυασμό μιας λίστας στοιχείων σε ένα ενιαίο κατάλογο

Πίνακας 3-1: operators της ροής δεδομένων που προσφέρονται από το Yahoo Pipes

Εκτός αυτού, για να εκτελέσει μια επεξεργασία για το σύνολο των δεδομένων, διατίθενται οι παρακάτω τελεστές που περιγράφονται στον Πίνακα 3-2.

Operator	Description
Reverse	Εάν τα feeds έχουν ταξινομηθεί, το reverse module παρέχει έναν τρόπο για να αλλάξει η σειρά των feeds, αναστρέφοντας τη σειρά όλων των στοιχείων σε μια ροή δεδομένων.
Sort	Χρησιμοποιείται για να ταξινομηθεί ένα feed με βάση οποιοδήποτε στοιχείο του, όπως ο τίτλος. Τα στοιχεία μπορούν να ταξινομηθούν σε αύξουσα ή φθίνουσα σειρά.
Truncate	Το module αυτό επιστρέφει ένα καθορισμένο αριθμό στοιχείων από την κορυφή ενός feed.
Tail	Το module αυτό περικόπτει ένα feed κρατώντας τα τελευταία N στοιχεία του, όπου N είναι ένας αριθμός που καθορίζετε.
Count	Το module αυτό μετρά τον αριθμό των στοιχείων του feed των εισροών και δίνει σαν έξοδο τον αριθμό αυτο.
Filter	Ο Filter operator χρησιμοποιείται για την εξαγωγή συγκεκριμένων στοιχείων από ένα feed που πληρούν την συνθήκη του Filter
Unique	Το module αυτό καταργεί items που περιέχουν διπλά string.
String Operators	Αυτά τα modules βοηθούν στο να χειραγωγούνται και να συνδυάζονται strings
Simple Math	Αυτο το module εκτελεί βασικές αριθμητικές πράξεις, όπως πρόσθεση και αφαίρεση.
Date Operators	Αυτά τα modules μπορούν να εκτελέσουν την επεξεργασία μιας ημερομηνίας καθώς μπορούν να δημιουργήσουν ένα date object από μια αλφαριθμητική τιμή.
URL	Αυτό το module δημιουργεί URLs βασιζόμενο στο κλασικό τρόπο ή στο Web 2.0 query-string format από μια σειρά πεδίων εισαγωγής.

Πίνακας 3-2: operators της ροής δεδομένων που προσφέρονται από το Yahoo Pipes

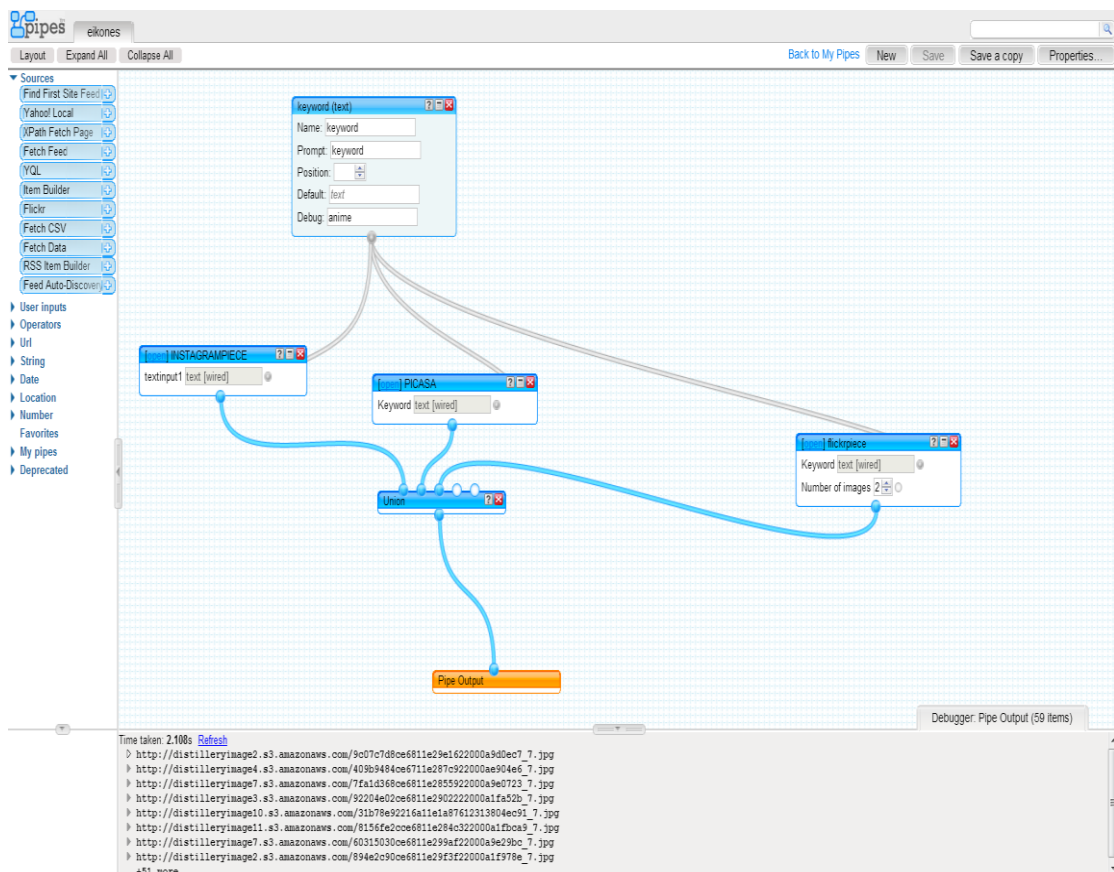
Τα pipes αποθηκεύουν όλα τα feeds που επισκέπτονται στο δικό τους server. Όπως προαναφέρθηκε τα δημιουργημένα pipes φιλοξενούνται στο server του Yahoo και μπορούν είτε να προσπελαστούν από έναν client RSS ή JSON μέσω της μοναδικής διεύθυνσης URL τους ή να εμφανίζονται στον παρεχόμενο yahoo map. Εκτός αυτού, τα pipes μπορούν να χρησιμοποιηθούν σαν στοιχεία Mashup για να οικοδομηθεί ένα πιο σύνθετο pipe ή τα αποτελέσματα τους μπορούν να συνδυαστούν με άλλα εργαλεία που μπορούν να επεξεργαστούν RSS feeds.

Το Yahoo Pipes είναι ένα επεκτάσιμο εργαλείο. Εάν δεν προσφέρονται κάποιες λειτουργίες για τις ανάγκες των τελικών χρηστών, ο χρήστης μπορεί να δημιουργήσει μια υπηρεσία web και να την καλέσει από το σύστημα μέσω της υπηρεσίας διασύνδεσης Web. Η εξωτερική υπηρεσία είναι προσβάσιμη μέσω JSON Interface και η έξοδος της πρέπει να είναι ένας τύπος δεδομένων που υποστηρίζεται από το εργαλείο. Η προστιθέμενη λειτουργικότητα είναι ορατή μόνο στον ιδιοκτήτη και δεν μπορεί να μοιραστεί με ολόκληρη την κοινότητα.

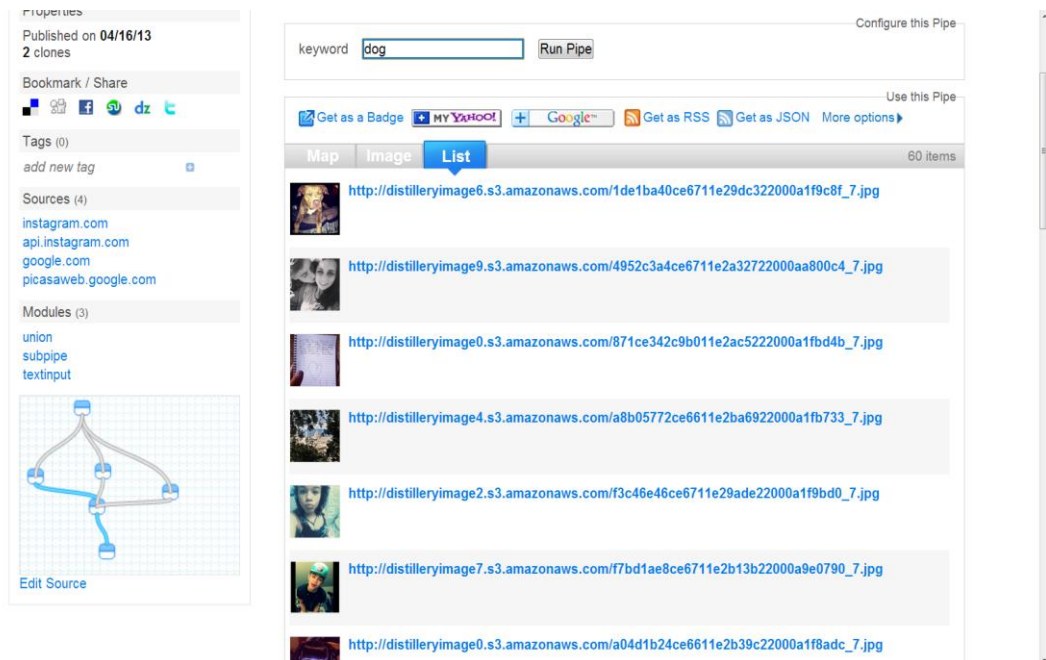
3.2.2 Περιγραφή υλοποίησης

Η πειραματική εφαρμογή που αναπτύχθηκε με το Yahoo! Pipes είχε ως σκοπό την αναζήτηση φωτογραφιών από τρεις πηγές(flickr,Instagram, picasa). Συγκεκριμένα ο ενδιαφερόμενος εισάγοντας μια λέξη-κλειδί ο μηχανισμός εμφανίζει αποτελέσματα-εικόνες από τις τρεις προαναφερόμενες πηγές, συνδυάζοντάς τα και προσθέτοντας δίπλα στην κάθε εικόνα τον σύνδεσμο της με την πηγή της.

Για τη δημιουργία της παραπάνω εφαρμογής Mashup αρχικά επιλέγεται το module «text input», στο οποίο ο επισκέπτης εισάγει την λέξη κλειδί για την οποία γίνεται η αναζήτηση, το λήμμα αυτό της αναζήτησης στη συνέχεια συνδέεται με τα δημιουργημένα sub-pipes modules τα οποία επιτελούν την αναζήτηση το κάθε ένα για μια εκ των τριών πηγών. Ενώ στην συνέχεια, όπως έχει αναφερθεί και στην σχετική ενότητα όπου περιγράφεται το Pipes, η συνένωση των αποτελεσμάτων κάθε sub-pipe module γίνεται μέσω του «Union» module με την έξοδο της εφαρμογής να λαμβάνεται συνδέοντας το module αυτό με το «Pipe Output» που αποτελεί και το τέλος της εφαρμογής ώστε να παρθούν τα τελικά αποτελέσματα. Τα παραπάνω και τα αποτελέσματα φαίνονται στις παρακάτω εικόνες.



