



Χαροκόπειο Πανεπιστήμιο
Τμήμα Πληροφορικής & Τηλεματικής

Πτυχιακή Εργασία

Unity: Περιβάλλον Κοινωνικής Δικτύωσης για τη Συνεργασία των
Μελών της Ακαδημαϊκής Κοινότητας

Δημιουργία και Χρήση Εφαρμογών

Κατσιβέλης-Περάκης Παναγής

AM: 20714

Επιβλέπουσα

Μάρα Νικολαΐδου, Αναπληρώτρια Καθηγήτρια

Μέλη της εξεταστικής επιτροπής

Δημοσθένης Αναγνωστόπουλος, Καθηγητής

Ηρακλής Βαρλάμης, Λέκτορας

Πρόλογος.....	4
Περίληψη	6
Abstract.....	7
Keywords	8
1. Εισαγωγή	9
1.1 Βασικοί ορισμοί	9
1.2 Τυπική λειτουργικότητα Ιστότοπου Κοινωνικής Δικτύωσης	10
1.2.1 Βασικές Λειτουργίες Ιστότοπων Κοινωνικής Δικτύωσης.....	10
1.2.2 Δημιουργία και χρήση εφαρμογών.....	12
1.3 Υποστήριξη και Ανάπτυξη Εφαρμογών σε Δημοφιλείς Ιστότοπους Κοινωνικής Δικτύωσης	13
1.3.1 Friendster	13
1.3.2 MySpace	15
1.3.3 Facebook.....	17
1.3.4 YouTube	19
1.3.5 Twitter	20
1.3.6 Google+	22
1.3.7 Συγκριτικός πίνακας υποστήριξης εφαρμογών από ΙΚΔ	24
2. Τεχνολογικό Υπόβαθρο Ανάπτυξης Εφαρμογών σε Ιστότοπους Κοινωνικής Δικτύωσης.....	26
2.1 Αρχιτεκτονική και χρησιμοποιούμενη Τεχνολογία Ανάπτυξης Ιστότοπων Κοινωνικής Δικτύωσης	26
2.2 Εργαλεία για τη δημιουργία και φιλοξενία εφαρμογών.....	27
2.2.1 Δημιουργία και χρήση εφαρμογών.....	27
2.2.3 OpenSocial.....	28
2.2.3.1 Επισκόπηση	28
2.2.3.2 Άτομα.....	29
2.2.3.3 Σχέσεις	29
2.2.3.4 Δραστηριότητες	30
2.2.3.5 Διατήρηση Δεδομένων – Δεδομένα Εφαρμογών	30
2.2.3.6 Όψεις.....	31
2.2.3.7 Γεγονότα Κύκλου Ζωής.....	32

2.2.3.8 Εφαρμογές του OpenSocial	32
2.2.4 Apache Shindig.....	33
3. Λειτουργικές Απαιτήσεις ενός ακαδημαϊκού ιστότοπου κοινωνικής δικτύωσης με έμφαση στην εκτέλεση εφαρμογών	34
3.1 Ορισμός Προβλήματος.....	34
3.1.1 Περιβάλλον Λειτουργίας	34
3.1.1.1 Χρήστες.....	34
3.1.1.2 Συμμετέχοντες.....	34
3.1.1.3 Σχέσεις	35
3.1.1.4 Βασικές Λειτουργίες	36
3.1.2 Στόχος Εργασίας.....	36
3.2 Λειτουργικές Απαιτήσεις	37
3.2.1 Παρεχόμενες υπηρεσίες και αξιολόγηση της αποτελεσματικότητας τους στο εσωτερικό μίας Ακαδημαϊκής Κοινότητας.....	37
3.2.2 Συνεισφορά κοινωνικής δικτύωσης στην αποτελεσματικότητα παροχής υπηρεσιών.....	39
3.2.3 Σενάριο Χρήσης	40
3.2.3.1 Φοιτητής.....	40
3.2.3.2 Καθηγητής	41
3.2.3.3 Γραμματέας.....	41
3.2.4 Εκμαίευση Απαιτήσεων	42
3.2.4.1 Χρήστες.....	42
3.2.4.2 Σχέσεις	42
3.2.4.3 Δεδομένα.....	42
3.2.4.4 Εγκατάσταση και χρήση εφαρμογών.....	43
4. Υλοποίηση.....	44
4.1 Περιβάλλον Υλοποίησης.....	44
4.1.1 Επίπεδα	44
4.1.2 Αρχιτεκτονική Shindig	47
4.2 Δημιουργία και Διαχείριση Εφαρμογών	50
4.2.1 Εγκατάσταση Shindig.....	50

4.2.1.1 Εγκατάσταση	50
4.2.1.2 Ενσωμάτωση Βάσης Δεδομένων	51
4.2.2 Επέκταση Shindig.....	52
4.2.2.1 Επέκταση Βάσης Δεδομένων.....	52
4.2.2.2 Επέκταση Shindig	56
4.2.2.3 Υποστηριζόμενες Κλήσεις.....	58
4.2.3 Ανάπτυξη Gadgets.....	63
4.2.3.1 Δημιουργία Gadgets.....	63
4.2.3.2 Υποστηριζόμενος Μηχανισμός Ενσωμάτωσης	66
4.2.3.3 Μηχανισμός Διαχείρισης Κύκλων Ζωής	68
5. Πιλοτική Εφαρμογή.....	71
5.1 Περιγραφή Περιβάλλοντος ΙΚΔ.....	71
5.2 Ανάπτυξη Πιλοτικών Εφαρμογών	75
5.2.1 Εφαρμογή για την αίτηση χορήγησης βεβαίωσης φοίτησης	76
5.2.1.1 Γενική Περιγραφή Υπηρεσίας	76
5.2.1.2 Σενάριο Χρήσης.....	76
5.2.1.3 Υλοποίηση	77
5.2.2 Εφαρμογή Ανάθεσης Πτυχιακής Εργασίας.....	82
5.2.2.1 Γενική Περιγραφή.....	82
5.2.2.2 Σενάριο Χρήσης.....	82
5.2.2.3 Υλοποίηση	84
6. Επισκόπηση	94
6.1 Συμπεράσματα.....	94
6.2 Προτεινόμενη μελλοντική μελέτη και πιθανές επεκτάσεις.....	96
Βιβλιογραφία	98

Πρόλογος

Η εκπόνηση της παρούσας πτυχιακής εργασίας ξεκίνησε στο Τμήμα Πληροφορικής και Τηλεματικής του Χαροκοπείου Πανεπιστημίου το Φεβρουάριο του 2011 και ολοκληρώθηκε τον Σεπτέμβριο του 2011. Η παρούσα εργασία αποτελεί φυσική συνέχεια εκείνης του φοιτητή Χούδρα Βαλεντίνο, με τίτλο: «Περιβάλλον Κοινωνικής Δικτύωσης για την Επικοινωνία των Διδασκόντων και των Φοιτητών του Πανεπιστημίου – Διαχείριση Σχέσεων».

Σε αυτό το σημείο επιθυμώ να εκφράσω τις ευχαριστίες μου σε όλους εκείνους των οποίων η συμβολή ήταν καθοριστική για την ολοκλήρωση του πονήματος αυτού:

Αρχικά, θα ήθελα να εκφράσω τις θερμότερες μου ευχαριστίες στην Αναπληρώτρια Καθηγήτρια, Μάρα Νικολαΐδου, επιβλέπουσα της παρούσας εργασίας, για την ασύγκριτη υποστήριξη, την καλή θέληση και το ενδιαφέρον που έδειξε προς την προσπάθεια που κατέβαλλα, όπως επίσης και συνολικά για το άψογο κλίμα που καλλιέργησε και το υψηλό επίπεδο σπουδών που διατήρησε σε συνεργασία και με τα υπόλοιπα μέρη του διδακτικού προσωπικού στο χώρο του Χαροκοπείου Πανεπιστημίου.

Εν συνεχεία, θα ήθελα να εκφράσω τις ευχαριστίες μου στα υπόλοιπα μέλη τις τριμελούς επιτροπής εξέτασης της πτυχιακής μου εργασίας: τον Πρύτανη του Χαροκοπείου Πανεπιστημίου και Πρόεδρο του Τμήματος Πληροφορικής και Τηλεματικής, Καθηγητή Δημοσθένη Αναγνωστόπουλο, όπως επίσης και τον Λέκτορα του εν λόγω Τμήματος, Ηρακλή Βαρλάμη για τα απολύτως εποικοδομητικά σχόλια και την αμέριστη υποστήριξη που επέδειξαν, καθ' όλη τη διάρκεια της προσπάθειας μου.

Επιπλέον, θα ήθελα να ευχαριστήσω ιδιαίτερα την διδάσκουσα βάσει του Π.Δ. 407/80 Ουρανία Χατζή, η οποία συνέβαλλε καθοριστικά στην εκπόνηση της πτυχιακής μου εργασίας, όντας πάντοτε διαθέσιμη και θετικά προδιατεθειμένη για ουσιαστική καθοδήγηση, ηθική και τεχνική, σε οποιαδήποτε περίπτωση ανάγκης κατά τη διάρκεια της προσπάθειας μου. Ακόμα, θα ήθελα να ευχαριστήσω εξίσου τον υποψήφιο διδάκτορα Αλέξανδρο Νταή για την συνεισφορά, τις φρέσκες ιδέες και τα

εποικοδομητικά σχόλια, χωρίς τα οποία η εκπόνηση αυτής της πτυχιακής εργασίας θα μπορούσε να βρίσκεται σε λάθος δρόμο.

Επίσης, θα ήθελα να ευχαριστήσω τον φίλο και συμφοιτητή μου Βαλεντίνο Χούδρα για την καθ' όλα άψογη συνεργασία, την ασύγκριτη επικοινωνία και την μοναδική αλληλοϋποστήριξη που είχαμε κατά τη διαδικασία ικανοποίησης των διαφορετικών απαιτήσεων που ορίστηκαν στα πλαίσια του κοινού θέματος που επιλέξαμε. Επιπλέον, θα ήθελα να ευχαριστήσω τον φίλο και συμφοιτητή μου Γεώργιο Μπατιστάτο για την αρμονική συνύπαρξη στο χώρο που εργαζόμασταν και για την ευγενική του συνεισφορά σε πολλά μικρά, αλλά σημαντικά ζητήματα της πτυχιακής μου εργασίας. Τέλος, θα ήθελα να ευχαριστήσω και τον φίλο και συμφοιτητή Ανδρέα Καπουράνη για τη μοναδική ηθική του υποστήριξη προς εμένα, όπως επίσης και η πίστη που έδειξε στις δυνατότητες μου, γεγονός που με βοήθησε να ανταπεξέλθω σε πολλά δύσκολα σημεία.

Αναμφίβολα επίσης, θα ήθελα να ευχαριστήσω την οικογένεια μου για την ασύγκριτη υπομονή, κατανόηση και καλή διάθεση που έδειξε καθόλη τη διάρκεια εκπόνησης της πτυχιακής μου εργασίας, υποστηρίζοντας με και προσφέροντας μου ό,τι ακριβώς χρειαζόμουν και ό,τι ακριβώς εύχεται κανείς από μία οικογένεια τόσο σε καθημερινές, όσο και σε δύσκολες στιγμές.

Τέλος, θα ήθελα να ευχαριστήσω τον εκλιπόντα πρώην Πρύτανη και Καθηγητή Γεώργιο Καραμπατζό για τη δημιουργία και τη διαμόρφωση του τέλειου κλίματος που είχα την τύχη να συναντήσω στο Χαροκόπειο Πανεπιστήμιο. Οι υπηρεσίες και το όραμα του για τη φοιτητική κοινότητα στάθηκαν φάρος στα φοιτητικά μας χρόνια και δεν πρόκειται να ξεχαστούν ποτέ.

Περίληψη

Η παρούσα πτυχιακή εργασία αφορά έναν ιστότοπο με το όνομα Unity, που βασίζεται στη λογική της κοινωνικής δικτύωσης και στοχεύει στη διευκόλυνση της συνεργασίας των μελών της ακαδημαϊκής κοινότητας με ηλεκτρονικό τρόπο. Στον ιστότοπο αυτό συμμετέχουν άτομα, όπως για παράδειγμα διδάσκοντες και φοιτητές (προπτυχιακοί ή μεταπτυχιακοί), οι οποίοι έχουν προκαθορισμένους ρόλους και συνεργάζονται με βάση κανόνες που προσδιορίζονται από του ρόλους αυτούς, σε αντίθεση με ένα τυπικό κοινωνικό δίκτυο, του οποίου τα μέλη αλληλεπιδρούν αυθαίρετα.

Η παρούσα εργασία επικεντρώνεται στην ενσωμάτωση εφαρμογών στην πλατφόρμα Unity, οι οποίες προσδίδουν πρόσθετη λειτουργικότητα στην υπάρχουσα πλατφόρμα. Οι εφαρμογές αυτές κατά κύριο λόγο αποτελούν την υλοποίηση υπηρεσιών που προσφέρονται στο εσωτερικό μίας ακαδημαϊκής κοινότητας, μεταξύ των μελών της. Η ενσωμάτωση τέτοιου είδους εφαρμογών στοχεύει στην αυτοματοποίηση συγκεκριμένων ακαδημαϊκών διαδικασιών, ώστε αυτές να επιταχύνονται και να οργανώνονται καλύτερα, εφόσον θα πραγματοποιούνται ηλεκτρονικά. Για την υλοποίηση του ιστότοπου ακαδημαϊκής κοινωνικής δικτύωσης, χρησιμοποιήθηκε το ανοικτό πρότυπο OpenSocial και, πιο συγκεκριμένα, η υλοποίησή του, το Apache Shindig. Με τη χρήση της τεχνολογίας αυτής, η πλατφόρμα Unity περιλαμβάνει τις σημαντικότερες λειτουργίες κοινωνικής δικτύωσης, ενώ ταυτόχρονα χρησιμοποιεί υποδομές που ορίζονται από ανοικτά πρότυπα για την προσφορά υπηρεσιών με τη μορφή εφαρμογών. Για το σκοπό αυτό και ειδικότερα για την υποστήριξη διαφορετικών κατηγοριών συμμετεχόντων και την υποστήριξη εξειδικευμένων σχέσεων, το πρότυπο του OpenSocial έχει επεκταθεί τόσο όσον αφορά τη δομή των δεδομένων, όσο και τις κλήσεις που ορίζει, μέσα από την επέκταση της αντίστοιχης τεχνολογίας του Apache Shindig. Εφόσον πραγματοποιηθεί η διαδικασία επέκτασης, η εν λόγω υποδομή είναι πλέον έτοιμη να υποστηρίξει την ενσωμάτωση και την χρήση εφαρμογών, οι οποίες αναγνωρίζουν την ιδιαίτερη φύση του κοινωνικού γράφου της υπηρεσίας.

Abstract

This thesis refers to an academic networking platform named “Unity”, which facilitates the electronic cooperation between members of the academic community. Unity brings together member of the University with different roles, who cooperate according to rules engendered by them, unlike typical popular social network, in which participants interact as they please. This cooperation is determined by several types of relationships, which are defined within the academic social network, divided into two categories: social and academic.

This thesis focuses on the integration of applications that bring additional functionality to the existing platform. These applications are primarily implementing services offered to the members of an academic community. The integration of such applications aims at automating specific academic processes, so that they are quicker completed and better organized.

The aforementioned academic social networking site was built based on the OpenSocial open standard and its implementation: Apache Shindig. By integrating this technology, Unity supports all major social networking features, while using the structures defined by open standards to offer services in the form of applications. For this purpose and in order to support different types of participants and special relations between them, the OpenSocial standard has been extended both in terms of data structures and calls made through the Apache Shindig technology. Once the extension process is done, Unity is ready to support the integration and use of applications that acknowledge the special nature of its community.

Λέξεις κλειδιά

Κοινωνικό δίκτυο, ιστότοπος κοινωνικής δικτύωσης, κοινωνική δικτύωση, ακαδημαϊκή κοινωνική δικτύωση, κοινωνικός γράφος, σχέσεις, επαφές, συμμετέχοντες, εφαρμογές, υπηρεσίες.

Keywords

Social Network, social networking site, social networking, academic social networking, social graph, relations, contacts, participants, applications, services.

1. Εισαγωγή

1.1 Βασικοί ορισμοί

Κοινωνικό δίκτυο είναι μία δομή αποτελούμενη από κόμβους, κάθε ένας από τους οποίους αντιπροσωπεύει ένα άτομο ή οργανισμό. Οι κόμβοι αυτοί είναι συνδεδεμένοι μεταξύ τους με μία σχέση αλληλεξάρτησης (interdependence), όπως για παράδειγμα φιλία, συγγένεια, σεξουαλικό ενδιαφέρον, οικονομική συναλλαγή, κοινά ενδιαφέροντα και πεποιθήσεις (Wikipedia, 2011).

Το σύνολο των κόμβων και των μεταξύ τους σχέσεων σε ένα κοινωνικό δίκτυο αποτελεί έναν περίπλοκο γράφο, του οποίου το μέγεθος μπορεί να περιορίζεται στα πλαίσια μίας οικογένειας ή να εκτείνεται ακόμα και σε διεθνές επίπεδο. Οι κόμβοι που είναι συνδεδεμένοι με ένα άτομο αποτελούν τις *κοινωνικές επαφές* ή, απλούστερα, τις *επαφές* του. Το άτομο με τις επαφές του έχει μία αποκλειστική σχέση επικοινωνίας και αλληλεπίδρασης, οι παράμετροι της οποίας περιγράφονται από μία δεδομένη ιστοσελίδα κοινωνικής δικτύωσης.

Υπηρεσία ή ιστότοπος κοινωνικής δικτύωσης είναι μία διαδικτυακή υπηρεσία, πλατφόρμα ή ιστότοπος (ΙΚΔ) που περιγράφει και αναπαριστά τη δομή ενός κοινωνικού δικτύου, καθώς επίσης και τους επιμέρους κόμβους-χρήστες και σχέσεις. Συγκεκριμένα:

1. Παρέχει σε κάθε άτομο-χρήστη ένα δικό του ψηφιακό *προφίλ* όπου προβάλλονται οι προσωπικές πληροφορίες που έχει επιλέξει.
2. Σε κάθε χρήστη δίνει πρόσβαση σε μία λίστα ατόμων με τα οποία είναι συνδεδεμένος (επαφές) και δυνατότητα αλληλεπίδρασης μαζί τους με τη χρήση διαφόρων τεχνολογιών, όπως την υπηρεσία ηλεκτρονικού ταχυδρομείου (e-mail) ή την υπηρεσία άμεσης ανταλλαγής μηνυμάτων (instant messaging).
3. Παρέχει τη δυνατότητα αναζήτησης των λιστών επαφών κάποιου χρήστη, το οποίο οδηγεί στη σύναψη δεσμών μεταξύ όλο και περισσότερων χρηστών.

Ο όρος *δικτύωση* αναφέρεται περισσότερο στην σύναψη σύνδεσης, σχέσης δηλαδή μεταξύ ατόμων που έχουν παρεμφερείς κοινωνικούς κύκλους ή κάποιο άλλο κοινό

σημείο αναφοράς. Η πρωτοπορία των κοινωνικών δικτύων λοιπόν έγκειται στη δημιουργία αλυσίδων γνωριμιών, οι οποίες καθιστούν εύκολη την μεταβίβαση πληροφορίας στον κοινωνικό γράφο (social graph).

Οι σχέσεις που συνάπτονται κατά τη διαδικασία της κοινωνικής δικτύωσης κατατάσσονται σε δύο κατηγορίες: τις *μονόδρομες (μονομερείς)* και τις *αμφίδρομες*. Μονόδρομες είναι οι σχέσεις οι οποίες συνδέουν δύο κόμβους στον κοινωνικό γράφο μονόδρομα. Σε αυτή την περίπτωση, η μία οντότητα συνδέεται με την άλλη, χωρίς να συμβαίνει απαραίτητα το αντίστροφο. Αντίθετα, στις αμφίδρομες σχέσεις υπάρχει αμφίδρομη επικοινωνία μεταξύ των κόμβων στο κοινωνικό γράφο, πράγμα που σημαίνει πως η σύνδεση των αντίστοιχων οντοτήτων συντελείται αμφιμερώς.

Επιπλέον, σε κάθε σύναψη σχέσης στα πλαίσια της κοινωνικής δικτύωσης, ορίζονται δύο συναλλασσόμενοι: ο *αιτών (requestor)* και ο *παραλήπτης (receiver)*. Ο αιτών είναι η οντότητα που πραγματοποιεί την αίτηση για την σύναψη σχέσης, ενώ ο παραλήπτης είναι η οντότητα που την δέχεται ή την απορρίπτει. Από εδώ και στο εξής, οι εν λόγω συναλλασσόμενοι θα αναφέρονται ως *υποκείμενο* και *αντικείμενο* της σχέσης, αντίστοιχα.

Τέλος, ως *συμμετέχοντες* σε μία κοινωνική συναλλαγή ορίζονται οι χρήστες εκείνοι του ΙΚΔ που καλούνται να αλληλεπιδράσουν είτε μέσω θεμελιωδών λειτουργιών αυτού, όπως είναι η σύναψη φιλίας, η δημοσίευση ενός μηνύματος κλπ, είτε μέσω κάποιων πρόσθετων εφαρμογών. Ουσιαστικά «αποτάσσονται» την απλή ιδιότητα του χρήστη, καθώς δεν χρησιμοποιούν ή «καταναλώνουν» μόνο πληροφορία, αλλά προσφέρουν και τη δική τους και συμμετέχουν στην μετάδοση πληροφορίας στον κοινωνικό γράφο της υπηρεσίας σε συνεργασία πάντα και με άλλους κόμβους.

1.2 Τυπική λειτουργικότητα Ιστότοπου Κοινωνικής Δικτύωσης

1.2.1 Βασικές Λειτουργίες Ιστότοπων Κοινωνικής Δικτύωσης

Κάθε ΙΚΔ απαρτίζεται από χρήστες-μέλη, οι οποίοι έχουν εγγραφεί σε αυτόν και, κατά συνέπεια, έχουν αποκτήσει κάποιο προφίλ. Κάνοντας είσοδο (login) στην

υπηρεσία με ένα μοναδικό αναγνωριστικό, έχουν τη δυνατότητα να επεξεργαστούν τις προσωπικές πληροφορίες ή το υλικό που εμφανίζεται στο προφίλ τους. Τέτοιες προσωπικές πληροφορίες μπορεί να είναι: προσωπικά ενδιαφέροντα, στοιχεία επικοινωνίας ή κατοικίας, πολιτικές και θρησκευτικές πεποιθήσεις κ.ά. Επιπλέον, πολλές φορές έχουν τη δυνατότητα να το καταστήσουν εν μέρει ή εξ ολοκλήρου ορατό από ομάδες χρηστών, ρυθμίζοντας τον βαθμό πρόσβασης αυτών στα προσωπικά τους στοιχεία.

Τα εν λόγω δικαιώματα πρόσβασης ρυθμίζονται σε μεγάλο βαθμό από τις *σχέσεις* που αποκτούν δύο συμμετέχοντες μεταξύ τους. Σε πολλές περιπτώσεις, οι σχέσεις αυτές είναι αμφίδρομες και απαιτούν την επικύρωση ενός αιτήματος φιλίας που αποστέλλει ένας συμμετέχων σε κάποιον άλλο, όπως για παράδειγμα η σχέση «φίλος» στο Facebook. Μετά την επικύρωση αυτή, ο χρήστης-αιτών αποκτά μία σχέση με τον χρήστη-παραλήπτη και αντίστροφα. Εναλλακτικά, σε πολλούς ΙΚΔ συναντώνται και *μονόδρομες σχέσεις*, κατά τις οποίες ένας συμμετέχων «εγγράφεται» στο προφίλ κάποιου άλλου ή, απλά, τον «ακολουθεί» για τυχόν ενημερώσεις και δημοσιεύσεις, όπως για παράδειγμα στο Twitter.

Οι σχέσεις, τις οποίες αποκτούν οι συμμετέχοντες σε έναν ΙΚΔ, ρυθμίζουν σε μικρό ή μεγάλο βαθμό τον βαθμό αλληλεπίδρασης που μπορούν να έχουν αυτοί μεταξύ τους. Με άλλα λόγια, οι σχέσεις δίνουν νόημα σε κάποιες δυνατότητες που παρέχει ένας ΙΚΔ μέσω διαφόρων λειτουργιών του, διευκολύνοντας την επικοινωνία και κατ' επέκταση την κοινωνική δικτύωση μεταξύ των συμμετεχόντων. Τέτοιες δυνατότητες μπορεί να είναι:

Ανταλλαγή μηνυμάτων. Σε κάθε ΙΚΔ δίνεται η δυνατότητα σε χρήστες να δημοσιεύουν μηνύματα που αφορούν προσωπικές τους σκέψεις ή ιδέες στο προφίλ τους ή σε προφίλ άλλων. Οι χρήστες που έχουν κάποια σχέση με αυτούς μπορούν συνήθως να δουν και να σχολιάσουν αυτές τις δημοσιεύσεις. Ακόμα, σε κάποιες περιπτώσεις υποστηρίζεται και η δυνατότητα ανταλλαγής άμεσων μηνυμάτων (chat) ή επίσης και η επικοινωνία μέσω κάμερας, όπως για παράδειγμα στο Google+.

Ανάρτηση υλικού. Πρόκειται για άλλο ένα χαρακτηριστικό που απαντάται σε όλους τους ΙΚΔ. Οι χρήστες έχουν τη δυνατότητα να αναρτούν εικόνες, βίντεο ή διευθύνσεις ιστοσελίδων από το διαδίκτυο, αναμένοντας πολλές φορές σχόλια ή απαντήσεις από άλλους.

Δημιουργία και συμμετοχή σε εικονικές ομάδες (groups). Πρόκειται για ομάδες ατόμων με κοινά ενδιαφέροντα, ιδέες ή πεποιθήσεις, οι οποίοι αποκτούν μία ιστοσελίδα στα πλαίσια μίας υπηρεσίας κοινωνικής δικτύωσης, όπου μπορούν να αλληλεπιδρούν συγκεντρωτικά και να δημιουργούν τις δικές τους μικρές, διαδικτυακές «κοινότητες».

Αλληλεπίδραση μέσω εφαρμογών. Σε πολλές ΙΚΔ, οι συμμετέχοντες μπορούν να αποκτούν πρόσβαση σε εφαρμογές (applications), οι οποίες παρέχουν επιπρόσθετη λειτουργικότητα από εκείνη του ιστότοπου. Πρόκειται κυρίως για παιχνίδια ή εφαρμογές που ενσωματώνουν τη λειτουργικότητα ξένων υπηρεσιών στον ΙΚΔ.

1.2.2 Δημιουργία και χρήση εφαρμογών

Δεν είναι λίγοι οι ΙΚΔ που προσφέρουν τη δυνατότητα δημιουργίας και φιλοξενίας εφαρμογών, στοχεύοντας στη δραστηριοποίηση προγραμματισμών για την επέκταση της λειτουργικότητας τους, μέσω της οποίας θα βελτιωθεί και η συνολική εμπειρία κοινωνικής δικτύωσης των χρηστών. Φροντίζουν, λοιπόν, να δημοσιοποιούν κάποιες προγραμματιστικές διεπαφές (APIs) για την εξαγωγή πληροφορίας από τον κοινωνικό τους γράφο. Τέτοιες διεπαφές παρέχουν κλήσεις σε προγραμματιστές, με την βοήθεια των οποίων εξάγουν την πληροφορία αυτή και, χρησιμοποιώντας την, κατασκευάζουν εφαρμογές, εσωτερικές ή εξωτερικές στην υπηρεσία.

