



Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης

**Σχολή Τεχνολογικών Εφαρμογών
Τμήμα Μηχανικών Πληροφορικής**



Πτυχιακή Εργασία

**Ανάπτυξη διαδικτυακής εφαρμογής για την
διαχείριση σύνθετων σχημάτων ειδοποίησης με
ενσωμάτωση υπηρεσιών νέφους**

Ανδροκλής Γρηγορίου (ΑΜ: 3668)

Επιβλέπων Καθηγητής: Δρ. Ακουμιανάκης Δημοσθένης

Επιτροπή Αξιολόγησης:

Δρ. Ακουμιανάκης Δημοσθένης
Δρ. Βιδάκης Νικόλαος
Δρ. Παπαδάκης Νικόλαος

Ημερομηνία Παρουσίασης: 2016-09-02

*Η παρούσα πτυχιακή εργασία είναι αφιερωμένη
στην μνήμη του παππού μου Ανδροκλή.*

Ευχαριστίες

Καταρχήν, θα ήθελα να ευχαριστήσω θερμά τον επιβλέπων καθηγητή μου κ. Ακουμιανάκη Δημοσθένη, για την καθοδήγηση και την βοήθεια του στην διπλωματική μου εργασία. Στη συνέχεια, θα ήθελα να ευχαριστήσω τον Κτιστάκη Γιώργο για την πολύτιμη βοήθεια του κατά την τεκμηρίωση αυτής της εργασίας. Επίσης, θα ήθελα να ευχαριστήσω το Εργαστήριο Τεχνολογίας & Συστημάτων Λογισμικού (iSTLab) για την φιλοξενία, τον κ. Βιδάκη Νικόλαο και όλα τα παιδιά που συμμετείχαν/ουν στο εργαστήριο για την οποιαδήποτε συνεισφορά τους. Τέλος, θέλω να ευχαριστήσω τους γονείς μου για την αρχή και το τέλος όλου αυτού του ταξιδιού.

Abstract

This thesis presents the design and development of a group notification mechanism which relies on current web standards and facilitates group work. In addition to its basic functionality, the mechanism exploits multiple notification channels (i.e., email, shared calendars, etc.) and offers an API which allows third party applications and services to exploit core functions. In terms of technical approach, the application follows the mashup philosophy to support flexible management of complex notification schemes. Users access the application using their Google account and once authorized, they can create notification schemes to be posted immediately or at a designated time in the future. Recipients of notifications include individuals or groups of users as delivered through the Google Contacts API. Users can set complex criteria which will allow the system to automatically select whether or not a notification should be sent. Such rules are correlated with absolute or qualified time limits, specific users or designated groups and the users' previous records of events or activities. A special feature of the application is the ability to create and save events on selected electronic calendars (i.e. Google Calendar), thus linking group notifications with flexible and personalized time management.

Σύνοψη

Αυτή η πτυχιακή εργασία πραγματεύεται την υλοποίηση ενός μηχανισμού ομαδικών ειδοποιήσεων υιοθετώντας επίκαιρες τάσεις και πρότυπα του διαδικτύου. Πέρα της βασικής λειτουργίας του, ο μηχανισμός προσφέρει επίσης, ένα API το οποίο επιτρέπει σε ξένες εφαρμογές τρίτων να αξιοποιούν τις υπηρεσίες που προσφέρονται από το μηχανισμό ειδοποιήσεων. Η τεχνική προσέγγιση που υιοθετήθηκε βασίζεται στη φιλοσοφία των Mashups και σε συνδυασμό με κάποια από τα Google APIs, υποστηρίζεται ένας ευέλικτος τρόπος διαχείρισης σύνθετων σχημάτων ειδοποίησης. Ειδικότερα, οι χρήστες της εφαρμογής αποκτούν πρόσβαση στην υπηρεσία χρησιμοποιώντας τον Google λογαριασμό τους (Google Account) και στη συνέχεια δημιουργούν και διαχειρίζονται τις ειδοποιήσεις που επιθυμούν να αποστείλουν (είτε άμεσα είτε σε κάποια στιγμή στο μέλλον) σε επιλεγμένους χρήστες ή ομάδες χρηστών από τις επαφές του χρήστη (όπως αυτές παρέχονται από την υπηρεσία Google Contacts). Ο χρήστης μπορεί κάθε φορά να θέτει σύνθετα κριτήρια βάση των οποίων θα αποστέλλεται η ειδοποίηση ή όχι. Τα κριτήρια αυτά αφορούν απόλυτα ή σχετικά χρονικά όρια, ποιους χρήστες ή ποιες ομάδες χρηστών πιθανών να ενδιαφέρουν, καθώς και την προγενέστερη συμμετοχή αυτών σε προγραμματισμένες δραστηριότητες ή εκδηλώσεις. Ένα ιδιαίτερο χαρακτηριστικό της εφαρμογής είναι η δυνατότητα δημιουργίας και καταχώρησης γεγονότων σε επιλεγμένα ηλεκτρονικά ημερολόγια του χρήστη (π.χ. Google Calendar) που έχει ως αποτέλεσμα τη διασύνδεση των ειδοποιήσεων με την ατομική διαχείριση χρόνου.

Πίνακας Περιεχομένων

Ευχαριστίες	i
Abstract	ii
Σύνοψη	iii
Πίνακας Περιεχομένων	4
Πίνακας Εικόνων.....	6
Λίστα Πινάκων.....	7
1 Εισαγωγή.....	8
1.1 Παγκόσμιος ιστός (Web).....	8
1.2 Web Application.....	9
1.3 Στόχοι πτυχιακής εργασίας.....	11
1.4 Δομή και οργάνωση της πτυχιακής εργασίας.....	11
2 Μηχανισμοί Ενημέρωσης Ομάδων: Επισκόπηση	12
2.1 Συνεργασία υποστηριζόμενη από υπολογιστή (CSCW)	12
2.2 Κατηγορίες ενημέρωσης	13
2.2.1 Alerts	13
2.2.2 Reminders.....	13
2.2.3 Notifications - Awareness.....	14
2.3 Ενημερότητα ως λειτουργία	15
2.3.1 Γνωσιακή ενίσχυση	15
2.4 Σχεδιαστικά ζητήματα.....	16
2.4.1 Ο χρόνος των ειδοποιήσεων - Πώς να προβλέψουμε την κατάλληλη στιγμή	17
2.4.2 Οι παραλήπτες – Πως καθορίζεται η ομάδα αποδεκτών	18
2.4.3 Το είδος της ενημέρωσης που ενδείκνυται	20
2.5 Τεχνολογικά ζητήματα.....	22
2.5.1 Mashups	22
2.5.2 Design Languages για υπηρεσίες ομαδικής ενημέρωσης.....	24
2.5.2.1 Microsoft Design Language: Metro	24
2.5.2.2 IBM Design Language	24
2.5.2.3 Snow White Design Language	24
2.5.2.4 Google Material Design	25
3 Μηχανισμός ομαδικών ενημερώσεων.....	26
3.1 Ειδικοί στόχοι.....	26
3.1.1 Περιφερειακή – Δευτερεύουσα λειτουργία	27
3.2 Τεχνολογικά εργαλεία και δεσμεύσεις	29
3.2.1 UseCase Diagram.....	29
3.2.2 Sequence Diagram.....	31

3.2.3	Διάγραμμα κλάσεων	32
3.2.4	APIs	34
3.2.5	Materialize Framework	36
3.2.6	Integrated Development Environment (IDE).....	37
3.2.7	Development Frameworks	37
3.2.8	Plugins	38
4	Περίπτωση χρήσης και πιλοτική επίδειξη	39
5	Κατακλείδα.....	45
5.1	Αποτίμηση.....	45
5.2	Μελλοντική Εργασία και Επεκτάσεις	45
	Βιβλιογραφία.....	46

Πίνακας Εικόνων

Εικόνα 1: Web Evolution	8
Εικόνα 2: Semantic Web	10
Εικόνα 3: CSCW Matrix	12
Εικόνα 4: Different kinds of Awareness	15
Εικόνα 5: Notifications by hour of the day	21
Εικόνα 6: Material perspective.....	25
Εικόνα 7: From email to calendar event.....	27
Εικόνα 8: Peripheral - Secondary Operation.....	28
Εικόνα 9: UseCase Diagram	30
Εικόνα 10: Sequence Diagram	31
Εικόνα 11: UML Diagram.....	33
Εικόνα 12: Google Contacts API	34
Εικόνα 13: Code Snippet - init.js (initComponents: Contacts Autocomplete).....	35
Εικόνα 14: DataBase (Google Spreadsheet)	36
Εικόνα 15: Group Notification Mechanism - Authentication	39
Εικόνα 16: Group Notification Mechanism - Main Screen.....	40
Εικόνα 17: Group Notification Mechanism - Schemes	40
Εικόνα 18: Group Notification Mechanism - Add new Scheme.....	41
Εικόνα 19: Event Settings	41
Εικόνα 20: Code Snippet - modal-actions.js (addCard)	42
Εικόνα 21: Code Snippet - GNMServlet.java (processRequest).....	43
Εικόνα 22: Code Snippet - GoogleCalendar.java (addEvent)	44

Λίστα Πινάκων

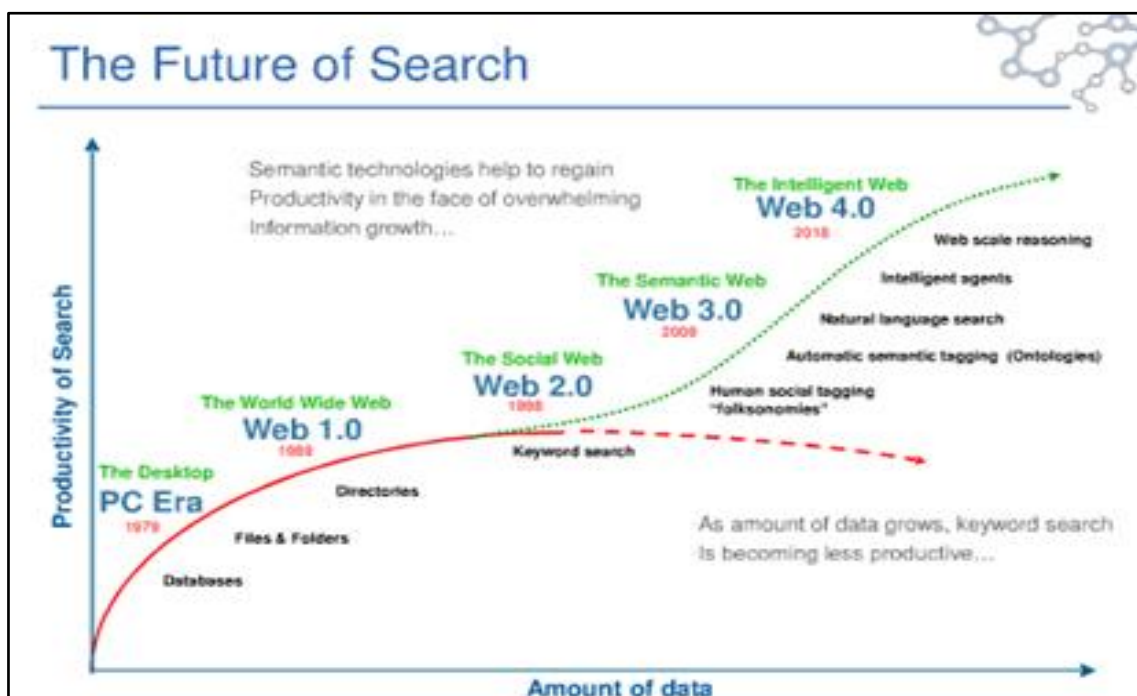
Πίνακας 1: Description of features from the NotifyMe dataset	18
Πίνακας 2: Time and place dispersion	21

1 Εισαγωγή

Ένα από τα αξιοσημείωτα τεχνολογικά επιτεύγματα στον κλάδο της Πληροφορικής τα τελευταία χρόνια είναι η έλευση και ο εξελικτικός μετασχηματισμός του παγκόσμιου ιστού. Δεδομένου ότι η παρούσα εργασία αφορά την ανάπτυξη διαδικτυακών εφαρμογών του παγκόσμιου ιστού, πριν αναφερθούμε στους επιμέρους στόχους της είναι χρήσιμο να επιχειρήσουμε μια γρήγορη επισκόπηση βασικών όρων με στόχο την καλύτερη κατανόηση τους.

1.1 Παγκόσμιος ιστός (Web)

Με τον όρο παγκόσμιος ιστός αναφερόμαστε σε ένα ανοιχτό σύστημα διασυνδεδεμένων πληροφοριών και πολυμεσικού περιεχομένου, που επιτρέπει στους χρήστες του Διαδικτύου να αναζητήσουν πληροφορίες μεταβαίνοντας από ένα έγγραφο στο άλλο. [1] Είναι μια τεράστια συλλογή από ψηφιακά έγγραφα (τις ιστοσελίδες), που βρίσκονται αποθηκευμένα σε υπολογιστές που είναι συνδεδεμένοι μέσω του Διαδικτύου. Παρότι η βασική ιδέα – της διασύνδεσης μέσω κοινών πρωτοκόλλων και υποδομών – παραμένει η ίδια, οι τεχνολογίες που τροφοδοτούν τη χρήση του Παγκόσμιου Ιστού παρουσιάζουν ραγδαία εξέλιξη, σε βαθμό τέτοιο που σήμερα συχνά αναφερόμαστε σε γενιές του διαδικτύου (βλέπε διάγραμμα). [2]



Εικόνα 1: Web Evolution

Η πρώτη περίοδος στην εξέλιξη του παγκόσμιου ιστού (δηλαδή πριν από το 1999) χαρακτηρίζεται από τη δυνατότητα δημοσίευσης πληροφορίας την οποία οι χρήστες μπορούσαν μόνο να διαβάσουν με τη χρήση κατάλληλων εφαρμογών. Η περίοδος αυτή είναι γνωστή ως “Read-Only” Web που σηματοδοτεί αυτή ακριβώς την (περιορισμένη) δυνατότητα των τελικών χρηστών. [1] Κεντρικές έννοιες αυτής της περιόδου είναι οι στατικού τύπου ιστοσελίδες, στις οποίες δεν υπήρχε ενεργή επικοινωνία ή ροή πληροφοριών από τον καταναλωτή (της πληροφορίας) προς στον παραγωγό (της πληροφορίας). Έτσι, διαμορφώθηκε ένα μοντέλο, αυτό του Web 1.0 που επέτρεπε στους χρήστες κατά βάση να αναζητούν πληροφορίες και να τις διαβάζουν.

Η διαπίστωση της έλλειψης ενεργητικής διάδρασης μεταξύ παραγωγών και καταναλωτών πληροφορίας του παγκόσμιου ιστού δρομολόγησε εξελίξεις που κατέληξαν στην τροποποίηση του

παραδείγματος του “Read-Only” Web και την προοδευτική μετεξέλιξη του σηματοδοτώντας το Web 2.0. Το ιδιαίτερο χαρακτηριστικό του Web 2.0 είναι ότι δρομολογείται μια αυξημένη συμμετοχή των χρηστών και εναλλακτικοί τρόποι δημιουργίας περιεχομένου από τους ίδιους τους χρήστες του διαδικτύου. Έτσι προοδευτικά, το κυρίαρχο παράδειγμα εξελίσσεται από το πρώιμο “Read-Only” στο “Read-Write” και το “Read-Write-Publish” web. [1]

Το Web 3.0 θα μπορούσε να ονομαστεί και ως “Read – Write – Execute” Web, ωστόσο, είναι δύσκολο να καταλάβουμε την αφηρημένη του μορφή, αν δεν επιχειρήσουμε να εξετάσουμε βασικά συστατικά του. Το πρώτο αφορά τη σημασιολογική σήμανση (Semantic Markup) δεδομένων που κατά βάση σχετίζεται με την δομημένη καταγραφή / αναπαράσταση δεδομένων έτσι ώστε να μπορεί να εξαχεται χρήσιμη πληροφορία σχετικά με τη σημασία τους. Αξίζει να σημειώσουμε ότι μια από τις μεγαλύτερες οργανωτικές προκλήσεις της διαχείρισης των δεδομένων που διακινούνται στον web ήταν η αξιοποίηση των συμφραζομένων έτσι ώστε να καθίσταται δυνατό σε εφαρμογές να αξιοποιούν όχι μόνο τα δεδομένα που αναπαρίστανται αλλά και τα άμεσα συσχετιζόμενα με αυτά. Με την σημασιολογική σήμανση είναι δυνατόν να κωδικοποιήσουμε τέτοιες πρόσθετες μορφές πληροφορίας και να τις καταστήσουμε άμεσα αξιοποιήσιμες από κατάλληλες εφαρμογές. Κατ’ αυτή την έννοια διαχωρίζονται τα δεδομένα αυτά καθ’ αυτά από τις μονάδες λογισμικού που τα παρέχουν, κι έτσι μπορούν να γίνουν εκτελέσιμα (βλέπε “Execute” στον ορισμό) από άλλες υπηρεσίες.

Η δεύτερη έννοια που είναι άρρηκτα συνδεδεμένη με την θεώρηση του “Read – Write – Execute” Web είναι αυτή των υπηρεσιών. Μια υπηρεσία Web είναι ένα σύστημα λογισμικού που έχει σχεδιαστεί για την υποστήριξη της αλληλεπίδρασης υπολογιστή-με-υπολογιστή μέσω του Διαδικτύου. Επί του παρόντος, χιλιάδες διαδικτυακές υπηρεσίες είναι διαθέσιμες. Ωστόσο, το Web 3.0 βρίσκεται στο επίκεντρο. Συνδυάζοντας τη σημασιολογική σήμανση με τις κλασσικές υπηρεσίες Web, το Web 3.0 υπόσχεται τη δυνατότητα οι εφαρμογές να μπορούν να μιλήσουν μεταξύ τους άμεσα, και για την ευρύτερες αναζητήσεις πληροφοριών μέσω απλούστερων διασυνδέσεων. [1]

Το Web 4.0 δεν είναι ακριβώς μια καινούργια έκδοση, αλλά είναι μια εναλλακτική έκδοση που συνδέει το Web με mobile περιβάλλοντα. Ειδικότερα, επιτρέπεται η σύνδεση όλων των συσκευές, ανεξαρτήτως τύπου και σε πραγματικό χρόνο. [1]

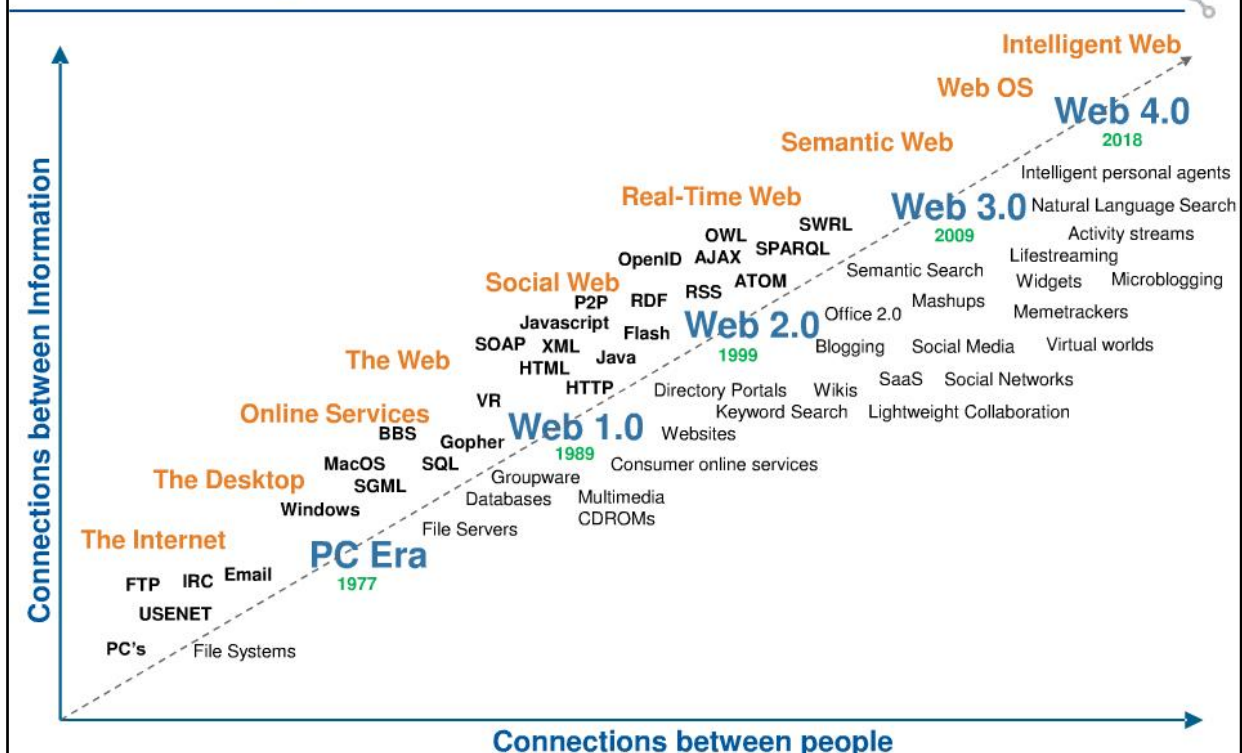
Παρότι υπάρχουν διάφορες εκδόξεις που σηματοδοτούν τα χαρακτηριστικά του επόμενου σταδίου του web (δηλ. το “The next web”), η επικρατέστερη τάση είναι αυτή που προσβλέπει στο διασυνδεδεμένο Ιστό (Web), ο οποίος θα επικοινωνεί μαζί μας όπως εμείς επικοινωνούμε μεταξύ μας. Για πολλούς, ο δόκιμος όρος είναι το “Συμβιωτικό” Web με κατεύθυνση προς το “Read – Write – Execution – Concurrency” Web. Η προοπτική αυτή είναι άρρηκτα συνδεδεμένη με μια περισσότερο συναισθηματική αλληλεπίδραση μεταξύ των ανθρώπων και των υπολογιστών. Σήμερα το Web είναι “συναισθηματικά” ουδέτερο, πράγμα το οποίο σημαίνει ότι το Web δεν αντιλαμβάνεται τα συναισθήματα και τα αισθήματα των χρηστών. Μια πιθανή εξέλιξη προς το “Συμβιωτικό” Web θα σημάωνε μετασχηματισμούς που θα καθιστούν τη χρήση του καθημερινή συνήθεια για πολλούς ανθρώπους. [1]

1.2 Web Application

Στην πληροφορική, ο όρος διαδικτυακή εφαρμογή σηματοδοτεί μια client-server εφαρμογή λογισμικού η οποία εκτελείται μέσα από ένα web browser. Είναι δε δημοφιλής λόγω της πανταχού παρουσίας των web browsers, και της ευκολίας ενός web browser ως client να ενημερώσει και να διατηρεί τις εφαρμογές web χωρίς τη διανομή και την εγκατάσταση του λογισμικού σε χιλιάδες υπολογιστές πελατών. Ευρέως χρησιμοποιούμενες web εφαρμογές περιλαμβάνουν webmail, online λιανικές πωλήσεις, ηλεκτρονικές δημοπρασίες, wikis, υπηρεσίες άμεσων μηνυμάτων και πολλές άλλες. Η ανάπτυξη των εφαρμογών αυτών ακολούθησε την εξέλιξη του Παγκόσμιου Ιστού με αποτέλεσμα οι βασικές τεχνολογίες να αλλάζουν προσφέροντας αυξημένες δυνατότητες στους χρήστες και περισσότερο πλούσιες και διαδραστικές εμπειρίες. Παρακάτω ακολουθεί μια συνοπτική αναφορά σε κομβικά σημεία και τεχνολογίες.

