

Ανοικτό Πανεπιστήμιο Κύπρου

Σχολή Θετικών και Εφαρμοσμένων Επιστημών

Μεταπτυχιακή Διατριβή στα Πληροφοριακά και Επικοινωνιακά Συστήματα



Ανάπτυξη Έξυπνου Βοηθού για το Twitter

Σταύρος Ελευθεράκης

**Επιβλέπων Καθηγητής
Δρ. Λοΐζος Μιχαήλ**

Ιούνιος 2017

Ανοικτό Πανεπιστήμιο Κύπρου

Σχολή Θετικών και Εφαρμοσμένων Επιστημών

Ανάπτυξη Έξυπνου Βοηθού για το Twitter

Σταύρος Ελευθεράκης

Επιβλέπων Καθηγητής
Δρ. Λοΐζος Μιχαήλ

Η παρούσα μεταπτυχιακή διατριβή υποβλήθηκε
προς μερική εκπλήρωση των απαιτήσεων για απόκτηση

μεταπτυχιακού τίτλου σπουδών
στα Πληροφοριακά Συστήματα

από τη Σχολή Θετικών και Εφαρμοσμένων Επιστημών
του Ανοικτού Πανεπιστημίου Κύπρου

Ιούνιος 2017

Περίληψη

Στόχος της μεταπτυχιακής διατριβής είναι η μελέτη και η ανάπτυξη ενός συστήματος έξυπνου βοηθού για το μέσο κοινωνικής δικτύωσης Twitter. Ο έξυπνος βοηθός θα είναι σε θέση να εκτελεί αυτόματα κάποιες βασικές λειτουργίες εκ μέρους του χρήστη στο Twitter, λαμβάνοντας υπόψιν μια σειρά από επιλογές, ενδιαφέροντα και ανάγκες του χρήστη.

Η σχετική μελέτη που έγινε πάνω στους έξυπνους βοηθούς, τα συστήματα συστάσεων (Recommendation Systems) και την πλατφόρμα του Twitter μας έδωσε κάποια χρήσιμα στοιχεία για το σχεδιασμό ενός τέτοιου συστήματος. Συγκεκριμένα, κρίθηκε απαραίτητη η χρήση Βάσης Δεδομένων που περιέχει πληροφορίες για τον χρήστη και η σύνδεση με το Twitter API για να πραγματοποιεί ο έξυπνος βοηθός ενέργειες εκ μέρους του χρήστη. Επίσης ο χρήστης θα δίνει κάποιες οδηγίες με τη μορφή κανόνων στο σύστημα που περιγράφουν τις ανάγκες και τα ενδιαφέροντά του. Ο έξυπνος βοηθός με τη χρήση ενός εργαλείου εξαγωγής λογικών συμπερασμάτων (FLASH) προτείνει και εκτελεί κάποιες ενέργειες σχετικές με το Twitter.

Αναπτύχθηκε έτσι μια web εφαρμογή (Twitter Assistant) που αποτελεί τη διεπαφή του χρήστη με τα παραπάνω υποσυστήματα, με χρήση των γλωσσών και μεθόδων προγραμματισμού PHP, HTML, CSS, JavaScript, AJAX και MySQL.

Για την πειραματική αξιολόγηση της εφαρμογής δημιουργήθηκαν κάποιες μελέτες περίπτωσης ή σενάρια που φανερώνουν ότι ο Twitter Assistant μπορεί να χρησιμοποιηθεί από κάποιο χρήστη τόσο για διαφημιστικούς λόγους, όσο και για την επίτευξη συγκεκριμένων προσωπικών στόχων. Επίσης με τη μέθοδο του ερωτηματολογίου λάβαμε σημαντική ανατροφοδότηση για την αποδοχή του συστήματος αλλά και σχόλια και παρατηρήσεις που υποδεικνύουν ενδεχόμενες μελλοντικές βελτιώσεις-επεκτάσεις του έξυπνου βοηθού.

Summary

The main goal of this postgraduate dissertation is to study and develop an Intelligent Personal Assistant (IPA) for Twitter social media, named as Twitter Assistant. The Twitter Assistant will be able to automatically perform some basic twitter functions on the user' s side, considering several personal options, interests and needs of the user.

The relative study done on Intelligent Personal Assistants, Recommendation Systems and Twitter platform gave us some indications on how to design such a system. A database is used to store all available user information and a connection with Twitter API is needed to make the assistant's actions on behalf of the user. Moreover, user will also give instructions in the form of logical rules (user policy) in the system, describing his needs and interests. The Twitter Assistant suggests and performs some actions related to Twitter, using the FLASH web service, a tool based on structured argumentation that deduces logic conclusions.

A web application (Twitter Assistant) was developed using the PHP, HTML, CSS, JavaScript, AJAX and MySQL programming languages and methods. This constitutes the user interface with all subsystems mentioned above (database, Twitter API, FLASH web service).

Some case studies (scenarios) have been developed to experimentally evaluate the Twitter Assistant system, proving that this system can be used either by companies to advertise their products/services or simple users to achieve their personal goals. In addition, using the method of questionnaire, a significant feedback was received on the system's acceptance, as well as comments and remarks for future enhancements and extensions of the Twitter Assistant.

Ευχαριστίες

Στο σημείο αυτό θα ήθελα να ευχαριστήσω θερμά τον επιβλέποντα της μεταπτυχιακής μου διατριβής, Δρ. Λοΐζο Μιχαήλ για τις ιδέες, τις συμβουλές και την πολύτιμη καθοδήγηση του καθ' όλη τη διάρκεια της προσπάθειας μου να ολοκληρώσω τη διατριβή.

Ευχαριστώ πολύ επίσης τον Δρ. Αντώνη Κάκα για τη βοήθεια που μου προσέφερε κατά την περίοδο της πειραματικής αξιολόγησης του συστήματος που αναπτύχθηκε.

Επίσης, θα ήθελα να ευχαριστήσω θερμά τη σύζυγό μου Σοφία και την κόρη μου Έλενα για την συμπαράσταση και την υπομονή τους για τον πολύτιμο χρόνο που έλειπα από κοντά τους κατά την εκπόνηση της μεταπτυχιακής μου διατριβής.

Περιεχόμενα

Περίληψη.....	1
Summary	2
Ευχαριστίες.....	3
Περιεχόμενα.....	4
Κεφάλαιο 1.....	7
Εισαγωγή	7
1.1 Αντικείμενο της Μεταπτυχιακής Διατριβής.....	8
1.2 Οργάνωση Κειμένου.....	8
Κεφάλαιο 2.....	11
Βιβλιογραφική Ανασκόπηση.....	11
2.1 Έξυπνοι Βοηθοί	11
2.1.1 Παραδείγματα Έξυπνων Βοηθών	12
2.2 Twitter Bots.....	14
2.3 Recommendation Systems	16
2.3.1 Εισαγωγή και μέθοδοι	16
2.3.2 Recommendation Systems για το Twitter	18
2.4 Knowledge-Based Recommendation Systems.....	20
2.5 Trigger-Action Programming.....	22
2.5.1 Μελέτη Περίπτωσης IFTTT [24].....	23
Κεφάλαιο 3.....	26
Θεωρητικό Πλαίσιο Έρευνας	26
3.1 Εισαγωγή στο Twitter.....	26
3.2 Twitter APIs.....	29
3.2.1 REST API.....	30
3.2.2 Streaming API [31]	31
3.2.3 Webhook API [32].....	32
3.2.4 Ads API [33]	33
3.3 Twitter Libraries / Codebird-php.....	33
3.4 Επιχειρηματολογία	35
3.4.1 Αφηρημένη επιχειρηματολογία	36

3.4.2	Λογική επιχειρηματολογία	37
3.4.3	Flash Web Service	38
Κεφάλαιο 4.....		41
Twitter Assistant.....		41
4.1	Περιγραφή Γενικής Ιδέας.....	41
4.1.1	Εισαγωγή	41
4.1.2	Πιθανά Σενάρια Έξυπνου Βοηθού	42
4.1.3	Πιθανές Είσοδοι και Ενέργειες Έξυπνου Βοηθού.....	43
4.2	Αρχιτεκτονική Συστήματος	46
4.3	Διεπαφή Χρήστη Twitter Assistant.....	49
4.3.1	Σύνδεση στο σύστημα (Login).....	49
4.3.2	Εγγραφή στο σύστημα (Registration).....	50
4.3.3	Twitter Wall.....	51
4.3.4	System State.....	52
4.3.5	User Settings.....	53
4.3.6	User Profile.....	54
4.3.7	User Policy	54
4.3.8	Query Results	58
4.3.9	Κουμπί “Save Profile and Policy”	58
4.3.10	Κουμπί “Submit Query”	59
4.3.11	Κουμπί “Execute Actions”	59
4.3.12	Assistant Wall.....	59
4.3.13	Execution Logs	60
4.4	Σενάρια-Παραδείγματα χρήσης Twitter Assistant.....	60
4.4.1	Παράδειγμα 1 (Χρήση όλων των πιθανών ενεργειών).....	61
4.4.2	Παράδειγμα 2 (χρήση ενδιάμεσων εννοιών).....	63
4.5	Βάση Δεδομένων	64
4.5.1	Πίνακας Στοιχείων Χρήστη	65
4.5.2	Πίνακας Tweets Χρήστη	67
4.6	FLASH web service.....	68
4.7	Twitter API (codebird-php).....	70
4.7.1	Πιστοποίηση Χρήστη.....	71
4.7.2	Συλλογή Tweets από το Home Timeline του Χρήστη	72

4.7.3 Εκτέλεση Ενεργειών Twitter.....	74
Κεφάλαιο 5.....	77
Αξιολόγηση.....	77
5.1 Μελέτες Περίπτωσης.....	77
5.1.1 Σενάριο Α.....	77
5.1.2 Σενάριο Β.....	79
5.1.3 Σενάριο Γ.....	81
5.2 Οργάνωση Πειράματος.....	83
5.3 Αποτελέσματα.....	83
5.3.1 Μελέτες Περίπτωσης.....	83
5.3.2 Αποτελέσματα Πειράματος.....	84
5.3.3 Ανάλυση Αποτελεσμάτων Ερωτηματολογίου.....	85
Κεφάλαιο 6.....	96
Επίλογος.....	96
6.1 Σύνοψη και Συμπεράσματα.....	96
6.2 Μελλοντικές Επεκτάσεις.....	97
Βιβλιογραφία.....	99
Πίνακας Εικόνων.....	103
Παράρτημα Α.....	105
Οδηγίες για την εισαγωγή του χρήστη στον Twitter Assistant.....	105
Παράρτημα Β.....	109
Πείραμα και ερωτηματολόγιο.....	109
B.1 Πείραμα.....	109
B.2 Ερωτηματολόγιο.....	118

Κεφάλαιο 1

Εισαγωγή

Ένα από τα σημαντικότερα ζητήματα που απασχολούν την επιστήμη των υπολογιστών και τον τομέα της τεχνητής νοημοσύνης είναι η επικοινωνία και διασύνδεση ανθρώπου-μηχανής. Η κλασική μορφή επικοινωνίας μεταξύ τους μέχρι πρόσφατα περιοριζόταν στην εισαγωγή κάποιων δεδομένων που έδινε ο άνθρωπος στον υπολογιστή μέσω πληκτρολογίου και ποντικιού και στην απόκριση του υπολογιστή μέσω κειμένων ή γραφικών παραστάσεων. Με την εξέλιξη της τεχνολογίας και την πληθώρα ηλεκτρονικών συσκευών που κυκλοφορούν (smartphones, tablets) αναπτύχθηκε η ανάγκη για αμφίδρομη επικοινωνία μεταξύ ανθρώπου και μηχανής.

Οι έξυπνοι βοηθοί (Intelligent/Virtual Personal Assistants) είναι ένα ενδιάμεσο μέσο που καλούνται να καλύψουν αυτή την ανάγκη. Ο στόχος είναι να δημιουργηθούν έξυπνοι βοηθοί που να καταλαβαίνουν τη φυσική γλώσσα του ανθρώπου. Υπάρχει ένας ξεχωριστός τομέας της επιστήμης υπολογιστών, η επεξεργασία φυσικής γλώσσας (Natural Language Processing) που ασχολείται με αυτό το αντικείμενο. Ένα άλλο στοιχείο που πρέπει να λάβει υπόψιν ένας έξυπνος βοηθός είναι να καταλάβει τις προτιμήσεις του χρήστη όπως αποτυπώνονται στο προφίλ του στις διάφορες διαδικτυακές υπηρεσίες που συνδέεται, αλλά και να κατανοήσει τα συναισθήματά του. Το τελευταίο επιτυγχάνεται τόσο με χρήση εργαλείων LIWC (Linguistic Inquiry and Word Count) όσο και με ανάλυση της ομιλίας του. Η χρήση και επικοινωνία με διάφορους αισθητήρες (πχ. GPS) μπορεί επίσης να βοηθήσει να κατανοήσουμε καλύτερα τον κόσμο γύρω μας. Μία εξίσου σημαντική πληροφορία που πρέπει να έχουν οι έξυπνοι βοηθοί, ιδιαίτερα όσοι σχετίζονται με μέσα κοινωνικής δικτύωσης, είναι ο χάρτης των συνδέσεων-επαφών μας με άλλους ανθρώπους. Τέλος, η εύρεση κατάλληλων τεχνικών για την ενσωμάτωση διάφορων διαδικτυακών εφαρμογών στον έξυπνο βοηθό είναι απαραίτητη, ώστε να υπάρχει ανταλλαγή πληροφοριών ανάμεσα στο χρήστη και τις εφαρμογές που τον ενδιαφέρουν.

1.1 Αντικείμενο της Μεταπτυχιακής Διατριβής

Τα μέσα κοινωνικής δικτύωσης χρησιμοποιούνται ολοένα και περισσότερο τόσο από απλούς χρήστες για διασκέδαση και επικοινωνία όσο και από επιχειρήσεις για λόγους διαφήμισης. Η ενασχόληση με τα μέσα αυτά συχνά απαιτεί πολύ χρόνο και είναι αντιπαραγωγική. Η ανάπτυξη ενός έξυπνου τεχνητού βοηθού που να αναλαμβάνει το σχεδιασμό και την προβολή της εικόνας του ατόμου ή της επιχείρησης σε μέσα κοινωνικής δικτύωσης αποτελεί ένα ενδιαφέρον ερευνητικό θέμα με πολλές πρακτικές εφαρμογές.

Σκοπός της διατριβής είναι η μελέτη και η ανάπτυξη λογισμικού που να λειτουργεί ως έξυπνος βοηθός για τη διαχείριση του λογαριασμού Twitter ενός χρήστη. Με αυτόν τον τρόπο, ο έξυπνος βοηθός θα είναι σε θέση να εκτελεί αυτόματα κάποιες βασικές λειτουργίες εκ μέρους του χρήστη στο Twitter, λαμβάνοντας υπόψιν μια σειρά από επιλογές, ενδιαφέροντα και ανάγκες του χρήστη.

Τα βασικά ερευνητικά ερωτήματα της διατριβής είναι τα παρακάτω :

1. Ποιες λειτουργίες και τι είδους εργασίες του χρήστη στο Twitter είναι δυνατόν να αυτοματοποιηθούν και να συμπεριληφθούν σε έναν τέτοιο έξυπνο βοηθό;
2. Ποιες τεχνολογίες, εργαλεία, γλώσσες προγραμματισμού κλπ. απαιτούνται για την ανάπτυξη ενός τέτοιου συστήματος;
3. Πόσο χρήσιμη και αποτελεσματική θα είναι μια τέτοια εφαρμογή για ένα χρήστη του Twitter;

1.2 Οργάνωση Κειμένου

Στην παρούσα ενότητα παρουσιάζεται η οργάνωση του κειμένου της μεταπτυχιακής διατριβής και μια περίληψη από τα βήματα που αναλύονται σε κάθε κεφάλαιό του.

Το 1^ο Κεφάλαιο περιέχει μια εισαγωγή για τους έξυπνους βοηθούς και το ρόλο που διαδραματίζουν στη σημερινή εποχή. Στη συνέχεια παρουσιάζεται το αντικείμενο και ο σκοπός της διατριβής, όπου τίθενται τα αντίστοιχα βασικά ερευνητικά ερωτήματα. Ακολουθεί μια περίληψη των εννοιών του κειμένου.

Το 2^ο Κεφάλαιο περιέχει την απαραίτητη Βιβλιογραφική Ανασκόπηση γύρω από τους έξυπνους βοηθούς, όπου και παρουσιάζονται οι δημοφιλέστεροι έξυπνοι βοηθοί που έχουν αναπτυχθεί. Στη συνέχεια παρουσιάζονται παρόμοια συστήματα που μπορούν να κατηγοριοποιηθούν ως Internet Bots και ως συστήματα συστάσεων (Recommendation Systems) και δίνονται παραδείγματα τέτοιων βοηθών με έμφαση στο Twitter. Στη συνέχεια γίνεται αναφορά στα συστήματα συστάσεων βασισμένα στη γνώση (Knowledge-based Recommendation Systems) και στον trigger-action προγραμματισμό. Η θεωρία που αναφέρεται στις δυο τελευταίες ενότητες μας ενέπνευσαν ώστε να προκύψει η ιδέα για την δημιουργία ενός έξυπνου βοηθού για το Twitter.

Το 3^ο Κεφάλαιο αποτελεί τη θεωρητική μελέτη που έγινε στο πλαίσιο της ανάλυσης των απαιτήσεων για τον σχεδιασμό του έξυπνου βοηθού για το Twitter. Συγκεκριμένα έγινε μια εισαγωγή στον κόσμο του Twitter και δόθηκε επεξήγηση σε ορισμένες βασικές έννοιες που πρέπει να γνωρίζει ο χρήστης για να χρησιμοποιήσει το βοηθό. Επίσης γίνεται παρουσίαση των Twitter APIs που είναι διαθέσιμα προς προγραμματιστές που επιθυμούν να χρησιμοποιήσουν την πλατφόρμα του Twitter σε κάποια εφαρμογή τους. Για την ανάπτυξη της δικής μας εφαρμογής θα χρησιμοποιήσουμε το REST API και τη βιβλιοθήκη ανοικτού κώδικα Codebird-php. Τέλος γίνεται μια αναφορά στα είδη επιχειρηματολογίας (αφηρημένη και λογική) και στο εργαλείο εξαγωγής λογικών συμπερασμάτων FLASH που χρησιμοποιήσαμε για την υλοποίηση του έξυπνου βοηθού.

Στο 4^ο Κεφάλαιο γίνεται μια αναλυτική παρουσίαση του έξυπνου βοηθού που υλοποιήθηκε στα πλαίσια της μεταπτυχιακής διατριβής (Twitter Assistant). Περιγράφεται η γενική ιδέα που εξυπηρετεί καθώς και η αρχιτεκτονική του συστήματος κατά την φάση του σχεδιασμού. Στη συνέχεια δίνονται αναλυτικές οδηγίες χρήσης του συστήματος, αναλύοντας όλα τα πεδία στο γραφικό περιβάλλον της διεπαφής χρήστη, που αποτελεί τον κορμό του συστήματος. Επίσης περιγράφονται τα σημαντικότερα στοιχεία κατά την υλοποίηση του συστήματος και τη σύνδεση της διεπαφής χρήστη με τα υπόλοιπα υποσυστήματα (Βάση Δεδομένων, FLASH Web Service και Twitter REST API – codebird-php).

Το 5^ο Κεφάλαιο αναφέρεται στην πειραματική αξιολόγηση του Twitter Assistant. Δίνονται ορισμένες μελέτες περίπτωσης για να διαπιστωθεί η χρησιμότητα του έξυπνου βοηθού. Στη συνέχεια περιγράφεται η διαδικασία σύμφωνα με την οποία διεξήγαμε ένα πείραμα χρήσης του συστήματος που αναπτύχθηκε και τέλος αναλύονται τα αποτελέσματα του πειράματος.

Το 6^ο Κεφάλαιο αποτελεί τον επίλογο, όπου παρουσιάζονται τα συμπεράσματα που προέκυψαν κατά τη μελέτη, το σχεδιασμό και την υλοποίηση του συστήματος Twitter Assistant. Επίσης γίνονται προτάσεις για πιθανές βελτιώσεις και επεκτάσεις του συστήματος που αναπτύχθηκε.

Κεφάλαιο 2

Βιβλιογραφική Ανασκόπηση

2.1 Έξυπνοι Βοηθοί

Ο έξυπνος βοηθός (Intelligent/Virtual Personal Assistant) είναι ένας πράκτορας λογισμικού που μπορεί να εκτελεί διάφορες εργασίες και υπηρεσίες εκ μέρους ενός ανθρώπου-χρήστη του βοηθού. Αυτές οι εργασίες βασίζονται σε δεδομένα που μπορεί να εισάγει ο χρήστης μέσω των προτιμήσεών του και από μια σειρά από δεδομένα που μπορεί να λάβει ο έξυπνος βοηθός από διάφορες πηγές του διαδικτύου, όπως για παράδειγμα καιρός, μετοχές, πρόγραμμα ταινιών στον κινηματογράφο, προγνωστικά αγώνων κ.ά. Η παρακάτω εικόνα αποτελεί μια γενική αναπαράσταση των εισόδων και των εξόδων ενός έξυπνου βοηθού.



Εικόνα 2.1 : Είσοδοι και εξοδοι ενός έξυπνου βοηθού

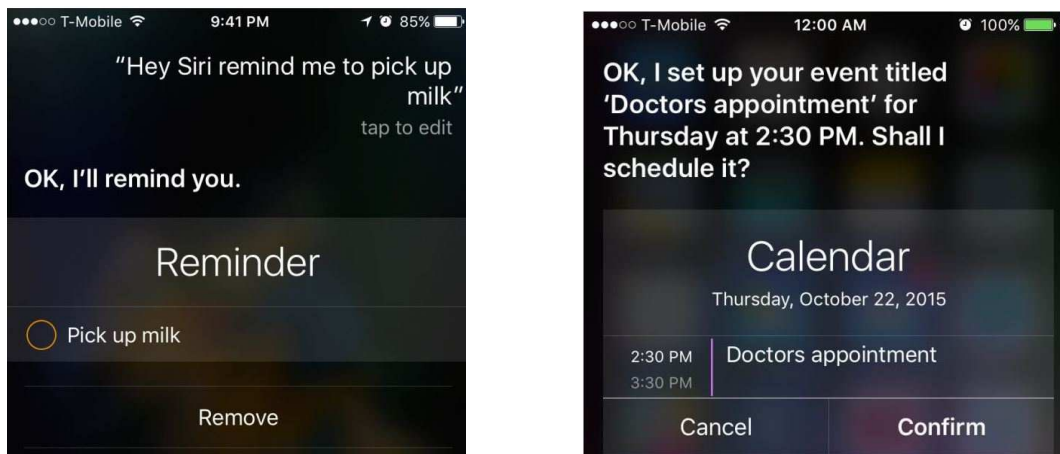
Η ανάπτυξη και η σημαντική πρόοδος που σημειώθηκε στον τομέα της Τεχνητής Νοημοσύνης συνέβαλε στην δημιουργία σημαντικών έξυπνων βοηθών. Κοινό χαρακτηριστικό όλων των έξυπνων βοηθών είναι η ενσωμάτωση συστημάτων αναγνώρισης ομιλίας (speech recognition), φυσικής επεξεργασίας γλώσσας (natural language processing) και η σύνδεση με μεγάλες βάσεις δεδομένων σημασιολογικής αναζήτησης (semantic search databases). Είναι γεγονός πως οι γνωστότερες εταιρίες-κολοσσοί της πληροφορικής έχουν επενδύσει πολλά στην ανάπτυξη τέτοιων εργαλείων τα τελευταία χρόνια.

2.1.1 Παραδείγματα Έξυπνων Βοηθών

Στην ενότητα αυτή παρουσιάζονται οι πιο δημοφιλείς έξυπνοι βοηθοί που κυκλοφορούν ως προϊόντα λογισμικού σε smartphones ή υπηρεσίες διαδικτύου.

Siri

Ένας από τους πρώτους έξυπνους βοηθούς που αναπτύχθηκαν είναι ο Siri της εταιρίας Apple (από το 2011). Ο Siri αναλαμβάνει να δέχεται τις φωνητικές εντολές μας και να εκτελεί τις ανάλογες εργασίες. Για παράδειγμα, αυτό μας δίνει τη δυνατότητα να στέλνουμε φωνητικά μηνύματα, να καλούμε κάποιον στο τηλέφωνο, να επιλέγουμε τη μουσική που θέλουμε να ακούσουμε ή να μαθαίνουμε την ακριβή θέση μας. Επίσης μπορούμε να κάνουμε στο βοηθό μίβοα οποιαδήποτε ερώτηση και αυτός μας απαντάει αναλόγως ή εκτελεί την εντολή που του υπαγορεύουμε. Ο Siri συνεργάζεται με πολλές υπηρεσίες διαδικτύου όπως οι iMessage, Mail, Calendars, Reminders, Notes, Stocks, Weather, Music, Safari, Google κ.ά. Υποστηρίζεται από τα λειτουργικά συστήματα iOS, watchOS, macOS, tvOS, ενώ είναι διαθέσιμος και σε αυτοκίνητα που διαθέτουν το ολοκληρωμένο σύστημα διασκέδασης CarPlay [1][2]. Στις παρακάτω εικόνες φαίνονται κάποια παραδείγματα χρήσης ημερολογίου και υπενθύμισης στον Siri.

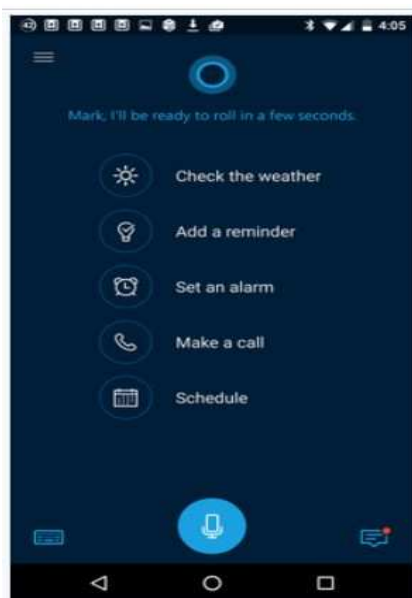


Εικόνα 2.2 : Παραδείγματα χρήσης Siri

Cortana

Ο Cortana είναι ο αντίστοιχος έξυπνος βοηθός που έχει αναπτύξει η εταιρία Microsoft. Μερικές από τις λειτουργίες που προσφέρει είναι να διευκολύνει το χρήστη του σε συνομιλίες, κλήσεις, μηνύματα, ημερολόγιο, υπενθυμίσεις, αφυπνίσεις, σημειώσεις, χάρτες και οδηγίες. Μπορεί να δώσει επίσης κάθε είδους πληροφορία σχετική με μουσική, καιρό, πτήσεις, αθλητικά,

οικονομικά, υγεία και άσκηση κ.ά. Επίσης, όσο περισσότερο χρησιμοποιείται ο Cortana τόσο πιο εξατομικευμένη είναι και η εμπειρία του χρήστη, καθώς λαμβάνει υπόψιν του τις προηγούμενες προτιμήσεις και αναζητήσεις του. Ένα επιπλέον χαρακτηριστικό του Cortana σε σχέση με άλλους έξυπνους βοηθούς είναι ότι επιτρέπει στους χρήστες να επιλέξουν κάποιο επίπεδο προστασίας της ιδιωτικότητάς τους καθορίζοντας ποιες πληροφορίες θα συλλέγονται από το βοηθό και ποιες όχι. Επίσης ο βοηθός έχει ένα σύστημα υπενθυμίσεων που μπορεί να συσχετιστεί με μία συγκεκριμένη επαφή του χρήστη, οπότε για παράδειγμα θα ενεργοποιείται η υπενθύμιση όταν υπάρχει κάποια κλήση μεταξύ του χρήστη και της συγκεκριμένης επαφής. Ο Cortana υποστηρίζεται από τα λειτουργικά συστήματα Windows 10, Windows 10 Mobile, Windows Phone 8.1, Microsoft Band, Xbox One, iOS και Android[3][4]. Το γραφικό περιβάλλον (GUI) του έξυπνου βοηθού Cortana φαίνεται στην παρακάτω εικόνα.

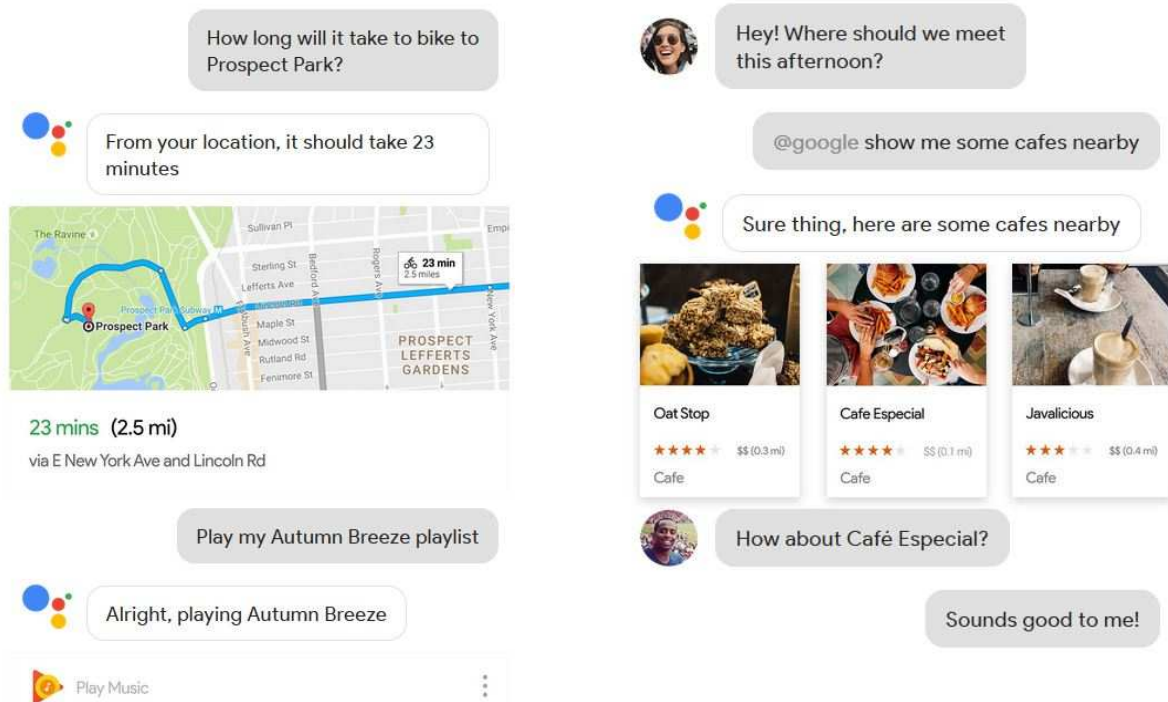


Εικόνα 2.3 : Γραφικό περιβάλλον Cortana

Google Assistant

Το αντίστοιχο προϊόν της εταιρίας Google ονομάζεται Google Assistant. Διαθέτει μια διεπαφή χρήστη για επικοινωνία σε φυσική γλώσσα για να απαντάει σε ερωτήσεις, να κάνει συστάσεις και να πραγματοποιεί εργασίες αναζητώντας σχετικές πληροφορίες από μια σειρά από διαδικτυακές υπηρεσίες. Επίσης μπορεί να δώσει στο χρήστη κάποιες πληροφορίες που προβλέπει ότι μπορεί να φανούν χρήσιμες σε αυτόν. Ένα επιπλέον χαρακτηριστικό του είναι ότι αναγνωρίζει επαναλαμβανόμενες ενέργειες του χρήστη στον υπολογιστή που είναι εγκατεστημένο, όπως κοινές τοποθεσίες, συνεχόμενες συναντήσεις του χρήστη στο

ημερολόγιο και ίδιες αναζητήσεις εμφανίζοντας σχετικές πληροφορίες στο χρήστη με τη μορφή καρτών. Υπάρχουν πολλά είδη καρτών ανάλογα με τις πάμπολλες εφαρμογές που υποστηρίζει ο έξυπνος βοηθός, όπως για παράδειγμα πληροφορίες για ειδήσεις, ξενοδοχεία, πτήσεις, καιρό, μετοχές κ.ά. [5] [6]. Στην παρακάτω εικόνα φαίνονται παραδείγματα από το περιβάλλον του Google Assistant.



Εικόνα 2.4 : Παραδείγματα χρήσης Google Assistant

2.2 Twitter Bots

Οι έξυπνοι βοηθοί που αναφέρθηκαν στην ενότητα 2.1 προσφέρουν ορισμένες δυνατότητες αλληλεπίδρασης με το Twitter. Για παράδειγμα αναφέρεται πως όλοι τους μπορούν να πραγματοποιήσουν απλή δημοσίευση (post) αλλά και αναζήτηση tweets. Ωστόσο, από σχετική βιβλιογραφική αναζήτηση που έγινε δε φαίνεται να υπάρχει κάποιος ολοκληρωμένος έξυπνος βοηθός εξειδικευμένος στις λειτουργίες του Twitter.

Αντίθετα, υπάρχουν αρκετές εφαρμογές που εκτελούν ορισμένες ενέργειες του Twitter που ταιριάζουν περισσότερο στην κατηγορία των Internet Bots, αναφερόμενοι συχνά και ως Twitter Bots. Με τις εφαρμογές αυτές είναι δυνατόν να αυτοματοποιηθούν κάποιες ενέργειες που κλασικά γίνονται από έναν λογαριασμό Twitter. Σύμφωνα με το άρθρο της Mowbray (2014) [7] έχει υπολογιστεί ότι το 24% όλων των δημόσιων tweets που δημοσιεύονται

προέρχονται από κάποιου είδους Twitter Bot. Τυπικά, ένα Twitter Bot μπορεί να στείλει αυτόματα tweets ή Direct Messages (DMs), παρόλο που ο χρήστης του μπορεί ταυτόχρονα να στείλει και κάποιο μη-αυτοματοποιημένο tweet μέσω του λογαριασμού του. Πολλά bots επίσης κάνουν ή αποδέχονται αιτήματα για φιλία (friend request) αυτόματα. Το περιεχόμενο των tweets που προέρχονται από bots μπορεί να είναι προσχεδιασμένο από τον χρήστη του ή μπορεί σε ορισμένες περιπτώσεις να προκύπτει αλγοριθμικά εξαρτώμενο από τη συμπεριφορά άλλων χρηστών του Twitter. Το Twitter πάντως ενθαρρύνει τη χρήση αυτοματοποίησης και δεν απαιτεί από bots λογαριασμούς τη δήλωση τους ότι λειτουργούν ως bots. Ωστόσο, απαγορεύει κάποιους τύπους αυτοματοποιημένης συμπεριφοράς που υιοθετούνται από spammers [8].

Υπάρχουν διάφορες κατηγορίες Twitter Bots ανάλογα με το σκοπό που εξυπηρετεί το καθένα. Η δημοφιλέστερη χρήση των Twitter Bots είναι η διαφήμιση, οπότε μπορούν απλά να στέλνουν κάποιο μήνυμα με διαφημιστικό περιεχόμενο ή να προσελκύουν καινούριους χρήστες και πιθανούς πελάτες. Άλλα bots έχουν ως στόχο απλώς την ενημέρωση των χρηστών για διάφορα θέματα όπως π.χ ειδήσεις, καιρός, μετοχές κ.ά. Επίσης, υπάρχει μια κατηγορία bots που σκοπεύει στη διασκέδαση των χρηστών αναζητώντας αλληλεπίδραση με αυτούς, ενώ δε λείπουν και tweets που προέρχονται από κάποια συσκευή που έχει δυνατότητα επικοινωνίας με το διαδίκτυο και κατά συνέπεια με το Twitter, σύμφωνα με την τεχνολογία “Internet of Things”. Φυσικά, υπάρχουν και κάποιες εφαρμογές που έχουν κακόβουλα κίνητρα (Spambots), όπως να αποκτήσουν πρόσβαση στο λογαριασμό του χρήστη, να διαδώσουν ψεύτικες ειδήσεις, να αναδείξουν ή να αποκρύψουν κατά περίπτωση πολιτικές συζητήσεις (Mowbray, 2014) [7]. Γι’ αυτό το λόγο υπάρχει ειδική ομάδα στο Twitter που ασχολείται με την ανίχνευση και την απαγόρευση τέτοιων λογαριασμών που παραβιάζουν τους κανόνες ορθής λειτουργίας [9].

Ως προς την αξιολόγηση των Twitter Bots, σχετική μελέτη των Edwards et al. (2016) [10] εκτιμά ότι μπορεί να είναι το ίδιο αποτελεσματικά όσο και ένας ανθρώπινος παράγοντας στις περιοχές της γνωστικής επεξεργασίας (συσχέτιση μεταξύ νέας πληροφορίας και υφιστάμενης γνώσης), της αναζήτησης πληροφορίας και των συμπερασμάτων μάθησης.