Εξωτερικές εφαρμογές είναι εκείνες, οι οποίες ενσωματώνονται σε εξωτερικές ιστοσελίδες, ξένες προς τον ΙΚΔ και περικλείουν συνήθως ένα μέρος της λειτουργικότητας του. Χαρακτηριστικό παράδειγμα τέτοιων εφαρμογών είναι διάφορες υπηρεσίες ιστοσελίδων σε μορφή widgets, μέσω των οποίων μπορεί ένας χρήστης να πραγματοποιεί λειτουργίες ενός ΙΚΔ (πχ. δημοσίευση μηνυμάτων), χωρίς να τον έχει επισκεφτεί. Επίσης, τέτοιες εφαρμογές συναντά κανείς και σε κινητές συσκευές, οι οποίες κάνουν χρήση των δημοσιοποιημένων APIs, ενσωματώνοντας γνωστές λειτουργίες ΙΚΔ.

Αντίθετα, *εσωτερικές εφαρμογές* είναι εκείνες, οι οποίες ενσωματώνονται στο εσωτερικό του ΙΚΔ και επεκτείνουν τη λειτουργικότητα του. Χαρακτηριστικό παράδειγμα τέτοιων εφαρμογών είναι τα παιχνίδια που συναντά κανείς σε έναν ΙΚΔ ή ακόμα και διάφορες υπηρεσίες για ανταλλαγή μηνυμάτων. Για την πρόσβαση των χρηστών προς αυτές, συνήθως απαιτείται κάποια επιβεβαίωση των στοιχείων τους

και η χορήγηση δικαιωμάτων προς την αντίστοιχη εφαρμογή για ανάγνωση/επεξεργασία των πληροφοριών του χρήστη.

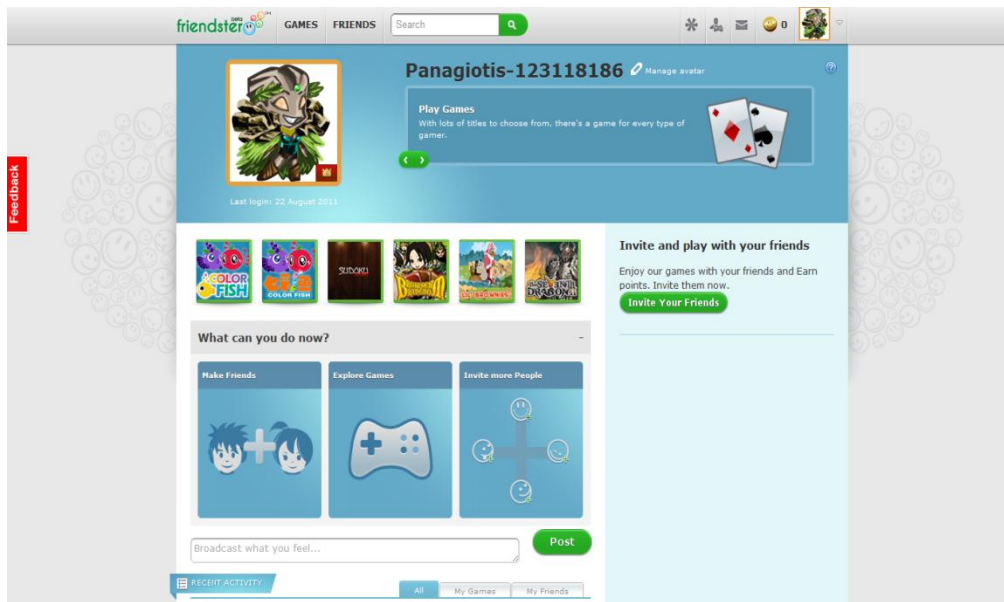
Εκτός από τα απαραίτητα εργαλεία για τη δημιουργία εσωτερικών εφαρμογών, ένας ΙΚΔ παρέχει συνήθως και κάποια εργαλεία για την ενσωμάτωση αυτής σε συγκεκριμένα σημεία του. Συγκεκριμένα, κάθε εφαρμογή ενσωματώνεται ως *gadget*, δηλαδή ως ένα στιγμιότυπο της λειτουργικότητας της, αποδιδόμενο συνήθως σε γλώσσα JavaScript. Για την πρόσβαση των χρηστών του ΙΚΔ προς αυτά τα gadgets, συνήθως απαιτείται κάποια επιβεβαίωση των στοιχείων τους και η χορήγηση δικαιωμάτων προς την αντίστοιχη εφαρμογή για ανάγνωση/επεξεργασία των πληροφοριών του χρήστη.

1.3 Υποστήριξη και Ανάπτυξη Εφαρμογών σε Δημοφιλείς Ιστότοπους Κοινωνικής Δικτύωσης

1.3.1 Friendster

Το Friendster αποτελεί έναν από τους πρώτους δημοφιλείς ΙΚΔ, ο οποίος μέχρι πρόσφατα παρείχε συμβατική λειτουργικότητα που συναντά κανείς σε πολλούς άλλους. Πλέον, όμως, λειτουργεί ως «ιστότοπος κοινωνικής διασκέδασης», εστιάζοντας περισσότερο σε εφαρμογές παιχνιδιών και μουσικής. Συγκεκριμένα, σε κάθε χρήστη παρέχει ένα ψηφιακό προφίλ που ονομάζεται Avatar (Εικόνα 1.1), μέσω του οποίου δρα στα πλαίσια της υπηρεσίας.

Η αλληλεπίδραση μεταξύ των χρηστών ορίζεται κατά κύριο λόγο από τις μεταξύ τους σχέσεις. Βάσει αυτών, μπορούν να επικοινωνούν μεταξύ τους με ανταλλαγή μηνυμάτων ή να συμμετέχουν από κοινού σε εφαρμογές-παιχνίδια. Η πρόοδος τους στα παιχνίδια αυτά ανταμείβεται με ψηφιακά κέρματα (coins), με τα οποία μπορούν να εξαργυρώνουν διάφορες ψηφιακές ή εμπορικές προσφορές.



Εικόνα 1.1 – Αρχική Σελίδα του Friendster το 2011

Πριν τη νέα έκδοση του Friendster, μέσω της υπηρεσίας Friendster Developer Platforms, παρέχονταν στους χρήστες η δυνατότητα να χρησιμοποιήσουν δεδομένα από τον ΙΚΔ για να δημιουργήσουν τις δικές τους εφαρμογές (Apps) πάνω σε αυτόν. Αυτές οι εφαρμογές μπορούσαν να φιλοξενηθούν στην υπηρεσία ως gadgets που ενσωματώνονταν στην ίδια μέσω της τεχνολογίας των *iframes* ή απλά να δρουν ως εξωτερικές εφαρμογές από την υπηρεσία.

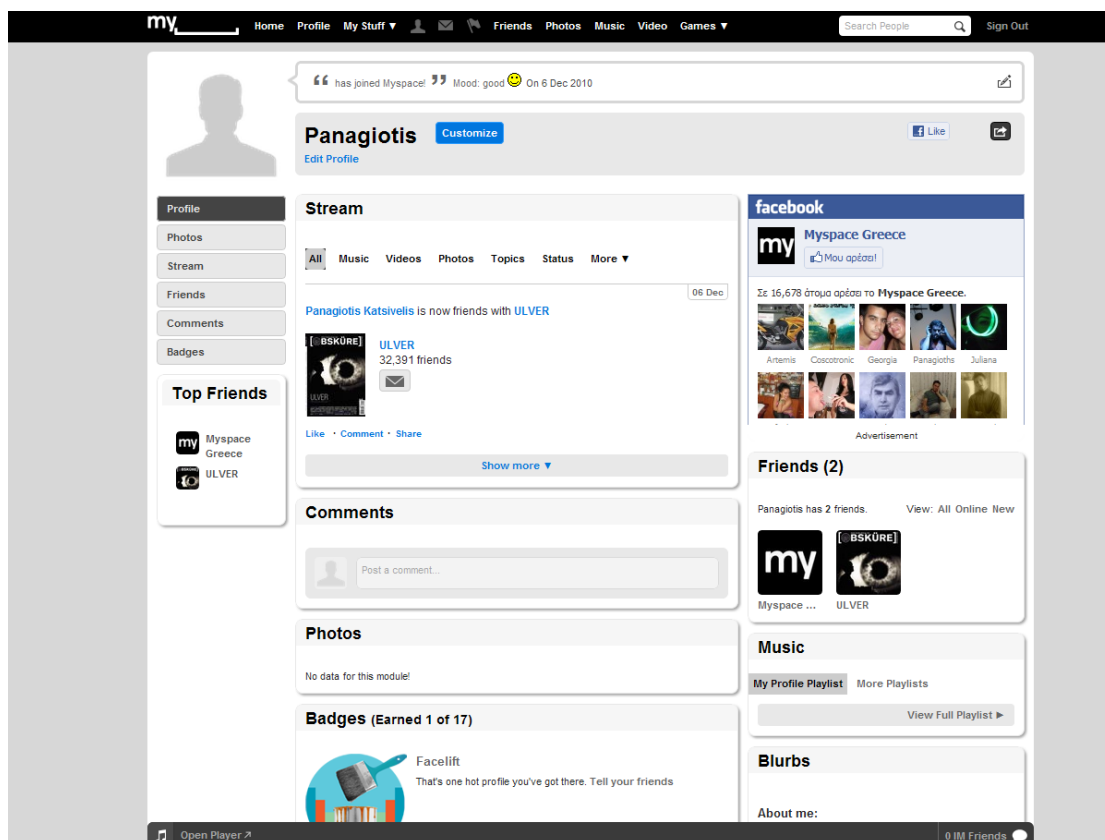
Οι προγραμματιστικές πλατφόρμες που παρείχε το Friendster για αυτό το σκοπό ήταν δύο:

- Η πλατφόρμα του Opensocial, στην οποία είχε στηριχτεί και το παλιό Friendster παρείχε ως ένα βαθμό έτοιμες συναρτήσεις και μηχανισμούς για την υποστήριξη και εξυπηρέτηση κλήσεων για την εξαγωγή και αποθήκευση πληροφορίας από και προς τον ΙΚΔ.
- Η πλατφόρμα V1, η οποία ήταν η πρώτη πλατφόρμα ανάπτυξης που διαμορφώθηκε για αυτό το σκοπό. Συγκεκριμένα, παρείχε πρόσβαση στα δεδομένα του Friendster μέσα από μία REST-side διεπαφή. Ο χρήστης πραγματοποιούσε κλήσεις προς συναρτήσεις του API του Friendster και ύστερα χειριζόταν τα δεδομένα που του επιστρέφονταν για τη δημιουργία μίας δικής του mashup εφαρμογής.

Παρόλο που πλέον δεν γίνεται χρήση των προαναφερθέντων προτύπων, το Friendster αποτελεί έναν ΙΚΔ, του οποίου η λειτουργία περιστρέφεται πλήρως γύρω από τη χρήση εφαρμογών. Όσον αφορά την προσθήκη εφαρμογών σήμερα, το Friendster δεν έχει προχωρήσει ακόμα σε έκδοση των νέων APIs που υποστηρίζει.

1.3.2 MySpace

Το MySpace αποτελεί έναν από τους δημοφιλέστερους ΙΚΔ στην ιστορία της κοινωνικής δικτύωσης. Στα οκτώ χρόνια λειτουργίας του έχει ενσωματώσει σε μεγάλο βαθμό τη βασική λειτουργικότητα ενός συμβατικού ΙΚΔ, δίνοντας στους χρήστες τη δυνατότητα για δημιουργία προφίλ (Εικόνα 1.2), σύναψη αμφίδρομων σχέσεων φιλίας, ανταλλαγή μηνυμάτων, ανάρτηση υλικού κλπ. Παράλληλα, όμως, κυρίως από το 2005 και ύστερα παρείχε νέες, εξειδικευμένες λειτουργίες προς τους χρήστες, δίνοντας τους τη δυνατότητα να δημιουργήσουν *καλλιτεχνικά* προφίλ. Μέσα από αυτά τα προφίλ μπορεί κανείς να προωθήσει καλλιτεχνικά και, κυρίως, μουσικά έργα, όπως επίσης να προσπελάσει και τα αντίστοιχα καλλιτεχνικά έργα άλλων.



Εικόνα 1.2 – Προφίλ χρήστη MySpace το 2011

Μέσω της υπηρεσίας MySpace Developer Platform, παρέχεται στους χρήστες η δυνατότητα να χρησιμοποιήσουν πληροφορία αποθηκευμένη στον ΙΚΔ για να δημιουργήσουν τις δικές τους εφαρμογές, εσωτερικές και εξωτερικές του MySpace.

Όπως και στο Friendster, οι εσωτερικές εφαρμογές του MySpace κάνουν χρήση της έκδοσης 1.0 του OpenSocial. Τέτοιες εφαρμογές ενσωματώνονται ως gadgets εντός της υπηρεσίας με τον μηχανισμό των *iframes*. Κάθε gadget είναι ένα αρχείο XML που περιλαμβάνει τα στοιχεία (features) που χρησιμοποιεί, όπως επίσης και κώδικα σε JavaScript, HTML και CSS που εκτελεί τις λειτουργίες του gadget.

Οι περισσότερες OpenSocial κλήσεις εντός του gadget πραγματοποιούνται με ετικέτες δεδομένων (data tags). Για παράδειγμα, η ετικέτα:

```
<os:PeopleRequest key='friends' userId="john.doe" groupid="@friends" />
```

αντιστοιχεί στην κλήση που επιστρέφει τους φίλους του χρήστη με αναγνωριστικό *john.doe*.

Τέλος, τα gadgets μπορούν να προστεθούν σε συγκεκριμένα σημεία του ΙΚΔ, τα οποία είναι:

- Λίστες Εφαρμογών (Application Listings).
- Προφίλ χρηστών (δημόσια και ορατά από όλους)
- Αρχική σελίδα υπηρεσίας (ορατά από τους χρήστες που τα εγκαθιστούν).
- Μία σελίδα «Canvas» (μία σελίδα αφιερωμένη εξολοκλήρου σε εφαρμογές).
- Ενημερώσεις φίλων
- Προσκλήσεις/Ενημερώσεις (προσκλήσεις χρηστών για χρήση εφαρμογών – οι εφαρμογές μπορούν να αποστείλουν μηνύματα στους χρήστες).

Όσον αφορά της εξωτερικές εφαρμογές του MySpace (MySpaceID), οι προγραμματιστές μπορούν να τις χρησιμοποιήσουν κάνοντας χρήση του REST API του MySpace, για να ανακτήσουν κοινωνική πληροφορία από τον ΙΚΔ. Σημειώνεται πως για την πρόσβαση στις εν λόγω εφαρμογές απαιτείται και η αντίστοιχη άδεια και επιβεβαίωση από το MySpace σύμφωνα με το πρότυπο OAuth.

Τέλος, αξίζει να σημειωθεί πως στα πλαίσια αυτής της πλατφόρμας, το MySpace παρέχει διάφορα Software Development Kits (SDK's) για την ανάπτυξη τέτοιων ειδών εφαρμογών στα πλαίσια του ΙΚΔ.

1.3.3 Facebook

Το Facebook.com ιδρύθηκε το 2004 από τον M. Zuckerberg, ως ένας ΙΚΔ που απευθυνόταν μόνο σε φοιτητές του πανεπιστημίου του Harvard στις ΗΠΑ. Σήμερα, το Facebook.com έχει παραπάνω από 500.000.000 εγγεγραμμένους ενεργούς χρήστες και παραπάνω από 1700 υπαλλήλους (Facebook, 2010), όντας ο πιο δημοφιλής ΙΚΔ στον κόσμο.



Εικόνα 1.3 – Προφίλ χρήστη Facebook το 2011

Για ακόμα μία φορά, οι λειτουργίες που παρέχει ο εν λόγω ΙΚΔ είναι γνωστές. Ένας χρήστης μπορεί να δημιουργήσει το αντίστοιχο προφίλ του, να συνάψει σχέσεις φιλίας με άλλους χρήστες ή να εγγραφεί σε ενημερώσεις των προφίλ τους, όπως επίσης και να συμμετάσχει σε εικονικές ομάδες, να αναρτά μηνύματα ή υλικό στο προφίλ του ή σε προφίλ άλλων και να σχολιάζει αντίστοιχες δημοσιεύσεις αυτών. Η πιο διάσημη λειτουργία του ΙΚΔ αυτού είναι ίσως η λειτουργία «like», με την οποία ένας χρήστης δηλώνει ότι του αρέσει μία ανάρτηση ενός άλλου χρήστη.

Όσον αφορά το κομμάτι των εφαρμογών, το Facebook υποστηρίζει υπηρεσίες για την έκθεση του κοινωνικού του γράφου. Πρακτικά, κάθε οντότητα στο εσωτερικό του ΙΚΔ χαρακτηρίζεται από ένα μοναδικό αναγνωριστικό (id). Το Graph API που έχει δημοσιεύσει το Facebook, φροντίζει για την εξεύρεση και ανάκτηση πληροφορίας που αφορά οντότητες με ένα συγκεκριμένο κωδικό. Για παράδειγμα, ένα παράδειγμα

αίτησης για ανάκτηση πληροφορίας μπορεί να είναι το URL: <https://graph.facebook.com/btaylor>, το οποίο επιστρέφει πληροφορίες για τον χρήστη με κωδικό «btaylor» σε μορφή JSON. Ίδιας μορφής αιτήματα μπορούν να πραγματοποιηθούν και για την ανάκτηση πληροφοριών για άλλα κοινωνικά αντικείμενα, όπως επίσης και για τους φίλους που έχει ένας χρήστης, τις σελίδες που ακολουθεί κ.ο.κ. Επιπλέον, υπάρχει και η δυνατότητα πραγματοποίησης επερωτήσεων στη βάση δεδομένων του ΙΚΔ, μέσω της FQL (Facebook Query Language), μίας γλώσσας που μοιάζει με SQL, αλλά που χρησιμοποιείται για εξαγωγή συγκεκριμένης κοινωνικής πληροφορίας. Αυτή την πληροφορία μπορούν να χρησιμοποιήσουν προγραμματιστές για να αναπτύξουν τις δικές τους εφαρμογές, εσωτερικές ή εξωτερικές, βασισμένες στον κοινωνικό γράφο του ΙΚΔ.

Όσον αφορά την υποστήριξη εσωτερικών εφαρμογών, όπως για παράδειγμα παιχνίδια, το Facebook τις ενσωματώνει σε μία κενή περιοχή που ονομάζεται Canvas. Στο συγκεκριμένο σημείο, φορτώνεται το περιεχόμενο μίας σελίδας όπου «τρέχει» η εφαρμογή μέσω της HTML ταμπέλας *iframe*. Η εφαρμογή φροντίζει να προσκομίζει τα απαραίτητα στοιχεία του χρήστη που την προσπελαύνει, μέσω ενός μηχανισμού OAuth, εφόσον φυσικά ο ίδιος χρήστης το επιτρέψει. Από αυτό το σημείο, η εφαρμογή μπορεί να ανακτήσει, να δημοσιεύσει πληροφορίες, αλλά και να αποστείλει αιτήματα σε χρήστες, μέσω διάφορων «κοινωνικών καναλιών» (social channels), όπως ορίζονται από το Facebook.

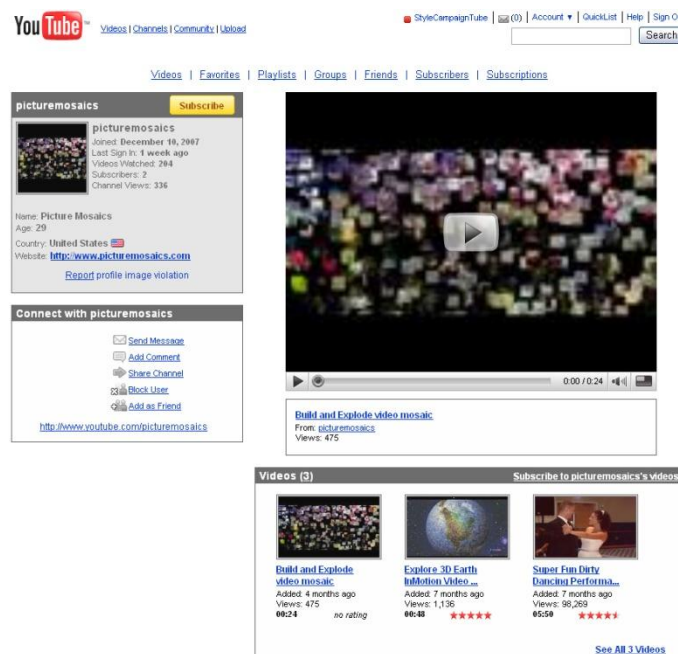
Τέλος, η υπηρεσία παρέχει στους προγραμματιστές διάφορα Εργαλεία Ανάπτυξης Εφαρμογών (SDK's) για διάφορες πλατφόρμες, αλλά και διάφορα προγραμματιστικά γούστα. Συγκεκριμένα, παρέχονται τα εξής:

- Javascript SDK, για την πρόσβαση στην κοινωνική πληροφορία μέσω Javascript
- PHP SDK, για την πρόσβαση στην κοινωνική πληροφορία μέσω PHP
- iOS SDK, για την πιο εύκολη ενσωμάτωση της πλατφόρμας του Facebook σε iPhones
- Android SDK, για την πιο εύκολη ενσωμάτωση της πλατφόρμας του Facebook σε Android smartphones.

YouTube 1.3.4 YouTube

Το YouTube αποτελεί έναν από τους διασημότερους ιστότοπους παγκοσμίως, ο οποίος έχει ενσωματώσει σε μεγάλο βαθμό στοιχεία κοινωνικής δικτύωσης. Συγκεκριμένα, πρόκειται για μια διαδικτυακή κοινότητα όπου οι χρήστες αναρτούν, μοιράζονται ή απλά βλέπουν βίντεο για να διατηρήσουν, εξελίξουν ή να δημιουργήσουν κοινωνικές σχέσεις (Lange, 2007). Στον ιστότοπο αυτό, το προφίλ που συναντά κανείς αποκαλείται «κανάλι» (channel), το οποίο περιέχει όλες τις προσωπικές πληροφορίες κάθε χρήστη.

Στο YouTube δίνεται η δυνατότητα δημιουργίας σχέσης φιλίας μεταξύ συμμετεχόντων, όπως επίσης και η εγγραφή ενός συμμετέχοντος στο κανάλι κάποιου άλλου (Εικόνα 1.4), για ενημέρωση σχετικά με την ανάρτηση ενός νέου βίντεο. Επιπλέον, στους συμμετέχοντες δίνεται η δυνατότητα επικοινωνίας μέσω ανταλλαγής προσωπικών μηνυμάτων, ανάρτησης μηνυμάτων σε κανάλια άλλων, όπως επίσης και μέσω του σχολιασμού ή της βαθμολόγησης των βίντεο που έχουν αναρτήσει οι ίδιοι ή άλλοι. Τέλος, στο YouTube παρέχονται κάποιες, εξειδικευμένες λειτουργίες, όπως είναι η αναζήτηση βίντεο ή η δημιουργία λίστας αναπαραγωγής κάποιας συλλογής από βίντεο.



Εικόνα 1.4 – «Κανάλι» χρήστη στο YouTube.com το 2011

Όπως γίνεται εύκολα αντιληπτό, ο χαρακτήρας του YouTube είναι προσανατολισμένος γύρω από μία λειτουργία: την προβολή βίντεο. Συνεπώς, αυτός ο ιδιαίτερος ΙΚΔ δεν υποστηρίζει τη φιλοξενία εσωτερικών εφαρμογών. Ωστόσο, παρέχει κάποια APIs, μέσω των οποίων ένας χρήστης μπορεί να προσπελάσει την πληροφορία (κοινωνική και μη) που παρέχει η υπηρεσία, για χρήση αυτής σε εξωτερικές εφαρμογές και όχι μόνο. Πιο συγκεκριμένα, τα παρεχόμενα APIs είναι δύο:

- Data API, το οποίο βοηθά στην ενσωμάτωση λειτουργικότητας της υπηρεσίας σε ιστοσελίδες και εξωτερικές εφαρμογές. Η λειτουργικότητα αυτή αφορά όχι μόνο βίντεο, αλλά και σχόλια σε αυτά ή ακόμα και σχέσεις μεταξύ χρηστών.
- Player APIs, τα οποία δίνουν τη δυνατότητα ελέγχου των ιδιοτήτων του Video Player του YouTube που έχει ενσωματωθεί σε μία ιστοσελίδα.

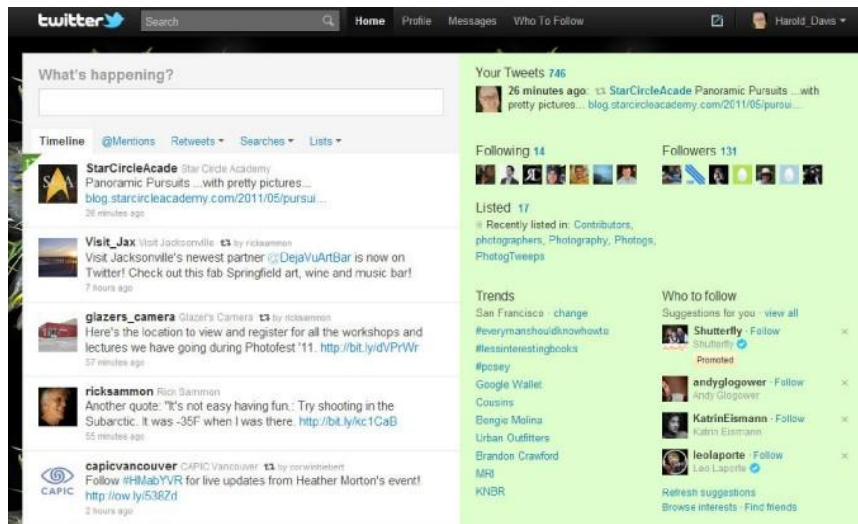
Παράλληλα, το YouTube παρέχει κάποια εργαλεία για την εμφάνιση στατιστικών στοιχείων που αφορούν συγκεκριμένα βίντεο μέσω των αιτημάτων προς τα API (Developer Dashboard), όπως επίσης και εργαλεία για την διακίνηση βίντεο και την αλληλεπίδραση χρηστών μέσω αυτής από ξένες ιστοσελίδες (YouTube Direct).



1.3.5 Twitter

Το Twitter είναι ένας απλοϊκός στη σύλληψη, αλλά αρκετά δημοφιλής ΙΚΔ. Η κεντρικότερη λειτουργία που παρέχει είναι το *tweet*, δηλαδή η δημοσίευση ενός περιορισμένου μήκους μηνύματος άμεσα, το οποίο σε κάποιο βαθμό συνδυάζει χαρακτηριστικά άμεσης επικοινωνίας και blogs. Η δημοσίευση αυτή συνήθως απαντά άτυπα στην ερώτηση: «Τι κάνεις τώρα;», η οποία τίθεται προς τον χρήστη (Edward Mischaud, 2007). Παράλληλα, η δημοσίευση αυτή μπορεί να επισημαίνεται (tag) και να κατηγοριοποιείται ή να προσωποποιείται, αναφερόμενη σε συγκεκριμένους συμμετέχοντες του ΙΚΔ.