Κατά την πρώτη φάση του Παγκόσμιου Ιστού κάθε ιστοσελίδα ‘παραδινότανε’ στον πελάτη ως ένα στατικό έγγραφο, και ήταν κυρίως η περιήγηση (δηλ. η εναλλαγή μεταξύ ιστοσελίδων) που συνιστούσε τη διαδραστική εμπειρία. Ως αποτέλεσμα, κάθε σημαντική αλλαγή στην ιστοσελίδα απαιτούσε ένα ταξίδι μετ’ επιστροφής πίσω στο διακομιστή για να ανανεωθεί ολόκληρη η σελίδα. [3]

The Intelligence is in the Connections



Εικόνα 2: Semantic Web

Το 1995, η Netscape εισήγαγε μια γλώσσα σήμανσης (για την πλευρά του πελάτη) με την ονομασία JavaScript η οποία επιτρέπει στους προγραμματιστές να προσθέτουν κάποια δυναμικά στοιχεία στη διεπαφή χρήστη που τρέχει από την πλευρά του πελάτη. Έτσι, αντί να αποστέλλονται δεδομένα στο διακομιστή, προκειμένου να δημιουργήσει μια ολόκληρη ιστοσελίδα, τα ενσωματωμένα scripts της κατεβασμένης σελίδας μπορούν να εκτελέσουν διάφορες εργασίες, όπως η επικύρωση των εισόδων του χρήστη ή την εμφάνιση / απόκρυψη τμημάτων της σελίδας. [3]

Το 1996, η Macromedia εισήγαγε το Flash, ένα διανυσματικό κινούμενο player που θα μπορούσε να προστεθεί σε προγράμματα περιήγησης ως plug-in για να ενσωματώσει κινούμενες εικόνες στις ιστοσελίδες. Το Flash επέτρεπε τη χρήση μιας scripting γλώσσας για να προγραμματίσει τις αλληλεπιδράσεις από την πλευρά του πελάτη χωρίς την ανάγκη να επικοινωνήσει με το διακομιστή. [3]

Το 1999, η έννοια “web εφαρμογή” εισήχθη στη γλώσσα Java. Την εποχή εκείνη είχε ήδη αναπτυχθεί η JavaScript και η XML, αλλά η Ajax δεν είχε ακόμη επινοηθεί και το αντικείμενο XMLHttpRequest είχε μόλις πρόσφατα εισαχθεί στον Internet Explorer 5 ως αντικείμενο της τεχνολογίας ActiveX της Microsoft [3].

Το 2005, επινοήθηκε η τεχνολογία Ajax που επέτρεψε σε εφαρμογές όπως το Gmail να κάνουν την πλευρά του πελάτη όλο και πιο διαδραστική. Η βασική καινοτομία ήταν ότι ένα script της ιστοσελίδας μπορούσε να επικοινωνήσει με το διακομιστή για την αποθήκευση / ανάκτηση δεδομένων χωρίς τη λήψη ολόκληρης της ιστοσελίδας [3].

Το 2011, ολοκληρώθηκε η έκδοση της γλώσσας HTML5, που παρέχει γραφικές και πολυμεσικές δυνατότητες, χωρίς την ανάγκη plug-ins από την πλευρά του πελάτη. Στο HTML5 εμπλουτίζεται επίσης το σημασιολογικό περιεχόμενο των εγγράφων. Το μοντέλο αντικειμένου εγγράφων (DOM) και τα APIs αποτελούν θεμελιώδη στοιχεία της συγγραφής HTML5. Το WebGL API άνοιξε το δρόμο για τα προηγμένα 3D γραφικά που βασίζονται στον HTML5 καμβά και τη γλώσσα JavaScript. Αυτά έχουν μεγάλη σημασία για τη δημιουργία πλούσιων εφαρμογών web ανεξαρτήτως browser [3].

1.3 Στόχοι πτυχιακής εργασίας

Στόχος της πτυχιακής εργασίας είναι να μελετήσει βασικές τεχνολογίες της τρέχουσας τεχνολογικής τροχιάς και να αναπτύξει μια ενδεικτική εφαρμογή ειδοποίησης ομάδων που να αξιοποιεί πολλαπλά εναλλακτικά κανάλια ενημέρωσης όπως το ηλεκτρονικό ταχυδρομείο και υπηρεσίες ημερολογίου.

Οι εφαρμογές ειδοποίησης είναι ένας συνδυασμός υλικού και λογισμικού που παρέχουν τη δυνατότητα προώθησης ενός μηνύματος σε ένα σύνολο παραληπτών. Τα συστήματα αυτά αποτελούν μια σημαντική πτυχή στις σύγχρονες εφαρμογές Web. Στη παρούσα εργασία προσβλέπουμε στην ανάπτυξη ενός ευέλικτου μηχανισμού ειδοποιήσεων που να επιτρέπει:

- Αξιοποίηση κοινών επαφών μελών της ομάδας.
- Αλληλεπίδραση μεταξύ μελών μιας ομάδας με διαφορετικά μέσα και υπηρεσίες π.χ. ημερολόγια ανάλογα με το καθήκον που επιτελούν.
- Αυξημένη κινητικότητα των μελών που καθιστούν απαραίτητη την προσαρμογή της ενημέρωσης σε διαφορετικές τερματικές συσκευές.

Κεντρική απαίτηση για τον εν λόγω μηχανισμό είναι η δυνατότητα επεκτασιμότητάς του έτσι ώστε να καθίσταται δυνατός ο εναρμονισμός του μηχανισμού με καθιερωμένα πρότυπα του διαδικτύου και να επιτρέπεται η προσαρμογή του σε διαφορετικά περιβάλλοντα χρήσης, τερματικές συσκευές και υπηρεσίες. Επιπλέον, μια τέτοια δυνατότητα προσφέρεται και για την αξιοποίηση εναλλακτικών καναλιών ενημέρωσης. Για τις ανάγκες της πτυχιακής μελετήθηκαν κανάλια ενημέρωσης όπως τα παρακάτω:

- Το email το οποίο εξακολουθεί να είναι σημαντικό κανάλι ασύγχρονης επικοινωνίας.
- Το ημερολόγιο το οποίο αφενός υιοθετεί μια χρονική μεταφορά αλληλεπίδρασης ευρέως χρησιμοποιούμενη και σχεδόν καθολικά αποδεκτή και αφετέρου αξιοποιείται από πλήθος εφαρμογών οι οποίες βασίζονται σε ημερολόγια για να προσδιορίσουν χρονική ακολουθία καθηκόντων.
- Υπηρεσίες τρίτων που μπορούν να αξιοποιήσουν τον μηχανισμό που προτείνεται στη παρούσα εργασία για να υλοποιήσουν τους δικούς τους εξειδικευμένους μηχανισμούς ενημερότητας.

1.4 Δομή και οργάνωση της πτυχιακής εργασίας

Η παρούσα αναφορά χωρίζεται σε πέντε (5) κεφάλαια τα οποία συνοψίζουν την προσπάθεια του συγγραφέα τόσο στο θεωρητικό σκέλος και προετοιμασία της πτυχιακής όσο και στο σχεδιαστικό και αναπτυξιακό τμήμα αυτής. Ειδικότερα, το **Κεφάλαιο 1** αποτελεί την εισαγωγή όπου παρουσιάζονται βασικοί όροι και συνοψίζεται η θεματολογία της πτυχιακής εργασίας. Στο **Κεφάλαιο 2** παρουσιάζονται συνοπτικά ερευνητικές εργασίες που έχουν άμεση συνάφεια με το επιλεγμένο γνωστικό αντικείμενο. Ειδικότερα, γίνεται αναφορά στο ευρύτερο πεδίο της συνεργασίας με την υποστήριξη Η/Υ, στο ρόλο των ειδοποιήσεων είτε για ενημέρωση μελών ομάδων είτε για την εκτέλεση συνεργατικού έργου, καθώς και σε τρέχουσες τάσεις που αφορούν την σχεδιαστική και ανάπτυξη διαδικτυακών εφαρμογών. Στο **Κεφάλαιο 3** επιχειρείται μια συνοπτική παρουσίαση του μηχανισμού ομαδικών ενημερώσεων εστιάζοντας σε βασικές λειτουργίες που υποστηρίζονται. Στο **Κεφάλαιο 4** αποτελεί το πυρήνα της διπλωματικής εργασίας. Εδώ γίνεται εκτενής αναφορά στο μηχανισμό και τον τρόπο που υλοποιήθηκε. Επίσης, παρουσιάζονται τα βασικά του συστατικά ώστε ο αναγνώστης να αποκτήσει τις βάσεις για να μπορέσει να τον χρησιμοποιήσει ο ίδιος. Τα παραπάνω προσεγγίζονται με τη χρήση ενός γενικού σεναρίου χρήσης που επεξηγείται με ενδεικτικές εικόνες και αποσπάσματα κώδικα. Η αναφορά ολοκληρώνεται με το **Κεφάλαιο 5** όπου γίνεται σύνοψη των βασικών επιτευγμάτων, παράθεση συμπερασμάτων καθώς και καταγραφή προτάσεις που θα μπορούσαν να αποτελέσουν μελλοντικές επεκτάσεις και βελτιώσεις των αποτελεσμάτων της παρούσας εργασίας.

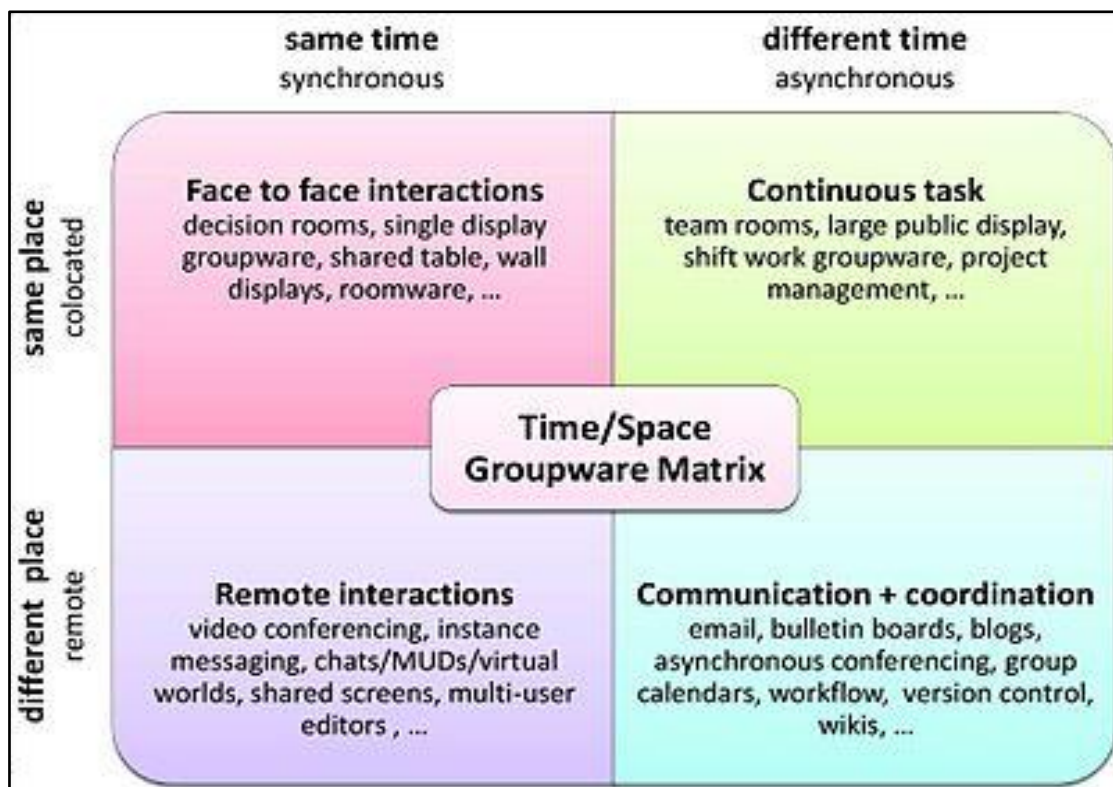
2 Μηχανισμοί Ενημέρωσης Ομάδων: Επισκόπηση

Στο κεφάλαιο αυτό επιχειρούμε μια συνοπτική ανασκόπηση των βασικών θεωρητικών και τεχνολογικών εννοιών που συνδέονται με την παρούσα εργασία ξεκινώντας από το ευρύτερο γνωστικό αντικείμενο της συνεργατικής τεχνολογίας και εξειδικεύοντας στα επιμέρους θέματα που σχετίζονται με την ενημέρωση ομάδων, τις σχεδιαστικές και τεχνολογικές πτυχές της παρούσας εργασίας.

2.1 Συνεργασία υποστηριζόμενη από υπολογιστή (CSCW)

Ο όρος ‘συνεργασία υποστηριζόμενη από υπολογιστή’ ή Computer – Supported Cooperative Work (CSCW) παρουσιάστηκε στη βιβλιογραφία για πρώτη φορά από τους Irene Greif και Paul M. Cashman, το 1984, σε μια συνεδρία εργασίας όπου συμμετείχαν επιστήμονες που τους ενδιέφερε το ευρύτερο θέμα της συνεργασίας εταίρων με τη χρήση της τεχνολογίας. Ακολούθησαν προσπάθειες ορισμού του όρου σαν αυτή του Wilson (1991) όπου ο προσδιορισμός ‘CSCW’ αφορά το συνδυασμό της κατανόησης του τρόπου με τον οποίο οι άνθρωποι εργάζονται σε ομάδες με τεχνολογίες δικτύωσης των υπολογιστών και του υλικού που αξιοποιείται, δηλαδή του λογισμικού, των υπηρεσιών και των τεχνικών που εφαρμόζονται. Επίσης δόκιμος είναι και ο όρος groupware που σηματοδοτεί την κατηγορία τεχνολογικών εργαλείων και τεχνικών που αξιοποιούνται για την υποστήριξη αυτής της μορφής συνεργασίας. Ας σημειωθεί ότι πολλοί συγγραφείς θεωρούν τους όρους CSCW και groupware ως συνώνυμα, ενώ άλλοι ισχυρίζονται ότι, το groupware αναφέρεται σε πραγματικά συστήματα που βασίζονται στον υπολογιστή, ενώ το CSCW εστιάζει στη μελέτη των εργαλείων και τεχνικών του groupware καθώς επίσης και των ψυχολογικών, κοινωνικών, και οργανωτικών τους επιπτώσεων. [4]

Σήμερα ως groupware συστήματα θεωρούνται αυτά που καλύπτουν τις εναλλακτικές μορφές συνεργασίας που συνοψίζονται στην εικόνα 3. Η εικόνα αποτυπώνει τον πίνακα που παρουσιάστηκε το 1988 από τον Johansen και το 1995 από τον Baecker βάση του οποίου η συνεργασία εξετάζεται σε σχέση με δύο διαστάσεις – τον τόπο και τον χρόνο. Η διάσταση του τόπου διαχωρίζει το συνεργατικό έργο που επιτελείται από ομάδα εταίρων που βρίσκονται στο ίδιο γεωγραφικό σημείο από αυτό που είναι υλοποιείται σε διαφορετικά σημεία από διάσπαρτους εταίρους. Η διάσταση του χρόνου, στην



Εικόνα 3: CSCW Matrix

ουσία, διαχωρίζει το συνεργατικό έργο σε σύγχρονο (τα μέλη εργάζονται την ίδια ώρα) και ασύγχρονο (τα μέλη εργάζονται στο χρόνο που τους εξυπηρετεί) ανεξαρτήτως του γεωγραφικού σημείου ή τόπου που (τα μέλη) βρίσκονται. [4]

Η παρούσα εργασία εστιάζει στην κατηγορία εργαλείων συνεργασίας που αποσκοπούν κυρίως στην επικοινωνία και συντονισμό (βλέπε Communication + Coordination στην εικόνα 3) και ειδικότερα στο ρόλο και την συνεισφορά μηχανισμών ομαδικών ενημερώσεων / ειδοποιήσεων μιας και αποτελεί την κατεξοχήν περιοχή ενδιαφέροντος.

2.2 Κατηγορίες ενημέρωσης

Οι ενημερώσεις / ειδοποιήσεις είναι ένα σημαντικό συστατικό της συνεργασίας μελών μιας ομάδας και εξυπηρετούν πολλαπλούς σκοπούς με κυρίαρχο αυτόν της δημιουργίας κοινού τόπου διαμέσου της ενημέρωσης των εμπλεκόμενων εταίρων σχετικά με κάποιο συμβάν, όπως η υλοποίηση ενός στόχου ή ένα γεγονός το οποίο θα γίνει σε μελλοντικό χρόνο. Αυτό μεταξύ των άλλων επιτρέπει σε χρήστες που συνεργάζονται να έχουν επίγνωση των προθέσεων, δράσεων και των αποτελεσμάτων των υπολοίπων μελών. Για το σκοπό αυτό υπάρχουν διάφορων ειδών κατηγορίες ή είδη ενημέρωσης ομάδων, με ενδεικτικότερους αυτούς που αναλύονται παρακάτω.

2.2.1 Alerts

Η κατηγορία αυτή έχει να κάνει με την επικοινωνία μεταξύ μηχανής-χρήστη η οποία είτε είναι σημαντική είτε έχει σαν παράγοντα το χρόνο. Ένα alert μπορεί να είναι μια υπενθύμιση από ένα ηλεκτρονικό ημερολόγιο ή μια ειδοποίηση για ένα νέο μήνυμα, η οποία ωστόσο δεν επηρεάζει έμμεσα και δεν δεσμεύει τη τρέχουσα δραστηριότητα του χρήστη. Με άλλα λόγια είναι στη διακριτική ευχέρεια του χρήστη να επιλέξει το πως θα χειριστεί το alert.

Τα alerts συνήθως, παραδίδονται μέσω ενός συστήματος ειδοποίησης. Ένα τέτοιο σύστημα είναι σε θέση να “αντιλαμβάνεται” συγκεκριμένα γεγονότα και να δρομολογεί κατάλληλα την ενημέρωση του χρήστη. Κλασσικές υπηρεσίες που αξιοποιούνται για το σκοπό αυτό είναι το e-mail ή SMS. Πιο προηγμένα συστήματα (π.χ. AOL) παρέχουν στους χρήστες την επιλογή να επιλέξουν τον τρόπο παροχής της ειδοποίησης, όπως παραδείγματος χάριν μέσω αποστολής e-mail, αποστολή σύντομων μηνυμάτων (SMS) ή άμεσων μηνυμάτων (IM), ενεργοποίηση φωνητικών μηνυμάτων (μέσω φωνητικών πυλών), ειδοποιήσεων στην επιφάνεια εργασίας, κλπ. Καινοτόμες προσεγγίσεις παρέχουν στους χρήστες τη δυνατότητα να προγραμματίσουν τις δικές τους ειδοποιήσεις (όπως για παράδειγμα το Google Calendar).

Ενδεικτικά παραδείγματα σε υπηρεσίες όπως το Booking, το RentalCars, και πολλές άλλες χρησιμοποιούν alerts για να ενημερώσουν τους χρήστες – πελάτες για τυχόν προσφορές. Alerts σε κοινωνικά μέσα όπως το LinkedIn, το ReaserchGate, το Facebook, και άλλα, χρησιμοποιούνται για να ενημερώσουν το χρήστη για κάποιο γεγονός που συνέβη στο αντίστοιχο κοινωνικό περίγυρο του χρήστη (π.χ. αποστολή μηνύματος ή κοινοποίηση ενός επιτεύγματος ή νέας ανάρτησης που ίσως ενδιαφέρει).

2.2.2 Reminders

Τα reminders είναι μια ειδική κατηγορία ενημερώσεων που αποσκοπεί κυρίως στο να προειδοποιήσει το χρήστη για σημαντικά γεγονότα που απαιτούν την προσοχή του. Στις περισσότερες περιπτώσεις οι ειδοποιήσεις αυτού του είδους προέρχονται από κάποιο εργαλείο (π.χ. ένα ημερολόγιο) ή/και λίστα γεγονότων (π.χ., γενέθλια, γιορτές, επετείους, προγραμματισμένες πτήσεις, διαλέξεις, κλπ.) ο χρόνος των οποίων προκαλεί την ενεργοποίηση της εκάστοτε τεχνικής υπενθύμισης όπως παραδείγματος χάριν τα pop-up παράθυρα διαλόγου, ακουστικές ειδοποιήσεις, αποστολή μηνυμάτων, κλπ.). Είναι επίσης σύνηθες οι ειδοποιήσεις αυτής της κατηγορίας να αφορούν προ-εγκατεστημένα γεγονότα όπως εθνικές επέτειοι, ημερομηνίες ορόσημα, κλπ.

Μια σημαντική εξέλιξη των τελευταίων ετών είναι η διασύνδεση συστημάτων web με μια σειρά από online υπηρεσίες υπενθύμισης. Οι χρήστες οι οποίοι έχουν γίνει συνδρομητές στις υπηρεσίες αυτές, συνήθως λαμβάνουν τις υπενθυμίσεις τους μέσω e-mail ή SMS. Ενδεικτικά παραδείγματα σε υπηρεσίες

όπως το Booking, το RentalCars, και πολλές άλλες χρησιμοποιούν reminders για να ευχηθούν στους χρήστες – πελάτες τους στα γενέθλια τους ή αν ο χρήστης – πελάτης έχει κάνει κάποιου είδους κράτηση στην υπηρεσία, να τον υπενθυμίσει όταν πλησιάζει η ημερομηνία κράτησης. Ένα άλλο παράδειγμα είναι η κράτηση αεροπορικού εισιτηρίου, κατά την οποία όταν πλησιάζει η ημερομηνία αναχώρησης, μια υπενθύμιση αποστέλλεται στον χρήστη για να κάνει check-in. Reminders σε κοινωνικά μέσα όπως το LinkedIn, το ReaserchGate, το Facebook, και άλλα, χρησιμοποιούνται για να υπενθυμίσουν στο χρήστη για συμβάν το οποίο έχει δηλώσει ενδιαφέρον, ότι πλησιάζει η ημερομηνία.

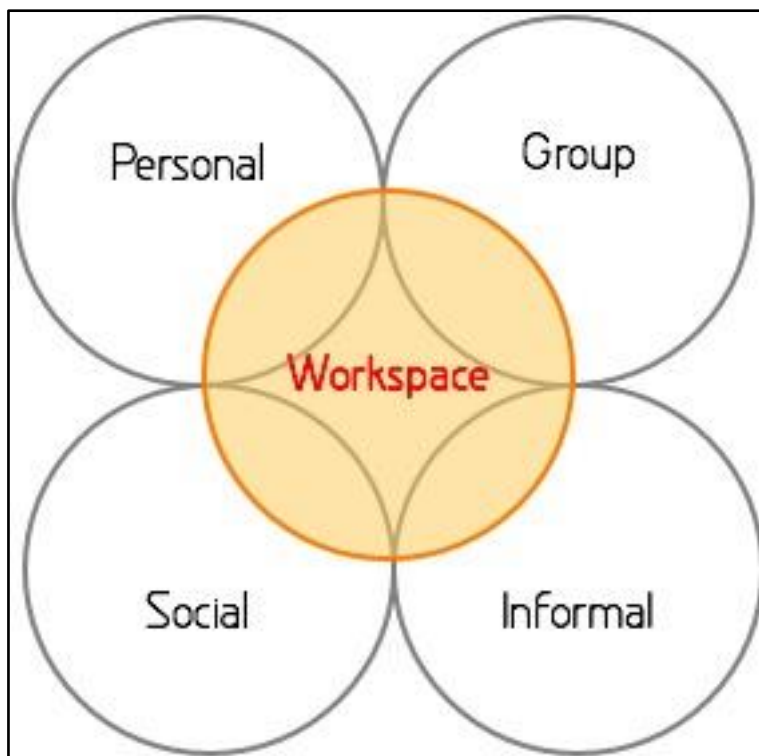
2.2.3 Notifications - Awareness

Ένα τελευταίο είδος ειδοποιήσεων αφορά την ενημέρωση του χρήστη σχετικά με επίκαιρα θέματα του ενδιαφέροντος του που λαμβάνουν χώρα είτε διαφορετικά γεωγραφικά σημεία (π.χ. τίτλους ειδήσεων, ειδικές πωλήσεις, online προόδους μιας δημοπρασίας, βαθμολογίες ενός αθλητικού παιχνιδιού,) είτε σε εφαρμογές που βρίσκονται εγκατεστημένες στο χώρο εργασίας τους ή τον υπολογιστή του (π.χ. νεοαφιχθέντα email ή μηνύματα, κλήσεις, κλπ.). Σε αντίθεση με τις προηγούμενες δύο κατηγορίες ειδοποιήσεων, τα notifications αυτής της μορφής μπορούν να υιοθετούν εναλλακτικό γλωσσικό ιδίωμα όπως π.χ. σύμβολα ή αναδυόμενους διαλόγους με στόχο να προσελκύσουν το ενδιαφέρον και την προσοχή του. [5]

Ενδεικτικά παραδείγματα σε υπηρεσίες όπως το Booking, το RentalCars, και πολλές άλλες χρησιμοποιούν notifications και awareness για να παρουσιάσουν στο χρήστη στοχευμένες διαφημίσεις ή και στοχευμένες προσφορές που πιθανόν να τον ενδιαφέρουν βάση των αναζητήσεων που έκανε μέσα στην υπηρεσία. Notifications και Awareness σε κοινωνικά μέσα όπως το LinkedIn, το ReaserchGate, το Facebook, και άλλα, χρησιμοποιούνται για να παρουσιάσουν στο χρήστη στοχευμένες διαφημίσεις ή/και πεπραγμένα γνωστών ή φίλων. Στα παραδείγματα αυτά, οι ειδοποιήσεις εμφανίζονται κατά τη διάρκεια που ο χρήστης χρησιμοποιεί κάποια υπηρεσία ή κάποιο κοινωνικό μέσο.