Εικόνα 3-2: Υλοποίηση της εφαρμογής στο Yahoo! Pipes

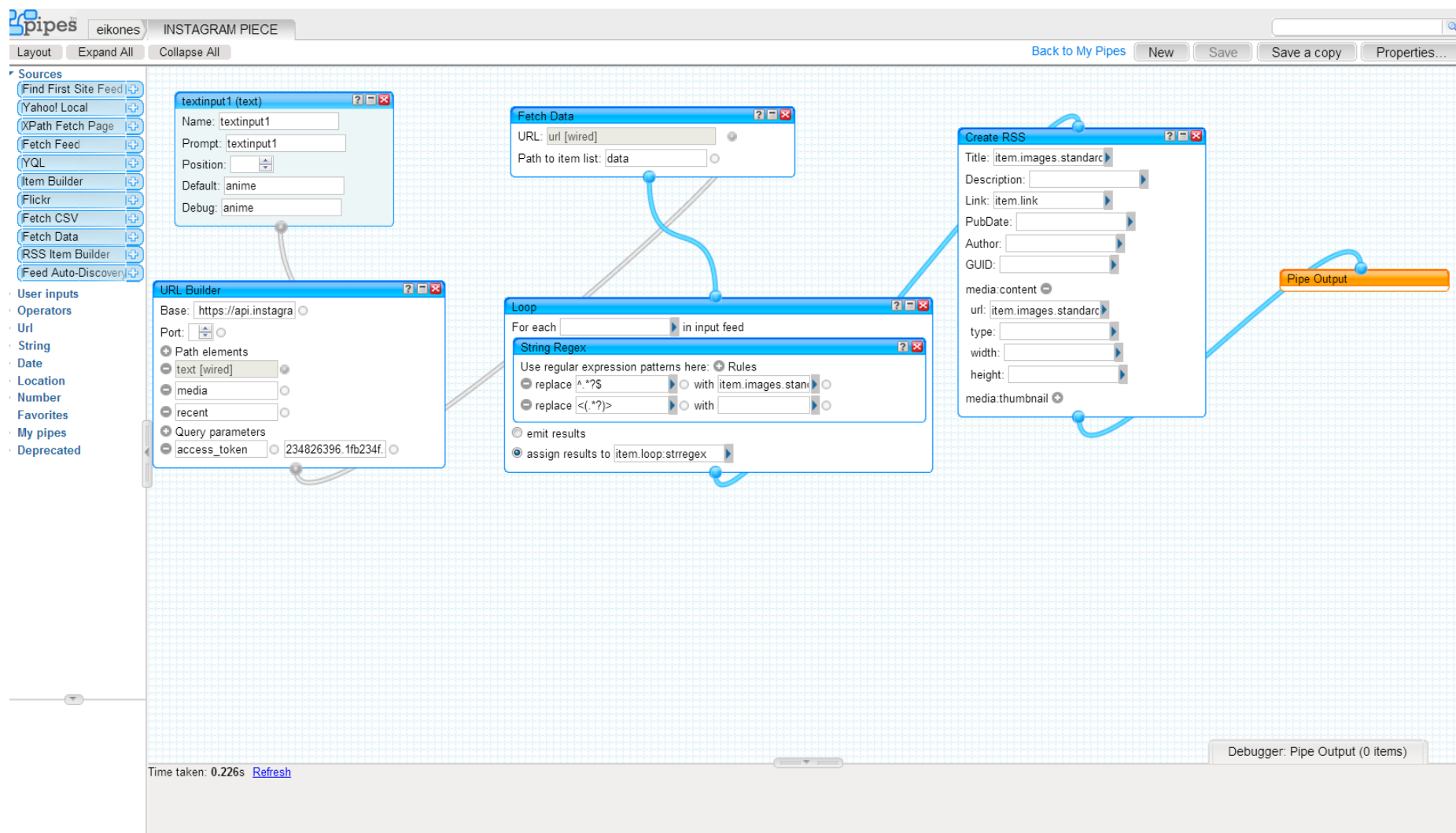


Εικόνα 3-3: Αποτελέσματα της εκτέλεσης της εφαρμογής

Όσον αφορά κάθε sub-pipe module γίνεται αναλυτική περιγραφή κάθε module παρακάτω:

Instagram module

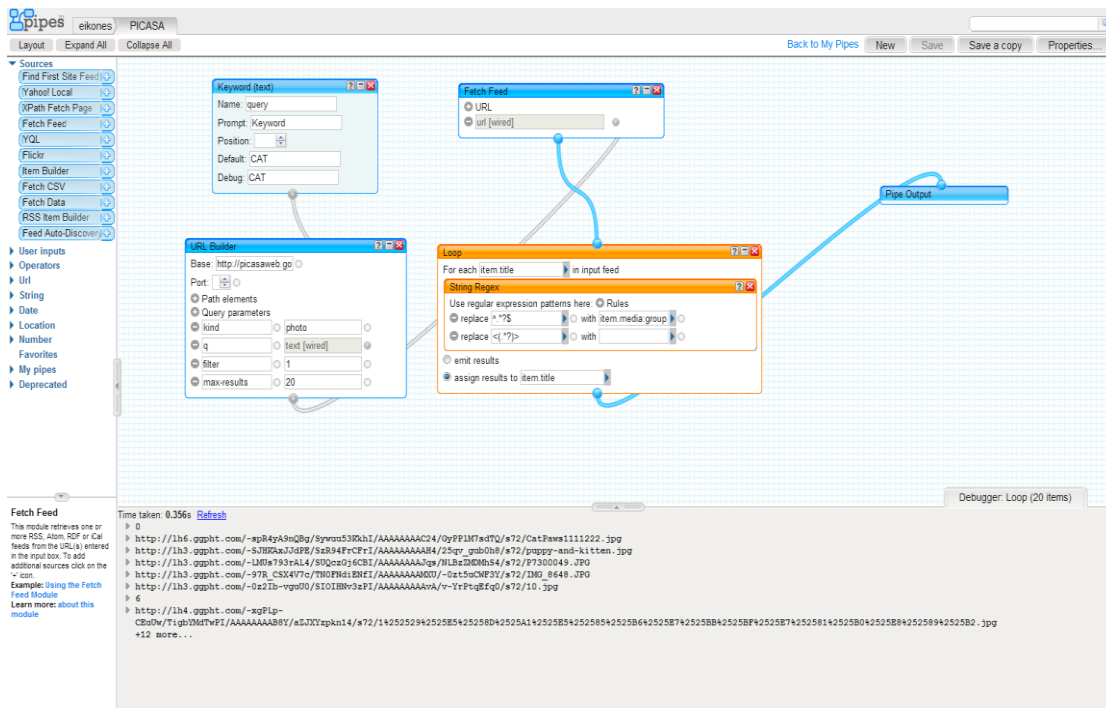
Και εδώ η εκκίνηση της διαδικασίας ξεκινά μέσω ενός module τύπου «text input» ώστε να πραγματοποιηθεί η εισαγωγή της λέξης- κλειδιού σύμφωνα με την οποία πραγματοποιείται η αναζήτηση στο Instagram. Στη συνέχεια συνδέεται η λέξη-κλειδί με το module «URL Builder» όπου δημιουργείται το query-URL που πραγματοποιεί την σύνδεση με το API του Instagram ,παρέχοντας τις κατάλληλες παραμέτρους που χρειάζεται για την σύνδεση, ώστε στη συνέχεια μέσω του module «Fetch Data» η απάντηση από το σταλμένο query να μετατρέπεται από μορφή JSON σε μια λίστα από στοιχεία που βασίζονται σε κάποια παράμετρο. Έπειτα πραγματοποιείται σύνδεση αυτής της λίστας με το «Loop» module που περιέχοντας το module «String Regex» επι της ουσίας πραγματοποιεί μια εκκαθάριση της προαναφερθείσας λίστας κρατώντας μόνο τις τιμές που μας ενδιαφέρουν για κάθε αντικείμενο της λίστας επαναληπτικά. Τέλος γίνεται σύνδεση της εξόδου αυτού του module με το «Create RSS» module όπου δημιουργείται μια ροή RSS με τις παραμέτρους της λίστας αντικειμένων που παίρνει σαν είσοδο και επιλέγοντας σαν έξοδο της ροής την εικόνα και τον σύνδεσμο κάθε αντικειμένου οδηγώντας έτσι στην ολοκλήρωση του sub-pipe δένοντας ως έξοδο του την συγκεκριμένη όρη. Η υλοποίηση που περιγράφηκε φαίνεται στην επόμενη εικόνα.



Εικόνα 3-4: Υλοποίηση του Instagram. Μέρος της εφαρμογής στο Yahoo! Pipes

Picasa module

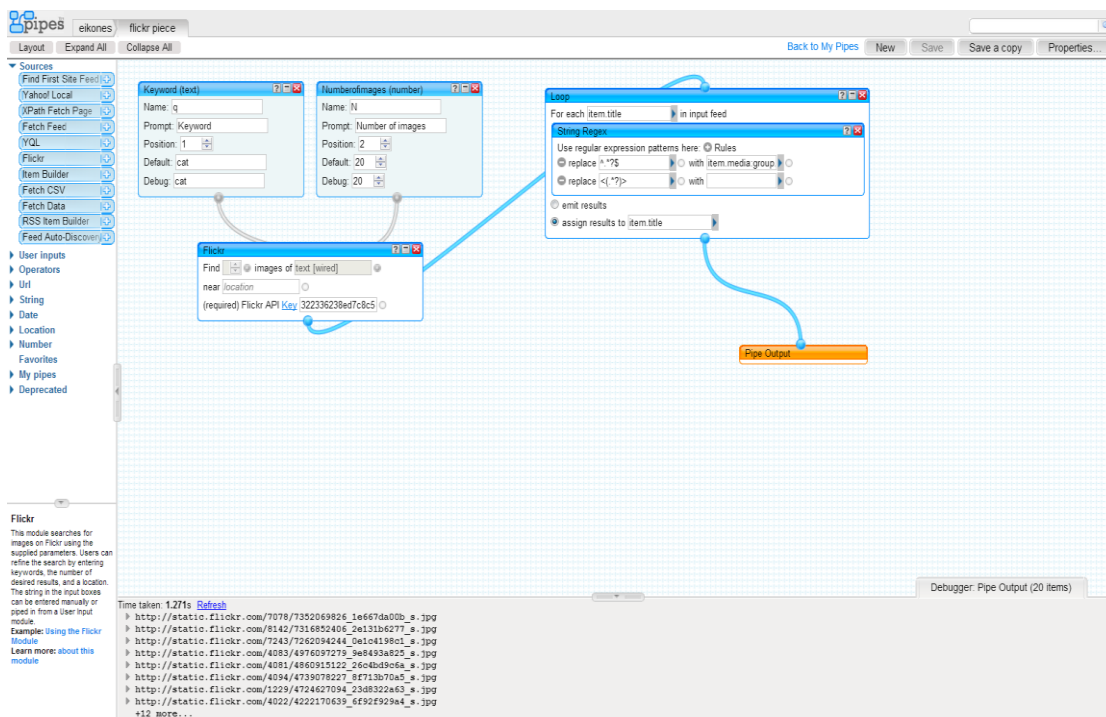
Και σε αυτή την περίπτωση η εκκίνηση της διαδικασίας ξεκινά μέσω ενός module τύπου «text input» ώστε να πραγματοποιηθεί η εισαγωγή της λέξης-κλειδιού σύμφωνα με την οποία πραγματοποιείται η αναζήτηση στο Instagram. Στη συνέχεια συνδέεται η λέξη- κλειδί με το module «URL Builder» όπου δημιουργείται το query-URL που πραγματοποιεί την σύνδεση με το API του picasa ,παρέχοντας τις κατάλληλες παραμέτρους που χρειάζεται για την σύνδεση, ώστε στη συνέχεια μέσω του module «Fetch Feed» να λαμβάνεται από το URL η απάντηση σε μορφή ροής RSS σε αυτή τη περίπτωση. Τέλος η ροή αυτή συνδέεται με το module «Loop» όπου και πάλι εσωτερικά του περιλαμβάνει το module «String Regex» που επιτελεί φιλτράρισμα των δεδομένων της ροής επαναληπτικά για κάθε στοιχείο της και η έξοδος του «Loop» module ταξινομεί βάση του τίτλου κάθε στοιχείου της ροής. Η υλοποίηση που περιεγράφηκε φαίνεται στην παρακάτω εικόνα.