Τα tweets ενός χρήστη μαζί με ελάχιστες προσωπικές πληροφορίες και μία προσωπική φωτογραφία αποτελούν το *προφίλ* του (Εικόνα 1.5).



Εικόνα 1.5 – Αρχική Σελίδα του Twitter το 2011

Μία σχέση στο Twitter εξασφαλίζει την πρόσβαση που έχει ένας συμμετέχων στα tweets ενός άλλου. Η σχέση αυτή ονομάζεται follower και είναι μονόδρομη, καθώς ένας συμμετέχων «ακολουθεί» κάποιον άλλον, έτσι ώστε να ενημερώνεται για νέα tweets του. Έτσι, για έναν συμμετέχοντα, τα άτομα που ακολουθεί αποτελούν τις *Following* επαφές του, ενώ αυτοί που τον ακολουθούν αποτελούν τους *Followers* του.

Εφόσον το Twitter είναι απλό στην φιλοσοφία του, δεν υποστηρίζει την ενσωμάτωση εφαρμογών στο εσωτερικό του, αλλά την ενσωμάτωση μέρους της λειτουργικότητας του σε εξωτερικές εφαρμογές. Προς αυτή την κατεύθυνση, έχουν δημοσιοποιηθεί τρία APIs:

- Search API

Με το Search API, δίνεται η ευκαιρία στους χρήστες να ασκούν επερωτήσεις (queries), ώστε να ανακτούν πληροφορία της υπηρεσίας, χρησιμοποιώντας λέξεις κλειδιά, αναφορές χρηστών σε tweets ή tweets συγκεκριμένων χρηστών.

- REST API

Με το Rest API, δίνεται η ευκαιρία στους χρήστες να προσπελαίνουν πρωταρχική κοινωνική πληροφορία του ΙΚΔ, όπως για παράδειγμα ημερομηνίες ενημερώσεων προφίλ χρηστών, αλλαγή status ή πληροφορίες

κάποιου χρήστη. Επιπλέον, μέσω του εν λόγω API, ένας χρήστης μπορεί να δημιουργεί και να δημοσιεύει tweets, να απαντήσει σε tweets και γενικά να προσπελάσει μεγάλο μέρος της λειτουργικότητας του ΙΚΔ εξωτερικά από αυτόν.

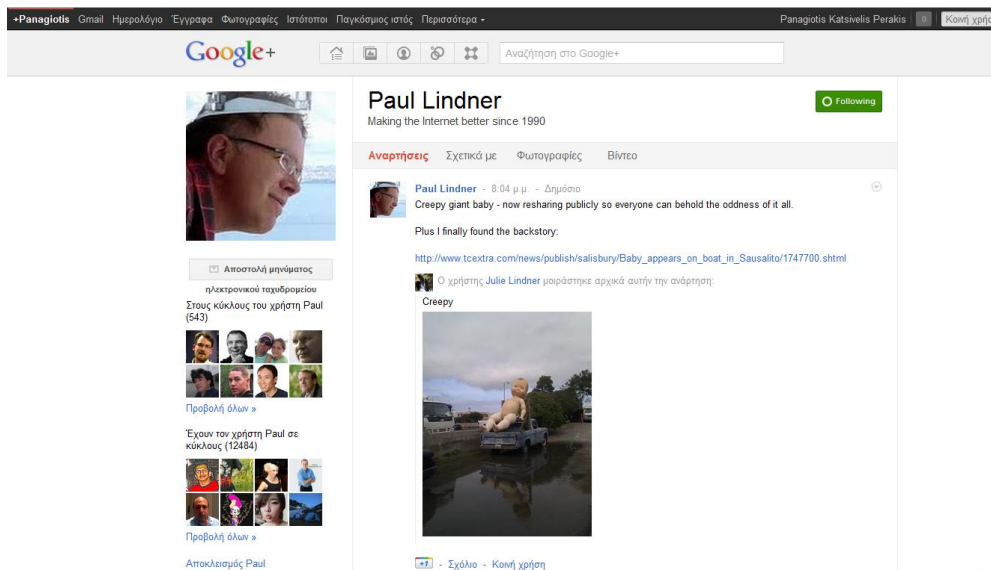
Σημειώνεται πως οι κλήσεις που μπορούν να πραγματοποιηθούν προς το API αυτό κατά τη διάρκεια μιας ώρας είναι περιορισμένες.

- Streaming API

Το Streaming API του Twitter αφορά χρήστες και προγραμματιστές που θέλουν να χρησιμοποιήσουν κυρίως μεγάλο όγκο πληροφορίας από την υπηρεσία, με σκοπό την εξαγωγή στατιστικών συμπερασμάτων. Η αλληλεπίδραση με το API αυτό προϋποθέτει την εγκαθίδρυση και συντήρηση μίας HTTP σύνδεσης.

1.3.6 Google+

Το Google+ (Google Plus) αποτελεί μία από τις σημαντικότερες πρόσφατες υπηρεσίες που ανέπτυξε η Google, αποτελώντας ένα αρκετά διαφορετικό και πολλά υποσχόμενο ΙΚΔ. Παρόλες τις ομοιότητες που παρουσιάζει με άλλους, «παραδοσιακούς» ΙΚΔ όσων αφορά τις βασικές λειτουργίες επικοινωνίας (δημοσίευση/σχολιασμός μηνυμάτων) και αυτοπροβολής (προφίλ για κάθε χρήστη – Εικόνα 1.6), το Google Plus πραγματοποιεί ένα άλλο είδος προσέγγισης στις σχέσεις που αναπτύσσονται μεταξύ των συμμετεχόντων. Συγκεκριμένα, ένας συμμετέχων πραγματοποιεί αποκλειστικά μονόδρομες σχέσεις, ακολουθώντας άλλους, και εντάσσοντας τους σε ομάδες ή *κύκλους επαφών*. Σε κάθε κύκλο μπορεί να προσδώσει συγκεκριμένα δικαιώματα πρόσβασης προς τις πληροφορίες που εμφανίζονται στο προφίλ του.



Εικόνα 1.6 – Προφίλ χρήστη στο Google Plus το 2011

Εφόσον υπάρχει η αντίστοιχη σχέση και τα αντίστοιχα δικαιώματα για μία συναλλαγή, οι συμμετέχοντες μπορούν να επικοινωνήσουν κυρίως με ανταλλαγή προσωπικών μηνυμάτων, τη δημοσίευση ή το σχολιασμών μηνυμάτων, τη δημοσίευση υλικού, καθώς και την ηλεκτρονική «συνάντηση» (Hangout) με ανταλλαγή άμεσων μηνυμάτων (chat) ή τη χρήση καμερών. Τέλος, μία ακόμα αξιοσημείωτη λειτουργία του Google Plus είναι η υπηρεσία +1, μέσω της οποίας οι συμμετέχοντες δηλώνουν ποια δημοσιευθέν αντικείμενο θεωρούν ενδιαφέρον

Όσον αφορά το κομμάτι των εφαρμογών, το Google Plus δημοσίευσε πρόσφατα μία σειρά από APIs για τη δημιουργία τόσο εσωτερικών, όσο και εξωτερικών εφαρμογών:

Οι *εξωτερικές εφαρμογές* κάνουν χρήση μίας διεπαφής που τους παρέχει τη δυνατότητα να πραγματοποιούν REST κλήσεις προς τον κοινωνικό γράφο της υπηρεσίας και να αποκομίσουν κοινωνική πληροφορία που αφορά: Άτομα (People), Δραστηριότητες Ατόμων (Activities) και Σχόλια σε δραστηριότητες (Comments).

Οι *εσωτερικές εφαρμογές*, αντίθετα με εκείνες σε πολλούς ΙΚΔ, προσανατολίζονται πλήρως στην επικοινωνία μεταξύ των συμμετεχόντων. Γι' αυτό το λόγο ενσωματώνονται αποκλειστικά στις τοποθεσίες άμεσης επικοινωνίας, δηλαδή τα Hangouts και αφορούν μόνο τους συμμετέχοντες σε αυτές. Συγκεκριμένα, οι εσωτερικές εφαρμογές κάνουν χρήση ενός JavaScript API που παρέχει συναρτήσεις που χειρίζονται τις εξής οντότητες:

- Συμμετέχοντες (participants) στο Hangout
- Hangout, δηλαδή οι παράμετροι της τοποθεσίας
- Πόροι του Hangout, δηλαδή οι κάμερες, τα μικρόφωνα και ηχεία
- Δεδομένα, τα οποία αφορούν την κατάσταση της εφαρμογής
- Περιβάλλον, δηλαδή η γραφική διεπιφάνεια (user interface), την οποία παρέχει η εφαρμογή.

Κάνοντας χρήση των παραπάνω συναρτήσεων, ένας προγραμματιστής μπορεί να δημιουργήσει μία εφαρμογή που να διαχειρίζεται τις παραμέτρους της επικοινωνίας, ώστε βάσει αυτών να δημιουργεί πιο διασκεδαστικά περιβάλλοντα αλληλεπίδρασης μεταξύ των συμμετεχόντων. Μία τέτοια εφαρμογή δεν είναι τίποτα άλλο από ένα XML αρχείο με ένα μπλοκ κώδικα σε JavaScript που χειρίζεται τα γεγονότα (events) που προκύπτουν κατά την αλληλεπίδραση των χρηστών. Εφόσον ο προγραμματιστής ολοκληρώσει την ανάπτυξη της εν λόγω εφαρμογής, οφείλει να την καταχωρήσει στην υπηρεσία, ώστε αυτή να γίνει διαθέσιμη προς χρήση από τους συμμετέχοντες στα Hangouts.

1.3.7 Συγκριτικός πίνακας υποστήριξης εφαρμογών από ΙΚΔ

Στον παρακάτω Πίνακα 1.1 εμφανίζονται οι δημοφιλέστεροι ΙΚΔ παγκοσμίως και το είδος των εφαρμογών για τις οποίες παρέχει διεπαφές κάθε ένας από αυτούς:

	Friendster (2011)	Myspace	Facebook	YouTube	Twitter	LinkedIn	Orkut	Hi5	Google+
Διεπαφές για ανάπτυξη και φιλοξενία εσωτερικών εφαρμογών		✓	✓			✓	✓	✓	✓
Διεπαφές για ανάπτυξη εξωτερικών		✓	✓	✓	✓	✓	✓	✓	✓

Πίνακας 1.1 – Λειτουργίες δημοφιλών Ιστοσελίδων Κοινωνικής Δικτύωσης

2. Τεχνολογικό Υπόβαθρο Ανάπτυξης Εφαρμογών σε Ιστότοπους Κοινωνικής Δικτύωσης

2.1 Αρχιτεκτονική και χρησιμοποιούμενη Τεχνολογία Ανάπτυξης Ιστότοπων Κοινωνικής Δικτύωσης

Η αρχιτεκτονική που ακολουθεί η συντριπτική πλειοψηφία των ΙΚΔ είναι αυτή των τριών επιπέδων(three-tier architecture), τα οποία έχουν ως εξής (Εικόνα 2.1):

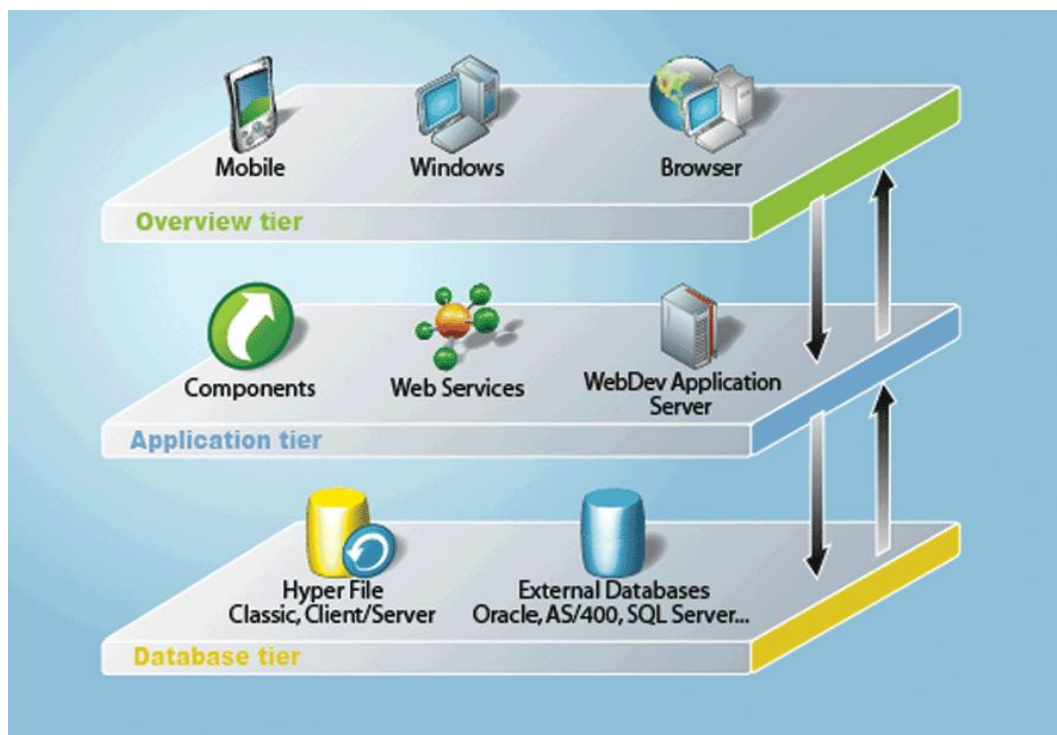
1. Επίπεδο παρουσίασης
2. Επίπεδο εφαρμογής
3. Επίπεδο βάσης δεδομένων

Στο υπόβαθρο της εν λόγω αρχιτεκτονικής βρίσκεται το επίπεδο βάσης δεδομένων, όπου συναντά κανείς την αποθήκη της κοινωνικής πληροφορίας μίας ΙΚΔ. Το επίπεδο εφαρμογή πραγματοποιεί κλήσεις προς αυτό για εγγραφή και ανάκτηση της πληροφορίας σε μορφή σχεσιακών δεδομένων. Παράλληλα, στο συγκεκριμένο επίπεδο συναντά κανείς και διάφορα συστήματα που εξυπηρετούν την επιτάχυνση των επερωτήσεων προς τη βάση δεδομένων ή φροντίζουν για τη διατήρηση των εξαγόμενων δεδομένων, με τελικό σκοπό την επεκτασιμότητα (scalability) της υπηρεσίας. Τέτοιο παράδειγμα μηχανισμού είναι το Memcached, το οποίο χρησιμοποιείται από πολλούς δημοφιλείς ΙΚΔ, όπως το Facebook.

Στο δεύτερο επίπεδο της αρχιτεκτονικής τριών επιπέδων βρίσκεται το επίπεδο εφαρμογής, στο οποίο βρίσκεται υλοποιημένη όλη η λειτουργικότητα ενός ΙΚΔ. Εδώ συναντά κανείς γλώσσες και τεχνολογίες (πχ. PHP, JSP, Perl), οι οποίες τρέχουν πάνω σε Application ή Web Servers και εκτελούν λειτουργίες, πραγματοποιώντας κλήσεις προς τη βάση δεδομένων για να εγγράψουν ή να ανακτήσουν πληροφορία. Σημειώνεται πως σε αυτό το επίπεδο ανήκει οποιοσδήποτε μηχανισμός που είναι

υπεύθυνος για την ενσωμάτωση και την εξυπηρέτηση αναγκών των εφαρμογών που υπάρχουν σε έναν ΙΚΔ.

Το επίπεδο επισκόπησης είναι εκείνο το οποίο βρίσκεται πιο κοντά στο χρήστη και φροντίζει για την συγκρότηση όλης της προαναφερθείσας υποδομής σε ένα τελικό αποτέλεσμα, με το οποίο αλληλεπιδρά ο χρήστης. Εδώ, κατά κύριο λόγο συναντά κανείς γλώσσες προγραμματισμού και τεχνολογίες που χρησιμοποιούνται για την κατασκευή και τη δόμηση του περιεχομένου ιστοσελίδων. Τέτοιες γλώσσες είναι η HTML, η CSS και η JavaScript.



Εικόνα 2.1 – Αρχιτεκτονική Τριών Επιπέδων

2.2 Εργαλεία για τη δημιουργία και φιλοξενία εφαρμογών

2.2.1 Δημιουργία και χρήση εφαρμογών

Όπως μπορεί να παρατηρηθεί στον Πίνακα 1.1 σχεδόν όλες οι δημοφιλέστεροι ΙΚΔ παρέχουν τα APIs τους ώστε βάσει αυτών να κατασκευαστούν εφαρμογές εσωτερικές και εξωτερικές σε αυτούς. Τα παρεχόμενα APIs κατά κύριο λόγο κάνουν χρήση του πρωτοκόλλου REST για την πραγματοποίηση κλήσεων μέσω URLs. Η τεχνολογία

αυτή υλοποιείται κυρίως με τη χρήση HTTP Requests, ενώ χρησιμοποιούνται τα πρότυπα XML και JSON για την αναπαράσταση των δεδομένων.

Η λειτουργικότητα των εφαρμογών συνήθως αναπτύσσονται σε γλώσσα JavaScript, ενώ η εμφάνιση ανατίθεται σε κλασσικά πρότυπα όπως HTML και CSS. Οι ΙΚΔ συνήθως παρέχουν προς τους προγραμματιστές ένα *JavaScript API*, μέσω του οποίου μπορούν να χρησιμοποιήσουν διάφορες ειδικές κλήσεις ή συναρτήσεις ειδικά σχεδιασμένες ώστε να προσανατολίζονται στις ιδιαιτερότητες του κοινωνικού τους γράφου. Όσον αφορά την ενσωμάτωση εφαρμογών στο εσωτερικό ενός ΙΚΔ, στις περισσότερες περιπτώσεις, αυτή γίνεται με την τεχνολογία των *iframes*, HTML ετικέτες δηλαδή που φορτώνουν σε ένα πλαίσιο μίας ιστοσελίδας το αντικείμενο ενός URL. Ταυτόχρονα, πολλές φορές απαιτείται ένας μηχανισμός αυθεντικοποίησης (authentication) για την ασφάλεια των προσωπικών πληροφοριών που προσπελούνται μέσα από μία εφαρμογή.

Παρόλο που οι τεχνολογίες που χρησιμοποιούνται ανά τους ΙΚΔ είναι σχετικά πανομοιότυπες, παρατηρεί κανείς από τις διαφορετικές λειτουργίες τους ότι τα APIs τους είναι εξίσου διαφορετικά και δε μπορούν να χρησιμοποιηθούν με κάποιο ενιαίο τρόπο, ώστε να εξασφαλίζεται η μεταφερσιμότητα και η διαλειτουργικότητα των εφαρμογών. Αυτή την πρόκληση προσπάθησε να αντιμετωπίσει η Google, ανακοινώνοντας το 2007 το πρότυπο OpenSocial.

2.2.3 OpenSocial

2.2.3.1 Επισκόπηση

Το OpenSocial είναι ένα σύνολο από APIs για τη δημιουργία κοινωνικών εφαρμογών που τρέχουν στο Διαδίκτυο. Στόχος του OpenSocial είναι να καταστήσει πιο διαθέσιμες και σε περισσότερους χρήστες εφαρμογές, παρέχοντας ένα κοινό προγραμματιστικό περιβάλλον, που μπορεί να χρησιμοποιηθεί σε πολλά διαφορετικά πλαίσια. Οι προγραμματιστές μπορούν να δημιουργήσουν εφαρμογές με βασικές γλώσσες όπως JavaScript και HTML, που λειτουργούν σε ιστοσελίδες κοινωνικής δικτύωσης όπου εφαρμόζονται τα OpenSocial APIs. Οι ιστοσελίδες αυτές, γνωστές ως OpenSocial Containers, επιτρέπουν στους προγραμματιστές να έχουν πρόσβαση

στις κοινωνικές τους πληροφορίες τους. Σαν αντάλλαγμα, λαμβάνουν μια μεγάλη γκάμα εφαρμογών για τους χρήστες τους (OpenSocial Specification, 2007).

Το OpenSocial παρέχει ένα τρόπο για υπηρεσίες κοινωνικής δικτύωσης να εκθέτουν την κοινωνική τους πληροφορία και να τη διαδίδουν σε ολόκληρο το γράφο τους. Επιπλέον, προβλέπει τη διατήρηση των δεδομένων των εφαρμογών που «τρέχουν» στα πλαίσια της υπηρεσίας, ορίζοντας παράλληλα τους διαφορετικούς τρόπους με τους οποίους μπορεί προσπελαστούν αυτές. Τέλος, ορίζει ένα API για τον τρόπο με τον οποίο θα γίνεται ο χειρισμός των δεδομένων από εξωτερικούς χρήστες και υπηρεσίες.

Παρακάτω παρατίθενται βασικές έννοιες κοινωνικής δικτύωσης μέσα από τη σκοπιά του OpenSocial.

2.2.3.2 Άτομα

Η οντότητα Άτομο (Person) είναι η πιο σημαντική οντότητα που δομεί ένα κοινωνικό δίκτυο. Όλοι οι τρόποι αλληλεπίδρασης και ο κοινωνικός γράφος που ορίζουν οι χρήστες της υπηρεσίας έχουν σαν βασικό δομικό στοιχείο και κινητήρια δύναμη το *άτομο*, το οποίο έχει αποθηκευμένη όλη την κοινωνική πληροφορία που αφορά κάθε χρήστη ξεχωριστά. Το OpenSocial φροντίζει να εκθέτει την πληροφορία αυτή, παρέχοντας τον αντίστοιχο μηχανισμό για την υποστήριξη, διαχείριση και εξυπηρέτηση κλήσεων προς τη βάση δεδομένων της υπηρεσίας.

Επιπλέον, όπως ορίζει το OpenSocial, σε κάθε κοινωνική συναλλαγή, δρουν δύο στιγμιότυπα της οντότητας «άτομο»: ο *Viewer* (*θεατής*) και ο *Owner* (*κάτοχος*). Για παράδειγμα, αν σε μία ιστοσελίδα κοινωνικής δικτύωσης ο X επισκεφτεί το προφίλ του Y, τότε ο X επισημαίνεται ως Viewer, ενώ ο Y επισημαίνεται αντίστοιχα ως Owner. Οι δύο αυτές εκφάνσεις εξυπηρετούν τη διαφοροποίηση της λειτουργίας των εφαρμογών, ανάλογα με τα συναλλασσόμενα άτομα σε κάθε περίπτωση. Για παράδειγμα, μία εφαρμογή-gadget που είναι εγκατεστημένο στο προφίλ ενός χρήστη (Owner) μπορεί να εμφανίζει διαφορετικές πληροφορίες σε αυτόν από εκείνες που εμφανίζει σε έναν επισκέπτη στο προφίλ (Viewer).

2.2.3.3 Σχέσεις

Μία υπηρεσία κοινωνικής δικτύωσης αποκτά υπόσταση όταν υπεισέρχεται σε αυτή η έννοια των σχέσεων. Οι σχέσεις μεταξύ ατόμων είναι αυτές που απαρτίζουν τον κοινωνικό γράφο και ρυθμίζουν απόλυτα την αλληλεπίδραση μεταξύ των χρηστών. Το OpenSocial ορίζει συμβατικά τη σχέση «φίλος» (friend) μεταξύ των ατόμων και τη χρησιμοποιεί για την ομαδοποίηση των επαφών ενός χρήστη, ώστε να έχει πρόσβαση σε αυτές μέσω των εφαρμογών. Αυτό όμως δεν σημαίνει πως θεωρείται αυτονόητη η ύπαρξη σχέσεων μεταξύ δύο ή περισσότερων εμπλεκομένων σε μία κοινωνική συναλλαγή.

2.2.3.4 Δραστηριότητες

Όλες οι σύγχρονες ιστοσελίδες κοινωνικής δικτύωσης περιλαμβάνουν την καταγραφή και την εμφάνιση δραστηριοτήτων (activities), ώστε να υπάρχει ένα ιστορικό με τις ενέργειες κάθε ατόμου που προέκυψαν κατά την αλληλεπίδραση του με την υπηρεσία. Το OpenSocial εκθέτει ροές δραστηριοτήτων που αφορούν ενέργειες ενός ατόμου όπως η αλλαγή προσωπικών πληροφοριών, η σύναψη μίας νέας σχέσης, η εγκατάσταση ενός νέου gadget κλπ. Οι ροές αυτές μπορεί να αφορούν ένα συγκεκριμένο άτομο ή άτομα τα οποία έχουν συνάψει κάποια σχέση με αυτό.

Επιπλέον, το OpenSocial αναφορικά με τις Δραστηριότητες, υποστηρίζει την κατηγοριοποίηση τους βάσει κάποιου συγκεκριμένου πεδίου «περίληψης» (summary). Με αυτόν τον τρόπο, οι Δραστηριότητες οργανώνονται σε δέσμες (bundles), ώστε να είναι πιο εύκολα προσβάσιμες και καλύτερα οργανωμένες.

2.2.3.5 Διατήρηση Δεδομένων – Δεδομένα Εφαρμογών

Το OpenSocial, εκτός των άλλων, φροντίζει για την διατήρηση των συνεδριών (sessions) στις εφαρμογές που τρέχουν σε μία υπηρεσία κοινωνικής δικτύωσης. Αυτό το πραγματοποιεί με τη βοήθεια ενός μηχανισμού αποθήκευσης δεδομένων, των λεγόμενων δεδομένων εφαρμογών (Application Data ή AppData). Το OpenSocial προβλέπει λειτουργίες για την ανάκτηση, την ενημέρωση ή δημιουργία και την διαγραφή αυτών των δεδομένων.

Όσον αφορά την αποθήκευση τους, τα AppData οργανώνονται σε χάρτες. Ένας χάρτης αποτελεί την δήλωση του αποθηκευτικού χώρου που διαθέτει ένα Άτομο σε μία εφαρμογή. Σε δεύτερο επίπεδο, κάθε χάρτης αντιστοιχεί σε ζεύγη από κλειδιά και τις αντίστοιχες τιμές τους. Για παράδειγμα, έστω ότι ο χρήστης Tom χρησιμοποιεί μία

εφαρμογή όπου χρησιμοποιεί ψηφιακά κέρματα (coins). Η εφαρμογή αυτή απαιτεί την διατήρηση μίας τέτοιας πληροφορίας, οπότε κατά την έναρξη χρήσης της δημιουργείται ένας νέος χάρτης, ο οποίος αντιστοιχεί τον χρήστη Tom με τη συγκεκριμένη εφαρμογή. Τα δεδομένα που θα αντιστοιχούν στον χάρτη αυτό θα έχουν σαν κλειδί το όνομα «κέρματα» (coins) και σαν τιμή τον αριθμό των κερμάτων που έχει συγκεντρώσει ο Tom κατά τη χρήση της εφαρμογής.