Υπάρχουν 5 τύποι ενημέρωσης στην κατηγορία αυτή, οι οποίοι είναι: [6]

- **Personal awareness:** Αφορούν πληροφορίες τις οποίες οι χρήστες διατηρούν για τον εαυτό τους και τον ρόλο τους στη ομάδα. Μπορεί να είναι είτε σύγχρονες ειδοποιήσεις (π.χ. τρέχουσα τοποθεσία μέσα στο σύστημα) είτε ασύγχρονες ειδοποιήσεις (π.χ. ιστορικό παρουσίας ενός χρήστη στο σύστημα).
- **Informal awareness:** Αφορούν ειδοποιήσεις που έχουν στόχο να προσφέρουν στο χρήστη μια γενική αίσθηση του ποιος είναι τριγύρω (δηλ. συνδεδεμένος σε ένα σύστημα) και τι κάνει ή επιδιώκει να κάνει. Η κατηγορία αυτή άτυπων ειδοποιήσεων προσφέρουν ένα είδος άτυπης επίγνωση που συχνά διευκολύνει τυπικές αλληλεπίδρασης.
- **Social awareness:** Η κατηγορία αυτή αφορά ειδοποιήσεις που το περιεχόμενός τους είναι σχετικό με τον περίγυρο ενός χρήστη σε ένα συλλογικό πλαίσιο (π.χ. μια online συνομιλία). Συχνά οι ειδοποιήσεις αυτές σηματοδοτούν είτε το πλήθος των χρηστών που συμμετέχουν είτε τη συναισθηματική τους κατάσταση είτε τις επιλογές και δεσμεύσεις τους σε επιμέρους ζητήματα ή την τρέχουσα επικαιρότητα.
- **Group-structural awareness:** Ειδοποιήσεις που σχετίζονται με τους ρόλους των μελών μιας ομάδας, τις αρμοδιότητες και τις ευθύνες τους, τις θέσεις/απόψεις τους σε ένα θέμα, το καθεστώς και τις διαδικασίες της ομάδας, τις επιμέρους δραστηριότητες ή / και πράξεις μελών της ομάδας, κλπ.
- **Workspace awareness:** Αφορά την παρουσία του χρήστη στο χώρο εργασίας και τι οι άλλοι χρήστες κάνουν: συνεχόμενη ενημέρωση σχετικά με τις αλληλεπιδράσεις των άλλων χρηστών στο χώρο εργασίας. Αυτό σημαίνει επίγνωση πάνω σε τι εργάζονται οι άλλοι, τι κάνουν και τι πρόκειται να γίνει στη συνέχεια, και που πρόκειται να γίνει. Η πληροφορία αυτή είναι χρήσιμη για πολλές από τις δραστηριότητες σε συνεργατική συνεργασία, για συντονισμό των ενεργειών, τη διαχείριση σύζευξης, για τον διάλογο μεταξύ των χρηστών για το έργο, για την πρόβλεψη των δράσεων των άλλων και την ευκαιρία για να βοηθήσει ο ένας τον άλλο.



Εικόνα 4: Different kinds of Awareness

2.3 Ενημερότητα ως λειτουργία

Από την προηγούμενη ανάλυση προκύπτει ότι ένας συνεχώς αυξανόμενος αριθμός διαδικτυακών υπηρεσιών και συστημάτων οικειοποιούνται των ειδοποιήσεων προκειμένου να προσφέρουν στους χρήστες/πελάτες τους βέλτιστη ενημέρωση για καθημερινές δραστηριότητες ή/και επαγγελματικές ενασχολήσεις. Η τάση αυτή επιτείνεται από τις δυνατότητες που προσφέρουν τα αναδυόμενα συστήματα απανταχού υπολογισμού (Ubiquitous Computing ή UbiComp). Όλο και περισσότερα, απλά αντικείμενα και φυσικοί χώροι προσφέρουν υπολογιστική ισχύ για τον χρήστη: τηλέφωνα, συστήματα ψυχαγωγίας, αυτοκίνητα, σαλόνια αεροδρόμιων, καφετέριες, κλπ. Μια φυσική συνέπεια του γεγονότος αυτού είναι η ώθηση της πληροφορικής πέρα από την επιφάνεια εργασίας, η κλιμάκωση της χρήσης τόσο στο χώρο όσο και στο χρόνο και κατ' επέκταση η αύξηση της ποσότητας και του περιεχομένου των εν δυνάμει ειδοποιήσεων. Συνεπώς, η διαχείριση των ειδοποιήσεων καθίσταται κρίσιμο ζήτημα, δεδομένου ότι είναι δυνατόν οι ειδοποιήσεις να καταστούν ανασταλτικός παράγοντας με αρνητικά αποτελέσματα στην παραγωγικότητα του χρήστη, ειδικά όταν ο χρήστης είναι επικεντρωμένος σε ένα καθήκον. Με άλλα λόγια, αφού η ανθρώπινη προσοχή είναι περιορισμένη, η ευαισθητοποίηση των χρηστών σε σχετικές με το καθήκον πληροφορίες ή γεγονότα με τρόπο που να προκαλεί την ελάχιστη πίεση στους γνωστικούς του πόρους, υπόσχεται να είναι όλο και πιο πολύτιμη σε μεμονωμένους χρήστες αλλά και ομάδες συνεργαζόμενων [5]. Επομένως, η κατανόηση της γνωσιακής λειτουργίας των ειδοποιήσεων είναι σημαντική για τον προσδιορισμό των βασικών σχεδιαστικών παραμέτρων ενός συστήματος διαχείρισης ειδοποιήσεων.

2.3.1 Γνωσιακή ενίσχυση

Μεταξύ των διαφόρων τύπων ειδοποιήσεων, η κατηγορία των notifications – awareness για ομάδες χρηστών αφορά κυρίως την ευαισθητοποίηση των μελών κατά τη διάρκεια εκτέλεσης του πρωτεύοντος καθήκοντος [9]. Η έννοια των πρωτεύοντος καθήκοντος παραπέμπει σε τεκταινόμενα που συμβαίνουν μέσα στα κεφάλια των ανθρώπων. Παρόλα αυτά, οι άνθρωποι βρίσκονται και λειτουργούν στον πραγματικό κόσμο, δέχονται διαφορετικά ερεθίσματα και ανταποκρίνονται σε αυτά με

διαφορετικό τρόπο και σε διαφορετικούς χρόνους. Στις περιπτώσεις δε που η λειτουργία αυτή απαιτεί συνεργατική εκτέλεση καθηκόντων είναι προφανές ότι υπάρχουν προϋποθέσεις προκειμένου να δημιουργηθεί κοινή βάση κατανόησης για το ποιος είναι κοντά, τι κάνει, με ποιο τρόπο, κλπ. Με άλλα λόγια απαιτούνται μηχανισμοί διακριτικής παρακολούθησης της κατάστασης των τεχνουργημάτων, των εμπλεκομένων και του τρόπου που αυτά αλληλοεπηρεάζονται. Επομένως, η έκβαση ενός συνεργατικού καθήκοντος επηρεάζεται από το περιβάλλον στο οποίο τα μέλη εργάζονται συνεργατικά, το είδος των συστημάτων που χρησιμοποιούν για να υποστηρίξουν κατανομημένη συνεργασία, τα καθήκοντα που εκτελούν και τον τύπο των ομάδων τους. Αναλυτικότερα: [7]

- **Environment: Shared Workspaces.** Πολλές ομάδες συχνά εργάζονται πάνω σε ένα κοινόχρηστο οπτικό χώρο όπου μπορούν να δουν, να δημιουργήσουν, και χειριστούν αντικείμενα που σχετίζονται με τις δραστηριότητές τους. Ένα παράδειγμα είναι μια επίπεδη επιφάνεια (π.χ., ένα μεγάλος πίνακας) πάνω στον οποίο τα αντικείμενα μπορούν να τοποθετούνται και να χειραγωγούνται, και γύρω από αυτήν τη επιφάνεια, μια μικρή ομάδα ανθρώπων να μπορούν να συνεργάζονται.
- **Systems: real-time distributed groupware.** Αυτά τα συστήματα επιτρέπουν στις ομάδες να συνεργάζονται ταυτόχρονα, αλλά από διαφορετικούς εργασιακούς χώρους.
- **Tasks: generation and execution.** Πρωτογενής τύποι εργασίας σε κοινούς χώρους εργασίας είναι δραστηριότητες παραγωγής και εκτέλεσης, όπου οι άνθρωποι δημιουργούν νέα αντικείμενα, περιηγούνται μέσα από ένα χώρο αντικειμένων, ή χειρίζονται ήδη υπάρχοντα αντικείμενα.
- **Groups: small groups and mixed-focus collaboration.** Μικρές ομάδες μεταξύ δύο και πέντε ατόμων έχουν κατά κύριο στόχο την εκτέλεση των καθηκόντων σε αυτούς τους μεσαίους χώρους εργασίας. Οι ομάδες αυτές συχνά ενασχολούνται σε συνεργασία μικτής-εστίασης, όπου οι άνθρωποι μετατοπίζονται συχνά μεταξύ των ατομικών και των κοινών δραστηριοτήτων κατά τη διάρκεια μιας περιόδου εργασίας.

Επίγνωση (awareness) είναι η γνώση που αποκτάται μέσω της αλληλεπίδρασης μεταξύ ενός παράγοντα και του περιβάλλοντος του και προσδιορίζεται από ορισμένα βασικά χαρακτηριστικά:

- Την τρέχουσα κατάσταση του συγκεκριμένου περιβάλλοντος και των χρηστών που συμμετέχουν
- Τυχόν αλλαγές που λαμβάνουν χώρα στο ίδιο το περιβάλλον και τις συνιστώσες του, ως συνέπεια είτε 'εξωτερικών' είτε 'εσωτερικών' παραμέτρων
- Επιρροή και επιπτώσεις που τα παραπάνω έχουν στην εκτέλεση έργου.

2.4 Σχεδιαστικά ζητήματα

Από τη μέχρι τώρα ανάλυση προκύπτει ότι η συνεργατική εκτέλεση καθηκόντων προβάλλει ιδιαίτερες απαιτήσεις καθώς τα μέλη συνήθως βρίσκονται σε απόσταση και δεν εμπλέκονται ταυτόχρονα στο ίδιο πρωτεύον καθήκον. Συνεπώς, ενώ είναι απαραίτητο τα μέλη της ομάδας να έχουν επίγνωση των μεταξύ τους προσθέσεων, δράσεων και αποτελεσμάτων, αυτό πρέπει να επιτυγχάνεται με τρόπο που να μην επηρεάζει αρνητικά την αποτελεσματικότητά τους.

Συστήματα ειδοποίησης συνήθως υποστηρίζουν την επίγνωση της παρουσίας, των καθηκόντων και των ενεργειών των συνεργατών σε επιλεγμένα αντικείμενα, αλλά δεν υποστηρίζουν επαρκώς την επίγνωση των σύνθετων δραστηριοτήτων [8]. Σε όλες τις περιπτώσεις τρία είναι τα κύρια ζητήματα που βρίσκονται στο επίκεντρο της σχεδίασης μηχανισμών ειδοποιήσεων και ενημερότητας. Το πρώτο έχει να κάνει με το χρόνο, δηλαδή πότε πρέπει να σταλεί μια ειδοποίηση. Το δεύτερο έχει να κάνει με τους αποδέκτες δηλ. σε ποιους πρέπει να σταλεί μια ειδοποίηση. Τέλος, το τρίτο αφορά το είδος της ειδοποίησης. Τα τρία αυτά challenges αναλύονται συνοπτικά στις επόμενες παραγράφους.

2.4.1 Ο χρόνος των ειδοποιήσεων - Πώς να προβλέψουμε την κατάλληλη στιγμή

Ένας αυξανόμενος αριθμός ειδοποιήσεων που απαιτούν την προσοχή του χρήστη, συχνά φτάνουν σε ακατάλληλη στιγμή, με τις περισσότερες από αυτές τις στιγμές ο χρήστης να αγνοεί την ειδοποίηση. Ψυχολογικές μελέτες έχουν δείξει ότι η έναρξη των αλληλεπιδράσεων σε ακατάλληλες στιγμές μπορεί να γίνει πηγή της διακοπής. Μια τέτοια διακοπή μπορεί να επηρεάσει αρνητικά το έργο του χρόνου ολοκλήρωσης, ποσοστό σφάλματος, και συναισθηματική κατάσταση του χρήστη.

Ωστόσο, μέχρι τώρα, το επίκεντρο ήταν αν η ειδοποίηση έχει παραδοθεί και όχι το πραγματικό περιεχόμενο της κοινοποίησης. Για παράδειγμα, μια ειδοποίηση από μια συζήτηση σε ένα μηχανισμό chat από έναν φίλο μπορεί να είναι εξαιρετικά διασπαστική, αν παραδοθεί κατά τη διάρκεια μιας συνάντησης. Όμως, μια ειδοποίηση μέσω ηλεκτρονικού ταχυδρομείου από ένα συνεργάτη του έργου θα μπορούσε να είναι αποδεκτή από τον χρήστη, και σε ορισμένες περιπτώσεις, θεωρείται πολύ χρήσιμη.

Ο Sahami et al. έδειξε ότι οι κοινοποιήσεις που ενεργοποιούνται από διαφόρων ειδών εφαρμογές δίνονται διαφορετική σπουδαιότητα από τους χρήστες. Ταυτόχρονα, οι Pielot et al. ισχυρίστηκε ότι η ανταπόκριση των χρηστών επηρεάζεται από την κοινωνική πίεση. Αυτό υποδηλώνει ότι η απόφαση του χρήστη για το πώς να ανταποκριθεί σε μια ειδοποίηση γίνεται μετά την εξέταση του τίτλου της ειδοποίησης, που δίνει μια ένδειξη σχετικά με τις πληροφορίες που περιέχονται σε αυτή.

Από την άλλη, ο Grandhi et al. πρότεινε ότι μόνο βασισμένοι στη γνώση και τη διαχείριση διακοπής, συχνά τα συστήματα αποτυγχάνουν να συμπεράνουν αν οι διακοπές είναι ενοχλητικές ή όχι επειδή δεν λαμβάνουν υπόψη τον αποστολέα διακοπής και τις πληροφορίες που αποστέλλονται. Έτσι, είναι καλύτερα να προβλέπεται μια κατάλληλη στιγμή για να παραδοθεί μια ειδοποίηση λαμβάνοντας υπόψη και το περιεχόμενο της αλλά και το περιβάλλον στο οποίο θα παραδοθεί. Για να κατανοήσουμε καλύτερα τον ορισμό αυτό, ας εξετάσουμε ένα σενάριο στο οποίο ένα άτομο στο χώρο εργασίας του, λαμβάνει μια ειδοποίηση σχετικά με η άφιξη ενός μηνύματος ηλεκτρονικού ταχυδρομείου από έναν συνάδελφο που εργάζονται για το ίδιο έργο και λαμβάνει και ένα νέο σχόλιο σε ένα από τα κοινωνικά της δίκτυα. Δεδομένων αυτών των συνθηκών, ο χρήστης αποδέχεται την ειδοποίηση μέσω ηλεκτρονικού ταχυδρομείου αφήνοντας την ειδοποίηση στο κοινωνικό της δίκτυο. Αργότερα, στο δρόμο για το σπίτι, ο χρήστης μπαίνει να δει την ειδοποίηση και να διαβάσει το νέο σχόλιο. Σε αυτό το σενάριο, αφού μόνο μία κοινοποίηση διαβάστηκε, είναι αδύνατο να αποφασιστεί κατά πόσον η στιγμή αποστολής είναι η κατάλληλη ή όχι χρησιμοποιώντας μόνο το περιβάλλον του χρήστη. Και οι δύο κοινοποιήσεις έφτασαν την ίδια στιγμή, αλλά έγιναν δεκτές σε διαφορετικά χρονικά πλαίσια.

Επομένως, είναι το περιεχόμενο κοινοποίησης που επιτρέπει στον παραλήπτη να αποφασίσει αν θα την αποδεχθεί ή όχι στο τρέχον χρονικό πλαίσιο. Έτσι, το περιεχόμενο της ειδοποίησης και το περιβάλλον του χρήστη μαζί παίζουν σημαντικό ρόλο στον εντοπισμό μιας κατάλληλης χρονικής στιγμής για να παραδοθεί μια ειδοποίηση. Ωστόσο, για λόγους προστασίας της ιδιωτικής ζωής μπορεί να μην είναι εφικτό να εκμεταλλεύεται το περιεχόμενο των ειδοποιήσεων, διότι ενδέχεται να περιέχουν εξαιρετικά ευαίσθητες πληροφορίες.

Κατά συνέπεια, η χρήση του τίτλου της ειδοποίησης που περιέχει περισσότερες υψηλού επιπέδου, αφηρημένες, αλλά και χρήσιμες πληροφορίες για την κατηγορία των πληροφοριών που περιέχονται στο ίδιο το μήνυμα. Πιο συγκεκριμένα, ο τίτλος μπορεί να χρησιμοποιηθεί για να χαρακτηρίσει την κατηγορία της ειδοποίησης [9].

Feature	Description
Arrival time	Time at which a notification arrives in the notification bar
Removal time	Time at which a notification is removed from the notification bar
Response time	Difference between arrival and removal time
Notification response	Whether the notification was clicked or not (boolean)
Sender application	Name and package of an application which triggers a notification
Notification title	Title of a notification displayed in the notification bar
Alert type	Signals used to alert the user for a notification: sound, vibrate, and LED
Physical activity	Current activity of a user
Location	Current location of a user
Surrounding sound	Whether the user is in a silent environment or not (boolean)
WiFi connectivity	Whether the phone is connected to a WiFi or not (boolean)
Proximity	Whether the user was proximate to the phone in the last one minute or not (boolean)
Phone's status	Whether the phone was in use in the last one minute or not (boolean)
Ringer mode	Current ringer mode: sound, vibrate and LED

Πίνακας 1: Description of features from the NotifyMe dataset

Η android εφαρμογή NotifyMe τρέχει στο παρασκήνιο της συσκευής και παρακολουθεί διακριτικά τις ειδοποιήσεις και το πλαίσιο στο οποίο είναι τοποθετημένες. Στηρίζεται στην Υπηρεσία Android Notification Listener για τον εντοπισμό των ειδοποιήσεων, και χρησιμοποιεί το Google Activity Recognition API και το ESSensorManager, για να συγκεντρώσει τις πληροφορίες του πλαισίου. Ο Πίνακας 1 παραθέτει την περιγραφή των χαρακτηριστικών που συλλαμβάνονται από την εφαρμογή NotifyMe. Αξίζει να σημειωθεί ότι το “Alert Type” υποδεικνύει τα σήματα που χρησιμοποιούνται από μια ειδοποίηση για να ειδοποιηθεί ο χρήστης, ενώ το “Ringer Mode” αναφέρεται σε ρυθμίσεις κλήσεων του τηλεφώνου.

Για να εξαχθεί η απάντηση των χρηστών σε μια ειδοποίηση, η εφαρμογή NotifyMe ελέγχει αν η εφαρμογή που προκάλεσε την ειδοποίηση ξεκίνησε μετά το χρόνο αφαίρεσης της εν λόγω ειδοποίησης. Εφόσον οι περισσότερες από τις ειδοποιήσεις προκαλούνται από chat και email εφαρμογές, οι χρήστες συνήθως κάνουν “κλικ” σε τέτοιες ειδοποιήσεις για να διαβάσουν το πλήρες κείμενο και να απαντήσουν. Ως εκ τούτου, μπορούμε να υποθέσουμε ότι οι χρήστες κάνουν “κλικ” στις ειδοποιήσεις που φτάνουν την κατάλληλη στιγμή. Η προσέγγισή αυτή έχει περιορισμούς, επειδή ορισμένες ειδοποιήσεις, οι οποίες δεν απαιτούν περαιτέρω δράση, ενδέχεται ο χρήστης να μην κάνει “κλικ” αλλά να δει και απορρίψει την ειδοποίηση. [9]

2.4.2 Οι παραλήπτες – Πως καθορίζεται η ομάδα αποδεκτών

Υπάρχουν διάφορες τεχνικές σχετικά με τον καθορισμό / την θέσπιση εικονικών ομάδων για την ομαδική κοινοποίηση. Οι τεχνικές αυτές περιλαμβάνουν:

- Σχηματισμός ομάδας βάση ετικέτας (Tag-based Group Formation)
- Σχηματισμός ομάδας βάση γεωγραφικής θέσης (Location-based Group Formation)
- Σχηματισμός ομάδας βάση αντικειμένων (Object-Based Group Formation)

Ο σχηματισμός ομάδας βάση ετικέτας βασίζεται στην αντιμετώπιση των συμφερόντων των χρηστών, τις τρέχουσες προτιμήσεις και το υπόβαθρο των ετικετών. Για παράδειγμα, η ετικέτα “LTU” μπορεί να χρησιμοποιηθεί ως ετικέτα για τον σχηματισμό μιας ομάδας χρηστών με ένα υπόβαθρο ετικετών LTU. Στη συνέχεια, διαφορετικές παράμετροι φιλτραρίσματος μπορούν να βελτιστοποιήσουν τη λειτουργικότητα, όπως η χρήση της εγγύτητας των επαφών και της κοινωνικής δύναμης μεταξύ τους. Η εγγύτητα υπολογίζεται βάση του ιδιοκτήτη της ομάδας (την τρέχουσα θέση του κυρίως χρήστη), ενώ η κοινωνική δύναμη συλλέγεται από το Acquisition System Guidance Framework (ASG), το οποίο διαχειρίζεται και κατατάσσει τις επαφές βάση του ιστορικού της επικοινωνίας τους. Τότε, οι επαφές ενός χρήστη μπορούν να φιλτράρονται, όπου η ακτίνα εγγύτητας βασίζεται στην τοποθεσία του χρήστη. Μια άλλη προσέγγιση του μηχανισμού αυτού είναι οι επαναλαμβανόμενες λειτουργίες που χρησιμοποιούνται για τη μεταρρύθμιση της ομάδας με νέες επαφές. Σε αυτή την περίπτωση, κάθε φορά που η υπηρεσία ASG ενημερώνει τα δεδομένα του χρήστη, οι σχηματισμοί λειτουργίας της ομάδας ελέγχουν κατά πόσον οι νέες επαφές πληρούν τα κριτήρια για να γίνουν μέλη της ομάδας και, αν ναι γίνονται οι απαραίτητες μεταρρυθμίσεις στην ομάδα με τους νέους συμμετέχοντες.