Ορισμένα παραδείγματα Twitter Bots αναφέρονται στη λίστα που ακολουθεί :

- Ο “CoBot” είναι ένας έξυπνος βοηθός που υποδεικνύει σε κάποιον χρήστη του Twitter ποιός από τους followers του είναι περισσότερο ικανός να απαντήσει σε κάποια ερώτηση του χρήστη, ανάλογα με τα ενδιαφέροντά του. Αυτά προκύπτουν μετά από

κατάλληλη στατιστική ανάλυση και χρήση similarity metrics στα προφίλ των χρηστών και στο περιεχόμενο των tweets τους. (Edwards et al. , 2013) [11]

- Ο “Tadvice” είναι ένας έξυπνος βοηθός που λαμβάνοντας υπόψιν το προφίλ του χρήστη, του προτείνει να κάνει ή να μην κάνει retweet σε κάθε καινούριο μήνυμα που εμφανίζεται στο χρονολόγιό του. Έτσι ο χρήστης γνωρίζει καλύτερα τους followers του και μπορεί να επεκτείνει καλύτερα το κοινωνικό του δίκτυο. (Nasirifard et al., 2011) [12]
- Η εφαρμογή “commun.it” δημιουργεί προγραμματισμένες δημοσιεύσεις (posts) και έχει τη δυνατότητα να βοηθήσει τον χρήστη να επεκτείνει την επιρροή του επιλέγοντας ποιοτικούς followers. [13]
- Η διαδικτυακή εφαρμογή “www.tweetify.us” επιλέγει τυχαία νέα στοιχεία από το περιεχόμενο του χρήστη, όπως δικές του ιστοσελίδες ή blogs, δημιουργεί διάφορες κατηγορίες tweets και τα κοινοποιεί με προγραμματισμένα posts, ώστε να κρατάει πάντα ενήμερο το κοινό που τον ακολουθεί. [14]

2.3 Recommendation Systems

2.3.1 Εισαγωγή και μέθοδοι

Η πληθώρα των πηγών που υπάρχουν στο διαδίκτυο για πληροφορίες ποικίλου είδους (όπως προϊόντα, ειδήσεις, ταινίες, μουσική κλπ) πολλές φορές δυσκολεύει τους χρήστες να επιλέξουν την καλύτερη ή την πιο κατάλληλη, καθώς απαιτείται να αφιερώσουν χρόνο ή να έχουν συγκεκριμένη γνώση για το θέμα που τους ενδιαφέρει. Τα συστήματα εισηγήσεων (Recommendation Systems) μπορούν να δώσουν λύση στο παραπάνω πρόβλημα, καθώς βοηθούν τους χρήστες να διαλέξουν τις καταλληλότερες πηγές που τους ενδιαφέρει ή είναι χρήσιμες γι’ αυτούς. Η ερευνητική περιοχή που ασχολείται με τα συστήματα εισηγήσεων έχει γίνει ιδιαίτερα δημοφιλής τα τελευταία χρόνια, μιας και παρουσιάζει πολλές πρακτικές εφαρμογές. Για παράδειγμα, τέτοιου είδους συστήματα έχουν αναπτυχθεί για την επιλογή ειδησεογραφικών άρθρων, βιβλίων, ταινιών, μουσικών CD, ταξιδιών κλπ.

Οι προτάσεις που κάνει ένα σύστημα εισηγήσεων βασίζονται στα διαθέσιμα δεδομένα που λαμβάνει το σύστημα. Αυτά τα δεδομένα μπορεί να είναι κάποια likes/dislikes που έχει κάνει ο χρήστης ή κάποια πιθανή βαθμολόγηση προϊόντος ή υπηρεσίας που χρησιμοποίησε. Επίσης μπορεί να είναι κάποιες προτιμήσεις του χρήστη στις σελίδες που έχει επισκεφτεί ή ακόμα και εξωτερικές πληροφορίες για τον χρήστη που μπορούν να συλλεχθούν από άλλες πηγές. Τα δεδομένα μπορεί να προκύπτουν ακόμα και από συνδυασμό των παραπάνω πληροφοριών.

Οι μέθοδοι που εφαρμόζονται στα συστήματα εισηγήσεων εξαρτώνται από τις διαθέσιμες πληροφορίες και είναι οι παρακάτω :

1. Συνεργατικό φιλτράρισμα (Collaborative Filtering) : Είναι η περισσότερη διαδεδομένη μέθοδος, όπου συνήθως η βαθμολογία που δίνεται σε κάποιο συγκεκριμένο προϊόν/υπηρεσία από τον χρήστη συγκρίνεται με τις αντίστοιχες βαθμολογίες άλλων (παρόμοιων) χρηστών με τη βοήθεια ενός μέτρου ομοιότητας (similarity measure). Γνωστοί αλγόριθμοι που χρησιμοποιούνται για τον υπολογισμό αυτού του μέτρου είναι ο κ-πλησιέστερος γείτονας (k-nearest neighbor) και ο συντελεστής συσχέτισης Pearson (Pearson correlation). Αν σε έναν παρόμοιο χρήστη (με κοινά χαρακτηριστικά) αρέσει ένα αντικείμενο/πληροφορία τότε είναι πολύ πιθανό να αρέσει και στον χρήστη που μας ενδιαφέρει. Η μέθοδος αυτή μπορεί να βασίζεται είτε στο αντικείμενο (item based collaborative filtering), είτε στο χρήστη (user based collaborative filtering).

Το συνεργατικό φιλτράρισμα είναι σχετικά απλό, αποτελεσματικό και ακριβές. Αντίθετα παρουσιάζει τα παρακάτω μειονεκτήματα :

- “Cold start” problem: Τα συστήματα απαιτούν μεγάλο όγκο δεδομένων για έναν συγκεκριμένο χρήστη για να αρχίσουν να κάνουν ακριβείς συστάσεις
- Scalability problem: Μεγάλη υπολογιστική ισχύς απαιτείται για να υπολογιστούν συστάσεις σε συστήματα με εκατομμύρια χρήστες και προϊόντα.
- Sparsity problem: Ο αριθμός των προϊόντων που έχουν πουληθεί σε μεγάλα e-commerce sites είναι εξαιρετικά μεγάλος. Από την άλλη, ακόμα και οι πιο δραστήριοι χρήστες έχουν βαθμολογήσει ελάχιστα προϊόντα από το σύνολο που υπάρχει στην αντίστοιχη βάση δεδομένων. Έτσι ακόμα και τα πιο δημοφιλή προϊόντα έχουν λίγες κριτικές.

2. Φιλτράρισμα με βάση το περιεχόμενο (Content-based Filtering): Αυτή η μέθοδος προτείνεται αν ο χρήστης έχει δείξει κάποια προτίμηση σε συγκεκριμένα αντικείμενα στο παρελθόν (πχ. έχει αγοράσει προϊόντα). Το σύστημα συστάσεων σε αυτή την περίπτωση θα προτείνει κάποιο παρόμοιο αντικείμενο-προϊόν. Σε αυτή τη μέθοδο αναζητούνται ομοιότητες μεταξύ αντικειμένων, ενώ στο συνεργατικό φιλτράρισμα ομοιότητες μεταξύ των χρηστών. Επιλέγεται αυτή η μέθοδος όταν υπάρχουν διαθέσιμες πολλές πηγές πληροφοριών.

Με τη μέθοδο αυτή εξαλείφονται τα προβλήματα που υπάρχουν στο συνεργατικό φιλτράρισμα, ενώ οι συστάσεις που δίνονται βασίζονται στις προτιμήσεις του χρήστη. Αντίθετα, η μέθοδος βασίζεται στην προηγούμενη γνώση του χρήστη.

3. Υβριδικό φιλτράρισμα (Hybrid Filtering) : Μπορεί επίσης να ακολουθηθεί συνδυασμός των δυο παραπάνω μεθόδων για να αυξηθεί η αποτελεσματικότητα ενός συστήματος συστάσεων.

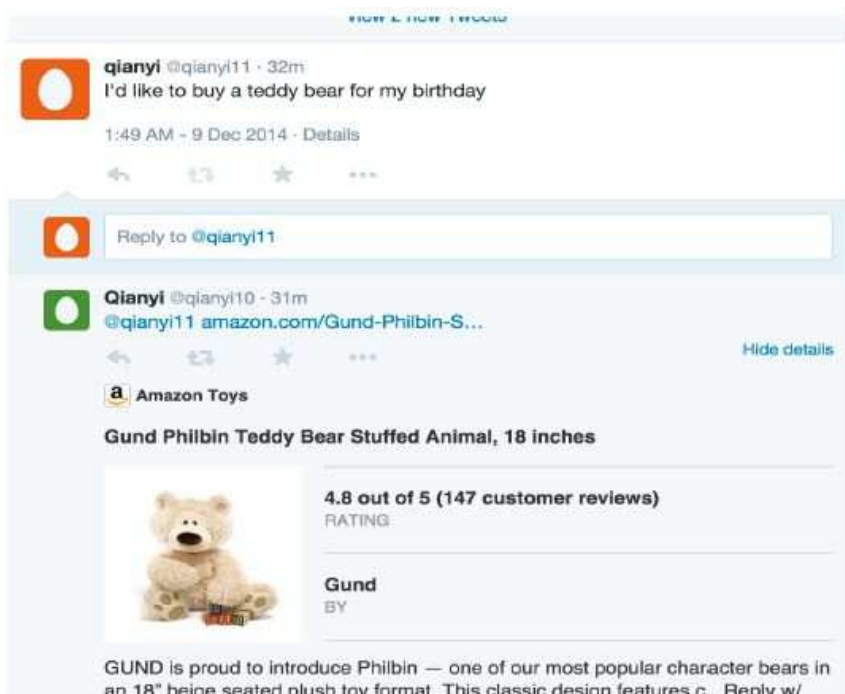
2.3.2 Recommendation Systems για το Twitter

Στους έξυπνους βοηθούς για το Twitter συγκαταλέγονται επίσης συστήματα συστάσεων (Recommendation Systems). Στην ορολογία του Twitter, το hashtag είναι μια λέξη ή φράση που ακολουθείται από το σύμβολο # και μπορεί να τοποθετηθεί στο περιεχόμενο ενός tweet. Με τον τρόπο αυτό μπορεί να γίνει διαχωρισμός των tweets σε θέματα, ώστε κάθε χρήστης να μπορεί να κατηγοριοποιεί τις δημοσιεύσεις του ή να βρίσκει θέματα που τον ενδιαφέρουν. Πολλά συστήματα συστάσεων έχουν αναπτυχθεί με βάση αυτό το χαρακτηριστικό. Στη μελέτη των Otsuka et al. (2014) [15] προτείνεται ένα σύστημα συστάσεων που βοηθάει τον χρήστη να επιλέξει προσωπικά hashtags που να σχετίζονται με τα ενδιαφέροντά του, όπως προκύπτει από τα tweets που έχει δημιουργήσει. Μια παρόμοια μελέτη των Rahman et al. (2013) [16] συσχετίζει 2 ή περισσότερα hashtags που βρίσκονται στο περιεχόμενο των tweets και μπορούν να έχουν κοινά χαρακτηριστικά. Το σύστημα συστάσεων που έχει δημιουργηθεί (TREK) προτείνει στον χρήστη τα κατάλληλα hashtags ανάλογα με το μέτρο σύγκρισης που κρίνει τη σχετική συνάφεια.

Μια ενδιαφέρουσα πρόταση παρουσιάζεται στη μελέτη των AlMeshary et al. (2013) [17], όπου λαμβάνεται υπόψιν επιπλέον και η τοποθεσία που βρίσκεται ο χρήστης. Συγκεκριμένα η ιδέα είναι να βοηθηθεί κάποιος χρήστης του Twitter να αναπτύξει το κοινωνικό δίκτυο του μετά

από αλλαγή στην περιοχή ή στην χώρα διαμονής του. Το σύστημα προτείνει στον χρήστη followings, followers και hashtags βασισμένα τόσο στα ενδιαφέροντα του όσο και στην γεωγραφική περιοχή που μένει. Μάλιστα υποστηρίζει μετάφραση των tweets που έχει πραγματοποιήσει ο χρήστης μεταξύ δυο γλωσσών (για παράδειγμα Αγγλικά και Αραβικά) με χρήση τεχνικών NLP και του Google Translate API.

Πολύ συχνή χρήση συστημάτων συστάσεων γίνεται για διαφημιστικούς και εμπορικούς σκοπούς. Στη μελέτη των Liu et al. (2014) [18] παρουσιάζεται ένα σύστημα που προτείνει σε χρήστες του Twitter αγορά συγκεκριμένων προϊόντων για τα οποία ο χρήστης έχει εκδηλώσει ενδιαφέρον (μέσω των tweets που έχει κάνει). Στην εικόνα που ακολουθεί φαίνεται ένα παράδειγμα χρήσης του συγκεκριμένου συστήματος.



Εικόνα 2.5 : Αυτόματη απάντηση του Recommendation System

Ορισμένα παραδείγματα συστημάτων συστάσεων αναφέρονται στη λίστα που ακολουθεί :

- Το "Trupi" είναι ένα σύστημα συστάσεων (Recommendation System) για το Twitter που βασίζεται στα προσωπικά ενδιαφέροντα του χρήστη. Στόχος του είναι να παρουσιάσει τα tweets που εμφανίζονται στο χρονολόγιο του χρήστη με σειρά ώστε τα πιο ενδιαφέροντα να εμφανίζονται πρώτα. (Elmongui et al. , 2015) [19]

- Το “Twilite” είναι ένα άλλο σύστημα συστάσεων που βασίζεται στο πιθανολογικό μοντέλο και κάνει χρήση της τεχνικής LDA (Latent Dirichlet Allocation). Ο βοηθός αυτός προτείνει τους top-k χρήστες που μπορεί να ακολουθήσει ένας χρήστης του συστήματος καθώς και να διαβάσει τα top-k tweets από έναν συγκεκριμένο χρήστη που τον ενδιαφέρει. Για να το πετύχει αυτό το σύστημα εκμεταλλεύεται τόσο τις σχέσεις φιλιάς που υπάρχουν στο Twitter, όσο και το περιεχόμενο των tweets. (Kim et al., 2014) [20]
- Ένα παρόμοιο σύστημα προτείνεται και στην έρευνα των Dhillon et al. (2016) [21], όπου με τη μέθοδο της ομαδοποίησης και χρήση της τεχνικής LDA επιτυγχάνεται η μοντελοποίηση θεμάτων στο Twitter. Με αυτό τον τρόπο ο χρήστης μπορεί να ταξινομηθεί σε κάποια ομάδα που προκύπτει από τα αποτελέσματα του LDA μοντέλου και να επιλέξει τους κατάλληλους followings.

2.4 Knowledge-Based Recommendation Systems

Μια ειδική κατηγορία συστημάτων εισηγήσεων αποτελούν αυτά που βασίζονται στη γνώση και είναι γνωστά με τον όρο Knowledge-Based Recommendation Systems. Ένα τέτοιο σύστημα προτείνει προϊόντα ανάλογα με τις ανάγκες και τις προτιμήσεις του χρήστη, οπότε πρέπει να υπάρχει μαζεμένη γνώση σχετικά με τον χρήστη του συστήματος. Αντίθετα, η συμπεριφορά άλλων χρηστών δε λαμβάνεται σημαντικά υπόψιν σε αυτού του είδους τα συστήματα εισηγήσεων.

Σύμφωνα με τους Bouraga et al. (2014) [22] τα συστήματα εισηγήσεων βασισμένα στη γνώση αντιμετωπίζουν τα προβλήματα που υπάρχουν σε άλλα συστήματα συστάσεων (cold-start, scalability, sparsity problem). Επίσης επειδή η βάση γνώσης που υπάρχει στο σύστημα προέρχεται από τον χρήστη και είναι απαλλαγμένη από θορύβους, οι συστάσεις που κάνουν αυτά τα συστήματα είναι περισσότερο αξιόπιστες. Το σημαντικότερο θέμα που μας απασχολεί σε αυτά τα συστήματα είναι η δημιουργία της βάσης γνώσης, η οποία είναι ένα πολύπλοκο έργο. Αυτό συχνά απαιτεί σημαντική γνώση πάνω σε κάποιο τομέα και εξειδίκευση σε θέματα αναπαράστασης γνώσης.

Υπάρχουν δύο βασικές μέθοδοι που χρησιμοποιούνται στα συστήματα εισηγήσεων βασισμένα στη γνώση. Η πρώτη βασίζεται στη συγκεκριμένη ειδική περίπτωση που εξετάζει το σύστημα

εισηγήσεων, αναφερόμενη ως “case-based approach”. Σε αυτή την τεχνική ο χρήστης εισάγει στο σύστημα την περιγραφή του προβλήματος που θέλει να λύσει. Οι συστάσεις τότε δημιουργούνται με βάση τις προτιμήσεις του χρήστη και τη βάση γνώσης του συστήματος. Από την άλλη, η δεύτερη μέθοδος εφαρμόζει μια τεχνική με προσέγγιση βασισμένη στο περιεχόμενο (content-based approach). Αρχικά δημιουργούνται η βάση γνώσης και το προφίλ του χρήστη και στη συνέχεια υπολογίζεται ένα μέτρο ομοιότητας με σκοπό να ταιριάζει στοιχεία από τη βάση γνώσης και τις προτιμήσεις του χρήστη.

Ανεξάρτητα από τη μέθοδο που χρησιμοποιείται, τα δύο κύρια στοιχεία που συναντώνται στα συστήματα εισηγήσεων βασισμένα στη γνώση είναι η βάση γνώσης και το προφίλ του χρήστη. Ανάλογα με το είδος του συστήματος η βάση γνώσης μπορεί να είναι μια απλή βάση δεδομένων ή μπορεί να περιέχει την οντολογία ενός τομέα-συνόλου (domain ontology) ή κάποια εξειδικευμένη (expert) γνώση. Η φύση της βάσης γνώσης συχνά σχετίζεται με τη μέθοδο που ακολουθείται για τις συστάσεις και επηρεάζει η μία την άλλη. Το προφίλ του χρήστη (user profile) είναι πολύ σημαντικό για τα συστήματα εισηγήσεων, αφού στόχος τους είναι να προτείνουν εξατομικευμένες λύσεις για τον συγκεκριμένο χρήστη. Το περιεχόμενο του προφίλ χρήστη εξαρτάται από την μεθοδολογία και την τεχνική που εφαρμόζει το σύστημα. Σε γενικές όμως γραμμές μπορούμε να πούμε ότι αποτελείται από τις προτιμήσεις, τις ανάγκες, τις επιλογές και τα ενδιαφέροντα του χρήστη.

Η μελέτη των Bouraga et al. (2014) [22] προσπαθεί να κατηγοριοποιήσει διάφορα συστήματα εισηγήσεων με βάση τη γνώση. Όπως αναφέρεται, τα κύρια στοιχεία που παίζουν ρόλο σε αυτά τα συστήματα είναι τα εξής:

- Η φύση του προβλήματος, ο σκοπός του συστήματος εισηγήσεων και η λύση που προτείνεται από αυτό. Σε αυτά τα στοιχεία περιλαμβάνονται η βάση γνώσης, η στρατηγική που ακολουθείται καθώς και το περιεχόμενο των συστάσεων.
- Το προφίλ του χρήστη (user profile) : Στο σημείο αυτό εξετάζουμε με ποιό τρόπο συλλέγουμε τις πληροφορίες για το προφίλ του χρήστη. Οι πληροφορίες μπορούν να αντληθούν από τη διαδικτυακή συμπεριφορά του χρήστη (πχ. προηγούμενες αγορές, ιστορικό, clicks κλπ) ή από την άμεση συμμετοχή του χρήστη απαντώντας σε ερωτηματολόγια, συνεντεύξεις κ.ά. Επιπλέον μας απασχολεί το πότε δημιουργείται το προφίλ του χρήστη. Σε ορισμένες περιπτώσεις, όλες οι πληροφορίες που αφορούν το προφίλ του χρήστη είναι διαθέσιμες προτού ξεκινήσει το σύστημα να κάνει τις

εισηγήσεις του. Μια άλλη δυνατότητα είναι να δημιουργείται το προφίλ του χρήστη δυναμικά, όταν αυτό απαιτείται και έτσι οι προτιμήσεις του χρήστη να ανακαλύπτονται σταδιακά. Η διαδικασία αυτή συμβαίνει ενώ το σύστημα εισηγήσεων βρίσκεται σε λειτουργία, αφορά δηλαδή διαδραστικά συστήματα.

- Βαθμός Αυτοματοποίησης : Ανάλογα με τη μέθοδο που ακολουθείται μπορεί να χρειάζονται ορισμένες παρεμβάσεις του χρήστη στις φάσεις λειτουργίας ενός συστήματος εισηγήσεων. Για παράδειγμα κατά τη δημιουργία του προφίλ του χρήστη, οι σχετικές πληροφορίες είτε αντλούνται αυτόματα είτε τις προσθέτει ο χρήστης. Επίσης η δημιουργία των κανόνων συστάσεων (recommendation rules) μπορεί να γίνει αυτόματα από συλλογή κατάλληλων δεδομένων. Από την άλλη, μπορεί να μεσολαβήσει ένα στάδιο ελέγχου των συστάσεων που παράχθηκε αυτόματα από ειδικούς (experts) πριν καταλήξουν στον τελικό χρήστη του συστήματος.

2.5 Trigger-Action Programming

Τα τελευταία χρόνια έχει παρατηρηθεί μια ραγδαία αύξηση στη χρήση συσκευών (IoT) “Internet of Things” και μέσων κοινωνικής δικτύωσης. Οι χρήστες των συσκευών αυτών συχνά θέλουν να συνδυάσουν τις λειτουργίες που τους παρέχονται με κάποια on-line υπηρεσία, οπότε αναζητούν τα κατάλληλα εργαλεία που θα τους προσφέρει αυτή τη δυνατότητα. Η πιο γνωστή προσέγγιση για να το πετύχουν αυτό είναι το μοντέλο trigger-action. Οι χρήστες μπορούν να ρυθμίσουν τη συμπεριφορά ενός συστήματος προσδιορίζοντας μία αλλαγή στο σύστημα (trigger) που να προκαλεί μια επακόλουθη ενέργεια (action). Η λειτουργία αυτή μοιάζει με ένα μικρό πρόγραμμα για αυτοματοποίηση κάποιων ενεργειών, που αποκαλείται “συνταγή” (recipe).

Η απλότητα αυτής της λειτουργίας επιτρέπει τους χρήστες που δεν ξέρουν να προγραμματίζουν να μπορούν να ασχοληθούν με αυτό. Ακόμα και χρήστες χωρίς καθόλου τεχνικό υπόβαθρο μπορούν να καθορίσουν τους στόχους τους φτιάχνοντας τις δικές τους “συνταγές”. Στην έρευνα των Ur et al. (2014) [23] παρουσιάζεται ένα πείραμα που έγινε για τη χρησιμότητα του trigger-action προγραμματισμού. Σε αυτή συμπεραίνεται ότι οι χρήστες με προηγούμενη εμπειρία προγραμματισμού δεν είχαν σημαντικό πλεονέκτημα απέναντι σε άλλους χρήστες στη δημιουργία “συνταγών”. Από την άλλη η εξοικείωση με το σύστημα που

χρησιμοποίησαν παίζει καθοριστικό ρόλο, αφού τη δεύτερη ή την επόμενη “συνταγή” κατόρθωσαν να τη φτιάξουν πολύ πιο γρήγορα από την πρώτη.

Αρκετά websites έχουν υιοθετήσει την προσέγγιση του trigger-action προγραμματισμού και έχουν γίνει ιδιαίτερα δημοφιλή μιας και οι χρήστες τους έχουν γράψει συνολικά εκατοντάδες χιλιάδες “συνταγές”. Τα πιο γνωστά από αυτά είναι το IFTTT.com [24] (αναλύεται διεξοδικότερα στην επόμενη ενότητα) και το Zapier.com [25]

2.5.1 Μελέτη Περίπτωσης IFTTT [24]

Σε αυτή την ενότητα παρουσιάζεται το ενδιαφέρον παράδειγμα της διαδικτυακής εφαρμογής IFTTT, που λειτουργεί ως έξυπνος βοηθός για τον χρήστη που θέλει να αυτοματοποιήσει κάποια ενέργειά του στο διαδίκτυο. Συγκεκριμένα ο χρήστης δημιουργεί και στη συνέχεια χρησιμοποιεί απλούς υποθετικούς κανόνες (if-then-that), που όταν ενεργοποιούνται εκτελούν κάποια ενέργεια εκ μέρους του χρήστη σε κάποια διαδικτυακή υπηρεσία που χρησιμοποιεί. Οι κανόνες αυτοί αναφέρονται και ως “συνταγές” (recipes). Με αυτόν τον τρόπο συνδέονται μεταξύ τους διάφορες υπηρεσίες με στόχο πάντα την εξυπηρέτηση του χρήστη, ανάλογα με τα ενδιαφέροντα και τις προτιμήσεις του. Μερικές από τις υπηρεσίες που υποστηρίζονται είναι το email, ημερολόγιο, επαφές χρήστη, υπενθυμίσεις, μέσα κοινωνικής δικτύωσης (πχ. Facebook, Twitter, Instagram κ.ά), ειδήσεις (πχ. Fox News, The New York Times, ESPN, Wikipedia), καιρός κ.ά. Είναι δυνατή ακόμα η σύνδεση με ηλεκτρικές συσκευές και αυτοκίνητα που είναι συμβατά με υπηρεσίες που προσφέρονται μέσω της τεχνολογίας “Internet of Things”.

Οι δυνατότητες που έχει ένας χρήστης να φτιάξει τους δικούς του κανόνες είναι σχεδόν ατελείωτες. Για παράδειγμα, κάποιος χρήστης μπορεί να κάνει post το περιεχόμενο που επιθυμεί μία φορά σε όλα τα μέσα κοινωνικής δικτύωσης που χρησιμοποιεί ή άτομα που ασχολούνται με το μάρκετινγκ μπορούν να βλέπουν σε πραγματικό χρόνο τις αναφορές που γίνονται σε εταιρίες που τους ενδιαφέρουν μέσω RSS feeds. Άλλο παράδειγμα αποτελεί ο διαμοιρασμός φωτογραφιών που αποθηκεύονται σε κάποιο ειδικό μέσο (πχ. Flickr) προς όλα τα κοινωνικά δίκτυα που είναι συνδεδεμένος ο χρήστης (πχ. Facebook, Twitter, Instagram). Η διαδικασία που ακολουθεί ένας χρήστης για να δημιουργήσει μία “συνταγή” (recipe) είναι απλή:

Επιλέγει το κανάλι εισόδου (channel), από την πληθώρα που υπάρχουν στη διαθέσιμη λίστα.

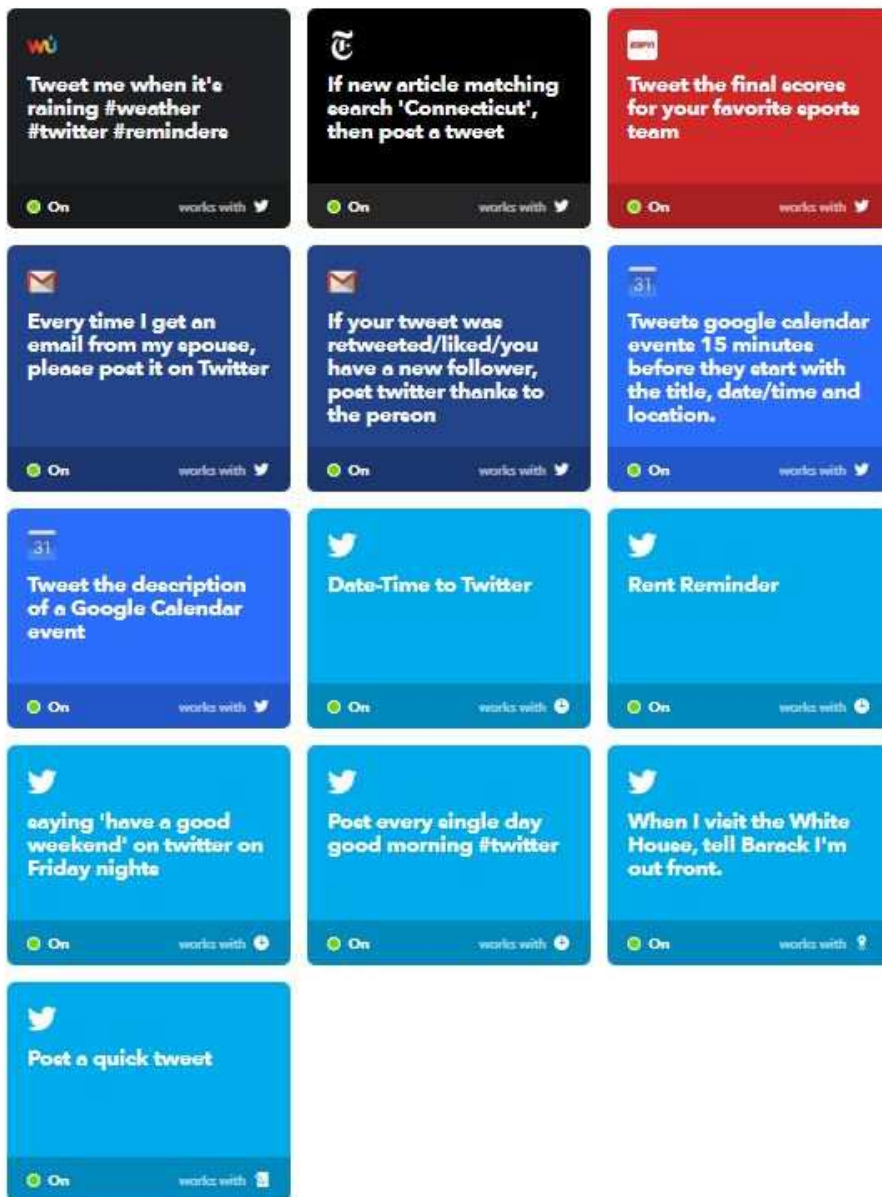
1. Εξουσιοδοτεί το IFTTT να χρησιμοποιεί το κανάλι εισόδου (channel) που επιλέχθηκε.
2. Επιλέγει το στοιχείο που ενεργοποιεί τον κανόνα (trigger) από τα διαθέσιμα που του προσφέρει το κανάλι εισόδου (channel) που επέλεξε.
3. Επιλέγει την εφαρμογή στην οποία θέλει να εκτελέσει μια ενέργεια και τη συγκεκριμένη ενέργεια (action) από τις διαθέσιμες που του προσφέρει η εφαρμογή.
4. Συμπληρώνει τις απαραίτητες πληροφορίες για να παραμετροποιήσει τα επιλεγμένα στοιχεία ενεργοποίησης (trigger) και ενέργειας (action).
5. Προαιρετικά, κοινοποιεί την συνταγή (recipe) έτσι ώστε και άλλοι χρήστες να μπορούν να την χρησιμοποιήσουν.

Μια αναπαράσταση “συνταγής” φαίνεται στην παρακάτω εικόνα. Η συγκεκριμένη έχει την ερμηνεία : “κάθε μέρα στις 6 το απόγευμα άναψε τα φώτα της κουζίνας”



Εικόνα 2.6 : Παράδειγμα συνταγής στο IFTTT

Στο πλαίσιο της μεταπτυχιακής διατριβής, όπου ασχοληθήκαμε με το μέσο κοινωνικής δικτύωσης Twitter, έγινε χρήση της εφαρμογής IFTTT και ενεργοποιήσαμε κάποιους κανόνες που συνδέουν διάφορες διαδικτυακές υπηρεσίες με τον λογαριασμό Twitter που διαθέτουμε. Στην παρακάτω εικόνα φαίνονται οι κανόνες που δημιουργήσαμε. Στη συνέχεια μέσω του λογαριασμού Twitter που χρησιμοποιήθηκε επαληθεύτηκαν οι ενέργειες που εκτέλεσε η εφαρμογή IFTTT, όπως το post συγκεκριμένων μηνυμάτων ανάλογα με τον κανόνα που ενεργοποιήθηκε.



Εικόνα 2.7 : Παραδείγματα κανόνων στο IFTTT για το Twitter

Κεφάλαιο 3

Θεωρητικό Πλαίσιο Έρευνας

3.1 Εισαγωγή στο Twitter

Το Twitter είναι μια υπηρεσία microblogging που επιτρέπει στους χρήστες του να στέλνουν και να διαβάζουν σύντομα μηνύματα (μέχρι 140 χαρακτήρες), τα οποία ονομάζονται tweets. Ο περιορισμός αυτός προήλθε από το γεγονός ότι η εφαρμογή αρχικά σκόπευε να στέλνει τα μηνύματα αυτά μέσω SMS. Τα μηνύματα μπορούν να αναγνωστούν και από μη συνδεδεμένους χρήστες, αλλά μόνο οι συνδεδεμένοι χρήστες μπορούν να δημοσιεύουν κείμενα. Μερικοί χρήσιμοι ορισμοί και λειτουργίες του Twitter είναι οι παρακάτω :

- **Tweet :** Ένα σύντομο κείμενο/δημοσίευση από ένα χρήστη του Twitter που έχει μέγιστο μέγεθος 140 χαρακτήρων. Τα Tweets περιέχουν ενημερώσεις σε δραστηριότητες των χρηστών, μοιράζουν χρήσιμες πληροφορίες, προωθούν καταστάσεις άλλων χρηστών, συνομιλούν με άλλους κλπ. Ο περιορισμός σε 140 χαρακτήρες επηρεάζει την έκφραση των χρηστών.
- **Hashtag :** Συμβολίζοντας τη λέξη που ακολουθεί το σύμβολο “#” (π.χ #CollectiveIntelligence), το Hashtag είναι μια σύμβαση της εφαρμογής για να καθορίζει θέματα ορισμένα από τους χρήστες. Με αυτό τον τρόπο είναι δυνατό να εντοπίζεται εύκολα ένα θέμα συζήτησης και ακόμα να κατηγοριοποιούνται συζητήσεις ανάλογα με το θέμα.
- **URLs :** Τα Tweets μπορεί να περιέχουν συνδέσμους (links) σε ιστοσελίδες, blogs κλπ. Οι χρήστες του Twitter χρησιμοποιούν συνοπτικές εκδόσεις των URLs αυτών, με τη χρήση εξωτερικών υπηρεσιών (πχ. nokia.ly/1ON3uQ5)

- **Reply :** Το Reply είναι μια λειτουργία που παρέχεται από την πλατφόρμα του Twitter για να επικοινωνεί κανείς άμεσα με τον δημιουργό ενός Tweet. Πατώντας απλά το κουμπί “Reply” ο χρήστης απαντά σε κάποιο Tweet. Για παράδειγμα, ο χρήστης user1 δημιουργεί το Tweet “το σημερινό μάθημα ήταν πολύ ενδιαφέρον!”, ενώ ο χρήστης user2 χρησιμοποιώντας το “Reply” απαντάει με το Tweet “@user1 ναι, πράγματι ήταν πολύ καλό!». Η λειτουργία του Reply αυτόματα εισάγει στο Tweet το όνομα χρήστη του εκκινήτη της συνομιλίας.
- **Retweet :** Το Retweet προωθεί ένα Tweet από κάποιον χρήστη προς τους Followers του, με τρόπο παρόμοιο ενός email forwarding. Ο χρήστης που χρησιμοποιεί αυτή τη λειτουργία αναφέρει την πηγή με τη μορφή “RT @username”. Για παράδειγμα το Tweet “RT @user1: εξαιρετικό το καινούριο προϊόν...” φανερώνει πως ο καινούριος χρήστης έκανε ένα Retweet στο παραπάνω Tweet από τον χρήστη user1.
- **Mention :** Το Mention (αναφορά) αναγνωρίζει έναν χρήστη από το συμβολικό χαρακτήρα ‘@’, χωρίς να χρησιμοποιηθεί η λειτουργία “Reply”. Για παράδειγμα ένα Tweet που περιέχει Mention είναι το “Σ’ ευχαριστώ @user1. Οι συμβουλές σου ήταν πολύτιμες.”
- **Like :** Με το Like ένας χρήστης δείχνει την προτίμησή του σε κάποιο συγκεκριμένο tweet και έτσι με αυτό τον τρόπο το κατηγοριοποιεί στα «αγαπημένα» του.