2.2.3.6 Όψεις

Στα πλαίσια μίας υπηρεσίας κοινωνικής δικτύωσης, το OpenSocial αναγνωρίζει διαφορετικές όψεις (views) ή περιοχές μέσα στις οποίες λειτουργούν τα διαθέσιμα gadgets της. Οι όψεις που υποστηρίζονται είναι οι εξής:

- *Canvas*

Ένα gadget στην όψη canvas αποδίδεται σε μία διευρυμένη περιοχή, καταλαμβάνοντας το μεγαλύτερο μέρος της οθόνης του χρήστη και είναι ορατό από οποιονδήποτε επισκέπτη-θεατή (Viewer). Τέτοια gadgets συναντά κανείς κυρίως σε κεντρικές σελίδες ιστοσελίδων κοινωνικής δικτύωσης, όπου εμφανίζονται ροές φίλων κλπ.

- *Home*

Ένα gadget στην όψη home αποδίδεται σε μία «ιδιωτική» ιστοσελίδα, όπου ο Owner είναι και Viewer και συνήθως καταλαμβάνει πολύ μικρό χώρο στην οθόνη. Τέτοια όψη έχουμε, για παράδειγμα, όταν ένας χρήστης θέλει να επεξεργαστεί τις ρυθμίσεις που αφορούν το προφίλ του.

- *Preview*

Ένα gadget στην όψη preview αποδίδεται χωρίς να λαμβάνει υπ' όψιν κάποιον Owner ή Viewer. Τέτοια όψη προκύπτει όταν ένας χρήστης επισκέπτεται μία ιστοσελίδα κοινωνικής δικτύωσης στην οποία δεν έχει εγγραφεί.

- *Profile*

Ένα gadget στην όψη profile αποδίδεται μαζί με άλλα gadgets που μπορεί ένας χρήστης να έχει εγκαταστήσει στο προφίλ του. Σε αυτή την περίπτωση, είναι προσβάσιμο από όλους τους θεατές (Viewers) του προφίλ αυτού.

Φυσικά, σε μία υπηρεσία κοινωνικής δικτύωσης, δεν είναι απαραίτητο να χρησιμοποιηθούν (όλες) οι παραπάνω όψεις. Επιπλέον, ένα gadget μπορεί να αλλάζει όψεις κατά την περίοδο λειτουργίας του, πράγμα που σημαίνει ότι ταυτόχρονα θα αλλάζει Owners και Viewers ή ακόμα και τις πληροφορίες που διατίθενται σε αυτό μέσω του URL της σελίδας που βρίσκονται.

2.2.3.7 Γεγονότα Κύκλου Ζωής

Ολοκληρώνοντας τις προδιαγραφές των εφαρμογών που τρέχουν σε έναν ΙΚΔ, το OpenSocial παρέχει την λειτουργικότητα για τον ορισμό διαφόρων γεγονότων που αφορούν τον κύκλο ζωής των gadgets που είναι εγκατεστημένα σε μία υπηρεσία (Life-Cycle Events), όπως η εγκατάσταση, η απεγκατάσταση, η αναφορά ενός gadget ως ανεπιθύμητου κλπ. Πιο συγκεκριμένα, επιτρέπει στους προγραμματιστές να καθορίζουν διευθύνσεις URL όπου ο Container διατηρεί πληροφορίες για γεγονότα που αφορούν συγκεκριμένες οντότητες του Opensocial, σε μορφή αναγνωριστικών (id's). Έτσι, με ένα ping προς το αντίστοιχο URL, επιστρέφεται μία λίστα από γεγονότα που αφορούν έναν χρήστη με ένα συγκεκριμένο ID. Οι διαθέσιμες διευθύνσεις URL ορίζονται στο XML αρχείο ενός gadget και συνήθως ακολουθούν κάποιο πρότυπο που ορίζει η ίδια η υπηρεσία.

2.2.3.8 Εφαρμογές του OpenSocial

Το πρότυπο OpenSocial αναγνωρίστηκε και υιοθετήθηκε από αρκετά διάσημες υπηρεσίες, όταν εμφανίστηκε. Οι πιο γνωστές από αυτές είναι:

- MySpace
- Friendster
- Hi5
- Orkut
- Netlog
- Ning
- Yahoo!

Παρόλο που δεν εφαρμόζεται στους πιο διάσημους ΙΚΔ αυτή τη στιγμή (Facebook, Twitter), το OpenSocial παραμένει ίσως το πιο γνωστό σημείο αναφοράς για τη δημιουργία ενός ΙΚΔ και ιδιαίτερα για την φιλοξενία εφαρμογών από αυτή. Υπάρχει μάλιστα και η αντίστοιχη προσπάθεια για υλοποίηση του με το όνομα Shindig, η οποία έχει συγκεντρώσει αρκετό προγραμματιστικό ενδιαφέρον και συμβαδίζει αλληλένδετα με το εν λόγω πρότυπο.

2.2.4 Apache Shindig

Το Shindig είναι ένα project ανοικτού κώδικα που ξεκίνησε το Δεκέμβριο του 2007 από τον Οργανισμό Apache σαν υλοποίηση (reference implementation) του προτύπου OpenSocial. Σε γενικές γραμμές, το Shindig περιέχει κώδικα εξυπηρετητή (server) και εξυπηρετημένου (client), ώστε να παρέχει τις παρακάτω λειτουργικότητες:

- Gadget Container σε γλώσσα JavaScript, δηλαδή το απαραίτητο υπόβαθρο για την υποστήριξη εφαρμογών σε μορφή gadgets από μία υπηρεσία. Η JavaScript φροντίζει για την ασφάλεια, την επικοινωνία, το γραφικό περιβάλλον, καθώς και για την υποστήριξη μηχανισμού εισαγωγής στοιχείων-λειτουργιών (features).
- Gadget Server, δηλαδή έναν server, ο οποίος αποτελεί μία έκδοση ανοικτού κώδικα του gmodules.com και που είναι υπεύθυνος για την απόδοση του XML κώδικα ενός gadget σε JavaScript και HTML, ώστε να μπορεί ο Container να το αφομοιώσει και να το εκτελέσει.
- OpenSocial Container JavaScript, δηλαδή ένα περιβάλλον σε JavaScript στην κορυφή του Gadget Container, το οποίο παρέχει λειτουργικότητα που προβλέπει το OpenSocial (προφίλ, φίλοι, δραστηριότητες).
- OpenSocial Gateway Server, δηλαδή μία ανοικτού κώδικα υλοποίηση για την επικοινωνία των πληροφοριών του Container με τον server, όπως επίσης και τις υλοποιήσεις των OpenSocial REST APIs, ώστε να μπορεί οποιοσδήποτε να συνδέσει το Shindig με το δικό του υπόβαθρο.

Επιπλέον, σαν βασικός στόχος του Shindig, ορίστηκε η “γλωσσική ουδετερότητα”. Γι' αυτό το λόγο, το Shindig κυκλοφορεί σε δύο εκδόσεις: σε Java σε PHP.

3. Λειτουργικές Απαιτήσεις ενός ακαδημαϊκού ιστότοπου κοινωνικής δικτύωσης με έμφαση στην εκτέλεση εφαρμογών

3.1 Ορισμός Προβλήματος

3.1.1 Περιβάλλον Λειτουργίας

Η παρούσα εργασία αφορά έναν ιστότοπο ακαδημαϊκής κοινωνικής δικτύωσης με την ονομασία «Unity», ο οποίος ενσωματώνει βασικά χαρακτηριστικά και λειτουργίες ενός ΙΚΔ, επεκτείνοντας τα με σκοπό την ενίσχυση της εκπαιδευτικής διαδικασίας και εκμάθησης, καθώς και της συνεργασίας των μελών μίας ακαδημαϊκής κοινότητας με ηλεκτρονικό τρόπο. Επιπλέον, η υπηρεσία αυτή διαχωρίζει τα άτομα-μέλη σύμφωνα με την ακαδημαϊκή τους υπόσταση, ορίζοντας παράλληλα διαφορετικές σχέσεις μεταξύ τους, πάντα σε αντιστοιχία με εκείνες που ισχύουν και στην πραγματικότητα.

3.1.1.1 Χρήστες

Αρχικά, χρήστες του ΙΚΔ έχουν οριστεί όλοι όσοι σχετίζονται με την λειτουργία ενός πανεπιστημιακού τμήματος στο οποίο απευθύνεται αυτή η εργασία. Για να εξασφαλιστεί κάτι τέτοιο, κατά τη διαδικασία εγγραφής (registration) κάθε νέου μέλους πραγματοποιείται *ταυτοποίηση* του, με τη βοήθεια της υπηρεσίας LDAP του Πανεπιστημιακού Ιδρύματος, έτσι ώστε να εξασφαλίζεται πέρα από την ταυτοποίηση και η *μοναδικότητα* του. Συνεπώς, χρήστες στην υπηρεσία είναι εκείνοι οι οποίοι έχουν καταχωρηθεί με ένα μοναδικό αναγνωριστικό (κατά προτίμηση e-mail) στο μητρώο του Ιδρύματος.

3.1.1.2 Συμμετέχοντες

Οι χρήστες, τώρα πλέον συμμετέχοντες στην υπηρεσία, αντιστοιχίζονται σε στιγμιότυπα της οντότητας “άτομο” (person). Η οντότητα αυτή έχει ως αναγνωριστικό μία ακολουθία χαρακτήρων, η οποία θα αναφέρεται στο εξής ως person ID και που εκμαιεύεται από τη διεύθυνση e-mail των χρηστών του Ιδρύματος. Επιπλέον, για να διαχωριστούν οι συμμετέχοντες ανάλογα με τον ακαδημαϊκό τους

ρόλο, ορίστηκαν τρεις γενικές κατηγορίες, οι οποίες αποτελούν διαφορετικές εκφάνσεις της οντότητας αυτής:

- **Students**, στην οποία ανήκουν προπτυχιακοί, μεταπτυχιακοί και διδακτορικοί φοιτητές
- **Teaching Staff**, στην οποία ανήκουν μέλη ΔΕΠ και εξωτερικοί ή συμβασιούχοι διδάσκοντες
- **Administrative Staff**, στην οποία ανήκουν απασχολούμενοι στη Γραμματεία και σε λοιπές υπηρεσίες του Πανεπιστημίου

3.1.1.3 Σχέσεις

Οι συμμετέχοντες του ακαδημαϊκού κοινωνικού δικτύου μπορούν να συνδεθούν μεταξύ τους *ακαδημαϊκά* και *κοινωνικά*. Ως ακαδημαϊκή σχέση νοείται η σχέση που συνάπτεται μεταξύ δύο ατόμων εξαιτίας των ακαδημαϊκών τους ρόλων, ενώ, αντίθετα, κοινωνική είναι η σχέση που συνάπτεται εξαιτίας κάποιου κοινωνικού αισθήματος που νιώθει ο ένας για τον άλλο, όπως η φιλία ή ο σεβασμός. Η μόνη κοινωνική σχέση που ορίζεται στα πλαίσια του Unity είναι:

- Η σχέση *Fellow* (*συντρόφου*) μεταξύ δύο συμμετεχόντων, η οποία συνάπτεται ανεξάρτητα από τους ακαδημαϊκούς τους ρόλους και είναι *αμφίδρομη*.

Επιπρόσθετα, ορίζονται δύο ακαδημαϊκές σχέσεις, τέτοιες ώστε να εξυπηρετούν κυρίως τα συμφέροντα των χρηστών που είναι φοιτητές. Αυτές είναι:

- Η σχέση *Teacher* (*καθηγητής*), η οποία είναι *μονόδρομη*. Μία τέτοια σχέση συνεπάγεται ότι ένας συμμετέχων που είναι φοιτητής «ακολουθεί» έναν καθηγητή.
- Η σχέση *Facilitator* (*μεσολαβητής*), η οποία είναι επίσης *μονόδρομη*. Μία τέτοια σχέση συνεπάγεται ότι ένας συμμετέχων που είναι φοιτητής «ακολουθεί» ένα μέλος του προσωπικού της ακαδημαϊκής κοινότητας.

Στον παρακάτω πίνακα φαίνονται πιο αναλυτικά οι σχέσεις, το είδος και ο χαρακτηρισμός τους. Στις στήλες “Υποκείμενο” και “Αντικείμενο” ορίζεται αφενός η ομάδα στην οποία ανήκει ο αιτών (requestor) τη σχέση και αφετέρου η ομάδα στην

οποία ανήκει ο παραλήπτης της αίτησης (receiver), αντίστοιχα. Με αστερίσκο δηλώνεται ότι το άτομο δεν χρειάζεται να ανήκει σε συγκεκριμένη ομάδα για να συμμετάσχει στη σύναψη της συγκεκριμένης σχέσης.

Σχέση	Είδος	Υποκείμενο	Αντικείμενο	Χαρακτηρισμός
Fellow	αμφίδρομη	*	*	κοινωνική
Teacher	μονόδρομη	Students	Teaching staff	ακαδημαϊκή
Facilitator	μονόδρομη	Students	Administrative staff	ακαδημαϊκή

Πίνακας 3.1 – Είδη σχέσεων

3.1.1.4 Βασικές Λειτουργίες

Γενικά, οι λειτουργίες που υποστηρίζει ο ΙΚΔ είναι οι εξής:

- Εγγραφή Χρήστη
- Επεξεργασία στοιχείων
- Αναζήτηση χρηστών
- Σύναψη σχέσεων με άλλους χρήστες
- Δημιουργία εικονικών ομάδων (groups) και ομάδων μαθημάτων
- Αναζήτηση εικονικών ομάδων και ομάδων μαθημάτων
- Εγγραφή/Απεγγραφή σε εικονικές ομάδες και ομάδες μαθημάτων
- Δημοσίευση μηνυμάτων σε ομάδες χρηστών και σε ομάδες

3.1.2 Στόχος Εργασίας

Στόχος αυτής της εργασίας είναι η περιγραφή της επέκτασης της υπηρεσίας που περιγράφηκε παραπάνω, ώστε να υποστηρίζει την φιλοξενία εφαρμογών, στις οποίες αντικατοπτρίζονται οι συναλλαγές και οι υπηρεσίες που συναντώνται στο εσωτερικό

μίας ακαδημαϊκής κοινότητας. Οι εν λόγω εφαρμογές θα πρέπει να δρουν πάνω στο γράφο του κοινωνικού δικτύου, στηριζόμενες σε πληροφορία που λαμβάνουν από αυτόν και αποθηκεύοντας σε αυτόν τη δική τους, όταν κάτι τέτοιο κρίνεται σκόπιμο.

Η πρόσβαση σε αυτές τις εφαρμογές θα πρέπει να γίνεται με τρόπο που να σέβεται τις ακαδημαϊκές συσχετίσεις των συμμετεχόντων, προσομοιάζοντας όσο το δυνατόν καλύτερα τις υπηρεσίες στα πλαίσια των οποίων οι συμμετέχοντες αυτοί αλληλεπιδρούν στην πραγματικότητα. Ταυτόχρονα όμως, οι εφαρμογές αυτές δεν πρέπει να επισκιάζουν την δυνατότητα κοινωνικής δικτύωσης, όπως αυτή παρέχεται από τον υπάρχοντα ΙΚΔ, αλλά να προσφέρονται παράλληλα με αυτή, για την επίτευξη μίας καλύτερης ποιότητας υπηρεσίας.

Στα πλαίσια της επίτευξης του παραπάνω στόχου, θα αναλυθούν τεχνολογίες και μεθοδολογίες που μπορούν να χρησιμοποιηθούν για την εκπόνηση της επέκτασης ενός ιστότοπου ακαδημαϊκής κοινωνικής δικτύωσης. Στο τέλος θα παρουσιαστούν δύο πιλοτικές εφαρμογές, οι οποίες αντιστοιχούν σε υπηρεσίες που συναντά κανείς στο εσωτερικό μίας ακαδημαϊκής κοινότητας.

3.2 Λειτουργικές Απαιτήσεις

3.2.1 Παρεχόμενες υπηρεσίες και αξιολόγηση της αποτελεσματικότητας τους στο εσωτερικό μίας Ακαδημαϊκής Κοινότητας

Υπηρεσίες σε μία ακαδημαϊκή κοινότητα είναι οι ακαδημαϊκές συναλλαγές, στις οποίες συμμετέχουν τα μέλη της. Θεωρείται στα πλαίσια της εργασίας αυτής, πως τελικός αποδέκτης αυτών των συναλλαγών είναι οι φοιτητές (προπτυχιακοί ή μεταπτυχιακοί), οι οποίοι συνήθως αιτούνται τη χορήγηση ενός πόρου και οι συμμετέχοντες των υπολοίπων ακαδημαϊκών ρόλων αναλαμβάνουν να την αποδεκτούν και να την επικυρώσουν. Ενδεικτικά, κάποιες τέτοιες υπηρεσίες είναι:

- Υπηρεσία για την αίτηση χορήγησης αναλυτικής βαθμολογίας
- Υπηρεσία για την γνωστοποίηση αποτελεσμάτων εξετάσεων
- Υπηρεσία για την αίτηση χορήγησης υποτροφίας
- Υπηρεσία ενημέρωσης κατάστασης δανεισμού από την βιβλιοθήκη

- Υπηρεσία ενημέρωσης για τη σύγκληση Γενικών Συνελεύσεων φοιτητών
- Υπηρεσία για την δήλωση μαθημάτων επιλογής

Εάν εξετάσει κάποιος υπηρεσίες όπως οι παραπάνω που παρέχονται στα μέλη μίας ακαδημαϊκής κοινότητας, διαπιστώνει κάποιες ομοιότητες ως προς τις διαδικασίες που υποστηρίζουν. Πιο συγκεκριμένα, οι διαδικασίες και η αλληλουχία αυτών που συναντώνται κατά κύριο λόγο είναι οι παρακάτω:

1. Ο φοιτητής επισκέπτεται τον πάροχο υπηρεσιών *προς φοιτητές*, ο οποίος μπορεί να είναι κάποιος καθηγητής ή κάποιο μέλος του προσωπικού του Πανεπιστημίου.
2. Ο πάροχος τον υποδέχεται και κατόπιν αυτού ο φοιτητής αιτείται την χορήγηση κάποιου πόρου ή την τροποποίηση κάποιων υπάρχοντων πόρων.
3. Ο πάροχος αποδέχεται ή απορρίπτει την παραπάνω ενέργεια.
4. Ύστερα από κάποιο χρονικό διάστημα, ο φοιτητής επισκέπτεται ξανά τον πάροχο ή ένα χώρο ανακοινώσεων, ενημερώνεται για την αποδοχή/απόρριψη της ενέργειας του και δρα αντίστοιχα.
5. Συμπληρωματικά, εάν ο πάροχος είναι καθηγητής, κάποιες φορές απαιτείται η επικύρωση παροχής ενός πόρου από κάποιο μέλος του προσωπικού.
6. Η παροχή της υπηρεσίας ολοκληρώνεται και ο πάροχος ετοιμάζεται για την εξυπηρέτηση νέων αιτήσεων.

Κάποια προβλήματα κατά την τήρηση της παραπάνω αλληλουχίας διαδικασιών είναι γνωστά από την καθημερινή ζωή. Δεν είναι λίγες οι φορές, όπου παρουσιάζονται κωλύματα κατά τη διεκπεραίωση διαδικασιών, ειδικά μάλιστα όταν αυτές πρέπει να πραγματοποιηθούν για παραπάνω από έναν συμμετέχοντες. Για παράδειγμα, όταν ένας φοιτητής αιτείται την χορήγηση μίας βεβαίωσης σπουδών από τη γραμματεία του Τμήματος όπου φοιτά, πολλές φορές χρειάζεται να αναμένει για ένα απροσδιόριστο χρονικό διάστημα την ολοκλήρωση της ίδιας διαδικασίας και για άλλους φοιτητές που προηγήθηκαν πριν εξυπηρετηθεί. Παράλληλα, πολλές φορές απαιτείται η φυσική παρουσία του φοιτητή στην υπηρεσία, ώστε να ενημερωθεί για την πρόοδο των αιτημάτων του, γεγονός που καταδεικνύει την έλλειψη επικοινωνίας μεταξύ των συναλλασσόμενων της υπηρεσίας.

Επιπλέον, σε υπηρεσίες που εμπλέκονται παραπάνω από δύο συναλλασσόμενοι συναντά κανείς προβλήματα στον συντονισμό διαδικασιών. Είτε εξαιτίας ανθρώπινων λαθών στην επικοινωνία, είτε εξαιτίας πολυπλοκότητας των ίδιων των διαδικασιών, η συνεχής ενημέρωση των συναλλασσομένων αποτυγχάνει και ο χρόνος διεκπεραίωσης μίας υπηρεσίας αυξάνεται σημαντικά. Συνεπώς, πολλές φορές απαιτείται ο καλύτερος συντονισμός των συμμετεχόντων σε μία συναλλαγή, παράμετρος η οποία καθορίζεται από την *αυτοματοποίηση* των επιμέρους της διαδικασιών.

3.2.2 Συνεισφορά κοινωνικής δικτύωσης στην αποτελεσματικότητα παροχής υπηρεσιών

Για την αποτελεσματικότητα των παρεχόμενων υπηρεσιών, θα μπορούσε κανείς να εξετάσει τους συναλλασσόμενους και τις ακαδημαϊκές τους συσχετίσεις μέσα από το πρίσμα της κοινωνικής δικτύωσης. Μία πρώτη παρατήρηση είναι πως κάθε υπηρεσία δεν αποτελεί ένα ενιαίο σύνολο διαδικασιών που διεκπεραιώνει ένας συμμετέχων, αλλά δέσμες διαδικασιών, κάθε μία από τις οποίες εκτελείται από διαφορετικό συμμετέχοντα που παίρνει μέρος στην συναλλαγή. Αυτό πρακτικά σημαίνει πως μία υπηρεσία έχει πολλές «όψεις», μία για κάθε συμμετέχοντα που την προσπελαύνει. Τέτοιου είδους διάκριση των συμμετεχόντων και των αντίστοιχων διαδικασιών τους μπορεί σε επόμενο επίπεδο να εξυπηρετήσει τον καλύτερο συντονισμό τους.

Επιπλέον, σε μία υπηρεσία που συναντάται σε μία ακαδημαϊκή κοινότητα είναι εμφανές το στοιχείο της *συμμετοχικότητας*. Οι φοιτητές *επιλέγουν* την υπηρεσία και, ύστερα, το άτομο που θα τους εξυπηρετήσει. Με άλλα λόγια, τα αιτήματα που πραγματοποιούνται στα πλαίσια υπηρεσιών «στοχεύουν» σε συγκεκριμένα άτομα και όχι προς όλους, καθώς κάτι τέτοιο θα πρόσθετε ανεπιθύμητη πολυπλοκότητα στη διαδικασία διεκπεραίωσης των υπηρεσιών. Αυτού του είδους η στόχευση είναι η έκφραση των ακαδημαϊκών σχέσεων ή συσχετίσεων μεταξύ των μελών μίας ακαδημαϊκής κοινότητας, παράγοντες οι οποίοι ορίζονται πλήρως στα πλαίσια της ακαδημαϊκής κοινωνικής δικτύωσης.

Τέλος, μέσω ενός ΙΚΔ μπορεί να δοθεί η δυνατότητα *αυτοματοποίησης* των υπηρεσιών. Εφόσον οποιαδήποτε διαδικασία μπορεί να πραγματοποιηθεί ηλεκτρονικά, μπορεί να προγραμματιστεί ώστε να βρίσκεται σε χρονική συνάφεια με τις υπόλοιπες. Κάτι τέτοιο πρακτικά σημαίνει *ενορχήστρωση* των διαδικασιών,

γεγονός που ελαττώνει την πολυπλοκότητα τους, καθώς και τις φυσικές τους απαιτήσεις από τους χρήστες. Για παράδειγμα, ο φοιτητής που έχει αιτηθεί τη χορήγηση μίας βεβαίωσης σπουδών από τη γραμματεία δεν θα χρειαστεί να επισκέπτεται αυτή ή ένα χώρο ανακοινώσεων τακτικά για να ενημερωθεί για την πορεία του αιτήματος του. Η όλη ενημέρωση θα πραγματοποιείται στην οθόνη του υπολογιστή του έγκαιρα.

3.2.3 Σενάριο Χρήσης

Ως παράδειγμα για τα ανωτέρω παρουσιάζεται ως σενάριο χρήσης η ολοκλήρωση μίας αίτησης για τη χορήγηση υποτροφίας προς έναν φοιτητή στον ακαδημαϊκό ΙΚΔ Unity. Νοούνται στα πλαίσια του σεναρίου οι παρακάτω συναλλασσόμενοι:

- Ένα μέλος προσωπικού (γραμματέας) που επιβλέπει τη σύνταξη της αίτησης και συγκεντρώνει τα απαραίτητα δικαιολογητικά,
- ένας φοιτητής που επιμελείται της συμπλήρωσης των προσωπικών στοιχείων που τον αφορούν και αναλαμβάνει την προσκόμιση μίας συστατικής επιστολής από κάποιον καθηγητή και, τέλος,
- ένας καθηγητής, ο οποίος δέχεται/απορρίπτει το αίτημα του φοιτητή για τη χορήγηση συστατικής επιστολής

3.2.3.1 Φοιτητής

Οι ενέργειες του φοιτητή, καθώς χρησιμοποιεί την εφαρμογή είναι οι εξής:

1. Ο φοιτητής αποκτά πρόσβαση στην εφαρμογή, εξαιτίας του ακαδημαϊκού του ρόλου.
2. Μέσω της εφαρμογής, ζητά από τον γραμματέα άδεια για την συμπλήρωση της παραπάνω αίτησης.
3. Εφόσον ο γραμματέας δεκτεί, ο φοιτητής αρχίζει την συγκέντρωση των στοιχείων του, αποθηκεύοντας τα σε προσωπικό του χώρο.
4. Παράλληλα, αποστέλλει ένα αίτημα για χορήγηση συστατικής επιστολής προς έναν καθηγητή.

5. Λαμβάνει την απάντηση και, αν είναι θετική, αποστέλλει την ολοκληρωμένη φόρμα με τα στοιχεία του και την επιβεβαίωση για χορήγηση συστατικής επιστολής στον γραμματέα.
6. Αν η απάντηση του καθηγητή είναι αρνητική, επιλέγει άλλο καθηγητή, μέχρι να πραγματοποιηθεί η ενέργεια νούμερο 5.
7. Αναμένει επιβεβαίωση από το γραμματέα. Αν τελικά η αίτηση του απορριφθεί, επεξεργάζεται τα στοιχεία του μέχρι να γίνει δεκτή.

3.2.3.2 Καθηγητής

Οι αντίστοιχες ενέργειες του καθηγητή είναι οι εξής:

1. Ο καθηγητής αποκτά πρόσβαση στην εφαρμογή, εξαιτίας του ακαδημαϊκού του ρόλου.
2. Από το εσωτερικό της εφαρμογής, ενημερώνεται για τυχόν νέα αιτήματα για τη χορήγηση συστατικής επιστολής.
3. Εφόσον προκύψει κάποιο αίτημα, αποκτά τη δυνατότητα να το αποδεκτεί ή να το απορρίψει

3.2.3.3 Γραμματέας

Οι αντίστοιχες ενέργειες του γραμματέα είναι οι εξής:

1. Ο γραμματέας αποκτά πρόσβαση στην εφαρμογή, εξαιτίας του ακαδημαϊκού του ρόλου.
2. Μέσω της εφαρμογής, ενημερώνεται για τυχόν νέα αιτήματα για άδειες συμπλήρωσης αιτήσεων.
3. Στα αιτήματα που προκύπτουν, εξετάζει τα στοιχεία των αιτούντων και τα δέχεται/απορρίπτει αντίστοιχα.
4. Λαμβάνει φόρμες προσωπικών στοιχείων και επιβεβαιώσεις χορήγησης συστατικών επιστολών.
5. Ελέγχει την ορθότητα των παραπάνω, εγκρίνει την σύνταξη αίτησης για χορήγηση υποτροφίας και την αναλαμβάνει ο ίδιος εκτός εφαρμογής.