Μια εναλλακτική μέθοδος, κάνει χρήση της τοποθεσίας ή της εγγύτητας των δεδομένων. Για παράδειγμα, χώροι συνάντησης, όπως ένα σουπερ μάρκετ, ένα αεροδρόμιο, ένας σταθμός λεωφορείων ή τρένων, ένα σινεμά και ούτω καθεξής μπορεί να παρέχει το λόγο για τον καθορισμό της λίστας των παραληπτών μιας ομάδας σε μια υπηρεσία ειδοποίησης. Εδώ, η ιδέα είναι ο σχηματισμός προσωρινών ομάδων ανθρώπων που είναι διαθέσιμοι σε μια θέση μια συγκεκριμένη χρονική στιγμή. Αυτού του είδους η ομάδα μπορεί να αξιοποιηθεί για τον διαμοιρασμό πολυμεσικού ψηφιακού περιεχομένου το οποίο σχετίζεται με την τοποθεσία, όπως κάποια διαφήμιση, ανακοίνωση, εσωτερικοί χάρτες, και ούτω καθεξής. Μπορεί επίσης να βοηθήσει στην συνάντηση των φίλων που είναι διαθέσιμοι στη συγκεκριμένη θέση. Οι ομάδες διαμορφώνονται με ένα επαναλαμβανόμενο τρόπο, έτσι ώστε ο χρήστης καλείται στην ομάδα όταν οι περιορισμοί εγγύτητας ικανοποιούνται. Το περιεχόμενο δεν θα μπορούσε να αποστέλλεται σε χρήστες που δεν πληρούν τους περιορισμούς εγγύτητας.

Τέλος, με το σχηματισμό ομάδας που βασίζεται σε αντικείμενα, συνηθίζεται να θεωρούνται ως συμμετέχοντες στην ομάδα όχι μόνο άνθρωποι χρήστες, αλλά και έξυπνες συσκευές του χρήστη (όπως κινητά τηλέφωνα, συσκευές tablet, φορητούς υπολογιστές, τηλεοράσεις, ψυγεία, ψηφιακές μηχανές προβολής και αυτοκίνητα) ή συγκεκριμένες ψηφιακές υπηρεσίες (π.χ. e-mail, ημερολόγια). Ένα αντικείμενο μπορεί να συνδέεται με πολλά ανθρώπινα όντα ή με μηχανικές οντότητες. Για παράδειγμα, μια οικογένεια μπορεί να θεωρηθεί ως ένα αντικείμενο, το οποίο έχει μια φυσική τοποθεσία (το σπίτι) και η οποία συνδέεται με τα μέλη της οικογένειας (που ζουν στο σπίτι), καθώς και τις έξυπνες συσκευές στο σπίτι. Ως εκ τούτου, για να σχηματίσουν μια ομάδα με βάση την οικογένεια, σαν αντικείμενο, όλες αυτές οι οντότητες θα κληθούν στην ομάδα.

Οι τρεις αυτές μέθοδοι που αναφέρθηκαν νωρίτερα εμπίπτουν σχηματισμό με βάση τα συμφοραζόμενα με τα οποία μια ομάδα μπορεί να συγκριθεί με τον εντοπισμό των συνθηκών υπό τις οποίες κάθε μία από τις μεθόδους μπορεί να είναι προτιμότερη. Η σύγκριση γίνεται με τους ακόλουθους παράγοντες:

- Συμφοραζόμενη πληροφορία: Η συμφοραζόμενη πληροφορία είναι ένας παράγοντας ζωτικής σημασίας για το σχηματισμό ομάδας με βάση τα συμφοραζόμενα. Οι ετικέτες, οι γεωγραφικές θέσεις, οι σχέσεις και ούτω καθεξής αποτελούν παράδειγμα συμφοραζόμενης πληροφορίας που χρησιμοποιείται για τον σχηματισμό της ομάδας.
- Υποψήφιοι ομάδας: Δεν αρκεί πάντα να λαμβάνονται υπόψη μόνο οι άνθρωποι χρήστες ως υποψήφιοι ομάδας, με τη συμπερίληψη έξυπνων συσκευές ως συμμετέχοντες της ομάδας βελτιώνεται η επικοινωνία της ομάδας. Για παράδειγμα, το αυτοκίνητό σας μπορεί να είναι μέλος μιας ομάδας, καθώς μπορείτε να αλληλεπιδράσετε με το αυτοκίνητό σας για να ξεκινήσει η θέρμανση μια ώρα πριν φύγετε από το γραφείο το χειμώνα. Στο ίδιο χρονικό διάστημα, μπορεί να ενημερώσετε τα μέλη της οικογένειάς σας να ετοιμαστούν για ένα δείπνο, έτσι ώστε όταν φτάσετε στο σπίτι, σας να τους παραλάβετε και να πάτε στο εστιατόριο. Το σίγουρο είναι ότι το ημερολογιακό στοιχείο του ξενοδοχείου προστίθεται ως μέλος της ομάδας σας, το οποίο σας δίνει τη δυνατότητα να κλείσετε ένα τραπέζι εγκαίρως.
- Μορφή της ομάδας: Δύο τύποι ομάδων εξετάζονται. Μακροχρόνιες ομάδες, όπου οι συμμετέχοντες μπορούν να εξελιχθούν επαναλαμβανόμενα και να επικοινωνούν σε ομάδες που μπορεί να παραμείνουν με την πάροδο του χρόνου, όπως είναι μια ομάδα επικοινωνίας για μια

ποδοσφαιρική ομάδα. Βραχυπρόθεσμες ομάδες, όπου μια ομάδα μπορεί να σχηματιστεί για προσωρινούς σκοπούς, όπως είναι η δημιουργία μιας σχέσης με ένα εμπορικό-mall όπου ένας χρήστης επισκέπτεται. Έτσι για να μπορέσει ο χρήστης να λαμβάνει όλες τις προσφορές με έκπτωση και την τοποθεσία των φίλων του στο εμπορικό κέντρο.

- Σκοποί: Σε γενικές γραμμές, υπάρχουν δημόσιοι και ιδιωτικοί σκοποί για τη διαμόρφωση μιας ομάδας, ωστόσο οι κοινωνικοί και οι παραγωγικοί σκοποί είναι επίσης σημαντικοί για τον σχηματισμό ομάδων. Κοινωνικοί σκοποί είναι για παράδειγμα, η πρόθεση του χρήστη να αυξήσει την αλληλεπίδραση μεταξύ των φίλων και της οικογένειάς του. Παραγωγικοί σκοποί περιλαμβάνουν την προσθήκη συσκευών σε μια ομάδα, για την αύξηση της παραγωγικότητας για μια συγκεκριμένη εργασία.

2.4.3 Το είδος της ενημέρωσης που ενδείκνυται

Το δεύτερο ζήτημα σχετίζεται με το είδος της ενημέρωσης που ενδείκνυται κάθε φορά. Για το σκοπό αυτό, οι ειδοποιήσεις διακρίνονται με γνώμονα την κατηγορία του μέσου που αξιοποιούν.

- **Ασύγχρονα μέσα**

Μετάδοση πληροφορίας σε διαφορετικές χρονικές στιγμές για κάθε διακριτό χρήστη. Ένα παράδειγμα, το οποίο χρησιμοποιείται κατά κόρον μέχρι και σήμερα, είναι το ηλεκτρονικό ταχυδρομείο (email).

Σε οργανισμούς υψηλής απόδοσης η πολυπλοκότητα των καθηκόντων απαιτεί εντατική ομαδική εργασία. Η έρευνα για CSCW διερευνά τις μεθόδους για την ηλεκτρονική υποστήριξη της συνεργασίας. Πιο παλιά η CSCW έρευνα επικεντρωνόταν κυρίως στη σύγχρονη συνεργασία (συνεργατικοί συντάκτες κειμένου σε πραγματικό χρόνο, άμεσα μηνύματα, βιντεοδιάσκεψεις, κ.λπ.). Εντυπωσιακά αποτελέσματα έχουν επιτευχθεί σε αυτόν τον τομέα, ωστόσο, στις μέρες μας, λόγω της ανάθεσης εργασιών σε τρίτα εξωτερικά μέλη της ομάδας τα οποία συχνά εργάζονται σε διαφορετικές ζώνες ώρας, άρχισε να ενδιαφέρει η παραγωγή εφαρμογών που υποστηρίζουν ασύγχρονη συνεργασία.

Δεδομένου ότι στην ασύγχρονη συνεργασία η άλλη πλευρά δεν μπορεί να εξηγήσει τις δραστηριότητές της, ένα βασικό ζήτημα για μια επιτυχή ασύγχρονη συνεργασία είναι η παροχή ευαισθητοποίησης αλλαγών: τη δυνατότητα να παρακολουθούνται οι αλλαγές που έγιναν σε ένα συνεργατικό τεχνούργημα σε κάποιο χώρο εργασίας από τους άλλους συμμετέχοντες κατά την πάροδο του χρόνου.

Τρέχον ασύγχρονα συνεργατικά εργαλεία υποστηρίζουν την ευαισθητοποίηση αλλαγών σε ένα χώρο εργασίας με δύο τρόπους. Μπορούν είτε να αποθηκεύουν και να δείξουν μια ιστορία από γεγονότα που έγιναν από την τελευταία επίσκεψη (συνειδητοποίηση των γεγονότων του παρελθόντος) ή επιτρέπουν την καταχώρηση ενδιαφέροντος σε ορισμένα γεγονότα και την ειδοποίηση όταν η περίπτωση ενδιαφέροντος λάβει χώρα (την ευαισθητοποίηση των μελλοντικών γεγονότων). [10]

- **Σύγχρονα μέσα**

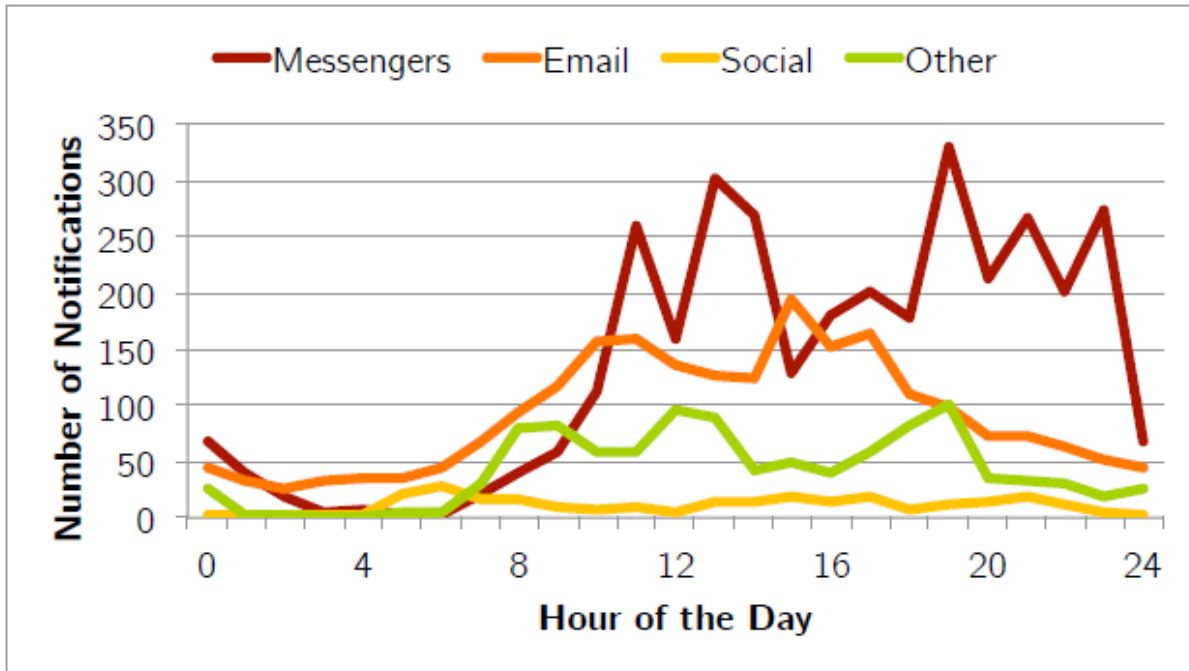
Σε αντίθεση με τα ασύγχρονα μέσα ενημερότητας, τα σύγχρονα μέσα είναι τα μέσα με τα οποία όλοι οι χρήστες λαμβάνουν την πληροφορία συγχρόνως. Παραδείγματα χρήσης σύγχρονων μέσων είναι το skype, με το οποίο πετυχαίνουμε σύγχρονη επικοινωνία μεταξύ 2 ή περισσότερων χρηστών.

Τα τελευταία χρόνια η χρήση των “έξυπνων” κινητών σαν σύγχρονο μέσο επικοινωνίας έχει αναπτυχθεί ραγδαία, τοποθετώντας τα στην πρώτη θέση στη λίστα σύγχρονων μέσων επικοινωνίας. Οι ειδοποιήσεις στα κινητά τηλέφωνα προειδοποιούν τους χρήστες για νέα μηνύματα, e-mail, ενημερώσεις κοινωνικών δικτύων, άλλες εκδηλώσεις και για αναβαθμίσεις εφαρμογών.

Μιας βδομάδας μελέτη, στην οποία συμμετείχαν 15 χρήστες με κινητά τηλέφωνα, όπου συλλέχθηκαν ειδοποιήσεις μέσω μιας smartphone εφαρμογής καταγραφής παράλληλων

υποκειμενικών αντιλήψεων αυτών των ειδοποιήσεων μέσω ενός online ημερολόγιου. Έδειξε ότι οι συμμετέχοντες είχαν να ασχοληθεί με 63,5 ειδοποιήσεις κατά μέσο όρο ανά ημέρα, ως επί το πλείστο από μηνύματα και e-mail.

Αν το τηλέφωνο είναι σε σιωπηλή λειτουργία ή όχι, οι ειδοποιήσεις συνήθως βλέπονται μέσα σε λίγα λεπτά. Η κοινωνική πίεση σε προσωπική επικοινωνία ήταν ανάμεσα στους κύριους λόγους. Ενώ ένας αυξανόμενος αριθμός των ειδοποιήσεων που συνδέθηκε με την αύξηση των αρνητικών συναισθημάτων, που λαμβάνουν περισσότερα μηνύματα και ενημερώσεις κοινωνικών δικτύων, έκανε τους συμμετέχοντες να αισθάνονται περισσότερο συνδεδεμένοι με άλλους.



Εικόνα 5: Notifications by hour of the day

Τα ευρήματά υποδηλώνουν ότι η αποφυγή των ειδοποιήσεων μπορεί να είναι βιώσιμη για επαγγελματική επικοινωνία, ενώ στην προσωπική επικοινωνία, οι προσεγγίσεις θα πρέπει επικεντρωθούν στη διαχείριση των προσδοκιών. [8]

Place	Time	
	Same	Different
	Synchronous / proximate	Anytime / same place (virtual)
	Synchronous / dispersed	Asynchronous / dispersed

Πίνακας 2: Time and place dispersion

Synchronous/ dispersed: Τα υποστηριζόμενα εργαλεία που κατατάσσονται σε αυτή την κατηγορία περιλαμβάνουν πραγματικό χρόνο, αλλά διασκορπισμένη τοποθεσία αλληλεπίδρασης. Τα εργαλεία αυτά περιλαμβάνουν συνομιλίες, βίντεο-διάσκεψης, εργαλεία που υποστηρίζουν βίντεο και ήχο, και οπτική αναπαράσταση που μπορεί να δείξει τις καταστάσεις των άλλων, τις κινήσεις και τη θέση τους.

Asynchronous / dispersed: Τα υποστηριζόμενα εργαλεία που κατατάσσονται σε αυτή την κατηγορία περιλαμβάνουν διάφορα χρόνου και αλληλεπίδραση διασκορπισμένης τοποθεσίας. Τα εργαλεία αυτά περιλαμβάνουν ηλεκτρονικό ταχυδρομείο, συνεδριακές εγκαταστάσεις ηλεκτρονικών υπολογιστών (Hiltz & Wellam, 1997), το οποίο μπορεί να υποστηρίξει τους ανθρώπους να συνεργάζονται με επιτυχία όταν διανέμονται σε διαφορετικούς χρόνους και θέσεις [11].

- **Υπηρεσίες (calendars, social software) με την ενημερότητα ως δευτερεύουσα ή φωλιασμένη λειτουργία**

Πολλές υπηρεσίες χρησιμοποιούν την ενημερότητα ως δευτερεύουσα ή φωλιασμένη λειτουργία. Ένα παράδειγμα είναι η TopHost, η οποία έχει ως κύρια λειτουργία της την παροχή υπηρεσιών στους χρήστες – πελάτες της και σαν δευτερεύουσα αλλά και φωλιασμένη λειτουργία έχει την ενημερότητα, καθώς χρησιμοποιεί και σύγχρονα και ασύγχρονα μέσα για αυτήν. Η υπηρεσία αυτή χρησιμοποιεί και σύγχρονη αλλά και ασύγχρονη επικοινωνία. Ως σύγχρονη επικοινωνία το IM (Instant Messaging) και σαν ασύγχρονη το e-mail. Ένα άλλο παράδειγμα είναι η υπηρεσία FareCompare, στην οποία ο χρήστης αναζητά αεροπορικά εισιτήρια και επίσης έχει την επιλογή να ενημερωθεί για αλλαγή στις τιμές για τον προορισμό που έχει επιλέξει. Η υπηρεσία αυτή χρησιμοποιεί την ενημερότητα ως δευτερεύουσα λειτουργία καθώς πρωτεύουσα λειτουργία έχει την αναζήτηση και αγορά εισιτηρίων.

2.5 Τεχνολογικά ζητήματα

Έχοντας παρουσιάσει συνοπτικά το ευρύτερο αντικείμενο της τρέχουσας εργασίας, η ενότητα αυτή επικεντρώνεται σε ορισμένα τεχνολογικά ζητήματα που σχετίζονται με την υλοποίηση της εφαρμογής ομαδικών ειδοποιήσεων. Ειδικότερα, θα αναφερθούμε στην κατηγορία εφαρμογών πολυτοποίησης δεδομένων ή mashups και τις σχεδιαστικές γλώσσες που προσφέρονται σήμερα για την ανάπτυξη τέτοιων εφαρμογών.

2.5.1 Mashups

Ο όρος mashup προσδιορίζει μια γενιά διαδραστικών εφαρμογών Web που βασίζεται σε περιεχόμενο που ανακτάται από εξωτερικές πηγές δεδομένων και αξιοποιείται για τη δημιουργία εντελώς νέων και καινοτόμων υπηρεσιών. Με άλλα λόγια, πρόκειται για εφαρμογές που η λειτουργία τους είναι άρρηκτα συνδεδεμένη με δεδομένα ή/και πληροφορία που προέρχεται από εξωτερικές πηγές / εφαρμογές υιοθετώντας ένα προδιαγεγραμμένο πρωτόκολλο επικοινωνίας. Συνδυάζοντας περισσότερες από μια τέτοιες εφαρμογές μας προκύπτει η δημιουργία νέων υπηρεσιών. Πρόκειται για ένα χαρακτηριστικό γνώρισμα της δεύτερης γενιάς των εφαρμογών Web, η οποία είναι ανεπίσημα γνωστή ως Web 2.0.

Μια mashup εφαρμογή αποτελείται από τρία διαφορετικά κομμάτια τα οποία είναι λογικά και φυσικά ασύνδετα μεταξύ τους (είναι πιθανό να διαχωρίζονται τόσο από το δίκτυο όσο και από οργανωτικά όρια):

- **The API/content providers:** Πρόκειται για υπηρεσίες παρόχους περιεχομένου που χρησιμοποιείται από το Mashup. Στην εργασία αυτή, για παράδειγμα, οι πάροχοι είναι οι υπηρεσίες της Google. Για να διευκολυνθεί η ανάκτηση των δεδομένων ή για την ενσωμάτωση μιας λειτουργίας, οι πάροχοι συχνά εκθέτουν το περιεχόμενό τους μέσω Web-πρωτοκόλλων όπως REST, Web Services, και RSS / Atom, Java Servlets (περιγράφεται παρακάτω). Ωστόσο, πολλές ενδεχομένως ενδιαφέρουσες πηγές δεδομένων δεν παρέχουν (ακόμη) ευκολία στην χρήση των APIs τους. Mashups που αξιοποιούν περιεχόμενο από ιστοσελίδες όπως η Wikipedia, TV Guide, και σχεδόν όλες οι ιστοσελίδες της κυβέρνησης και τους δημόσιου τομέα το κάνουν με μια τεχνική γνωστή ως screen scraping. Σε αυτό το πλαίσιο, η τεχνική screen scraping υποδηλώνει τη διαδικασία με την οποία ένα εργαλείο προσπαθεί να εκμαιεύσει πληροφορίες από τον πάροχο του περιεχομένου, προσπαθώντας να αναλύσει τις σελίδες Web του παρόχου, οι οποίες αρχικά προορίζονταν για τον άνθρωπο.
- **The mashup site:** Είναι ο χώρος που φιλοξενεί τη mashup σελίδα. Πρόκειται για ένα αρκετά ενδιαφέρον τμήμα καθώς είναι το σημείο όπου η λογική του mashup κατοικεί, αλλά όχι κατ' ανάγκη το σημείο όπου εκτελείται. Από τη μια πλευρά, τα mashups μπορούν να υλοποιηθούν με παρόμοιο τρόπο όπως οι παραδοσιακές εφαρμογές Web, με τη χρήση

server-side τεχνολογιών δυναμικής γενιάς του περιεχομένου, όπως Java servlets, CGI, PHP ή ASP. Από την άλλη, το περιεχόμενο του mashup μπορεί να παράγεται απευθείας μέσα από το πρόγραμμα περιήγησης του πελάτη μέσω client-side scripting (δηλαδή, JavaScript) ή applets. Η client-side λογική είναι συχνά ο συνδυασμός του άμεσα ενσωματωμένου κώδικα στις σελίδες του mashup στο Web, καθώς και scripting βιβλιοθήκες API ή applets (τα οποία παρέχονται από τους παρόχους περιεχομένου) που αναφέρονται μέσα από τις ιστοσελίδες τους. Εφαρμογές mashups που χρησιμοποιούν αυτή την προσέγγιση μπορούν να χαρακτηριστούν ως πλούσιες εφαρμογές Internet (Rich Internet Applications RIAs), που σημαίνει ότι είναι πολύ προσανατολισμένη προς τη διαδραστική εμπειρία του χρήστη. Ας σημειωθεί ότι 'πλούσιες' εφαρμογές Διαδικτύου είναι ένα σήμα κατατεθέν για αυτό που σήμερα ονομάζεται Web 2.0. Τα οφέλη της πολτοποίησης των APIs στην πλευρά του πελάτη είναι ότι η υπάρχει λιγότερη επιβάρυνση για τον διακομιστή του mashup (τα δεδομένα μπορούν να ανακτηθούν απευθείας από τον πάροχο περιεχομένου) και ότι είναι πιο ομαλή η εμπειρία χρήστη (τμηματικές ενημερώσεις περιεχομένου στις σελίδες χωρίς να χρειάζεται η ανανέωση ολόκληρης της σελίδας). Συχνά οι mashups εφαρμογές χρησιμοποιούν ένα συνδυασμό και των δύο λογικών server και client-side για να επιτευχθεί η συνάρθρωση των δεδομένων τους.

- **The client's Web browser:** Αυτό είναι όπου η εφαρμογή καθίσταται γραφικά και όπου η αλληλεπίδραση του χρήστη λαμβάνει χώρα. Όπως περιγράφεται παραπάνω, mashups συχνά χρησιμοποιούν client-side λογική για να συγκεντρώσουν και να συνθέσουν το περιεχόμενο.