Εικόνα 3-5: Υλοποίηση του Picasa μέρους της εφαρμογής στο Yahoo! Pipes

Flickr module

Στην περίπτωση του flickr το Yahoo! Pipes παρέχει ένα σχεδόν αυτοματοποιημένο module που κάνει αναζήτηση στο flickr βάση συγκεκριμένων παραμέτρων που συμπληρώνονται στο module. Έτσι για την υλοποίηση της αναζήτησης που χρειάζεται η εφαρμογή το sub-pipe ξεκινά με δυο module, με ένα «Text Input» module που παίρνει σαν είσοδο την λέξη-κλειδί και ενός «Number Input» module που του ορίζεται σαν προεπιλεγμένη τιμή το 20 και που θα χρησιμοποιηθούν σαν είσοδοι στο module «flickr» αποτελώντας παραμέτρους του με το πρώτο module να δίνει την λέξη σύμφωνα με την οποία θα πραγματοποιηθεί η αναζήτηση και το δεύτερο να δίνει τον αριθμό των αποτελεσμάτων που θα επιστρέψει το module. Τέλος η RSS ροή που επιστρέφει το module του flickr εισάγεται και σε αυτήν την περίπτωση στο module «Loop» που περικλείει και πάλι το module «String Regex» φιλτράροντας τα δεδομένα της ροής επιστρέφοντας τα και εισάγοντας τα σαν είσοδο στο «Pipe Output» ολοκληρώνοντας το sub-pipe. Η υλοποίηση που περιεγράφηκε φαίνεται στην παρακάτω εικόνα.



Εικόνα 3-6: Υλοποίηση του Flickr μέρους της εφαρμογής στο Yahoo! Pipes

3.3 Υλοποίηση με το WebHookIt

3.3.1 Περιγραφή πλατφόρμας

Το WebHookIt είναι ένα open-source έργο ακόμα σε alpha στάδιο ως προς την ανάπτυξη του, που κυκλοφόρησε υπό την άδεια του MIT τον Ιανουάριο του 2011. Αποτελεί μια πλατφόρμα Visual scripting για το web χρησιμοποιώντας τα Node.js και MongoDB παρέχοντας ένα οπτικό μοντέλο προγραμματισμού για τη δημιουργία απλών webservices, mashups, εντάξεις web εφαρμογών, συνδέσεις μεταξύ webservices, event-driven προγραμματισμό web, εν συντομία παρέχει δυνατότητες προγραμματισμού του διαδικτύου. Θα μπορούσε επίσης να χαρακτηριστεί σαν open-source Yahoo! Pipes, αλλά με ένα ευρύτερο πεδίο εφαρμογών. Η κεντρική ιδέα του βασίζεται στο να έχουμε μια πολύ HTTP-centric visual γλώσσα ώστε να παραχθεί ένας κώδικας-κόλλα μεταξύ web-υπηρεσιών. Είναι φτιαγμένο ώστε να υπάρχει η δυνατότητα προσαρμογής, τροποποίησης και βελτίωσης του ,προσθέτοντας του βασικά modules(γραμμένα σε Node.js), προσαρμόζοντας το UI του (forms και modules, χρησιμοποιώντας τις βιβλιοθήκες του WireIt και του inputEx) και τη μηχανή εκτέλεσης του και τέλος συνθέτοντας και επαναχρησιμοποιώντας τα modules. Λόγω των παραπάνω είναι ιδανικό για ειδοποιήσεις (e-mail, sms, IM, κοινωνικά δίκτυα),την ενσωμάτωση και επικοινωνία υπηρεσιών , plugins εφαρμογών web (Webhooks),αλυσιδωτή σύνδεση web-services και παροχή απλών webservices.Δεν είναι μια φιλοξενούμενη υπηρεσία, όπως το Yahoo! Pipes, αλλά μπορεί να εγκατασταθεί σε τοπικό επίπεδο στον υπολογιστή του χρήστη ή σε κάποιο server. Όπως προαναφέρθηκε το webhookit χρησιμοποιεί τα MongoDB και Node.js οπότε η εγκατάσταση του προϋποθέτει και την εγκατάσταση τους καθώς επίσης και κάποιων javascript βιβλιοθηκών που συμβάλλουν στην δημιουργία και την εκτέλεση του και εμπεριέχονται στον κώδικα-υλοποίηση της πλατφόρμας. Πιο αναλυτικά το webhookit αποτελείται από:

Το περιβάλλον διεπαφής του , με βασικο συστατικό του τον visual-editor που θα αναλυθεί παρακάτω, στηρίζεται στις ακόλουθες βιβλιοθήκες:

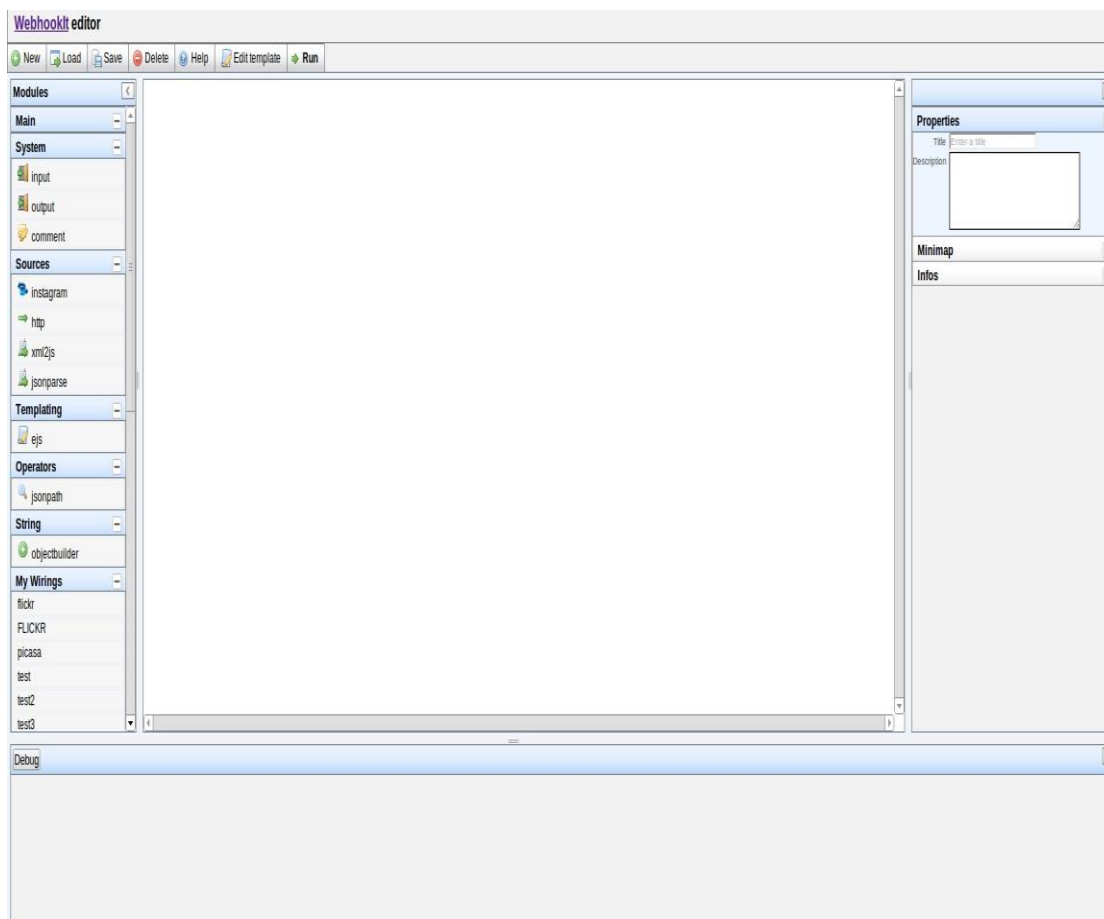
- Το WireIt μια open-source βιβλιοθήκη JavaScript, για τη δημιουργία πλήρους web graph editors για εφαρμογές ροών δεδομένων, οπτικές γλώσσες προγραμματισμού, γραφική μοντελοποίηση ή editor γραφημάτων. Στην περίπτωση του webhookit χρησιμοποιούνται ο WiringEditor, Wires και άλλα widgets της που βασίζονται στην YUI βιβλιοθήκη.
- Το inputEx ένα open-source πλαίσιο javascript για την οικοδόμηση πεδίων και φορμών για τις εφαρμογές web που χρησιμοποιούν τη YUI3 βιβλιοθήκη. Όλα τα πεδία και οι φόρμες διαμορφώνονται χρησιμοποιώντας JSON ή δημιουργούνται μέσω Javascript.

Στην περίπτωση του webhookit μέσω του inputEx δημιουργούνται τα απαραίτητα forms, fields, και widgets που χρειάζεται η εφαρμογή.

Εν συνεχεία από την πλευρά του διακομιστή το webhookit στηρίζεται στα ακόλουθα:

- Το MongoDB (από το "humongous") ένα open source [document-oriented](#) σύστημα βάσης δεδομένων υπό την ανάπτυξη και την υποστήριξη της εταιρείας [10gen](#). Αντί για την αποθήκευση των δεδομένων σε πίνακες όπως γίνεται σε μια «κλασική» σχεσιακή βάση δεδομένων, η MongoDB αποθηκεύει δομημένα δεδομένα σε JSON τύπου έγγραφα με δυναμικά σχήματα (η MongoDB καλεί το format BSON), καθιστώντας την ολοκλήρωση των δεδομένων σε ορισμένους τύπους εφαρμογών ευκολότερη και ταχύτερη
- Το Node.js ένα server-side σύστημα λογισμικού που έχει σχεδιαστεί για τη σύνταξη κλιμακούμενων εφαρμογών Internet, κυρίως web servers. Τα προγράμματα είναι γραμμένο στην πλευρά του διακομιστή σε JavaScript, χρησιμοποιώντας event-driven, ασύγχρονη I / O για την ελαχιστοποίηση των γενικών εξόδων και τη μεγιστοποίηση της κλιμάκωσης. Το Node.js περιέχει μια ενσωματωμένη βιβλιοθήκη διακομιστή HTTP, καθιστώντας τη δυνατή να τρέξει ένα web server χωρίς τη χρήση εξωτερικού λογισμικού, όπως Apache , και επιτρέπει περισσότερο έλεγχο του τρόπου λειτουργίας του web server. Το Node.js επιτρέπει σε web developers να δημιουργήσουν μια ολόκληρη διαδικτυακή εφαρμογή JavaScript, είτε server-side είτε client-side στην περίπτωση του webhookit server-side.
- Το npm που αποτελεί έναν διαχειριστή πακέτων για το Node.js και τα πακέτα-module του webhookit.
- Το ExpressJS που αποτελεί ένα RESTful website framework.
- Την βιβλιοθήκη Mongo-native στην ουσία τον MongoDB driver-wrapper για το Node.js ώστε να γίνεται η μεταξύ τους σύνδεση.
- Την βιβλιοθήκη cron που αποτελεί έναν χρονοπρογραμματιστή για τις δημιουργούμενες εφαρμογές μας
- Την βιβλιοθήκη του EJS που αποτελεί ένα ενσωματωμένο σύστημα προτυποποίησης Javascript και στη περίπτωση του webhookit χρησιμοποιείται για την εναλλακτική αναπαράσταση των αποτελεσμάτων των εξόδων των παραγόμενων εφαρμογών.

Το παρακάτω Σχήμα δείχνει το περιβάλλον του visual editor του εργαλείου.



Εικόνα 3-7: Το περιβάλλον του editor του WebHookIt

Το περιβάλλον του editor αποτελείται από τέσσερις κύριες περιοχές: την γραμμή πλοήγησης στην κορυφή, την εργαλειοθήκη στα αριστερά, το καμβά εργασίας στο κέντρο, καθώς και το πανελ εντοπισμού σφαλμάτων και παραγωγής εξόδου στο κάτω μέρος.