6. Ενημερώνει τον φοιτητή για τις ενέργειες του.

3.2.4 Εκμείευση Απαιτήσεων

Η παραπάνω εφαρμογή είναι ένα απλό δείγμα που υποδεικνύει την πρόσθετη λειτουργικότητα που θα μπορούσε να ενσωματωθεί στον ΙΚΔ. Είναι φανερό πως η υλοποίηση κάθε εφαρμογής τέτοιου είδους κινείται σε τέσσερις βασικούς άξονες: τους *χρήστες*, τις *σχέσεις* μεταξύ τους, τα *δεδομένα* που αποθηκεύει η εφαρμογή και, τέλος, την *ενσωμάτωση* της στον ΙΚΔ.

3.2.4.1 Χρήστες

Το γεγονός ότι οι εφαρμογές αποτελούν υλοποίηση πανεπιστημιακών υπηρεσιών επιτάσσει το διαχωρισμό των συναλλασσόμενων ανάλογα με τον ακαδημαϊκό τους ρόλο. Κάθε συμμετέχων, δηλαδή, αποκτά πρόσβαση αυστηρά σε μία συγκεκριμένη όψη κάποιας εφαρμογής και όχι στις υπόλοιπες. Συνεπώς:

- Είναι εύλογος ο διαχωρισμός μίας πανεπιστημιακής υπηρεσίας-εφαρμογής σε *υπο-εφαρμογές*, τόσες όσοι είναι και οι συναλλασσόμενοι σε αυτή.
- Πρέπει να ορίζονται για κάθε υπο-εφαρμογή αντίστοιχα *δικαιώματα πρόσβασης*, ώστε να μπορεί να προσπελαστεί από συμμετέχοντες συγκεκριμένων ακαδημαϊκών ρόλων, εάν όχι όλων.

3.2.4.2 Σχέσεις

Για να έρθουν δύο συμμετέχοντες σε συναλλαγή μεταξύ τους μέσω μίας εφαρμογής, είναι φυσικό πως πρέπει να υπάρχει η αντίστοιχη σχέση μεταξύ τους, όπως ακριβώς συμβαίνει και στους περισσότερους ΙΚΔ που φιλοξενούν εφαρμογές. Η ύπαρξη σχέσεων είναι αναγκαία, καθώς μέσω αυτών εκφράζονται οι *ακαδημαϊκές συσχετίσεις* μεταξύ των συμμετεχόντων. Επιπλέον, οι ίδιοι επιλέγουν εκείνους με τους οποίους θα έρθουν σε συναλλαγή, γεγονός που ενισχύει το στοιχείο της *συμμετοχικότητας* των μελών της ακαδημαϊκής κοινότητας στις ακαδημαϊκές διαδικασίες.

3.2.4.3 Δεδομένα

Στο παραπάνω σενάριο έγινε φανερή η ανάγκη για την αποθήκευση δεδομένων που προκύπτουν κατά τη χρήση μίας εφαρμογής. Για παράδειγμα, ένας γραμματέας θα χρειαστεί να αποθηκεύει κάπου τα αιτήματα φοιτητών που προκύπτουν ή ένας καθηγητής θα χρειαστεί να αποθηκεύει με κάποιο τρόπο τη βαθμολογία των φοιτητών πριν την δημοσιοποιήσει. Κρίνεται, λοιπόν, απαραίτητη η ύπαρξη ενός μηχανισμού που θα αποθηκεύει και θα ανακτά δεδομένα *ενός συγκεκριμένου συμμετέχοντος* για *μία συγκεκριμένη εφαρμογή*, ώστε να είναι εφικτή η χρήση πολλαπλών εφαρμογών από πολλαπλούς συμμετέχοντες.

3.2.4.4 Εγκατάσταση και χρήση εφαρμογών

Μία εφαρμογή δεν πρέπει να λειτουργεί σαν ξεχωριστό κομμάτι του ΙΚΔ, αλλά σαν δομικό στοιχείο που δρα στο εσωτερικό του. Κάθε συμμετέχων πρέπει να έχει πρόσβαση στις διαθέσιμες εφαρμογές συγκεντρωτικά, να επιλέγει εκείνες που θέλει να χρησιμοποιήσει και, τέλος, να μπορεί να τις προσπελαύνει με διαφανή τρόπο μέσα από τον ΙΚΔ. Αυτού του είδους η εγκατάσταση εξυπηρετεί από τη μία πλευρά την σημαντική διευκόλυνση των συμμετεχόντων και από την άλλη τη διάνθιση της υπηρεσίας με πρόσθετη λειτουργικότητα, η οποία δεν αποπροσανατολίζει τους συμμετέχοντες από τον κοινωνικό χαρακτήρα της, αλλά τους δίνει κίνητρο για να πλοηγηθούν σε αυτή, να εξερευνήσουν και να εκμεταλλευτούν τα σημαντικά οφέλη που τους παρέχει.

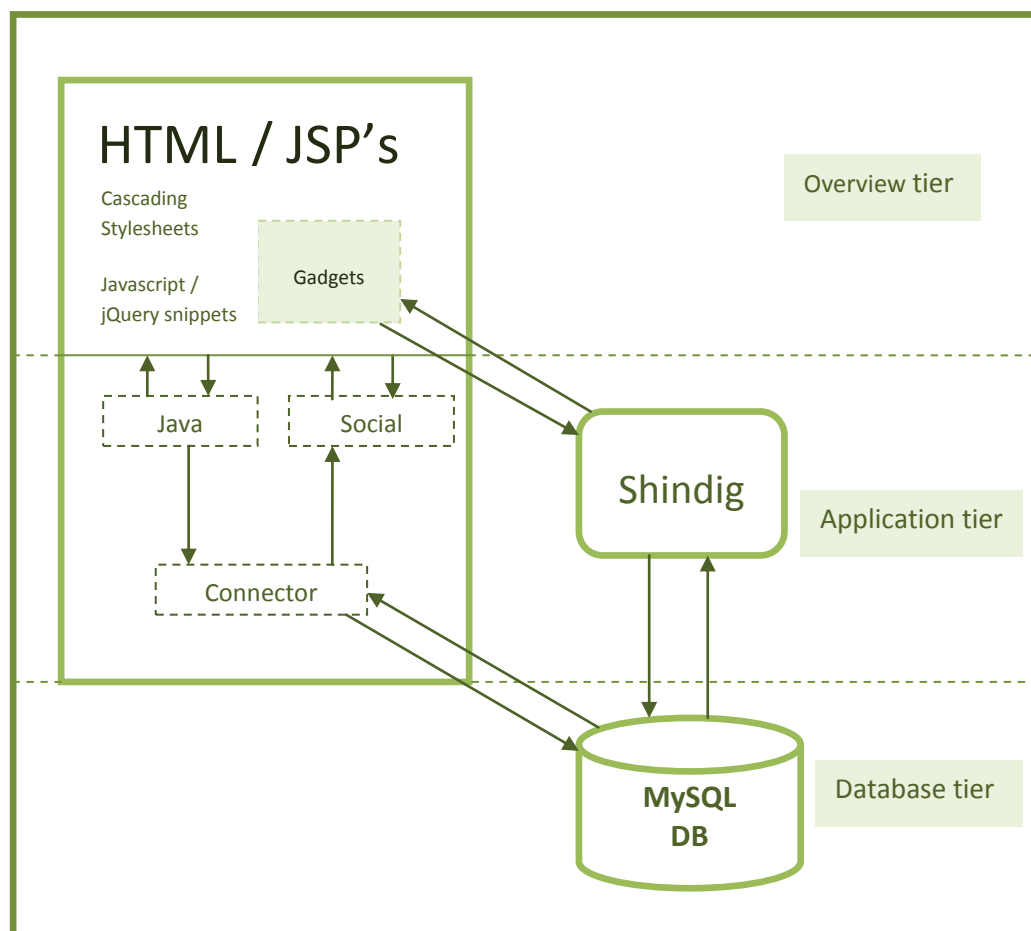
Επιπλέον, για να επισφραγιστεί η επιτυχής ενσωμάτωση των εφαρμογών, θα πρέπει οι ίδιες όχι μόνο να λαμβάνουν, αλλά και να εκπέμπουν πληροφορία στον κοινωνικό γράφο της υπηρεσίας. Αυτό σημαίνει πως οποιαδήποτε ακαδημαϊκή συναλλαγή πρέπει να εξωτερικεύει την αντίστοιχη *δραστηριότητα* των συναλλασσόμενων. Με αυτό τον τρόπο, διατηρείται ένα ιστορικό των ακαδημαϊκών ζυμώσεων των συμμετεχόντων προς ενημέρωση των υπολοίπων, ενώ ταυτόχρονα οι εφαρμογές προωθούνται από κόμβο σε κόμβο στον κοινωνικό γράφο.

4. Υλοποίηση

4.1 Περιβάλλον Υλοποίησης

4.1.1 Επίπεδα

Η υλοποίηση του Unity ακολουθεί την αρχιτεκτονική των τριών επιπέδων, όπως αυτή ορίστηκε στο δεύτερο μέρος της παρούσας εργασίας. Στην Εικόνα 3.1 παρουσιάζεται σχηματικά η αρχιτεκτονική του Unity, αναλυμένη στα τρία διαφορετικά επίπεδα: επίπεδο επισκόπησης (overview tier), επίπεδο εφαρμογής (application tier) και επίπεδο βάσης δεδομένων (database tier).



Εικόνα 3.1 – Σχηματική απεικόνιση της αρχιτεκτονικής του Unity

- *Επίπεδο Επισκόπησης*

Το Επίπεδο Επισκόπησης αφορά το μέρος του ΙΚΔ που βρίσκεται στο έμπροσθεν (front-end) της υπηρεσίας και εμφανίζεται στον περιηγητή ιστού (browser) του

χρήστη σε μορφή ιστοσελίδων. Κύριο μέλημα σε αυτό το επίπεδο είναι η φροντίδα για το γραφικό περιβάλλον της υπηρεσίας, δηλαδή ο τρόπος με τον οποίο ο χρήστης θα αλληλεπιδρά με αυτή. Τέλος, το Επίπεδο Επισκόπησης αναλαμβάνει την διατήρηση των *συνεδριών (sessions)* κατά την πλοήγηση ενός χρήστη στον ΙΚΔ, κρατώντας κάθε φορά διαθέσιμα τα προσωπικά του στοιχεία, ώστε να μην χρειάζεται να επικοινωνεί κάθε φορά με τη βάση δεδομένων.

Η γλώσσα που χρησιμοποιήθηκε κατά κύριο λόγο σε αυτό το επίπεδο είναι η HTML (έκδοση 5), η οποία μορφοποιήθηκε με CSS (έκδοση 3) για τη δόμηση του περιεχομένου των ιστοσελίδων. Για την αλληλεπίδραση του χρήστη με αυτές, χρησιμοποιήθηκε η γλώσσα JavaScript, όπως και jQuery που αποτελεί μία βιβλιοθήκη της. Σε JavaScript υλοποιήθηκαν και τα Servlets, μέσω των οποίων επικοινωνεί το παρόν επίπεδο με το επίπεδο εφαρμογής.

- *Επίπεδο Εφαρμογής*

Το Επίπεδο Εφαρμογής αποτελεί το στρώμα της αρχιτεκτονικής στο οποίο υλοποιούνται όλες οι λειτουργίες του ΙΚΔ, οι οποίες δε νοούνται χωρίς την απαραίτητη κοινωνική πληροφορία. Συνεπώς, το επίπεδο αυτό υποστηρίζει, δρομολογεί και διαχειρίζεται κλήσεις από και προς την βάση δεδομένων. Οι κλήσεις αυτές μπορούν να διακριθούν σε κλήσεις *ανάγνωσης* και κλήσεις *εγγραφής*:

- Οι κλήσεις ανάγνωσης επιστέφουν δεδομένα από τη βάση, τα οποία χειρίζονται ειδικοί μηχανισμοί και τα αποτελέσματα τα αποστέλλουν στο Επίπεδο Επισκόπησης, για να προβληθούν στο χρήστη.
- Οι κλήσεις εγγραφής φροντίζουν για την ενημέρωση της βάσης δεδομένων που βρίσκεται στο Τρίτο Επίπεδο, εγκαθιδρύοντας την αντίστοιχη σύνδεση κάθε φορά. Ειδική μνεία λαμβάνεται για περιπτώσεις όταν η εγγραφή δεν είναι επιτυχής.

Στο επίπεδο αυτό ανήκουν όλες οι Java κλάσεις που έχουν υλοποιηθεί για την απευθείας σύνδεση με τη βάση δεδομένων. Επιπρόσθετα, στο επίπεδο αυτό αφενός εκτελείται ο κώδικας που είναι προορισμένος για τον server από τα Java Server Pages (JSPs) και αφετέρου ενεργοποιούνται και εκτελούνται τα Java Servlets. Οι τεχνολογίες αυτές είναι υπεύθυνες για την διαχείριση περιεχομένου του ΙΚΔ που θα εμφανιστεί στο χρήστη και την επικοινωνία με τις συναρτήσεις που συνδέονται

απευθείας με τη βάση δεδομένων, αντίστοιχα. Ακόμα, τα Social Objects είναι ένα άλλο δομικό στοιχείο αυτού του επιπέδου τα οποία έχουν ως στόχο την οργάνωση της πληροφορίας από την βάση δεδομένων με ομοιόμορφο τρόπο. Δηλαδή, μετατρέπουν τα δεδομένα των κυριότερων «οντοτήτων» της σχεσιακής βάσης δεδομένων σε κοινωνικά αντικείμενα τα οποία μπορούν να διαχειριστούν με μεγάλη ευκολία.

Τέλος, το Επίπεδο Εφαρμογής είναι υπεύθυνο για την φιλοξενία και την απόδοση εφαρμογών (applications) σε μορφή gadgets που τρέχουν στην ΙΚΔ, κατά τρόπο που ορίζει το πρότυπο του OpenSocial. Συγκεκριμένα, η εν λόγω λειτουργικότητα παρέχεται πλήρως από την τεχνολογία Apache Shindig που αποτελεί υλοποίηση του προτύπου αυτού και που ενσωματώθηκε πλήρως στις απαιτήσεις της υπηρεσίας. Η αρχιτεκτονική και η λειτουργία του Shindig αναλύεται εκτενέστερα σε παρακάτω ενότητα.

- *Επίπεδο Βάσης Δεδομένων*

Το Επίπεδο Βάσης Δεδομένων αφορά φυσικά τη βάση δεδομένων της υπηρεσίας, η οποία αποτελεί την αποθήκη δεδομένων της. Στο σχήμα της αναλύονται πλήρως όλες οι οντότητες που απαρτίζουν τον κοινωνικό γράφο, οι σχέσεις που έχουν μεταξύ τους, όπως επίσης και διάφορες, βοηθητικές οντότητες που βοηθούν στο στήσιμο και την υποστήριξη των υπηρεσιών της ΙΚΔ.

Όσον αφορά την τεχνολογία, επιλέχθηκε η χρήση μίας MySQL βάσης δεδομένων. Η επιλογή αυτή έγινε κυρίως γιατί η MySQL :

- είναι ελεύθερο λογισμικό, το οποίο διατίθεται δωρεάν
- παρέχει υψηλή απόδοση ακόμα και για μεγάλο όγκο δεδομένων
- είναι εύχρηστη λειτουργικά και διαχειριστικά
- υποστηρίζεται από μεγάλο μερίδιο της προγραμματιστικής κοινότητας
- χρησιμοποιείται ευρύτατα σε συνδυασμό με το Apache Shindig

4.1.2 Αρχιτεκτονική Shindig

Όπως προαναφέρθηκε, για την υποστήριξη της λειτουργικότητας των εφαρμογών, χρησιμοποιείται αποκλειστικά το Apache Shindig. Ακολουθώντας τις επιταγές του OpenSocial, οι λειτουργίες του περιστρέφονται γύρω από τρία βασικά στοιχεία ενός κοινωνικού δικτύου: τα Άτομα (People), τις Δραστηριότητες (Activities) και τα Δεδομένα Εφαρμογών (Application Data ή AppData). Πιο συγκεκριμένα:

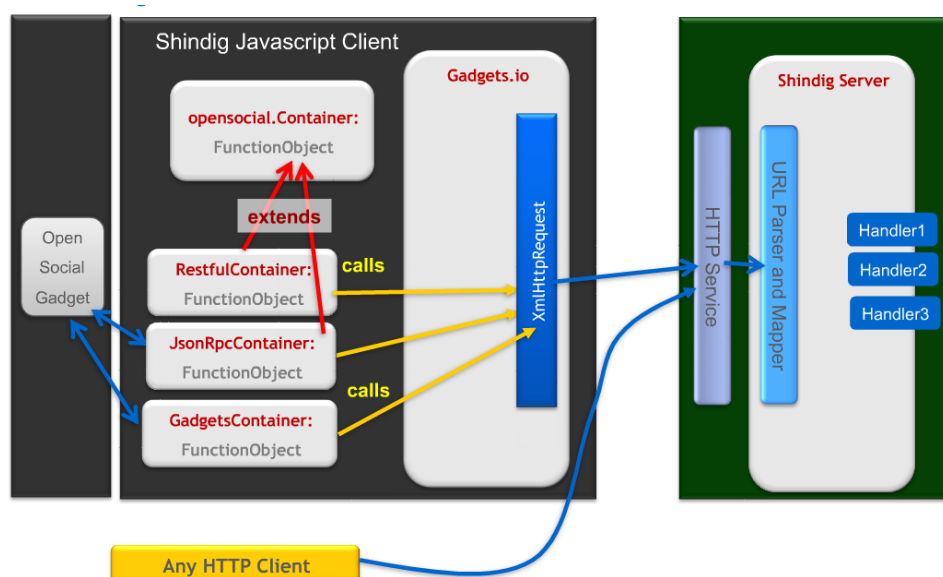
- Τα Άτομα αποτελούν την κινητήρια δύναμη της κοινωνικής δικτύωσης. Το Shindig, ακολουθώντας το OpenSocial, υλοποιεί μόνο ένα γενικό πρότυπο της οντότητας «Άτομο», το οποίο περιλαμβάνει αμιγώς κοινωνική πληροφορία. Επιπλέον, παρέχει την απαραίτητη υποδομή για διενέργεια κλήσεων για την εξαγωγή της πληροφορίας αυτής, καθώς και τη χρήση της στα πλαίσια εφαρμογών.
- Οι Δραστηριότητες είναι η επίσημη καταγραφή των κοινωνικών αλληλεπιδράσεων μεταξύ των χρηστών. Το Shindig ορίζει πάλι ένα γενικό πρότυπο Δραστηριοτήτων, το οποίο περιλαμβάνει πληροφορία για το *υποκείμενο* μίας συγκεκριμένης ενέργειας στα πλαίσια του ΙΚΔ, καθώς και για την ίδια την ενέργεια. Επιπλέον, κι εδώ παρέχεται η απαραίτητη υποδομή για την εξαγωγή και χρήση αυτής της πληροφορίας από εφαρμογές, όπως επίσης και για τη δημιουργία Δραστηριοτήτων μέσα από εφαρμογές.
- Τα Δεδομένα Εφαρμογών είναι τα δεδομένα εκείνα που προκύπτουν κατά τη χρήση εφαρμογών από τους χρήστες. Η ταξινόμηση τους γίνεται σε χάρτες (Application Data Maps) για κάθε εφαρμογή ανά χρήστη, οι οποίοι σε δεύτερο επίπεδο αναλύονται σε δυάδες από πεδία και τιμές (Application Data Values). Το Shindig παρέχει την απαραίτητη υποδομή για την ανάκτηση αυτών των δεδομένων με ειδικές κλήσεις, όπως επίσης και για τη δημιουργία, τη διαγραφή και την επεξεργασία τους μέσα από εφαρμογές.

Για να εξεταστεί καλύτερα ο τρόπος με τον οποίο το Shindig χειρίζεται τις παραπάνω τρεις βασικές οντότητες, αξίζει να εξηγηθούν συνοπτικά τα βασικά δομικά του στοιχεία, τα οποία συμβατικά διαχωρίζονται σε δύο συνιστώσες: εκείνη του εξυπηρετητή (server) και εκείνη του εξυπηρετούμενου (client).

Η συνιστώσα του client αναλαμβάνει τη δρομολόγηση των κλήσεων που αφορούν τις βασικές οντότητες του κοινωνικού δικτύου (Person, Activity, AppData), κατηγοριοποιώντας τις σε *REST* και *JSON-RPC* κλήσεις, που ορίζει ο OpenSocial Container

(όπως παρουσιάζονται στην Εικόνα 4.2). Οι κλήσεις REST (http://en.wikipedia.org/wiki/Representational_State_Transfer, August 2011) έχουν την μορφή ενός URL. Μία τέτοια κλήση με ένα συγκεκριμένο URL σε ένα Web Browser, επιστρέφει την αντίστοιχη πληροφορία απο τον server σε μορφή JSON (<http://www.json.org/>, August 2011). Στην περίπτωση των RPC κλήσεων χρησιμοποιείται το απλοποιημένο πρωτόκολλο JSON-RPC, στο πρότυπο του οποίου ορίζονται και κάποιες απλές κλήσεις που χρησιμοποιούνται από gadgets. Και τα δύο είδη κλήσεων τρέχουν πάνω στον Restful και στον JSON-RPC Container αντίστοιχα στο front-end του Shindig, οι οποίοι τις προωθούν σε μορφή XML-HTTP αιτημάτων στον Shindig Server.

Ο server με τη σειρά του δεσμεύει τις παραπάνω κλήσεις και εφόσον τις «φιλτράρει» ανά οντότητα που αφορούν, τις χειρίζεται με τις αντίστοιχες *Handler* κλάσεις: *PersonHandler*, *ActivitiesHandler* και *AppDataHandler*. Οι κλάσεις αυτές εξάγουν τις παραμέτρους των κλήσεων και με αυτές επικαλούνται οι αντίστοιχες *Service* κλάσεις: *PersonService*, *ActivityService*, *AppDataService*, στις οποίες κατασκευάζονται οι επερωτήσεις (queries) που είτε επιστρέφουν κοινωνική πληροφορία σε JSON μορφή, είτε αποθηκεύουν νέα κοινωνική πληροφορία.



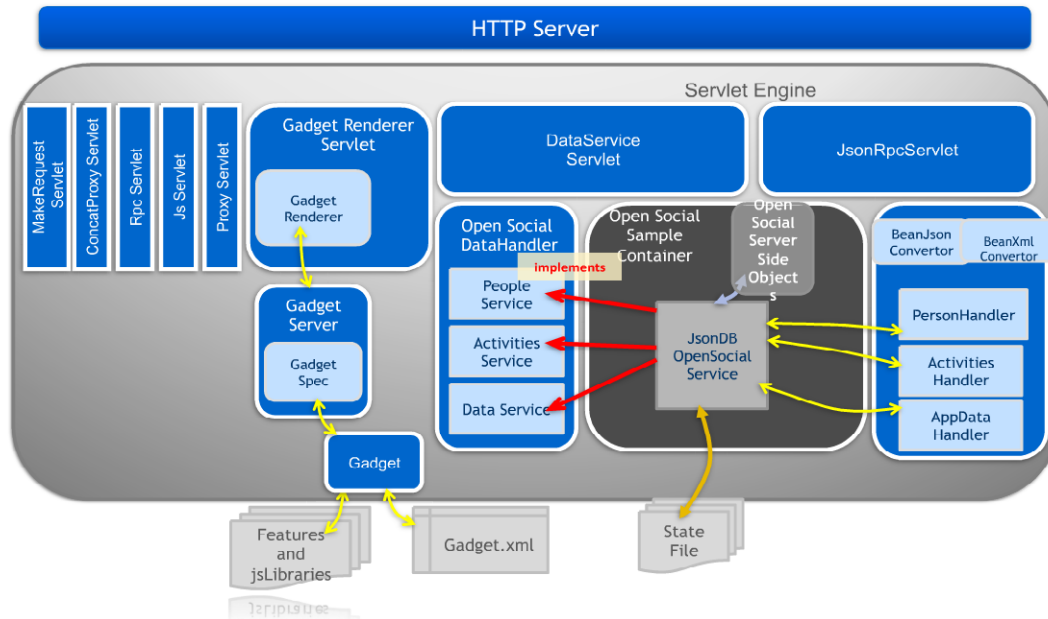
Εικόνα 4.2 – Client-Server ποή στο Shindig (Rajdeep Dua, 2008)

Στο υπόβαθρο του Shindig, ορίζονται επίσης οι βασικές κλάσεις που αντιστοιχούν στις βασικές οντότητες του κοινωνικού δικτύου: PersonDb, FriendDb, ActivityDb, ApplicationDataMapDb, ApplicationDataValueDb. Στις κλάσεις αυτές απεικονίζονται όλα τα γνωρίσματα κάθε οντότητας και γίνεται η αντιστοίχιση τους με τους αντίστοιχους πίνακες και γνωρίσματα στη βάση δεδομένων μέσω της τεχνολογίας του Java Persistence API.

Το Java Persistence API (JPA) είναι μία υπηρεσία, η οποία επιτρέπει τη *διαχείριση σχεσιακών δεδομένων* μέσω της Java, όπου κάθε Java κλάση αντιστοιχίζεται σε μία *σχέση* από μία βάση δεδομένων, ενώ κάθε *εγγραφή* της σχέσης αντιστοιχίζεται σε ένα *γνώρισμα της κλάσης Java*. Παράλληλα, με τη χρήση της γλώσσας JPQL (Java Persistence Query Language), μπορεί κανείς να πραγματοποιήσει επερωτήσεις (queries) στη βάση δεδομένων και να διαχειριστεί *αντικείμενα*, αντί για σχεσιακά δεδομένα. Με αυτό τον τρόπο επιτυγχάνεται η ευκολότερη διαχείριση της πληροφορίας που διατίθεται από την βάση δεδομένων μιας υπηρεσίας, όπως επίσης και η διατήρηση (persistence) των δεδομένων, πράγμα που ουσιαστικά σημαίνει ότι ο χρόνος ορατότητας τους γίνεται ίσος με το χρόνο ζωής των αντίστοιχων αντικειμένων.

Από τα παραπάνω θα μπορούσε να συμπεράνει κανείς ότι οι κλήσεις του Shindig που ανακτούν πληροφορία σε τελική ανάλυση δεν είναι τίποτα άλλο από επερωτήσεις σε JPQL προς τη βάση δεδομένων. Γι' αυτό το λόγο, όλες οι διαθέσιμες επερωτήσεις (κλήσεις) που έχουν νόημα στην παρούσα εργασία ορίζονται στις κλάσεις PersonDb, FriendDb, ActivityDb και ApplicationDataMapDb σε μορφή απλών συμβολοσειρών. Τις συμβολοσειρές αυτές χρησιμοποιούν οι αντίστοιχες κλάσεις-υλοποιήσεις των Service κλάσεων: PersonServiceDb, ActivityServiceDb και AppDataServiceDb, προσθέτουν προαιρετικά πεδία για να παραμετροποιήσουν τις αντίστοιχες κλήσεις και, τέλος, τις εκτελούν.