Όσο αφορά τις τεχνολογίες που συνιστούν το συνθετικό ιστό μιας mashup εφαρμογής, αυτές ποικίλουν και περιλαμβάνουν τις ακόλουθες:

- **SOAP (Simple Object Access Protocol) and REST (Representational State Transfer):** Τόσο το SOAP όσο και το REST είναι πρωτόκολλα ουδέτερης πλατφόρμας για την επικοινωνία με απομακρυσμένες υπηρεσίες. Ως μέρος του μοντέλου αρχιτεκτονικής service-oriented, οι πελάτες μπορούν να χρησιμοποιήσουν το SOAP και το REST για να αλληλεπιδρούν με απομακρυσμένες υπηρεσίες χωρίς τη γνώση της πλατφόρμας της εφαρμογής τους: η λειτουργικότητα της υπηρεσίας αποδίδεται πλήρως από την περιγραφή των μηνυμάτων που ζητά και από την απάντηση που δέχεται.
- **RSS (Rich Site Summary) and ATOM:** RSS είναι μια οικογένεια μορφοποιημένων αναδημοσιεύσεων βασισμένη σε XML. Με τον όρο αναδημοσίευση εννοούμε, όταν μια ιστοσελίδα θέλει να διανέμει το περιεχόμενό της, δημιουργεί ένα έγγραφο RSS και καταχωρεί το έγγραφο με έναν εκδότη RSS. Στη συνέχεια, ένας RSS πελάτης μπορεί να ελέγξει την τροφοδοσία του εκδότη για το νέο περιεχόμενο και να το διαχειριστεί αναλόγως. Το Atom είναι ένα νεότερο, αλλά παρόμοιο με το RSS πρωτόκολλο αναδημοσίευσης. Είναι ένα προτεινόμενο πρότυπο της Internet Engineering Task Force (IETF) και επιδιώκει να διατηρεί καλύτερα μεταδεδομένα από το RSS, παρέχει καλύτερη και πιο αυστηρή τεκμηρίωση, και ενσωματώνει την έννοια των δομών για κοινή αναπαράσταση δεδομένων. Αυτές οι τεχνολογίες αναδημοσίευσης αξιοποιούνται κυρίως από mashups που συναθροίζουν ειδήσεις ή blogs.
- **Java Servlet:** Είναι ένα πρόγραμμα Java που επεκτείνει τις δυνατότητες ενός διακομιστή. Μπορεί να ανταποκριθεί σε οποιουδήποτε τύπο αιτημάτων, οι οποίοι συνήθως εκτελούνται από τις διαδικτυακές εφαρμογές που φιλοξενούνται σε διακομιστές Web. Επικοινωνούν με οποιοδήποτε πρωτόκολλο πελάτη-διακομιστή, αλλά πιο συχνά χρησιμοποιείται η επικοινωνία με το πρωτόκολλο HTTP. Έτσι, ένας προγραμματιστής λογισμικού μπορεί να χρησιμοποιήσει ένα servlet για να προσθέσει δυναμικό περιεχόμενο σε έναν web server, χρησιμοποιώντας την πλατφόρμα της Java. Servlets μπορούν να παραχθούν αυτόματα από Java Server Pages (JSP) από τον JavaServer Pages compiler. Η διαφορά μεταξύ των servlets και των JSP είναι ότι οι servlets συνήθως ενσωματώνουν

HTML στο κώδικα Java, ενώ οι JSPs ενσωματώνουν κώδικα Java στην HTML. Μια χρήση, η οποία αξιοποιήθηκε και στα πλαίσια της τρέχουσας εργασίας είναι ο συνδυασμός servlets με JSPs και αποτελεί μια γεύση του Model-View-Controller (MVC). [12]

- **Screen scraping:** Όπως αναφέρθηκε προηγουμένως, η έλλειψη των APIs από τους παρόχους περιεχομένου συχνά αναγκάζει τους προγραμματιστές mashup να καταφύγουν στην τεχνική screen scraping, προκειμένου να ανακτήσουν τις πληροφορίες που αναζητούν να πολτοποιήσουν. Είναι η διαδικασία χρήσης εργαλείων λογισμικού για την ανάλυση του περιεχομένου που γράφτηκε αρχικά για την κατανάλωση από τον άνθρωπο προκειμένου να εξαχθούν σημασιολογικές δομές δεδομένων που αντιπροσωπεύουν την εν λόγω πληροφορία που μπορεί να χρησιμοποιηθεί και να χειραγωγηθεί μέσω του προγραμματισμού. Πολλά mashups χρησιμοποιούν την τεχνική αυτή για ανάκτηση δεδομένων, ειδικά όταν η ανάκτηση των δεδομένων γίνεται από δημόσιους τομείς.

2.5.2 Design Languages για υπηρεσίες ομαδικής ενημέρωσης

Μια σχεδιαστική γλώσσα ή λεξιλόγιο σχεδιασμού είναι ένα γενικό σχήμα, που καθοδηγεί στο σχεδιασμό μιας σουίτας προϊόντων ή αρχιτεκτονικών ρυθμίσεων. Σχεδιαστές που επιθυμούν να δώσουν στη δική τους σουίτα προϊόντων μια μοναδική, αλλά συνεπή εμφάνιση και αίσθηση ορίζουν μια γλώσσα σχεδιασμού για αυτό, η οποία μπορεί να περιγράψει τις επιλογές για τις πτυχές του σχεδιασμού, όπως υλικά (materials), συνδυασμούς χρωμάτων (color schemes), σχήματα (shapes), πρότυπα (patterns), επιφάνειες (textures), ή διατάξεις (layouts). Στη συνέχεια ακολουθεί ο σχεδιασμός κάθε αντικειμένου στη σουίτα. Συνήθως, οι σχεδιαστικές γλώσσες δεν είναι αυστηρά καθορισμένες, δηλαδή ο σχεδιαστής κάνει ένα πράγμα με παρόμοιο τρόπο όπως ένα άλλο. Σε άλλες περιπτώσεις είναι, έτσι ώστε τα προϊόντα να αποκτήσουν μια ισχυρή θεματική ποιότητα. Για παράδειγμα, αν και υπάρχει μεγάλη ποικιλία στη σχεδίαση ενός σκακιού, τα κομμάτια μέσα σε αυτό το σύνολο είναι συνήθως θεματικά συνεπείς. Στο σενάριο των γραφικών διεπαφών χρήστη (GUIs), κατευθυντήριες γραμμές για τη διεπαφή μπορούν να θεωρηθούν ως γλώσσες σχεδιασμού για εφαρμογές.

2.5.2.1 Microsoft Design Language: Metro

Είναι μια σχεδιαστική γλώσσα που δημιουργήθηκε από τη Microsoft κατά κύριο λόγο για τις διεπαφές χρήστη. Η σχεδιαστική γλώσσα αυτή έχει ενσωματωθεί σε πολλά προϊόντα της εταιρείας, όπως του λογισμικού συστήματος Xbox 360, Xbox One, τα Windows 8, Windows Phone, και Outlook.com.

2.5.2.2 IBM Design Language

Η γλώσσα αυτή είναι ένα σύνολο κατευθυντήριων γραμμών διαβίωσης που επικοινωνεί με προϊόντα της συγκεκριμένης εταιρείας. Η γλώσσα έχει σχεδιαστεί για να είναι πιο πολύ διδακτική παρά καθοδηγητική. Η προσπάθεια δημιουργίας της γλώσσας αυτής είναι σχετικά νέα και σκοπίμως δημιουργημένη για να εξελιχθεί μέσω της ανατροφοδότησης από τις ομάδες παραγωγής και των χρηστών.

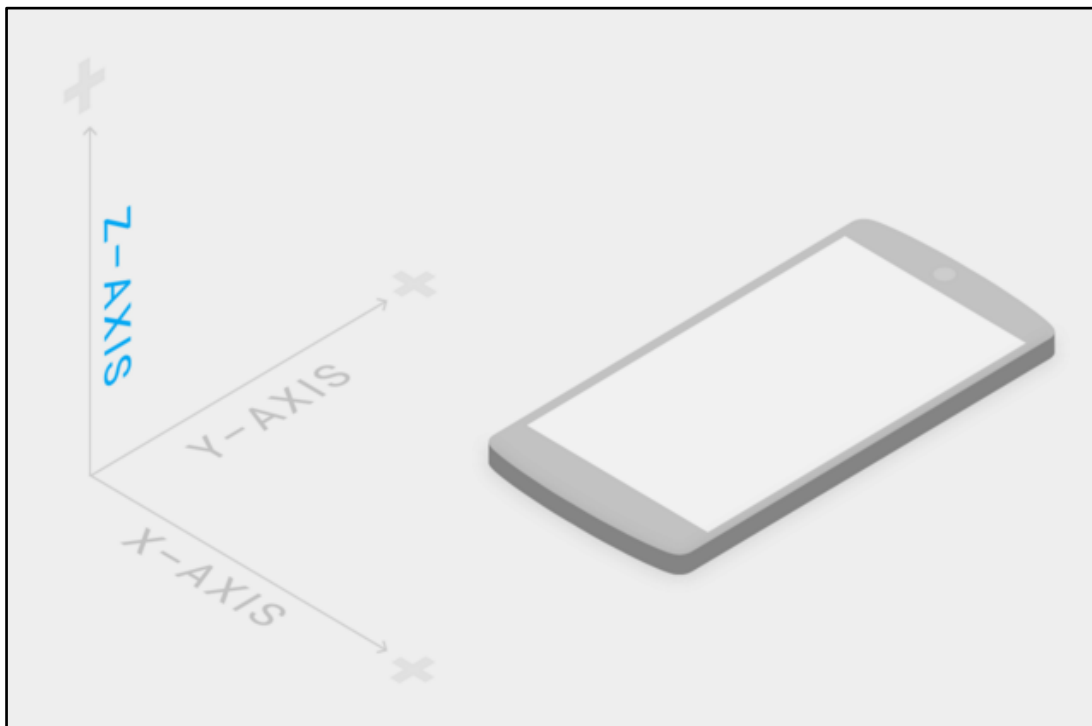
2.5.2.3 Snow White Design Language

Η σχεδιαστική γλώσσα αυτή ενίσχυσε την παγκόσμια φήμη της Apple, δίνοντας σχεδιαστικές τάσεις στη βιομηχανία των υπολογιστών, και έτσι διαμορφώθηκε η αντίληψη των υπολογιστών στην κατασκευή και στον επιχειρηματικό κόσμο.

2.5.2.4 Google Material Design

Το Material Design είναι ένα σύνολο σχεδιαστικών κανόνων (design language) που δημιουργήθηκε από τη Google. Επεκτείνει το μοντέλο των "καρτών" (πρωτοεμφανίστηκε στο Google Now) που είναι σχεδιασμός, βασισμένος σε διάταξη πίνακα (grid layout), με κινούμενα στοιχεία και μεταβάσεις που ανταποκρίνονται στις κινήσεις των χρηστών, και με χρήση padding και εφέ βάθους με κατάλληλο φωτισμό και σκίαση των αντικειμένων. Το material έχει φυσικές επιφάνειες και άκρα. Οι σκιές δίνουν νόημα σε αυτό που αγγίζεις. Η Google υποστηρίζει ότι η καινούρια σχεδιαστική τους γλώσσα βασίζεται στο χαρτί και το μολύβι.

Το Material Design σταδιακά επεκτείνεται σε όλα τα προϊόντα της Google τόσο στο web όσο και στο mobile, προσφέροντας ομαλή εμπειρία σε όλες τις πλατφόρμες και εφαρμογές της. Η Google έχει κυκλοφορήσει επίσης και APIs (Application Programming Interfaces) για τους προγραμματιστές που θέλουν να ενσωματώσουν την σχεδιαστική αυτή γλώσσα στις εφαρμογές τους. Το υλικό περιβάλλον είναι ένας 3D χώρος, επομένως όλα τα αντικείμενα στον χώρο αυτόν έχουν x, y και z διαστάσεις. Ο άξονας z είναι κάθετα ευθυγραμμισμένος με το επίπεδο της οθόνης, με τις θετικές τιμές του άξονα να εκτείνονται προς τον παρατηρητή. Κάθε φύλλο του υλικού καταλαμβάνει μια ενιαία θέση κατά μήκος του άξονα z και έχει ένα σταθερό πάχος του 1dp, που ισοδυναμεί με ένα pixel του πάχους στις οθόνες με πυκνότητα pixel 160. Στο διαδίκτυο, ο άξονας z χρησιμοποιείται για layering και όχι για την προοπτική του υλικού. Ο 3D κόσμος προσομοιάζεται από το χειρισμό του άξονα y.



Εικόνα 6: Material perspective

3 Μηχανισμός ομαδικών ενημερώσεων

Η δημιουργία εφαρμογών χρησιμοποιώντας καθιερωμένες μεθόδους απαιτεί συχνά μια τεράστια προσπάθεια για ανάπτυξη. Μία από τις μεγαλύτερες προκλήσεις είναι να βρούμε τις προσεγγίσεις που επιτρέπουν την οικονομικά αποδοτική ανάπτυξη εφαρμογών για πολλαπλές συσκευές με ποικίλα τεχνικά χαρακτηριστικά. Μια τέτοια νέα προσέγγιση είναι η συγγραφή ενιαίου περιεχομένου web σε μια ανεξάρτητη συσκευής γλώσσα σήμανσης (device independent markup language), η οποία προσαρμόζεται για να καλύπτει τα ειδικά χαρακτηριστικά της συσκευής που έχει πρόσβαση.

Με τον όρο device independence εννοούμε τη διαδικασία δημιουργίας μιας εφαρμογής λογισμικού η οποία είναι σε θέση να λειτουργήσει σε ένα ευρύ φάσμα συσκευών, ανεξάρτητα από το υλικό στο οποίο χρησιμοποιείται το λογισμικό. Αντιθέτως, device dependent εννοούμε το λογισμικό που απευθύνεται σε συγκεκριμένα χαρακτηριστικά του υλικού και λειτουργεί με ένα μόνο είδος περιφερειακής συσκευής.

Με τον όρο μηχανισμός ομαδικών ενημερώσεων αναφερόμαστε σε μια πλατφόρμα η οποία έχει σαν κύρια λειτουργία της την παράδοση ενός μηνύματος σε μια μικρή ή μεγάλη ομάδα ανθρώπων. Κατά παράδοση, αυτά τα συστήματα προσφέρουν μονόδρομη ενημέρωση, δηλαδή παράδοση μηνυμάτων μέσω e-mail ή μήνυμα κειμένου στο κινητό. Η πτυχιακή εργασία αναφέρεται σε ένα μηχανισμό ομαδικών ενημερώσεων ο οποίος χρησιμοποιεί την αποστολή e-mail για την αποστολή των μηνυμάτων.

Ο μηχανισμός προσφέρει τα εξής:

- Δυνατότητα σύνδεσης / πρόσβασης από οποιοδήποτε υπολογιστή ή κινητό μέσω του web browser δίχως απαιτήσεις για ειδικές εφαρμογές.
- Τα μηνύματα μπορούν να σταλούν άμεσα ή σε κάποια καθορισμένη στιγμή στο μέλλον.
- Αξιοποίηση ημερολογίου για την δημιουργία γεγονότος, υποστηρίζοντας κατ' αυτόν τον τρόπο τις ειδοποιήσεις ως περιφερειακή - δευτερεύουσα λειτουργία (γνωσιακή ενίσχυση μελών ομάδων).
- Ικανότητα διαχείρισης ειδοποιήσεων που αποστέλλονται σε εκατοντάδες ανθρώπους σε δευτερόλεπτα.
- Εξειδικευμένο API, το οποίο επιτρέπει σε εφαρμογές τρίτων να χρησιμοποιούν τις υπηρεσίες που προσφέρονται.

3.1 Ειδικοί στόχοι

Στόχος της πτυχιακής εργασίας είναι να μελετήσει βασικές τεχνολογίες της τρέχουσας τεχνολογικής τροχιάς και να αναπτύξει μια ενδεικτική εφαρμογή ειδοποίησης ομάδων που να αξιοποιεί πολλαπλά εναλλακτικά κανάλια ενημέρωσης όπως το ηλεκτρονικό ταχυδρομείο και υπηρεσίες ημερολογίου.

Οι εφαρμογές ειδοποίησης είναι ένας συνδυασμός υλικού και λογισμικού που παρέχουν τη δυνατότητα προώθησης ενός μηνύματος σε ένα σύνολο παραληπτών. Τα συστήματα αυτά αποτελούν μια σημαντική πτυχή στις σύγχρονες εφαρμογές Web. Στη παρούσα εργασία προσβλέπουμε στην ανάπτυξη ενός ευέλικτου μηχανισμού ειδοποιήσεων που να επιτρέπει:

- Αξιοποίηση κοινών επαφών μελών της ομάδας.
- Αλληλεπίδραση μεταξύ μελών μιας ομάδας με διαφορετικά μέσα και υπηρεσίες π.χ. ημερολόγια ανάλογα με το καθήκον που επιτελούν.
- Αυξημένη κινητικότητα των μελών που καθιστούν απαραίτητη την προσαρμογή της ενημέρωσης σε διαφορετικές τερματικές συσκευές.

Η επεκτασιμότητα του μηχανισμού εδράζεται στην απαίτηση εναρμονισμού του με καθιερωμένα πρότυπα του διαδικτύου έτσι ώστε να καθίσταται δυνατή η προσαρμογή του σε διαφορετικά περιβάλλοντα χρήσεις, τερματικές συσκευές και υπηρεσίες. Επιπλέον, μια τέτοια δυνατότητα επιτρέπει την αξιοποίηση εναλλακτικών καναλιών ενημέρωσης. Για τις ανάγκες της πτυχιακής μελετήθηκαν κανάλια ενημέρωσης όπως τα παρακάτω:

- Το email το οποίο εξακολουθεί να είναι σημαντικό κανάλι ασύγχρονης επικοινωνίας.
- Το ημερολόγιο το οποίο αφενός υιοθετεί μια χρονική μεταφορά αλληλεπίδρασης ευρέως χρησιμοποιούμενη και σχεδόν καθολικά αποδεκτή και αφετέρου αξιοποιείται από πλήθος εφαρμογών οι οποίες βασίζονται σε ημερολόγια για να προσδιορίσουν χρονική ακολουθία καθηκόντων.
- Υπηρεσίες τρίτων που μπορούν να αξιοποιήσουν τον μηχανισμό που προτείνεται στη παρούσα εργασία για να υλοποιήσουν τους δικούς τους εξειδικευμένους μηχανισμούς ενημερότητας.

3.1.1 Περιφερειακή – Δευτερεύουσα λειτουργία

Από σχήμα ειδοποίησης σε event στον Calendar. Μέχρι σήμερα είδαμε ότι από ένα συμβατικό Google Calendar υπάρχει η δυνατότητα αποστολής ειδοποίησης (π.χ. email) για την ενημέρωση των guest στο συγκεκριμένο event. Ο μηχανισμός αυτός έρχεται να αντιστρέψει τους ρόλους της λειτουργίας αυτής και να παρέχει την δυνατότητα δημιουργίας event σε κάποιον συμβατικό Google Calendar και μη.

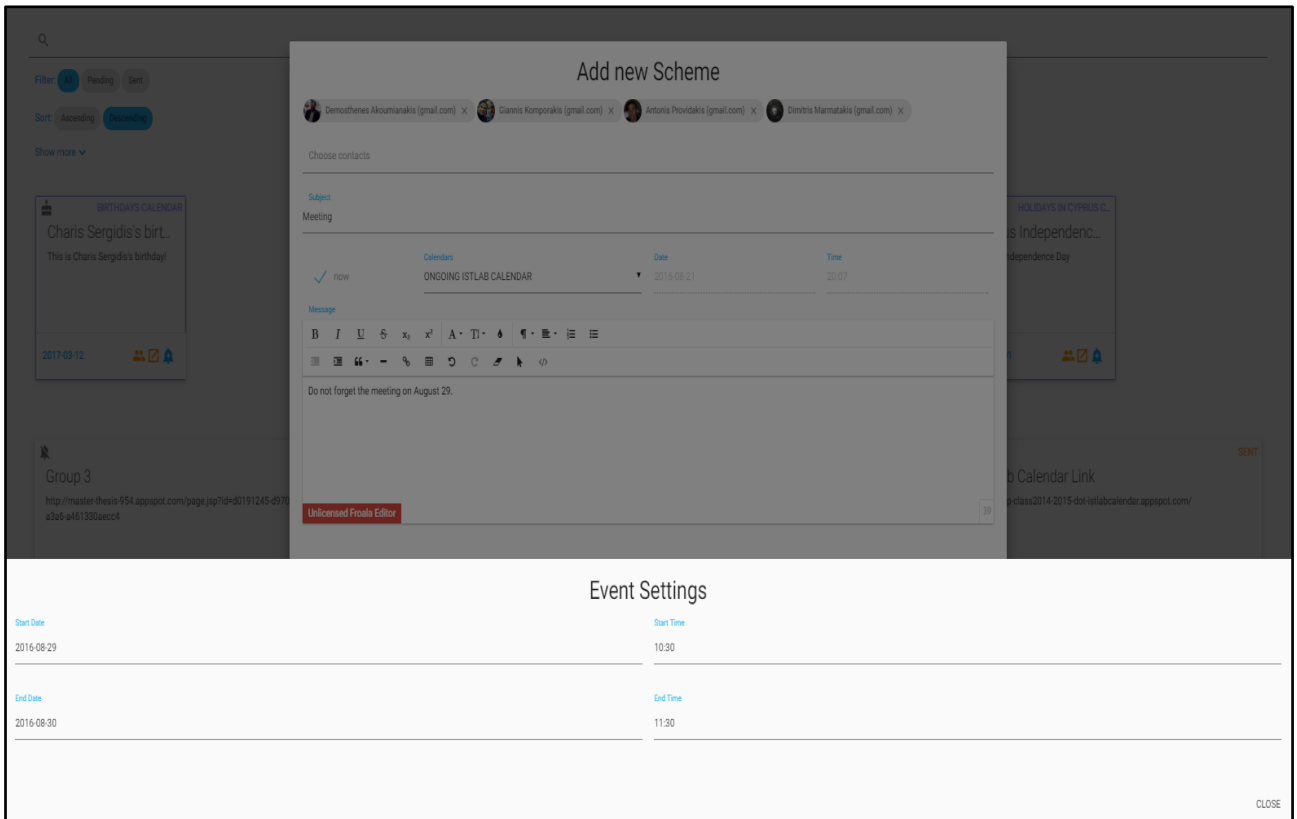
Συγκεκριμένα ο μηχανισμός επιτρέπει στο χρήστη να δημιουργήσει ένα σχήμα ειδοποίησης και μέσα από αυτό να επιλέξει κάποιον από τους Google Calendar του για να δημιουργήσει την συγκεκριμένη ειδοποίηση σαν event. Πιο κάτω (βλ. Εικόνα 8) αναπαρίσταται η δημιουργία ενός σχήματος ειδοποίησης και η επιλογή του Google Calendar για την δημιουργία του event.

Η μετάβαση από το σχήμα ειδοποίησης σε event γίνεται με τον εξής τρόπο:



Εικόνα 7: From email to calendar event

- Οι επιλεγμένοι παραλήπτες στο σχήμα ειδοποίησης, γίνονται guest στο event που θα δημιουργηθεί.
- Το θέμα του σχήματος γίνεται ο τίτλος του event.
- Το μήνυμα που θα αποσταλεί γίνεται το description του event.
- Από το παράθυρο που εμφανίζεται μετά από την επιλογή του Google Calendar επιλέγεται η ημερομηνία και η ώρα που θα λάβει χώρα το event (βλ. Εικόνα 8).