Η εργαλειοθήκη περιλαμβάνει modules, τα δομικά στοιχεία της οπτική γλώσσας του WebHookIt. Τα modules που διατίθενται δίνονται στο επόμενο πίνακα :

Module	Περιγραφή
Input	προσθέτει μια νέα καταχώρηση στους παραμέτρους της εφαρμογής. Κάθε input ορίζει ένα πεδίο φόρμας χρησιμοποιώντας τη βιβλιοθήκη inputEx.
Output	αποτελεί τις επιστρεφόμενες τιμές της εφαρμογής. Μια εφαρμογή μπορεί να έχει πολλαπλά output
comment	Εισαγάγει σχόλια στην εφαρμογή
http	Εκτελεί http GET , POST ,DELETE και HEAD σε συγκεκριμένα url παρέχοντας τις κατάλληλες παραμέτρους και επιλέγοντας τον encoding των αποτελεσμάτων που επιστέφονται
xml2js	Μετατρέπει xml δεδομένα που πιθανώς παράγονται από εξόδους module σε μορφή json
jsonparse	Μετατρέπει τα json δεδομένα που παίρνει σαν είσοδο σε αντικείμενα
ejs	Εισαγάγει εναλλακτική μορφή template εξόδου
jsonpath	Εξαγάγει μέρη του json αρχείου που χρειαζόμαστε απορρίπτοντας ότι δεν χρειάζεται
objectbuilder	Δημιουργεί αντικείμενα από δεδομένα των υπολοίπων module με σκοπό την περαιτέρω επεξεργασία τους

Πίνακας 3-3: Modules δημιουργίας mashup εφαρμογών του WebHookIt

Πέρα αυτών των module υπάρχει επίσης η δυνατότητα δημιουργίας καινούριων module και η ενσωμάτωση τους και αυτών στην εργαλειοθήκη. Πιο συγκεκριμένα διατίθεται η επιλογή μεταξύ δυο ειδών module, αυτών που έχουν προκύψει από υλοποιημένες εφαρμογές σαν

συνδυασμούς των παραπάνω modules και module-πυρήνα δηλαδή module που δημιουργούνται και εισάγονται στο webHookIt μέσω javascript υλοποιήσεων.

Η γραμμή πλοήγησης πέρα των κλασσικών δυνατοτήτων(new,load,save,delete) δίνει την δυνατότητα επεξεργασίας του template της εξόδου της εφαρμογής καθώς επίσης παρέχει και το κουμπί που εκτελεί την εφαρμογή

Το πάνελ εντοπισμού σφαλμάτων και παραγωγής εξόδου προσφέρει αποσφαλμάτωση του γραφικού κώδικα της εφαρμογής καθώς δίνει την δυνατότητα επιθεώρησης των τιμών επιστροφής κάθε υπο-module της εφαρμογής που δημιουργείται.

Για να δημιουργηθεί σωστά μια εφαρμογή πρέπει να περιέχει modules εισόδου και εξόδου έτσι ώστε μέσω των modules εισόδου να δημιουργηθεί στην συνέχεια μια ενιαία html φόρμα εισαγωγής των απαραίτητων παραμέτρων που απαιτεί η εκτέλεση της εφαρμογής και μέσω των Modules εξόδου να ορισθεί η μορφή του τερματικού της εξόδου της.

Πέρα του editor η διεπαφή του webhookit περιέχει μια συλλογή εργαλείων η οποία αποτελείται από την διαχείριση των υφιστάμενων πακέτων που χρησιμοποιεί η πλατφόρμα και εγκατάσταση καινούριων αν διατίθενται, μια λίστα με τις τελευταίες εκτελέσεις των εφαρμογών που περιλαμβάνει στοιχεία της εκτέλεσης τους και μια λίστα των εφαρμογών που δίνει δυνατότητες επεξεργασίας των template εξόδου που χρησιμοποιούν, χρονοπρογραμματισμού της εκτέλεσης της κάθε εφαρμογής καθώς επίσης και την παροχή των απαραίτητων στοιχείων δηλαδή σε ποια διεύθυνση URL πρέπει να καλεστούν για να εκτελεσθούν.

WebHookIt

DASHBOARDWIRINGSLISTERS TOOLSEDITORSIGN OUT

Welcome, 632718

Packages

Installed packages

Name	Version	Author	Description
undefined	undefined	Isaac Z. Schlueter	A package manager for node
mongodb	undefined	Vladimir Dronnikov	mongodb session store for connect
expat	undefined	Peter Galbraith	Simple XML to JavaScript object converter that uses Expat, a fast XML parser.
comment	undefined	Eric Abouaf	Comment module for WebHookIt
ejs	undefined	Eric Abouaf	EJS templating module for WebHookIt
http	undefined	Eric Abouaf	HTTP module for WebHookIt
jsonpath	undefined	Eric Abouaf	JSONPath module for WebHookIt
objectbuilder	undefined	Eric Abouaf	ObjectBuilder module for WebHookIt
xml2js	undefined	Eric Abouaf	xml2js-expat module for WebHookIt
jsonparse	undefined	Eric Abouaf	JSON.parse module for WebHookIt

Available packages

Name	Version	Author	Description
No more package to install			

About packages

Packages are distributed through the [npm](http://npmjs.org/) package manager: <http://npmjs.org/>

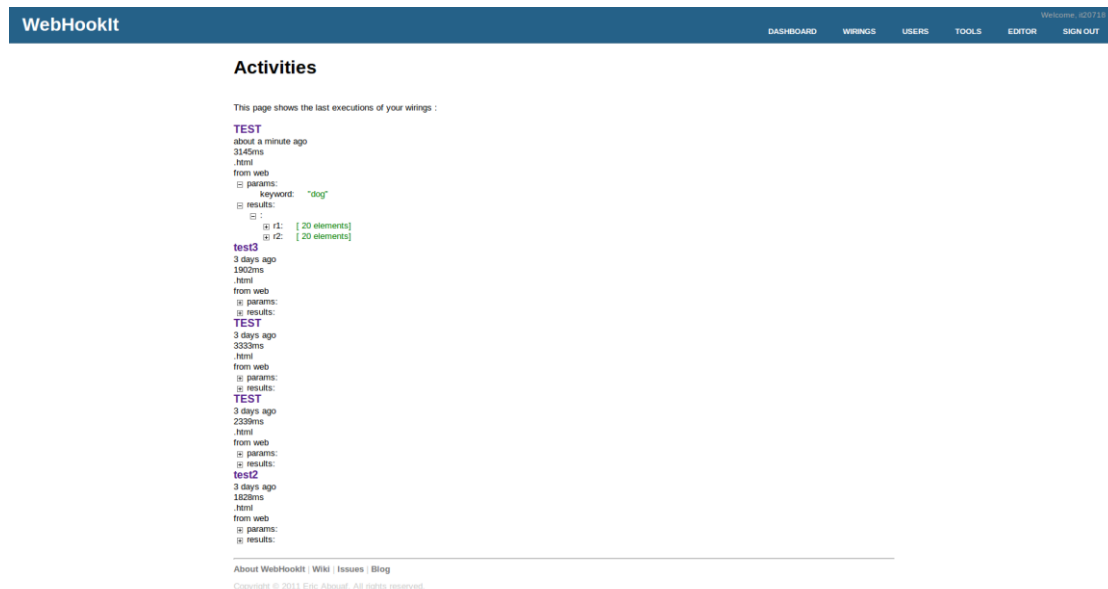
You can contribute packages by publishing a package to the [npm registry](#) called "webhookit-my-module".

More details on the [WebHookIt documentation](#).

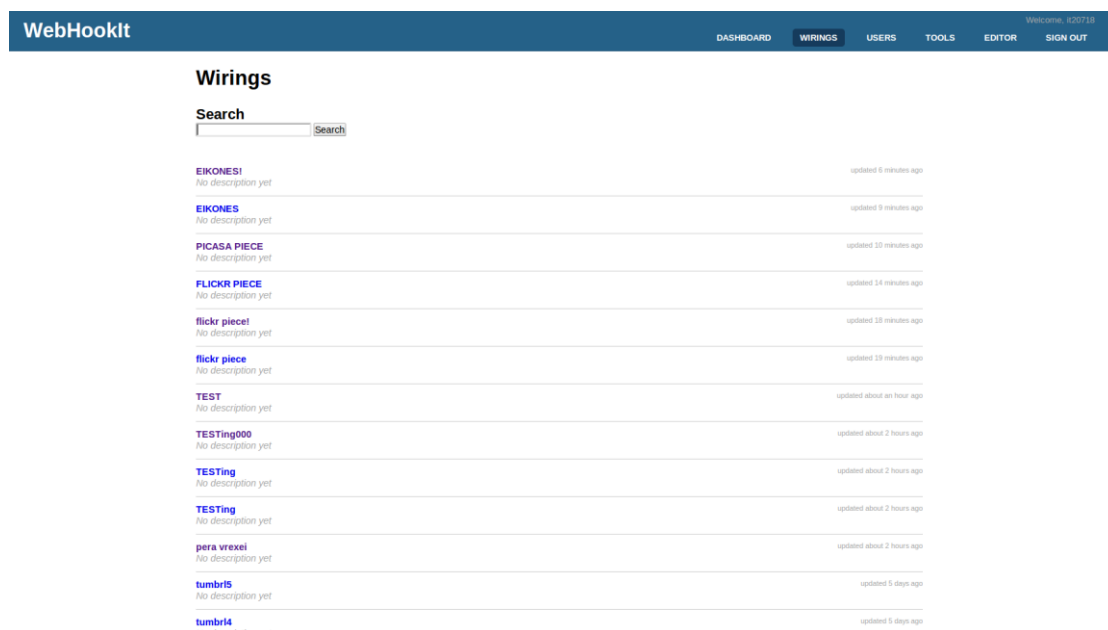
About WebHookIt | Wiki | Issues | Blog

Copyright © 2011 Eric Abouaf. All rights reserved.

Εικόνα 3-8: Διαχείριση πακέτων της εφαρμογής



Εικόνα 3-9: Εκτελέσεις δημιουργημένων εφαρμογών του WebHookIt



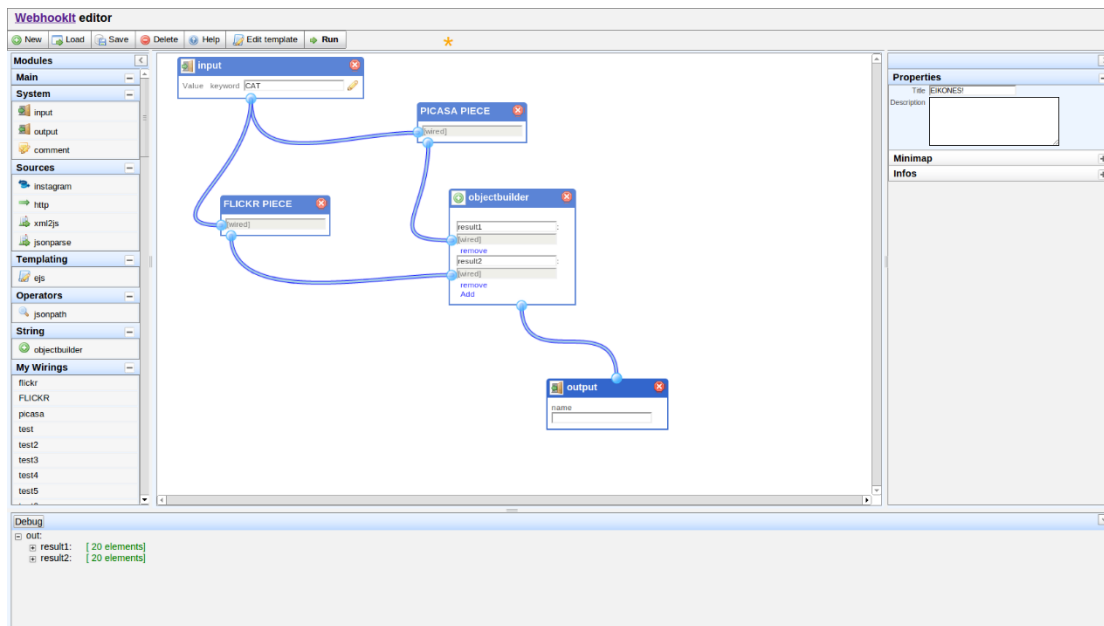
Εικόνα 3-10: Κατάλογος υπάρχοντων εφαρμογών για περαιτέρω επεξεργασία και εκτέλεσή τους

3.3.2 Περιγραφή υλοποίησης

Κατά την υλοποίηση της πειραματικής εφαρμογής με το WebHookIt το αρχικό πλάνο ήταν να γίνει και σε αυτή τη περίπτωση αναζήτηση βάση λέξης κλειδιού στα Flickr , Instagram και Picasa. Δυστυχώς όμως καθώς όπως προαναφέρθηκε η πλατφόρμα είναι ακόμα σε alpha έκδοση

υστερεί ως προς τις τελευταίες εκδόσεις των βιβλιοθηκών από τις οποίες αποτελείται με αποτέλεσμα με το API του Instagram, που εκ των τριών API (Flickr, Instagram, Picasa) είναι το νεότερο, να αποτυγχάνει η σύνδεση του με το WebHookIt καθώς η διαδικασία της πιστοποίησης της εφαρμογής που απαιτείται για να επιστραφούν τα αποτελέσματα που ζητούνται δεν είναι συμβατά με την έκδοση του node.js που χρησιμοποιεί η πλατφόρμα. Έτσι η όποια απόπειρα είτε τροποποίησης της διαδικασίας που χρησιμοποιήθηκε από τα άλλα δυο API οδηγούσε σε μη τερματιζόμενη σύνδεση με το API με αποτέλεσμα να αποτυγχάνει η εφαρμογή και η όποια απόπειρα δημιουργίας module κορμού της πλατφόρμας με χρήση javascript βιβλιοθηκών για την χρηστή του Instagram API αποτύγχανε λόγω μη συμβατών εκδόσεων του node.js στη κάθε βιβλιοθήκη.