Τέλος, όπως φαίνεται και στην Εικόνα 4.3, μεταξύ των κλάσεων Handlers και των κλάσεων Services παρεμβάλεται η υπηρεσία JsonDB που παρέχει δειγματικά δεδομένα, πάνω στα οποία δρουν οι προεπιλεγμένες (default) κλήσεις του Shindig. Συνεπώς, αν μία υπηρεσία κατέχει μία δική της βάση δεδομένων, η οποία καλείται να αλληλεπιδράσει το Shindig, θα πρέπει να αφαιρεθούν όλες οι εξαρτήσεις από την JsonDB υπηρεσία, ώστε να ενσωματωθεί κατάλληλα η νέα βάση δεδομένων.



Εικόνα 4.3 – Δομικά στοιχεία του Shindig (Rajdeep Dua, 2008)

4.2 Δημιουργία και Διαχείριση Εφαρμογών

4.2.1 Εγκατάσταση Shindig

4.2.1.1 Εγκατάσταση

Την τελευταία έκδοση του Shindig μπορεί να προμηθευτεί κανείς από τη κεντρική του ιστοσελίδα (<http://shindig.apache.org>), είτε σε PHP, είτε σε Java. Η έκδοση που χρησιμοποιήθηκε στα πλαίσια της εργασίας είναι η 2.0 σε Java, η οποία αναφέρεται στην έκδοση 0.9 των προδιαγραφών του OpenSocial.

Πριν την εγκατάσταση του Shindig, απαιτείται η εγκατάσταση κάποιων βασικών εργαλείων, τα οποία είναι:

- ✓ Java (JRE/JDK) για να υποστηρίξει το σύστημα που φιλοξενεί το Shindig την αντίστοιχη γλώσσα.
- ✓ Ένας Subversion client για την ενημέρωση των εκδόσεων του κώδικα του Shindig.

- ✓ Το εργαλείο Apache Maven για να πραγματοποιείται η μεταγλώττιση και κτίσιμο (build) του κώδικα.

Εφόσον γίνει η εγκατάσταση των παραπάνω, τα αρχεία του Shindig μπορούν να τοποθετηθούν σε κάποιο φάκελο και να εγκατασταθούν με κάποιες απλές εντολές. Οι εντολές αυτές πραγματοποιούν τη μεταγλώττιση του κώδικα και ενεργοποιούν έναν τοπικό Jetty Web Server (<http://jetty.codehaus.org/jetty/>), ο οποίος κάνει διαθέσιμες της υπηρεσίες του Shindig, *τοπικά* στον υπολογιστή που είναι τοποθετημένο. Έστερα κι από αυτό το βήμα, υπάρχει η δυνατότητα προσπέλασης του OpenSocial Container που παρέχεται και να εκτελεστούν κάποια gadgets πιλοτικά με κάποια δοκιμαστικά δεδομένα που παρέχονται σε μορφή JSON σαν είσοδος στο Shindig.

4.2.1.2 Ενσωμάτωση Βάσης Δεδομένων

Το επόμενο βήμα προς την ολοκλήρωση της ενσωμάτωσης του Shindig είναι η σύνδεση μίας ήδη υπάρχουσας βάσης δεδομένων -με κοινωνικά δεδομένα- σε αυτό. Ευτυχώς, το Shindig φροντίζει για τη δημιουργία *πρότυπων σχεσιακών δεδομένων* με όλα τα κοινωνικά πεδία που ορίζει το OpenSocial και όχι μόνο. Συνεπώς, πρέπει να υπάρχει μία βάση δεδομένων, η οποία να τα αποθηκεύει. Στα πλαίσια αυτής της εργασίας έγινε χρήση μίας MySQL βάσης δεδομένων, ο χειρισμός της οποίας ανατέθηκε σε έναν τοπικό MySQL Server στο ίδιο μηχάνημα όπου εγκαταστάθηκε και δοκιμάστηκε το Shindig.

Έστερα, ακολουθώντας λεπτομερείς οδηγίες από το Wiki του Apache Shindig (<https://cwiki.apache.org/SHINDIG>) εκτελέστηκαν τα παρακάτω βήματα :

1. Η τροποποίηση των παρεχόμενων χαρακτηριστικών (features) του Shindig, ώστε να περιλαμβάνουν πραγματικές (και όχι πιλοτικές) υλοποιήσεις των οντοτήτων τις οποίες χειρίζεται. Πρόκειται για τις υλοποιήσεις των Java κλάσεων PersonService, AppDataService, ActivityService κλπ, οι οποίες προσφέρουν βασικές υπηρεσίες του κοινωνικού δικτύου.
2. Η εισαγωγή νέων χαρακτηριστικών, τα οποία επιτρέπουν στο Shindig να αναγνωρίζει εξωτερικές βάσεις δεδομένων και να συνδέεται με αυτές με τους κατάλληλους οδηγούς (drivers).

3. Η ενσωμάτωση κάποιων Java κλάσεων και η παραγωγή κάποιων κλειδιών ασφαλείας SSL για να επιτραπεί ο χειρισμός των προστατευμένων δεδομένων του Shindig με τη βοήθεια του OAuth πρωτοκόλλου.
4. Η εισαγωγή των στοιχείων προσπέλασης για την ενσωμάτωση εξωτερικής βάσης δεδομένων στο Shindig (τοποθεσία MySql server, ονομασία βάσης, κωδικός χρήστη).
5. Η αποτροπή της εκ νέου δημιουργίας μίας βάσης δεδομένων, κάθε φορά που επανεκκινείται το Shindig.

Αφού πραγματοποιήθηκαν τα παραπάνω βήματα, δοκιμάστηκαν απλές κλήσεις REST όπως η παρακάτω, σε έναν Web Browser :

<http://localhost:8080/social/rest/people/user1/@self>

Η κλήση αυτή επέστρεψε πληροφορία που είναι αποθηκευμένη στη βάση δεδομένων για τον χρήστη με αναγνωριστικό *user1*. Εάν όμως έπρεπε να γίνει κάποια επέκταση της βάσης δεδομένων, ώστε να περιέχει περισσότερη ή ακόμα και διαφορετική πληροφορία, θα έπρεπε να αποτυπωθεί αυτή την αλλαγή και στο ίδιο το Shindig με μια διαδικασία προσαρμογής των πεδίων που επιστρέφουν τέτοιου είδους κλήσεις.

4.2.2 Επέκταση Shindig

4.2.2.1 Επέκταση Βάσης Δεδομένων

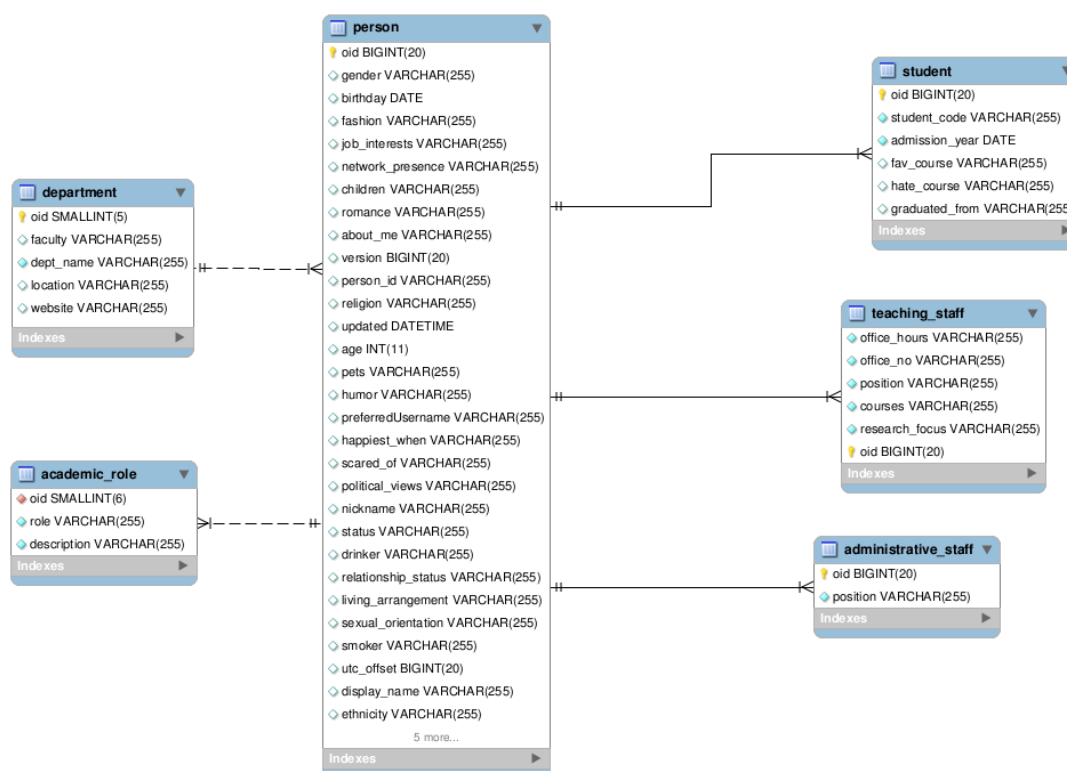
Κατά την ανάπτυξη του βασικού κορμού της υπηρεσίας Unity, η πρότυπη βάση δεδομένων του Shindig κρίθηκε ανεπαρκής για την υποστήριξη ενός ακαδημαϊκού ΙΚΔ. Αυτό συνέβη διότι τα χαρακτηριστικά ενός ατόμου που ορίζει το OpenSocial είναι κατά βάση *κοινωνικά*. Συνεπώς, πριν από οποιαδήποτε άλλη διαδικασία, η βάση δεδομένων πρέπει να επεκταθεί, ώστε να υποστηρίζει και την *ακαδημαϊκή διάσταση* των οντοτήτων και των σχέσεων, όπως αυτές ορίστηκαν στο προηγούμενο Μέρος. Κύριος σκοπός αυτού του εγχειρήματος είναι η δυνατότητα να προσπελάζεται η εν λόγω επιπρόσθετη πληροφορία μέσα από εφαρμογές.

Η βάση δεδομένων του Shindig περιλαμβάνει μόνο ένα γενικό πρότυπο συμμετεχόντων, το οποίο ορίζεται από τον πίνακα *person*, ο οποίος διατηρεί κατά

κύριο λόγο την κοινωνική πληροφορία που αφορά ένα άτομο. Όμως, για να γίνει εμφανής και η αντίστοιχη ακαδημαϊκή πληροφορία, πρέπει να οριστούν νέοι πίνακες – εξειδικεύσεις του υπάρχοντος, οι οποίοι να ακολουθούν τις παραδοχές που έχουν γίνει για τις διαφορετικές εκφάνσεις της οντότητας Person.

Για να είναι εφικτό κάτι τέτοιο, εισήχθηκε ένα πεδίο για τον ακαδημαϊκό ρόλο του ατόμου (*academic_role*). Παράλληλα, αντιστοιχίστηκαν οι ορισμοί των τριών βασικών οντοτήτων σε τρεις διαφορετικούς πίνακες, *student*, *teaching_staff*, *administrative_staff*. Οι πίνακες αυτοί αναφέρονται, μέσω foreign key, στο πεδίο κλειδί (*oid*) του πίνακα *person* και αποθηκεύουν λεπτομερώς ακαδημαϊκή πληροφορία για κάθε ρόλο και συμμετέχοντα του ακαδημαϊκού κοινωνικού δικτύου.

Επιπλέον, ένα άτομο σε ένα Πανεπιστημιακό Ίδρυμα εντάσσεται πάντα σε κάποιο υποτομέα, είτε αυτός είναι ένα τμήμα, είτε κάποια υπηρεσία. Έτσι, δημιουργώντας ένα νέο πίνακα *department*, ορίζονται και οι υποτομείς αυτοί, στους οποίους εν συνεχεία αντιστοιχίζονται όλες οι εγγραφές του πίνακα *person* μέσω του πεδίου *dept*, το οποίο είναι ένα ακόμα ξένο κλειδί στο πεδίο *oid* του πίνακα *department*. Παρακάτω, στην Εικόνα 4.4 φαίνονται σχηματικά οι πέντε νέοι πίνακες που αφορούν την *μοντελοποίηση* των ακαδημαϊκών οντοτήτων και ο τρόπος σύνδεσής τους με τον πίνακα *person*.



Εικόνα 4.4 – Σχήμα Βάσης Δεδομένων των νέων πινάκων και σχέσεων

Με αυτές τις ενέργειες ολοκληρώνονται οι ορισμοί των βασικών και πρωταρχικών οντοτήτων του κοινωνικού δικτύου. Οι αλλαγές αυτές, με τις κατάλληλες ενέργειες θα είναι ορατές από το Shindig και κατ' επέκταση από τις ίδιες τις εφαρμογές που θα φιλοξενούνται από την ΙΚΔ και θα επιστρέφονται ως επιπλέον χαρακτηριστικά της οντότητας Person.

Σε επόμενο στάδιο, εξετάζεται πως αναπαρίστανται στη βάση δεδομένων τα είδη των σχέσεων μεταξύ των βασικών οντοτήτων. Το πρότυπο του OpenSocial, ακολουθώντας τη λογική των περισσότερων δημοφιλέστερων ιστοσελίδων κοινωνικής δικτύωσης, αναγνωρίζει ένα τύπο σχέσης: τη *σχέση φιλίας (friend)*, η οποία αποτυπώνεται στον αντίστοιχο πίνακα *friend* στη βάση δεδομένων. Όμως, όπως προαναφέρθηκε, στα πλαίσια του ακαδημαϊκού κοινωνικού δικτύου, αυτή η σχέση από μόνη της δεν περιγράφει πλήρως τους συσχετισμούς ρόλων στα πλαίσια μίας ακαδημαϊκής κοινότητας.

Για να επιλυθεί αυτό το ζήτημα, δε δημιουργήθηκε νέος πίνακας στο σχήμα της βάσης, καθώς μία τέτοια ενέργεια θα συγκρουόταν με τις επιταγές του ήδη υπάρχοντος προτύπου. Αντίθετα, προστίθεται ένα επιπλέον πεδίο *ftype* στον πίνακα *friend*, στο οποίο απεικονίζεται το είδος της σχέσης. Με αυτόν τον τρόπο, μπορούν πλέον να οριστούν διαφορετικά “χρώματα” -εκφάνσεις δηλαδή- της απλής σχέσης *friend*. Αυτό που αλλάζει για κάθε έκφανση είναι ο τρόπος που καταγράφεται στον πίνακα. Η λογική για κάτι τέτοιο είναι η ακόλουθη:

- Εάν η σχέση είναι αμφίδρομη (*fellow*), απεικονίζεται με δύο εγγραφές στον πίνακα *friend*. Έτσι, αν ο συμμετέχων Α στείλει ένα αίτημα φιλίας στον συμμετέχοντα Β και αυτός αποδεχτεί, θα προκύψουν οι εξής εγγραφές:

oid	score	version	person_id	friend_id	type
...	...	...	id _A	id _B	fellow
...	...	...	id _B	id _A	fellow

- Εάν η σχέση είναι μονόδρομη (*tutor, facilitator*), απεικονίζεται με μία και μόνο εγγραφή στον πίνακα *friend*. Για παράδειγμα, αν ο συμμετέχων Α επιλέξει να συνδεθεί με τον Β μονοσήμαντα, θα προκύψει η εξής εγγραφή:

oid	score	version	person_id	friend_id	type
...	...	...	id _A	id _B	tutor/ facilitator

Άλλη μία επέκταση άξια αναφοράς είναι εκείνη που έχει γίνει στον πίνακα *activity*. Όπως αναφέρθηκε στην προηγούμενη ενότητα, η νεότερη έκδοση του OpenSocial (2.0) υιοθετεί το πρότυπο του Activity Streams 1.0 (<http://activitystrea.ms>) για τη διαχείριση της ενημέρωσης χρηστών για τις δραστηριότητες των φίλων τους. Η έκδοση του Apache Shindig που χρησιμοποιήθηκε, όμως, δεν ακολουθεί τις επιταγές του νέου OpenSocial και, κατ' επέκταση, του Activity Streams. Σύμφωνα με το τελευταίο, τα πεδία που ορίζονται για κάθε δραστηριότητα είναι τα εξής:

- actor: ο χρήστης-υποκείμενο της δραστηριότητας,
- verb: το κατηγορημα της δραστηριότητας που καταδεικνύει την ενέργεια,
- object: η οντότητα-αντικείμενο της δραστηριότητας,
- target: η οντότητα ή η συλλογή οντοτήτων που ενημερώνεται για τη δραστηριότητα.

Για παράδειγμα, στη δραστηριότητα “Geraldine posted a photo to her album”, actor ορίζεται ο χρήστης Geraldine, verb είναι η ενέργεια, δηλαδή η δημοσίευση μίας φωτογραφίας (“posted a photo”) και object είναι η τοποθεσία όπου τη δημοσίευσε, δηλαδή το album της (“to her album”).

Επιστρέφοντας τώρα στον πίνακα *activity* που δημιούργησε το Shindig, τα πεδία που μπορούν να αντιστοιχηθούν στο παραπάνω θεωρητικό μοντέλο είναι: το πεδίο *userId* και το πεδίο *title*, τα οποία αντιστοιχίζονται με τις έννοιες actor και verb αντίστοιχα. Τέλος, για να ολοκληρωθεί η ταύτιση με το μοντέλο του Activity Streams, ορίζονται και δύο νέα πεδία: *object* και *target* με τις αντίστοιχες προφανείς συσχετίσεις.

Εν συνεχεία, η επόμενη κίνηση αφορά την εγκατάσταση των εφαρμογών από τους χρήστες στην ΙΚΔ, στοιχείο που δεν αφορά τόσο τη λειτουργία του Shindig, αλλά εκείνη του κύριου κορμού του Unity. Εφόσον προβλέπεται η παροχή ακαδημαϊκών υπηρεσιών στους χρήστες σε μορφή εφαρμογών, είναι εύλογη η παραδοχή ότι δε μπορούν όλοι οι χρήστες να έχουν πρόσβαση στις ίδιες εφαρμογές. Αυτό σημαίνει πως πρέπει να γίνεται έλεγχος της δυνατότητας κάθε χρήστη να εγκαταστήσει μια εφαρμογή βάσει πάντα του ακαδημαϊκού του ρόλου. Συνεπώς, για κάθε εφαρμογή ορίζονται και τα αντίστοιχα δικαιώματα εγκατάστασης ανά ακαδημαϊκό ρόλο. Η απεικόνιση αυτών των δικαιωμάτων πραγματοποιείται σε ένα νέο πίνακα: τον πίνακα *application_access*, ο οποίος περιέχει τα πεδία *app_id* (αναγνωριστικό εφαρμογής) και *academic_role* (ακαδημαϊκός ρόλος). Σημειώνεται πως αμφότερα τα δύο πεδία αποτελούν το πρωτεύον κλειδί του πίνακα. Έτσι, για να οριστεί ότι μία εφαρμογή μπορεί να εγκατασταθεί μόνο από προπτυχιακούς φοιτητές, αρκεί μία εγγραφή με το αναγνωριστικό της εφαρμογής και τον αντίστοιχο ακαδημαϊκό ρόλο αυτών.

4.2.2.2 Επέκταση Shindig

Οποιαδήποτε τροποποίηση στην πρότυπη βάση δεδομένων δεν είναι σε καμία περίπτωση ορατή από το Shindig, καθώς το JPA δεν αντικατοπτρίζει τα νέα πεδία που έχουν προστεθεί. Επιπλέον, οι νέες κλήσεις χρειάζεται όχι μόνο να οριστούν και στο Shindig, αλλά και να υλοποιηθούν με JPQL επερωτήσεις. Συνεπώς, χρειάζεται να ανατρέξει κανείς για άλλη μία φορά στις κλάσεις *PersonDb*, *FriendDb* και *ActivityDb*.

Στις κλάσεις αυτές, όπως έχει αναφερθεί, γίνεται η αντιστοίχιση των οντοτήτων του συστήματος με τους αντίστοιχους πίνακες που τις απεικονίζουν στη βάση δεδομένων μέσω του JPA. Εδώ, λοιπόν, γίνεται η εισαγωγή των νέων πεδίων σε μορφή μεταβλητών, οι οποίες γίνονται προσπελάσιμες και τροποποιήσιμες μέσω των αντίστοιχων *Get* και *Set* συναρτήσεων από άλλες κλάσεις. Επιπλέον, εδώ ορίζονται οι πρωταρχικές JPQL επερωτήσεις που στη συνέχεια θα χρησιμοποιηθούν από τις *Service* κλάσεις, για να εμπλουτιστούν με περισσότερα κριτήρια. Με την επεξεργασία αυτών, μπορεί κανείς να επεκτείνει τις αντίστοιχες κλήσεις, ώστε αυτές να επιστρέφουν άλλου είδους πληροφορία. Η τελική μορφή μίας επεκταμένης JPQL επερώτησης είναι συνήθως ως εξής:

@Entity

```
@Table(name = "people")

@NamedQueries(value = {

    @NamedQuery(name = PersonDb.FINDBY_PERSONID,

        query = "select NEW org.apache.shindig.social.opensocial.jpa.PersonDb(p.id,
p.displayName) from PersonDb p where p.id = :id ") })
```

Είναι εύκολα αντιληπτό ότι γίνεται σε μεγάλο βαθμό χρήση ετικετών (annotations) της Java, με τις οποίες ορίζονται οι αντιστοιχίσεις και το είδος των αντιστοιχίσεων με τους πίνακες της βάσης δεδομένων, καθώς και οι επερωτήσεις πάνω σε αυτούς. Φυσικά, μία τελευταία παρατήρηση στα παραπάνω είναι ότι μία τέτοιου είδους επερώτηση χειρίζεται και επιστρέφει αντικείμενα σε Java. Λόγου χάρη, στο παραπάνω παράδειγμα ορίζεται η συσχέτιση της επερώτησης με τον πίνακα people (μέσω της ετικέτας «@Table»), ενώ παρακάτω ορίζεται μία λίστα από επερωτήσεις πάνω στον πίνακα αυτό (μέσω των ετικετών «@NamedQuery»). Η πληροφορία εκμαιεύεται πλέον από αντικείμενα της κλάσης PersonDb, η οποία αποτελεί απεικόνιση της οντότητας «Άτομο».

Η παραπάνω υλοποίηση αφορά την εξαγωγή *παραμετροποιημένης πληροφορίας* από τη βάση δεδομένων. Όσον αφορά την *εισαγωγή* ή την *τροποποίηση* πληροφορίας, η διαδικασία είναι πολύ πιο απλή. Αρκεί μόνο μία απλή ανάθεση των νέων χαρακτηριστικών στα Java αντικείμενα, χωρίς τη διενέργεια επερωτήσεων. Οι νέες αλλαγές γίνονται απευθείας διατηρητές (persisted) και αποτυπώνονται και στη βάση δεδομένων.

Με τις παραπάνω παραδοχές, η εισαγωγή νέων πεδίων στο Shindig είναι πλέον εύκολη υπόθεση. Στην κλάση PersonDb εισάγονται τα πεδία για τον ακαδημαϊκό ρόλο του χρήστη και για το τμήμα/υπηρεσία στην οποία είναι ενταγμένος. Στην κλάση FriendDb ορίζεται το πεδίο ftype, μέσω του οποίου ορίζεται το είδος της σχέσης μεταξύ χρηστών. Τέλος, στην κλάση ActivityDb εισάγονται τα δύο χαρακτηριστικά object και target, όπως αυτά ορίστηκαν στον αντίστοιχο πίνακα της βάσης δεδομένων.

Το επόμενο και τελευταίο βήμα για την επέκταση του Shindig είναι ο ορισμός των *συλλογών* (*groups*) των ατόμων από τις οποίες επιστρέφει πληροφορία μία κλήση. Για παράδειγμα, στην παρούσα υπηρεσία μπορεί μία κλήση να επιστρέφει πληροφορία από τους καθηγητές (*tutors*) ενός φοιτητή, συλλογή που έχει προκύψει από την αντίστοιχη σχέση που έχει συνάψει ο φοιτητής με χρήστες συγκεκριμένου ακαδημαϊκού ρόλου. Το Shindig υποστηρίζει κλήσεις που αφορούν έναν χρήστη (*self*), εκείνες που αφορούν την συγκεκριμένη ομάδα «φίλοι» (*friends*) και, τέλος, κλήσεις που αφορούν όλες τις επαφές ενός χρήστη (*all*). Οι νέες ομάδες που προστίθενται είναι πλέον γνωστές: *fellows*, *tutors*, *facilitators* στη θέση της παρωχημένης πλέον ομάδας «φίλοι». Ταυτόχρονα, οι νέες κλήσεις που λαμβάνουν τις ομάδες αυτές σαν ορίσματα δεν είναι τίποτα παραπάνω από αντίγραφα των παλαιότερων.

4.2.2.3 Υποστηριζόμενες Κλήσεις

Με τη διενέργεια των παραπάνω βημάτων, από το χαμηλό επίπεδο του JPA, προκύπτουν νέες REST και JSON-RPC κλήσεις, οι οποίες έχουν επεκταθεί ώστε να εξυπηρετούν τις ανάγκες της δεδομένης ακαδημαϊκής ΙΚΔ. Οι κλήσεις αυτές μπορούν να διακριθούν σε τρεις μεγάλες κατηγορίες:

- **People:** κλήσεις που επιστρέφουν πληροφορία για τους χρήστες του κοινωνικού δικτύου
- **Activities:** κλήσεις που επιστρέφουν πληροφορία για τις δραστηριότητες των χρηστών στην υπηρεσία
- **AppData:** κλήσεις που επιστρέφουν πληροφορία που σχετίζεται με τις τοπικές μεταβλητές των εφαρμογών

Για την διευκόλυνση της περιγραφής των κλήσεων, όπως αυτές ορίστηκαν και επεκτάθηκαν στα πλαίσια της υπηρεσίας Unity, παρατίθενται οι Πίνακες 4.1 και 4.2 που παρουσιάζουν τις κλήσεις REST και JSON-RPC, αντίστοιχα.

Ομάδα κλήσεων	Κλήση / Περιγραφή
People	/social/rest/people/{guid}/@all Επιστρέφει συλλογή ατόμων με τα οποία έχει συνδεθεί το άτομο {guid} και τις πληροφορίες τους. /social/rest/people/{guid}/@fellows Αποτελεί τροποποίηση της αντίστοιχης, βασικής κλήσης που υποστηρίζει το Shindig για την ανάκτηση συλλογής των φίλων (friends) ενός ατόμου. Επιστρέφει συλλογή ατόμων που έχουν σχέση fellow με το άτομο {guid} και τις πληροφορίες τους. /social/rest/people/{guid}/@tutors Αποτελεί τροποποίηση της αντίστοιχης, βασικής κλήσης που υποστηρίζει το Shindig για την ανάκτηση συλλογής των φίλων (friends) ενός ατόμου. Επιστρέφει συλλογή ατόμων/καθηγητών που ακολουθεί το άτομο {guid} και τις πληροφορίες τους. /social/rest/people/{guid}/@facilitators Αποτελεί τροποποίηση της αντίστοιχης, βασικής κλήσης που υποστηρίζει το Shindig για την ανάκτηση συλλογής των φίλων (friends) ενός ατόμου. Επιστρέφει συλλογή ατόμων/μελών προσωπικού που ακολουθεί το άτομο {guid} και τις πληροφορίες τους. /social/rest/people/{guid}/@self Επιστρέφει τις προσωπικές πληροφορίες του ατόμου {guid}.
Activities	/social/rest/activities/{guid}/@self/{appid} Επιστρέφει συλλογή από δραστηριότητες του ατόμου {guid}. Επεκτάθηκε ώστε να επιστρέφει προαιρετικά δραστηριότητες που προέκυψαν από μία συγκεκριμένη εφαρμογή {appid}.