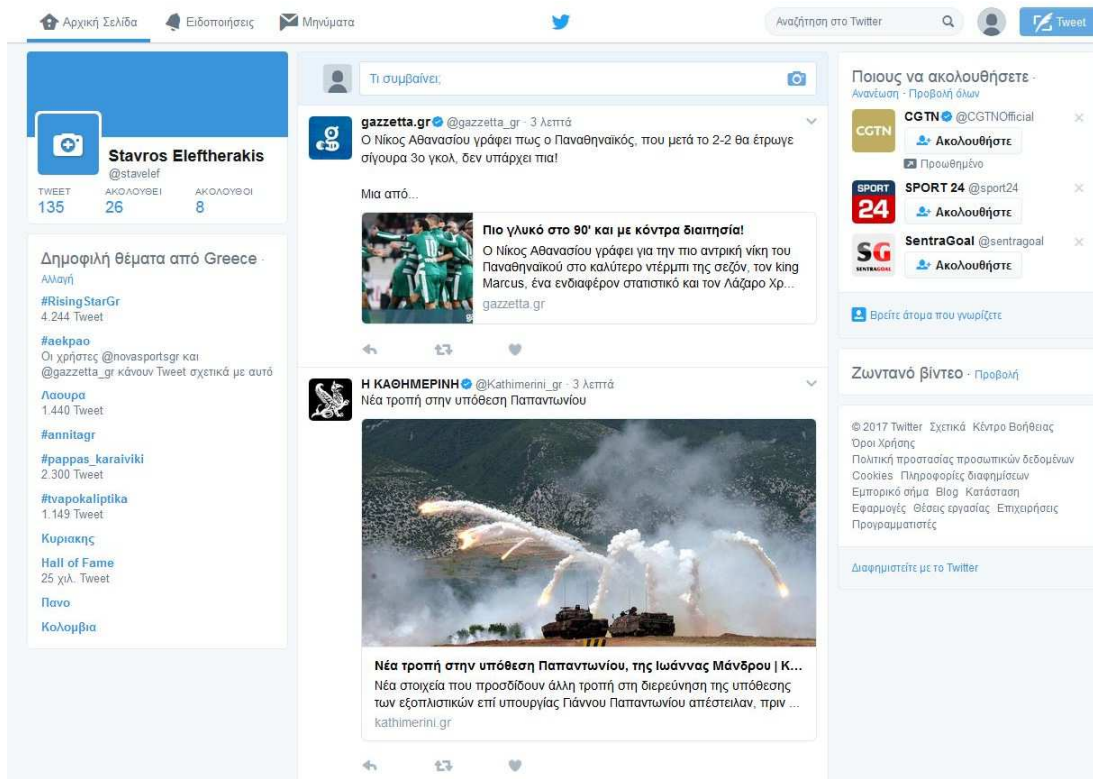
Το Twitter μέσω της λειτουργίας του “follow” επιτρέπει τη δημιουργία του κοινωνικού δικτύου. Το μοντέλο αυτό είναι διαφορετικό από το μοντέλο «φιλίας» που παρουσιάζεται στα περισσότερα κοινωνικά δίκτυα, επειδή ένας χρήστης μπορεί να «ακολουθεί» κάποιον άλλο, χωρίς να ισχύει το αντίστροφο. Για να γίνει περισσότερο κατανοητό αυτό το μοντέλο «φιλίας» παρακάτω ξεχωρίζουμε τις έννοιες “following” και “followers”.

- **Following :** Όταν «ακολουθήσεις» (κάνεις follow) έναν λογαριασμό στο Twitter, τότε η αρχική σου σελίδα (προσωπικός τοίχος) θα περιλαμβάνει τα tweets αυτού του λογαριασμού και ταυτόχρονα ο δικός σου λογαριασμός θα εμφανίζεται δημόσια στη λίστα με τους followers του λογαριασμού που έχει επιλέξει. Συνήθως «ακολουθούμε» έναν λογαριασμό εάν ενδιαφερόμαστε για τα θέματα που προβάλλει ή αν είμαστε fan του κατόχου αυτού του λογαριασμού. Με αυτό τον τρόπο μπορούμε να βλέπουμε όλα

τα νέα tweets από τους λογαριασμούς που μας ενδιαφέρουν χωρίς να χρειαστεί να μπαίνουμε ξεχωριστά στο προφίλ του κάθε χρήστη.

- **Followers** : Για τον ίδιο λόγο άλλοι χρήστες μπορούν να ακολουθήσουν τον λογαριασμό σου, επειδή θέλουν να βλέπουν τα tweets που έκανες. Όταν έχεις «ακόλουθους» (followers) στο Twitter, γνωρίζεις ότι υπάρχει ένα ακροατήριο που θα διαβάσει τα tweets σου. Πολλές εταιρίες, συγγραφείς και bloggers χρησιμοποιούν το Twitter τόσο για να μάθουν πληροφορίες όσο και για να διαφημιστούν. Γι'αυτό το λόγο οι περισσότεροι χρήστες του Twitter προτιμούν να έχουν πολλούς «ακόλουθους» (followers) .

Το γραφικό περιβάλλον του Twitter είναι σχετικά απλό και ένα παράδειγμα της αρχικής του σελίδας δίνεται στην παρακάτω εικόνα. Σε αυτή μπορούμε να ξεχωρίσουμε ορισμένα στοιχεία που αναφέρθηκαν παραπάνω (στην αριστερή στήλη), όπως το όνομα χρήστη (@stavelef), ο αριθμός των tweets που έχει κάνει ο χρήστης (135), ο αριθμός των followers του χρήστη (8) και ο αριθμός άλλων χρηστών που ακολουθεί ο χρήστης @stavelef (26). Στο κεντρικό μέρος της σελίδας (Twitter Wall) μπορούμε να δούμε τα τελευταία posts που έγιναν είτε από τον χρήστη @stavelef είτε από κάποιο άλλο χρήστη που ακολουθεί ο @stavelef. Επίσης, στην αριστερή στήλη, εμφανίζονται τα 10 δημοφιλέστερα θέματα στην τοποθεσία που έχει δηλώσει ο χρήστης (πχ. Ελλάδα) τη συγκεκριμένη χρονική στιγμή.



Εικόνα 3.1 : Το γραφικό περιβάλλον του Twitter

3.2 Twitter APIs

Το Twitter παρέχει κάποια APIs (Application Programming Interface) για να μπορούν οι προγραμματιστές να δημιουργούν εφαρμογές που να διαβάζουν ή να γράφουν δεδομένα σε αυτό. Οι τέσσερις βασικοί τύποι δεδομένων που ανταλλάσσονται μέσα στο Twitter είναι τα Tweets, οι χρήστες (Users), οι οντότητες (entities) και οι τοποθεσίες (Places).

Τα tweets αποτελούν το σημαντικότερο στοιχείο σε όλο το Twitter και αναφέρονται επίσης και ως “status update”. Τα tweets γενικά μπορούν να ενσωματωθούν (embedded), να αποτελούν απάντηση σε κάποιο άλλο tweet (replied to), να γίνουν liked, unliked και να διαγραφούν (deleted). Στον παρακάτω πίνακα φαίνεται μέρος της δομής δεδομένων ενός αντικειμένου tweet (οι βασικότερες παράμετροι που θα χρησιμοποιήσουμε κατά την ανάπτυξη του συστήματος έξυπνου βοηθού). Η αναλυτική δομή δεδομένων ενός tweet παρουσιάζεται στο [26].

Παράμετρος	Τύπος	Περιγραφή
created_at	String	Ο χρόνος (UTC ημερομηνία και ώρα) τη στιγμή που δημιουργήθηκε το Tweet. Παράδειγμα: "created_at": "Wed Aug 27 13:08:45 +0000 2008"
id	Int64	Η ακέραια αναπαράσταση του μοναδικού αναγνωριστικού για το συγκεκριμένο Tweet. Παράδειγμα: "id": 114749583439036416
id_str	String	Η αναπαράσταση συμβολοσειράς του μοναδικού αναγνωριστικού για το συγκεκριμένο Tweet. Παράδειγμα: "id_str": "114749583439036416"
lang	String	Η γλώσσα που αναγνωρίζεται μηχανικά ότι χρησιμοποιείται σε ένα κείμενο tweet. Παράδειγμα: "lang": "en"
text	String	Το πραγματικό κείμενο (σε UTF-8 format) που αποτελεί το tweet. Παράδειγμα: "text": "Tweet Button, Follow Button, and Web Intents javascript now support SSL http://t.co/9fbA0oYy ^TS"
user	Users	Ο χρήστης που έχει στείλει το συγκεκριμένο tweet. Στις πληροφορίες του χρήστη περιέχονται δεδομένα που αφορούν το όνομα λογαριασμού του, την ημερομηνία που δημιούργησε το λογαριασμό του, την τοποθεσία του κ.ά. Αναλυτική δομή δεδομένων για τον χρήστη παρουσιάζεται στο [27].

Πίνακας 3.2 : Μέρος της δομής δεδομένων της οντότητας Tweet.

3.2.1 REST API

Το REST (Representational State Transfer) αποτελεί την βασική αρχιτεκτονική αρχή του διαδικτύου. Κύριο χαρακτηριστικό του είναι ότι οι εξυπηρετούμενοι (clients) και οι εξυπηρετητές (servers) μπορούν να αλληλεπιδρούν μεταξύ τους με πολύπλοκους τρόπους, χωρίς ο εξυπηρετούμενος να γνωρίζει εκ των προτέρων για τον εξυπηρετητή και τους πόρους που αυτός διαθέτει. Ο βασικός περιορισμός είναι ότι εξυπηρετητής και εξυπηρετούμενος πρέπει να συμφωνούν στο μέσο που θα χρησιμοποιηθεί, το οποίο στην περίπτωση του διαδικτύου είναι η γλώσσα HTML (HyperText Markup Language). Ένα API που υπακούει στις αρχές του REST δεν απαιτεί από τον εξυπηρετούμενο να γνωρίζει κάτι σχετικά με την δομή του API. Αντί αυτού, ο εξυπηρετητής πρέπει να παρέχει οποιαδήποτε πληροφορία χρειάζεται ο εξυπηρετούμενος για να αλληλεπιδράσει με την υπηρεσία.

Τα REST APIs του Twitter [28] παρέχουν σε προγραμματιστές πρόσβαση για ανάγνωση και εγγραφή δεδομένων, από και προς το Twitter. Ο προγραμματιστής μπορεί να δημοσιεύσει ένα καινούργιο Tweet, να διαβάσει το προφίλ ενός χρήστη, δεδομένα σχετικά με τους ακόλουθους αυτού κλπ. Το REST API κάνει ταυτοποίηση των εφαρμογών και των χρηστών του Twitter μέσω του OAuth, το οποίο επιτρέπει ασφαλή αποστολή εγκεκριμένων αιτημάτων πρόσβασης στο API με τη χρήση διαπιστευτηρίων μοναδικών για κάθε χρήστη. Εφόσον τηρούνται οι προϋποθέσεις το API παρέχει απαντήσεις σε μορφή JSON προς το περιβάλλον που τις ζητά. Θεωρητικά, όλες τις ενέργειες που μπορεί να κάνει κάποιος χρήστης στο Twitter μπορεί να τις πραγματοποιήσει μέσω μιας εξωτερικής εφαρμογής με χρήση του REST API. Ο πλήρης κατάλογος των εντολών αυτών παρουσιάζεται στο [29].

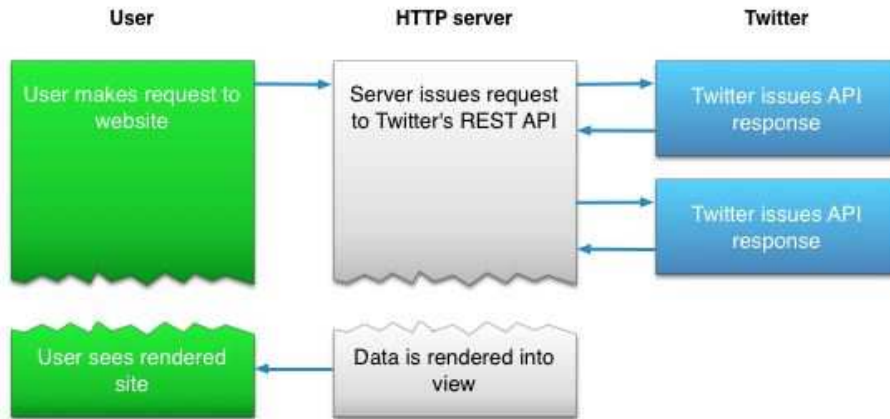
Το Twitter θέτει κάποιους περιορισμούς σχετικά με τη λήψη δεδομένων εντός σύντομων χρονικών διαστημάτων. Οι περιορισμοί αυτοί εφαρμόζονται ανά λογαριασμό χρήστη ή για την ακρίβεια ανά διαπιστευτηρίου πρόσβασης (access token) του χρήστη και έχουν χρονικό περιθώριο 15 λεπτών. Κάθε εντολή πρόσβασης στο REST API έχει το δικό του όριο. Πάντως οι περισσότερες αιτήσεις πρόσβασης (GET) δεν μπορούν να ξεπερνούν τις 15 κλήσεις σε 15 λεπτά (για παράδειγμα η ανάγνωση των τελευταίων tweets που εμφανίζονται στον τοίχο ενός χρήστη) ή σε άλλες περιπτώσεις τις 180 κλήσεις σε 15 λεπτά (για παράδειγμα η αναζήτηση tweets με βάση κάποιο κριτήριο). Περισσότερες πληροφορίες για τους περιορισμούς αυτούς στο [30].

3.2.2 Streaming API [31]

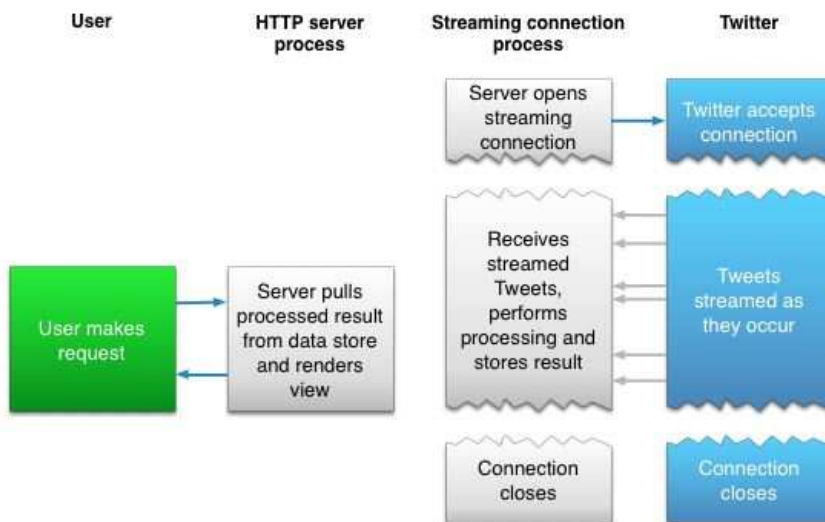
Γενικά τα Streaming APIs παρέχουν στους προγραμματιστές απευθείας πρόσβαση με χαμηλή καθυστέρηση (latency) στην παγκόσμια ροή δεδομένων που προέρχεται από το Twitter. Ένας streaming client μπορεί να καταλάβει αλλαγές σε tweets και σε άλλα συμβάντα που έχουν συμβεί σε πραγματικό χρόνο χωρίς να πρέπει να συνδεθεί ξανά με κάποιο REST τερματικό σημείο (endpoint).

Η βασικότερη διαφορά του Streaming API από το REST API είναι ότι απαιτεί να υπάρχει μια σταθερά ανοικτή HTTP σύνδεση. Στις εφαρμογές που χρησιμοποιούν το Streaming API διαχωρίζεται το πρόγραμμα (process) που συντηρεί τη σύνδεση με το Twitter με το πρόγραμμα που ασχολείται με τις HTTP αιτήσεις (βλέπε εικόνα 3.3). Αντίθετα σε αυτές που χρησιμοποιούν το REST API το ίδιο πρόγραμμα λαμβάνει τις αιτήσεις του χρήστη, προωθεί τις αιτήσεις αυτές στο Twitter API και στέλνει πίσω τις απαντήσεις που λαμβάνει από το Twitter

API στο χρήστη (βλέπε εικόνα 3.4). Πάντως, αν σκοπός του προγραμματιστή είναι να κάνει απλές αναζητήσεις ή να κάνει δημοσιεύσεις καινούριων tweets προτείνεται η χρήση του REST API.



Εικόνα 3.3 : Αρχιτεκτονική REST API



Εικόνα 3.4 : Αρχιτεκτονική Streaming API

3.2.3 Webhook API [32]

Γενικά η διαφορά ενός Webhook API από ένα τυπικό API είναι ότι δεν απαιτείται κάποιο ειδικό αίτημα, όπως στην κλασική περίπτωση αιτήματος-απάντησης, αλλά υπάρχει αποστολή δεδομένων όταν αυτά είναι διαθέσιμα ή όταν υπάρχει κάποια αλλαγή κατάστασης. Αυτό προγραμματιστικά γίνεται συνήθως με την ενεργοποίηση κάποιων callback συναρτήσεων.

Τα Webhook APIs του Twitter παρέχουν στους προγραμματιστές πρόσβαση σε δεδομένα του λογαριασμού σε πραγματικό χρόνο. Αφού ρυθμιστεί μια URL ως διεύθυνση της εφαρμογής του και γίνει ο απαραίτητος έλεγχος πρόσβασης, το Twitter ξεκινά να στέλνει δεδομένα (με τη μέθοδο POST) κατευθείαν στο URL που έχει υποδείξει ο χρήστης. Προς το παρόν υπάρχει ένα μόνο διαθέσιμο Webhook API, το Account Activity API, το οποίο ακόμα βρίσκεται σε δοκιμαστική λειτουργία (beta release).

3.2.4 Ads API [33]

Το Ads API επιτρέπει σε τρίτες εταιρίες χρησιμοποιήσουν τη διαφημιστική πλατφόρμα του Twitter για να αναπτύξουν τις δικές τους διαφημιστικές λύσεις και εφαρμογές. Μάλιστα επιλεγμένες εταιρίες έχουν τη δυνατότητα να δημιουργήσουν εργαλεία για να διαχειριστούν και να «τρέξουν» διαφημιστικές καμπάνιες μέσω του Twitter.

3.3 Twitter Libraries / Codebird-php

Υπάρχουν πολλές βιβλιοθήκες λογισμικού ανοικτού κώδικα που δημιουργήθηκαν για την επικοινωνία με την πλατφόρμα του Twitter, γραμμένες σε διάφορες γλώσσες προγραμματισμού όπως ASP, C++, Clojure, ColdFusion, .NET, Go, Java, Javascript/node.js, Objective-C, Perl, PHP, Python, Ruby. [34]

Επειδή μέρος της μεταπτυχιακής διατριβής αποτελεί η ανάπτυξη μιας διαδικτυακής εφαρμογής μας ενδιαφέρουν γλώσσες προγραμματισμού που προσφέρονται για το σκοπό αυτό και συγκεκριμένα γραμμένες στη γλώσσα PHP. Στον παρακάτω πίνακα παρουσιάζονται μερικές βιβλιοθήκες PHP που προτείνονται από τη σελίδα του Twitter.

Βιβλιοθήκη	Παρατηρήσεις
tmhOAuth	PHP βιβλιοθήκη για OAuth. Έχει πάνω από 3 χρόνια να γίνει κάποιο update.
twitteroauth	Δημοφιλής PHP βιβλιοθήκη για χρήση του OAuth Rest API του Twitter
140dev Twitter Framework	Απλοποιημένο interface για το Streaming API του Twitter. Κατάλληλο για συλλογή tweets σε βάσεις δεδομένων και παρουσίασή τους σε ιστοσελίδες
Twitter API: Engagement Programming	Περιέχει παραδείγματα PHP κώδικα

codebird-php	Παρέχει εύκολη πρόσβαση στα REST API, Collections API, Streaming API, TON (Object Nest) API και Ads API όλα σε μια PHP βιβλιοθήκη
CodeIgniter-Twitter-Search-Library	Προσφέρετε για την αναζήτηση tweets με χρήση keywords χρησιμοποιώντας τα Search API και Streaming API του Twitter
freebird-php	Interface που υποστηρίζει μόνο τη μεθοδολογία Application Only Authentication
PHP OAuth API	PHP class που υλοποιεί OAuth authorization και καλεί APIs με OAuth tokens.
Twitter-API-PHP	Πολύ απλός wrapper για το v1.1 REST API του Twitter που υποστηρίζει μόνο λίγες μεθόδους.
TwitterOAuth	Απλή PHP βιβλιοθήκη για το v1.1 REST API του Twitter.

Πίνακας 3.5 : Βιβλιοθήκες PHP για το Twitter

Από τις παραπάνω επιλέχτηκε να χρησιμοποιηθεί η βιβλιοθήκη **codebird-php** [35] για τους εξής λόγους :

1. Διαθέτει αρκετά καλό documentation
2. Βρέθηκαν κάποιες πηγές στο διαδίκτυο με θετικά σχόλια για τη χρήση της βιβλιοθήκης αυτής
3. Έχει εύκολο τρόπο για την εξουσιοδότηση του χρήστη (authentication)
4. Εφαρμόζει μια κατανοητή αντιστοιχία (mapping) μεταξύ των συναρτήσεων της βιβλιοθήκης και των συναρτήσεων των APIs του Twitter που καλούν αυτές. Έτσι είναι πιο εύκολο για έναν προγραμματιστή να χρησιμοποιήσει τις συναρτήσεις της βιβλιοθήκης διαβάζοντας μόνο το documentation των Twitter APIs.

Στα παρακάτω παραδείγματα φαίνεται ο απλός τρόπος με τον οποίο πραγματοποιείται authentication για τον χρήστη και γίνεται το κάλεσμα συναρτήσεων για την επικοινωνία με το Twitter API.

Παράδειγμα Authentication

```
require_once ('codebird.php');
\Codebird\Codebird::setConsumerKey('YOURKEY', 'YOURSECRET');
$cb = \Codebird\Codebird::getInstance();
$cb->setToken($_SESSION['oauth_token'], $_SESSION['oauth_token_secret']);
```

Παράδειγμα καλέσματος συνάρτησης από το API

```
//διάβασμα των tweets του προσωπικού τοίχου (wall) του χρήστη
$response = (array) $cb->statuses_homeTimeline();
print_r($response);

//δημοσίευση tweet
$response = $cb->statuses_update('status=Whohoo, I just Tweeted!');
```

3.4 Επιχειρηματολογία

Η επιχειρηματολογία αποτελεί μια γνωστική διαδικασία που παίζει σημαντικό ρόλο στην καθημερινή ζωή των ανθρώπων, αφού μέσα από αυτή μπορούν να εκφραστούν, να προτείνουν λύσεις, να υπερασπιστούν μια ιδέα ή να διαφωνήσουν με κάποια άλλη ιδέα. Στην έρευνα που πραγματοποίησε ο Dung et al. (1995) [36] μελετάει τους μηχανισμούς της επιχειρηματολογίας και προτείνει ένα θεωρητικό πλαίσιο όπου θα μπορούσαν να υλοποιηθούν σε υπολογιστικά συστήματα. Στον τομέα της Τεχνητής Νοημοσύνης έχουν γίνει πολλές μελέτες για την ανάλυση της δομής των επιχειρημάτων και την ανάπτυξη συστημάτων που βασίζονται στη θεωρία που ανέπτυξε ο Dung. Η γενική ιδέα που υποστηρίζεται στη θεωρία αυτή για το συλλογισμό των επιχειρημάτων είναι πως ένα επιχείρημα θεωρείται αξιόπιστο αν μπορεί να υπερασπιστεί επιτυχώς την πρότασή του απέναντι σε άλλα επιχειρήματα που του αντιτίθενται. Από μια άποψη ο κλασικός συλλογισμός βασίζεται στην ιδέα πως ένα συμπέρασμα που υποστηρίζεται από κάποια επιχείρημα είναι έγκυρο αν δεν υπάρχει κάποιο αντεπιχείρημα. Από την άλλη, ο συλλογισμός μπορεί να σχετίζεται με μια συγκεκριμένη γνώμη ή προσωπική άποψη για ένα συγκεκριμένο θέμα. Και οι δύο περιπτώσεις συλλογισμού αποτελούν μορφές επιχειρηματολογίας.

Μπορούμε να διαχωρίσουμε την επιχειρηματολογία σε δύο τύπους:

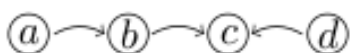
- Μονολογική (Monological) για μια συγκεκριμένη ανάλυση ή παρουσίαση κάποιας πληροφορίας από ένα άτομο, π.χ. κάποιο πολιτικό λόγο πριν τις εκλογές

- Διαλογική (Dialogical) για ανταλλαγή πληροφοριών μεταξύ διαφορετικών ατόμων, πχ. ένα debate μεταξύ των αρχηγών πολιτικών κομμάτων πριν τις εκλογές

Με βάση μια διαφορετική προσέγγιση μπορούμε να ξεχωρίσουμε την επιχειρηματολογία σε δύο τύπους, την αφηρημένη (abstract argumentation) και τη βασισμένη στη λογική (logic-based argumentation).

3.4.1 Αφηρημένη επιχειρηματολογία

Η αφηρημένη επιχειρηματολογία (abstract argumentation) αποτελεί το θεωρητικό μέρος της προσέγγισης του Dung, όπου δίνονται ορισμοί και κανόνες που χρησιμοποιούνται για την αποδοχή των επιχειρημάτων. Ένα σύστημα αφηρημένης επιχειρηματολογίας είναι μια συλλογή από επιχειρήματα που αντιπροσωπεύουν δεδομένα ή μια πρόταση. Οι διαμάχες μεταξύ των επιχειρημάτων δηλώνονται με τη χρήση του συμβόλου “->” μεταξύ τους. Για παράδειγμα η δήλωση “x -> y” σημαίνει πως το επιχείρημα x επιτίθεται στο επιχείρημα y. Το σύστημα αυτό μπορεί να αναπαρασταθεί με ένα κατευθυνόμενο γράφο, όπου οι κόμβοι δηλώνουν τα επιχειρήματα και τα βέλη δηλώνουν τη σχέση επίθεσης μεταξύ των επιχειρημάτων, όπως φαίνεται στην παρακάτω εικόνα.



Εικόνα 3.6 : Παράδειγμα αφηρημένης επιχειρηματολογίας

Ένα πλαίσιο αφηρημένης επιχειρηματολογίας προσδιορίζεται ως ένα ζευγάρι $S = \{ A, R \}$ όπου :

- A είναι το σύνολο των επιχειρημάτων
- R είναι το σύνολο των επιθέσεων μεταξύ των επιχειρημάτων

Στο παράδειγμα της εικόνας 3.6 το σύνολο $A = \{ a, b, c, d \}$ περιέχει τα 4 επιχειρήματα (a,b,c,d), ενώ το σύνολο $R = \{ (a, b), (b, c), (d, c) \}$ περιέχει 3 επιθέσεις (το a επιτίθεται στο b, το b επιτίθεται στο c, το d επιτίθεται στο c).

Οι βασικότεροι κανόνες-ορισμοί που έχουν δοθεί στο θεωρητικό αυτό πλαίσιο είναι οι εξής :

- ένα επιχείρημα $a \in A$ είναι acceptable σε σχέση με ένα σύνολο $E \subseteq A$, αν και μόνο αν το σύνολο E υπερασπίζεται το επιχείρημα a , δηλαδή $\forall b \in A$ τέτοιο ώστε $(b,a) \in R$ να ισχύει $\exists c \in E$ τέτοιο ώστε $(c,b) \in R$
- ένα σύνολο επιχειρημάτων E είναι conflict-free αν δεν υπάρχουν επιθέσεις μεταξύ των επιχειρημάτων του, δηλαδή $\forall a,b \in E$ να ισχύει $(a,b) \notin R$
- ένα σύνολο επιχειρημάτων E είναι admissible αν και μόνο αν είναι conflict-free και όλα τα επιχειρήματα που περιέχει είναι acceptable σε σχέση με το E .

Η αφηρημένη επιχειρηματολογία αποτελεί μια απλή μέθοδο αναπαράστασης των επιθέσεων μεταξύ των επιχειρημάτων και περιλαμβάνει ορισμούς για μια συνολική εκτίμηση της κατάστασης. Ωστόσο, έχει περιορισμένη εκφραστικότητα και απουσιάζει από αυτή μια μέθοδος για να εξαγονται συμπεράσματα από κάποια βάση γνώσης. Υπάρχουν ορισμένες επεκτάσεις της αφηρημένης επιχειρηματολογίας που καλύπτουν τις αδυναμίες αυτές, όπως η βασισμένη στη λογική (logic-based), η βασισμένη στις τιμές (value-based) και η βασισμένη στις υποθέσεις (assumption-based) επιχειρηματολογία.

3.4.2 Λογική επιχειρηματολογία

Η λογική επιχειρηματολογία (logic-based argumentation) ασχολείται με επιχειρήματα βασισμένα σε κάποιο σύστημα λογικής. Στην κλασική λογική επιχείρημα είναι μια σειρά από συλλογισμούς που οδηγούν σε ένα συμπέρασμα. Η λογική επιχειρηματολογία είναι περισσότερο εκφραστική και υποστηρίζει λεπτομερή παρουσίαση της γνώσης και μεθόδους για την εξαγωγή συμπερασμάτων. Η κατηγορία αυτή συναντάται εξίσου και με τον όρο δομημένη επιχειρηματολογία (structured argumentation).

Στη δομημένη επιχειρηματολογία υποθέτουμε μια επίσημη γλώσσα για την αναπαράσταση της γνώσης και καθορίζουμε πως τα επιχειρήματα και τα αντεπιχειρήματα μπορούν να δημιουργηθούν από αυτή τη γνώση. (Besnard et al., 2014) [37]

Στην περίπτωση αυτή ένα επιχείρημα δεν είναι μια αφηρημένη έννοια, αλλά ένα ζεύγος όπου το πρώτο μέλος του αποτελεί το ελάχιστο συνεπές σύνολο τύπων που είναι ικανό να αποδείξει

έναν τύπο από το δεύτερο μέρος του επιχειρήματος. Το επιχείρημα αυτό αναπαριστάται ως ζεύγος (Φ, a) τέτοιο ώστε :

- το Φ να είναι συνεπές ($\Phi \not\vdash \perp$)
- το Φ να συμπεραίνει το a ($\Phi \vdash a$)
- δεν υπάρχει κανένα $\Phi' \subset \Phi$ τέτοιο ώστε $\Phi' \vdash a$

Λόγω του γεγονότος ότι η λογική επιχειρηματολογία είναι περισσότερο πρακτική, προσφέρεται να δώσει λύσεις σε πραγματικά προβλήματα. Έτσι έχουν αναπτυχθεί πολλές εφαρμογές για συστήματα υποστήριξης αποφάσεων (decision support systems) που βασίζονται στη λογική επιχειρηματολογία και υλοποιούνται με χρήση γλωσσών λογικού προγραμματισμού, όπως για παράδειγμα η PROLOG. Μερικά παραδείγματα αφορούν συστήματα ηλεκτρονικής δημοκρατίας, συστήματα υποστήριξης αποφάσεων για ιατρικά θέματα, εφαρμογές διαμοιρασμού ιδεών, συστήματα για διαπραγματεύσεις μεταξύ πολλαπλών πρακτόρων λογισμικού, εφαρμογές για το semantic web κ.ά.

3.4.3 Flash Web Service

Στην ενότητα αυτή παρουσιάζεται ένας μηχανισμός γνωστικού συλλογισμού (cognitive reasoning) που μπορεί να χρησιμοποιηθεί για την ανάπτυξη εφαρμογών έξυπνων βοηθών (Michael, 2016) [38]. Ο μηχανισμός αυτός ανήκει στην κατηγορία συστημάτων που βασίζονται στην δομημένη (structured) ή λογική (logic-based) επιχειρηματολογία και ονομάζεται FLASH. Το FLASH βασίζεται στην αναπαράσταση κοινής λογικής από μια σειρά κανόνων. Οι κανόνες αυτοί μπορεί να συσχετίζονται μεταξύ τους με βάση κάποια προτεραιότητα.

Ο αλγόριθμος του FLASH εφαρμόζει επαναληπτικά κάποιους κανόνες σε αντιλήψεις (percepts) που λαμβάνονται εξωτερικά από κάποια πηγή πληροφοριών για να καταλήξει σε ορισμένα συμπεράσματα. Βασίζεται στη θεωρία του argumentation framework [39], με τη διαφορά ότι δεν είναι απαραίτητο για μια υπόθεση (argument) να είναι έγκυρη να υπερασπίζεται όλες τις άλλες ενάντιες υποθέσεις (attacks). Σημαντικό ρόλο στο μηχανισμό αυτό παίζει η επέκταση της βάσης γνώσης μέσω της προσθήκης καινούριων κανόνων που μπορεί να αντιτίθενται στους προϋπάρχοντες κανόνες και έτσι να εξαχθούν τελικά διαφορετικά συμπεράσματα. Ο μηχανισμός FLASH μπορεί να ταξινομηθεί ως ένα fast thinking system καθώς χρησιμοποιεί

ορισμένα μόνο από τα χαρακτηριστικά που απαιτούνται για την ανάπτυξη expert systems (slow thinking systems). Αυτό γίνεται σκοπίμως με στόχο να μπορεί να χρησιμοποιηθεί το σύστημα από έναν οποιοδήποτε μέσο χρήστη στον τομέα που τον ενδιαφέρει.

Παρακάτω παρουσιάζεται ένα παράδειγμα χρήσης του FLASH web service σε μια δοκιμαστική βάση γνώσης σχετική με "bird watching". Οι κανόνες που συνθέτουν τη βάση γνώσης φαίνονται στο πρώτο πεδίο (Common Knowledge-Rules), ενώ οι αντιλήψεις-percepts στο δεύτερο πεδίο (Query-Facts).

```
Common Knowledge
Insert Rules:
rule1 :: penguin implies -flying.
rule2 :: bird implies flying.
rule3 :: penguin implies bird.
rule4 :: feathers implies bird.
rule5 :: antarctica, bird, funny implies penguin.
rule6 :: flying implies wings.
rule7 :: -flying implies -feathers.
rule1>>rule2.

Query
Insert Facts:
antarctica, funny, feathers
```

Πίνακας 3.7 : Παράδειγμα χρήσης FLASH web service

Από τις δοθείσες αντιλήψεις (antarctica, funny, feathers) μπορούν να εξαχθούν με τη σειρά τα εξής συμπεράσματα σύμφωνα με τη βάση γνώσης (κανόνες). Σε κάθε βήμα μεγαλώνει η λίστα των συμπερασμάτων που βρίσκονται μέσα στις αγκύλες.

[antarctica, funny, feathers] $\xrightarrow{\text{rule4}}$ [antarctica,funny,feathers,bird]

[antarctica, funny, feathers, bird] $\xrightarrow{\text{rule5}}$ [antarctica, funny, feathers, bird, penguin]

[antarctica,funny,feathers,bird,penguin] $\xrightarrow{\text{rule2}}$ [antarctica,funny,feathers,bird,penguin,flying]

[antarctica,funny,feathers,bird,penguin] rule1 → [antarctica,funny,feathers,bird,penguin,-flying]

Στα τελευταία δυο βήματα παρατηρούμε πως υπάρχει μια αντίθεση. Από τον κανόνα rule2 εξάγουμε το συμπέρασμα flying, ενώ από τον κανόνα rule1 το αντίθετο -flying. Στην περίπτωση αυτή τη λύση δίνει η προτεραιότητα που έχει οριστεί ανάμεσα στους δύο κανόνες (rule1>>rule2.), οπότε η τελική λίστα με τα συμπεράσματα θα περιλαμβάνει το -flying.

Ο αλγόριθμος του FLASH web service πραγματοποιεί τους συλλογισμούς που αναφέρθηκαν στα παραπάνω βήματα και επιστρέφει την τελική λίστα των συμπερασμάτων όπως παρακάτω:

conclusion: [antarctica, funny, feathers, bird, penguin, -flying]

Κεφάλαιο 4

Twitter Assistant

4.1 Περιγραφή Γενικής Ιδέας

4.1.1 Εισαγωγή

Τα τελευταία χρόνια παρατηρείται μια ολοένα και αυξανόμενη τάση των ανθρώπων να χρησιμοποιούν τα μέσα κοινωνικής δικτύωσης για να επικοινωνούν ή να ενημερώνονται για θέματα που τους ενδιαφέρουν. Η ενασχόληση με τα μέσα αυτά συχνά απαιτεί πολύ χρόνο. Σχετικές μελέτες αναφέρουν ότι ένας μέσος άνθρωπος μπορεί να αφιερώνει σχεδόν 2 ώρες καθημερινά στο σύνολο των μέσων κοινωνικής δικτύωσης που χρησιμοποιεί (πχ YouTube, Facebook, Twitter, Instagram).

Από την άλλη η εξέλιξη στον τομέα της Τεχνητής Νοημοσύνης επιτρέπει την ανάπτυξη έξυπνων βοηθών και συστημάτων συστάσεων (recommendation systems) βασισμένα στη γνώση. Με τη χρήση τέτοιων ψηφιακών βοηθών είναι δυνατόν ένας χρήστης του διαδικτύου να χρησιμοποιήσει ποιοτικότερα (περισσότερο στοχευμένα) τα μέσα κοινωνικής δικτύωσης και έτσι να αφιερώσει λιγότερο χρόνο για να επιτύχει τον προσωπικό του στόχο.

Η γενική ιδέα της μεταπτυχιακής διατριβής είναι να αναπτυχθεί ένα σύστημα έξυπνου βοηθού για το γνωστό μέσο κοινωνικής δικτύωσης Twitter.