Για το λόγο αυτό η υλοποίηση της εφαρμογής περιορίστηκε στην αναζήτηση εικόνων μόνο στα Flickr και Picasa. Η υλοποίηση λοιπόν της εφαρμογής ξεκινά με την εισαγωγή στον καμβά του «Input» module που χρησιμοποιείται ώστε να δημιουργηθεί μια φόρμα εισαγωγής για την λέξη-κλειδί που θα δίνει ο χρήστης και βάση της οποίας θα γίνεται η αναζήτηση. Στη συνέχεια η λέξη-κλειδί μέσω του module παρέχεται σε άλλα δυο modules τα «FLICKR PIECE» module και «PICASA PIECE» module που αποτελούν υπο-εφαρμογες υλοποιημένες μέσω συνδυασμού των προϋπαρχόντων module και που πραγματοποιούν την αναζήτηση στις δυο πηγές. Στη συνέχεια οι έξοδοι των δυο module συνδέονται με ένα «objectbuilder» module όπου οι έξοδοι των module συνενώνονται δημιουργώντας ένα αντικείμενο που περιέχει τα αποτελέσματα της αναζήτησης τα οποία δίνονται μετέπειτα σαν είσοδος στο «output» module για να ολοκληρωθεί η εφαρμογή και να εμφανιστούν τα αποτελέσματα. Η εκτέλεση της εφαρμογής γίνεται μέσω μιας οθόνης όπου ο χρήστης καλείται να εισαγάγει την λέξη κλειδί και αφού πατήσει το κουμπί εκτέλεσης του επιστρέφονται τα αποτελέσματα. Τα παραπάνω μαζί με την οθόνη των αποτελεσμάτων δίνονται στις παρακάτω εικόνες.



Εικόνα 3-11: Η υλοποίηση της εφαρμογής στο WebHookIt

WebHookIt
DASHBOARD
WIRINGS
USERS
TOOLS
EDITOR
SIGN OUT
Welcome, k20718

EIKONES!

No description yet
updated 6 minutes ago
created 6 minutes ago

Cron
Cron Rule: undefined [Edit](#)

Source
Using 5 modules :
input
output
PICASA PIECE
FLICKR PIECE
objectbuilder
[Edit source](#)

Template
Content-Type: undefined
[Edit](#)

Public execution
Public URL: not defined

Run
Parameters
keyword: CAT

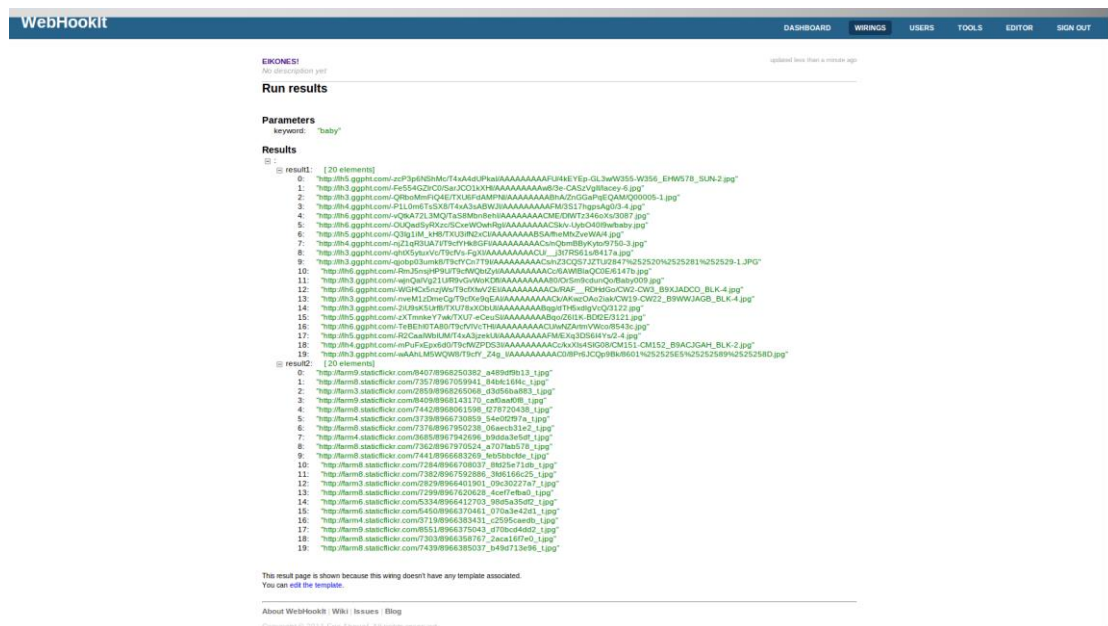
[Run](#)

[Run without template](#)
[Hide the url](#)
`/wirings/51b066bde5bf5ed90c000011/run?keyword=CAT`

About WebHookIt | Wiki | Issues | Blog
Copyright © 2011 Eric Abouaf. All rights reserved.

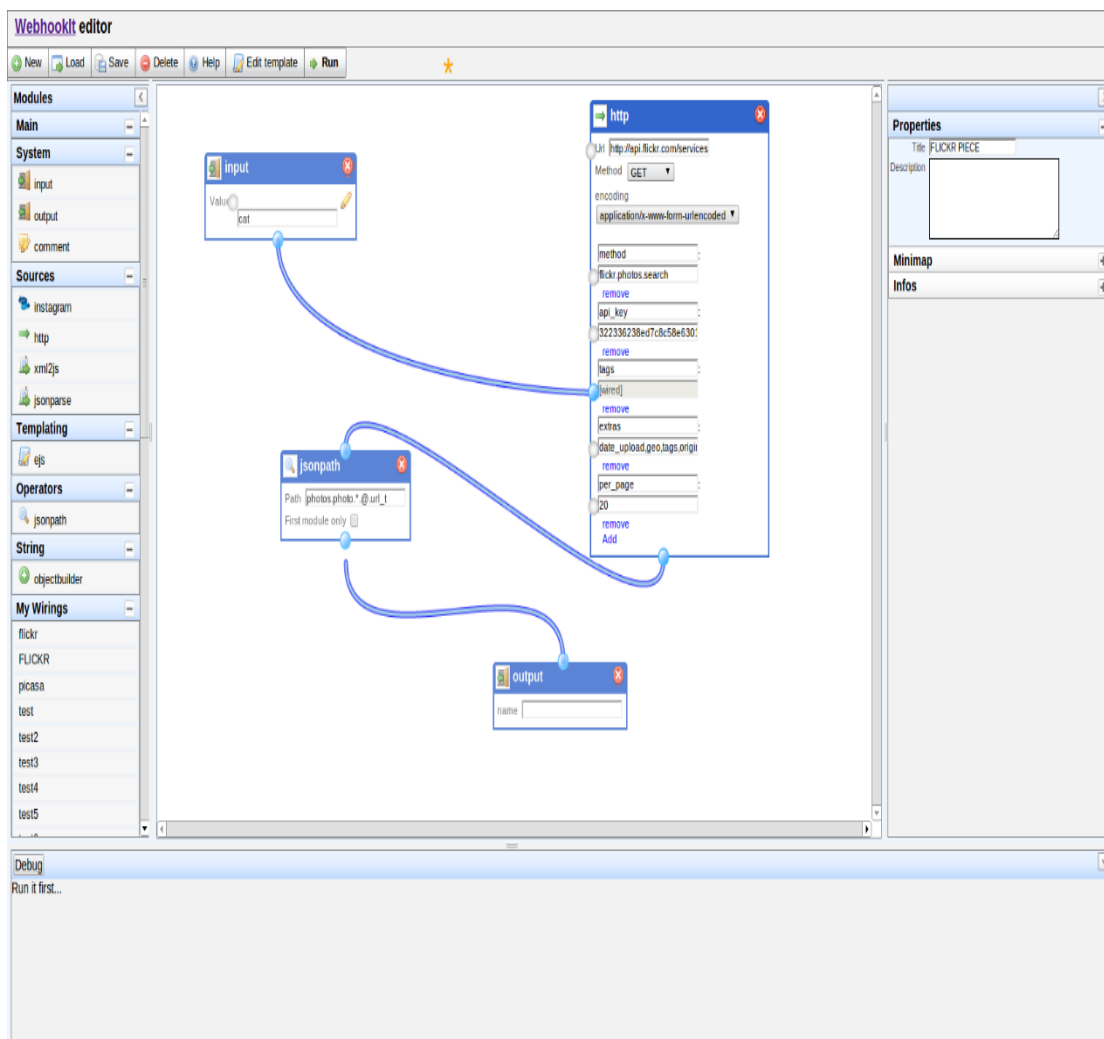
localhost:8124/wirings/51b066bde5bf5ed90c000011#

Εικόνα 3-12: Η σελίδα εκτέλεσης της εφαρμογής στο WebHookIt



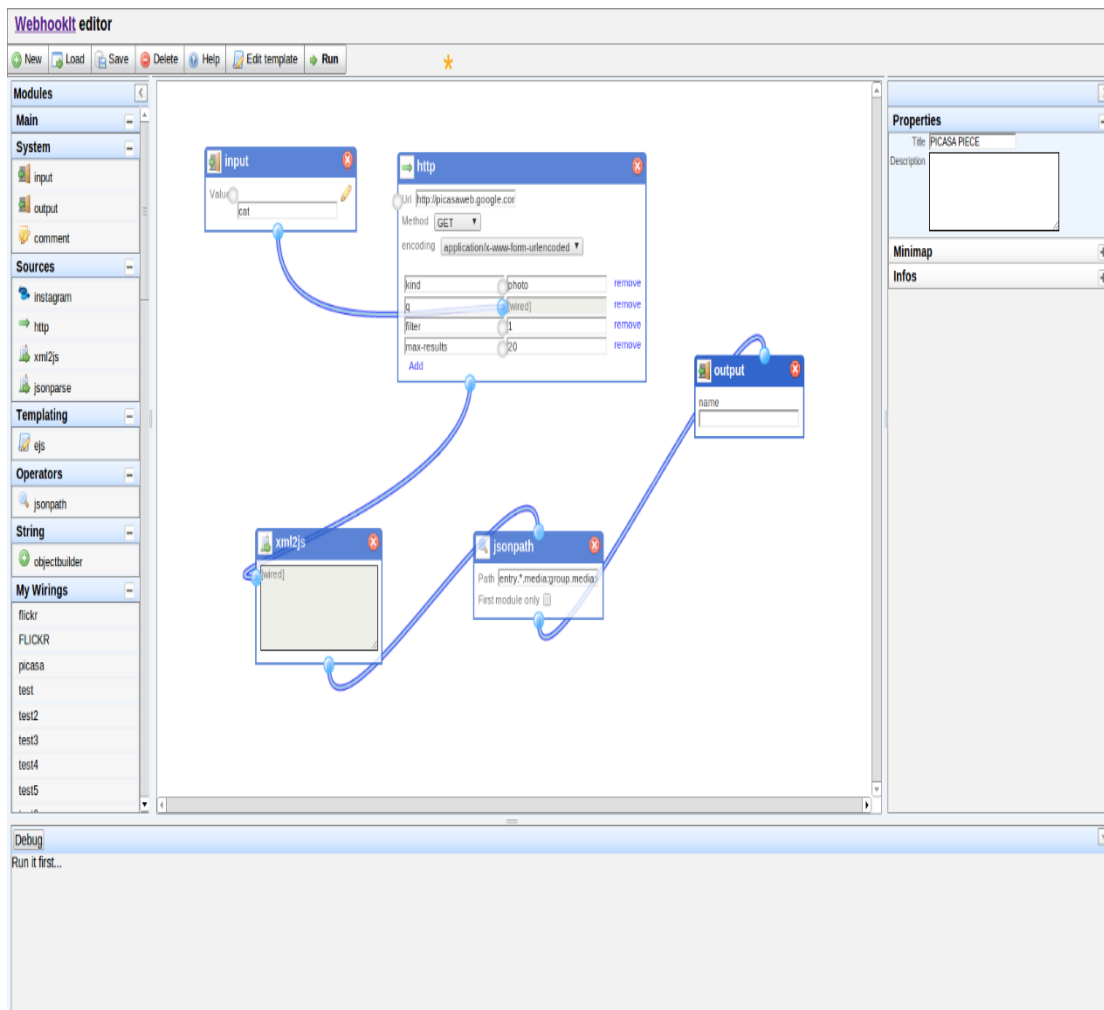
Εικόνα 3-13: Αποτελέσματα εκτέλεσης της εφαρμογής στο WebHookIt