/social/rest/activities/{guid}/@fellows/{appid}

Αποτελεί τροποποίηση της αντίστοιχης, βασικής κλήσης που υποστηρίζει το Shindig για την ανάκτηση των δραστηριοτήτων των φίλων (friends) ενός ατόμου. Επιστρέφει συλλογή από δραστηριότητες των ατόμων που έχουν σχέση fellow με το άτομο {guid}. Επεκτάθηκε ώστε να επιστρέφει προαιρετικά δραστηριότητες για την εφαρμογή {appid}.

/social/rest/activities/{guid}/@tutors/{appid}

Αποτελεί τροποποίηση της αντίστοιχης, βασικής κλήσης που υποστηρίζει το Shindig για την ανάκτηση των δραστηριοτήτων των φίλων (friends) ενός ατόμου. Επιστρέφει συλλογή από δραστηριότητες των ατόμων που έχουν σχέση tutor με το άτομο {guid}. Επεκτάθηκε ώστε να επιστρέφει προαιρετικά δραστηριότητες για την εφαρμογή {appid}.

/social/rest/activities/{guid}/@facilitators/{appid}

Αποτελεί τροποποίηση της αντίστοιχης, βασικής κλήσης που υποστηρίζει το Shindig για την ανάκτηση των δραστηριοτήτων των φίλων (friends) ενός ατόμου. Επιστρέφει συλλογή από δραστηριότητες των ατόμων που έχουν σχέση facilitator με το άτομο {guid}. Επεκτάθηκε ώστε να επιστρέφει προαιρετικά δραστηριότητες για την εφαρμογή {appid}.

/social/rest/activities/{guid}/@facilitated/{appid}

Αποτελεί τροποποίηση της αντίστοιχης, βασικής κλήσης που υποστηρίζει το Shindig για την ανάκτηση των δραστηριοτήτων των φίλων (friends) ενός ατόμου. Επιστρέφει συλλογή από δραστηριότητες των ατόμων τα οποία έχουν σαν facilitator το άτομο {guid}. Επεκτάθηκε ώστε να επιστρέφει προαιρετικά δραστηριότητες για την εφαρμογή {appid}.

/social/rest/activities/{guid}/@students/{appid}

	Αποτελεί τροποποίηση της αντίστοιχης, βασικής κλήσης που υποστηρίζει το Shindig για την ανάκτηση των δραστηριοτήτων των φίλων (friends) ενός ατόμου. Επιστρέφει συλλογή από δραστηριότητες των ατόμων τα οποία έχουν σαν tutor το άτομο {guid}. Επεκτάθηκε ώστε να επιστρέφει προαιρετικά δραστηριότητες για την εφαρμογή {appid}.
AppData	/social/rest/appdata/{guid}/@self/{appid} Επιστρέφει συλλογή από δεδομένα εφαρμογών για τον χρήστη {guid} και την εφαρμογή {appid}. /social/rest/appdata/{guid}/@fellows/{appid} Επιστρέφει συλλογή από δεδομένα εφαρμογών των ατόμων που έχουν σχέση fellow με τον χρήστη {guid} για την εφαρμογή {appid}. /social/rest/appdata/{guid}/@facilitators/{appid} Επιστρέφει συλλογή από δεδομένα εφαρμογών των ατόμων που έχουν σχέση facilitator με τον χρήστη {guid} για την εφαρμογή {appid}. /social/rest/appdata/{guid}/@tutors/{appid} Επιστρέφει συλλογή από δεδομένα εφαρμογών των ατόμων που έχουν σχέση tutor με τον χρήστη {guid} για την εφαρμογή {appid}.

Πίνακας 4.1 – Περιγραφή των REST κλήσεων του Apache Shindig

Ομάδα κλήσεων	Κλήση / Περιγραφή
People	people.get Επιστρέφει πληροφορίες για άτομο ή άτομα που ανήκουν σε μία συγκεκριμένη ομάδα. Πρόκειται για το JSON-RPC ισοδύναμο

	των REST κλήσεων που αφορούν άτομα. <i>Ορίσματα:</i> id ατόμου, id συλλογής ατόμων
Activities	activities.create Δημιουργεί δραστηριότητα για ένα συγκεκριμένο άτομο. <i>Ορίσματα:</i> id ατόμου και οποιοδήποτε άλλο πεδίο (προαιρετικά) activities.get Επιστρέφει τις δραστηριότητες ενός ατόμου ή μίας συλλογής ατόμων. Πρόκειται για το JSON-RPC ισοδύναμο των REST κλήσεων που αφορούν τις δραστηριότητες ατόμων. <i>Ορίσματα:</i> id ατόμου, id συλλογής, id εφαρμογής (προαιρετικό) activities.delete Σβήνει όλες τις δραστηριότητες ενός χρήστη που αφορούν μία συγκεκριμένη εφαρμογή. <i>Ορίσματα:</i> id χρήστη, id εφαρμογής
AppData	appdata.get Επιστρέφει τα δεδομένα εφαρμογών ενός ατόμου ή μίας συλλογής ατόμων. Πρόκειται για το JSON-RPC ισοδύναμο των REST κλήσεων που αφορούν τα δεδομένα εφαρμογών ατόμων. <i>Ορίσματα:</i> id ατόμου, id συλλογής ατόμων, id εφαρμογής appdata.update Τροποποιεί τα δεδομένα ενός χρήστη για μία συγκεκριμένη εφαρμογή. <i>Ορίσματα:</i> id εφαρμογής, id πεδίου, id νέας τιμής πεδίου

Πίνακας 4.2 – Περιγραφή των JSON-RPC κλήσεων του Apache Shindig

4.2.3 Ανάπτυξη Gadgets

4.2.3.1 Δημιουργία Gadgets

Πλέον έχει οριστεί το υπόβαθρο πάνω στο οποίο στηρίζονται οι εφαρμογές. Τώρα, όσον αφορά το τελικό αποτέλεσμα στον χρήστη, δηλαδή τα Gadgets, τα πράγματα είναι σχετικά απλούστερα:

Γενικά, ένα Gadget δεν είναι τίποτα άλλο παρά ένα αρχείο σε XML, το οποίο είναι δομημένος ως εξής:

```
<?xml version="1.0" encoding="UTF-8"?>

<Module>

  <ModulePrefs title="A Title">

    <Require feature="opensocial-0.8"/>

    ...

  </ModulePrefs>

  <Content type="html">

    <![CDATA[

      <script type="text/javascript">

        ...

        gadgets.util.registerOnLoadHandler(init);

      </script>

      <div id="id"/>

      ...

    ]]>

  </Content>

</Module>
```


Στο στοιχείο (element) «Content», περιέχεται ένα μπλοκ κώδικα σε JavaScript και HTML που στην ουσία εκτελεί όλες τις λειτουργίες του Gadget, χρησιμοποιώντας προαιρετικά και τις κλήσεις που υποστηρίζει το Shindig. Στο στοιχείο «ModulePrefs» περιλαμβάνονται τα επιπλέον στοιχεία (features) τα οποία χρησιμοποιεί το Gadget για να λειτουργήσει, όπως για παράδειγμα μία διεπιφάνεια κλήσεων του OpenSocial (osapi). Τέλος, αυτά τα αρχεία φιλοξενούνται από τον Gadget Server του Shindig, ο οποίος φροντίζει για την απόδοση και την εκτέλεση του κώδικα τους, παρέχοντας σε αυτά όλες τις κλήσεις που υποστηρίζει ο OpenSocial Container.

Ο προγραμματισμός ενός Gadget –εφόσον γίνεται κατά κύριο λόγο σε JavaScript– είναι πρωτίστως στηριζόμενος στα γεγονότα που προκύπτουν κατά την προσπέλαση του από έναν χρήστη. Τέτοια γεγονότα μπορεί να είναι: το «κλικ» του ποντικιού, το πάτημα ενός κουμπιού στο πληκτρολόγιο, η αλλαγή της επιλογής ενός χρήστη σε μία λίστα κλπ. Το γεγονός αυτό τονίζει ιδιαίτερα την δυναμικότητα των Gadgets, παρόλο που φορτώνονται ή φαίνονται να φορτώνονται ως ιστοσελίδες.

Για τις OpenSocial κλήσεις μέσα από ένα Gadget, υπάρχει το αντίστοιχο «ελαφρύ» (lightweight) JavaScript API, εφεξής αναφερόμενο ως osapi. Τα αιτήματα μέσω αυτού στηρίζονται στο πρωτόκολλο JSON-RPC και κάνουν χρήση στην είσοδο, όσο και στην έξοδο τους δεδομένων σε JSON μορφή. Τα βασικά αιτήματα για άλλη μια φορά μπορούν να διακριθούν στις τρεις βασικές ομάδες, που αντιστοιχίζονται στις JSON-RPC κλήσεις όπως φαίνονται στον πίνακα 4.13.

Ομάδα κλήσεων	Αίτημα / Περιγραφή
People	osapi.people.get Αντίστοιχο της JSON-RPC κλήσης: people.get. osapi.people.getViewer Επιστρέφει πληροφορίες που αφορούν το άτομο – θεατή (viewer) του Gadget. osapi.people.getOwner

	Επιστρέφει πληροφορίες που αφορούν το άτομο – κάτοχο (owner) του Gadget.
Activities	osapi.activities.create Αντίστοιχο της JSON-RPC κλήσης: activities.create. osapi.activities.get Αντίστοιχο της JSON-RPC κλήσης: activities.get. osapi.activities.delete Αντίστοιχο της JSON-RPC κλήσης: activities.delete.
AppData	osapi.appdata.get Αντίστοιχο της JSON-RPC κλήσης: appdata.get. osapi.appdata.update Αντίστοιχο της JSON-RPC κλήσης: appdata.update.

Πίνακας 4.3: Αιτήματα/Εντολές του osapi και η αντιστοίχηση τους σε κλήσεις

Τα παραπάνω αιτήματα μπορούν να χρησιμοποιηθούν αυτούσια στο εσωτερικό ενός gadget. Η διενέργεια ενός αιτήματος απαιτεί και την ύπαρξη μίας JavaScript συνάρτησης, η οποία χειρίζεται τα δεδομένα που επιστρέφονται (callback function). Ένα τέτοιο παράδειγμα μεμονωμένου αιτήματος θα μπορούσε να είναι το εξής:

```
osapi.people.get({userId: '@owner', groupId: '@friends'}).execute(
function(ownerFriends) {
    if (!result.error) {
        alert('You have ' + ownerFriends.length + ' friends');
    }
});
```

Στο παραπάνω παράδειγμα, γίνεται χρήση του αιτήματος osapi.people.get με ορίσματα το αναγνωριστικό του χρήστη (userId) για τον οποίον πραγματοποιείται η

αντίστοιχη κλήση, καθώς και η συλλογή χρηστών (groupId) για την οποία αναζητείται πληροφορία. Πιο συγκεκριμένα, το παραπάνω αίτημα επιστρέφει πληροφορίες για τους φίλους (friends) του κατόχου (owner) του gadget. Εφόσον πραγματοποιηθεί το αίτημα, γίνεται έλεγχος για τυχόν λάθη κατά την πραγματοποίησή του, αλλιώς εμφανίζεται ένα μήνυμα με το μέγεθος της επιστρεφόμενης λίστας φίλων στο χρήστη.

Εκτός των μεμονωμένων αιτημάτων, μέσω του osapi μπορεί κανείς να πραγματοποιήσει μία *δέσμη* (batch) αιτημάτων, πραγματοποιώντας στην ουσία παραπάνω από μία κλήσεις ταυτόχρονα στο Shindig, για να χειριστεί συγκεντρωτικά τα δεδομένα που επιστρέφονται. Ένα παράδειγμα παράλληλων αιτημάτων θα μπορούσε να είναι το εξής:

```
var batch = osapi.newBatch().

    add("viewer", osapi.people.getViewer()).

    add('activities', osapi.activities.get({ userId : '@viewer', groupId : '@self'})).

    batch.execute(function(result) {

        if (!result.error) {

            alert('Hi, ' + result.viewer.name + '!');

            alert('You have ' + result.activities.length + ' activities.');

        }

    });
```

Στις πρώτες γραμμές ορίζεται η δέσμη αιτημάτων (osapi.newBatch()), στην οποία προστίθενται αιτήματα με τη χρήση της συνάρτησης add, τα αποτελέσματα των οποίων ονοματίζονται για να ξεχωρίζουν («viewer» και «activities»). Οι παραπάνω δύο κλήσεις επιστρέφουν τον θεατή του gadget (viewer) και τις δραστηριότητες του ίδιου. Με την εκτέλεση αυτών των δύο αιτημάτων, εάν δεν υπάρξει κάποιο λάθος, εμφανίζονται δύο μηνύματα στο χρήστη με το όνομα του και το μέγεθος της επιστρεφόμενης λίστας δραστηριοτήτων.

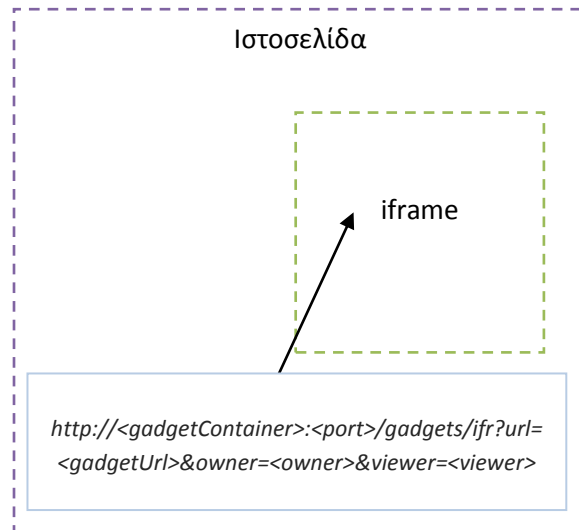
4.2.3.2 Υποστηριζόμενος Μηχανισμός Ενσωμάτωσης

Επόμενο βήμα για την εισαγωγή εφαρμογών στην υπηρεσία Unity, είναι να βρεθεί ένας μηχανισμός ενσωμάτωσης τους. Το OpenSocial ορίζει πως η ενσωμάτωση αυτή θα γίνεται μέσω της HTML ετικέτας `iframe`, η οποία παράγει ένα πλαίσιο σε μία ιστοσελίδα, μέσα στο οποίο φορτώνεται το αντικείμενο μίας δοθείσας διεύθυνσης URL. Στην περίπτωση αυτή, τα `iframes` που χρησιμοποιούνται δέχονται σαν είσοδο τη διεύθυνση URL του Gadgets Container και τα κάποια επιπλέον ορίσματα που αφορούν το προς ενσωμάτωση gadget:

- `?syndQ`: το όνομα του Container
- `&container`: το όνομα του Container
- `&viewer`: ο κωδικός του viewer
- `&owner`: ο κωδικός του owner
- `&aid`: ο κωδικός της εφαρμογής
- `&mid`: το νούμερο του module όπου φιλοξενείται το gadget
- `&country`: ο κωδικός της χώρας (πχ. «GR»)
- `&lang`: ο κωδικός της γλώσσας (πχ. «EL»)
- `&view`: το όνομα της όψης από την οποία προσπελάζεται το gadget.
- `#view-params`: ένα αντικείμενο JSON που περιέχει πληροφορίες για τις παραμέτρους της περιοχής όπου είναι ενσωματωμένο το gadget.
- `&parent`: η διεύθυνση URL του Container
- `&st`: ένα κωδικοποιημένο token
- `&n`: μία κωδικοποιημένη μορφή του XML του gadget
- `&url`: η διεύθυνση URL του gadget
- `#rpctoken`: ένας τυχαίος, μοναδικός αριθμός

Η «κατασκευή» ενός τέτοιου URL είναι μία διαδικασία που αναλαμβάνει ο server, δηλαδή ο κύριος κορμός της υπηρεσίας Unity, συμπεριλαμβάνοντας όσες παραμέτρους από τις παραπάνω είναι απαραίτητες. Εάν όλα πάνε καλά, η ιστοσελίδα

θα φορτώσει σε ένα iframe πλαίσιο το επιθυμητό gadget με όλη του τη λειτουργικότητα με τη βοήθεια του αντίστοιχου URL, όπως στο παράδειγμα της εικόνας 3.5. Παράλληλα, Ο προγραμματιστής μπορεί να προσαρμόσει την εμφάνιση του iframe μέσα από τον κώδικα CSS της ιστοσελίδας, ώστε να ενσωματωθεί κατάλληλα σε αυτήν.



Εικόνα 3.5 – Παράδειγμα ενσωμάτωσης gadget σε ιστοσελίδα μέσω iframe

4.2.3.3 Μηχανισμός Διαχείρισης Κύκλων Ζωής

Τελικό στάδιο για την πλήρη ενσωμάτωση των gadgets είναι η δημιουργία ενός μηχανισμού διαχείρισης των κύκλων ζωής τους. Σε όλους τους ΙΚΔ, η χρήση εφαρμογών είναι προαιρετική και οι χρήστες μπορούν με αυτή τη λογική να προσθέσουν, να αφαιρέσουν ή ακόμα και να απενεργοποιήσουν κάποιες από τις διαθέσιμες εφαρμογές. Στο Unity ακολουθείται ακριβώς η ίδια λογική, η οποία εν ολίγοις εστιάζει στις παρακάτω ενέργειες επί των εφαρμογών:

- Κάθε χρήστης αποκτά πρόσβαση σε μία λίστα από εφαρμογές.

Όταν ένας χρήστης αιτείται να προσπελάσει όλες τις διαθέσιμες εφαρμογές που μπορεί να εγκαταστήσει, η υπηρεσία Unity συμβουλεύεται τη βάση δεδομένων και, συγκεκριμένα, τον πίνακα `application_access`, επιστρέφοντας τις εφαρμογές εκείνες που απευθύνονται στον ακαδημαϊκό ρόλο του χρήστη.

- *Ο χρήστης προσθέτει όποια ή όποιες εφαρμογές επιθυμεί από τις παραπάνω στην συλλογή του. Αντίστοιχα, διαγράφει όσες δεν επιθυμεί να χρησιμοποιεί πλέον.*

Εάν επιλεχτεί κάτι τέτοιο, καλείται το αντίστοιχο Java Servlet (από τον κύριο κορμό της υπηρεσίας) που φροντίζει για την ενεργοποίηση μίας ειδικής Java κλάσης που είναι υπεύθυνη για τις παραπάνω λειτουργίες. Η κλάση αυτή δημιουργεί κάποιες εγγραφές στη βάση δεδομένων, οι οποίες αφορούν τα Δεδομένα Εφαρμογών (AppData) της προς εγκατάσταση εφαρμογής. Τα βήματα που ακολουθούνται είναι τα εξής:

1. Αρχικά, παράγεται ο αντίστοιχος χάρτης (Application Data Map) του χρήστη για τη συγκεκριμένη εφαρμογή. Πρακτικά, δηλαδή, η κατασκευάστρια Java κλάση προσθέτει στον πίνακα application_datamap της βάσης δεδομένων μία εγγραφή με τα αναγνωριστικά του χρήστη και της εφαρμογής, υποδηλώνοντας τον «χώρο» αποθήκευσης που ορίζεται για τον συγκεκριμένο συμμετέχοντα.
2. Ύστερα, δημιουργούνται οι εγγραφές για τις δυνάδες πεδίων και τιμών (Application Data Values) που αντιστοιχούν στον χάρτη αυτό. Ο πίνακας application_property της βάσης δεδομένων διατηρεί την πληροφορία για τα πεδία και τις αρχικές, προεπιλεγμένες τιμές τους και τροφοδοτεί την κατασκευάστρια Java κλάση, η οποία δημιουργεί τις κατάλληλες εγγραφές στον πίνακα application_datavalue της βάσης.
3. Τέλος, εφόσον όλα τα παραπάνω βήματα έχουν πραγματοποιηθεί χωρίς σφάλματα, δημιουργείται και μία τελευταία εγγραφή στον πίνακα person_application της βάσης δεδομένων, η οποία υποδηλώνει ότι ένας συγκεκριμένος χρήστης έχει εγκαταστήσει τη συγκεκριμένη εφαρμογή.

Αντίστοιχα, εάν ο χρήστης επιλέξει τη διαγραφή της εφαρμογής, θα καλείται το αντίστοιχο Java Servlet που θα επικοινωνεί με την ίδια κλάση, αυτή τη φορά ζητώντας διαγραφή όσων είχε αποθηκεύσει. Πρώτα διαγράφονται τα πεδία και οι τιμές από τον πίνακα application_datavalue, ύστερα οι χάρτες από τον πίνακα application_datamap και, τέλος, η εξάρτηση του χρήστη από την εφαρμογή από τον πίνακα person_application.

- Ο χρήστης προσπελαύνει και «τρέχει» εφαρμογές από το προφίλ του μέσω gadgets, όπου ενσωματώνονται με τη μορφή iframes, όπως ακριβώς ορίζει το Shindig.

Η εγκατάσταση εφαρμογών ουσιαστικά σημαίνει προσθήκη εφαρμογών σε μία λίστα, προσωπική για κάθε χρήστη, η οποία μπορεί εύκολα να παραχθεί από τον πίνακα person_application της βάσης δεδομένων. Σε κάθε προφίλ, ο κάτοχος αποκτά πρόσβαση στις εφαρμογές της δικής του λίστας και πλοηγείται ελεύθερα. Παράλληλα, του δίνονται δυνατότητες απόκρυψης των gadgets που τρέχει ή ακόμα και η δυνατότητα ανανέωσης του περιεχομένου τους, για διευκόλυνση της ενημέρωσης του χρήστη για αλλαγές των δεδομένων εφαρμογών. Όλες οι παραπάνω λειτουργίες πραγματοποιούνται με ειδικές συναρτήσεις σε JavaScript που ενισχύουν την αλληλεπίδραση του χρήστη με την υπηρεσία.

5. Πιλοτική Εφαρμογή

5.1 Περιγραφή Περιβάλλοντος ΙΚΔ

Στα πλαίσια της πιλοτικής υλοποίησης εφαρμογών, θεωρούμε ότι η υπηρεσία Unity έχει εγκατασταθεί και είναι προσαρμοσμένη στα πλαίσια της ακαδημαϊκής κοινότητας τους Χαροκοπείου Πανεπιστημίου. Χρήστες του ΙΚΔ μπορούν να γίνουν όλα τα μέλη της ακαδημαϊκής κοινότητας του Χαροκοπείου Πανεπιστημίου που διαθέτουν ηλεκτρονικό ταχυδρομείο (email) από το Πανεπιστήμιο της μορφής *@hua.gr. Από αυτή τη διεύθυνση προκύπτει και το μοναδικό αναγνωριστικό κάθε συμμετέχοντα, το οποίο δεν είναι άλλο από τη συμβολοσειρά αριστερά του συμβόλου «@». Σημειώνεται πως κατά την εγγραφή κάθε συμμετέχοντα, όπως επίσης και κατά τη διαδικασία εισόδου (login) του στην υπηρεσία γίνεται ταυτοποίηση των στοιχείων που εισάγει με την υπηρεσία LDAP του ιδρύματος.

Εφόσον πραγματοποιηθεί η είσοδος ενός χρήστη στην υπηρεσία, η πρόσβαση σε κάθε ιστοσελίδα του Unity καθορίζεται από δύο πολύ σημαντικές παραμέτρους: τον *θεατή* (*viewer*) και τον *κάτοχο* (*owner*), όπως αυτές έχουν οριστεί σε προηγούμενες ενότητες. Κάθε ιστοσελίδα που προσπελάζεται στα πλαίσια της κοινωνικής δικτύωσης ανήκει σε κάποιον συμμετέχοντα-κάτοχο. Οι συμμετέχοντες που την προσπελαίνουν καθίστανται αυτόματα θεατές με κατά κύριο λόγο περιορισμένα δικαιώματα προβολής. Οι δύο αυτοί παράμετροι ορίζονται στη συνεδρία (session) της επίσκεψης ενός χρήστη και «περνούν» από σελίδα σε σελίδα μέσω των διευθύνσεων URL τους.

Οι διαφορετικές ιστοσελίδες που μπορεί να επισκεφτεί ένας χρήστης της υπηρεσίας Unity είναι σε μεγάλο βαθμό αυτές που θα συναντά κανείς σε οποιοδήποτε ΙΚΔ. Συγκεκριμένα, οι διαθέσιμες, βασικές ιστοσελίδες είναι οι εξής:

- **Login**, δηλαδή η ιστοσελίδα εισόδου ενός μέλους στην υπηρεσία (Εικόνα 5.1)
- **Register**, η ιστοσελίδα εγγραφής νέου μέλους στο Unity (Εικόνα 5.2)
- **Home**, δηλαδή η αρχική σελίδα της υπηρεσίας
- **Profile**, η σελίδα προφίλ του χρήστη

- **Groups**, η βασική σελίδα παρουσίασης των εικονικών ομάδων (groups)
- **Group**, η ιστοσελίδα όπου απεικονίζεται μία εικονική ομάδα (group)
- **Apps**, η σελίδα επιλογής ειδικών εφαρμογών για εγκατάσταση (gadgets)
- **Search**, η σελίδα αναζήτησης χρηστών ή εικονικών ομάδων.



Εικόνα 5.1: Φόρμα ιστοσελίδας εισόδου στην υπηρεσία Unity.

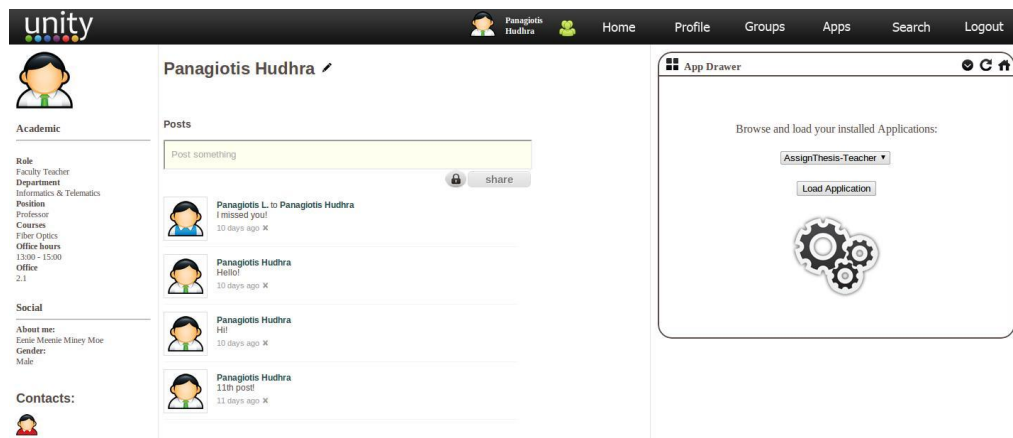
Εικόνα 5.2: Φόρμα ιστοσελίδας εγγραφής νέου χρήστη στο Unity.