Εικόνα 8: Peripheral - Secondary Operation

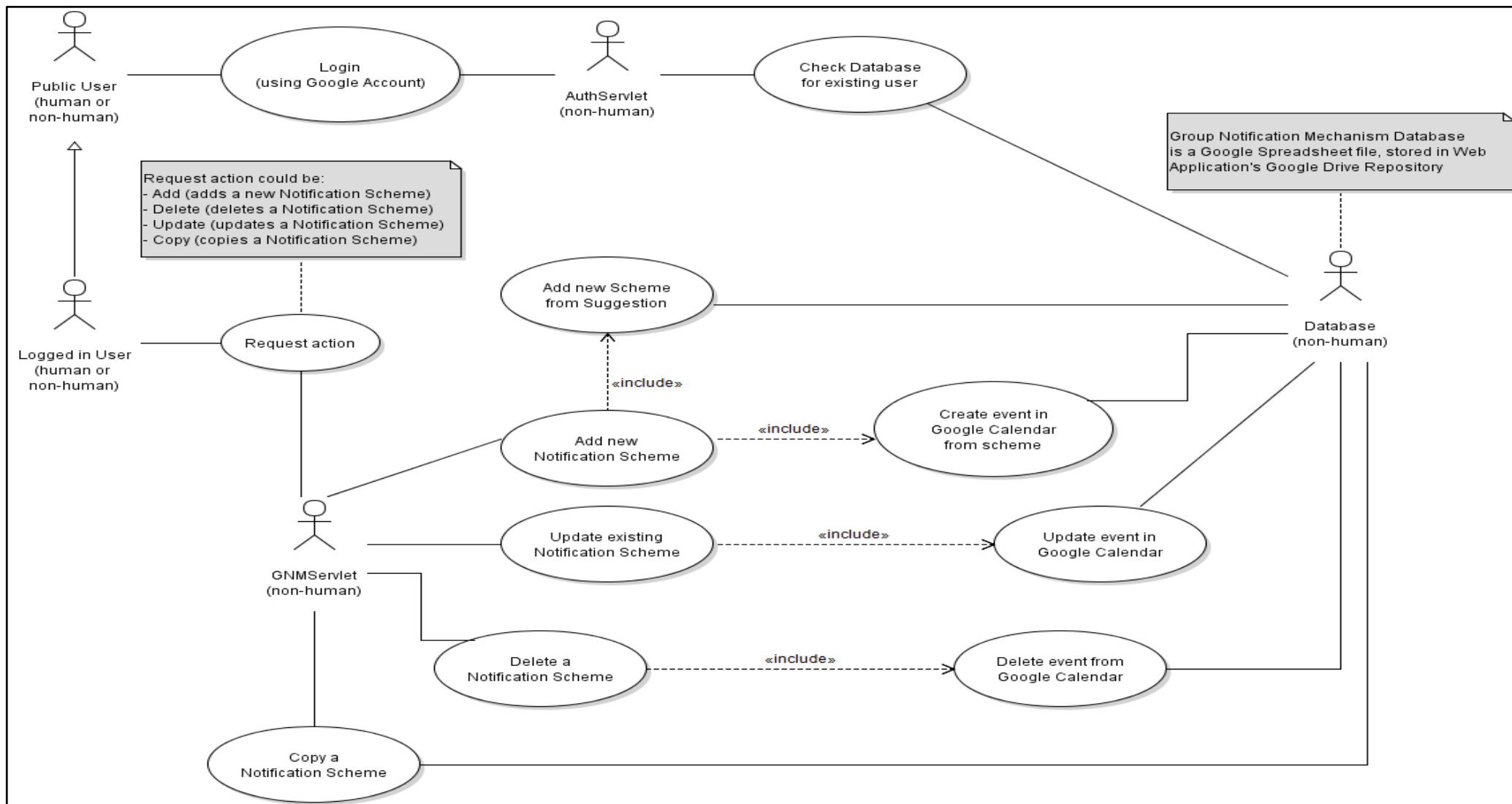
3.2 Τεχνολογικά εργαλεία και δεσμεύσεις

3.2.1 UseCase Diagram

Το διάγραμμα περιπτώσεων χρήσης χρησιμοποιείται για να μοντελοποιήσει το πλαίσιο λειτουργίας του συστήματος καθώς και τις προδιαγραφές του. Είναι μια αναπαράσταση των λειτουργικών απαιτήσεων του συστήματος που αποτυπώνει τις βασικές λειτουργίες του και δείχνει τη σχέση μεταξύ αυτών και του χρήστη. Κατά συνέπεια μπορεί να προσδιορίσει τους διαφορετικούς τύπους χρηστών ενός συστήματος και τις διάφορες περιπτώσεις χρήσης και συχνά συνοδεύεται από άλλους τύπους διαγραμμάτων. Στην εικόνα 9 παρουσιάζεται ένα γενικό διάγραμμα περίπτωσης χρήσης της εφαρμογής το οποίο αναπαριστά ένα γενικό σενάριο. Αναλυτικότερα το διάγραμμα αποτυπώνει:

1. Ένας άγνωστος χρήστης εισέρχεται στο σύστημα της εφαρμογής.
2. Κάνει είσοδο με το Google λογαριασμό του. Αν είναι authenticated χρήστης τότε επιτυγχάνεται η είσοδος, αλλιώς ένα παράθυρο authentication εμφανίζεται και ζητά πρόσβαση σε κάποιες Google υπηρεσίες του χρήστη (όπως είναι το Google Calendar, το Google Contacts, κ.α.).
3. Στη συνέχεια, γίνεται έλεγχος στη βάση (Google Spreadsheet) για τον χρήστη και η ανάκτηση των ήδη υπάρχων σχημάτων ειδοποίησης.
4. Ως συνδεδεμένος χρήστης, εμφανίζονται τα σχήματα ειδοποίησης που έχει δημιουργήσει καθώς και προτάσεις για καινούργια σχήματα από ανερχόμενα events στον Google Calendar του.

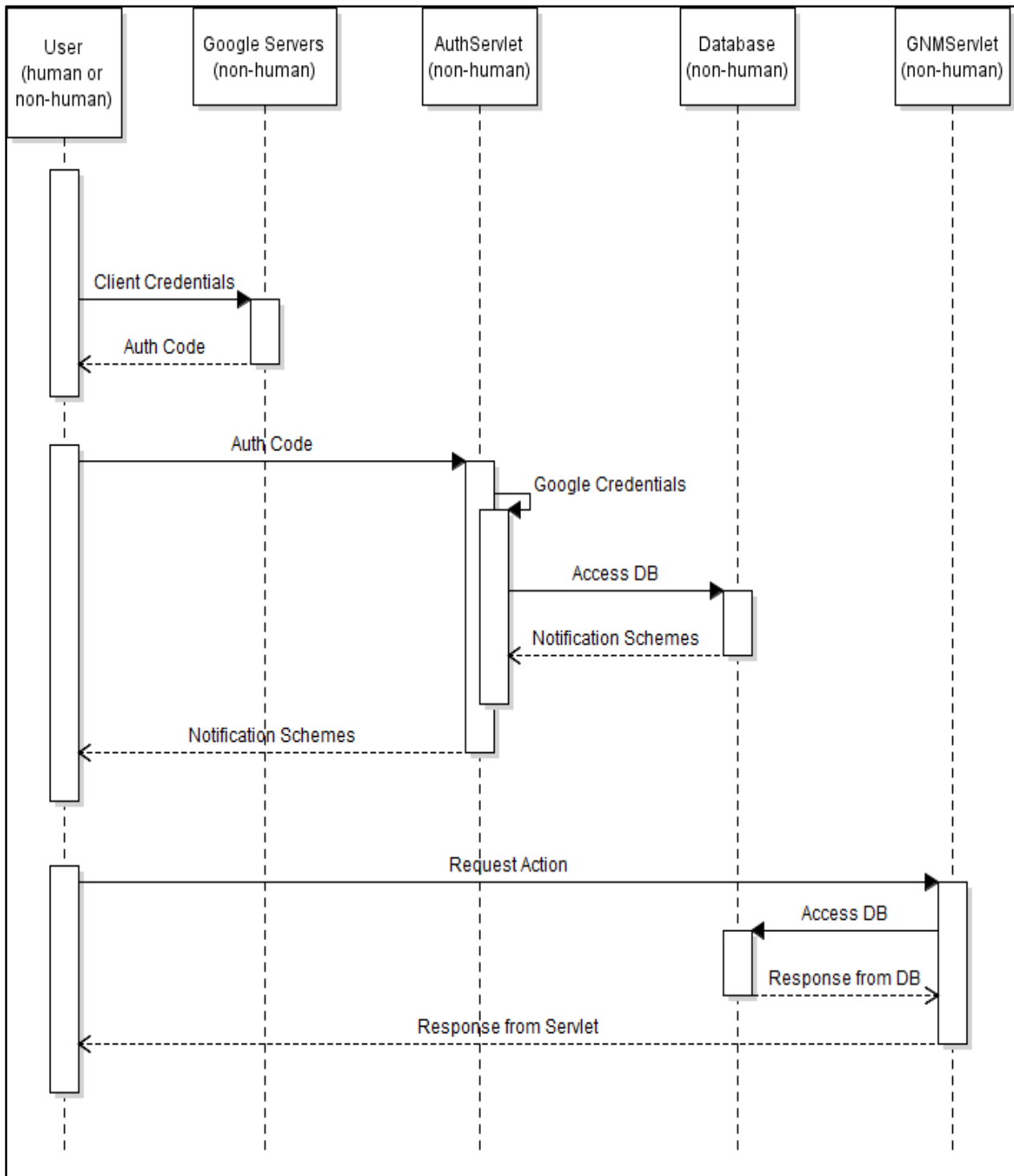
Τέλος, ο χρήστης έχει τη δυνατότητα να επεξεργαστεί τα σχήματα ειδοποίησης που έχει δημιουργήσει νωρίτερα, να διαγράψει κάποια σχήματα, να προσθέσει καινούργια σχήματα και να αντιγράψει σχήματα. Όλες αυτές οι δυνατότητες επιτυγχάνονται με τη χρήση AJAX requests στον GNMServlet , ο οποίος με τη σειρά του εκτελεί τις εντολές που χρειάζονται ανάλογα με το request, έπειτα επικοινωνεί με τη βάση της εφαρμογής για να γίνουν οι αντίστοιχες ενημερώσεις και τέλος στέλνει ένα μήνυμα στο front-end ως απάντηση στο request που έχει γίνει.



Εικόνα 9: UseCase Diagram

3.2.2 Sequence Diagram

Ένα διάγραμμα ακολουθίας είναι ένα διάγραμμα αλληλεπίδρασης που δείχνει πώς τα αντικείμενα λειτουργούν με το ένα το άλλο και με ποια σειρά. Δείχνει τις αλληλεπιδράσεις των αντικειμένων συναρτήσει του χρόνου. Απεικονίζει τα αντικείμενα και τις κλάσεις που εμπλέκονται στο σενάριο και την ακολουθία των μηνυμάτων που ανταλλάσσονται μεταξύ των αντικειμένων που απαιτούνται για τη διεξαγωγή της λειτουργικότητας του σεναρίου. Τα διαγράμματα ακολουθίας συχνά αποκαλούνται διαγράμματα εκδηλώσεων ή σενάρια εκδηλώσεων.



Εικόνα 10: Sequence Diagram

Το πιο πάνω διάγραμμα ακολουθίας δείχνει τα βήματα που εκτελούνται όταν ένας εξουσιοδοτημένος χρήστης διαχειρίζεται τα σχήματα ειδοποίησης του. Αναλυτικότερα:

- Στην χρονική στιγμή 0, ο χρήστης επιθυμεί να συνδεθεί στην εφαρμογή, με την βοήθεια κάποιων Client Credentials γίνεται το authentication με τους servers της Google και επιστρέφεται ένας κωδικός auth.
- Στην χρονική στιγμή 1, ο κωδικός αυτός αποστέλλεται στο back-end της εφαρμογής, συγκεκριμένα στον AuthServlet, όπου δημιουργεί Google Credentials και στη συνέχεια επικοινωνεί με τη βάση για να ανακτήσει τα σχήματα ειδοποίησης του χρήστη. Τα σχήματα αυτά επιστρέφονται στο front-end της εφαρμογής. Έπειτα ο συνδεδεμένος χρήστης μπορεί να διαχειριστεί τα σχήματα ειδοποίησης που είχε δημιουργήσει νωρίτερα.

Για την διαχείριση των σχημάτων ειδοποίησης του χρήστη η βασική ροή είναι η ακόλουθη. Αποστέλλεται ένα αίτημα στον GNMServlet το οποίο περιλαμβάνει και ένα τύπο, με τον οποίο ξεχωρίζει τη λειτουργία που επιθυμεί να κάνει ο χρήστης (ADD, DELETE, UPDATE, COPY). Στη συνέχεια, ο servlet αυτός επικοινωνεί με τα αντίστοιχα APIs και την βάση για να γίνουν οι αντίστοιχες αλλαγές. Τέλος, επιστρέφει ένα μήνυμα στο front-end αν έγινε επιτυχώς το αίτημα ή όχι.

3.2.3 Διάγραμμα κλάσεων

UML (Unified Modeling Language) είναι ένα πρότυπο γλώσσας, για τον καθορισμό, την οπτικοποίηση, την κατασκευή, και την τεκμηρίωση των αντικειμένων των συστημάτων λογισμικού. Είναι μια εικαστική γλώσσα η οποία χρησιμοποιείται για να κάνει σχεδιαγράμματα για το λογισμικό. Επομένως, η UML μπορεί να περιγραφεί ως μιας γενικής χρήσης οπτική γλώσσα μοντελοποίησης για την απεικόνιση, την διευκρίνιση, την κατασκευή και την δόμηση του συστήματος λογισμικού. Γενικά, χρησιμοποιείται για τη μοντελοποίηση συστημάτων λογισμικού, αλλά δεν περιορίζεται σε αυτό το όριο. Χρησιμοποιείται, επίσης, για τη μοντελοποίηση μη συστημάτων λογισμικού, όπως είναι η ροή της διαδικασίας σε μία μονάδα παραγωγής κ.λπ. Η UML δεν είναι άλλη μια γλώσσα προγραμματισμού, είναι εργαλεία τα οποία μπορούν να χρησιμοποιηθούν για τη δημιουργία κώδικα σε διάφορες γλώσσες προγραμματισμού χρησιμοποιώντας UML διαγράμματα. Έχει άμεση σχέση με αντικειμενοστραφή ανάλυση και σχεδίαση. Το UML διάγραμμα της πτυχιακής εργασίας αναλυτικότερα (βλ. Εικόνα 11):

GNMConstants: Αποτελεί μια βοηθητική κλάση, στην οποία υπάρχουν οι σταθερές του συστήματος, όπως είναι τα Google Credentials τόσο για τους χρήστες όσο και για την εφαρμογή.

Auth2Servlet: Δέχεται αιτήματα από την εφαρμογή όταν ένας χρήστης συνδεθεί. Ελέγχει τη βάση αν υπάρχουν σχήματα για τον συγκεκριμένο χρήστη και τα επιστρέφει στο front-end σε μορφή json.

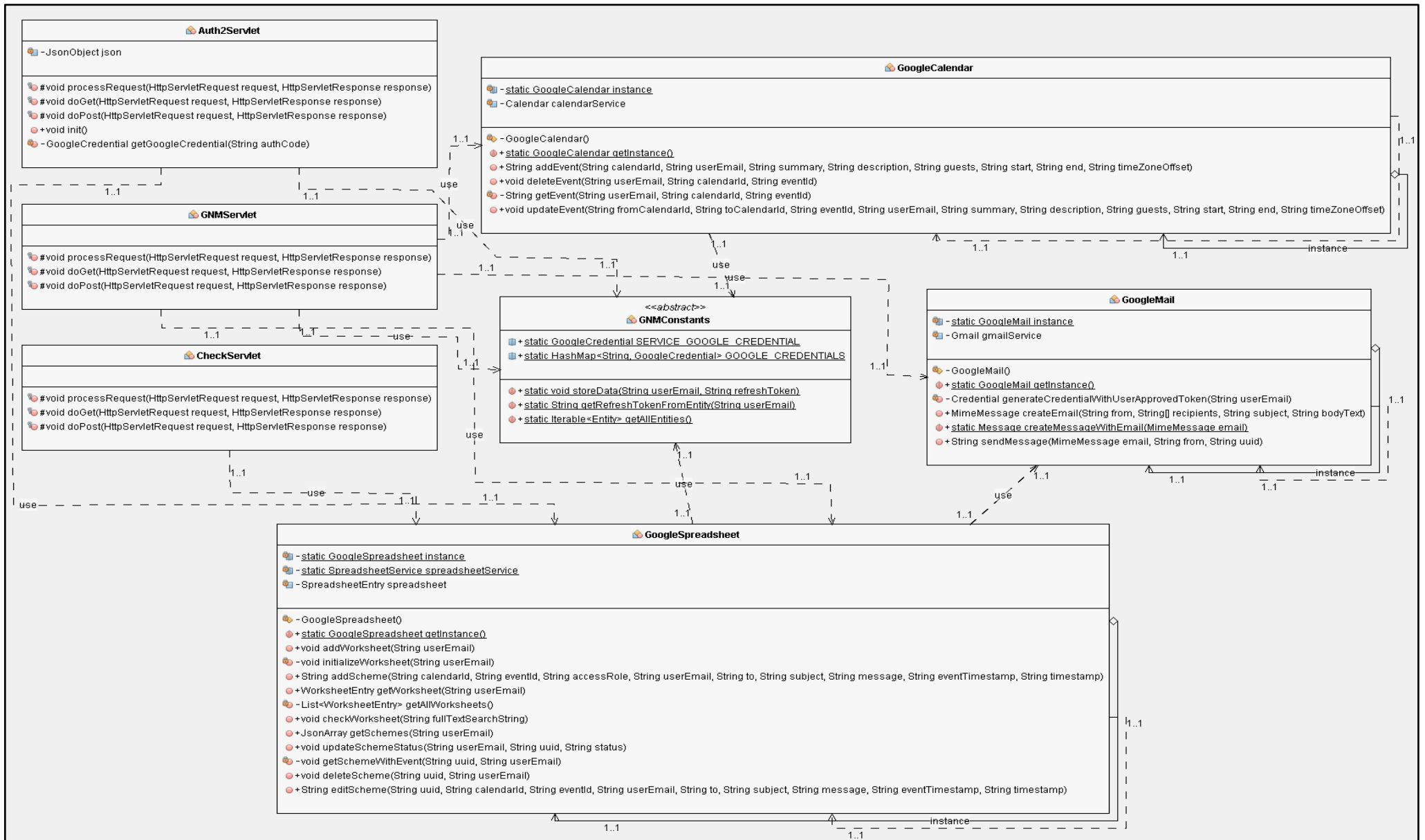
CheckServlet: Λειτουργεί σαν scheduler, ο οποίος ανά τακτά χρονικά διαστήματα ελέγχει τον χρόνο ζωής των credentials των χρηστών. Αν κοντεύουν να λήξουν τα ανανεώνει. Αυτό γίνεται για να υπάρχει η δυνατότητα αποστολής email από την πλευρά του χρήστη ακόμη και αν αυτός δεν είναι συνδεδεμένος στην εφαρμογή. Το δικαίωμα αποστολής email από την πλευρά του χρήστη το δίνει ο ίδιος ο χρήστης την πρώτη φορά που κάνει σύνδεση στην εφαρμογή.

GNMServlet: Αποτελεί τον βασικό servlet της εφαρμογής, καθώς η διαχείριση των σχημάτων ειδοποίησης γίνεται από αυτόν. Δέχεται ένα αίτημα από το front-end της εφαρμογής και αναλόγως με τον τύπο του αιτήματος, καλεί τα ανάλογα APIs για να γίνουν οι ανάλογες λειτουργίες και στην συνέχεια επικοινωνεί με την βάση για να γίνει η ανάλογη ενημέρωση. Τέλος, στέλνει στο front-end της εφαρμογής ένα μήνυμα επιτυχίας αν όλα έγιναν όπως πρέπει ή μήνυμα αποτυχίας αν κάτι πήγε στραβά.

GoogleMail: Βοηθητική κλάση για την επικοινωνία με το GMail API. Χρησιμοποιείται για να δημιουργεί emails και να τα αποστέλλει στους παραλήπτες.

GoogleCalendar: Βοηθητική κλάση για την επικοινωνία με το Google Calendar API. Χρησιμοποιείται για να δημιουργήσει νέα συμβάντα, να επεξεργαστεί ήδη υπάρχοντα και να διαγράψει συμβάντα από κάποιο Google Calendar.

GoogleSpreadsheet: Βοηθητική κλάση για την επικοινωνία με το Google Spreadsheet API. Χρησιμοποιείται για την service-to-service επικοινωνία μεταξύ της εφαρμογής και του API, και παρέχει πρόσβαση στη βάση της εφαρμογής.



Εικόνα 11: UML Diagram

3.2.4 APIs

Ένα API καθιστά δυνατή τη χρήση των προγραμμάτων μέσα από προγράμματα. Μπορεί, επίσης, να θεωρηθεί ως ενός πρωτοκόλλου, επειδή χρησιμοποιείται για τους ίδιους σκοπούς και καθορίζει τα ίδια πράγματα (οι πιθανές αλληλεπιδράσεις μεταξύ συνιστωσών, και τα δεδομένα που ανταλλάσσονται κατά την αλληλεπίδραση).

APIs, λοιπόν, είναι τα σύνολα των απαιτήσεων που καθορίζουν τον τρόπο με τον οποίο μία εφαρμογή μπορεί να επικοινωνήσει και να αλληλεπιδράσει με μια άλλη. Στην υλοποίηση του συστήματος πολλά Public APIs της Google έχουν χρησιμοποιηθεί.

Public API (που συχνά αναφέρεται και ως Open API) είναι διαθέσιμη διεπαφή προγραμματισμού εφαρμογής στο κοινό, η οποία παρέχει πρόσβαση στους προγραμματιστές μέσω κώδικα σε μια εφαρμογή λογισμικού. Επίσης, επιτρέπει στους προγραμματιστές να έχουν πρόσβαση σε ορισμένες εσωτερικές λειτουργίες ενός προγράμματος.

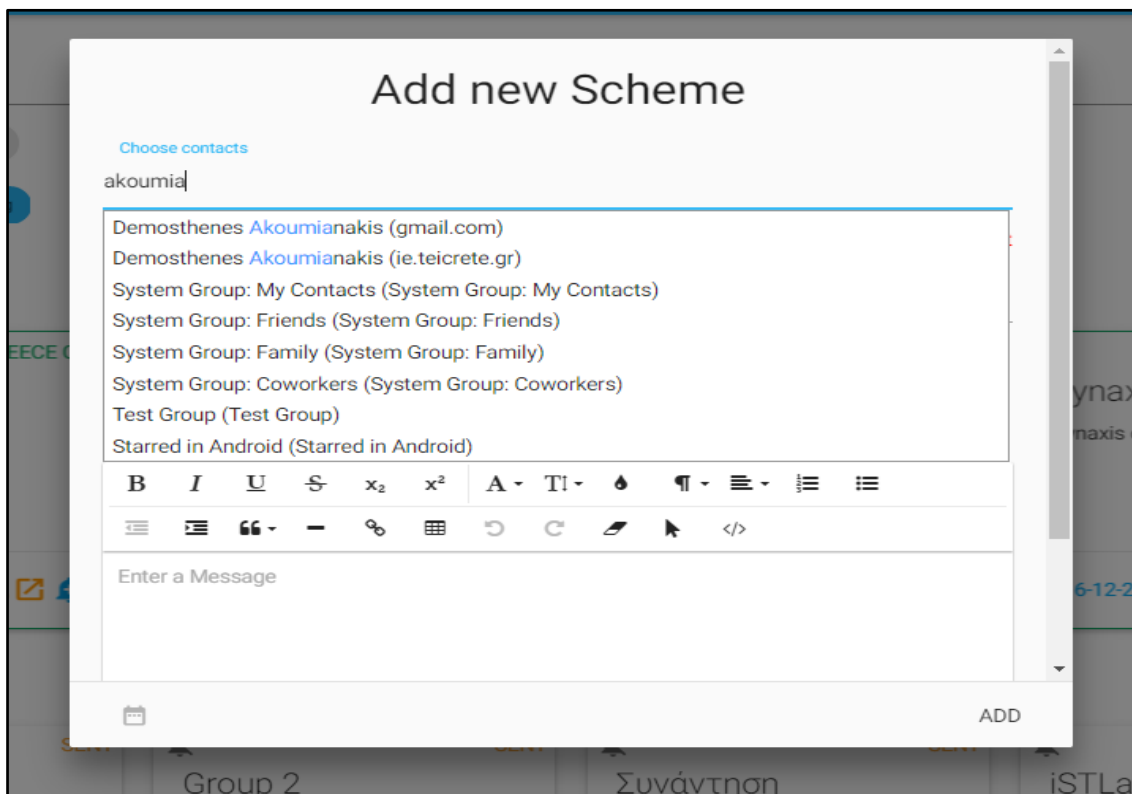
Με απλά λόγια, ένα API επιτρέπει σε ένα κομμάτι του λογισμικού να αλληλεπιδρά με ένα άλλο κομμάτι του λογισμικού.