Συνεχίζοντας την περιγραφή της υλοποίησης το «PICASA PIECE» module περιλαμβάνει ένα «input» module όπου γίνεται η εισαγωγή της λέξης κλειδιού στη συνέχεια το module αυτό συνδέεται με ένα «http» module μέσω του οποίου δημιουργείται η σύνδεση με το API του Flickr στέλνοντας ένα GET αίτημα στην υπηρεσία παρέχοντας τις απαραίτητες παραμέτρους και επιστρέφοντας τα αποτελέσματα σε JSON μορφή τα οποία στη συνέχεια τροφοδοτούνται σε ένα «jsonpath» module όπου φιλτράρονται επιστρέφοντας το URL κάθε εικόνας που τροφοδοτείται σαν είσοδος στο «output» module ολοκληρώνοντας την υπο-εφαρμογή. Η υλοποίηση της φαίνεται στο παρακάτω σχήμα.



Εικόνα 3-14: Υλοποίηση του picasa module-κομματιού της εφαρμογής στο WebHookIt

Τέλος η υλοποίηση του «FLICKR PIECE» module περιλαμβάνει ένα «input» module όπου γίνεται η εισαγωγή της λέξης-κλειδιού στη συνέχεια το module αυτό συνδέεται με ένα «http» module μέσω του οποίου δημιουργείται η σύνδεση με το API του Picasa στέλνοντας ένα GET αίτημα στην υπηρεσία παρέχοντας τις απαραίτητες παραμέτρους και επιστρέφοντας τα αποτελέσματα σε XML μορφή τα οποία στη συνέχεια τροφοδοτούνται σε ένα «xml2js» module μετατρέποντας τα από xml σε json. Έπειτα τροφοδοτούνται σε ένα «jsonpath» module όπου φιλτράρονται επιστρέφοντας το URL κάθε εικόνας που τροφοδοτείται σαν είσοδος στο «output» module ολοκληρώνοντας την υπο-εφαρμογή. Η υλοποίηση φαίνεται παρακάτω.



Εικόνα 3-15: Υλοποίηση του flickr module-κομματιού της εφαρμογής στο WebHookIt

4 ΣΥΓΚΡΙΤΙΚΗ ΑΞΙΟΛΟΓΗΣΗ

4.1 Σύγκριση Yahoo! Pipes & WebHookIt

4.1.1 Κριτήρια

Το πρώτο στάδιο για την σύγκριση των πλατφορμών που περιεγράφηκαν παραπάνω είναι ο προσδιορισμός των κριτηρίων αξιολόγησης. Η αξιολόγηση ξεκινά από το πρώτο βήμα, αυτό της εγκατάστασης της εκάστοτε πλατφόρμας, σε χρόνο και κυρίως σε δυσκολία. Πιο σημαντικά για επιχειρηματικούς σκοπούς είναι κάποια άλλα κριτήρια. Ένα τέτοιο είναι το ποιους τύπους εισόδου υποστηρίζει η εκάστοτε πλατφόρμα και ποιους τύπους εξόδου. Αυτό έχει ιδιαίτερη σημασία, διότι ένα Mashup είναι πολύ πιθανό να χρειάζεται να συνεργαστεί με άλλα προγράμματα της εταιρείας, όπως είναι βάσεις δεδομένων και ERPs (Enterprise Resource Planning). Ίσως το πιο ουσιαστικό κομμάτι της αξιολόγησης μπορεί να θεωρηθεί ο αριθμός των λειτουργιών που υποστηρίζει το εκάστοτε εργαλείο, όπως είναι το πλήθος των έτοιμων widgets που είναι διαθέσιμα. Όμως για να μπορεί να γίνει σωστή και αποτελεσματική διαχείριση των παρεχόμενων λειτουργιών, θα πρέπει να είναι όσο το δυνατόν πιο εύκολη η χρήση τους, να υπάρχει αποτελεσματικός μηχανισμός διαχείρισης λαθών (debugging) καθώς και να παρέχεται ένα ολοκληρωμένο εγχειρίδιο λειτουργίας της πλατφόρμας με πλήθος διαθέσιμων έτοιμων εφαρμογών που έχουν αναπτυχθεί είτε από στελέχη της εταιρείας ανάπτυξης της κάθε πλατφόρμας, ή από άλλους χρήστες, δηλαδή σωστή υποστήριξη. Σημειώνεται ότι για την αποτίμηση του κριτηρίου των υποστηριζόμενων λειτουργιών και του κριτηρίου των τύπων εξόδων, εξετάστηκαν κάποιοι επιμέρους τομείς. Συγκεκριμένα, για το πρώτο κριτήριο εξετάστηκε καταρχάς εάν οι πλατφόρμες μπορούν να επεξεργαστούν RSS ή/και Atom feeds, τα οποία όπως έχει επισημανθεί σε προηγούμενες ενότητες, είναι πλέον πολύ διαδεδομένα. Ένας δεύτερος τομέας που αξιολογήθηκε είναι αν οι πλατφόρμες μπορούν να υποστηρίξουν Web Services. Κάτι τέτοιο θα πρόσφερε πλήθος δυνατοτήτων στους προγραμματιστές οι οποίοι είτε έχοντας αναπτύξει οι ίδιοι τα Web Services, ή προμηθευτεί από άλλες πηγές, μπορούν να τα ενσωματώσουν στα αναπτυσσόμενα Mashups. Τέλος, στο θέμα των λειτουργιών, βαθμολογήθηκε θετικά η δυνατότητα χρησιμοποίησης γλωσσών προγραμματισμού, όπως είναι η JavaScript, έτσι ώστε ο έμπειρος προγραμματιστής να μπορεί να εξατομικεύσει το Mashup του και να προσθέσει λειτουργίες μέσω συναρτήσεων που πολλές φορές με γραφικό τρόπο δεν μπορούν να υλοποιηθούν. Από την άλλη πλευρά, για το κριτήριο της υποστήριξης των τύπων εξόδου, εξετάστηκαν δύο τομείς. Ο πρώτος είναι ο αριθμός, αλλά και η χρησιμότητα των μορφών αποθήκευσης που προσφέρει το κάθε εργαλείο και δεύτερον, εάν παρέχεται η δυνατότητα στο χρήστη να αποθηκεύσει το Mashup του σε δικό του δικτυακό τόπο.

4.1.2 Αξιολόγηση

Ξεκινώντας από το πρώτο κριτήριο αξιολόγησης, αυτό της εγκατάστασης, το Yahoo! Pipes, υπερέχει, εφόσον λειτουργεί δικτυακώς και δεν απαιτεί καμία εγκατάσταση εν σύγκριση με το WebHookIt που απαιτεί μια αρκετά πολύπλοκη διαδικασία εγκατάστασης που συμπεριλαμβάνει την εγκατάσταση πέραν της ίδιας της πλατφόρμας και την εγκατάσταση των Node.js, MongoDB και npm.

Επόμενο κριτήριο αξιολόγησης είναι αυτό της ευκολίας χρήσης. Το Pipes και το WebHookIt έχουν παρεμφερές περιβάλλον χρήσης και οι διαφορές των δύο πλατφορμών πρέπει να αναζητηθούν σε άλλα κριτήρια, εφόσον σε αυτό και οι δύο επιτρέπουν τη γραφική ένωση διαφόρων widgets για την παραγωγή νέων Mashups με το WebHookIt όμως να παρέχει λιγότερες έτοιμες δυνατότητες ένωσης των widgets, γεγονός που καθιστά ελαφρώς δυσκολότερη την πραγματοποίηση των επιθυμητών αποτελεσμάτων.

Επόμενο θέμα που εξετάζεται και σχετίζεται με τη λειτουργία της πλατφόρμας είναι το θέμα της αποσφαλμάτωσης, δηλαδή του μηχανισμού διαχείρισης λαθών και της άμεσης ενημέρωσης για την κάθε ενέργεια που πράττει ο χρήστης. Για το σκοπό της αξιολόγησης ελέγχθηκε η κάθε πλατφόρμα για την ύπαρξη και στην συνέχεια για τον τρόπο λειτουργίας του debugger. Τόσο το Yahoo! Pipes όσο και το WebHookIt επιτρέπουν να βλέπει ο χρήστης το αποτέλεσμα του κάθε module όταν το επιλέγει, μέσω ενός Debugger που είναι στο κάτω μέρος με τον debugger του Yahoo Pipes να είναι ελαφρώς πιο κατατοπιστικός σε περιπτώσεις σφαλμάτων σε σχέση με το WebHookIt καθώς τα τυχόν σφάλματα και οι τρόποι επίλυσης τους στην περίπτωση του δεύτερου φαίνονται στον terminal που τρέχει η εφαρμογή μας και όχι στον browser όπου την δημιουργούμε.

Κατά την ενασχόληση με την κάθε πλατφόρμα έγινε και αναζήτηση παραδειγμάτων και εγχειριδίων χρήσης. Η ύπαρξή τους θεωρείται πολύ σημαντική, τόσο για να μπορέσει ένας νέος χρήστης να λειτουργήσει το εκάστοτε εργαλείο, όσο και για να μπορέσει να πάρει ιδέες για να αναπτύξει πιο εξελιγμένες εφαρμογές. Η πρώτη πλατφόρμα, το Pipes, προσφέρει εκτός από ένα πολύ κατατοπιστικό video παρουσίασης της λειτουργίας της πλατφόρμας, online εγχειρίδια χρήσης του Editor, των Modules, forum, καθώς και πλήθος tutorials και ένα κατατοπιστικό troubleshooting guide. Επιπρόσθετα τα Pipes των χρηστών είναι διαθέσιμα δωρεάν, μαζί με τη δυνατότητα τροποποίησής τους. Το WebHookIt από την άλλη σαν μια πλατφόρμα που βρίσκεται ακόμα σε alpha στάδιο υστερεί κατά πολύ ως προς την υποστήριξη της καθώς το εγχειρίδιο χρήσης της είναι αρκετά φτωχό και υπάρχει μόνο ένα βίντεο παρουσίασης του και τα forums και η γενική του υποστήριξη είναι σχεδόν μηδαμινή.