Ένα διάσημο πρόσωπο (για παράδειγμα κάποιος πολιτικός, τραγουδιστής, αθλητής, ηθοποιός κλπ) διαθέτει συνήθως ένα μεγάλο κοινό (followers) που ενδιαφέρεται για τα νέα του ή τις απόψεις του, όπως αυτά προβάλλονται μέσα από το Twitter. Πιθανώς είναι δύσκολο να παρακολουθεί όλες τις συζητήσεις, τα σχόλια ή τυχόν ερωτήσεις που γίνονται προς αυτόν. Επίσης το ίδιο μπορεί να συμβεί με δημοφιλείς εταιρίες, οι οποίες συνήθως απασχολούν άτομα ως υπαλλήλους για να διαχειρίζονται τους λογαριασμούς που διαθέτουν σε διάφορα μέσα κοινωνικής δικτύωσης συμπεριλαμβανομένου και του Twitter. Οι παραπάνω περιπτώσεις

αποτελούν χαρακτηριστικά παραδείγματα στα οποία ένας έξυπνος βοηθός για το Twitter θα έδινε λύσεις σώζοντας πολύτιμο χρόνο του χρήστη που πρόκειται να τον χρησιμοποιήσει. Ακόμα ένα τέτοιο σύστημα θα μπορούσε να χρησιμοποιηθεί για εμπορικούς σκοπούς από εταιρίες που ενδιαφέρονται για στοχευμένες διαφημίσεις μέσα από την πλατφόρμα του Twitter.

Ένα έξυπνος βοηθός για το Twitter φυσικά θα μπορούσε να φανεί χρήσιμος και για έναν οποιοδήποτε απλό χρήστη του Twitter. Μέσα από μια τέτοια εφαρμογή είναι δυνατόν να συλλεχθούν πληροφορίες για τον χρήστη και να αυτοματοποιηθούν κάποιες ενέργειες που διαφορετικά θα έπρεπε να εκτελέσει χειροκίνητα ο χρήστης μέσα από την πλατφόρμα του Twitter. Οι παρακάτω υποενότητες αφορούν τη διερεύνηση πιθανών σεναρίων και ενεργειών που μπορεί να έχει ένας έξυπνος βοηθός του Twitter για να μπορεί να εξυπηρετήσει έναν απλό χρήστη.

4.1.2 Πιθανά Σεναρία Έξυπνου Βοηθού

Στόχος της ανάπτυξης ενός έξυπνου βοηθού για το Twitter είναι να αυτοματοποιήσει κάποιες ενέργειες που θα έκανε ο χρήστης ανάλογα με τους ιδιαίτερους σκοπούς του. Τα χαρακτηριστικά και οι ιδιαιτερότητες κάθε χρήστη μπορούν να εκφραστούν με την περιγραφή συγκεκριμένων σεναρίων. Πριν από την υλοποίηση του συστήματος, δημιουργήσαμε μια υποψήφια λίστα με παραδείγματα σεναρίων που ενδεχομένως να μπορούσε να εκτελέσει ο έξυπνος βοηθός εκ μέρους του χρήστη :

- Δημιουργία απλού tweet
- Προγραμματισμένο tweet, πχ κάθε μέρα στις 9.00πμ να αποστέλλει «Καλημέρα!»
- Προβολή επιλεγμένων tweets, πχ. ενός συγκεκριμένου ατόμου που ακολουθεί ή με φιλτράρισμα κάποιων keywords που ορίζει ο χρήστης
- Αυτόματη απάντηση (reply) σε κάποιο tweet που αναφέρεται στον χρήστη πχ. «Θα σας απαντήσω σε λίγο»
- Αυτόματο retweet επιλεγμένων tweets, πχ. των posts ενός συγκεκριμένου ατόμου που ακολουθεί ο χρήστης

- Αυτόματο like επιλεγμένων tweets
- Αποδοχή ή απόρριψη υποψήφιων followers (αυτών που έχουν κάνει αίτηση για να ακολουθούν το χρήστη)
- Αυτόματη απάντηση (reply) σε κάποιο tweet με κάποιο κανόνα πχ. αν βρει ένα post που περιέχει τις λέξεις “artificial” και “intelligence” να κάνει reply “I love artificial intelligence!”
- Να αποστέλλει μήνυμα «Χρόνια Πολλά!!!» αν βρει κάποιον από τους ακόλουθους του ή αυτούς που ακολουθεί να έχει γενέθλια τη συγκεκριμένη μέρα.
- Να κάνει αίτηση να ακολουθήσει άτομα της περιοχής του ή κάποιας άλλης επιθυμητής περιοχής για να μαθαίνει νέα από εκεί.

4.1.3 Πιθανές Είσοδοι και Ενέργειες Έξυπνου Βοηθού

Στα σενάρια που περιγράφηκαν στην προηγούμενη ενότητα ή σε οποιοδήποτε άλλο σενάριο αυτού του τύπου μπορούμε να ξεχωρίσουμε δυο διακριτές καταστάσεις. Η πρώτη αφορά την είσοδο ή έναυσμα (trigger) του σεναρίου και η δεύτερη το αποτέλεσμα ή ενέργεια του σεναρίου. Οι καταστάσεις αυτές αποτελούν ουσιαστικά τα trigger και action, όπως περιγράφηκαν στην ενότητα 2.5 για τον trigger-action Προγραμματισμό.

Για παράδειγμα για το σενάριο

«Να αποστέλλει μήνυμα «Χρόνια Πολλά!!!» αν βρει κάποιον από τους ακόλουθους του ή αυτούς που ακολουθεί να έχει γενέθλια τη συγκεκριμένη μέρα»

ως είσοδος μπορεί να θεωρηθεί τα γενέθλια ενός ατόμου, όπως είναι σημειωμένα στο ημερολόγιο του χρήστη ή η σημερινή ημέρα, ενώ ως ενέργεια η αποστολή του tweet με το συγκεκριμένο μήνυμα.

Πριν από την υλοποίηση του έξυπνου βοηθού καταγράφηκαν κάποιοι πιθανοί είσοδοι καθώς και πιθανές ενέργειες-αποτελέσματα του συστήματος.

Τις εισόδους τις διαχωρίσαμε σε δύο κατηγορίες. Αυτές που μπορεί να προέρχονται από το ίδιο το Twitter και αυτές που προέρχονται από τις ποικίλες πηγές του διαδικτύου. Μια λίστα με τέτοιες πιθανές εισόδους είναι η ακόλουθη:

1. Είσοδοι προερχόμενοι από το Twitter:

- Νέα αναφορά (mention) για το χρήστη
- Νέα αίτηση φιλίας (follow) από κάποιον άλλο χρήστη
- Νέο tweet από ένα συγκεκριμένο χρήστη
- Νέο tweet από αναζήτηση tweets με βάση κάποιο φίλτρο
- Νέο tweet από χρήστες συγκεκριμένης περιοχής

2. Είσοδοι προερχόμενοι από εξωτερικές πηγές του διαδικτύου:

- Ώρα και ημερομηνία
- Ημερολόγιο πχ. Google Calendar (τίτλος, περιοχή, περιγραφή, εκκίνηση γεγονότος, λήξη γεγονότος)

Παράδειγμα: Να βρίσκει τα γενέθλια ενός ατόμου και να στέλνουμε σχετικό post

- Email πχ Gmail

Παράδειγμα 1 : να στέλνει ένα ευχαριστήριο tweet στον χρήστη που έκανε retweet, like ή αίτηση για φιλία

Παράδειγμα 2: να στέλνει μέσω tweet μια ειδοποίηση ότι έχει λάβει κάποιο email από κάποιο συγκεκριμένο άτομο

- Γεωγραφική θέση πχ Android Location (είσοδος ή έξοδος από μια γεωγραφική περιοχή)
- Ειδήσεις

Παράδειγμα 1 : από το ESPN channel post το τελικό σκορ της αγαπημένης σου ομάδας

Παράδειγμα 2: post ενός άρθρου πχ. από New York Times που περιέχει ένα επιθυμητό θέμα

- Καιρός (Παράδειγμα: post ότι θα βρέξει στην περιοχή που είσαι για να πάρεις ομπρέλα!)

Από την άλλη, οι πιθανές ενέργειες ενός έξυπνου βοηθού για το Twitter είναι περισσότερο προδιαγεγραμμένες. Πρόκειται ουσιαστικά για τις ενέργειες που μπορούν να εκτελεστούν μέσω του REST API για το Twitter. Παρακάτω παρουσιάζεται μια λίστα με πιθανές ενέργειες του Twitter, προερχόμενες από POST μεθόδους του Twitter API (μαζί με τις παραμέτρους τους) :

- Post a tweet (text)
- Post a tweet with media (text, media)
- Retweet (tweet_id, text)
- Like a tweet post (tweet_id)
- Send a direct message (username, text)
- Follow/Unfollow a user (user_id)
- Block/Unblock a user (user_id)
- Mute/Unmute a user (user_id)
- Update account settings
- Update profile
- Update profile picture (image)

- Create/Update/Destroy a list (list_name)
- Subscribe/Unsubscribe to/from a list (list_id)
- Add/Delete members to/from a list (list_id)
- Create/Delete saved searches (query)

Για όλες τις παραπάνω λειτουργίες υπάρχουν και οι αντίστοιχες GET μέθοδοι του Twitter API, οι οποίες επίσης μπορούν να αποτελέσουν πιθανές ενέργειες-αποτελέσματα του έξυπνου βοηθού.

4.2 Αρχιτεκτονική Συστήματος

Στην ενότητα αυτή περιγράφεται η αρχιτεκτονική του συστήματος που αναπτύχθηκε για τον έξυπνο βοηθό για το Twitter. Το σύστημα ονομάστηκε Twitter Assistant και αποτελείται από τα παρακάτω υποσυστήματα.

1. Διεπαφή χρήστη
2. Twitter API (codebird-php)
3. FLASH web service
4. Βάση Δεδομένων (server)

Η διεπαφή χρήστη αποτελεί το κεντρικό σύστημα στον Twitter Assistant που συνδέεται με τα υπόλοιπα υποσυστήματα Twitter API, FLASH web service και Βάση Δεδομένων (server) όπως φαίνεται στην παρακάτω εικόνα.

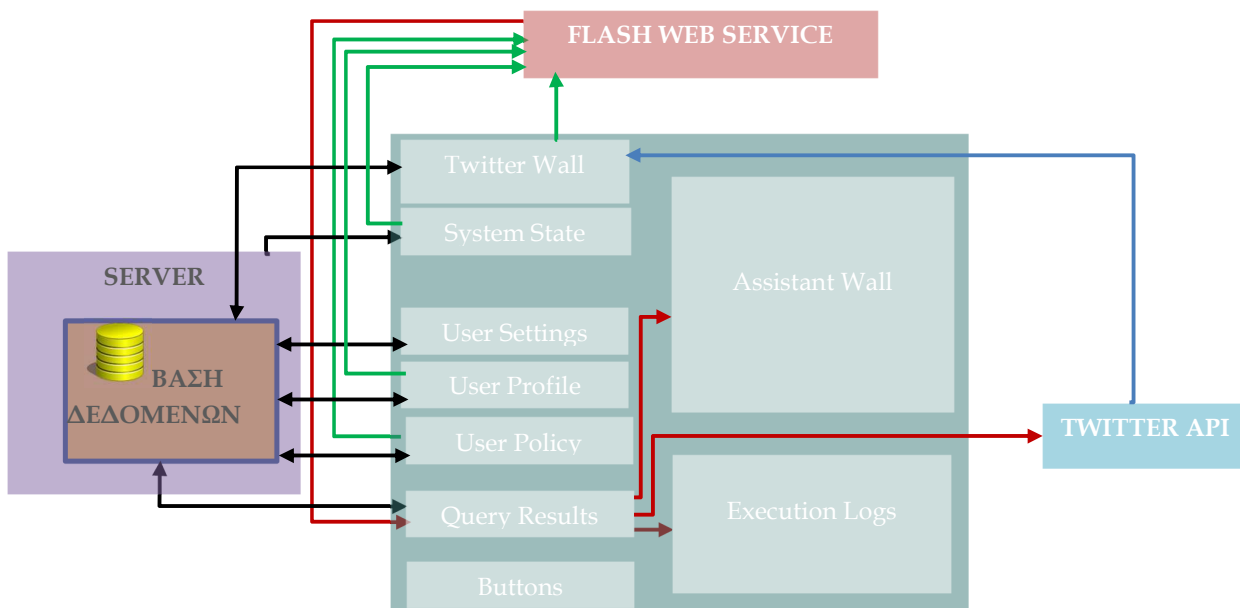


Εικόνα 4.1 : Αρχιτεκτονική συστήματος Twitter Assistant

Η κεντρική σελίδα της διεπαφής χρήστη χωρίζεται στα παρακάτω μέρη:

- Είσοδοι Συστήματος (Twitter Wall/System State)
- Είσοδοι Χρήστη (User Settings/User Profile/User Policy)
- Έξοδος Συστήματος (Query Results)
- Κουμπιά Ενεργειών Χρήστη (Buttons)
- Εμφάνιση Εξόδων Συστήματος (Assistant Wall)
- Εμφάνιση Πληροφοριών (Execution Logs)

Στην παρακάτω εικόνα φαίνεται με περισσότερη λεπτομέρεια η ροή πληροφορίας που παρατηρείται μεταξύ των διάφορων πεδίων της σελίδας διεπαφής χρήστη με τα άλλα υποσυστήματα.



Εικόνα 4.2 : Αναλυτικότερη αναπαράσταση της Αρχιτεκτονικής συστήματος Twitter Assistant

Η σειρά με την οποία γίνεται η ροή της πληροφορίας μεταξύ της διεπαφής χρήστη και των υποσυστημάτων είναι η ακόλουθη :

- Μετά την επιτυχημένη εισαγωγή του χρήστη στο σύστημα του έξυπνου βοηθού συμπληρώνονται τα πεδία που θεωρούμε ως εισόδους του συστήματος (Twitter Wall και System State).
- Συγκεκριμένα στο πεδίο Twitter Wall εμφανίζονται τα tweets του τοίχου του χρήστη μέσω του Twitter API και χρήση της βιβλιοθήκης codebird-php. Ταυτόχρονα τα tweets του τοίχου σώζονται στη Βάση Δεδομένων του server όπου είναι εγκατεστημένος ο Twitter Assistant.
- Την ίδια στιγμή στο πεδίο System State εμφανίζονται κάποιες πληροφορίες του server (ώρα, ημέρα, ημερομηνία, τοποθεσία).
- Ως εισόδους του χρήστη θεωρούμε τα πεδία User Settings, User Profile και User Policy. Τα πεδία αυτά αρχικά έχουν τις default τιμές τους (User Settings) ή είναι κενά (User Profile και User Policy).Ο χρήστης μετά τη συμπλήρωση των πεδίων αυτών μπορεί να σώσει τις τιμές τους στη Βάση Δεδομένων του συστήματος (επιλέγοντας το κατάλληλο κουμπί της διεπαφής).
- Ο χρήστης κάνει ένα αίτημα προς τον έξυπνο βοηθό (επιλέγοντας το κατάλληλο κουμπί της διεπαφής). Αυτό γίνεται μέσω του FLASH web service. Οι πληροφορίες που λαμβάνει το υποσύστημα FLASH είναι τόσο οι Είσοδοι Συστήματος (Twitter Wall και System State) όσο και οι Είσοδοι Χρήστη (User Profile και User Policy).
- Το FLASH web service αποκρίνεται στο παραπάνω αίτημα δίνοντας ως απάντηση πληροφορία που εμφανίζεται στο πεδίο Query Results.
- Ο χρήστης επιλέγει το κατάλληλο κουμπί διεπαφής για την εκτέλεση των ενεργειών που υπάρχει ως πληροφορία στο πεδίο Query Results. Αυτό έχει ως αποτέλεσμα την εμφάνιση φιλτραρισμένων-επιλεγμένων tweets στο πεδίο Assistant Wall ή την εκτέλεση ορισμένων ενεργειών στο Twitter μέσω του Twitter API και χρήση της βιβλιοθήκης codebird-php. Ταυτόχρονα εμφανίζονται στο πεδίο Execution Logs πληροφορίες σχετικά με την εκτέλεση των παραπάνω ενεργειών.

- Όταν ο χρήστης ξανασυνδεθεί στο σύστημα τα πεδία που αφορούν τις Εισόδους Συστήματος ενημερώνονται κατάλληλα με καινούριες πληροφορίες. Έτσι αν υπάρχουν νέα tweets για τον χρήστη αυτά προστίθενται στο πεδίο Twitter Wall. Το πεδίο System State επίσης περιέχει ενημερωμένες πληροφορίες.
- Αντίθετα στα πεδία που αφορούν τις Εισόδους Χρήστη ανακτώνται οι πληροφορίες από την Βάση Δεδομένων του συστήματος. Ο χρήστης μπορεί εκ νέου να αλλάξει αυτές τις τιμές και να προχωρήσει σε νέο αίτημα προς τον έξυπνο βοηθό.

4.3 Διεπαφή Χρήστη Twitter Assistant

Στην ενότητα αυτή περιγράφονται αναλυτικά τα πεδία της διεπαφής χρήστη του συστήματος Twitter Assistant.

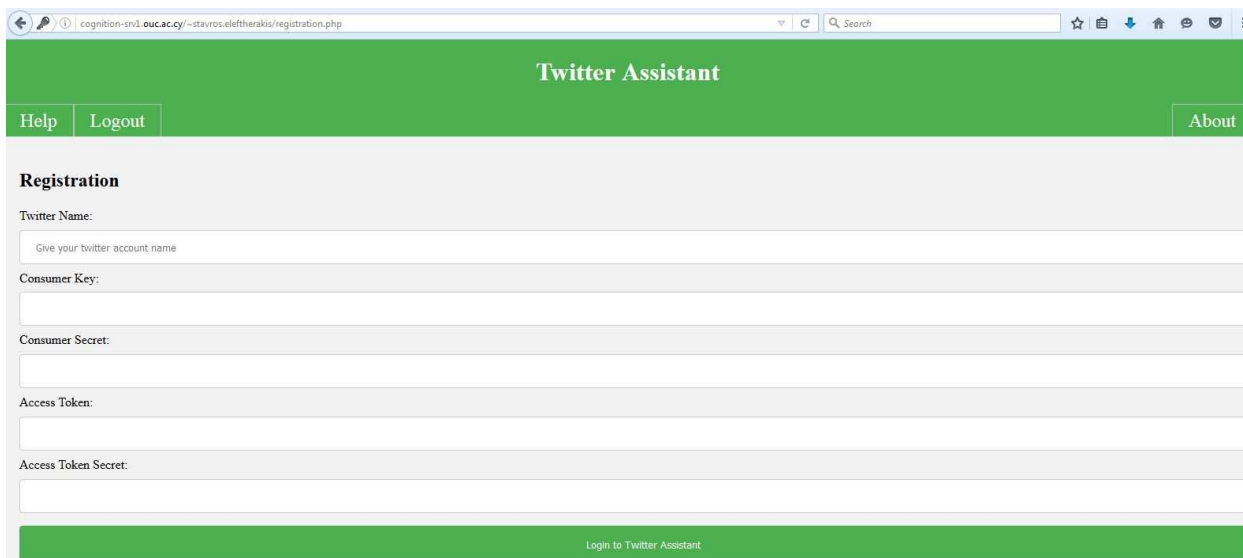
4.3.1 Σύνδεση στο σύστημα (Login)

Η αρχική σελίδα του συστήματος Twitter Assistant φαίνεται στην παρακάτω εικόνα. Ο χρήστης καλείται να επιλέξει ένα Username και ένα Password τα οποία θα χρησιμοποιεί κατά την είσοδό του στο σύστημα. Στη συνέχεια πατώντας το κουμπί “Login to Twitter Assistant” θα μεταφερθεί είτε στη σελίδα εγγραφής στο σύστημα (αν δεν έχει εγγραφεί έως τώρα) ή στην κεντρική σελίδα του Twitter Assistant.

Εικόνα 4.3 : Σελίδα σύνδεσης (Login) στο σύστημα του Twitter Assistant

4.3.2 Εγγραφή στο σύστημα (Registration)

Την πρώτη φορά που ο χρήστης θα προσπαθήσει να συνδεθεί στον Twitter Assistant μεταφέρεται στη σελίδα εγγραφής (registration) στο σύστημα, όπως φαίνεται στην παρακάτω εικόνα.




Εικόνα 4.4 : Σελίδα εγγραφής (Registration) στο σύστημα του Twitter Assistant

Εδώ καλείται να συμπληρώσει το όνομα που έχει στο λογαριασμό του στο Twitter (αναφερόμενο και ως screen name), καθώς και τα παρακάτω κλειδιά που απαιτούνται για την επικύρωση και την εξουσιοδότηση του χρήστη “Consumer Key”, “Consumer Secret”, “Access Token”, “Access Token Secret”. Αν ο χρήστης πατήσει το κουμπί “Help” στο πάνω μέρος της σελίδας θα λάβει αναλυτικές οδηγίες για την απόκτηση των κλειδιών αυτών. Οι οδηγίες αυτές παρουσιάζονται στο Παράρτημα Α.

Μετά την εισαγωγή των κλειδιών που αποκτήθηκαν στα αντίστοιχα πεδία ο χρήστης πατάει το κουμπί “Login to Twitter Assistant” για να συνδεθεί στο σύστημα. Μετά από την επιτυχημένη πιστοποίηση των στοιχείων του χρήστη για τη σύνδεση με την πλατφόρμα του Twitter τα στοιχεία αυτά αποθηκεύονται στη Βάση Δεδομένων και ο χρήστης μεταφέρεται στην κεντρική σελίδα του Twitter Assistant. Η σελίδα Registration παρακάμπτεται τις επόμενες φορές που ο χρήστης θα θέλει να συνδεθεί εκ νέου με το σύστημα του Twitter Assistant. Τα στοιχεία Username και Password της σελίδας σύνδεσης (Login) αρκούν.

4.3.3 Twitter Wall

Στο σημείο αυτό εμφανίζεται σε μορφή πίνακα η λίστα με τα tweets που περιλαμβάνει η αρχική σελίδα (τοίχος) του λογαριασμού Twitter του χρήστη. Η λίστα αυτή περιέχει τα tweets που έχει πραγματοποιήσει ο ίδιος ο χρήστης καθώς και τα tweets που έχουν κάνει άτομα τα οποία ακολουθεί ο χρήστης (following). Οι στήλες του πίνακα αντιπροσωπεύουν τον χρήστη που κάνει το tweet (User), το ίδιο το περιεχόμενο του tweet (Tweet) και την ακριβή ημερομηνία και ώρα που δημοσιεύτηκε το tweet (DateTime).



User	Tweet	DateTime
BBCWorld	No pause to attacks against IS in West Mosul	Sunday 26.03.17 17:05:13
BBCWorld	RT @BBCBreaking: Russian opposition leader Alexei Navalny arrested in Moscow at anti-corruption protest he organised, witnesses say	Sunday 26.03.17 14:45:09
BBCWorld	Mosul offensive: Iraq denies air strike caused civilian deaths	Sunday 26.03.17 14:41:11

Εικόνα 4.5 : Πίνακας με tweets (Twitter Wall)

Ο Twitter Assistant όταν λαμβάνει την πληροφορία του τοίχου κωδικοποιεί καθένα από τα tweets που εμφανίζονται στον πίνακα σύμφωνα με την παρακάτω φόρμα :

`message(USER, ID, TEXT, DATE)`

Οι παράμετροι του εκάστοτε μηνύματος message αντιπροσωπεύουν τα εξής :

- USER -> όνομα λογαριασμού του χρήστη στο Twitter (screen name) που πραγματοποιεί το συγκεκριμένο tweet
- ID -> το μοναδικό ID που αντιστοιχεί στο συγκεκριμένο tweet
- TEXT -> το περιεχόμενο (κείμενο) του συγκεκριμένου tweet
- DATE -> την ημερομηνία που το συγκεκριμένο tweet δημοσιεύτηκε με τη μορφή ΗΜΕΡΑ.ΜΗΝΑΣ.ΕΤΟΣ

Παράδειγμα:

```
message('PES700_0', '832901058645065728', 'Hello world!', '18.02.17')
```

Η ερμηνεία του tweet του παραδείγματος είναι η εξής : Το συγκεκριμένο tweet με το μήνυμα “Hello world!” έχει μοναδικό ID 832901058645065728 και το δημοσίευσε ο χρήστης PES700_0 στις 18 Φεβρουαρίου 2017

Την πληροφορία αυτή μπορεί να τη χρησιμοποιήσει ο χρήστης στους λογικούς κανόνες που γράφει στο πεδίο “User Policy” (ως συνάρτηση message). Το πεδίο Twitter Wall είναι disabled, οι χρήστες δε μπορούν να τροποποιήσουν τον πίνακα παρά μόνο να διαβάσουν το περιεχόμενό του.

4.3.4 System State

Το πεδίο αυτό περιέχει ορισμένες πληροφορίες του συστήματος που λαμβάνονται αυτόματα με τη σύνδεση του χρήστη στον Twitter Assistant, όπως φαίνεται στην παρακάτω εικόνα.



Εικόνα 4.6 : Πληροφορίες Συστήματος (System State)

Οι πληροφορίες που λαμβάνουμε σε αυτό το πεδίο είναι οι εξής :

- `sys_time` -> η τρέχουσα ώρα σε φόρμα 24h, πχ `sys_time(1738)` σημαίνει ώρα 17:38
- `sys_day` -> η τρέχουσα μέρα της εβδομάδας, πχ. `sys_day('Sunday')`
- `sys_date` -> η τρέχουσα ημερομηνία σε φόρμα ΗΜΕΡΑ.ΜΗΝΑΣ.ΕΤΟΣ , πχ. `sys_date('26.03.17')`
- `sys_location` -> η τοποθεσία του συστήματος (εκεί που τρέχει ο server του συστήματος), πχ. `sys_location('Limassol')`

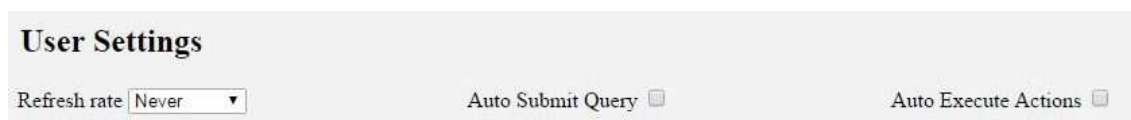
Τις πληροφορίες αυτές επίσης μπορεί να τις χρησιμοποιήσει ο χρήστης στους λογικούς κανόνες που γράφει στο πεδίο “User Policy”. Το πεδίο System State είναι disabled, οι χρήστες δε μπορούν να τροποποιήσουν τις πληροφορίες που περιέχει.

Αντίθετα, μπορούν να εξομοιώσουν ώρα, ημέρα, ημερομηνία και τοποθεσία της αρεσκείας τους δίνοντας τα αντίστοιχα στοιχεία στο πεδίο “User Profile”, πχ. `my_time('1330')`, `my_day('Friday')`, `my_date('01.05.17')`, `my_location('Athens')`.

4.3.5 User Settings

Ο χρήστης μπορεί να ρυθμίσει τα παρακάτω χαρακτηριστικά του έξυπνου βοηθού, όπως φαίνονται και στην εικόνα που ακολουθεί :

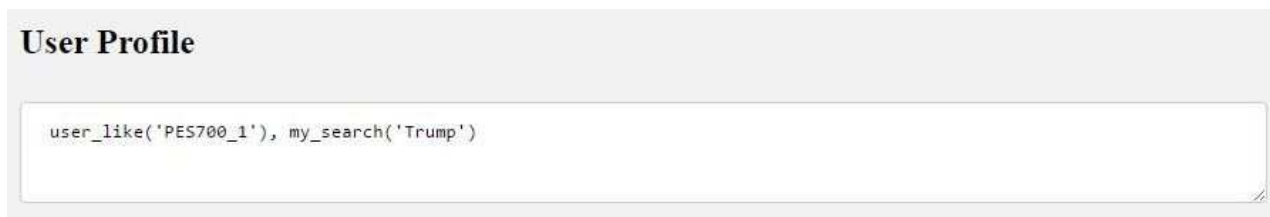
1. Refresh Rate :Ο χρήστης μπορεί να διαλέξει τον επιθυμητό ρυθμό ανανέωσης των εισόδων συστήματος του έξυπνου βοηθού (με χρήση ενός radio πεδίου). Αυτό ουσιαστικά σημαίνει πως οι πληροφορίες που περιέχονται στα πεδία “Twitter Wall” και “System State” μπορούν να ανανεώνονται αυτόματα κάθε 5, 10 ή 20 λεπτά. Η προκαθορισμένη (default) τιμή είναι “Never”, που σημαίνει μη-αυτόματη ανανέωση.
2. Auto Submit Query : Αν ο χρήστης ενεργοποιήσει το πεδίο αυτό (check box), τότε ο έξυπνος βοηθός θα κάνει τις προτάσεις του γεμίζοντας αυτόματα το πεδίο “Query Results”. Δηλαδή σε αυτή την περίπτωση παρακάμπτεται το πάτημα του κουμπιού “Submit Query” από το χρήστη. Η προκαθορισμένη (default) τιμή είναι Off (unchecked).
3. Auto Execute Actions : Αν ο χρήστης ενεργοποιήσει το πεδίο αυτό (check box), τότε ο έξυπνος βοηθός θα εκτελέσει αυτόματα τις προτεινόμενες ενέργειες που περιγράφονται στο πεδίο “Query Results”. Δηλαδή σε αυτή την περίπτωση παρακάμπτεται το πάτημα του κουμπιού “Execute Actions” από το χρήστη. Η προκαθορισμένη (default) τιμή είναι Off (unchecked).



Εικόνα 4.7 : Ρυθμίσεις χρήστη (User Settings)

4.3.6 User Profile

Στο πεδίο αυτό ο χρήστης μπορεί να δηλώσει κάποια χαρακτηριστικά, ιδιαιτερότητες, ανάγκες ή προτιμήσεις του, ώστε να χρησιμοποιήσει τον έξυπνο βοηθό για τον ιδιαίτερο σκοπό που θέλει. Αυτό γίνεται με την προσθήκη ενός ή περισσότερων κατηγορημάτων ελεύθερης-ανοικτής επιλογής του χρήστη, όπως στο παράδειγμα της παρακάτω εικόνας. Τα διαφορετικά στοιχεία που εισάγονται στο πεδίο αυτό πρέπει να χωρίζονται με κόμμα.



Εικόνα 4.8 : Προφίλ Χρήστη (User Profile)

Για το συγκεκριμένο παράδειγμα, μια ερμηνεία των στοιχείων που δίνονται είναι ότι ο χρήστης του έξυπνου βοηθού δείχνει ένα ενδιαφέρον για τα tweets του χρήστη "PES700_1" (`user_like('PES700_1')`) και ότι ενδιαφέρεται για αναζήτηση της λέξης (στη συγκεκριμένη περίπτωση ονόματος) "Trump" (`my_search('Trump')`).

Στο πεδίο αυτό επίσης ο χρήστης έχει τη δυνατότητα να εξομοιώσει πληροφορίες όπως η ώρα, η ημέρα, η ημερομηνία και η τοποθεσία της αρεσκείας του, πχ. `my_time('1330')`, `my_day('Friday')`, `my_date('01.05.17')`, `my_location('Athens')`.

4.3.7 User Policy

Στο πεδίο αυτό ο χρήστης δίνει τους λογικούς κανόνες που θα χρησιμοποιήσει ο έξυπνος βοηθός. Ο Twitter Assistant θα συνδυάσει τις πληροφορίες του συστήματος και του χρήστη με τους λογικούς κανόνες και θα προτείνει αναλόγως στο χρήστη κάποιες σχετικές ενέργειες του Twitter. Οι κανόνες συντακτικά πρέπει να ακολουθούν την παρακάτω συγκεκριμένη φόρμα :

`rule1 :: CONDITION-1, CONDITION-2, ..., CONDITION-n implies CONCLUSION.`

Η ερμηνεία του κανόνα (`rule1`) είναι η εξής : Αν ισχύουν ταυτόχρονα όλες οι υποθέσεις `CONDITION-1, CONDITION-2, ..., CONDITION-n` τότε ισχύει το συμπέρασμα `CONCLUSION` του κανόνα.

Κάθε λογικός κανόνας τερματίζει με το σύμβολο τελεία (.)

Ένας λογικός κανόνας μπορεί να καταλήγει σε αρνητικό συμπέρασμα προτάσσοντας το σύμβολο – στο CONCLUSION, όπως για παράδειγμα `-do_post(TEXT)`, που ερμηνεύεται ως «μην κάνεις δημοσίευση το κείμενο TEXT»

Μερικές φορές μπορεί να υπάρχει αντίθεση ανάμεσα στα συμπεράσματα λογικών κανόνων. Στην περίπτωση αυτή πρέπει να δώσουμε προτεραιότητα στους κανόνες με χρήση του συμβόλου `>>`, όπως για παράδειγμα `rule1>>rule2`.

Οι υποθέσεις προέρχονται είτε από τις πληροφορίες του συστήματος (Twitter Wall και System State) είτε από τις πληροφορίες του χρήστη (User Profile). Το συμπέρασμα μπορεί να είναι κάποια από τις προκαθορισμένες ενέργειες που υποστηρίζει ο Twitter Assistant (αναφερόμενη και ως ACTION) ή κάποια ενδιάμεση λογική έννοια. Με τη χρήση ενδιάμεσων εννοιών ο χρήστης είναι σε θέση να υλοποιήσει περισσότερα σύνθετα σενάρια που ικανοποιούν τις ανάγκες του.

Οι προκαθορισμένες ενέργειες (ACTIONS) που υποστηρίζει το σύστημα του Twitter Assistant δίνονται στον παρακάτω πίνακα:

ACTION	ΠΕΡΙΓΡΑΦΗ	ΠΑΡΑΔΕΙΓΜΑ
do_post(TEXT)	Δημοσίευση tweet με το κείμενο TEXT	do_post('Hello')
do_postUser(USER,TEXT)	Δημοσίευση tweet με το κείμενο TEXT, με αναφορά (mention) στον χρήστη USER (το screen name του λογαριασμού του στο Twitter)	do_postUser('Stavros','Hello')

do_retweet(ID)	Αναδημοσίευση (retweet) του tweet με το συγκεκριμένο ID	do_retweet('83290106266237632')
do_like(ID)	Επιλογή του tweet με το συγκεκριμένο ID ως αγαπημένο (like)	do_like('83290106266237632')
do_showList(TEXT)	Εμφάνιση των tweets που προέρχονται από χρήστες που ανήκουν στη λίστα TEXT. Ο χρήστης θα πρέπει να έχει δημιουργήσει τη λίστα αυτή στον λογαριασμό Twitter που διαθέτει.	do_showList('work') Εμφανίζονται όλα τα tweets από χρήστες που ανήκουν στη λίστα "work"
do_showUser(USER)	Εμφάνιση των tweets που έχουν δημοσιευτεί από τον χρήστη USER	do_showUser('Stavros')
do_showId(ID)	Εμφάνιση του tweet με το συγκεκριμένο μοναδικό ID	do_showId('83290106266237632')
do_showLocation(TEXT)	Εμφάνιση των tweets που έχουν δημοσιευτεί από μια συγκεκριμένη τοποθεσία TEXT	do_showLocation('Athens')
do_showDate(DATE)	Εμφάνιση των tweets που έχουν δημοσιευτεί μια συγκεκριμένη ημερομηνία της μορφής ΗΜΕΡΑ.ΜΗΝΑΣ.ΕΤΟΣ	do_showDate('18.02.17')
do_showText(TEXT)	Εμφάνιση των tweets που περιέχουν την φράση TEXT	do_showText('Open University of Cyprus')

Πίνακας 4.9 : Ενέργειες του Twitter Assistant

Ένα παράδειγμα λογικών κανόνων που κάνουν χρήση όλων των πιθανών ενεργειών του Twitter Assistant φαίνεται στην εικόνα 4.6 που ακολουθεί. Η πλήρης περιγραφή του σεναρίου που υλοποιούν αυτοί οι κανόνες υπάρχει στην ενότητα 4.4 του παρόντος κεφαλαίου. Στην ενότητα 4.4 επίσης παρουσιάζεται ένα σενάριο όπου γίνεται χρήση ενδιάμεσων εννοιών.