Τα Public APIs της Google είναι υλοποιήσιμα σε front-end (JavaScript) και σε back-end (.NET, Go, Java, Node.js, κ.τ.λ.). Στην εφαρμογή αυτή έχω χρησιμοποιήσει τα πιο κάτω APIs συνδυάζοντας και front-end (JavaScript) αλλά και back-end (Java) υλοποιήσεις για κάποια APIs:

- **Google Contacts API (front-end) [13]**

Το Google Contacts API επιτρέπει στις εφαρμογές να αντλούν, να εμφανίζουν και να διαχειρίζονται τις επαφές, ή και τις ομάδες επαφών του χρήστη. Οι επαφές αυτές είναι αποθηκευμένες στο Google λογαριασμό του χρήστη και οι περισσότερες υπηρεσίες της Google έχουν πρόσβαση στη λίστα αυτή.

Με τη χρήση του Google Contacts API υπάρχει η δυνατότητα δημιουργίας νέων επαφών, επεξεργασίας ή διαγραφής υπαρχουσών επαφών, καθώς και να αναζητήσης επαφών που ταιριάζουν με συγκεκριμένα κριτήρια.



Εικόνα 12: Google Contacts API

Η εφαρμογή χρησιμοποιεί το Google Contacts API για να αντλήσει τις επαφές και τις ομάδες επαφών του χρήστη από το Google λογαριασμό του. Κάθε φορά που ο χρήστης πληκτρολογεί στο πεδίο contacts της φόρμας καλείται το συγκεκριμένο API δίνοντας μια λίστα με τις επαφές βάση της πληκτρολόγησης του (βλ. Εικόνα 12). Για την κλήση του API χρησιμοποιήθηκε το plugin jQuery Autocomplete σε συνδυασμό με AJAX request (βλ. Εικόνα 13).

```
// JS front-end (init.js)

function initComponents() {
    $("#contacts").autocomplete({
        lookup: function (query, done) {
            var result = { suggestions: [] };
            var d1 = $.get("https://www.google.com/m8/feeds/groups/default/full "), function
                (groupResponse) {
                    ...
                };
            var d2 = $.get("https://www.google.com/m8/feeds/contacts/default/full?alt=json ")
                + "&q=" + query + "&max-results=100&v=3.0", function
                (contactResponse) {
                    ...
                };
            $.when(d1, d2).done(function () {
                done(result);
            });
        },
        onSelect: function (suggestion) {
            ...
        }
    });
}
```

Εικόνα 13: Code Snippet - init.js (initComponents: Contacts Autocomplete)

- **Google Calendar API (front-end & back-end) [14]**

Το Google Calendar API επιτρέπει στις εφαρμογές να βρίσκουν και να “βλέπουν” τα γεγονότα σε δημόσια ημερολόγια της Google. Εάν υπάρχει έγκριση, τότε μπορεί επίσης να αποκτήσει πρόσβαση και να τροποποιήσει ιδιωτικά ημερολόγια και γεγονότα σε εκείνα τα ημερολόγια.

Στο front-end αντλεί τα μελλοντικά γεγονότα και τα εμφανίζει σαν προτάσεις σχημάτων ειδοποίησης, και στο back-end διαχειρίζεται γεγονότα στους Google Calendar του χρήστη.

- **GMail API (back-end) [15]**

Το GMail API επιτρέπει στις εφαρμογές να έχουν πρόσβαση στο ηλεκτρονικό ταχυδρομείο του χρήστη καθώς και να στέλνουν email μέσω αυτού.

Η χρήση του API είναι για να δημιουργούνται και να αποστέλλονται emails από μέρους του χρήστη μέσω της εφαρμογής.

- **Google Drive API (back-end) [16]**

Το Google Drive API επιτρέπει στις εφαρμογές να έχουν πρόσβαση στο Google Drive είτε του χρήστη, είτε ένα κοινό Google Drive της εφαρμογής.

Το Google Drive API παρέχει δυνατότητες όπως:

- Διάβαση αρχείων από το Google Drive
- Ανέβαση αρχείων στο Google Drive
- Κατέβαση αρχείων από το Google Drive
- Συγχρονισμός αρχείων

Το API αυτό χρησιμοποιείται για αποθήκευση της “βάσης δεδομένων” της εφαρμογής. Η εφαρμογή χρησιμοποιεί ένα Google Spreadsheet στη θέση μιας κοινής σχεσιακής βάσης δεδομένων.

- **Google Spreadsheet API (back-end) [17]**

Το Google Spreadsheet API (ή Google Sheets API) επιτρέπει στις εφαρμογές να δημιουργούν και να διαμορφώνουν λογιστικά φύλλα.

Η εφαρμογή χρησιμοποιεί το API αυτό ως βάση δεδομένων για τα σχήματα ειδοποίησης του χρήστη. Το Spreadsheet είναι αποθηκευμένο στο Google Drive της εφαρμογής και σε συνδυασμό με το Google Drive API παρέχεται πρόσβαση στο Spreadsheet για ανάκτηση πληροφοριών κάθε χρήστη.

UUID	CALENDAR_ID	EVENT_ID	EVENT_TIMES	ACCESS_ROLE	RECIPIENTS	SUBJECT	MESSAGE	TIMESTAMP	STATUS
6c22cb59-5fa1-4	2kkopsdr98r946c	6fgrv89p9ec3dfv	2016-06-30 18:00	1	ant.providakis@	UsiXML	Να δούμε την ημερ	2016-04-01 13:20	SENT
d887c335-72f7-4	2kkopsdr98r946c	6smoiu71eajjuil	2016-06-30 18:00	1	androkliis.greg@	Notification litera	To literaturte revi	2016-04-01 13:20	SENT
68fdac92-a08f-4	2kkopsdr98r946c	9t59u1rcjvobcng	2016-06-30 18:00	1	androkliis.greg@	Papers	Compile a list of	2016-04-01 13:40	SENT
1358ae4d-aab2-4	8p0n8hong8jbiee	61kchajrflhm1iq	2016-06-30 18:00	1	androkliis.greg@	Meeting - Friday	Please be prese	2016-05-20 12:00	SENT
79ffa584-a668-4	99maciu7cpof4v	fo81hgknofu1b2	2016-06-30 18:00	1	androkliis.greg@	Meeting with Ptil	We are starting ε	2016-05-20 12:20	SENT
aa5bc27e-11aa-4	99maciu7cpof4v	9lsabq87ceh2khi	2016-06-30 18:00	1	ant.providakis@	Meeting	Meeting on frida	2016-05-20 12:50	SENT
a6c3e109-a5b3-4	99maciu7cpof4v	mmtko9a0lj6guk	2016-06-30 18:00	1	ant.providakis@	Test	This is a test me	2016-05-20 13:20	SENT
1e260054-1d25-4	99maciu7cpof4v	2e4u069lhcabjk6	2016-06-30 18:00	1	mitsakos.m@gm	Meeting on Frida	Meeting istLab	2016-05-20 13:50	SENT
d05daa67-9460-4	8p0n8hong8jbiee	rcsm801pbdu8ng	2016-06-30 18:00	1	dakoumianakis@	Assignment	You are kindly in	2016-05-22 16:10	SENT
7d213303-1c2a-4	5qf64hr4g5rbhpp	vtma1tkatks7pvj	2016-06-30 18:00	1	dakoumianakis@	Meeting with alu	You are kindly in	2016-05-22 16:30	SENT
5d9458de-1ab4-4	0	0		0	ant.providakis@	LAP lecture sch	Please visit http	2016-05-23 18:10	SENT
8232f4b8-6010-4	8p0n8hong8jbiee	vvctt1tac110ng2r	2016-05-26 14:20	1	dakoumianakis@	ffff	Hello world	2016-05-26 14:20	SENT
d14b57dc-4f25-4	8p0n8hong8jbiee	0	2016-06-03 14:00	1	androkliis.greg@	Test	Test description	2016-06-03 13:30	SENT
3f4c986d-0901-4	8p0n8hong8jbiee	0hknnised75si44	2016-06-11 02:00	1	androkliis.greg@	Intention to visit	Let me know if y	2016-06-11 02:30	SENT

Add 1000 more rows at bottom.

+ androkliis.greg@gmail.com mrgsgito@gmail.com dakoumianakis@gmail.com feresgeo@gmail.com giorgos.feretzakis@gmail.com mitsak

Εικόνα 14: DataBase (Google Spreadsheet)

Κάθε worksheet είναι και διαφορετικός χρήστης, και μέσα σε αυτό υπάρχουν τα σχήματα ειδοποίησης του χρήστη με τις απαραίτητες πληροφορίες (βλ. Εικόνα 14).

3.2.5 Materialize Framework

Μετά από την εμφάνιση του Material Design στο web, άρχισαν να αναπτύσσονται διάφορα CSS frameworks βασισμένα σε αυτό.

Ένα CSS framework είναι ένας σκελετός που χρησιμοποιούμε και μας παρέχει συγκεκριμένους κανόνες για να γράφουμε CSS κώδικα. Κοινώς μπορούμε να χρησιμοποιήσουμε ένα τέτοιο framework σε μία ιστοσελίδα και να γράψουμε CSS ώστε να της δώσουμε στυλ και να της ρυθμίσουμε την εμφάνιση.

Η μεγάλη διαφορά του απλού CSS κώδικα και του κώδικα που γράφουμε με τη χρήση ενός Framework είναι ότι εάν χρησιμοποιήσουμε κάποιο framework υπάρχουν εντολές CSS οι οποίες εάν τις γράφαμε σε ένα απλό αρχείο CSS δεν θα έκαναν καμία διαφορά. Τα CSS Frameworks λοιπόν είναι σχεδιασμένα έτσι ώστε με λιγότερο κώδικα να μπορούμε να έχουμε το ίδιο και καλύτερο αποτέλεσμα από ότι θα είχαμε με απλό CSS κώδικα.

Επίσης ένα CSS Framework έχει ένα τεράστιο πλεονέκτημα. Έχει αυτό που λέμε “cross browser compatibility”. Δηλαδή αυτό που φτιάχνουμε εμφανίζεται ίδιο και στο IE και στον mozilla και στον Chrome κλπ. Αυτό είναι πολύ σημαντικό διότι οι browsers μεταξύ τους έχουν κάποιες διαφορές στον τρόπο που διαβάζουν και εμφανίζουν τα CSS και έτσι πρέπει πάντα να βλέπουμε αν υπάρχουν bugs και να τα διορθώνουμε, ενώ συνήθως με ένα CSS Framework δεν χρειάζεται να γίνεται αυτό. [18]

Το CSS framework το οποίο χρησιμοποίησα στην εφαρμογή μου ονομάζεται Materialize και είναι βασισμένο στις αρχές του Material Design.

Η επιλογή του framework αυτού, μέσα από πολλά άλλα, έγινε βάση των πιο κάτω κριτηρίων:

- Οι κανόνες του CSS βασίζονται στους κανόνες του Bootstrap (δεν είναι Bootstrap), με αποτέλεσμα να είναι πιο γνώριμο προς εμένα και έτσι θα γινόταν η υλοποίηση πιο γρήγορα.
- Υπάρχει άμεση υποστήριξη από τους κατασκευαστές του framework καθώς επίσης και συνεχείς αναβαθμίσεις.

Λόγω του ότι τόσο το Material Design Lite, όσο και το Materialize δεν παρέχουν όλα τα εργαλεία τα οποία ένας προγραμματιστής χρειάζεται, έχουν υλοποιηθεί από τρίτους κάποιες επεκτάσεις του Materialize. Για τον σωστό σχεδιασμό του Group Notification Mechanism έχει συμπεριληφθεί η επέκταση για την επιλογή ώρας υλοποιημένη σε Materialize και λέγεται Materialize ClockPicker. [19] Επίσης και για την επιλογή ημερομηνίας το DatePicker. [20]

3.2.6 Integrated Development Environment (IDE)

Ένα ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) είναι μια σουίτα λογισμικού που συγκεντρώνει τα βασικά εργαλεία τα οποία οι προγραμματιστές χρειάζονται για να γράψουν και να δοκιμάσουν το λογισμικό τους. Τυπικά, ένα IDE περιέχει έναν επεξεργαστή κειμένου, ένα μεταγλωττιστή ή διερμηνέα και ένα πρόγραμμα εντοπισμού σφαλμάτων, στα οποία ο προγραμματιστής έχει πρόσβαση μέσω μιας ενιαίας γραφικής διεπαφής χρήστη (GUI). Ένα IDE μπορεί να είναι μια αυτόνομη εφαρμογή, ή μπορεί να συμπεριληφθεί ως μέρος ενός ή περισσότερων υπαρχόντων και συμβατών εφαρμογών.

Το NetBeans είναι μια πλατφόρμα ανάπτυξης λογισμικού γραμμένη σε Java. Η πλατφόρμα NetBeans επιτρέπει στις εφαρμογές να αναπτυχθούν από ένα σύνολο πολυμορφικών στοιχείων λογισμικού που ονομάζονται ενότητες. Οι εφαρμογές που βασίζονται στην πλατφόρμα NetBeans, συμπεριλαμβανομένου και του ολοκληρωμένου περιβάλλοντος ανάπτυξης (IDE) NetBeans, μπορούν να επεκταθούν από τρίτους προγραμματιστές. Το NetBeans είναι cross-platform IDE και τρέχει σε Microsoft Windows, Mac OS X, Linux, Solaris και άλλες πλατφόρμες που υποστηρίζουν ένα συμβατό JVM. [21]

Η ομάδα του NetBeans υποστηρίζει ενεργά το προϊόν και είναι ανοιχτή σε οποιοδήποτε feedback από την ευρύτερη κοινότητα.

Το NetBeans IDE απευθύνεται κυρίως για την ανάπτυξη εφαρμογών σε Java, αλλά υποστηρίζει επίσης και άλλες γλώσσες, ιδίως PHP, C / C ++ και HTML5. [22]

3.2.7 Development Frameworks

Development Frameworks είναι μια αφηρημένη έννοια στην οποία ένα λογισμικό το οποίο παρέχει μια γενική λειτουργικότητα μπορεί να αλλάξει επιλεκτικά από τον πρόσθετο κώδικα τον οποίο γράφει ο προγραμματιστής με τη βοήθεια του Development Framework αυτού, παρέχοντας έτσι συγκεκριμένες εφαρμογές λογισμικού.

Ο προγραμματισμός τόσο του front-end όσο και του back-end της εφαρμογής έγινε με την χρήση του ολοκληρωμένου περιβάλλοντος ανάπτυξης εφαρμογών NetBeans v. 8.1 σε συνδυασμό με το Google App Engine Plugin. [23]

Με την χρήση του jQuery (JavaScript Framework) [24] και με την βοήθεια πολλών plugins βασισμένων στο jQuery, τα οποία θα δούμε αναλυτικότερα πιο κάτω, υλοποιήθηκαν όλες οι λειτουργίες του front-end της εφαρμογής.

Το jQuery είναι μια μικρή, γρήγορη και πλούσια σε χαρακτηριστικά βιβλιοθήκη JavaScript. Κάνει τα πράγματα απλούστερα όπως διάσχιση και χειραγώγηση HTML εγγράφων, το χειρισμό γεγονότων, κινουμένων στοιχείων, καθώς και τη χρήση του Ajax. Έχει πολύ απλό και εύκολο στη χρήση API που λειτουργεί σε ένα πλήθος browsers.

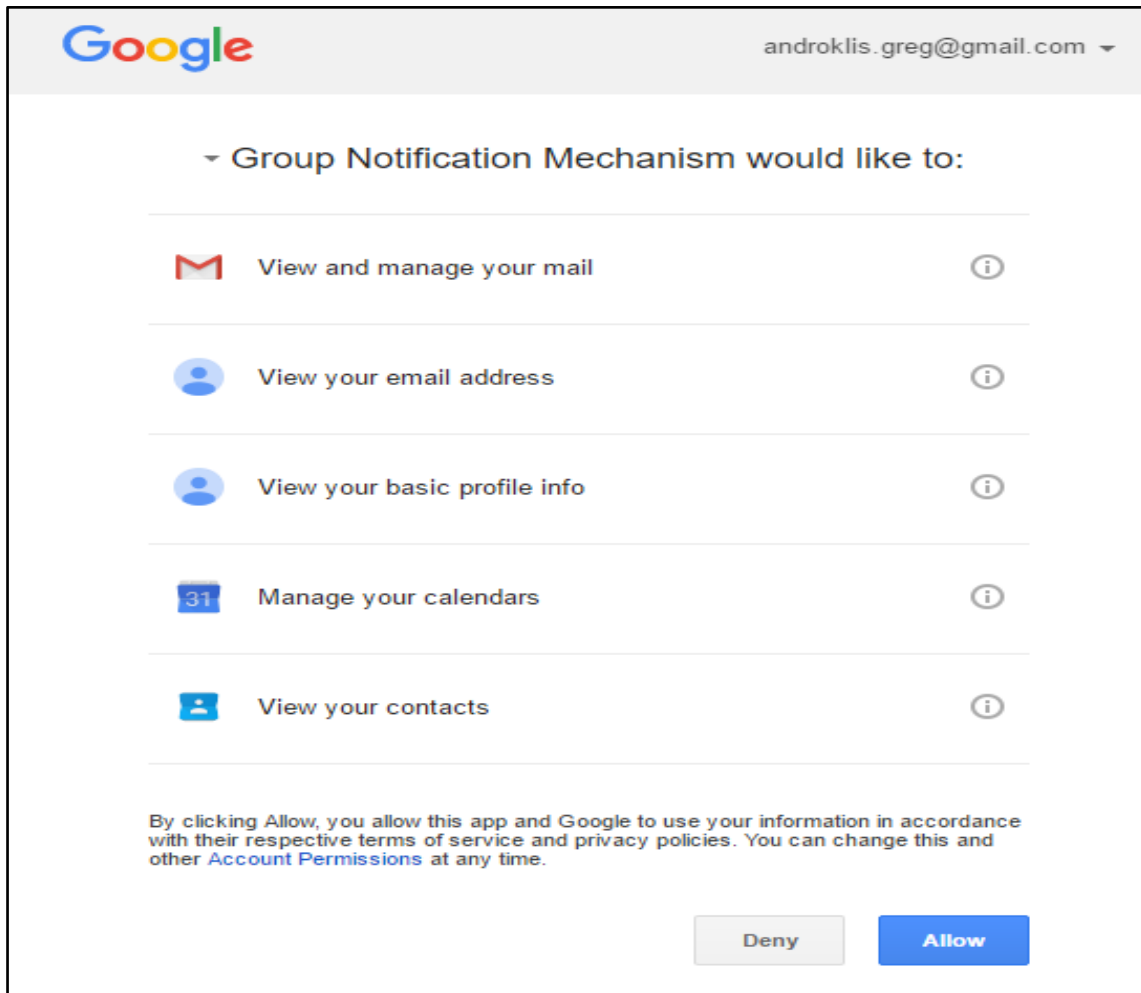
3.2.8 Plugins

Βασιζόμενα στη βιβλιοθήκη jQuery έχουν αναπτυχθεί πολλά άλλα plugins τα οποία έρχονται να επεκτείνουν και να προσθέσουν καινούριες λειτουργίες. Κάποιες από αυτά τα plugins έχουν χρησιμοποιηθεί για την υλοποίηση του front-end του Group Notification Mechanism. Αναλυτικότερα:

- **MixItUp** [25]: Το MixItUp είναι ένα jQuery plugin που παρέχει μηχανισμούς φιλτραρίσματος και ταξινόμησης με animation. Είναι ιδανικό για τη διαχείριση κάθε κατηγοριοποιημένου ή να διατεταγμένου περιεχόμενου, όπως portfolios, galleries και blogs, αλλά μπορεί επίσης να λειτουργήσει ως ένα ισχυρό εργαλείο το οποίο εμπλέκει UI εφαρμογές με την οπτικοποίηση δεδομένων.
- **jQuery Autocomplete** [26]: Επιτρέπει στους χρήστες να βρίσκουν και να επιλέγουν γρήγορα από μια προ συμπληρωμένη λίστα τιμών, καθώς πληκτρολογούν, αξιοποιώντας την αναζήτηση και το φιλτράρισμα.
- **jQuery Cookie** [27]: Ένα απλό και ελαφρύ jQuery plugin για την ανάγνωση, τη γραφή και τη διαγραφή των cookies.
- **Lunr** [28]: Είναι μια απλή μηχανή αναζήτησης πλήρους κειμένου που δουλεύει στην πλευρά του client. Έχει σχεδιαστεί για να είναι μικρό, αλλά πλήρως εξοπλισμένο, δίνοντάς τη δυνατότητα μιας εξαιρετικής εμπειρίας στην αναζήτηση χωρίς την ανάγκη εξωτερικών υπηρεσιών αναζήτησης.
- **OWL Carousel** [29]: Ένα jQuery plugin που επιτρέπει τη δημιουργία όμορφων και responsive carousel slider. Υποστηρίζεται και από οθόνες αφής.
- **Froala Editor** [30]: Ένα από τα καλύτερα προγράμματα επεξεργασίας HTML κειμένου WYSIWYG (What You See Is What You Get). Ένα απλό στη χρήση πλούσιο πρόγραμμα επεξεργασίας κειμένου, με υποστήριξη σε κινητές συσκευές. Το plugin είναι δωρεάν για μη εμπορικά έργα.