Για το τέλος αφέθηκαν τα τρία ίσως σημαντικότερα κριτήρια, οι εισόδοι, οι εξόδοι και οι λειτουργίες που υποστηρίζει η κάθε πλατφόρμα. Τα συγκεκριμένα περιεγράφηκαν αναλυτικά στο προηγούμενο κεφάλαιο που έγινε η περιγραφή των πλατφορμών. Εδώ απλά εξάγονται κάποια συμπεράσματα. Καταρχάς, το Pipes περιέχει όπως φαίνεται πολλά modules, τόσο εισόδου, όσο κυρίως λειτουργιών που επιτρέπουν τον συνδυασμό αυτών ενώ το WebHookIt περιέχει σαφώς λιγότερα με αποτέλεσμα μια πιο σύνθετη λειτουργία να απαιτεί από τον χρήστη γνώσεις javascript ώστε να κατασκευάσει το απαραίτητο module. Το Yahoo Pipes! επίσης παρέχει πολύ καλή υποστήριξη και στους τύπους εξόδου, περιλαμβάνοντας μεταξύ άλλων blogs, RSS και JavaScript, ενώ επιτρέπει την ελεύθερη δημοσίευση στον επίσημο δικτυακό τόπο υπερτερώντας του WebHookIt που έχει πιο περιορισμένους τύπους εξόδου και ακόμα δεν υπάρχει διαδικτυακός τρόπος που να επιτρέπει τον διαμοιρασμό των παραγόμενων εφαρμογών όπως το Yahoo! Pipes.

Όπως αναφέρθηκε και στην ενότητα της περιγραφής των κριτηρίων, για την αξιολόγηση των κριτηρίων του αριθμού των λειτουργιών και των μορφών αποθήκευσης, σημαντικό ρόλο διαδραμάτισαν και επιμέρους τομείς, οι οποίοι ελήφθησαν υπόψη και συγκεντρώθηκαν στον Πίνακα 1 που ακολουθεί.

Πλατφόρμα	RSS / Atom	Web Services	Γλώσσες	Μορφές Αποθήκευσης	Αποθήκευση σε προσωπικό Site
Yahoo! Pipes	NAI	NAI		TypePad, Blogger, Word Press, iGoogle, RSS, Netvibes, JSON, PHP, KLM	NAI
WebhookIt	OXI	NAI	JavaScript	XML,JSON	NAI

Πίνακας 4-1: Υποστηριζόμενες τεχνολογίες των Yahoo! Pipes και WebHookIt

Πέραν της σύγκρισης των δυο πλατφορμών μέσω των προαναφερόμενων κριτηρίων υπάρχουν μερικές ακόμα διαφορές μεταξύ τους. Με το WebHookIt δεν υπάρχουν περιορισμοί λειτουργίας καθώς ο χρήστης -προγραμματιστής έχει την δυνατότητα να συμπεριλάβει δικά του module λειτουργιών τα οποία μάλιστα μπορούν να δημιουργηθούν σε ευρεία γκάμα γλωσσών (ruby, python κτλ.) ενώ στην περίπτωση του Yahoo! Pipes τα module είναι αυστηρά καθορισμένα και η εισαγωγή “ξένων” web services μέσω κώδικα είναι ιδιαίτερα πολύπλοκη. Επίσης περιέχει ενσωματωμένο σύστημα χρονοπρογραμματισμού των παραγόμενων εφαρμογών δίνοντας την δυνατότητα εκτέλεσης των Pipes on demand δυνατότητα που το Yahoo! Pipes δεν παρέχει καθόλου. Επιπρόσθετα το WebHookIt είναι άκρως προσαρμόσιμο και ενσωματώνεται με ευκολία σε προσωπικά συστήματα ενώ το Yahoo Pipes περιορίζεται κατά πολύ ως προς αυτό μην αφήνοντας πολλά περιθώρια ελέγχου του. Τέλος το Yahoo Pipes είναι hosted service γεγονός που το κάνει πιο προσβάσιμο σε σχέση με WebHookIt καθώς επίσης παρέχει διαμοιρασμό Pipes μεταξύ των χρηστών του, παροχή που ακόμα το WebHookIt δεν υποστηρίζει. Σε μια προσπάθεια συνοψισμού των διαφορών προκύπτει το συμπέρασμα πως το Yahoo Pipes αποτελεί σαφώς πιο ενδεδειγμένη επιλογή για μη γνώστες ή αρχάριους προγραμματιστές καθώς παρέχει αρκετό υλικό εκμάθησης της πλατφόρμας και ήδη πραγματοποιημένων λειτουργιών από άλλους χρήστες. Επίσης παρέχει αυτοματοποιημένες διαδικασίες που απλοποιούν κάποιες διαδικασίες ενώ το WebHookIt αποτελεί καλύτερη επιλογή για πιο έμπειρους χρήστες που γνωρίζουν καλά προγραμματισμό και θέλουν να έχουν βαθύτερο έλεγχο σε κάθε κομμάτι του όλου συστήματος.

Στον παρακάτω πίνακα γίνεται μια γραφική απεικόνιση της σύγκρισης των δυο πλατφορμών.

Πλατφόρμα	Yahoo! Pipes	WebHookIt
Ευκολία εγκατάστασης	Δεν εγκαθίσταται καθώς διατίθεται Online (+)	Πολύπλοκη εγκατάσταση (-)
Ευκολία χρήσης	Visual editor πολλών λειτουργιών (+)	Visual editor περιορισμένων λειτουργιών (-)
Αποσφαλμάτωση	Πλήρως κατατοπιστικός debugger (+)	Λιγότερο κατατοπιστικός debugger (-)
Υποστήριξη	Πολλά διαφορετικά μέσα υποστήριξης (+)	Φτώχη υποστήριξη (-)
Είσοδοι	Feeds, CSV, limited generic XML (+)	XML,JSON (-)
Έξοδοι	TypePad, Blogger, Word Press, iGoogle, RSS, Netvibes, JSON, PHP, KLM (+)	XML,JSON (-)
Λειτουργίες	(+)	(-)
Διαμοιρασμός παραγόμενων εφαρμογών	Παρέχεται κοινοτητα διαμοιρασμου εφαρμογων (+)	Δεν παρέχεται κοινοτητα διαμοιρασμου εφαρμογων (-)
Δημιουργία πρόσθετων λειτουργιών	Περιορισμενη δυνατοτητα (-)	Πολλες δυνατοτητες (+)
Ενσωμάτωση σε προσωπικά συστήματα	Φτωχων δυνατοτητων ενσωματωση (-)	Πλήρης ελευθερία ενσωματωσης (+)

Πίνακας 4-2 Σύγκριση Yahoo! Pipes και WebHookIt

4.2 Σύγκριση υλοποίησης mashup με τη βοήθεια κάποιου εργαλείου ανάπτυξης mashup σε σχέση με την ανάπτυξη «με το χέρι»

Η δημιουργία μιας συνεκτικής εφαρμογής ενοποίησης δεδομένων και εφαρμογών χωρίς κάποια πλατφόρμα δημιουργίας mashups απαιτεί γνώσεις προγραμματισμού και βαθιά γνώση σχετικά με τα συστήματα και την σημασιολογία των πηγών δεδομένων ή τα πρωτόκολλα για την ανταλλαγή μηνυμάτων. Αν και η έλευση των νέων τεχνολογιών όπως η AJAX ή οι RESTful υπηρεσίες και τα micro formats όπως RSS ή Atom, έκανε την ανάπτυξη των mashups αρκετά πιο απλή με την ενσωμάτωση περιεχομένου ,εφαρμογών, ή user interfaces σε μεγάλο βαθμό να επικουρείται από «έξυπνα» components. Παρ 'όλα αυτά, η ανάπτυξη mashup προγραμματίζοντας το από την αρχή χωρίς την βοήθεια πλατφορμών εξακολουθεί να είναι προνόμιο των ειδικευμένων προγραμματιστών και να απαιτεί χρόνο.

Εξετάζοντας ένα χαρακτηριστικό παράδειγμα mashup όπως το HousingMaps, ένα mashup που συνδυάζει το Google Maps με ένα site αγγελιών, το Craigslist για την γεωγραφική απεικόνιση των αγγελιών πώλησης κατοικιών, η υλοποίηση του με το χέρι θα συμπεριλάμβανε τα εξής βήματα. Πρώτον, την εξοικείωση με τα δύο API που αποτελούν τις πηγές περιεχομένου, δηλαδή, το Craigslist και το Google Maps, καθώς και τον προσδιορισμό του τρόπου επαναχρησιμοποίησης ή εξαγωγής δεδομένων από τις δύο αυτές θέσεις. Ενώ για το Google Maps υπάρχει ένα κοινό JavaScript API, που μπορεί να αξιοποιηθεί, το Craigslist παρέχει τους καταλόγους του μέσω RSS με αποτέλεσμα προκειμένου να εξαχθούν τα στοιχεία ιδιοκτησίας και διευθύνσεων, τα RSS feeds από το Craigslist πρέπει να αναλυθούν και να ερμηνευθούν. Εν συνεχεία για την διαμόρφωση των clickable δεικτών που θα εμφανίζουν πληροφορίες ιδιοκτησίας σε αναδυόμενα παράθυρα πάνω σε κάθε κλικ δείκτη, απαιτείται αλληλεπίδραση με το Google Maps API, το οποίο παρέχεται σε JavaScript. Για την αυτόματη ενεργοποίηση του αναδυόμενου παραθύρου με τις πληροφορίες ιδιοκτησίας από την επιλογή ενός ακινήτου από τη λίστα, μια ειδική λειτουργία-συνάρτηση JavaScript πρέπει υλοποιηθεί, η οποία με βάση την επιλογή του ακινήτου από την παρεχόμενη λίστα του Craigslist και καλεί την Google Maps API να επιλέξει τον αντίστοιχο δείκτη στον χάρτη εμφανίζοντας το αντίστοιχο παράθυρο. Συνεχίζοντας αν και το Google Maps διαθέτει δικό του UI, επιτρέποντας στον χρήστη να επιλέξει ιδιότητες που έχουν εξαχθεί από το Craigslist απαιτεί από τον προγραμματιστή το γέμισμα και την κατάλληλη διαμόρφωση ενός κατάλληλου πίνακα. Τέλος, τα δύο συστατικά στοιχεία πρέπει να καθορισθούν σωστά για να σχηματίσουν το UI της σύνθετης εφαρμογής.

Σε αντίθεση με την παραπάνω διαδικασία η υλοποίηση του mashup που περιεγράφηκε μέσω μιας πλατφόρμας δημιουργίας mashups όπως το Yahoo! Pipes θα περιλάμβανε την επεξεργασία του Craigslist feed για τον εντοπισμό πληροφοριών θέσης (geo codes) με χρήση του locator extractor (συστατικό στοιχείο λειτουργίας του Yahoo! Pipes). Έπειτα οι προσδιοριζόμενες πληροφορίες τοποθεσίας θα προστίθονταν στο feed του Craigslist υπό την μορφή ενός συνδέσμου που θα επέτρεπε την εμφάνιση της διεύθυνση του ακινήτου στο χάρτη απλά με το πέρασμα του geo code στο GoogleMaps. Τέλος στην περίπτωση αυτής της υλοποίησης η διεπιφάνεια χρήστη (UI) δημιουργείται αυτόματα απαλλάσσοντας τον προγραμματιστή από την σχεδίαση και την μορφοποίηση του.