User Policy

```
r1 :: sys_time(X), #(X>1400) implies do_post('Good evening!').
r2 :: message(USER, ID, TEXT, DATE), #sub_string(TEXT, _, _, 'happy easter') implies do_postUser(USER, 'happy easter too!').
r3 :: message(USER, ID, TEXT, DATE), user_like(USER) implies do_retweet(ID).
r4 :: message(USER, ID, TEXT, DATE), user_like(USER) implies do_like(ID).
r5 :: atWork('') implies do_showList('work').
r6 :: message(USER, ID, TEXT, DATE), user_like(USER) implies do_showUser(USER).
r7 :: message(USER, ID, TEXT, DATE), #sub_string(TEXT, _, _, 'Putin') implies do_showId(ID).
r8 :: message(USER, ID, TEXT, DATE), my_location(LOCATION) implies do_showLocation(LOCATION).
r9 :: message(USER, ID, TEXT, DATE), my_date(DATE) implies do_showDate(DATE).
r10 :: message(USER, ID, TEXT, DATE), my_search(SEARCH) implies do_showText(SEARCH).
```

Εικόνα 4.10 : Κανόνες Χρήστη (User Policy)

Χρήση συναρτήσεων Prolog

Για να δημιουργήσουμε περισσότερο δυνατούς κανόνες οι χρήστες μπορούν να συμπεριλάβουν σε αυτούς κατηγορήματα της γλώσσας λογικού προγραμματισμού Prolog. Μερικά παραδείγματα χρήσης της Prolog δίνονται παρακάτω :

- Έλεγχος ισότητας : πχ. `#(USER=='Stavros')`
- Σύγκριση : πχ. `#(X>=1600)`
- Έλεγχος αν μια μεταβλητή είναι μέλος μιας λίστας :
πχ. `#member(X,['Saturday, 'Sunday'])`
- Έλεγχος αν μια λέξη ή φράση PHRASE είναι υποσύνολο ενός κειμένου TEST
 `#sub_string(TEXT, _, _, PHRASE)`
πχ. `#sub_string('I like programing', _, _, 'programing')`

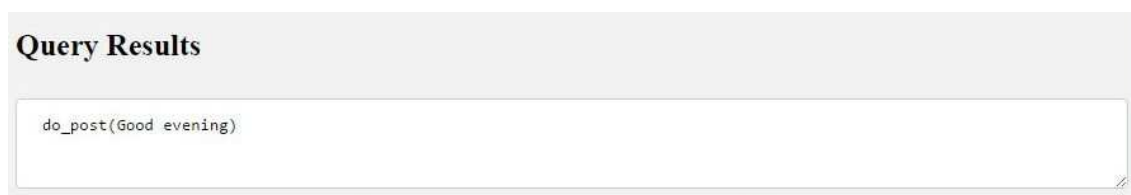
Παρατηρήσεις:

1. Οι παράμετροι οποιασδήποτε λογικής έννοιας πρέπει να δηλώνονται με τη χρήση μονών εισαγωγικών, όπως για παράδειγμα `sys_day('Saturday')`. Ακόμα και στις λογικές έννοιες που έχουν νόημα χωρίς κάποια παράμετρο πρέπει να χρησιμοποιούνται τα μονά εισαγωγικά. Αυτό είναι απαραίτητο για να μπορούμε να χρησιμοποιήσουμε την διαδικτυακή υπηρεσία FLASH για την εξαγωγή συμπερασμάτων από τους λογικούς κανόνες που φτιάχνουμε.
2. Οι παράμετροι μιας συνάρτησης με κεφαλαία γράμματα στους λογικούς κανόνες δηλώνουν την ύπαρξη κάποιας μεταβλητής, όπως για παράδειγμα `message(USER, ID, TEXT, DATE)`

4.3.8 Query Results

Στο πεδίο αυτό ο χρήστης λαμβάνει τις προτάσεις που κάνει το σύστημα του έξυπνου βοηθού και περιέχει όλες τις ενέργειες (ACTIONS) που θα εκτελέσει ο Twitter Assistant για αυτόν. Το πεδίο αυτό συμπληρώνεται όταν ο χρήστης πατάει το κουμπί “Submit Query” ή αυτόματα με τη σύνδεσή του στο σύστημα αν ο χρήστης έχει ενεργοποιήσει την επιλογή “Auto Submit Query”.

Η παρακάτω εικόνα αποτελεί ένα παράδειγμα όπου ο Twitter Assistant προτείνει στο χρήστη να στείλει tweet με το μήνυμα “Good evening”.



Εικόνα 4.11: Παράδειγμα πεδίου Query Results

4.3.9 Κουμπί “Save Profile and Policy”

Οι χρήστες του έξυπνου βοηθού πρέπει να πατάνε το κουμπί αυτό κάθε φορά που τροποποιούν το πεδίο “User Policy”. Αυτό είναι απαραίτητο, γιατί κάθε σύνολο κανόνων που στέλνουν οι χρήστες προς το υποσύστημα FLASH συνοδεύεται και από ένα μοναδικό domain ID που

αποτελεί τη βάση γνώσης του υποσυστήματος. Επίσης οι χρήστες μπορούν να πατούν το κουμπί αυτό όταν τροποποιούν το προφίλ τους (στο πεδίο “User Profile”) έτσι ώστε να σώζουν τις πληροφορίες αυτές στην βάση δεδομένων του συστήματος. Έτσι όταν συνδεθούν την επόμενη φορά στον Twitter Assistant τα στοιχεία αυτά θα υπάρχουν. Γενικά συνίσταται να σώζονται τακτικά οι πληροφορίες του “User Profile” και “User Policy”, ειδικά όταν είναι ενεργοποιημένη η λειτουργία αυτόματης ανανέωσης του έξυπνου βοηθού.

4.3.10 Κουμπί “Submit Query”

Όταν οι χρήστες έχουν συμπληρωμένα τα στοιχεία που αφορούν το προφίλ τους (User Profile) και τους λογικούς κανόνες που θέτουν (User Policy) πατούν το κουμπί αυτό για να στείλουν τις πληροφορίες αυτές στον έξυπνο βοηθό και συγκεκριμένα στο υποσύστημα FLASH.

4.3.11 Κουμπί “Execute Actions”

Όταν ο έξυπνος βοηθός έχει προτείνει στους χρήστες κάποιες προτεινόμενες ενέργειες (είναι συμπληρωμένο το πεδίο “Query Results”, οι χρήστες πατώντας του κουμπί αυτό ουσιαστικά δίνουν στον Twitter Assistant την εντολή να εκτελέσει τις προτάσεις που έχει κάνει.

4.3.12 Assistant Wall

Στο πεδίο αυτό (στο πάνω δεξιό μέρος της οθόνης) εμφανίζονται επιλεγμένα tweets από τον προσωπικό τοίχο των χρηστών, ανάλογα με τις αντίστοιχες ενέργειες που πρότεινε ο έξυπνος βοηθός. Σε περίπτωση που στις προτάσεις δεν υπάρχει καμία ενέργεια επιλογής-φιλτραρίσματος tweets στο πεδίο Assistant Wall εμφανίζεται ολόκληρος ο τοίχος των χρηστών (δηλαδή το πεδίο Assistant Wall ταυτίζεται με το πεδίο Twitter Wall).

Το πεδίο Assistant Wall ανανεώνεται κάθε φορά που οι χρήστες πατούν το κουμπί “Execute Actions” ή αυτόματα με τη σύνδεσή τους στο σύστημα αν έχουν ενεργοποιήσει την επιλογή “Auto Execute Actions”.

Ένα παράδειγμα με τα tweets που εμφανίζονται στο πεδίο Assistant Wall φαίνεται στην παρακάτω εικόνα.

Assistant Wall

Show tweets posted at : 19.03.17

User	Tweet	DateTime
stavelef	FT - Manchester City 1-1 Liverpool [ESPN]	Sunday 19.03.17 20:28:26

Show tweet with ID : 843963064085069825

User	Tweet	DateTime
BBCWorld	Putin hates Clinton: Focus on Russia	Tuesday 21.03.17 01:10:49

Show tweets from list : work

User	Tweet	DateTime
PES700_1	Happy Easter!	Friday 14.04.17 19:56:14

Εικόνα 4.12 : Παράδειγμα πεδίου Assistant Wall

4.3.13 Execution Logs

Σε αυτό το πεδίο (στο κάτω δεξιό μέρος της οθόνης) εμφανίζονται κάποιες χρήσιμες πληροφορίες όσον αφορά τη χρήση του έξυπνου βοηθού, όπως η επαλήθευση λογαριασμού των χρηστών, η καταγραφή των ενεργειών που έκανε ο Twitter Assistant κλπ. Ένα παράδειγμα του πεδίου “Execution Logs” φαίνεται στην παρακάτω εικόνα.

Execution Logs

```
26 March 2017 - 19:23:18 : User Stavros is verified!  
26 March 2017 - 19:23:19 : Show all tweets DONE  
26 March 2017 - 19:48:05 : User Profile & Policy updated succesfully  
26 March 2017 - 19:55:16 : Show all tweets DONE  
26 March 2017 - 19:55:17 : Post message 'Good evening' DONE.
```

Εικόνα 4.13 : Παράδειγμα πεδίου Execution Logs

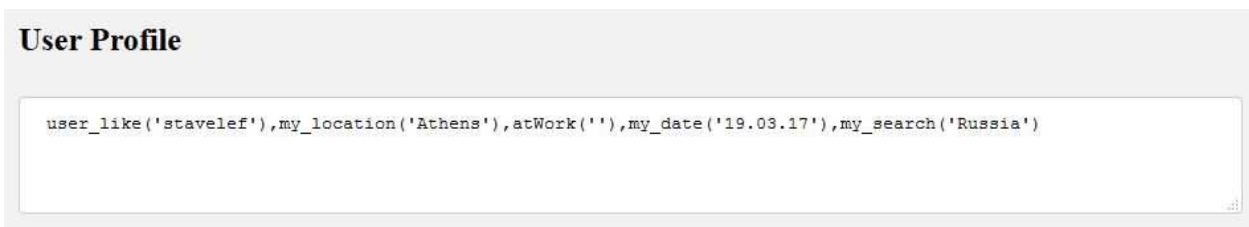
4.4 Σενάρια-Παραδείγματα χρήσης Twitter Assistant

Μετά την είσοδο του στο σύστημα του Twitter Assistant, ο χρήστης πατώντας το κουμπί “Help” (στο πάνω μέρος της σελίδας) λαμβάνει οδηγίες για το πως να γράφει λογικούς κανόνες και άλλες πληροφορίες ώστε να χρησιμοποιεί τον έξυπνο βοηθό προς όφελός του. Στην ενότητα αυτή παρουσιάζονται κάποια σενάρια-παραδείγματα χρήσης του έξυπνου βοηθού Twitter Assistant. Στο πρώτο παράδειγμα δημιουργούμε λογικούς κανόνες που έχουν ως

συμπεράσματα όλες τις προκαθορισμένες ενέργειες (ACTIONS) που υποστηρίζει ο έξυπνος βοηθός. Στο δεύτερο παράδειγμα γίνεται χρήση ενδιάμεσων εννοιών ώστε να μπορεί να υλοποιηθεί κάποιο περισσότερο σύνθετο σενάριο.

4.4.1 Παράδειγμα 1 (Χρήση όλων των πιθανών ενεργειών)

Το παρακάτω παράδειγμα περιέχει ένα σύνολο λογικών κανόνων που καταλήγει ένα προς ένα στις υποστηριζόμενες ενέργειες (ACTIONS) του Twitter Assistant, όπως περιγράφηκαν στον πίνακα 4.9. Στο πεδίο “User Profile” δηλώνουμε ορισμένα κατηγορήματα-έννοιες (εικόνα 4.14), που φανερώνουν τα ενδιαφέροντα, τις επιθυμίες ή τις ανάγκες του χρήστη.



Εικόνα 4.14 : Προφίλ Χρήστη (User Profile) για το σενάριο 1

Μια ερμηνεία των στοιχείων αυτών δίνεται παρακάτω :

`user_like('stavelef')` -> ενδιαφέρον για τον χρήστη “stavelef”

`my_location('Athens')` -> η τοποθεσία του χρήστη είναι “Athens”

`atwork(“)` -> δήλωση ότι ο χρήστης βρίσκεται στο γραφείο της δουλειάς του

`my_date('19.03.17')` -> ο χρήστης ενδιαφέρεται για tweets της ημερομηνίας 19.03.17

`my_search('Russia')` -> ο χρήστης ενδιαφέρεται να αναζητήσει tweets που αφορούν τη λέξη Russia

Οι λογικοί κανόνες του παραδείγματος δίνονται στην παρακάτω εικόνα:

User Policy

```
r1 :: sys_time(X),#(X>1400) implies do_post('Good evening!').
r2 :: message(USER, ID, TEXT, DATE), #sub_string(TEXT,_,_,_, 'happy easter') implies do_postUser(USER, 'happy
easter too!').
r3 :: message(USER, ID, TEXT, DATE), user_like(USER) implies do_retweet(ID).
r4 :: message(USER, ID, TEXT, DATE), user_like(USER) implies do_like(ID).
r5 :: atWork('') implies do_showList('work').
r6 :: message(USER, ID, TEXT, DATE), user_like(USER) implies do_showUser(USER).
r7 :: message(USER, ID, TEXT, DATE), #sub_string(TEXT,_,_,_, 'Putin') implies do_showId(ID).
r8 :: message(USER, ID, TEXT, DATE), my_location(LOCATION) implies do_showLocation(LOCATION).
r9 :: message(USER, ID, TEXT, DATE), my_date(DATE) implies do_showDate(DATE).
r10 :: message(USER, ID, TEXT, DATE), my_search(SEARCH) implies do_showText(SEARCH).
```

Εικόνα 4.15: Κανόνες Χρήστη (User Policy) για το παράδειγμα 1

Η ερμηνεία των λογικών κανόνων είναι η εξής :

r1-> δημοσίευσε το tweet “Good evening!” αν η ώρα είναι μετά τις 14.00

r2 -> απάντησε παρομοίως σε όποιον χρήστη σου στέλνει τις ευχές “happy easter”

r3 -> κάλε retweet όλα τα tweets που προέρχονται από έναν χρήστη USER της αρεσκείας σου

r4 -> κάλε like σε όλα τα tweets που προέρχονται από έναν χρήστη USER της αρεσκείας σου

r5 -> αν είσαι στη δουλειά εμφάνισε μόνο τα tweets που προέρχονται από άτομα του γραφείου σου

r6 -> εμφάνισε μόνο τα tweets που προέρχονται από έναν χρήστη USER της αρεσκείας σου

r7 -> εμφάνισε τα tweets που περιέχουν τη λέξη “Putin”

r8 -> εμφάνισε τα tweets που προέρχονται από μια τοποθεσία LOCATION της αρεσκείας σου

r9 -> εμφάνισε τα tweets που έχουν δημοσιευτεί σε μια ημερομηνία DATE της αρεσκείας σου

r10 -> εμφάνισε τα tweets που περιέχουν μια λέξη ή φράση SEARCH της αρεσκείας σου

Ο Twitter Assistant λαμβάνοντας υπόψιν τα tweets που εμφανίζονται στον προσωπικό τοίχο του χρήστη, τα ενδιαφέροντα του χρήστη (User Profile) και τους λογικούς κανόνες που έχει θέσει (User Policy) ανταποκρίνεται κάνοντας τις προτάσεις που εμφανίζονται στην παρακάτω εικόνα.

Query Results

```
do_like(843529614257541121),do_like(855819271699664896),do_like(855879779043684353),do_post(Good evening!),do_retweet(843529614257541121),do_retweet(855819271699664896),do_retweet(855879779043684353),do_showDate(19.03.17),do_showId(843963064085069825),do_showList(work),do_showLocation(Athens),do_showText(Russia),do_showUser(stavelef),do_postUser(PES700_2,happy easter too!)
```

Εικόνα 4.16 : Αποτελέσματα Twitter Assistant (Query Results)

4.4.2 Παράδειγμα 2 (χρήση ενδιάμεσων εννοιών)

Το παράδειγμα αυτό παρουσιάζει τη χρήση ενδιάμεσων εννοιών, που ο χρήστης μπορεί να εισάγει στους λογικούς κανόνες του, ώστε να δημιουργήσει περισσότερο σύνθετα σενάρια. Έστω ότι θέλουμε να υλοποιήσουμε το ακόλουθο σενάριο:

«Υποθέστε ότι θέλετε να στέλνετε κάθε μέρα αυτόματα ένα χαιρετιστήριο tweet ανάλογα με την ώρα, πχ. 'Good morning!' τις ώρες 6πμ-12, 'Good evening!' τις ώρες 12-9μμ και 'Good night!' τις ώρες 9μμ-12 βράδυ. Θέλουμε να εξασφαλίσουμε επίσης ότι το tweet αυτό θα στέλνεται μια φορά τη μέρα και δεν θα επαναλαμβάνεται κάθε φορά που τρέχουμε τον Twitter Assistant (πχ. δεν έχει καμία λογική να στέλνουμε tweet με καλημέρα κάθε 5 λεπτά!)»

Ένα σύνολο λογικών κανόνων που μπορούν να περιγράψουν το σενάριο αυτό φαίνονται στην παρακάτω εικόνα. Το συμπέρασμα **do_post** αποτελεί μια από τις προκαθορισμένες ενέργειες (ACTIONS) που μπορεί να πραγματοποιήσει ο έξυπνος βοηθός (σύμφωνα με τον πίνακα 4.9). Από την άλλη, το συμπέρασμα **to_post** είναι ένα κατηγορημα που ελεύθερα επινόησε ο χρήστης και μπορεί να το χρησιμοποιήσει (ως ενδιάμεση έννοια) στους λογικούς κανόνες που δίνει στον έξυπνο βοηθό.

User Policy

```
r1 :: sys_time(X), #(X>=0600), #(X<1200) implies to_post('Good morning!').
r2 :: sys_time(X), #(X>=1200), #(X<2100) implies to_post('Good evening!').
r3 :: sys_time(X), #(X>=2100) implies to_post('Good night!').
r4 :: to_post(TEXT) implies do_post(TEXT).
r5 :: message('PES700_0', ID, TEXT, DATE), sys_date(DATE), to_post(TEXT) implies -do_post(TEXT).
r5>>r4.
```

Εικόνα 4.17 : Κανόνες Χρήστη (User Policy) για το παράδειγμα 2

Παρακάτω δίνεται μια επεξήγηση των λογικών κανόνων του σεναρίου:

Η εντολή που λέει στον Twitter Assistant να κάνει tweet είναι η “do_post()”. Χρησιμοποιήσαμε την ενδιαμέση έννοια “to_post()” ως πρόθεση να στείλουμε κάποιο tweet.

Οι κανόνες r1 - r3 δηλώνουν την πρόθεση να στείλουμε tweet με το συγκεκριμένο κείμενο TEXT ανάλογα με την ώρα.

Ο κανόνας r4 δηλώνει ότι αν έχεις πρόθεση να στείλεις tweet με το κείμενο TEXT τότε και να το στείλεις. Ο κανόνας αυτός θα ενεργοποιηθεί την πρώτη φορά που θα στείλουμε το σχετικό tweet.

Ο κανόνας r5 δηλώνει ότι αν έχουμε πρόθεση να στείλουμε ένα tweet με το κείμενο TEXT, to_post(TEXT), αλλά ήδη υπάρχει ένα tweet στον τοίχο μας, message ('PES700_0', ID, TEXT, DATE), που έχουμε στείλει εμείς (PES700_0), με το ίδιο κείμενο TEXT, την ίδια μέρα DATE με την σημερινή sys_date(DATE) τότε να μην το στείλεις. Επειδή οι κανόνες r4,r5 έρχονται σε αντίφαση μεταξύ τους, δίνουμε προτεραιότητα στον r5 (r5>>r4.) για να πετύχουμε το σκοπό του σεναρίου.

4.5 Βάση Δεδομένων

Στο σύστημα του Twitter Assistant χρησιμοποιήθηκε βάση δεδομένων MySQL για δύο κύριους λόγους:

1. Αποθήκευση στοιχείων χρήστη και των κωδικών που απαιτούνται για την πιστοποίηση (authentication) και την εξουσιοδότηση (authorization) του για να μπορεί να χρησιμοποιεί το Twitter API. Έτσι η εγγραφή του χρήστη πραγματοποιείται μόνο την

πρώτη φορά που θα συνδεθεί με το σύστημα και η διαδικασία της πιστοποίησης και της εξουσιοδότησης απλοποιείται, αφού οι κωδικοί που απαιτούνται υπάρχουν στη Βάση Δεδομένων.

Επίσης όλες οι πληροφορίες που εισάγει ο χρήστης στα πεδία διεπαφής χρήστη “User Settings”, “User Profile” και “User Policy” αποθηκεύονται έτσι ώστε να μπορούν να ανακτηθούν σε κάποια επόμενη σύνδεση του χρήστη με το σύστημα.

2. Αποθήκευση της λίστας με τα tweets που εμφανίζονται στον τοίχο του χρήστη (home timeline). Με αυτό τον τρόπο αποφεύγουμε άσκοπες κλήσεις σε συναρτήσεις του Twitter API για να ψάχνουμε κάθε φορά από την αρχή tweets του τοίχου του χρήστη. Αντίθετα καλούμε κατάλληλες συναρτήσεις του Twitter API για να ενημερώνουμε τη Βάση Δεδομένων με καινούρια μόνο tweets του εμφανίστηκαν στον τοίχο του χρήστη.

Επίσης με τη χρήση Βάσης Δεδομένων απλοποιούνται ορισμένες ενέργειες του έξυπνου βοηθού που αφορούν εμφάνιση επιλεγμένων tweets από τον τοίχο του χρήστη. Η αναζήτηση (φιλτράρισμα των tweets) γίνεται με κατάλληλη ερώτηση SQL στη Βάση Δεδομένων και όχι με κλήση κάποιας συνάρτησης του Twitter API.

4.5.1 Πίνακας Στοιχείων Χρήστη

Αρχικά δημιουργήθηκε ο πίνακας **assistant_users**. Σκοπός του πίνακα αυτού είναι να υπάρχουν συγκεντρωμένα όλα τα στοιχεία που αφορούν τους χρήστες του συστήματος. Για κάθε νέο χρήστη του συστήματος πραγματοποιείται εγγραφή στον πίνακα αυτό και τα στοιχεία του ανανεώνονται αναλόγως. Παρακάτω φαίνονται τα πεδία του πίνακα **assistant_users** μαζί με την περιγραφή τους.

```
mysql> describe assistant_users;
+-----+-----+-----+-----+-----+-----+
| Field                | Type                | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| assistant_user_id    | int(6) unsigned    | NO   | PRI | NULL    | auto_increment |
| assistant_name       | varchar(20)         | YES  | UNI | NULL    |                |
| assistant_password   | varchar(20)         | YES  |     | NULL    |                |
| screen_name         | varchar(20)         | YES  | MUL | NULL    |                |
| consumer_key        | varchar(70)         | YES  |     | NULL    |                |
| consumer_secret     | varchar(70)         | YES  |     | NULL    |                |
| access_token        | varchar(70)         | YES  |     | NULL    |                |
| access_token_secret | varchar(70)         | YES  |     | NULL    |                |
| profile             | varchar(400)        | YES  |     | NULL    |                |
| rules               | varchar(4000)       | YES  |     | NULL    |                |
| rules_id            | int(6)              | YES  |     | NULL    |                |
| refresh_rate        | enum('Never', '5 minutes', '10 minutes', '20 minutes') | YES  |     | NULL    |                |
| auto_query          | tinyint(1)          | YES  |     | NULL    |                |
| auto_execution      | tinyint(1)          | YES  |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
14 rows in set (0.01 sec)
```

Εικόνα 4.18: Περιγραφή πίνακα assistant_users στη Βάση Δεδομένων MySQL

assistant_user_id: Το μοναδικό index (ID) που αντιστοιχεί σε έναν χρήστη του Twitter Assistant

assistant_name: Το όνομα (username) που χρησιμοποιεί ο χρήστης για να εισέλθει στο σύστημα του Twitter Assistant

assistant_password: Ο κωδικός (password) που χρησιμοποιεί ο χρήστης για να εισέλθει στο σύστημα του Twitter Assistant

screen_name: Το όνομα του λογαριασμού Twitter του χρήστη που θέλει να συνδέσει με τον Twitter Assistant

consumer_key: Το κλειδί “Consumer Key” που χρειάζεται ο χρήστης του Twitter Assistant για την πιστοποίηση (authentication) αιτημάτων (requests) προς την πλατφόρμα του Twitter

consumer_secret: Ο κωδικός “Consumer Secret” που χρειάζεται ο χρήστης του Twitter Assistant για την πιστοποίηση (authentication) αιτημάτων (requests) προς την πλατφόρμα του Twitter

access_token: Το σημείο πρόσβασης “Access Token” που χρειάζεται ο χρήστης του Twitter Assistant για να πραγματοποιεί αιτήματα (requests) προς το Twitter API εκ μέρους του λογαριασμού του

access_token_secret: Ο κωδικός του σημείου πρόσβασης “Access Token Secret” που χρειάζεται ο χρήστης του Twitter Assistant για να πραγματοποιεί αιτήματα (requests) προς το Twitter API εκ μέρους του λογαριασμού του

profile: Το πεδίο αυτό περιέχει τις πληροφορίες που αφορούν το προφίλ του χρήστη, όπως τις εισάγει στο πεδίο “User Profile”

rules: Το πεδίο αυτό περιέχει τους λογικούς κανόνες που θέτει ο χρήστης στο πεδίο “User Policy”

rules_id: Ο κωδικός της βάσης γνώσης που αντιστοιχεί σε ένα σύνολο λογικών κανόνων που θέτει ο χρήστης. Το πεδίο αυτό χρειάζεται για την επικοινωνία του συστήματος Twitter Assistant με το web service Flash, που χρησιμοποιείται για την εξαγωγή συμπερασμάτων από τους αντίστοιχους λογικούς κανόνες

refresh_rate: Ο επιθυμητός ρυθμός ανανέωσης του συστήματος Twitter Assistant. Μπορεί να λάβει τις τιμές “Never”, 5 λεπτά, 10 λεπτά και 20 λεπτά

auto_query: Boolean μεταβλητή που δηλώνει την ενεργοποίηση ή μη της λειτουργίας αυτόματου αιτήματος προς το web service FLASH

auto_execution: Boolean μεταβλητή που δηλώνει την ενεργοποίηση ή μη της λειτουργίας αυτόματης εκτέλεσης των ενεργειών που προτείνει το σύστημα του Twitter Assistant

4.5.2 Πίνακας Tweets Χρήστη

Για κάθε χρήστη του Twitter Assistant με το όνομα [screen_name] δημιουργείται ένας πίνακας που περιέχει τα tweets που υπάρχουν στον προσωπικό του τοίχο (home timeline) με το όνομα **tweets_<screen_name>**. Για παράδειγμα για τον χρήστη “pes700_user1” δημιουργείται ο αντίστοιχος πίνακας “tweets_pes700_user1”. Παρακάτω φαίνονται τα πεδία του πίνακα αυτού μαζί με την περιγραφή τους.

```
mysql> describe tweets_pes700_user1;
+-----+-----+-----+-----+-----+-----+
| Field          | Type                | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| tweet_id      | bigint(20) unsigned | NO   | PRI | NULL    |      |
| tweet_text    | varchar(160)        | NO   |     | NULL    |      |
| created_at    | varchar(30)         | NO   | MUL | NULL    |      |
| tweet_user_id | bigint(20) unsigned | NO   | MUL | NULL    |      |
| screen_name   | varchar(20)         | NO   | MUL | NULL    |      |
| name          | varchar(20)         | YES  | MUL | NULL    |      |
| tweet_user_location | varchar(30)        | YES  | MUL | NULL    |      |
+-----+-----+-----+-----+-----+-----+
7 rows in set (0.00 sec)
```

Εικόνα 4.19: Περιγραφή πίνακα tweets_pes700_user1 στη Βάση Δεδομένων MySQL

tweet_id: Το μοναδικό index (ID) που χαρακτηρίζει το tweet

tweet_text: Το κείμενο που περιέχει το tweet

created_at: Η ημερομηνία και η ώρα δημοσίευσης του tweet

tweet_user_id: Το μοναδικό index (ID) που χαρακτηρίζει τον χρήστη που έκανε τη δημοσίευση του tweet

screen_name: Το όνομα λογαριασμού του χρήστη που έκανε τη δημοσίευση του tweet

name: Το πλήρες όνομα του χρήστη που έκανε τη δημοσίευση του tweet (όπως το έχει δηλώσει στον λογαριασμό του στο Twitter)

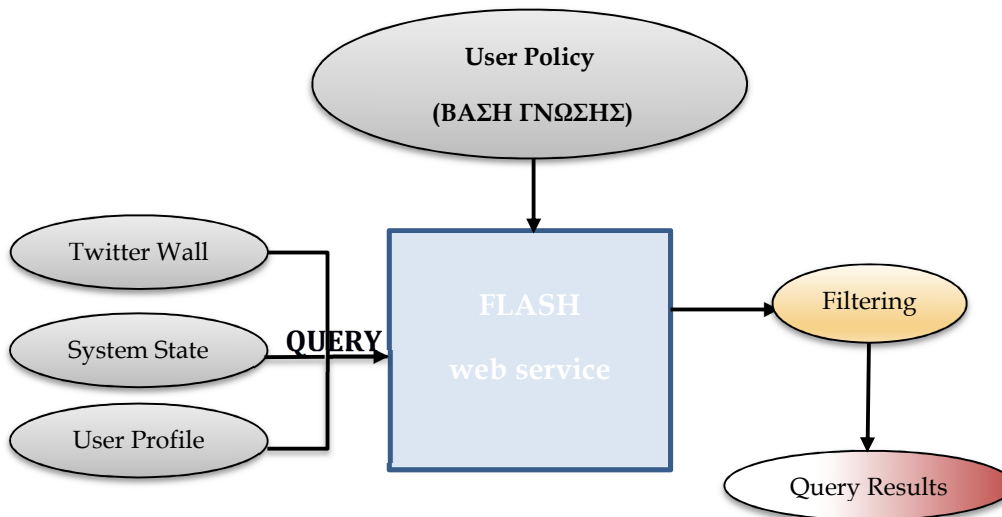
tweet_user_location: Η τοποθεσία που έχει δηλώσει ο χρήστης (στο λογαριασμό του στο Twitter) που έκανε τη δημοσίευση του tweet

4.6 FLASH web service

Στο σύστημα του Twitter Assistant χρησιμοποιήσαμε το εργαλείο γνωστικού συλλογισμού FLASH web service που περιγράψαμε στην ενότητα 3.4. για να εξάγουμε συμπεράσματα λογικών κανόνων.

Ως εισόδους του υποσυστήματος FLASH θεωρούμε τις Εισόδους Συστήματος (“Twitter Wall”, “System State”) και τις Εισόδους Χρήστη (“User Profile”, “User Policy”) που υπάρχουν στην διεπαφή χρήστη, όπως φαίνεται στην παρακάτω εικόνα. Σύμφωνα με τη θεωρία του

γνωστικού συλλογισμού ως Βάση Γνώσης θεωρούμε τους λογικούς κανόνες που εισάγει ο χρήστης στο πεδίο “User Policy”, ενώ ως δεδομένα ερώτησης (Query) θεωρούμε τις πληροφορίες που υπάρχουν στα πεδία “Twitter Wall”, “System State” και “User Profile”.



Εικόνα 4.20 : Είσοδοι και έξοδοι του υποσυστήματος FLASH web service

Το σύστημα του Twitter Assistant με χρήση της τεχνολογίας διαδικτύου AJAX καλεί τις παρακάτω μεθόδους του FLASH web service :

1. Μέθοδος **flash_set_rules** : Με τη μέθοδο αυτή αποθηκεύεται η Βάση Γνώσης (λογικοί κανόνες που θέτει ο χρήστης) στο σύστημα του FLASH. Ταυτόχρονα δίνεται στο FLASH κάποιο μοναδικό id που αντιστοιχεί στη συγκεκριμένη Βάση Γνώσης.
2. Μέθοδος **flash_run_query** : Με τη μέθοδο αυτή στέλνουμε τα δεδομένα που προέρχονται από τα πεδία “Twitter Wall”, “System State” και “User Profile” (Query) στο σύστημα του FLASH. Τα δεδομένα αυτά συγκρίνονται με την αντίστοιχη Βάση Γνώσης και το FLASH επιστρέφει στον Twitter Assistant τα λογικά συμπεράσματα που εξάγονται.

Στα λογικά συμπεράσματα που εξάγονται από το υποσύστημα FLASH ο Twitter Assistant πραγματοποιεί ένα είδος **φιλτραρίσματος** που διατηρεί ως τελικές προτάσεις του βοηθού μόνο τις προκαθορισμένες ενέργειες που υποστηρίζει σύμφωνα με τον πίνακα 4.9. Αυτό γίνεται με κατάλληλο parsing της απόκρισης του FLASH και συγκεκριμένα της

πληροφορίας που υπάρχει στο πεδίο “conclusion”. Σε αυτό το πεδίο εμφανίζονται όλα τα δεδομένα του Query (όλα τα tweets κωδικοποιημένα με το κατηγορημα message, οι πληροφορίες συστήματος sys_* , τα δεδομένα από το “User Profile”) καθώς και όλα τα συμπεράσματα των λογικών κανόνων (τόσο οι τελικές ενέργειες όσο και οι ενδιάμεσες έννοιες όπως περιγράφηκε στην ενότητα 4.3.5). Από όλα αυτά τα στοιχεία διατηρούμε μόνο τα συμπεράσματα που αντιστοιχούν σε ενέργειες προς το Twitter API (συμπεράσματα που αρχίζουν από do_*), τα οποία και εμφανίζονται στο πεδίο “Query Results”.

4.7 Twitter API (codebird-php)

Το σύστημα Twitter Assistant χρησιμοποιεί την διεπαφή προγραμματισμού εφαρμογών Twitter REST API για την ανάγνωση και την εγγραφή tweets. Επίσης για την εκτέλεση των άλλων ενεργειών που υποστηρίζονται, όπως η αναδημοσίευση (retweet), το like και η εμφάνιση tweets που προέρχονται από συγκεκριμένη λίστα χρηστών χρειάζεται το κάλεσμα κατάλληλων συναρτήσεων του REST API.

Η σύνδεση του έξυπνου βοηθού με το Twitter REST API χρειάστηκε για να καλύψουμε τις παρακάτω τρεις ανάγκες :

1. Πιστοποίηση του χρήστη
2. Συλλογή των tweets του τοίχου (home timeline) του χρήστη
3. Εκτέλεση ενεργειών Twitter

Χρησιμοποιήσαμε τη βιβλιοθήκη ανοικτού κώδικα codebird-php ως ενδιάμεση διεπαφή μεταξύ του Twitter Assistant και του REST API, ώστε να απλοποιήσουμε τη διαδικασία. Γενικά στη βιβλιοθήκη codebird-php υπάρχει ένα mapping μεταξύ συναρτήσεων της βιβλιοθήκης και των μεθόδων του Twitter REST API. Για τις υποστηριζόμενες ενέργειες του Twitter Assistant αυτό το mapping δίνεται παρακάτω :

Συνάρτηση Codebird-php	Μέθοδος Twitter REST API	Περιγραφή
account_verifyCredentials	GET account/verify_credentials	Έλεγχος λογαριασμού χρήστη
statuses_homeTimeline	GET statuses/home_timeline	Εμφάνιση tweets του τοίχου (wall) του χρήστη
statuses_update	POST statuses/update	Δημοσίευση Tweet
statuses_retweet_ID	POST statuses/retweet/:id	Αναδημοσίευση Tweet
favorites_create	POST favorites/create	Προσθήκη tweet στα αγαπημένα (like)
lists_statuses	GET lists/statuses	Εμφάνιση tweets που ανήκουν σε λίστα

Πίνακας 4.21 : Mapping συναρτήσεων codebird-php και μεθόδων Twitter REST API

4.7.1 Πιστοποίηση Χρήστη

Όταν ο χρήστης του συστήματος Twitter Assistant εισάγει τα στοιχεία του (username και password) στην αρχική οθόνη, ανακτούνται από τη Βάση Δεδομένων οι κωδικοί που έχει δηλώσει κατά την εγγραφή του στο σύστημα και με βάση αυτά τα στοιχεία γίνεται έλεγχος για την πιστοποίηση του. Αυτό γίνεται με χρήση της συνάρτησης “account_verifyCredentials” της βιβλιοθήκης codebird-php, όπως φαίνεται στο παρακάτω κομμάτι του κώδικα.

```
//check if user has a valid account in Twitter
require_once ('codebird.php');
\Codebird\Codebird::setConsumerKey($consumer_key, $consumer_secret);
$cb = \Codebird\Codebird::getInstance();
$cb->setToken($access_token, $access_token_secret);
$cb->setReturnFormat(CODEBIRD_RETURNFORMAT_ARRAY);

$reply = $cb->account_verifyCredentials(['include_entities' => false,
                                       'skip_status' => true,
                                       'include_email' => false]);
```

Εικόνα 4.22 : Χρήση της συνάρτησης “account_verifyCredentials”

Μόλις επιβεβαιωθούν τα στοιχεία του χρήστη ο Twitter Assistant μεταφέρεται στην κεντρική σελίδα του συστήματος και πλέον είναι δυνατόν να καλεστούν συναρτήσεις του Twitter API που αφορούν το λογαριασμό του συγκεκριμένου χρήστη. Επίσης ένα μήνυμα επιβεβαίωσης εμφανίζεται στο πεδίο “Execution Logs”, όπως στο παρακάτω παράδειγμα. Σε αντίθετη περίπτωση στο χρήστη εμφανίζεται ένα μήνυμα λάθους.