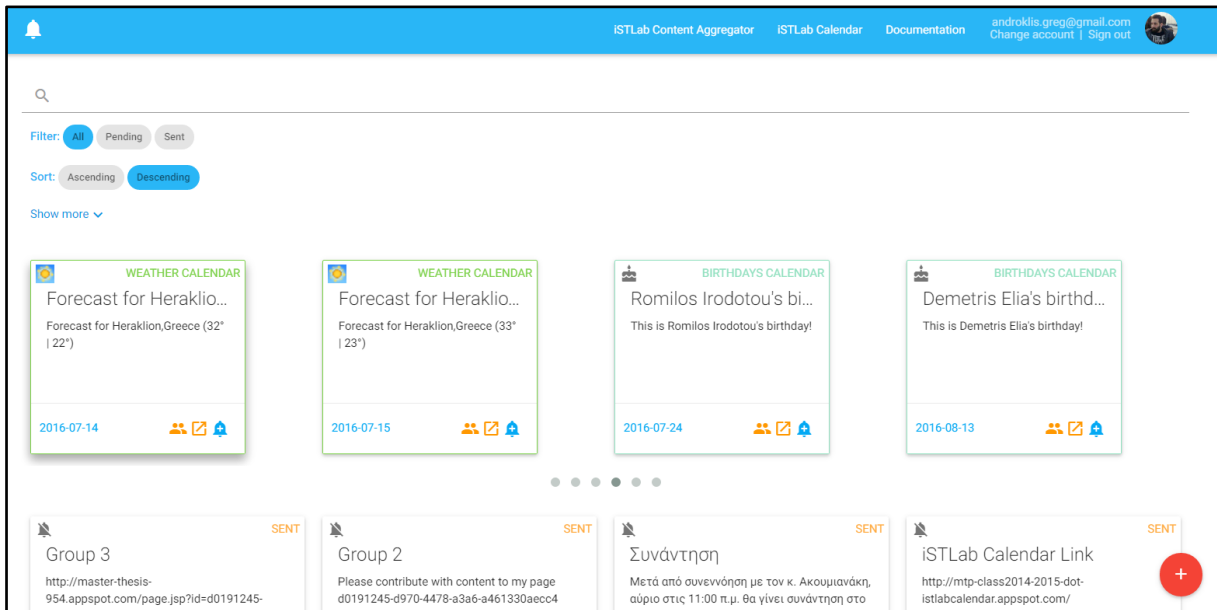
4 Περίπτωση χρήσης και πιλοτική επίδειξη

Ο χρήστης συνδέεται στο μηχανισμό χρησιμοποιώντας τον Google λογαριασμό του. Αν είναι η πρώτη φορά που συνδέεται στο μηχανισμό τότε πρέπει να δώσει πρόσβαση σε συγκεκριμένες υπηρεσίες της Google που χρησιμοποιεί η εφαρμογή για να συνδεθεί. (βλ. Εικόνα 15)



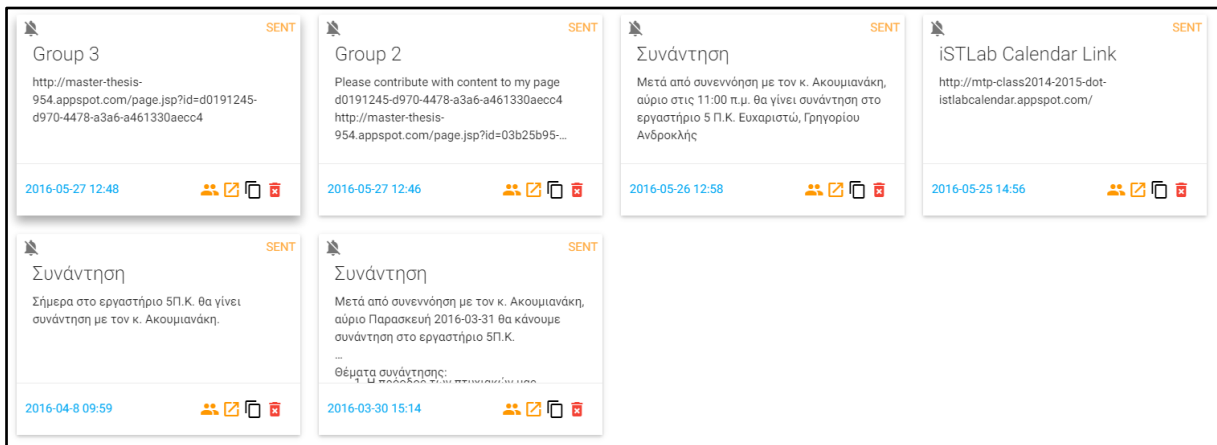
Εικόνα 15: Group Notification Mechanism - Authentication

Μετά από μια επιτυχημένη σύνδεση στον μηχανισμό, ο χρήστης βλέπει την κύρια διεπαφή (βλ. Εικόνα 16) στην οποία εμφανίζονται διάφορα προτεινόμενα συμβάντα από τον Google Calendar του για να δημιουργήσει μια ειδοποίηση βάση αυτών.



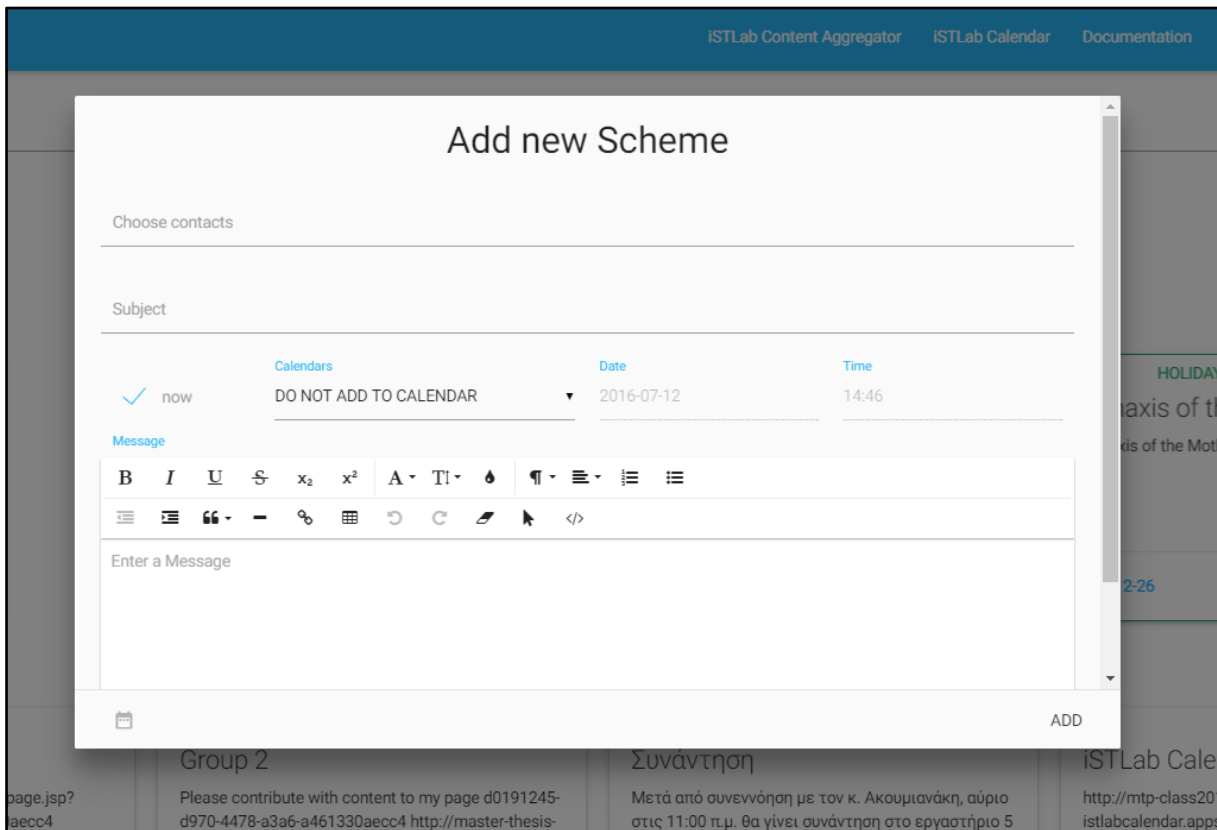
Εικόνα 16: Group Notification Mechanism - Main Screen

Κάτω από τα προτεινόμενα συμβάντα εμφανίζονται τα ήδη υπάρχοντα σχήματα ειδοποίησης, αν υπάρχουν. (βλ. Εικόνα 17)

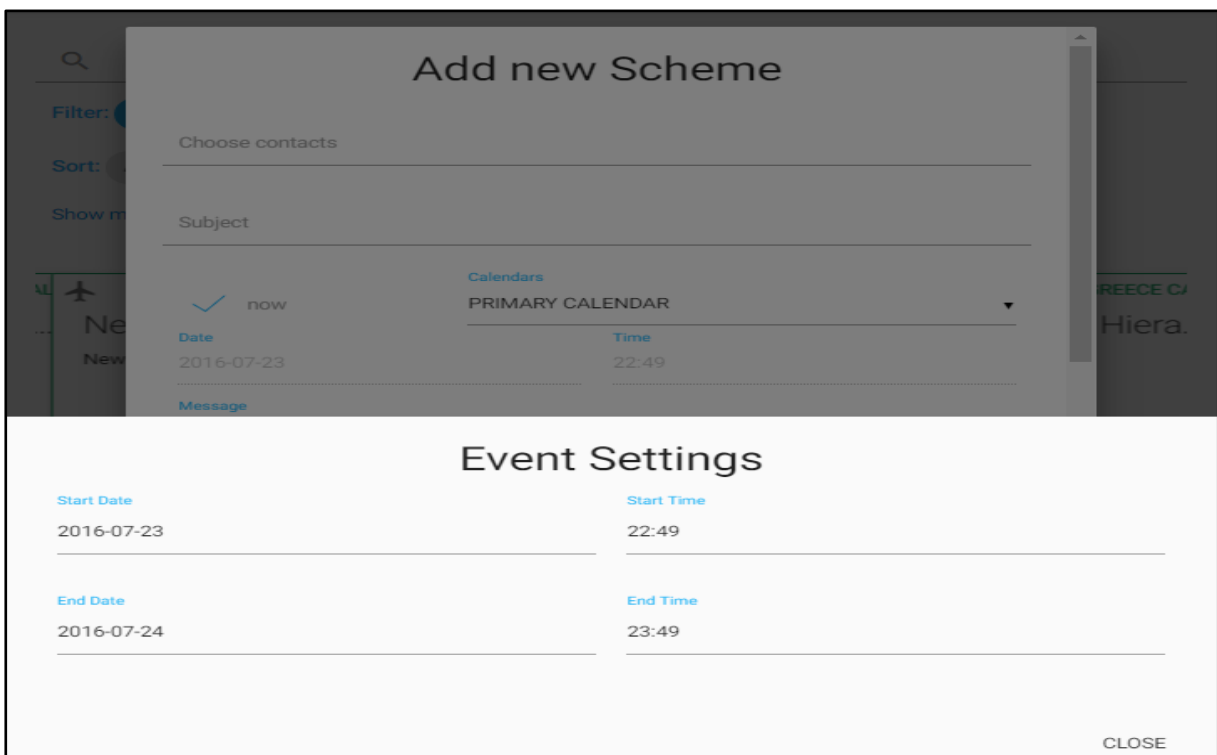


Εικόνα 17: Group Notification Mechanism - Schemes

Ο χρήστης δημιουργεί μια καινούργια ειδοποίηση έχοντας τη δυνατότητα να δημιουργηθεί, αν επιθυμεί, ένα event σε κάποιον Google Calendar της επιλογής του. Αυτόματα οι παραλήπτες γίνονται και guests στο συγκεκριμένο event. (βλ. Εικόνα 18)



Εικόνα 18: Group Notification Mechanism - Add new Scheme



Εικόνα 19: Event Settings

Επιλέγοντας ο χρήστης την δημιουργία ενός event σε κάποιον Google Calendar γίνονται τα εξής:

- Ο δημιουργός του σχήματος ειδοποίησης είναι και ο δημιουργός του event.
- Οι παραλήπτες είναι οι guests στο event.
- Το θέμα του σχήματος ειδοποίησης είναι ο τίτλος του event.
- Η ημερομηνία και η ώρα έναρξης και λήξης του event καθορίζεται αφού επιλεγεί πρώτα σε ποιόν Google Calendar θα μπει το event (βλ. Εικόνα 19).
- Η περιγραφή του event είναι το μήνυμα του σχήματος ειδοποίησης.

Πέρα από αυτές τις πληροφορίες που βλέπει ο χρήστης, υπάρχουν και πληροφορίες που αποστέλλονται στο back-end τις οποίες δεν βλέπει. Οι πληροφορίες αυτές συμπεριλαμβάνουν το id του Calendar στο οποίο θα μπει το event, καθώς και ένα κωδικό ο οποίος δείχνει τον τύπο του request που θα γίνει (π.χ. αν το request είναι η προσθήκη ενός νέου σχήματος ειδοποίησης ο κωδικός αυτός είναι "ADD"). Έχοντας αυτές τις πληροφορίες γίνονται οι ακόλουθες κλήσεις στον κώδικα:

Μετά την συμπλήρωση της φόρμας για την δημιουργία ενός νέου σχήματος ειδοποίησης καλείται η μέθοδος addCard (βλ. Εικόνα 20), η οποία βρίσκεται στο αρχείο modal-actions.js, και στην οποία γίνεται έλεγχος πληρότητας και η αποστολή της πληροφορίας στο back-end.

```
// JS front-end (modal-actions.js)

function addCard() {
  ...
  var json = {
    ...
    type: 'ADD'
  };

  $.post("GNMServlet", {json: json}, function (response, statusText, xhr) {
    ...
  }, "json");
}
}
```

Εικόνα 20: Code Snippet - modal-actions.js (addCard)

Στην συνέχεια, αφού γίνει η κλήση του GNMServlet και αποσταλεί η πληροφορία, γίνεται ξεκαθάρισμα του τύπου του request (βλ. Εικόνα 21) και ακολούθως οι κλήσεις μεθόδων που χρειάζεται ένα σχήμα για να δημιουργηθεί. Αν στις πληροφορίες υπάρχει id κάποιου Calendar τότε καλείται η μέθοδος addEvent, από την βοηθητική κλάση GoogleCalendar που βρίσκεται στο αρχείο GoogleCalendar.java και δημιουργείται ένα νέο event με τις πληροφορίες του σχήματος ειδοποίησης (βλ. Εικόνα 22).

```

// Java back-end (GNMServlet.java)

protected void processRequest(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    ...
    switch (request.getParameter("json[type]")) {
    case "ADD":
        ...
        if ((!"0".equals(eventId)) && (!"0".equals(calendarId))) {
            ...
        } else if ((!"0".equals(eventId)) && (!"0".equals(calendarId))) {
            eventId = GoogleCalendar.getInstance().addEvent(...);
        }
    }
    ...
    if ("true".equals(request.getParameter("json[now]"))) {
        ...
    } else {
        ...
    }

    break;
case "DELETE":
    ...
    break;
case "UPDATE":
    ...
    break;
case "COPY":
    ...
    break;
default:
    break;
    }
}
}

```

Εικόνα 21: Code Snippet - GNMServlet.java (processRequest)

```

// Java back-end (GoogleCalendar.java)

public String addEvent(String calendarId, String userEmail, String summary, String description,
String guests, String start, String end, String timeZoneOffset) throws IOException {

    calendarService      =      new      Calendar.Builder(new      NetHttpTransport(),
JacksonFactory.getDefaultInstance(),
GOOGLE_CREDENTIALS.get(userEmail)).setApplicationName("Group      Notification
Mechanism").build();

    Event event = new Event().setSummary(summary).setDescription(description);

    event.setStart(new EventDateTime().setDateTime(new DateTime(start.split(" ")[0] + "T"
+ start.split(" ")[1] + ":00" + timeZoneOffset + ":00")));
    event.setEnd(new EventDateTime().setDateTime(new DateTime(end.split(" ")[0] + "T" +
end.split(" ")[1] + ":00" + timeZoneOffset + ":00")));

    ArrayList<EventAttendee> attendees = new ArrayList<>();

    for (String guest : guests.split(",")) {
        attendees.add(new EventAttendee().setEmail(guest));
    }

    event.setAttendees(attendees);

    event = calendarService.events().insert(calendarId, event).execute();
    return event.getId();
}

```

Εικόνα 22: Code Snippet - GoogleCalendar.java (addEvent)

5 Κατακλείδα

Στο κεφάλαιο αυτό επιχειρείται μια συνολική αποτίμηση της πτυχιακής εργασίας και ενδεικτικές βελτιώσεις που θα μπορούσαν να υλοποιηθούν μελλοντικά για την αρτιότερη λειτουργία του μηχανισμού. Οι βελτιώσεις αυτές αφορούν κατά κύριο λόγο την επεκτασιμότητα του μηχανισμού ως προς τις λειτουργίες που παρέχει προς τον χρήστη καθώς και την ταχύτητα λειτουργίας του scheduler, ο οποίος ελέγχει κάθε λεπτό την βάση δεδομένων για να στείλει τις ειδοποιήσεις.

5.1 Αποτίμηση

Η πτυχιακή εργασία επέτρεψε στον συγγραφέα να εξοικειωθεί με τεχνολογίες και εργαλεία της τρέχουσας τεχνολογικής τροχιάς. Αποκτήθηκαν νέες δεξιότητες, τόσο σχεδιαστικές όσο και προγραμματιστικές, σε επίκαιρα γνωστικά αντικείμενα που υπερβαίνουν τα συμβατικά όρια ενός προπτυχιακού προγράμματος σπουδών. Επιπλέον, δόθηκε η ευκαιρία εμβάθυνσης σε βασικές έννοιες της συνεργατικής τεχνολογίας που συνιστά ένα αναδυόμενο γνωστικό πεδίο με σύνθετες απαιτήσεις.

Πέρα των τεχνικών ζητημάτων που χρειάστηκε να αντιμετωπιστούν, η ενασχόληση με το συγκεκριμένο θέμα σε ένα περιβάλλον εργαστηρίου, επέτρεψε στο συγγραφέα να οργανώσει το χρόνο του, να συντονιστεί με προσπάθειες συναδέλφων του και να δρομολογήσει την εργασία του με τρόπο τέτοιο που να συνεισφέρει στο συλλογικό αποτέλεσμα του εργαστηρίου.

Τέλος, προκαλεί ιδιαίτερη ικανοποίηση το γεγονός ότι κατά τη φάση υλοποίησης της πτυχιακής εργασίας δόθηκε έμφαση στην τεκμηρίωση σχεδιαστικών επιλογών, την επιλογή τεχνολογικών επίκαιρων τεχνολογικών εργαλείων και στην ανάπτυξη μιας υπηρεσίας η οποία λειτουργεί και διεκπεραιώνει τα προβλεπόμενα.

5.2 Μελλοντική Εργασία και Επεκτάσεις

Κάθε εργασία που εκτελείται σε στενά χρονικά περιθώρια, καταλήγει σε σειρά συστάσεων που θα μπορούσαν να υλοποιηθούν για την βελτίωση επιμέρους λειτουργικών συστατικών. Έτσι και στην περίπτωση της παρούσας εργασίας είναι σημαντικό να αναφερθούμε εν συντομία σε ορισμένες περιοχές που επιδέχονται τροποποιήσεις και βελτιώσεις. Οι επεκτάσεις αυτές αφορούν κυρίως την βελτιστοποίηση της επιλογής παραληπτών χρησιμοποιώντας τη γεωγραφική τους θέση εν συγκρίσει με την γεωγραφική θέση του δημιουργού. Αυτή η επέκταση είναι ιδανική για την αποστολή ειδοποίησης στους χρήστες οι οποίοι είναι μέσα στην ακτίνα ενός event το οποίο θα διεξαχθεί είτε στην περιοχή που είναι είτε σε κοντινή περιοχή. Μια άλλη προσέγγιση θα ήταν η επιλογή των παραληπτών βάση κάποιων topics ή tags που θα συνεισέφερε στην στοχευμένη ενημέρωση χρηστών που συνδέονται με κάποιας μορφή δεσμό (π.χ. μέλη της ίδιας ομάδας, κοινά ενδιαφέροντα, κ.λπ.) που θα μπορούσαν να αποτυπώνονται υπό μορφή topic ή tag. Τέλος, μια άλλη επέκταση, η οποία δεν έχει να κάνει με την επιλογή παραληπτών, θα ήταν η αποστολή SMS στο κινητό των χρηστών. Αυτή η επέκταση προϋποθέτει την πληρωμή υπηρεσίας για την αποστολή SMS μέσω της εφαρμογής αυτής.

Βιβλιογραφία

- [1] «Web,» [Ηλεκτρονικό]. Available: <https://flatworldbusiness.wordpress.com/flat-education/previiously/web-1-0-vs-web-2-0-vs-web-3-0-a-bird-eye-on-the-definition/>.
- [2] «World Wide Web,» [Ηλεκτρονικό]. Available: https://el.wikipedia.org/wiki/%CE%A0%CE%B1%CE%B3%CE%BA%CF%8C%CF%83%CE%BC%CE%B9%CE%BF%CF%82_%CE%99%CF%83%CF%84%CF%8C%CF%82.
- [3] «Web Application,» [Ηλεκτρονικό]. Available: https://en.wikipedia.org/wiki/Web_application.
- [4] «Computer-supported cooperative work,» [Ηλεκτρονικό]. Available: https://en.wikipedia.org/wiki/Computer-supported_cooperative_work.
- [5] M. v. Dantzich, D. Robbins, E. Horvitz και M. Czerwinski, «AVI '02 Proceedings of the Working Conference on Advanced Visual Interfaces,» 2002.
- [6] «Types of Awareness,» [Ηλεκτρονικό]. Available: <http://visual.placodermi.org/2009/04/05/types-of-awareness/>.
- [7] C. Gutwin και S. Greenberg, «The Importance of Awareness for Team Cognition in Distributed Collaboration,» 2002.
- [8] M. Pielot, K. Church και R. d. Oliveira, «An in-situ study of mobile phone notifications,» *Proceedings of the 16th international conference on Human-computer interaction with mobile devices & services*, 2014.
- [9] A. Mehrotra, R. Hendley, M. Mirco και V. Pejovic, «Designing content-driven intelligent notification mechanisms for mobile applications,» *UbiComp '15*, 2015.
- [10] «Asynchronous Collaboration Awareness: Expectation Fulfillment Notification,» [Ηλεκτρονικό]. Available: http://www.foldr.org/~irina/master_thesis/.
- [11] I.-C. Tsai, P. Nuankhieo, C.-H. Wu και Y.-W. Chang, «Supports of Group Awareness for Synchronous and Asynchronous Collaboration».
- [12] «Java Servlet,» [Ηλεκτρονικό]. Available: https://en.wikipedia.org/wiki/Java_servlet.
- [13] G. Developers, «Google Contacts API,» [Ηλεκτρονικό]. Available: <https://developers.google.com/google-apps/contacts/v3/>.
- [14] G. Developers, «Google Calendar API,» [Ηλεκτρονικό]. Available: <https://developers.google.com/google-apps/calendar/v3/reference/>.
- [15] G. Developers, «Gmail API,» [Ηλεκτρονικό]. Available: <https://developers.google.com/gmail/api/v1/reference/>.
- [16] G. Developers, «Google Drive API,» [Ηλεκτρονικό]. Available: <https://developers.google.com/drive/v3/reference/>.
- [17] G. Developers, «Google Sheets API,» [Ηλεκτρονικό]. Available: <https://developers.google.com/sheets/reference/rest/>.
- [18] A. Wang, A. Chang, A. Mark και K. Louie, «Materialize CSS Framework,» [Ηλεκτρονικό]. Available: <http://materializecss.com/>.
- [19] «Clock Picker,» [Ηλεκτρονικό]. Available: <https://github.com/chingyawhao/materialize-clockpicker/>.
- [20] «Date Picker,» [Ηλεκτρονικό]. Available: <http://amsul.ca/pickadate.js/>.
- [21] «Integrated Development Environment (IDE),» [Ηλεκτρονικό]. Available: <http://searchsoftwarequality.techtarget.com/definition/integrated-development-environment>.
- [22] «NetBeans IDE,» [Ηλεκτρονικό]. Available: <https://en.wikipedia.org/wiki/NetBeans/>.
- [23] «How to use the Google App Engine with the NetBeans IDE,» [Ηλεκτρονικό]. Available: <https://blog.idrsolutions.com/2015/09/how-to-use-the-google-app-engine-with-the-netbeans-ide/>.
- [24] «jQuery,» [Ηλεκτρονικό]. Available: <https://jquery.com/>.

- [25] «MixItUp,» [Ηλεκτρονικό]. Available: <https://mixitup.kunkalabs.com/>.
- [26] «jQuery Autocomplete,» [Ηλεκτρονικό]. Available: <https://jqueryui.com/autocomplete/>.
- [27] «jQuery Cookie,» [Ηλεκτρονικό]. Available: <https://plugins.jquery.com/cookie/>.
- [28] «Lunr,» [Ηλεκτρονικό]. Available: <http://lunrjs.com/>.
- [29] «OWL Carousel,» [Ηλεκτρονικό]. Available: <http://owlgraphic.com/owlcarousel/>.
- [30] «Froala Editor,» [Ηλεκτρονικό]. Available: <https://www.froala.com/>.
- [31] S. T. Iqbal και E. Horvitz, «Notifications and awareness: a field study of alert usage and preferences,» 2010.
- [32] Google, «Material design,» [Ηλεκτρονικό]. Available: <https://material.google.com/>.
- [33] V. Rathi, D. Cheng, H. S. Raje, N. N. Putrevu και J. Hou, «Dynamic, customizable and configurable notification mechanism». 2002.
- [34] J. M. Carroll, D. C. Neale, P. L. Isenhour, M. B. Rosson και D. McCrickard, «Notification and awareness: synchronizing task-oriented collaborative activity,» *International Journal of Human-Computer Studies*, 2003.
- [35] K. Ganwani και K. Jha, «Google Material Design - Visual Language Technology By Google,» *Journal of Global Research Computer Science & Technology*, τόμ. III, αρ. III, 2015.
- [36] J. P. Sousa και D. Garlan, «Improving User-Awareness by Factoring it Out of Applications,» *5th Annual Conference on Ubiquitous Computing*, 2003.
- [37] D. Merrill, «Mashups: The new breed of Web app,» 2006.