Μέσω γενίκευσης της σύγκρισης του παραπάνω παραδείγματος ,και από την μελέτη διαφόρων συχνά χρησιμοποιούμενων mashups διαφαίνεται ότι δημιουργώντας mashups «με το χέρι» σε σχέση με την δημιουργία τους μέσω ειδικών πλατφορμών προκύπτουν τα εξής συμπεράσματα-διαφορές. Η χειροκίνητη δημιουργία ενός mashup αποκλείει από την διαδικασία του μη έμπειρους προγραμματιστές καθώς απαιτεί αρκετά προχωρημένες γνώσεις προγραμματισμού αποτελώντας μια πολύπλοκη και χρονοβόρα διαδικασία . Επίσης δημιουργείται συχνά πρόβλημα με την διαχείριση των δεδομένων καθώς ένα mashup συχνά απαιτεί την επαλήθευση των δεδομένων του ή την μετατροπή τους σε μια ενιαία μορφή ,διαδικασίες που απλοποιεί ιδιαίτερα η χρήση ενός mashup tool καθώς τις παρέχει σχεδόν αυτοματοποιημένες. Επιπρόσθετα ο τρόπος αυτός κάνει ιδιαίτερα δύσκολη την μετέπειτα διαφοροποίηση της διαδικασίας καθώς δημιουργεί στατικές ροές μη επιτρέποντας την δυναμική διαχείριση τους. Τέλος ένα μεγάλο πλεονέκτημα των πλατφορμών δημιουργίας mashups είναι η δυνατότητα σύνδεσης mashups μεταξύ τους δυνατότητα που στην χειροκίνητη δημιουργία προϋποθέτει την δημιουργία επιπρόσθετου συνδετικού κώδικα. Από τα παραπάνω συνάγεται λοιπόν το συμπέρασμα ότι η χρήση mashup tools αποτελεί μεγάλη διευκόλυνση για τον εκάστοτε δημιουργό mashup καθώς απλοποιεί την διαδικασία και μειώνει κατά πολύ το χρόνο δημιουργίας της εφαρμογής.

5 ΣΥΜΠΕΡΑΣΜΑΤΑ & ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ

5.1 ΣΥΜΠΕΡΑΣΜΑΤΑ

Είναι αδιαμφισβήτητο γεγονός ότι οι σύγχρονες εφαρμογές κοινωνικής δικτύωσης έχουν κατακλύσει το διαδίκτυο και αποτελούν αναπόσπαστο κομμάτι της καθημερινότητας, σε ένα περιβάλλον που χαρακτηρίζεται από συνεχή μεταβολή και αυξημένες απαιτήσεις επικοινωνίας. Η διάσπαρτη διάδοση των δεδομένων μέσα από τους αναρίθμητους ιστότοπους που επισκεπτόμαστε καθημερινά, στο χαώδες οικοσύστημα του διαδικτύου, έχει ως συνέπεια να καθίσταται πολλές φορές αναποτελεσματική η επικοινωνία, η ενημέρωση μας και η εύρεση πραγματικά χρήσιμων πληροφοριών. Κρίνεται σκόπιμο λοιπόν, να καταφύγουμε σε νέες λύσεις οι οποίες ολοκληρώνουν τις υπάρχουσες εφαρμογές κοινωνικής δικτύωσης σε μία πιο σύνθετη εφαρμογή και αποτελεσματική. Μία τέτοια λύση καλείται να δώσει το Web 2.0 και πιο συγκεκριμένα τα Mashups τα οποία αποτελούν μια πολύ αποδοτική λύση φιλτραρίσματος και επεξεργασίας των πληροφοριών αυτών.

Στα πλαίσια αυτής της εργασίας μελετήθηκε επισταμένα το θεωρητικό υπόβαθρο της τεχνολογίας του Web 2.0 και όσων την αποτελούν δίνοντας μεγαλύτερη έμφαση σε ένα βασικό της κομμάτι τα Mashups. Πιο συγκεκριμένα εφόσον ορίστηκε η έννοια τους, μελετήθηκε η αρχιτεκτονική τους και έγινε μελέτη κάποιων εκ των πιο γνωστών πλατφορμών ανάπτυξης mashups. Στη συνέχεια υπήρξε μια πιο «χειροπιαστή» εξέταση των Mashups δημιουργώντας μια εφαρμογή άντλησης περιεχομένου και πιο συγκεκριμένα αναζήτησης εικόνων από γνωστούς ιστότοπους διαμοιρασμού εικόνων (Flickr ,Instagram και Picasa) βάση μιας λέξης-κλειδιού σε δυο διαφορετικές πλατφόρμες ανάπτυξης mashups, το Yahoo! Pipes και το WebHookIt. Η σύγκριση των δυο πλατφορμών αυτών κατέδειξε το Yahoo! Pipes καταλληλότερο για άτομα με περιορισμένες γνώσεις προγραμματισμού αλλά με πιο περιοριστικές δυνατότητες από άποψη συνδυασμού δυνατοτήτων και ελέγχου του συστήματος σε σχέση με το WebHookIt με την προϋπόθεση ολοκλήρωσης της ανάπτυξης του μιας και ακόμα αποτελεί μια πειραματική πλατφόρμα που βρίσκεται ακόμα σε alpha στάδιο.

Μέσα από την εργασία αυτή, διαφαίνεται ότι είναι πλέον αναγκαίο οι πλατφόρμες που αναπτύσσονται σήμερα να αντλούν περιεχόμενο από άλλες εφαρμογές, προκειμένου να ανταπεξέρχονται στις ανάγκες των χρηστών και να υπάρχει βέλτιστη χρήση και εκκαθάριση της πληθώρας των πληροφοριών που καλείται να αντιμετωπίσει στις ημέρες μας ο χρήστης του διαδικτύου καθώς με αυτόν τον τρόπο και εκείνος επωφελείται της πληθώρας των πολλών και διαφορετικών υπηρεσιών που προκύπτουν διευκολύνοντας τον ποικιλοτρόπως και οι προγραμματιστές δημιουργούν όλο και ευκολότερα νέες υπηρεσίες καθώς η εξέλιξη και ο

συνδυασμός ήδη δημιουργημένων εφαρμογών είναι σαφώς ευκολότερο έργο τόσο σε κόπο όσο και σε χρονική διάρκεια.

5.2 ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ

Όπως προαναφέρθηκε πλέον στο διαδίκτυο υπάρχει πληθώρα πληροφοριών που ο χρήστης αδυνατεί να τις διαχειριστεί από μόνος του με το web 2.0 και πιο συγκεκριμένα τα mashups να αποτελούν μια πολύ καλή λύση διαχείρισης τους. Προς το παρόν όμως παρότι ναι μεν δίνουν μεγάλες δυνατότητες συνδυασμού υπηρεσιών και διαχείρισης μεγάλου πλήθους πληροφοριών υστερούν αρκετά ως προς την δυνατότητα φορητοτητας της υπηρεσίας τους καθώς ακόμα δεν δίνεται η δυνατότητα δημιουργίας mashups μέσω smartphones. Με την εξέλιξη της τεχνολογίας και την είσοδο των smartphone συσκευών και γενικά της ευρυζωνικότητας οι χρήστες του διαδικτύου έχουν πρόσβαση σε αυτό και με άλλα μέσα πέρα του ηλεκτρονικού υπολογιστή καθ' όλη την διάρκεια της ημέρας με τις ιστοσελίδες κοινωνικής δικτύωσης να αποτελούν μεγάλο κομμάτι της καθημερινότητας των χρηστών , λόγω των παραπάνω λοιπών μια μελλοντική επέκταση των mashups θα ήταν η εξέλιξη των πλατφορμών δημιουργίας τους ώστε να δίνεται η δυνατότητα στον χρήστη δημιουργίας mashups εφαρμογών από το κινητό του τηλέφωνο. Αυτό βέβαια προϋποθέτει και την περαιτέρω εξέλιξη του τρόπου δημιουργίας των mashups απλοποιώντας και αυτοματοποιώντας ακόμα περισσότερο την σύνδεση των διαφόρων υπηρεσιών και των δεδομένων από τα οποία αποτελούνται.

6 Βιβλιογραφία

- [1] O'reilly, T. (2007). What is Web 2.0: Design patterns and business models for the next generation of software. Communications & strategies, (1), 17. Retrieved from http://papers.ssrn.com/sol3/papers.cfm?abstract_id=1008839

- [2] Encyclopedia - Britannica Online Encyclopedia. (n.d.). Retrieved July 8, 2013, from <http://www.britannica.com/>

- [3] The Wisdom of Crowds. (2013, June 6). In Wikipedia, the free encyclopedia. Re-trieved from https://en.wikipedia.org/w/index.php?title=The_Wisdom_of_Crowds&oldid=558593367

- [4] Ιστότοποι Google. (n.d.). Retrieved July 8, 2013, from <https://sites.google.com/?pli=1>

- [5] Mashup Dashboard - ProgrammableWeb. (n.d.). Retrieved July 8, 2013, from <http://www.programmableweb.com/mashups>

- [6] Brooks. (1987). No Silver Bullet Essence and Accidents of Software Engineering. Computer, 20(4), 10–19. doi:10.1109/MC.1987.1663532

- [7] Myers, B. A., Ko, A. J., & Burnett, M. M. (2006). Invited research overview: end-user programming. In CHI'06 extended abstracts on Human factors in computing systems (pp. 75–80). Retrieved from <http://dl.acm.org/citation.cfm?id=1125472>

- [8] Rode, J. (2005). Web application development by nonprogrammers: user-centered design of an end-user web development tool. Virginia Polytechnic Institute and State University. Retrieved from http://128.173.49.46/rid=1171391050093_719544497_53835/rode.pdf

- [9] Zang, N. (2008). Mashups for the web-active end user. The Pennsylvania State University. Retrieved from <https://etda.libraries.psu.edu/paper/9241/4574>

- [10] O'reilly, T. (2007). What is Web 2.0: Design patterns and business models for the next generation of software. *Communications & strategies*, (1), 17. Retrieved from http://papers.ssrn.com/sol3/papers.cfm?abstract_id=1008839

- [11] Liu, X., Hui, Y., Sun, W., & Liang, H. (2007). Towards service composition based on mashup. In *Services, 2007 IEEE Congress on* (pp. 332–339). Retrieved from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4278815

- [12] Floyd, I. R., Jones, M. C., Rathi, D., & Twidale, M. B. (2007). Web mash-ups and patchwork prototyping: User-driven technological innovation with web 2.0 and open source software. In *System Sciences, 2007. HICSS 2007. 40th Annual Hawaii International Conference on* (pp. 86–86). Retrieved from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4076545

- [13] Huynh, D. F., Karger, D. R., & Miller, R. C. (2007). Exhibit: lightweight structured data publishing. In *Proceedings of the 16th international conference on World Wide Web* (pp. 737–746). Retrieved from <http://dl.acm.org/citation.cfm?id=1242672>

- [14] SIMILE | Babel. (n.d.). Retrieved July 8, 2013, from <http://service.simile-widgets.org/babel/>

- [15] Apatar - Open Source Data Integration & ETL - Apatar - Open Source Data Integration & ETL. (n.d.). Retrieved July 8, 2013, from <http://www.apatar.com/>
- [16] Ennals, R. J., & Garofalakis, M. N. (2007). MashMaker: mashups for the masses. In Proceedings of the 2007 ACM SIGMOD international conference on Management of data (pp. 1116–1118). Retrieved from <http://dl.acm.org/citation.cfm?id=1247626>
- [17] JackBe Presto. (n.d.). Retrieved July 8, 2013, from <http://www.jackbe.com/products/presto>
- [18] Pipes: Rewire the web. (n.d.). Retrieved July 8, 2013, from <http://pipes.yahoo.com/pipes/>
- [19] Photo Site a Hit With Bloggers. (n.d.). WIRED. Retrieved July 8, 2013, from <http://www.wired.com/culture/lifestyle/news/2004/12/65958>
- [20] Flickr - Yahoo! (n.d.). Retrieved July 9, 2013, from <http://advertising.yahoo.com/article/flickr.html>
- [21] Flickr Boasts 6 Billion Photo Uploads. (n.d.). Retrieved July 8, 2013, from <http://news.softpedia.com/news/Flickr-Boasts-6-Billion-Photo-Uploads-215380.shtm>
- [22] Press Center • Instagram. (n.d.). Retrieved July 8, 2013, from <http://instagram.com/press/>
- [23] Facebook's Instagram says it has 90 million monthly active users | TechHive. (n.d.). Retrieved July 8, 2013, from <http://www.techhive.com/article/2025801/facebooks-instagram-says-it-has-90-million-monthly-active-users.html>

- [24] Picasa - Wikipedia, the free encyclopedia. (n.d.). Retrieved July 8, 2013, from <https://en.wikipedia.org/wiki/Picasa>