Εικόνα 4.23 : Επιτυχής είσοδος του χρήστη στο σύστημα του Twitter Assistant

4.7.2 Συλλογή Tweets από το Home Timeline του Χρήστη

Για τη συλλογή των Tweets του τοίχου ενός χρήστη χρησιμοποιήσαμε τη μέθοδο “GET statuses/home_timeline”. Αυτή επιστρέφει μια συλλογή από τα ποιά πρόσφατα Tweets και Retweets που δημοσιεύτηκαν από τον εξουσιοδοτημένο χρήστη ή από άλλους χρήστες (following) που ακολουθεί ο εξουσιοδοτημένος χρήστης. Η συλλογή αυτή παίζει κύριο ρόλο στο σύστημά μας, αφού φανερώνει τη σχέση του χρήστη με την υπηρεσία του Twitter και κρατάει μέχρι 800 Tweets.

Συχνά μια τέτοια λίστα μπορεί να περιέχει μεγάλο αριθμό από tweets και από την άλλη υπάρχει περιορισμός πόσα tweets μπορούν να ανακτηθούν από μια κλήση της μεθόδου (μέχρι 200 tweets). Η εφαρμογή που χρησιμοποιεί τη μέθοδο θα πρέπει να την καλεί επαναληπτικά, ώστε να χτίζει σωστά μια λίστα από tweets. Μια τεχνική που εφαρμόζεται είναι αυτή της **σελιδοποίησης** (paging) όπου τα tweets συλλέγονται κατά ομάδες-σελίδες (πχ. 5 σε κάθε επανάληψη).

Ωστόσο υπάρχουν περιπτώσεις που μπορεί να χαθούν tweets αν στηριχθούμε μόνο σε αυτή την τεχνική. Για παράδειγμα, αν ανάμεσα σε δυο κλήσεις της μεθόδου έχουν εντωμεταξύ προστεθεί κάποια καινούρια tweets, η μέθοδος θα επιστρέψει ορισμένα tweets που έχουν ήδη ανακτηθεί από προηγούμενη κλήση. Επίσης όταν θέλουμε να καλέσουμε τη μέθοδο μετά από κάποιο εύλογο χρονικό διάστημα θα θέλαμε η μέθοδος να επιστρέφει μόνο τα καινούρια tweets από την τελευταία φορά που συνδεθήκαμε στο σύστημα.

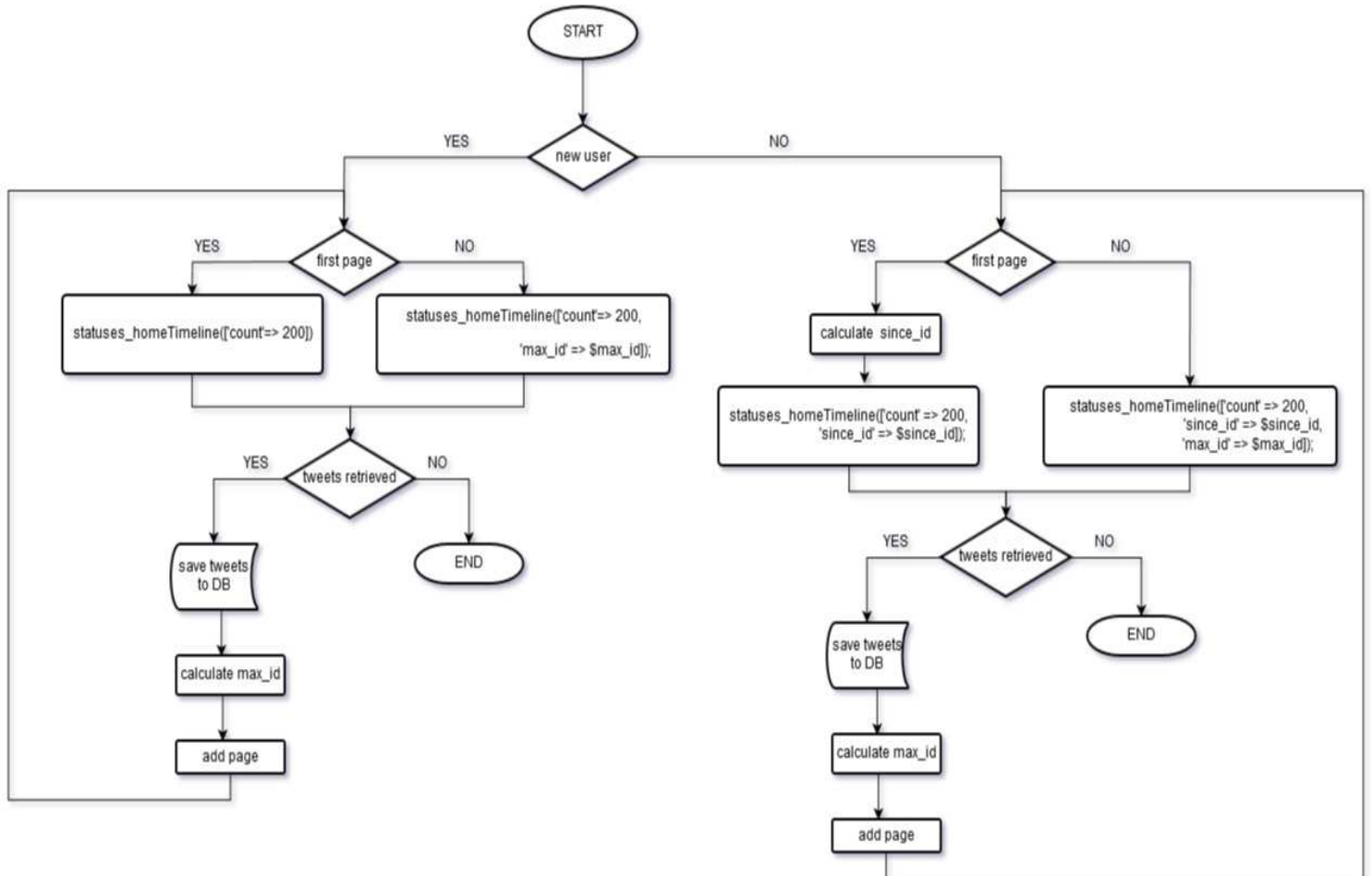
Για να καλύψουμε τις απαιτήσεις αυτές χρειάζεται να εφαρμόσουμε επιπλέον μια τεχνική **cursoring** που χρησιμοποιείται συχνά σε συλλογές ροών δεδομένων (data streaming). Η μέθοδος “GET statuses/home_timeline” επιτρέπει την εφαρμογή μιας τέτοιας τεχνικής μέσω κάποιων παραμέτρων που μπορούν να δοθούν κατά την κλήση της. Ο παρακάτω πίνακας περιέχει τις (προαιρετικές) παραμέτρους της μεθόδου.

Παράμετρος	Περιγραφή	Παράδειγμα τιμής
count	Ο αριθμός των tweets που μπορούν να ανακτηθούν. Ο μέγιστος αριθμός που επιτρέπεται είναι 200, ενώ η default τιμή είναι 20.	5
since_id	Επιστρέφει tweets με ID μεγαλύτερο από αυτό της παραμέτρου, δηλαδή πιο πρόσφατα από αυτό.	12345
max_id	Επιστρέφει tweets με ID μικρότερο ή ίσο από αυτό της παραμέτρου, δηλαδή πιο παλαιά από αυτό.	54321
trim_user	Boolean μεταβλητή που αν σεταριστεί TRUE κάθε tweet θα περιέχει ως πληροφορία για το χρήστη που δημοσίευσε το tweet μόνο το ID του. Αν σεταριστεί ως FALSE θα υπάρχει πλήρης περιγραφή του αντικειμένου user.	true
exclude_replies	Boolean μεταβλητή που αν σεταριστεί TRUE η μέθοδος δε θα επιστρέψει τα tweets που αποτελούν απαντήσεις (replies) σε άλλα tweets	true
include_entities	Boolean μεταβλητή που αν σεταριστεί FALSE τα tweets που επιστρέφει η μέθοδος δεν περιλαμβάνουν τη δομή entities	false

Πίνακας 4.24 : Παράμετροι της μεθόδου GET statuses/home_timeline

Με κατάλληλες κλήσεις της μεθόδου (με χρήση των παραμέτρων “max_id” και “since_id”) μπορούμε να επιτύχουμε τη σωστή συλλογή των tweets χωρίς να έχουμε πλεονασμό δεδομένων.

Στο σύστημα του Twitter Assistant που αναπτύχθηκε η συλλογή των Tweets του τοίχου ενός χρήστη γίνεται με κάλεσμα της συνάρτησης **getTweetsDB()**. Παρακάτω δίνεται το διάγραμμα ροής της συνάρτησης όπου χρησιμοποιήθηκε η τεχνική του cursoring.

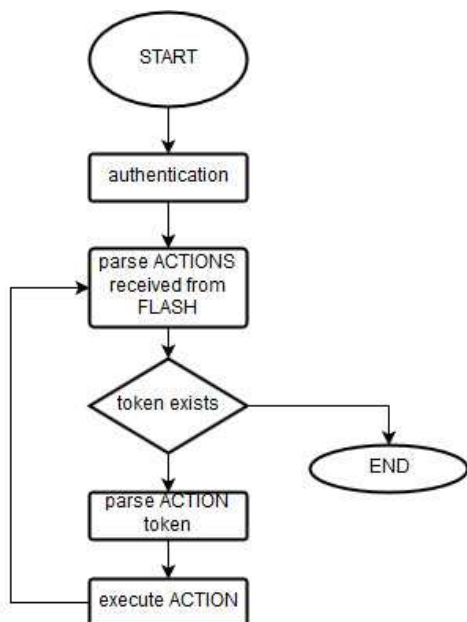


Εικόνα 4.25 : Διάγραμμα ροής της συνάρτησης getTweetsDB()

4.7.3 Εκτέλεση Ενεργειών Twitter

Η εκτέλεση των ενεργειών που προτείνει ο Twitter Assistant γίνεται με το πάτημα του κουμπιού “Execute Actions” και ενώ έχει συμπληρωθεί το πεδίο “Query Results” από τα λογικά συμπεράσματα που επέστρεψε το υποσύστημα FLASH. Η διαδικασία που ακολουθείται φαίνεται στο παρακάτω διάγραμμα ροής. Μετά από την πιστοποίηση του χρήστη, πραγματοποιείται ένας διαχωρισμός των ενεργειών που προτείνονται και για κάθε ενέργεια

καλείται η αντίστοιχη συνάρτηση που υλοποιήθηκε για να επικοινωνήσει με το Twitter API, ώστε να εκτελεστεί τελικά η ενέργεια.



Εικόνα 4.26 : Διάγραμμα ροής της εκτέλεσης ενεργειών Twitter

Οι ενέργειες που μπορεί να εκτελέσει ο Twitter Assistant χωρίζονται σε δυο κατηγορίες :

1. Η πρώτη κατηγορία περιλαμβάνει τις ενέργειες για δημοσίευση tweet, δημοσίευση tweet με αναφορά σε κάποιον χρήστη, αναδημοσίευση (retweet), like και εμφάνιση tweets από συγκεκριμένη λίστα χρηστών. Στον παρακάτω πίνακα παρουσιάζονται όλες οι ενέργειες αυτής της κατηγορίας, οι συναρτήσεις που υλοποιήθηκαν για αυτές, οι αντίστοιχες συναρτήσεις από τη βιβλιοθήκη codebird-php και οι μέθοδοι του Twitter API που χρησιμοποιήθηκαν.

ACTION	Συνάρτηση Twitter Assistant	Συνάρτηση codebird-php	Μέθοδος Twitter API
do_post(TEXT)	post	statuses_update	POST statuses/update
do_postUser(USER,TEXT)	post	statuses_update	POST statuses/update
do_retweet(ID)	retweet	statuses_retweet_ID	POST statuses/retweet/:id
do_like(ID)	like	favorites_create	POST favorites/create
do_showList(TEXT)	showList	lists_statuses	GET lists/statuses

Πίνακας 4.27 : Ενέργειες (ACTIONS) που χρησιμοποιούν μεθόδους του Twitter API

2. Η δεύτερη κατηγορία περιλαμβάνει ενέργειες που σχετίζονται με αναζήτηση ή φιλτράρισμα των tweets που έχει ο χρήστης στον τοίχο του (Twitter Wall). Η επεξεργασία σε αυτή την περίπτωση γίνεται με αιτήματα προς τη Βάση Δεδομένων όπου είναι αποθηκευμένα τα tweets του τοίχου κάθε χρήστη. Τα αποτελέσματα των ενεργειών αυτών εμφανίζονται στο πεδίο “Assistant Wall” του συστήματος. Για αυτές τις ενέργειες δεν απαιτείται χρήση μεθόδων του Twitter API. Ο παρακάτω πίνακας παρουσιάζει τις ενέργειες φιλτραρίσματος των tweets με τις αντίστοιχες συναρτήσεις που υλοποιήθηκαν.

ACTION	Συνάρτηση Twitter Assistant
do_showUser(USER)	showTweetsUser
do_showId(ID)	showTweetsId
do_showLocation(TEXT)	showTweetsLocation
do_showDate(DATE)	showTweetsDate
do_showText(TEXT)	showTweetsText

Πίνακας 4.28 : Ενέργειες (ACTIONS) φιλτραρίσματος του Twitter Wall του χρήστη

Στην περίπτωση που ανάμεσα στις προτεινόμενες ενέργειες του Twitter Assistant δεν υπάρχει κάποια από τη δεύτερη κατηγορία (ενέργειες φιλτραρίσματος) στο πεδίο “Assistant Wall” εμφανίζονται όλα τα tweets που υπάρχουν και στον τοίχο του χρήστη (Twitter Wall).

Κεφάλαιο 5

Αξιολόγηση

5.1 Μελέτες Περίπτωσης

Ένας τρόπος για να διαπιστώσουμε τη χρησιμότητα του έξυπνου βοηθού Twitter Assistant που αναπτύχθηκε είναι να δημιουργήσουμε κάποιες μελέτες περίπτωσης που να αντιστοιχούν σε κάποια αληθοφανή σενάρια. Επιπλέον όσο πιο πολύπλοκα είναι τα σενάρια τόσο περισσότερο φανερώνονται οι δυνατότητες του έξυπνου βοηθού. Σε τέτοιου είδους σενάρια συχνά κάποιοι λογικοί κανόνες αντιτίθενται μεταξύ τους και χρειάζεται να δοθεί προτεραιότητα σε αυτούς. Παρακάτω δίνονται τρία διαφορετικού είδους σενάρια που αντιπροσωπεύουν τις ιδιαιτερότητες, προτιμήσεις και ανάγκες του υποψήφιου χρήστη του έξυπνου βοηθού. Στόχος του πειράματος είναι να χρησιμοποιήσουμε επαρκώς τον Twitter Assistant ώστε :

- 1) Να «μεταφράσουμε» τα σενάρια σε λογικούς κανόνες και σε δεδομένα του προφίλ του χρήστη (δηλαδή να συμπληρώσουμε κατάλληλα τα πεδία “User Policy” και “User Profile”)
- 2) Να εκτελέσουμε επιτυχώς τους στόχους που θέτει το σενάριο

Για να μπορέσουμε να υλοποιήσουμε τα σενάρια δημιουργήσαμε τους παρακάτω λογαριασμούς στο Twitter για τον σκοπό του πειράματος (“PES700_t”, “PES700_1”, “PES700_2”, “shop_pes”) και γράψαμε ορισμένα tweets που σχετίζονται με τα σενάρια. Μάλιστα για ορισμένους από αυτούς δηλώσαμε και μια υποτιθέμενη τοποθεσία π.χ Αθήνα για το χρήστη “PES700_1” και Θεσσαλονίκη για το χρήστη “PES700_2”.

5.1.1 Σενάριο A

«Υποθέστε ότι είστε ο διαχειριστής σε κατάστημα ηλεκτρονικών και θέλετε να στέλνετε το παρακάτω tweet-διαφήμιση 'Hello!Check our shop for electronics!' σε όποιο χρήστη φαίνεται να

ψάχνει ή να ζητάει κάποια ηλεκτρονική συσκευή (πχ. laptops, smartphones). Εξαιρούνται τα tweets που προέρχονται από ανταγωνίστριες εταιρίες. (shop_pes)»

Μία πιθανή αντιστοιχία του παραπάνω σεναρίου σε δεδομένα (προφίλ) και λογικούς κανόνες του χρήστη είναι η εξής.

User Profile:

```
my_competitor('shop_pes')
```

User Policy:

```
r1 :: message(USER, ID, TEXT, DATE), #member(PRODUCT, ['laptop', 'smartphone']),  
#sub_string(TEXT, _, _, PRODUCT), #member(VERB, ['want', 'like', 'search', 'find']),  
#sub_string(TEXT, _, _, VERB) implies to_postUser(USER, 'Check our shop for electronics!').
```

```
r2 :: to_postUser(USER, TEXT) implies do_postUser(USER, TEXT).
```

```
r3 :: to_postUser(USER, TEXT), my_competitor(USER) implies -do_postUser(USER, TEXT).
```

```
r3 >> r2.
```

Η ερμηνεία του προφίλ και των λογικών κανόνων του χρήστη είναι η εξής:

my_competitor('shop_pes') → Δηλώνουμε το χρήστη 'shop_pes' ως ανταγωνίστρια εταιρία

r1 → Ψάχνουμε σε όλα τα tweets του τοίχου μας κάποια πιθανή αναφορά σε επιθυμία του χρήστη (με αναζήτηση λέξεων πχ. 'want', 'like', 'search', 'find') στα προϊόντα που πουλάμε (με αναζήτηση λέξεων πχ. 'laptop', 'smartphone'). Αν βρεθούν τέτοια tweets από το χρήστη @USER τότε σκοπεύουμε (to_postUser) να στείλουμε το μήνυμα 'Check our shop for electronics!' με αναφορά στον χρήστη @USER

r2 → Αν σκοπεύουμε να στείλουμε ένα μήνυμα TEXT με αναφορά σε κάποιο χρήστη @USER (to_postUser) τότε θα εκτελέσουμε την απαιτούμενη ενέργεια για να το στείλουμε με αναφορά στο χρήστη @USER (do_postUser)

r3 \longrightarrow Αν σκοπεύουμε να στείλουμε ένα μήνυμα TEXT με αναφορά σε κάποιο χρήστη @USER (to_postUser) και ο χρήστης @USER είναι ανταγωνίστρια εταιρία (my_competitor) τότε δεν στέλνουμε το μήνυμα (-do_postUser)

r3>>r2. \longrightarrow Σε περίπτωση που βρούμε κάποιο σχετικό μήνυμα από χρήστη ανταγωνίστριας εταιρίας ενεργοποιούνται οι κανόνες r2 και r3 τα συμπεράσματα των οποίων έρχονται σε αντίθεση. Με τον ορισμό της προτεραιότητας αυτής κατορθώνουμε να μη στείλουμε το μήνυμα στην περίπτωση αυτή, ενώ να το στείλουμε σε οποιοδήποτε άλλο χρήστη.

5.1.2 Σενάριο B

«Υποθέστε ότι είστε ένας εργαζόμενος σε εταιρία fast food στην Αθήνα, με ωράριο 9πμ-5μμ και είστε vegetarian. Δεν θέλετε να λαμβάνεται μηνύματα (στο πεδίο Assistant Wall) που αφορούν διαφημίσεις για fast food,εκτός και εάν προέρχονται από το αφεντικό σας (PES700_2) και είτε είναι σε ώρες εργασίας ή απευθύνει κάποιο ερώτημα σε εσάς (ακόμα εκτός ωρών εργασίας, αλλά μόνο αν βρίσκεστε σε αποστολή από το αφεντικό σας).»

Μία πιθανή αντιστοιχία του παραπάνω σεναρίου σε δεδομένα (προφίλ) και λογικούς κανόνες του χρήστη είναι η εξής :

User Profile:

```
my_location('Athens'), outsideWork(""), my_Boss('PES700_2'), vegetarian("")
```

User Policy:

```
r1 :: sys_time(X), #(X>=0900), #(X<1700) implies isWorkingHour("").
```

```
r2 :: message(USER,ID,TEXT,DATE), #sub_string(TEXT,_,_,'fast food'), vegetarian("") implies -do_showId(ID).
```

```
r3 :: message(USER,ID,TEXT,DATE), #sub_string(TEXT,_,_,'fast food'), my_Boss(USER), isWorkingHour("") implies do_showId(ID).
```

```
r4 :: message(USER, ID, TEXT, DATE), my_Boss(USER), #sub_string(TEXT, __, __, '?'),  
outsideWork("") implies do_showId(ID).
```

```
r5 :: message(USER, ID, TEXT, DATE), #sub_string(TEXT, __, __, 'fast food') implies do_showId(ID).
```

```
r3 >> r2.
```

```
r4 >> r2.
```

```
r2 >> r5.
```

Η ερμηνεία του προφίλ και των λογικών κανόνων του χρήστη είναι η εξής:

`my_location('Athens')` → Δηλώνουμε την τοποθεσία του χρήστη (Athens)

`outsideWork("")` → Δηλώνουμε ότι ο χρήστης βρίσκεται σε αποστολή από το αφεντικό του

`my_Boss('PES700_2')` → Δηλώνουμε το χρήστη "PES700_2" ως αφεντικό του χρήστη του έξυπνου βοηθού

`vegetarian("")` → Δηλώνουμε ότι ο χρήστης του έξυπνου βοηθού είναι vegetarian

`r1` → Δηλώνουμε το ωράριο του εργαζομένου-χρήστη του συστήματος (9πμ-5μμ)

`r2` → Αν ο χρήστης είναι vegetarian δε θα εμφανιστούν τα μηνύματα του τοίχου του που περιέχουν τη φράση πχ. 'fast food'

`r3` → Εμφανίζονται τα μηνύματα του τοίχου του χρήστη που περιέχουν τη φράση πχ. 'fast food', προέρχονται από το αφεντικό του χρήστη και βρισκόμαστε σε ώρα εργασίας

`r4` → Εμφανίζονται τα μηνύματα του τοίχου του χρήστη που περιέχουν κάποια ερώτηση από το αφεντικό του χρήστη όταν ο χρήστης βρίσκεται σε κάποια αποστολή από το αφεντικό του

r5 → Εμφανίζονται τα μηνύματα του τοίχου του χρήστη που περιέχουν τη φράση πχ. 'fast food'

r3>>r2.
r4>>r2 → Καθορίζονται κατάλληλα οι προτεραιότητες των κανόνων ώστε να επιτευχθεί το σενάριο που θέλουμε να υλοποιήσουμε
r2>>r5.

5.1.3 Σενάριο Γ

«Υποθέστε ότι είστε ένας αθλητής της ομάδας Celtics και τις ημέρες που η ομάδα σας έχει αγώνες δεν πρέπει να βλέπετε tweets σχετικά με την ομάδα σας για να μην επηρεαστείτε (στο πεδίο Assistant Wall), εκτός αν προέρχονται από τον προπονητή σας (PES700_2). Επίσης θέλετε να κάνετε like στα tweets που αναφέρονται σε νίκη της ομάδας σας, εκτός από αυτά του προπονητή σας (για να μη θεωρηθεί ότι θέλετε να κερδίσετε την εύνοια του)»

Μία πιθανή αντιστοιχία του παραπάνω σεναρίου σε δεδομένα (προφίλ) και λογικούς κανόνες του χρήστη είναι η εξής :

User Profile:

```
my_team('Celtics'), my_coach('PES700_2')
```

User Policy:

```
r1 :: sys_date(X), #member(X,['12.03.17','18.03.17','26.03.17']) implies today_match('').
```

```
r2 :: message(USER,ID,TEXT,DATE), my_team(TEAM), sub_string(TEXT,_,_,TEAM),  
today_match('') implies -do_showId(ID).
```

```
r3 :: message(USER,ID,TEXT,DATE), my_coach(USER), my_team(TEAM),  
#sub_string(TEXT,_,_,TEAM) implies do_showId(ID).
```

```

r4      ::      message(USER,ID,TEXT,DATE),      #sub_string(TEXT,_,_,_'Celtics'),
#sub_string(TEXT,_,_,_'win') implies do_like(ID).

r5      ::      message(USER,ID,TEXT,DATE),      #sub_string(TEXT,_,_,_'Celtics'),
#sub_string(TEXT,_,_,_'win'), my_coach(USER) implies -do_like(ID).

r6 :: message(USER,ID,TEXT,DATE) implies do_showId(ID).

r3>>r2.

r2>>r6.

r5>>r4.

```

Η ερμηνεία του προφίλ και των λογικών κανόνων του χρήστη είναι η εξής:

`my_team('Celtics')` → Δηλώνουμε την ομάδα ('Celtics') για την οποία ενδιαφερόμαστε

`my_coach('PES700_2')` → Δηλώνουμε το όνομα χρήστη ('PES700_2') του προπονητή μας

`r1` → Δηλώνουμε το πρόγραμμα αγώνων της ομάδας, πχ στις '12.03.17', '18.03.17', '26.03.17'

`r2` → Τις ημέρες που η ομάδα έχει αγώνα (`today_match("")`), δεν εμφανίζονται τα tweets που έχουν αναφορά στην ομάδα (`-do_showID`)

`r3` → Εμφανίζονται τα tweets (`do_showID`) που έχουν αναφορά στην ομάδα και προέρχονται από τον προπονητή (`my_coach(USER)`) ανεξαρτήτου ημέρας.

`r4` → Κάνουμε like (`do_like`) στα tweets του τοίχου που αναφέρονται σε νίκη (αναζήτηση της λέξης πχ. 'win') της ομάδας (αναζήτηση της λέξης πχ. 'Celtics')

`r5` → Δεν κάνουμε like (`-do_like`) στα tweets του τοίχου που αναφέρονται σε νίκη (αναζήτηση της λέξης πχ. 'win') της ομάδας (αναζήτηση της λέξης πχ. 'Celtics') αν προέρχονται από τον προπονητή (`my_coach(USER)`)

r6 → Γενικός κανόνας για να εμφανίζονται (do_showID) όλα τα tweets του τοίχου

r3>>r2.

r2>>r6.

r5>>r4.

Καθορίζονται κατάλληλα οι προτεραιότητες των κανόνων ώστε να επιτευχθεί το σενάριο που θέλουμε να υλοποιήσουμε

5.2 Οργάνωση Πειράματος

Για την αποδοχή και την αξιολόγηση του συστήματος που αναπτύχθηκε κρίθηκε σκόπιμο να γίνει μια παρουσίαση-επίδειξη του έξυπνου βοηθού Twitter Assistant στην ομάδα μεταπτυχιακών φοιτητών του μαθήματος “EPL680-Cognitive Programming” του Πανεπιστημίου Κύπρου. Στη συνέχεια ζητήθηκε από τους φοιτητές να πάρουν μέρος σε ένα Πείραμα [Παράρτημα B.1], όπου καλούνται να χρησιμοποιήσουν τον Twitter Assistant για να εκτελέσουν ορισμένες ενέργειες, να προσπαθήσουν να υλοποιήσουν τα σενάρια της ενότητας 5.1 ή και να προτείνουν κάποιο δικό τους σενάριο.

Επιπλέον δημιουργήθηκε ένα ερωτηματολόγιο [Παράρτημα B.2] με σκοπό την αξιολόγηση του συστήματος Twitter Assistant που αναπτύχθηκε, ως προς την ευχρηστία του, την ευκολία μάθησης και την συνολική ικανοποίηση του χρήστη που το δοκίμασε. Το ερωτηματολόγιο δημιουργήθηκε με χρήση της εφαρμογής Google Forms.

5.3 Αποτελέσματα

5.3.1 Μελέτες Περίπτωσης

Η υλοποίηση των μελετών περίπτωσης που περιγράφηκαν στην ενότητα 5.1 και η εκτέλεση των αντίστοιχων ενεργειών από το σύστημα του Twitter Assistant, φανερώνει πως ο έξυπνος βοηθός που αναπτύχθηκε είναι σε θέση να δώσει λύσεις σε πραγματικά προβλήματα των χρηστών του.

Συγκεκριμένα η μελέτη περίπτωσης του σεναρίου Α (ενότητα 5.1.1) έδειξε πως ο Twitter Assistant μπορεί κάλλιστα να χρησιμοποιηθεί για διαφημιστικούς λόγους. Έτσι για παράδειγμα ένας χρήστης μπορεί να παραμετροποιήσει τον βοηθό κατάλληλα ώστε για να πραγματοποιεί στοχευμένες διαφημίσεις κάποιου προϊόντος ή υπηρεσίας. Επίσης μπορεί να κάνει αυτόματα retweet ή like σε συγκεκριμένα tweets που αναφέρονται σε αντίστοιχα προϊόντα ή υπηρεσίες.

Οι μελέτες περίπτωσης των σεναρίων Β και Γ (ενότητες 5.1.2-5.1.3) από την άλλη δείχνουν πως ο έξυπνος βοηθός μπορεί να χρησιμοποιηθεί από οποιοδήποτε μέσο χρήστη του Twitter που μπορεί να παραμετροποιήσει τον βοηθό για να επιτύχει συγκεκριμένους προσωπικούς στόχους, ανάλογα με τις επιθυμίες, τις επιλογές, τις ανάγκες και τις προτεραιότητές του.

5.3.2 Αποτελέσματα Πειράματος

Όσον αφορά το πείραμα, από το σύνολο των ατόμων που κλήθηκαν να συμμετέχουν σε αυτό μόνο δύο άτομα απάντησαν ολοκληρωμένα και έστειλαν συμπληρωμένο πίσω το Ερωτηματολόγιο Πειράματος [Παράρτημα Β1]. Από τις απαντήσεις τους φαίνεται πως κατάφεραν να δημιουργήσουν τους λογικούς κανόνες που χρειαζόταν για την υλοποίηση των σεναρίων σε ικανοποιητικό βαθμό και να εκτελέσουν αντίστοιχα τις προτάσεις που τους πρότεινε ο έξυπνος βοηθός. Επίσης κατόρθωσαν να υλοποιήσουν ένα δικό τους σενάριο που αντιπροσωπεύει κάποια ιδέα χρήσης του έξυπνου βοηθού.

Για την δημιουργία και την εκτέλεση των σεναρίων οι χρήστες χρειάστηκαν περίπου 50 λεπτά. Επίσης οι απαντήσεις τους για το πόσο εύκολο-δύσκολο ήταν να υλοποιήσουν τα σενάρια (σε κλίμακα likert 1-5) επαληθεύει την εσκεμμένη κλιμακωτή δυσκολία των σεναρίων που επιλέχτηκαν για το πείραμα. Ενδιαφέρον παρουσιάζει επίσης το γεγονός ότι δημιούργησαν σχετικά εύκολα και γρήγορα το δικό τους σενάριο στο τελευταίο στάδιο του πειράματος. Αυτό φανερώνει πως ένας χρήστης μπορεί να χρησιμοποιήσει αποδοτικά τον Twitter Assistant μετά από κατάλληλη εξοικείωση με αυτό.

Γεγονός είναι πως η συμμετοχή στο πείραμα απαιτούσε συνολικά την αφιέρωση σημαντικού χρόνου (1-2 ώρες). Αρχικά έπρεπε ο χρήστης να δημιουργήσει λογαριασμό στο Twitter, αλλά και να ακολουθήσει μια διαδικασία για να αποκτήσει πρόσβαση στο λογαριασμό του μέσω από τον έξυπνο βοηθό. Επιπλέον έπρεπε να μελετήσει προσεχτικά τις σχετικές οδηγίες για το πώς να δημιουργήσει τους λογικούς κανόνες μέσω των οποίων ο έξυπνος βοηθός ανταποκρίνεται

στα αιτήματά του. Η ιδιαιτερότητα της χρήσης του έξυπνου βοηθού μέσω δημιουργίας λογικών κανόνων σε συνδυασμό με το γεγονός πως οι περισσότεροι δεν ήταν εξοικειωμένοι με το Twitter συνέβαλλε στη μικρή σχετικά ολοκληρωμένη συμμετοχή στο πείραμα.

5.3.3 Ανάλυση Αποτελεσμάτων Ερωτηματολογίου

Όπως επαληθεύτηκε από τη βάση δεδομένων που επικοινωνεί με το σύστημα, 10 άτομα συμμετείχαν στο πείραμα γράφοντας κάποιους λογικούς κανόνες, δηλαδή χρησιμοποίησαν τον έξυπνο βοηθό. Έτσι απάντησαν στο Ερωτηματολόγιο χρήσης του Twitter Assistant, αξιολογώντας τον έξυπνο βοηθό και δίνοντας κάποια σημαντικά σχόλια και παρατηρήσεις.

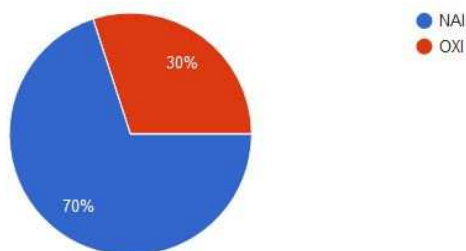
Οι πρώτες ερωτήσεις του ερωτηματολογίου είναι γενικές και αφορούν τη σχέση των χρηστών με τους έξυπνους βοηθούς και το μέσο κοινωνικής δικτύωσης Twitter.

Από το σύνολο των εθελοντών μόνο το 30% έχει χρησιμοποιήσει κατά το παρελθόν κάποιον έξυπνο βοηθό (Siri, Google Assistant, Cortana, Google Now), ενώ κανένας δεν έχει χρησιμοποιήσει κάποιον έξυπνο βοηθό σχετικό με το Twitter.

Η σχέση των χρηστών με το Twitter απεικονίζεται στις εικόνες 5.1- 5.3

Έχετε ενεργό λογαριασμό στο Twitter?

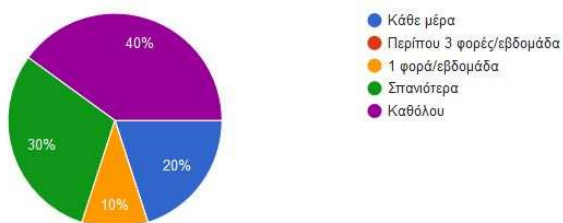
10 απαντήσεις



Εικόνα 5.1 : Ερώτηση 1

Πόσο συχνά χρησιμοποιείτε το Twitter?

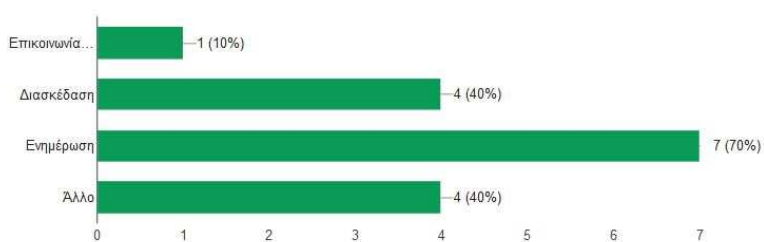
10 απαντήσεις



Εικόνα 5.2 : Ερώτηση 2

Για ποιο λόγο χρησιμοποιείται το Twitter?

10 απαντήσεις

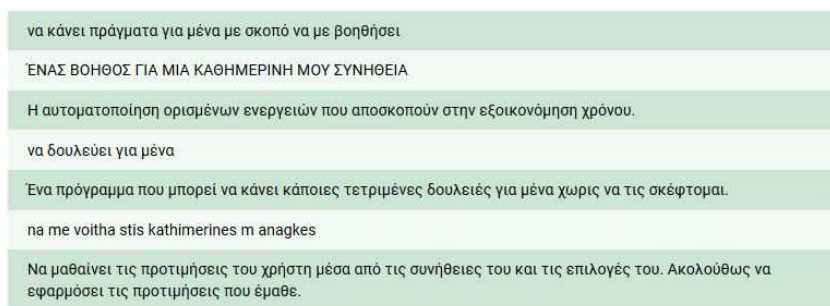


Εικόνα 5.3 : Ερώτηση 3

Οι επόμενες ερωτήσεις των εικόνων 5.4 – 5.5 παρουσιάζουν το πώς φαντάζονται οι εθελοντές έναν έξυπνο βοηθό και πως έναν εξειδικευμένο για την ασχολία τους με το Twitter.

Τι σημαίνει για εσάς ένας έξυπνος ψηφιακός βοηθός?

7 απαντήσεις



Εικόνα 5.4 : Ερώτηση 4

Πως φαντάζεστε έναν έξυπνο ψηφιακό βοηθό για το Twitter? Τι χαρακτηριστικά πρέπει να έχει?

9 απαντήσεις

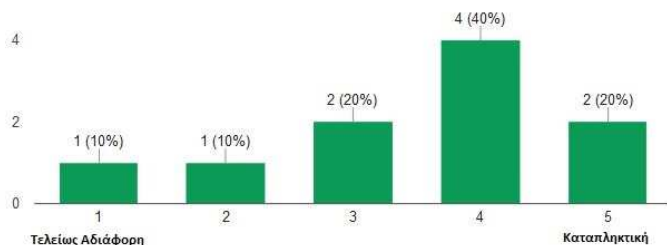
να κάνει πράγματα για μένα με σκοπό να με βοηθήσει
ΝΑ ΜΕ ΒΟΗΘΑ ΝΑ ΚΑΝΩ ΣΥΓΚΕΚΡΙΜΕΝΕΣ ΕΝΕΡΓΕΙΕΣ
ευκολία χρήσης
Να πραγματοποιεί ενέργειες βάση του προφίλ μου (π.χ. ενδιαφέροντα, ακόλουθους)
Αυτοματοποίηση καθημερινών μου ενεργειών στο twitter.
να δείχνει οτι μπορεί να με ενδιαφέρει
Να κάνει tweet, να παίρνει tweets από λογαριασμούς που ακολουθά κάποιος
ease to use, friendly
Να μπορεί να εκτελεί ενέργειες ανάλογα με τα διαφορετικές προτιμήσεις των χρηστών

Εικόνα 5.5 : Ερώτηση 5

Γενικά, η ιδέα να χρησιμοποιήσουν έναν έξυπνο βοηθό για το Twitter άρεσε στους περισσότερους εθελοντές (60%), όπως φαίνεται στην εικόνα 5.6

Πως σας φαίνεται η ιδέα να χρησιμοποιήσετε έναν έξυπνο ψηφιακό βοηθό για το Twitter?

10 απαντήσεις



Εικόνα 5.6 : Ερώτηση 6

Οι κατηγορίες των χρηστών που θα μπορούσαν να ωφεληθούν από έναν έξυπνο βοηθό για το Twitter σύμφωνα με τους εθελοντές παρουσιάζονται στην εικόνα 5.7

Ποιούς ή ποιές κατηγορίες χρηστών πιστεύετε θα βοηθούσε ένας έξυπνος βοηθός για το Twitter?

8 απαντήσεις

advance users
συχνοί χρήστες
Ανθρώπους που χρησιμοποιούν το twitter σαν εργαλείο δουλειάς (π.χ. γραμματέας, προώθηση προϊόντων).
Επιχειρήσεις, Ηθοποιούς, Τραγουδιστές, Καθημερινούς Χρήστες
αυτούς που δεν έχουν αρκετό ελεύθερο χρόνο για να ασχοληθούν με όλα τα tweets στο twitter
Επιχειρηματίες
ολους τοys χρήstes p εκoun syxni epafi me to Twitter
Χρήστες με μεγάλο αριθμό followers

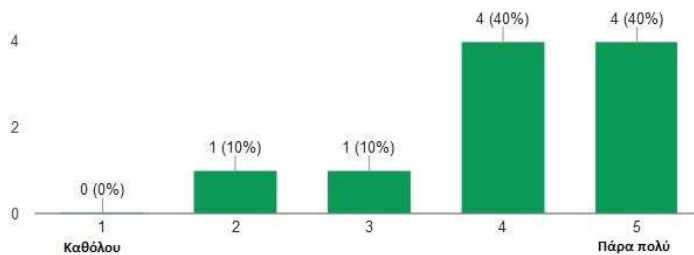
Εικόνα 5.7 : Ερώτηση 7

Ακολουθούν ερωτήσεις που αφορούν καθαρά το σύστημα Twitter Assistant που αναπτύχθηκε

Στην ερώτηση για την ευκολία σύνδεσης στον Twitter Assistant οι χρήστες του συστήματος δήλωσαν συνολικά ικανοποιημένοι (80%) , όπως φαίνεται στην Εικόνα 5.8

Πόσο εύκολο είναι να συνδεθείτε στο σύστημα του Twitter Assistant?

10 απαντήσεις

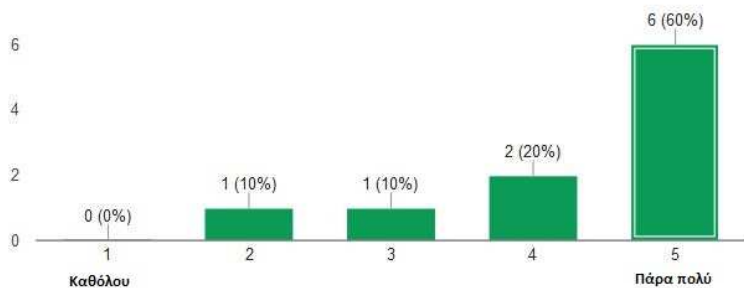


Εικόνα 5.8 : Ερώτηση 8

Για τις οδηγίες που υπάρχουν διαθέσιμες στην web διεπαφή του χρήστη, τόσο για τη σύνδεση στον Twitter Assistant όσο και για τη χρήση του οι χρήστες δήλωσαν ικανοποιημένοι κατά 80% και 90% αντίστοιχα, όπως φαίνεται στις εικόνες 5.9-5.10

Πόσο βοηθητικές είναι οι οδηγίες για την απόκτηση των κλειδιών που απαιτούνται για την εισαγωγή στο σύστημα?

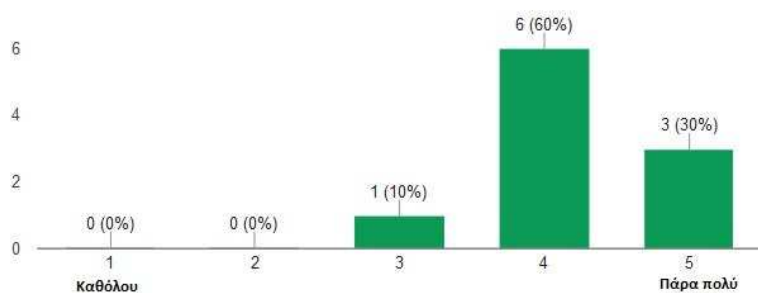
10 απαντήσεις



Εικόνα 5.9 : Ερώτηση 9

Πόσο βοηθητικές είναι οι οδηγίες για τη χρήση του συστήματος Twitter Assistant?

10 απαντήσεις

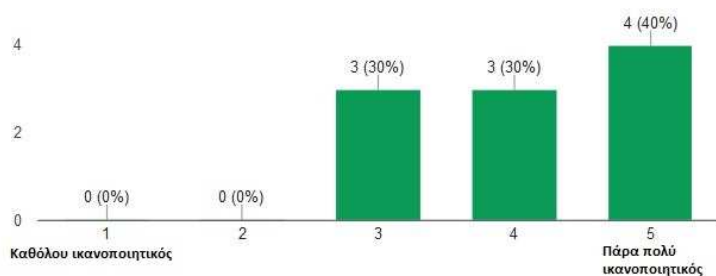


Εικόνα 5.10 : Ερώτηση 10

Για το αισθητικό αποτέλεσμα του πίνακα που εμφανίζεται στο Twitter Wall, οι χρήστες βαθμολόγησαν θετικά σε ποσοστό 70%, όπως φαίνεται στην εικόνα 5.11

Πως βρίσκεται την παρουσίαση του πίνακα Twitter Wall?

10 απαντήσεις

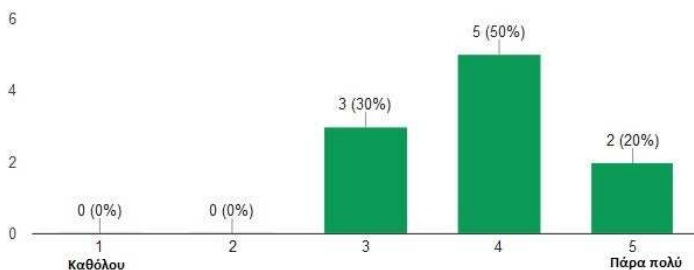


Εικόνα 5.11 : Ερώτηση 11

Οι χρήστες βρήκαν επίσης κατανοητές (ποσοστό 70%) τις πληροφορίες του συστήματος που εμφανίζονται στο πεδίο "System State" (ώρα, ημέρα, ημερομηνία, τοποθεσία), όπως φαίνεται στην εικόνα 5.12

Είναι κατανοητές οι πληροφορίες του συστήματος (System State)?

10 απαντήσεις

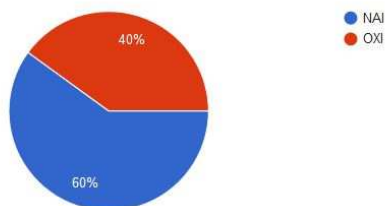


Εικόνα 5.12 : Ερώτηση 12

Όσον αφορά τη λειτουργικότητα του πεδίου "User Settings" οι εθελοντές τη βρήκαν χρήσιμη σε ποσοστό 66,6% σύμφωνα με τις απαντήσεις που έδωσαν και παρουσιάζονται στις εικόνες 5.13-5.14

Χρησιμοποίησα τις δυνατότητες που παρέχει το πεδίο "User Settings"?

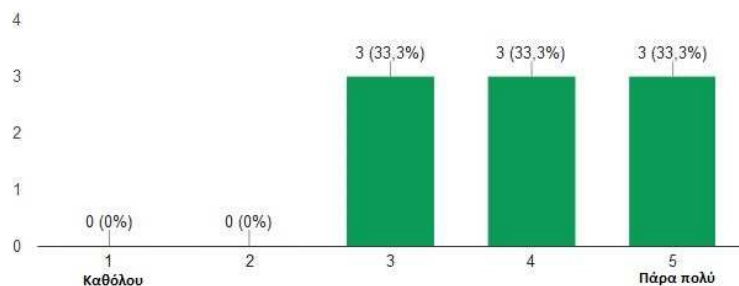
10 απαντήσεις



Εικόνα 5.13 : Ερώτηση 13

Βρίσκετε χρήσιμη τη λειτουργικότητα των επιλογών "User Settings"?

9 απαντήσεις

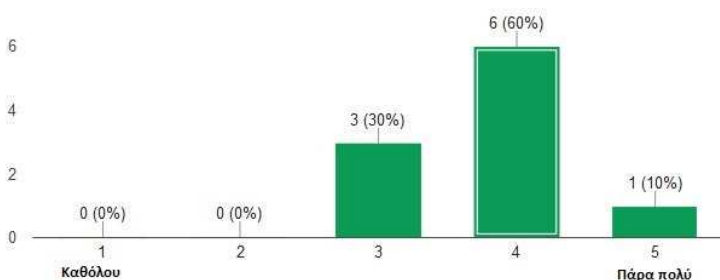


Εικόνα 5.14 : Ερώτηση 14

Οι εθελοντές σε ποσοστό 70% βρήκαν εύκολο τον τρόπο με τον οποίο συμπληρώνουν το προφίλ τους στο πεδίο "User Profile" (Εικόνα 5.15). Από την άλλη, για τη συμπλήρωση των λογικών κανόνων στο πεδίο "User Policy" ένα ποσοστό 60% τη χαρακτήρισε ως μέτριας δυσκολίας, φανερώνοντας ενδεχομένως ένα σημείο που χρειάζεται βελτίωση (Εικόνα 5.16)

Πόσο εύκολο είναι να δηλώσετε εντολές στο "User Profile"?

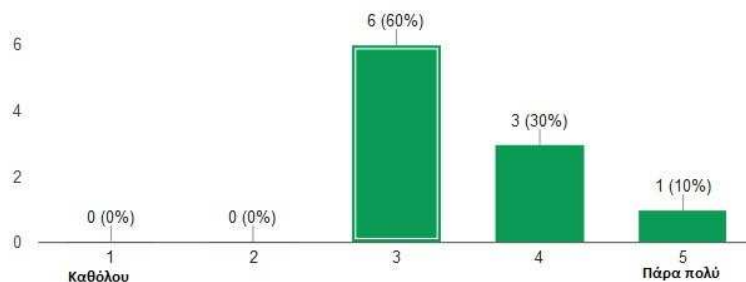
10 απαντήσεις



Εικόνα 5.15: Ερώτηση 15

Πόσο εύκολο είναι να δηλώσετε κανόνες στο "User Policy"?

10 απαντήσεις

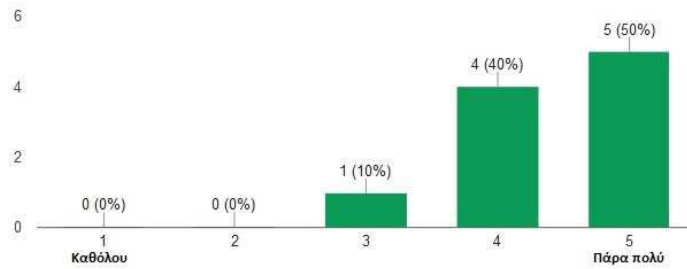


Εικόνα 5.16: Ερώτηση 16

Η λειτουργικότητα των κουμπιών του web interface του Twitter Assistant ήταν κατανοητή σε ποσοστό 90% (εικόνα 5.17), ενώ τα αποτελέσματα που δείχνει ο έξυπνος βοηθός στο πεδίο "Assistant Wall" είναι κατανοητός σε ποσοστό 80% (εικόνα 5.18)

Είναι κατανοητή η λειτουργικότητα των κουμπιών ("Save Profile and Policy", "Submit Query" και "Execute Actions")?

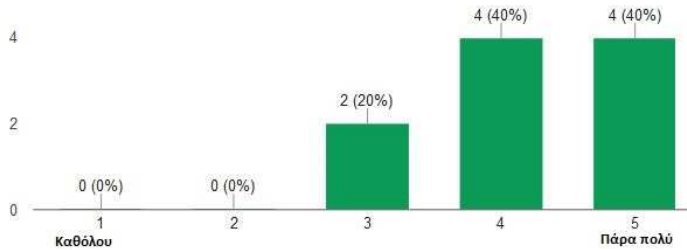
10 απαντήσεις



Εικόνα 5.17: Ερώτηση 17

Είναι κατανοητά τα αποτελέσματα των ενεργειών του βοηθού στο πεδίο Assistant Wall?

10 απαντήσεις

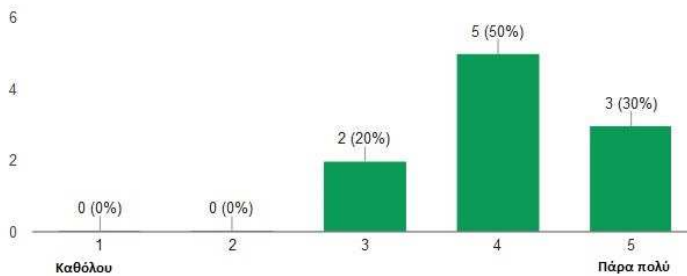


Εικόνα 5.18: Ερώτηση 18

Επίσης για τα μηνύματα που εμφανίζονται στο πεδίο "Execution Logs" κατά τη χρήση του Twitter Assistant οι εθελοντές απάντησαν πως είναι χρήσιμα σε ποσοστό 80%, όπως φαίνεται στην εικόνα 5.19

Είναι χρήσιμα τα logs που λάβατε στο πεδίο "Execution Logs"

10 απαντήσεις

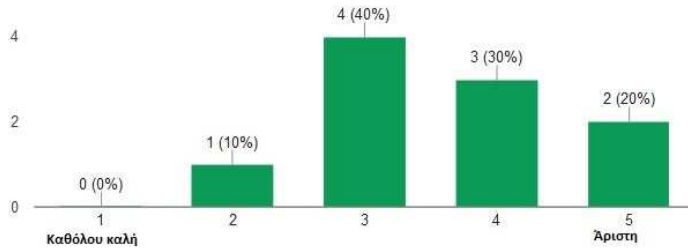


Εικόνα 5.19: Ερώτηση 19

Γενικά ως προς το αισθητικό αποτέλεσμα ποσοστό 50% των χρηστών δηλώνει συνολικά ικανοποιημένο, ενώ η πλειοψηφία 40% βαθμολόγησε με 3 σε κλίμακα 1-5 (εικόνα 5.20). Το γραφικό περιβάλλον κρίνεται αποδεκτό με περιθώρια για βελτίωση.

Πώς αξιολογείτε την εφαρμογή ως προς το αισθητικό αποτέλεσμα?

10 απαντήσεις

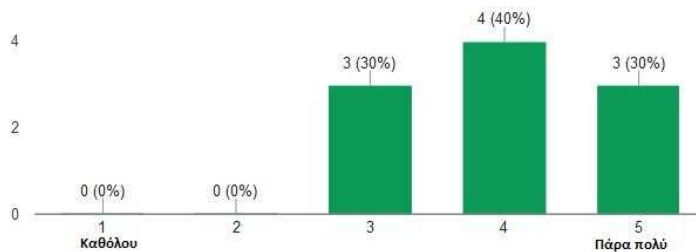


Εικόνα 5.20: Ερώτηση 20

Οι χρήστες του Twitter Assistant βρήκαν χρήσιμη την εφαρμογή σε ποσοστό 70% (εικόνα 5.21), ενώ το 60% θα το πρότεινε και σε άλλα άτομα για να το χρησιμοποιήσουν (εικόνα 5.22)

Πιστεύετε ότι η εφαρμογή είναι χρήσιμη?

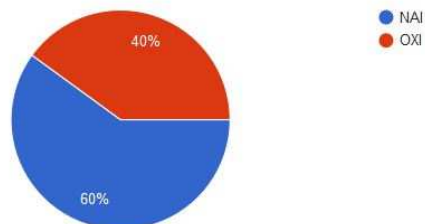
10 απαντήσεις



Εικόνα 5.21: Ερώτηση 21

Θα συστήνατε τον Twitter Assistant στους φίλους σας?

10 απαντήσεις



Εικόνα 5.22: Ερώτηση 22

Τέλος, οι εθελοντές έδωσαν σχόλια για τα στοιχεία που τους άρεσαν στο σύστημα, για αυτά που δεν τους άρεσαν και για πιθανές προσθήκες-επεκτάσεις στο σύστημα, ώστε να γίνει αποτελεσματικότερο (εικόνες 5.23-5.25).

Ως θετικά στοιχεία αναφέρονται η αυτοματοποίηση που προσφέρει ο βοηθός, τα βοηθητικά μηνύματα που εμφανίζονται στο πεδίο “Execution Logs” και το γεγονός ότι ο χρήστης έχει όλη την πληροφορία που χρειάζεται σε μία μόνο ιστοσελίδα.

Από την άλλη, οι σημαντικότερες δυσκολίες αναφέρονται στον τρόπο δημιουργίας των λογικών κανόνων, απ’ όπου προκύπτουν και οι προτάσεις για βελτίωση του έξυπνου βοηθού.

Τι σας άρεσε περισσότερο στο σύστημα του Twitter Assistant?

7 απαντήσεις

execution logs
αυτοματοποιημένη εκτέλεση εντολών
Η χρησιμότητά του.
Μια σελίδα μόνο
Το σύστημα κανόνων με το οποίο θέτεις κανόνες και τις εκτελά στο Twitter
Η αυτοματοποίηση κάποιων ενεργειών / λειτουργιών του Twitter
Η δυνατότητα αυτόματης αποστολή tweet

Εικόνα 5.23: Ερώτηση 23

Τι δεν σας άρεσε ή σας δυσκόλεψε στο σύστημα του Twitter Assistant?

8 απαντήσεις

too much complicate
ΜΕ ΔΥΣΚΟΛΕΨΕ Ο ΤΡΟΠΟΣ ΓΡΑΦΗΣ ΤΩΝ ΕΝΤΟΛΩΝ ΓΙΑ ΕΚΤΕΛΕΣΗ
interface
Ο ορισμός των κανόνων.
Ότι δεν υπάρχει μήνυμα εάν έχει κάποιο λάθος ή κάποιο πρόβλημα (error) έχει συμβεί.
Να γράφεις τους κανόνες
Τα πολύπλοκα σενάρια
Η έγγραφη στην υπηρεσία απαιτεί Consumer Key, Access Token κλπ. Η δήλωση κανόνων User Policy, User Settings, θα μπορούσε να είναι πιο φιλική προς τον χρήστη. Κάποιοι απλοί κανόνες θα μπορούσαν να δημιουργηθούν μέσα από ένα μενού επιλογών. Η υφισταμένη φόρμα δήλωσης κανόνων θα μπορούσε να εμφανίζεται στον χρήστη, στην περίπτωση που θα επιθυμούσε να γράψει ένα πιο εξειδικευμένο κανόνα που δεν θα μπορεί να δηλωθεί από το μενού.

Εικόνα 5.24: Ερώτηση 24

Τι θα θέλατε να προσθέσετε στο σύστημα του Twitter Assistant?

7 απαντήσεις

none
υπόδειξη λαθών στο συντακτικό
Σύστημα αναγνώρισης λαθών στον ορισμό κανόνων.
Πιο εύκολη δημιουργία user policies, ίσως με κουμπία που όταν τα πατάω να προσθέτουν αυτόματα text στο user policy.
λιο πιο αυτοματοποιημένος τρόπος γραφής κανόνων.
Τίποτα
Κάποια φόρμα με βήματα όπου μέσα από κάποιες επιλογές του χρήστη θα μπορούσαν να δημιουργηθούν κανόνες

Εικόνα 5.25: Ερώτηση 25

Κεφάλαιο 6

Επίλογος

6.1 Σύνοψη και Συμπεράσματα

Στα πλαίσια της μεταπτυχιακής διατριβής αναπτύχθηκε ένα web σύστημα έξυπνου βοηθού (Twitter Assistant) για το μέσο κοινωνικής δικτύωσης Twitter, που εγκαταστάθηκε σε server του Ανοικτού Πανεπιστημίου Κύπρου στην παρακάτω διεύθυνση :

<http://cognition-srv1.ouc.ac.cy/~stavros.eleftherakis/>

Το 1^ο ερευνητικό ερώτημα αφορά διερεύνηση στο τι θα μπορούσε να κάνει ένας τέτοιος έξυπνος βοηθός, που να εκτελεί αυτόματα εκ μέρους του χρήστη ενέργειες σχετικές με το λογαριασμό του στο Twitter. Μετά από μελέτη της σχετικής βιβλιογραφίας και της πλατφόρμας του Twitter, ως λειτουργίες του συστήματος επιλέχτηκαν ορισμένες από τις βασικές ενέργειες του Twitter, όπως η δημοσίευση μηνύματος (tweet), η αναδημοσίευση (retweet), το “like” και η αναζήτηση tweets με διάφορα κριτήρια.

Το 2^ο ερευνητικό ερώτημα αποτελεί ουσιαστικά το σχεδιασμό και την υλοποίηση του έξυπνου βοηθού. Εμπνευστήκαμε από την ιδέα του trigger-action προγραμματισμού για τη δημιουργία σεναρίων που καταλήγουν σε ενέργειες σχετικές με το Twitter. Για να έχουμε όμως τη δυνατότητα να δημιουργήσουμε περισσότερο σύνθετα σενάρια χρησιμοποιήθηκαν επιπλέον κάποια στοιχεία που συναντώνται στα συστήματα συστάσεων βασισμένων στη γνώση (knowledge-based recommendation systems) ή αλλιώς γνωστικά συστήματα (cognitive systems). Συγκεκριμένα, συμπεριλάβαμε ως εισόδους του έξυπνου βοηθού ένα σύνολο από λογικούς κανόνες που δημιουργεί ο χρήστης (ως βάση γνώσης) και πληροφορίες που περιγράφουν τις ανάγκες, τις ιδιαιτερότητες και τις προτιμήσεις του χρήστη (ως προφίλ χρήστη).

Επίσης, χρησιμοποιήσαμε ένα εργαλείο εξαγωγής λογικών συμπερασμάτων (web service FLASH) που ανάλογα με τις εισόδους του έξυπνου βοηθού προτείνει τις κατάλληλες ενέργειες

σχετικά με το Twitter. Επιπλέον, κρίθηκε απαραίτητη η χρήση Βάσης Δεδομένων που περιέχει πληροφορίες για τον χρήστη και η σύνδεση με το Twitter REST API μέσω μιας βιβλιοθήκης ανοικτού κώδικα που επιλέχτηκε (codebird-php) για το σκοπό αυτό.

Για την υλοποίηση του έξυπνου βοηθού αναπτύχθηκε μια web εφαρμογή (Twitter Assistant), που αποτελεί τη γραφική διεπαφή του χρήστη με τα παραπάνω υποσυστήματα (Βάση Δεδομένων, FLASH, Twitter API) με χρήση των γλωσσών και μεθόδων προγραμματισμού PHP, HTML, CSS, JavaScript, AJAX και MySQL.

Το 3^ο ερευνητικό ερώτημα αποτελεί την πειραματική αξιολόγηση του Twitter Assistant. Από αυτή προκύπτει πως με το σύστημα που αναπτύχθηκε μπορούν να υλοποιηθούν αρκετά πολύπλοκα σενάρια που να εξυπηρετούν τις ιδιαίτερες ανάγκες, επιλογές και ενδιαφέροντα του κάθε χρήστη. Ο έξυπνος βοηθός μπορεί να χρησιμοποιηθεί τόσο για διαφημιστικούς όσο και για προσωπικούς λόγους.

Το σύστημα αξιολογήθηκε επίσης από μια ομάδα μεταπτυχιακών φοιτητών του μαθήματος «Γνωστικός Προγραμματισμός» του Πανεπιστημίου Κύπρου. Από την ανάλυση των αποτελεσμάτων στο ερωτηματολόγιο που τους δόθηκε προκύπτει πως το σύστημα του Twitter Assistant κρίνεται χρήσιμο, λειτουργικό και με κατανοητές οδηγίες σύνδεσης και χρήσης. Ως προς το αισθητικό αποτέλεσμα κρίνεται ικανοποιητικό με περιθώρια βελτίωσης.

Ένα σημείο που πρέπει να προσεχτεί, είναι πως για να μπορέσει ένας απλός χρήστης να δημιουργήσει με άνεση τους δικούς του λογικούς κανόνες θα πρέπει να αφιερώσει σημαντικό χρόνο για να εξοικειωθεί με το σύστημα. Οι προτάσεις για βελτίωση του συστήματος που δίνονται στην επόμενη ενότητα αφορούν κυρίως τη διαπίστωση αυτή.

6.2 Μελλοντικές Επεκτάσεις

Το σύστημα Twitter Assistant που αναπτύχθηκε αποτελεί ένα καλό παράδειγμα δημιουργίας και χρήσης έξυπνου βοηθού για το μέσο κοινωνικής δικτύωσης Twitter, αφού ανταποκρίνεται ικανοποιητικά στις βασικές λειτουργίες που θέλαμε να επιτύχουμε.

Ωστόσο, όπως προκύπτει και από τα σχόλια των χρηστών που το δοκίμασαν, υπάρχουν περιθώρια για βελτίωση. Οι σημαντικότερες ιδέες και προτάσεις για βελτίωση του έξυπνου βοηθού που αναπτύχθηκε είναι οι εξής :

- Προσθήκη συντακτικού ελέγχου των λογικών κανόνων που δημιουργεί ένας χρήστης στο πεδίο 'User Policy' και κατάλληλη ενημέρωση του σε περίπτωση λάθους, ώστε να τους διορθώνει ανάλογα.
- Βελτίωση του σχεδιασμού της δημιουργίας λογικών κανόνων, ώστε να είναι περισσότερο φιλικός προς το χρήστη. Για παράδειγμα θα μπορούσε κάποιος να προσθέτει κατηγορήματα στους λογικούς κανόνες επιλέγοντας κουμπιά ή στοιχεία από κάποια λίστα , drop-down menu κτλ.
- Μια άλλη πρόταση για την ευκολότερη δημιουργία λογικών κανόνων είναι η προσθήκη στο υπάρχον σύστημα κάποιου ειδικού πακέτου λογισμικού που να μεταφράζει τη φυσική γλώσσα σε σύνταξη Prolog, προτού στείλουμε την πληροφορία αυτή στο web service FLASH.
- Προσθήκη επιπλέον λειτουργιών-ενεργειών σχετικά με το Twitter (συστάσεις για επιλογή followers, συστάσεις για παρακολούθηση επιλεγμένων θεμάτων, εμφάνιση σημαντικότερων tweets , απομάκρυνση υβριστικών λέξεων ή φράσεων από τα tweets)
- Σύνδεση έξυπνου βοηθού και με διαφορετικά μέσα κοινωνικής δικτύωσης, πχ. Facebook από τον ίδιο λογαριασμό του έξυπνου βοηθού.
- Σύνδεση έξυπνου βοηθού με επιπλέον web services, όπως για παράδειγμα email, ημερολόγιο χρήστη κ.ά

Βιβλιογραφία

- [1] Siri [Online] Available: <https://en.wikipedia.org/wiki/Siri> [03/06/2017].
- [2] Siri [Online] Available: <http://www.apple.com/ios/siri/> [03/06/2017]
- [3] Cortana [Online] Available: [https://en.wikipedia.org/wiki/Cortana_\(software\)](https://en.wikipedia.org/wiki/Cortana_(software)) [03/06/2017]
- [4] Cortana, Microsoft Support [Online] Available: <https://support.microsoft.com/el-gr/help/17214/windows-10-what-is> [03/06/2017]
- [5] Google Assistant [Online] Available: https://en.wikipedia.org/wiki/Google_Assistant [03/06/2017]
- [6] Google Assistant [Online] Available: <https://assistant.google.com/> [03/06/2017]
- [7] Mowbray, Miranda. "Automated twitter accounts." *Twitter and Society* (2014): 183-94.
- [8] Twitter, Automation rules [Online] Available: <https://support.twitter.com/articles/76915#> [03/06/2017]
- [9] Twitter, The Twitter Rules [Online] Available: <https://support.twitter.com/articles/18311#> [03/06/2017]
- [10] Edwards, Chad, et al. "Differences in Perceptions of Communication Quality between a Twitterbot and Human Agent for Information Seeking and Learning." *Computers in Human Behavior* 65 (2016): 666-71. Print.
- [11] Edwards Ashley, and Himanshu Sahni. "A Novel Social Search Assistant for Twitter.", 2013, Print.
- [12] Nasirifard, Peyman, and Conor Hayes. "Tadvise: A Twitter Assistant Based on Twitter Lists". *International Conference on Social Informatics*. Springer , 2011. 153-160. Print.

- [13] Commun.it [Online] Available: <https://commun.it/> [03/06/2017]
- [14] Tweetify.us [Online] Available: <http://www.tweetify.us/home.php> [03/06/2017]
- [15] Otsuka, Eriko, Scott A. Wallace, and David Chiu. "Design and Evaluation of a Twitter Hashtag Recommendation System". Proceedings of the 18th International Database Engineering & Applications Symposium. ACM , 2014. 330-333. Print.
- [16] Rahman, Mahmuda, Qinyun Zhu, and Edmund Szu-Li Yu. "TRECT: A Hashtag Recommendation System for Twitter." ACM RecSys, 2013. Print.
- [17] AlMeshary, Meshary, and Abdolreza Abhari. "A Recommendation System for Twitter Users in the Same Neighborhood". Proceedings of the 16th Communications & Networking Symposium. Society for Computer Simulation International , 2013. 1. Print.
- [18] Liu, Dongxue, et al. "Twitter-Based Products/Sales Events Recommendation System.", 2014 Print.
- [19] Elmongui, Hicham G., et al. "TRUPI: Twitter Recommendation Based on Users' Personal Interests". International Conference on Intelligent Text Processing and Computational Linguistics. Springer , 2015. 272-284. Print.
- [20] Kim, Younghoon, and Kyuseok Shim. "TWILITE: A Recommendation System for Twitter using a Probabilistic Model Based on Latent Dirichlet Allocation." Information Systems 42 (2014): 59-77. Print.
- [21] Dhillon, Paramvir Singh, and Ms Kamaljit Kaur. "Latent Dirichlet Allocation Based Recommender System for Twitter Followee Recommendation.", IRB Journal, 2016. Print.
- [22] Bouraga, Sarah, et al. "Knowledge-Based Recommendation Systems: A Survey." International Journal of Intelligent Information Technologies (IJIT) 10.2 (2014): 1-19. Print.
- [23] Ur, Blase, et al. "Practical Trigger-Action Programming in the Smart Home". Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM , 2014. 803-812. Print.
- [24] IFTTT [Online] Available: <https://ifttt.com/> [03/06/2017]

- [25] Zapier.com [Online] Available: <https://zapier.com/> [03/06/2017]
- [26] Tweets [Online] Available: <https://dev.twitter.com/overview/api/tweets> [03/06/2017]
- [27] Users [Online] Available: <https://dev.twitter.com/overview/api/users> [03/06/2017]
- [28] Twitter REST APIs [Online] Available: <https://dev.twitter.com/rest/public> [03/06/2017]
- [29] REST APIs, Reference Documentation [Online]
Available: <https://dev.twitter.com/rest/reference> [03/06/2017]
- [30] REST APIs, API Rate Limits [Online] Available: <https://dev.twitter.com/rest/public/rate-limiting> [03/06/2017]
- [31] Streaming APIs [Online] Available: <https://dev.twitter.com/streaming/overview>
[03/06/2017]
- [32] Webhook APIs [Online] Available: <https://dev.twitter.com/webhooks> [03/06/2017]
- [33] Twitter Ads API [Online] Available: <https://dev.twitter.com/ads> [03/06/2017]
- [34] Twitter Libraries [Online] Available: <https://dev.twitter.com/resources/twitter-libraries>
[03/06/2017]
- [35] codebird-php library [Online] Available: <https://github.com/jublonet/codebird-php/>
[03/06/2017]
- [36] Dung, Phan Minh. "On the Acceptability of Arguments and its Fundamental Role in Nonmonotonic Reasoning, Logic Programming and n-Person Games." *Artificial Intelligence* 77.2 (1995): 321-57. Print.
- [37] Besnard, Philippe, et al. "Introduction to Structured Argumentation." *Argument & Computation* 5.1 (2014): 1-4. Print.
- [38] Michael, Loizos. "Cognitive Reasoning and Learning Mechanisms". *Proceedings of the 4th International Workshop on Artificial Intelligence and Cognition (AIC)*. 2016. Print.

[39] Argumentation Framework [Online]

Available: https://en.wikipedia.org/wiki/Argumentation_framework [03/06/2017]

Πίνακας Εικόνων

Εικόνα 2.1 :	Είσοδοι και έξοδοι ενός έξυπνου βοηθού	11
Εικόνα 2.2 :	Παραδείγματα χρήσης Siri.....	12
Εικόνα 2.3 :	Γραφικό περιβάλλον Cortana	13
Εικόνα 2.4 :	Παραδείγματα χρήσης Google Assistant.....	14
Εικόνα 2.5 :	Αυτόματη απάντηση του Recommendation System	19
Εικόνα 2.6 :	Παράδειγμα συνταγής στο IFTTT	24
Εικόνα 2.7 :	Παραδείγματα κανόνων στο IFTTT για το Twitter	25
Εικόνα 3.1 :	Το γραφικό περιβάλλον του Twitter.....	29
Πίνακας 3.2 :	Μέρος της δομής δεδομένων της οντότητας Tweet.	30
Εικόνα 3.3 :	Αρχιτεκτονική REST API	32
Εικόνα 3.4 :	Αρχιτεκτονική Streaming API.....	32
Πίνακας 3.5 :	Βιβλιοθήκες PHP για το Twitter	34
Εικόνα 3.6 :	Παράδειγμα αφηρημένης επιχειρηματολογίας	36
Πίνακας 3.7 :	Παράδειγμα χρήσης FLASH web service	39
Εικόνα 4.1 :	Αρχιτεκτονική συστήματος Twitter Assistant.....	46
Εικόνα 4.2 :	Αναλυτικότερη αναπαράσταση της Αρχιτεκτονικής συστήματος Twitter Assistant	47
Εικόνα 4.3 :	Σελίδα σύνδεσης (Login) στο σύστημα του Twitter Assistant.....	49
Εικόνα 4.4 :	Σελίδα εγγραφής (Registration) στο σύστημα του Twitter Assistant.....	50
Εικόνα 4.5 :	Πίνακας με tweets (Twitter Wall)	51
Εικόνα 4.6 :	Πληροφορίες Συστήματος (System State)	52
Εικόνα 4.7 :	Ρυθμίσεις χρήστη (User Settings)	53
Εικόνα 4.8 :	Προφίλ Χρήστη (User Profile)	54
Πίνακας 4.9 :	Ενέργειες του Twitter Assistant	57
Εικόνα 4.10 :	Κανόνες Χρήστη (User Policy)	57
Εικόνα 4.11:	Παράδειγμα πεδίου Query Results	58
Εικόνα 4.12 :	Παράδειγμα πεδίου Assistant Wall	60
Εικόνα 4.13 :	Παράδειγμα πεδίου Execution Logs	60
Εικόνα 4.14 :	Προφίλ Χρήστη (User Profile) για το σενάριο 1	61
Εικόνα 4.15 :	Κανόνες Χρήστη (User Policy) για το παράδειγμα 1	62
Εικόνα 4.16 :	Αποτελέσματα Twitter Assistant (Query Results)	63
Εικόνα 4.17 :	Κανόνες Χρήστη (User Policy) για το παράδειγμα 2	64
Εικόνα 4.18 :	Περιγραφή πίνακα assistant_users στη Βάση Δεδομένων MySQL.....	66
Εικόνα 4.19 :	Περιγραφή πίνακα tweets_pes700_user1 στη Βάση Δεδομένων MySQL	68
Εικόνα 4.20 :	Είσοδοι και έξοδοι του υποσυστήματος FLASH web service.....	69
Πίνακας 4.21 :	Mapping συναρτήσεων codebird-php και μεθόδων Twitter REST API	71
Εικόνα 4.22 :	Χρήση της συνάρτησης “account_verifyCredentials”	72
Εικόνα 4.23 :	Επιτυχής είσοδος του χρήστη στο σύστημα του Twitter Assistant	72
Πίνακας 4.24 :	Παράμετροι της μεθόδου GET statuses/home_timeline	73
Εικόνα 4.25 :	Διάγραμμα ροής της συνάρτησης getTweetsDB()	74
Εικόνα 4.26 :	Διάγραμμα ροής της εκτέλεσης ενεργειών Twitter.....	75
Πίνακας 4.27 :	Ενέργειες (ACTIONS) που χρησιμοποιούν μεθόδους του Twitter API.....	75
Πίνακας 4.28 :	Ενέργειες (ACTIONS) φιλτραρίσματος του Twitter Wall του χρήστη	76
Εικόνα 5.1 :	Ερώτηση 1.....	85

Εικόνα 5.2 : Ερώτηση 2.....	86
Εικόνα 5.3 : Ερώτηση 3.....	86
Εικόνα 5.4 : Ερώτηση 4.....	86
Εικόνα 5.5 : Ερώτηση 5.....	87
Εικόνα 5.6 : Ερώτηση 6.....	87
Εικόνα 5.7 : Ερώτηση 7.....	88
Εικόνα 5.8 : Ερώτηση 8.....	88
Εικόνα 5.9 : Ερώτηση 9.....	89
Εικόνα 5.10 : Ερώτηση 10.....	89
Εικόνα 5.11 : Ερώτηση 11.....	89
Εικόνα 5.12 : Ερώτηση 12.....	90
Εικόνα 5.13 : Ερώτηση 13.....	90
Εικόνα 5.14 : Ερώτηση 14.....	90
Εικόνα 5.15 : Ερώτηση 15.....	91
Εικόνα 5.16: Ερώτηση 16.....	91
Εικόνα 5.17: Ερώτηση 17.....	92
Εικόνα 5.18: Ερώτηση 18.....	92
Εικόνα 5.19: Ερώτηση 19.....	92
Εικόνα 5.20: Ερώτηση 20.....	93
Εικόνα 5.21: Ερώτηση 21.....	93
Εικόνα 5.22: Ερώτηση 22.....	93
Εικόνα 5.23: Ερώτηση 23.....	94
Εικόνα 5.24: Ερώτηση 24.....	94
Εικόνα 5.25: Ερώτηση 25.....	95
Εικόνα A.1 : Σελίδα Login.....	105
Εικόνα A.2 : Σελίδα Registration.....	106
Εικόνα A.3 : Επιλογή “Create New App”.....	106
Εικόνα A.4 : Create an application.....	107
Εικόνα A.5 : Επιλογή “Create my access token”.....	107

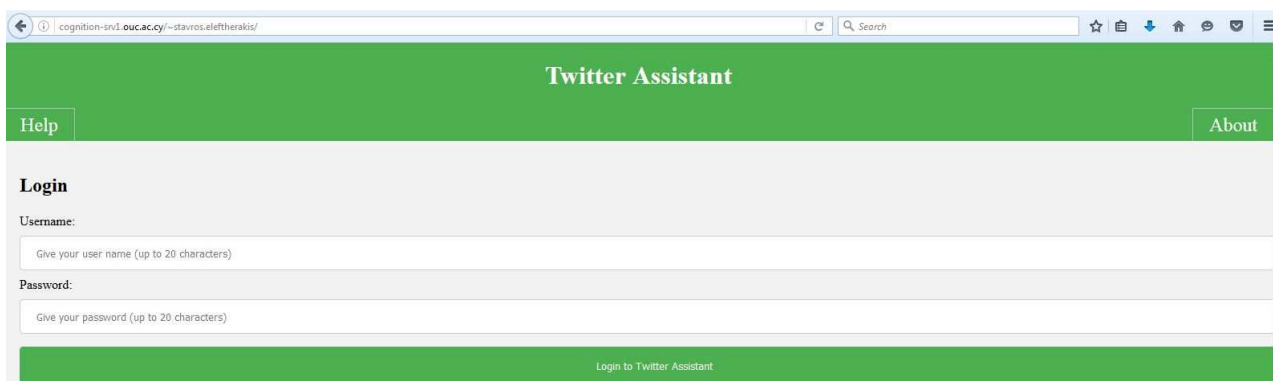
Παράρτημα Α

Οδηγίες για την εισαγωγή του χρήστη στον Twitter Assistant

Για τη σύνδεση με το σύστημα του Twitter Assistant ο χρήστης πρέπει να επισκεφτεί την παρακάτω ιστοσελίδα:

<http://cognition-srv1.ouc.ac.cy/~stavros.eleftherakis/>

Στην αρχική σελίδα ο χρήστης δίνει ένα Username και ένα password τα οποία θα χρησιμοποιεί για την είσοδό του στον Assistant, όπως φαίνεται στην παρακάτω εικόνα.



Εικόνα Α.1 : Σελίδα Login

Την πρώτη φορά που ο χρήστης θα προσπαθήσει να συνδεθεί στον Twitter Assistant καλείται να συμπληρώσει το όνομα που έχει στο λογαριασμό του στο Twitter (αναφερόμενο και ως screen name), καθώς και τα παρακάτω κλειδιά που απαιτούνται για την επικύρωση και την εξουσιοδότηση του χρήστη:

- Consumer Key
- Consumer Secret

- Access Token
- Access Token Secret

Εικόνα A.2 : Σελίδα Registration

Για την απόκτηση των κλειδιών αυτών χρειάζεται η δημιουργία μιας API εφαρμογής σύμφωνα με τα εξής βήματα:

- 1) Μετάβαση στην σελίδα <https://apps.twitter.com/>
- 2) “Sign in” με το όνομα και τον κωδικό του λογαριασμού Twitter που διαθέτει ο χρήστης.
- 3) Επιλογή “Create New App”

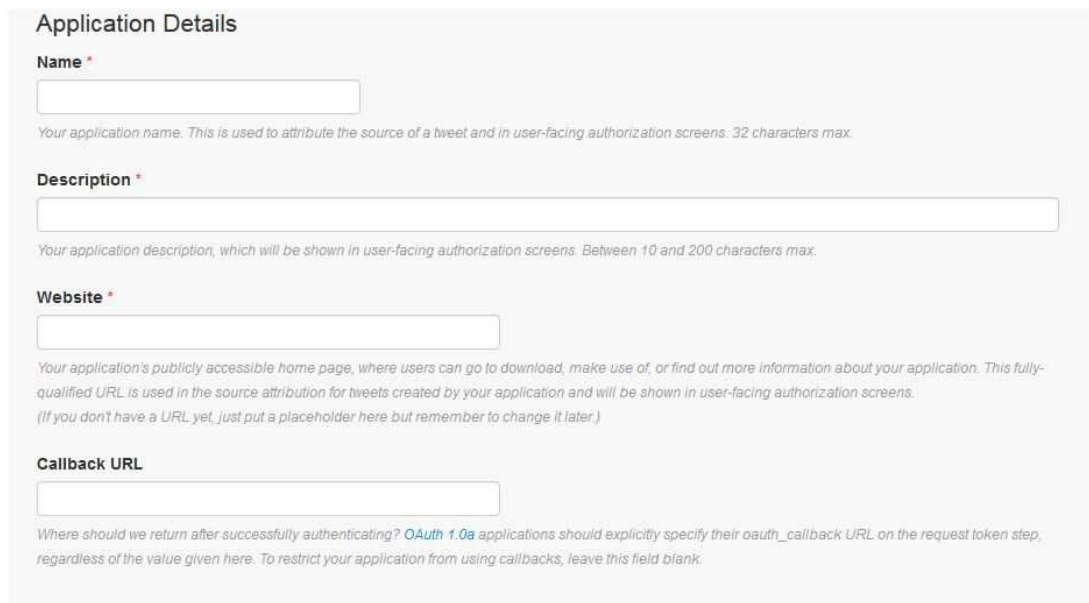


Εικόνα A.3 : Επιλογή “Create New App”

- 4) Συμπλήρωση των στοιχείων που απαιτούνται για την εφαρμογή (Name, Description, Website). Για τα πεδία Name, Description δώστε κάποια ενδεικτικά ονόματα, δεν έχει τόσο σημασία. Στο πεδίο Website αρκεί να βάλετε μια έγκυρη διεύθυνση (δηλαδή να ξεκινάει με <https://>) έστω και αν δεν υπάρχει η ιστοσελίδα. Στη συνέχεια κάντε check στο “Developer

Agreement” και “Create your Twitter Application”. Μετά το βήμα αυτό έχουν δημιουργηθεί τα στοιχεία “Consumer Key” και “Consumer Secret”.

Create an application



The screenshot shows the 'Application Details' form for creating a Twitter application. It includes four input fields: 'Name', 'Description', 'Website', and 'Callback URL'. Each field has a small text box and a descriptive note below it. The 'Name' field is limited to 32 characters. The 'Description' field is limited to 10-200 characters. The 'Website' field is for a publicly accessible home page. The 'Callback URL' field is for where to return after authentication.

Developer Agreement

Yes, I have read and agree to the [Twitter Developer Agreement](#).

Εικόνα A.4 : Create an application

5) Για την εξουσιοδότηση (authorization) της καινούριας εφαρμογής με τον λογαριασμό Twitter απαιτείται επιπλέον η δημιουργία ενός κλειδιού Access Token. Στο μενού την καινούριας εφαρμογής επιλογή του tab “Keys and Access Tokens” και “Create my access token”. Μετά το βήμα αυτό έχουν δημιουργηθεί τα στοιχεία “Access Token” και “Access Token Secret”.

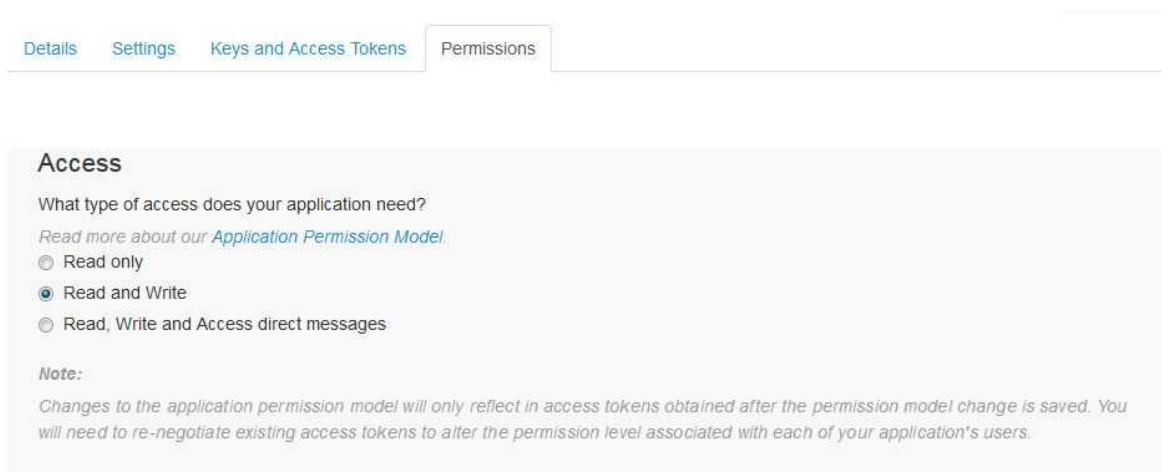


The screenshot shows a button labeled 'Create my access token' under the heading 'Token Actions'.

Εικόνα A.5 : Επιλογή “Create my access token”

6) Τα Tokens επιτρέπουν την API εφαρμογή να διαβάζει πληροφορίες από το Twitter (εύρεση tweets, lists κλπ). Αλλάζουμε την επιλογή για δικαιώματα σε “Read and Write”,

επιλέγοντας το Tab “Permissions”, για να μπορεί η εφαρμογή να γράφει και στο Twitter (δημιουργία tweets, retweet, like κλπ).



Εικόνα Α.6 : Επιλογή δικαιωμάτων “Read and Write”

7) Τα κλειδιά αυτά μετά από κάποιο έλεγχο αποθηκεύονται στην Βάση Δεδομένων του συστήματος, οπότε την επόμενη φορά που ο χρήστης θα θέλει να συνδεθεί στον Twitter Assistant πρέπει να κάνει Login μόνο με το Username και το Password που δήλωσε στην αρχική σελίδα (Login)

Παράρτημα Β

Πείραμα και ερωτηματολόγιο

B.1 Πείραμα

ΠΕΪΡΑΜΑ TWITTER ASSISTANT

Ο Twitter Assistant είναι ένα σύστημα έξυπνου ψηφιακού βοηθού με στόχο τη διευκόλυνση του χρήστη στη διαχείριση του λογαριασμού Twitter που διαθέτει. Ο Twitter Assistant είναι σε θέση να εκτελεί αυτόματα κάποιες βασικές λειτουργίες εκ μέρους του χρήστη στο Twitter (πχ. tweet, like, retweet), λαμβάνοντας υπόψιν μια σειρά από επιλογές και ενδιαφέροντα του χρήστη.

Μέρος της διπλωματικής διατριβής μου αποτελεί και η πειραματική αξιολόγηση του συστήματος που αναπτύχθηκε. Για το σκοπό αυτό δημιουργήθηκε το παρόν κείμενο, όπου οι συμμετέχοντες καλούνται να χρησιμοποιήσουν τον Twitter Assistant για να ορίσουν κάποιους λογικούς κανόνες που να εξυπηρετούν τους χρήστες στο τι θέλουν να κάνει ο Twitter Assistant γι' αυτούς.

Το πείραμα χωρίζεται σε τρία μέρη Α, Β και Γ.

Στο Α' μέρος ζητείται από τους συμμετέχοντες να γράψουν απλούς κανόνες, όπου θα οδηγήσουν τον Twitter Assistant να εκτελέσει κάποιες συγκεκριμένες ενέργειες.

Στο Β' μέρος οι συμμετέχοντες καλούνται να μπου στη θέση πιθανών χρηστών που θέλουν να υλοποιήσουν συγκεκριμένα σενάρια. Οι συμμετέχοντες θα προσπαθήσουν να καταγράψουν το προφίλ του χρήστη και κάποιους λογικούς κανόνες, σύμφωνα με το σενάριο, όπου θα οδηγήσουν τον Twitter Assistant να εκτελέσει ενέργειες που αναφέρονται σε αυτό. Κατόπιν θα προσπαθήσουν να προσθέσουν κάποιον κανόνα που θα αυξήσει την πολυπλοκότητα του σεναρίου.

Στο Γ' μέρος ζητείται από τους συμμετέχοντες να σκεφτούν ένα δικό τους σενάριο, όπου θα μπορούσαν να εφαρμόσουν με τον Twitter Assistant. Θα προσπαθήσουν με χρήση κοινής λογικής να κωδικοποιήσουν το σενάριο γράφοντας τους δικούς τους κανόνες και να δοκιμάσουν αν ο Twitter Assistant ανταποκρίνεται σωστά στις απαιτήσεις του σεναρίου.

Στη συνέχεια, οι συμμετέχοντες καλούνται να συμπληρώσουν το Ερωτηματολόγιο Πειράματος, όπου ουσιαστικά καταγράφουν τις απαντήσεις που έδωσαν στα διάφορα μέρη του πειράματος.

Τέλος, οι συμμετέχοντες καλούνται να συμπληρώσουν ένα γενικό ερωτηματολόγιο για την αξιολόγηση του συστήματος που αναπτύχθηκε, που μπορούν να βρουν στο παρακάτω link:

<https://docs.google.com/forms/d/e/1FAIpQLSfg7dPT-Gh7b2aitnajWwZ8MLst-6b-2vImBXwMmbeWgeU2UA/viewform>

Τα ερωτηματολόγια είναι ανώνυμα και όλες οι πληροφορίες που συλλέγονται θα χρησιμοποιηθούν ως αυστηρώς εμπιστευτικές και αποκλειστικά για τις ανάγκες της έρευνας που αφορά την μεταπτυχιακή διατριβή «Ανάπτυξη έξυπνου βοηθού για το Twitter (Twitter Assistant)»

Σας ευχαριστώ εκ των προτέρων για το χρόνο σας.

Ελευθεράκης Σταύρος

stavros.eleftherakis@st.ouc.ac.cy

Για την συμμετοχή στο πείραμα θα χρειαστούν τα παρακάτω :

- Δημιουργία ενός καινούριου δοκιμαστικού λογαριασμού στο Twitter

<https://twitter.com/signup>

- Στο Twitter κάντε follow τους χρήστες PES700_t , PES700_1 , PES700_2 , shop_pes
- Αποκτήστε τους κωδικούς που απαιτούνται για τη χρήση του συστήματος Twitter Assistant σύμφωνα με τις οδηγίες:

http://cognition-srv1.ouc.ac.cy/~stavros.eleftherakis/create_account.docx

- Διαβάστε τις οδηγίες για τη χρήση του Twitter Assistant:

http://cognition-srv1.ouc.ac.cy/~stavros.eleftherakis/help_page.docx

ΠΕΙΡΑΜΑ Α

Προσπαθήστε να γράψετε απλούς κανόνες στο πεδίο “User Policy”, ώστε ο Twitter Assistant να εκτελέσει τις ακόλουθες ενέργειες εκ μέρους σας. Κατόπιν συμπληρώστε το ερωτηματολόγιο (Ερωτήσεις A0-A4)

A1. Στείλτε ένα οποιοδήποτε tweet με κάποια συνθήκη του κατηγορήματος `sys_time`

π.χ. Αν η ώρα είναι μεταξύ 1μμ και 2μμ στείλτε tweet με το μήνυμα 'time for lunch'

A2. Στείλτε ένα οποιοδήποτε tweet με κάποια συνθήκη του κατηγορήματος `sys_day`, που να δηλώνει μια εβδομαδιαία δραστηριότητά σας

π.χ. κάθε Σάββατο στείλτε tweet με το μήνυμα 'today will go cinema'

A3. Βρείτε όλα τα tweets που περιέχουν τη λέξη 'OUC' και εμφανίστε τα στον Assistant Wall

A4. Κάντε like σε όλα τα tweets που περιέχουν τη λέξη 'OUC'

ΠΕΙΡΑΜΑ Β

Παρακάτω δίνονται κάποια σενάρια πιθανών χρηστών B1-B3. Προσπαθήστε να γράψετε κάποιους λογικούς κανόνες (στο πεδίο “User Policy”) και πιθανά χαρακτηριστικά του χρήστη (στο πεδίο “User Profile”) σύμφωνα με το σενάριο, όπου θα οδηγήσουν τον Twitter Assistant να εκτελέσει τις ενέργειες που αναφέρονται στο σενάριο. Κατόπιν συμπληρώστε το ερωτηματολόγιο (Ερωτήσεις B1-B3)

Παράδειγμα

Υποθέστε ότι θέλετε να στέλνετε κάθε μέρα αυτόματα ένα χαιρετιστήριο tweet ανάλογα με την ώρα, πχ. 'Good morning!' τις ώρες 6πμ-12, 'Good evening!' τις ώρες 12-9μμ και 'Good night!' τις ώρες 9μμ-12 βράδυ. Θέλουμε να εξασφαλίσουμε επίσης ότι το tweet αυτό θα στέλνεται μια

φορά τη μέρα και δεν θα επαναλαμβάνεται κάθε φορά που τρέχουμε τον Twitter Assistant (πχ. δεν έχει καμία λογική να στέλνουμε tweet με καλημέρα κάθε 5 λεπτά!)

Οι κανόνες που υλοποιούν το σενάριο αυτό είναι οι παρακάτω:

r1 :: sys_time(X), #(X>=0600), #(X<1200) implies to_post('Good morning!').

r2 :: sys_time(X), #(X>=1200), #(X<2100) implies to_post('Good evening!').

r3 :: sys_time(X), #(X>=2100) implies to_post('Good night!').

r4 :: to_post(TEXT) implies do_post(TEXT)

r5 :: message('PES700_0',ID,TEXT,DATE), sys_date(DATE), to_post(TEXT)
implies -do_post(TEXT).

r5>>r4.

Η εντολή που λέει στον Twitter Assistant να κάνει tweet είναι η “do_post()”. Χρησιμοποιήσαμε την ενδιαμέση έννοια “to_post()” ως πρόθεση να στείλουμε κάποιο tweet.

Οι κανόνες r1-r3 δηλώνουν την πρόθεση να στείλουμε tweet με το συγκεκριμένο κείμενο TEXT ανάλογα με την ώρα.

Ο κανόνας r4 δηλώνει ότι αν έχεις πρόθεση να στείλεις tweet με το κείμενο TEXT τότε και να το στείλεις. Ο κανόνας αυτός θα ενεργοποιηθεί την πρώτη φορά που θα στείλουμε το σχετικό tweet.

Ο κανόνας r5 δηλώνει ότι αν έχουμε πρόθεση να στείλουμε ένα tweet με το κείμενο TEXT to_post(TEXT), αλλά ήδη υπάρχει ένα tweet message(...) που έχουμε στείλει εμείς (PES700_0), με το ίδιο κείμενο TEXT, την ίδια μέρα DATE τότε να μην το στείλεις. Επειδή οι κανόνες r4,r5 έρχονται σε αντίφαση μεταξύ τους, δίνουμε προτεραιότητα στον r5 (r5>>r4) για να πετύχουμε το σκοπό του σεναρίου.

B1. Υποθέστε ότι είστε ο διαχειριστής σε κατάσταση ηλεκτρονικών και θέλετε να στέλνετε το παρακάτω tweet διαφήμιση 'Hello!Check our shop for electronics!' σε όποιο χρήστη

φαίνεται να ψάχνει ή να ζητάει κάποια ηλεκτρονική συσκευή (πχ. laptops, smartphones). Εξαιρούνται τα tweets που προέρχονται από ανταγωνίστριες εταιρίες. (shop_pes)

B2. Υποθέστε ότι είστε ένας εργαζόμενος σε εταιρία fast food στην Αθήνα, με ωράριο 9πμ-5μμ και είστε vegetarian. Δεν θέλετε να λαμβάνεται μηνύματα που αφορούν διαφημίσεις για fast food, εκτός και εάν προέρχονται από το αφεντικό σας (PES700_2) και είτε είναι σε ώρες εργασίας ή απευθύνει κάποιο ερώτημα σε εσάς (ακόμα εκτός ωρών εργασίας, αλλά μόνο αν βρίσκεστε σε αποστολή από το αφεντικό σας).

B3. Υποθέστε ότι είστε ένας αθλητής μιας ομάδας πχ. 'Celtics' και τις ημέρες που η ομάδα σας έχει αγώνες δεν πρέπει να βλέπετε tweets σχετικά με την ομάδα σας (για να μην επηρεαστείτε), εκτός αν προέρχονται από τον προπονητή σας (PES700_2). Επίσης θέλετε να κάνετε like στα tweets που αναφέρονται σε νίκη της ομάδας σας, εκτός από αυτά του προπονητή σας (για να μη θεωρηθεί ότι θέλετε να κερδίσετε την εύνοια του).

ΠΕΙΡΑΜΑ Γ

Σκεφτείτε ένα δικό σας σενάριο, όπου θα μπορούσε να εφαρμοστεί ο Twitter Assistant. Προσπαθήστε με χρήση κοινής λογικής να γράψετε τους δικούς σας κανόνες (στο πεδίο "User Policy") και πιθανά χαρακτηριστικά του χρήστη (στο πεδίο "User Profile") που να υλοποιούν το σενάριό σας. Δοκιμάστε αν ο Twitter Assistant ανταποκρίνεται σωστά στις απαιτήσεις του σεναρίου. Κατόπιν συμπληρώστε το ερωτηματολόγιο (Ερωτήσεις Γ1-Γ5)

Ερωτηματολόγιο Πειράματος

A0 Δώστε το όνομα του δοκιμαστικού λογαριασμού Twitter που δημιουργήσατε.

.....

A1.1 Πόσο εύκολο ήταν να φτιάξετε κανόνες και να εκτελέσετε την ενέργεια A1 ?

Καθόλου Εύκολο 1 2 3 4 5 Πάρα πολύ εύκολο

A1.2 Πόσο χρόνο σας πήρε για την ενέργεια A1?

.....

A2.1 Πόσο εύκολο ήταν να φτιάξετε κανόνες και να εκτελέσετε την ενέργεια A2 ?

Καθόλου Εύκολο 1 2 3 4 5 Πάρα πολύ εύκολο

A2.2 Πόσο χρόνο σας πήρε για την ενέργεια A2?

.....

A3.1 Πόσο εύκολο ήταν να φτιάξετε κανόνες για να εκτελέσετε την ενέργεια A3 ?

Καθόλου Εύκολο 1 2 3 4 5 Πάρα πολύ εύκολο

A3.2 Πόσο χρόνο σας πήρε για την ενέργεια A3?

.....

A4.1 Πόσο εύκολο ήταν να φτιάξετε κανόνες για να εκτελέσετε την ενέργεια A4 ?

Καθόλου Εύκολο 1 2 3 4 5 Πάρα πολύ εύκολο

A4.2 Πόσο χρόνο σας πήρε για την ενέργεια A4?

.....

B1.1 Δώστε τους κανόνες που δώσατε για το σενάριο B1 στο πεδίο “User Policy” ή/και “User Profile”

.....
.....

B1.2 Πόσο εύκολο ήταν να φτιάξετε κανόνες για να εκτελέσετε το σενάριο B1?

Καθόλου Εύκολο 1 2 3 4 5 Πάρα πολύ εύκολο

B1.3 Πόσο χρόνο σας πήρε για την δημιουργία και εκτέλεση του σεναρίου B1?

.....

B1.4 Επεκτείνετε το σενάριο B1, προσθέτοντας έναν επιπλέον κανόνα της επιλογής σας. Δώστε τον κανόνα που προσθέσατε

.....
.....

B1.5 Περιγράψτε με λόγια τι κωδικοποιεί ο παραπάνω κανόνας που προσθέσατε

.....
.....

B2.1 Δώστε τους κανόνες που δώσατε για το σενάριο B2 στο πεδίο “User Policy” ή/και “User Profile”

.....
.....

B2.2 Πόσο εύκολο ήταν να φτιάξετε κανόνες για να εκτελέσετε το σενάριο B2?

Καθόλου Εύκολο 1 2 3 4 5 Πάρα πολύ εύκολο

B2.3 Πόσο χρόνο σας πήρε για την δημιουργία και εκτέλεση του σεναρίου B2?

.....

B2.4 Επεκτείνετε το σενάριο B2, προσθέτοντας έναν επιπλέον κανόνα της επιλογής σας. Δώστε τον κανόνα που προσθέσατε

.....
.....

B2.5 Περιγράψτε με λόγια τι κωδικοποιεί ο παραπάνω κανόνας που προσθέσατε

.....
.....

B3.1 Δώστε τους κανόνες που δώσατε για το σενάριο B3 στο πεδίο “User Policy” ή/και “User Profile”

.....
.....

B3.2 Πόσο εύκολο ήταν να φτιάξετε κανόνες για να εκτελέσετε το σενάριο B3?

Καθόλου Εύκολο 1 2 3 4 5 Πάρα πολύ εύκολο

B3.3 Πόσο χρόνο σας πήρε για την δημιουργία και εκτέλεση του σεναρίου B3?

.....

B3.4 Επεκτείνετε το σενάριο B3, προσθέτοντας έναν επιπλέον κανόνα της επιλογής σας. Δώστε τον κανόνα που προσθέσατε

.....
.....

B3.5 Περιγράψτε με λόγια τι κωδικοποιεί ο παραπάνω κανόνας που προσθέσατε

.....
.....

Γ1 Σκεφτήκατε κάποιο δικό σας σενάριο Γ?

ΝΑΙ ΟΧΙ

Γ2 Περιγράψτε το σενάριο αυτό με λόγια

.....
.....

Γ3 Δώστε τους κανόνες που δώσατε για το σενάριο Γ στο πεδίο “User Policy” ή/και “User Profile”

.....
.....

Γ4 Πόσο εύκολο ήταν να φτιάξετε κανόνες για να εκτελέσετε το σενάριο Γ?

Καθόλου Εύκολο 1 2 3 4 5 Πάρα πολύ εύκολο

Γ5 Πόσο χρόνο σας πήρε για την δημιουργία και εκτέλεση του σεναρίου Γ?

.....

B.2 Ερωτηματολόγιο

Ερωτηματολόγιο Twitter Assistant

Ο Twitter Assistant είναι ένα σύστημα έξυπνου ψηφιακού βοηθού που δημιουργήθηκε στα πλαίσια της μεταπτυχιακής μου διατριβής, το οποίο μπορείτε να βρείτε στην παρακάτω διεύθυνση:

<http://cognition-srv1.ouc.ac.cy/~stavros.eleftherakis/>

Στόχος του βοηθού αυτού είναι η διευκόλυνση του χρήστη στη διαχείριση του λογαριασμού Twitter που διαθέτει. Ο χρήστης γράφοντας κάποιους λογικούς κανόνες και σε συνδυασμό με μερικές πληροφορίες του συστήματος (πχ. ώρα, ημέρα, τοποθεσία) είναι δυνατό να οδηγήσει τον βοηθό στην αυτόματη εκτέλεση βασικών ενεργειών του Twitter. Για παράδειγμα, ο Twitter Assistant είναι σε θέση να εκτελέσει αυτόματα tweet, like, retweet ή κάποια αναζήτηση με διάφορα κριτήρια εκ μέρους του χρήστη, λαμβάνοντας υπόψιν μια σειρά από επιλογές και ενδιαφέροντα του χρήστη.

Σκοπός του παρόντος ερωτηματολογίου είναι η αξιολόγηση του συστήματος που αναπτύχθηκε ως προς τη χρησιμότητα και την ευχρηστία του συστήματος, την ευκολία μάθησης και την συνολική ικανοποίηση του χρήστη που δοκίμασε τον Twitter Assistant.

Αφού εγγραφείτε στο σύστημα του Twitter Assistant και έχετε δοκιμάσει την εφαρμογή δημιουργώντας τους δικούς σας κανόνες (έχετε συμμετέχει στο σχετικό πείραμα) απαντήστε στις ερωτήσεις της έρευνας.

Το ερωτηματολόγιο είναι ανώνυμο και όλες οι πληροφορίες που συλλέγονται θα χρησιμοποιηθούν ως αυστηρώς εμπιστευτικές και αποκλειστικά για τις ανάγκες της έρευνας που αφορά την μεταπτυχιακή διατριβή «Ανάπτυξη έξυπνου βοηθού για το Twitter (Twitter Assistant)»

Σας ευχαριστώ εκ των προτέρων για το χρόνο σας.

Ελευθεράκης Σταύρος
stavros.eleftherakis@st.ouc.ac.cy

ΕΠΟΜΕΝΟ

Σελίδα 1 από 3

Μην υποβάλετε ποτέ κωδικούς πρόσβασης μέσω των Φορμών Google.

Ερωτηματολόγιο Twitter Assistant

* Απαιτείται

Γενικές Ερωτήσεις

Γενικές ερωτήσεις για τη χρήση έξυπνων ψηφιακών βοηθών και του μέσου κοινωνικής δικτύωσης Twitter

Τι σημαίνει για εσάς ένας έξυπνος ψηφιακός βοηθός?

Η απάντησή σας

Έχετε χρησιμοποιήσει ποτέ κάποιου είδους έξυπνο βοηθό? *

- ΝΑΙ
 ΟΧΙ

Ποιους έξυπνους βοηθούς έχετε χρησιμοποιήσει?

Η απάντησή σας

Έχετε ενεργό λογαριασμό στο Twitter? *

- ΝΑΙ
 ΟΧΙ

Πόσο συχνά χρησιμοποιείτε το Twitter? *

- Κάθε μέρα
 Περίπου 3 φορές/εβδομάδα
 1 φορά/εβδομάδα
 Σπανιότερα
 Καθόλου

Για ποιο λόγο χρησιμοποιείται το Twitter? *

- Επικοινωνία με φίλους
 Διασκέδαση
 Ενημέρωση
 Άλλο:

Πως φαντάζεστε έναν έξυπνο ψηφιακό βοηθό για το Twitter?
Τι χαρακτηριστικά πρέπει να έχει?

Η απάντησή σας

Έχετε χρησιμοποιήσει ποτέ κάποιον έξυπνο βοηθό για το
Twitter? *

- ΝΑΙ
 ΟΧΙ

Ποιους έξυπνους βοηθούς για το Twitter έχετε
χρησιμοποιήσει?

Η απάντησή σας

Πως σας φαίνεται η ιδέα να χρησιμοποιήσετε έναν έξυπνο
ψηφιακό βοηθό για το Twitter? *

	1	2	3	4	5	
Τελείως Αδιάφορη	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Καταπληκτική

Ποιούς ή ποιές κατηγορίες χρηστών πιστεύετε θα βοηθούσε
ένας έξυπνος βοηθός για το Twitter?

Η απάντησή σας

ΠΙΣΩ

ΕΠΙΔΜΕΝΟ

Σελίδα 2 από 3

Μην υποβάλετε ποτέ κωδικούς πρόσβασης μέσω των Φορμών Google.

Ερωτηματολόγιο Twitter Assistant

* Απαιτείται

Ερωτήσεις για το σύστημα Twitter Assistant

Ερωτήσεις για τη λειτουργικότητα, την ευχρηστία και την κατανόηση του έξυπνου ψηφιακού βοηθού Twitter Assistant

Δώστε το username που δηλώσατε για την εγγραφή σας στο σύστημα του Twitter Assistant *

Η απάντησή σας

Πόσο εύκολο είναι να συνδεθείτε στο σύστημα του Twitter Assistant? *

	1	2	3	4	5	
Καθόλου	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Πάρα πολύ

Πόσο βοηθητικές είναι οι οδηγίες για την απόκτηση των κλειδιών που απαιτούνται για την εισαγωγή στο σύστημα? *

	1	2	3	4	5	
Καθόλου	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Πάρα πολύ

Πόσο βοηθητικές είναι οι οδηγίες για τη χρήση του συστήματος Twitter Assistant? *

	1	2	3	4	5	
Καθόλου	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Πάρα πολύ

Πως βρίσκεται την παρουσίαση του πίνακα Twitter Wall? *

	1	2	3	4	5	
Καθόλου ικανοποιητικό ς	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Πάρα πολύ ικανοποιητικό ς

Είναι κατανοητές οι πληροφορίες του συστήματος (System State)? *

	1	2	3	4	5	
Καθόλου κατανοητές	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Πάρα πολύ κατανοητές

Χρησιμοποιήσατε τις δυνατότητες που παρέχει το πεδίο "User Settings"? *

ΝΑΙ

ΟΧΙ

Βρίσκετε χρήσιμη τη λειτουργικότητα των επιλογών "User Settings"?

	1	2	3	4	5	
Καθόλου χρήσιμη	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Πάρα πολύ χρήσιμη

Πόσο εύκολο είναι να δηλώσετε εντολές στο "User Profile"? *

	1	2	3	4	5	
Καθόλου εύκολο	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Πάρα πολύ εύκολο

Πόσο εύκολο είναι να δηλώσετε κανόνες στο "User Policy"? *

	1	2	3	4	5	
Καθόλου εύκολο	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Πάρα πολύ εύκολο

Είναι κατανοητή η λειτουργικότητα των κουμπιών ("Save Profile and Policy", "Submit Query" και "Execute Actions")? *

	1	2	3	4	5	
Καθόλου κατανοητή	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Πάρα πολύ κατανοητή

Είναι κατανοητά τα αποτελέσματα των ενεργειών του βοηθού στο πεδίο Assistant Wall? *

	1	2	3	4	5	
Καθόλου	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Πάρα πολύ

Είναι χρήσιμα τα logs που λάβατε στο πεδίο "Execution Logs"? *

	1	2	3	4	5	
Καθόλου χρήσιμα	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Πάρα πολύ χρήσιμα

Πώς αξιολογείτε την εφαρμογή ως προς το αισθητικό αποτέλεσμα? *

1 2 3 4 5

Καθόλου καλή Άριστη

Πιστεύετε ότι η εφαρμογή είναι χρήσιμη? *

1 2 3 4 5

Καθόλου Πάρα πολύ

Θα συστήνατε τον Twitter Assistant στους φίλους σας? *

- ΝΑΙ
- ΟΧΙ

Τι σας άρεσε περισσότερο στο σύστημα του Twitter Assistant?

Η απάντησή σας

Τι δεν σας άρεσε ή σας δυσκόλεψε στο σύστημα του Twitter Assistant?

Η απάντησή σας

Τι θα θέλατε να προσθέσετε στο σύστημα του Twitter Assistant?

Η απάντησή σας

ΠΙΣΩ

ΥΠΟΒΟΛΗ

Σελίδα 3 από 3

Μην υποβάλλετε ποτέ κωδικούς πρόσβασης μέσω των Φορμών Google.