Η αρχική ιστοσελίδα του Unity περιλαμβάνει μόνο μία κύρια ροή δραστηριοτήτων των επαφών του χρήστη που πραγματοποίησε login. Σημειώνεται πως σε αυτή τη

σελίδα ο θεατής είναι αυτόματα και ο κάτοχος, καθώς κανένας χρήστης δε μπορεί να επισκεφτεί την αρχική σελίδα κάποιου άλλου.

Η ιστοσελίδα του *προφίλ* ενός συμμετέχοντα είναι το σημείο προβολής της ψηφιακής του ταυτότητας. Όπως φαίνεται και στην Εικόνα 5.3, μέσα σε ένα προφίλ ένας χρήστης να συναντήσει:

- Την ακαδημαϊκή και κοινωνική πληροφορία για τον κάτοχο, όπως η φωτογραφία του, ο ακαδημαϊκός του ρόλος, τα ενδιαφέροντα του κλπ
- Τη σχέση που έχει με τον συγκεκριμένο κάτοχο ή τη δυνατότητα σύναψης σχέσης μαζί του, εάν φυσικά δεν είναι ο ίδιος κάτοχος του προφίλ
- Τις δημοσιεύσεις που έχει κάνει ο κάτοχος ή δημοσιεύσεις για τις οποίες ορίστηκε παραλήπτης, καθώς και ένα πλαίσιο στο οποίο μπορεί να συντάξει κανείς τη δική του δημοσίευση προς τον κάτοχο, εάν φυσικά έχει τα απαραίτητα δικαιώματα
- Μία λίστα εφαρμογών με το όνομα App Drawer (συρτάρι εφαρμογών), όπου φιλοξενούνται εφαρμογές που έχει εγκαταστήσει ο κάτοχος, καθώς και κενός χώρος για την προσπέλαση των αντίστοιχων gadgets.



Εικόνα 5.3: Ιστοσελίδα προφίλ ενός χρήστη στο Unity.

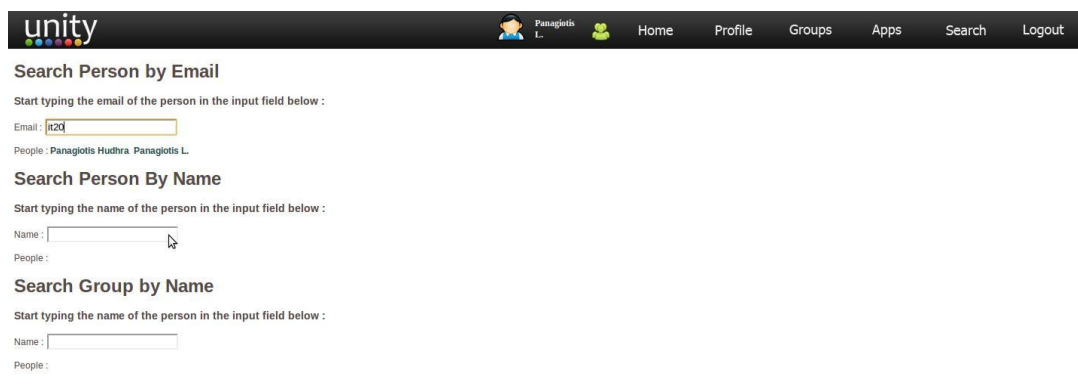
Στην ιστοσελίδα *Groups* γίνεται μία μικρή παρουσίαση του συνόλου των ενεργών εικονικών ομάδων του Unity. Οι εικονικές ομάδες χωρίζονται σε δύο κατηγορίες: ομάδες γενικού ενδιαφέροντος (fun groups) και ομάδες μαθημάτων (class groups). Στον επισκέπτη εμφανίζεται για κάθε ομάδα το όνομά της, ο διαχειριστής της και μία

περιγραφή για αυτή. Κάνοντας κλικ σε μία από αυτές, οδηγείται στην ιστοσελίδα της. Η σελίδα αυτή προσεγγίζει κατά πολύ, όσον αφορά το κομμάτι της εμφάνισης, το προφίλ ενός χρήστη, μόνο που εδώ ο θεατής συναντά:

- Πληροφορία που αφορά τη συγκεκριμένη ομάδα, όπως π.χ. τύπος, περιγραφή ομάδας και εάν ο ίδιος είναι μέλος σε αυτή. Αν όχι, στο κεντρικό μέρος εμφανίζεται η επιλογή εγγραφής στο group (join group), σε όσους συμμετέχοντες δεν είναι ήδη εγγεγραμμένοι.
- Μηχανισμό δημοσίευση και προβολής μηνυμάτων, όπως στο προφίλ χρήστη.
- Μία λίστα με τα μέλη της ομάδας
- Επιλογή για επεξεργασία της ομάδας ή διαγραφή μελών, εάν ο θεατής είναι και διαχειριστής της.

Η σελίδα *Apps* περιλαμβάνει όλες τις εφαρμογές που έχουν αναπτυχθεί για το Unity. Η σελίδα αποτελείται από δύο μέρη: εκείνο των διαθέσιμων εφαρμογών (Available Apps) και εκείνο των εγκατεστημένων εφαρμογών (App Drawer). Το πρώτο αναφέρεται στις εφαρμογές που έχει τη δυνατότητα να εγκαταστήσει ο χρήστης εξαιτίας του ακαδημαϊκού του ρόλου, ενώ το δεύτερο αναφέρεται στις εφαρμογές που έχει ήδη εγκαταστήσει ο χρήστης. Από αυτή την ιστοσελίδα μπορεί κανείς να εγκαταστήσει εφαρμογές, κάνοντας κλικ πάνω τους. Η εκτέλεσή της εφαρμογής όμως θα πραγματοποιηθεί στη σελίδα προφίλ του χρήστη.

Από το Unity δεν θα μπορούσε να απουσιάσει η λειτουργία αναζήτησης η οποία προσφέρεται στην σελίδα *Search*. Στο Unity υποστηρίζονται τρία διαφορετικά είδη αναζήτησης. Αναζήτηση χρήστη μέσω ονόματος, αναζήτηση χρήστη μέσω email και αναζήτηση εικονικών ομάδων μέσω ονόματος. Οι επιστρεφόμενοι χρήστες ή οι επιστρεφόμενες εικονικές ομάδες παίρνουν τη μορφή συνδέσμων που παραπέμπουν προς τις αντίστοιχες ιστοσελίδες. Η συγκεκριμένη ιστοσελίδα φαίνεται στην Εικόνα 5.4.



Εικόνα 5.4: Ιστοσελίδα αναζήτησης χρηστών και εικονικών ομάδων (Search)

5.2 Ανάπτυξη Πιλοτικών Εφαρμογών

Όπως έχει ήδη αναφερθεί, στόχος της παρούσας εργασίας είναι να εμπλουτιστεί η υπηρεσία Unity με εφαρμογές, στις οποίες θα αντικατοπτρίζονται οι υπηρεσίες που εμφανίζονται στο εσωτερικό μίας ακαδημαϊκής κοινότητας. Στα πλαίσια της πιλοτικής υλοποίησης, υπηρεσία ακαδημαϊκής κοινότητας έχει οριστεί οποιαδήποτε συναλλαγή μεταξύ ατόμων των τριών ευρείων ακαδημαϊκών ομάδων: φοιτητών, καθηγητών και μελών προσωπικού. Τονίζεται, όμως, ότι τελικοί αποδέκτες των υπηρεσιών αυτών θεωρούνται οι φοιτητές, στους οποίους άλλωστε απευθύνεται κατά κύριο λόγο ο ΙΚΔ. Επιπλέον, θεωρείται πως κάθε συναλλαγή είναι κατά κύριο λόγο ένα σύνολο από ενέργειες που διευκολύνουν την επικοινωνία μεταξύ των συμμετεχόντων για την τελική έγκριση χορήγησης κάποιου πόρου. Αυτό σημαίνει πως οι υπηρεσίες που επιλέγονται προς υλοποίηση δεν πραγματοποιούνται εξ ολοκλήρου ηλεκτρονικά. Αντιθέτως, απλώς αυτοματοποιούνται κάποιες διαδικασίες τους.

Στα πλαίσια μίας δοκιμαστικής υλοποίησης, επιλέχθηκαν προς πιλοτική ανάπτυξη δύο εφαρμογές που αντιστοιχούν σε βασικές υπηρεσίες μίας ακαδημαϊκής κοινότητας και που ικανοποιούν πλήρως τις παραπάνω προδιαγραφές: μία *εφαρμογή για την αίτηση χορήγησης βεβαίωσης φοίτησης* και μία *εφαρμογή για την ανάθεση πτυχιακής εργασίας*.

5.2.1 Εφαρμογή για την αίτηση χορήγησης βεβαίωσης φοίτησης

5.2.1.1 Γενική Περιγραφή Υπηρεσίας

Στις γραμματείες των Πανεπιστημιακών Ιδρυμάτων τυχαίνει αρκετά συχνά να γίνονται αιτήσεις για την χορήγηση βεβαίωσης φοίτησης σε φοιτητές. Οι συναλλασσόμενοι σε μία τέτοια υπηρεσία δεν είναι άλλοι από ενδιαφερόμενους φοιτητές και γραμματείς (δηλαδή μέλη του προσωπικού). Η διαδικασία γενικά απαιτεί από τους φοιτητές να επισκεφτούν τη γραμματεία του Τμήματος στο οποίο φοιτούν, να καταβάλλουν το αίτημα τους για να τους χορηγηθεί η βεβαίωση και ύστερα να επισκέπτονται τη γραμματεία συστηματικά για να ελέγχουν αν η βεβαίωση φοίτησης τους έχει εκδοθεί.

Με τη βοήθεια των εφαρμογών στην υπηρεσία Unity, μία τέτοια υπηρεσία μπορεί πλέον να παρέχεται ηλεκτρονικά, γλιτώνοντας από αρκετό κόπο τους φοιτητές και οργανώνοντας καλύτερα τα καθήκοντα της γραμματείας. Ο γραμματέας σε αυτή την περίπτωση φυσικά προσφέρει την υπηρεσία και το μόνο που απαιτείται από την πλευρά του φοιτητή είναι η δήλωση ότι θα συνεργαστεί με τον πάροχο, η οποία μεταφράζεται στη σύναψη της αντίστοιχης μονόδρομης σχέσης *facilitator*.

5.2.1.2 Σενάριο Χρήσης

Έστω δύο συμμετέχοντες που χρησιμοποιούν την εν λόγω εφαρμογή: ένας φοιτητής και ένας γραμματέας. Επιπλέον, θεωρείται πως στα πλαίσια της αλληλεπίδρασης τους μέσω αυτής υπάρχει η αντίστοιχη σχέση *facilitator* που απαιτείται. Ένα σενάριο χρήσης μίας τέτοιας εφαρμογής έχει ως εξής:

1. Ο φοιτητής επισκέπτεται την ιστοσελίδα Apps του Unity και πλοηγείται στις διαθέσιμες εφαρμογές που μπορεί να εγκαταστήσει.

2. Επιλέγει την εφαρμογή Studies Certificate App και την προσθέτει στη λίστα εφαρμογών του (App Drawer).
3. Επισκέπτεται το προφίλ του και προσπελαύνει την εφαρμογή που μόλις εγκατέστησε.
4. Από την οθόνη της εφαρμογής, αιτείται την χορήγηση μιας βεβαίωσης. Το αίτημα του προωθείται στους γραμματείς τους οποίους ακολουθεί.
5. Ένας εξ αυτών επισκέπτεται την εφαρμογή και παρατηρεί ότι υπάρχουν διαθέσιμα αιτήματα φοιτητών προς εξυπηρέτηση.
6. Εξυπηρετεί τα διαθέσιμα αιτήματα και ενημερώνει για την έκδοση των βεβαιώσεων τον φοιτητή.
7. Ο φοιτητής αποδέχεται την ενημέρωση και, εάν το επιθυμεί, επαναχρησιμοποιεί την εφαρμογή.

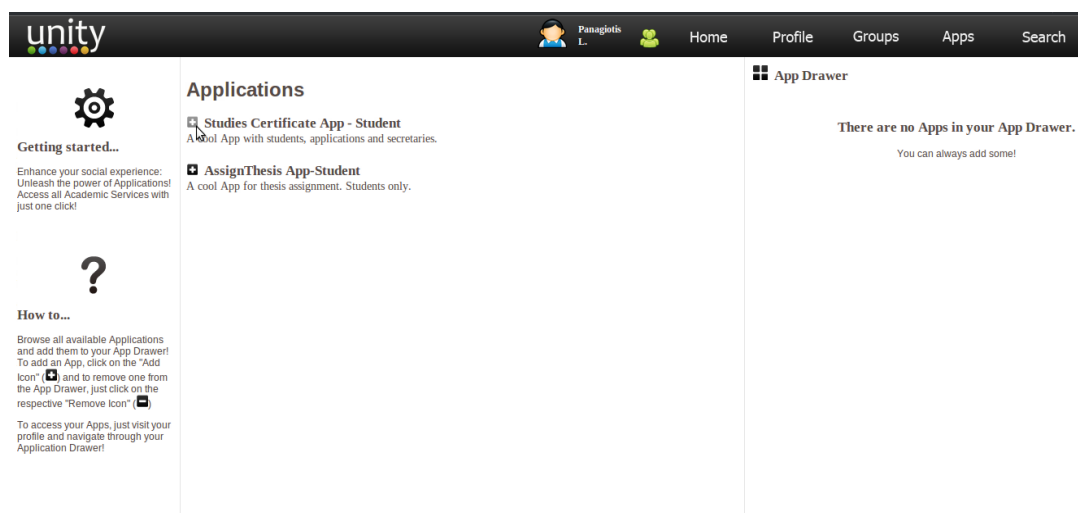
5.2.1.3 Υλοποίηση

Η υλοποίηση μίας εφαρμογής, όπως ορίστηκε παραπάνω, ξεκινάει από την απόφαση σχετικά με τον αριθμό των όψεων της. Οι προδιαγραφές της εφαρμογής καταδεικνύουν την ύπαρξη δύο ειδών συμμετεχόντων, οπότε οι διαφορετικές όψεις που θα πρέπει να έχει αυτή είναι επίσης δύο, μία για καθέναν από αυτούς. Συνεπώς, δύο είναι και τα gadgets που θα δημιουργηθούν, ώστε καθένα από αυτά να λειτουργεί για συγκεκριμένους συμμετέχοντες.

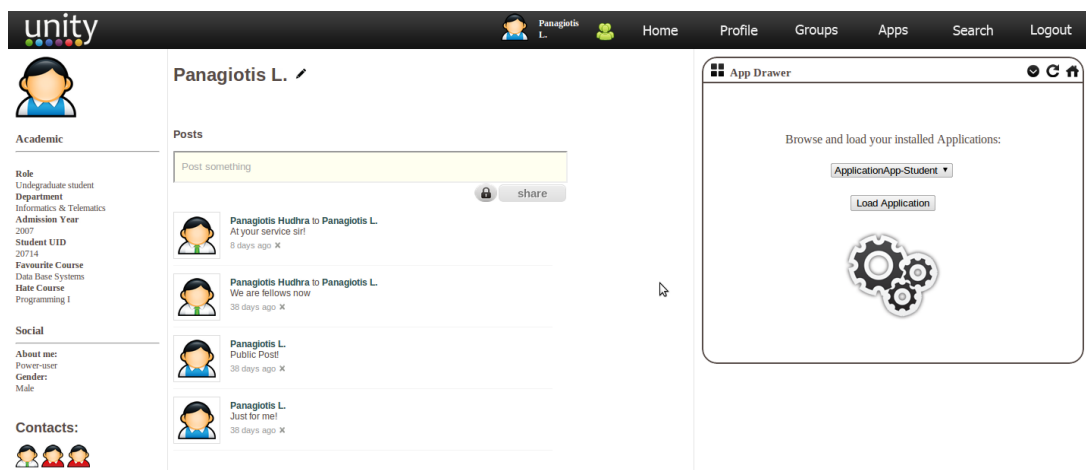
Το gadget του φοιτητή θα είναι εκείνο που θα διατηρεί έναν αποθηκευτικό χώρο, ώστε να αποθηκεύεται κάθε φορά ο αριθμός των αιτημάτων του. Συνεπώς διατηρούνται AppData για το gadget αυτό, τα οποία προσπελαύνουν αμφότερες οι

δύο εφαρμογές, πραγματοποιώντας σε αυτά τροποποιήσεις. Παράλληλα, για την ενημέρωση των συμμετεχόντων σχετικά με αλλαγές στη λειτουργία των gadgets εξαιτίας κάποιων γεγονότων, παράγονται οι αντίστοιχες δραστηριότητες (activities) που προβάλλονται στις αρχικές σελίδες και των δύο.

Πριν αναλυθεί η λειτουργία των gadgets, εννοείται πως οι χρήστες έχουν προβεί στις κατάλληλες ενέργειες για την εγκατάστασή τους. Αυτό σημαίνει πως αμφότεροι έχουν επισκεφτεί την ιστοσελίδα Apps που φαίνεται στην εικόνα 5.5 και έχουν προσθέσει στις συλλογές εφαρμογών τους (App Drawers) την συγκεκριμένη. Τα AppData του gadget του φοιτητή τότε αρχικοποιούνται και αυτό μαζί με το άλλο γίνονται διαθέσιμα μέσα από τις ιστοσελίδες των προφίλ των συμμετεχόντων. Ύστερα από την ολοκλήρωση της εγκατάστασης, ο φοιτητής πλέον είναι έτοιμος να προσπελάσει το gadget από το προφίλ του, μέσω του «παραθύρου» που φαίνεται στην Εικόνα 5.6. Στη συνέχεια, έχει τη δυνατότητα να ελαχιστοποιήσει το παράθυρο, να ανανεώσει το περιεχόμενο του ή να γυρίσει στο αρχικό μενού του για να επιλέξει ένα άλλο gadget προς προβολή.

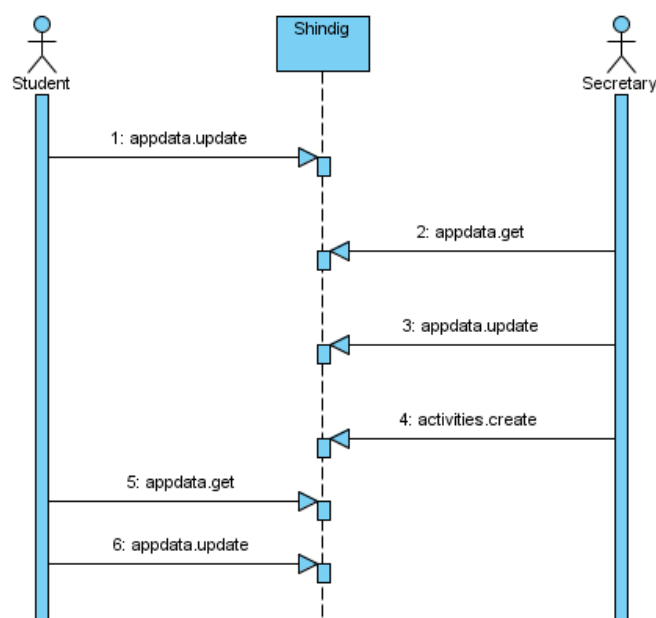


Εικόνα 5.5: Ιστοσελίδα Εγκατάστασης Εφαρμογών (Apps)



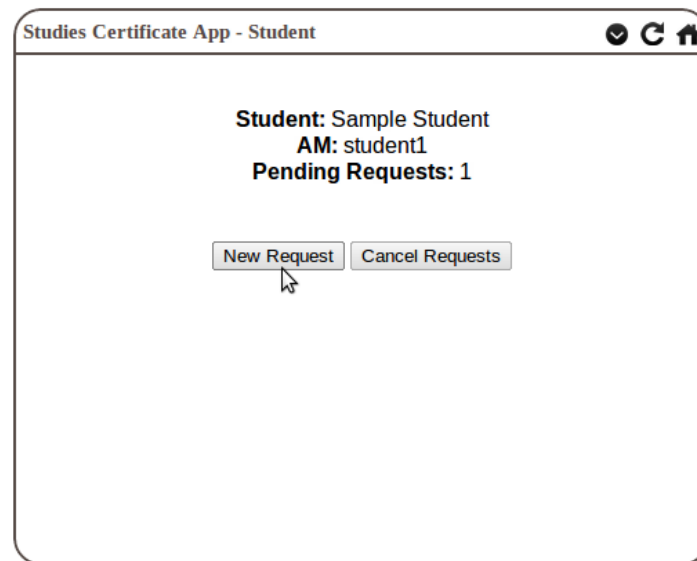
Εικόνα 5.6: Προφίλ χρήστη και «παράθυρο» φόρτωσης gadgets στα δεξιά της οθόνης

Εφόσον προσπελαστεί το gadget για την αίτηση χορήγησης βεβαίωσης σπουδών (εμφανιζόμενο με το όνομα “Studies Certificate App – Student”), ξεκινάει η αλληλεπίδραση με το Shindig. Η αλληλουχία των κλήσεων που πραγματοποιεί κάθε συμμετέχων μέσω της διάδρασης με το gadget του παρουσιάζεται στο διάγραμμα της Εικόνας 5.7, το οποίο κατόπιν αναλύεται εκτενέστερα.



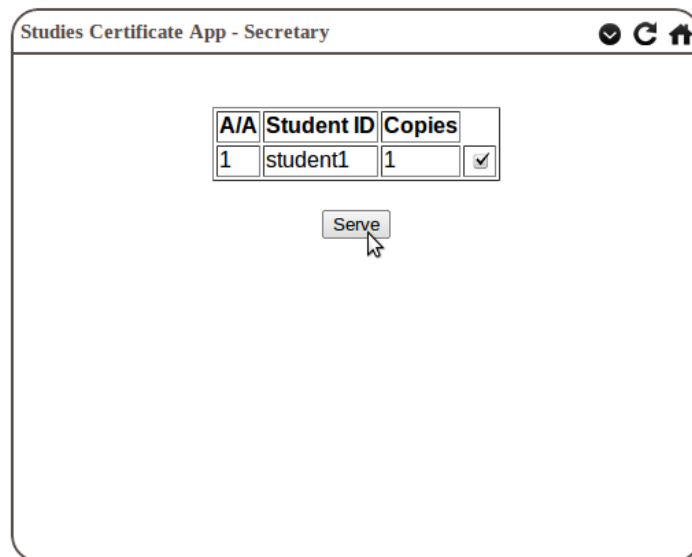
Εικόνα 5.7: Ακολουθιακό διάγραμμα κλήσεων εφαρμογής για την αίτηση χορήγησης βεβαίωσης σπουδών

Το βήμα 1 του διαγράμματος αντιστοιχεί στην κλήση που πραγματοποιείται για την καταχώρηση μιας αίτησης. Επικεντρώνοντας στο παράθυρο φόρτωσης των gadgets, ο φοιτητής έχει επισκεφτεί το gadget και στην αρχική του οθόνη επιλέγει την πραγματοποίηση αιτήματος, όπως φαίνεται στην Εικόνα 5.8. Τότε, πραγματοποιείται μία κλήση μέσω της εντολής `osapi.appdata.update`, η οποία τροποποιεί την τιμή των εκκρεμούντων αιτημάτων του φοιτητή, ώστε να αυξηθούν κατά ένα.



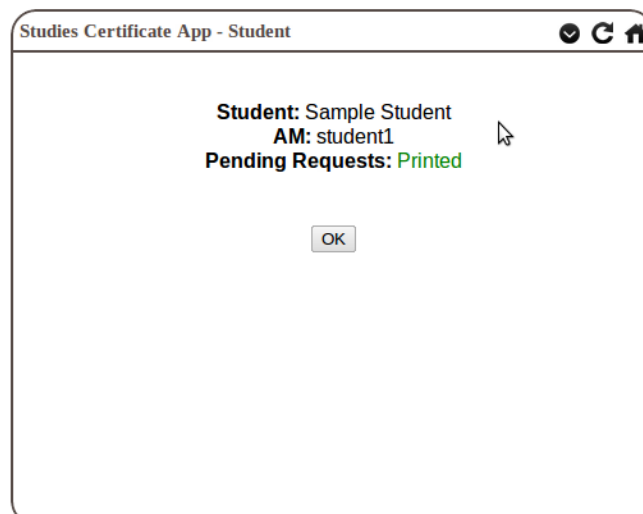
Εικόνα 5.8: Οθόνη πραγματοποίησης αιτήματος φοιτητή για χορήγηση βεβαίωσης σπουδών (gadget φοιτητή)

Στη συνέχεια, περνώντας στο gadget του γραμματέα (Studies Certificate App - Staff), λαμβάνονται τα δεδομένα εφαρμογών όλων των φοιτητών που τον ακολουθούν με την εντολή `osapi.appdata.get` και κατασκευάζεται μία λίστα με τα εκκρεμούντα, προς εξυπηρέτηση αιτήματα (βήμα 2). Ακολουθώντας το παραπάνω παράδειγμα, η λίστα αυτή θα έχει τη μορφή που φαίνεται στην Εικόνα 5.9. Στη συνέχεια, εάν ο γραμματέας εξυπηρετήσει τα αιτήματα ενός φοιτητή, πραγματοποιεί την κλήση `osapi.appdata.update` και τροποποιεί τα δεδομένα του φοιτητή με κάποια συγκεκριμένη τιμή-κλειδί (βήμα 3). Τέλος, δημιουργεί και δημοσιεύει την δραστηριότητα που αναφέρεται στην εξυπηρέτηση των αιτημάτων του φοιτητή (βήμα 3), μέσω της οποίας γνωστοποιείται και στον φοιτητή έμμεσα η πρόοδος αυτών.



Εικόνα 5.9: Οθόνη εξυπηρέτησης αιτημάτων φοιτητών (gadget γραμματέα)

Επιστρέφοντας τώρα στο gadget του φοιτητή, τα δεδομένα ελέγχονται τακτικά μέσω της κλήσης `osapi.appdata.get` για τυχόν αλλαγές. Εφόσον γίνει η εξυπηρέτηση των αιτημάτων από το γραμματέα, το gadget του φοιτητή διαβάζει τη νέα τιμή-κλειδί και εμφανίζει το αντίστοιχο μήνυμα ενημερώνει τον ίδιο για την πραγματοποίηση του αιτήματος του (βήμα 5), όπως φαίνεται στην Εικόνα 5.10. Το τελικό βήμα 6 πραγματοποιείται εφόσον ο φοιτητής επιθυμήσει να ξαναχρησιμοποιήσει την εφαρμογή. Τότε πραγματοποιείται μία κλήση `osapi.appdata.update`, η οποία ορίζει τον αριθμό των αιτημάτων στο μηδέν και η όλη διαδικασία μπορεί να επαναληφθεί από την αρχή.



Εικόνα 5.10: Ενημέρωση χρήστη για την ολοκλήρωση του αιτήματος του.

5.2.2 Εφαρμογή Ανάθεσης Πτυχιακής Εργασίας

5.2.2.1 Γενική Περιγραφή

Μία ακόμα υπηρεσία που απαιτεί σε σημαντικό βαθμό αυτοματοποίηση διαδικασιών στα πλαίσια μίας ακαδημαϊκής κοινότητας είναι εκείνη της ανάθεσης πτυχιακής εργασίας σε κάποιο φοιτητή. Για την υπηρεσία Unity, μία αντίστοιχη εφαρμογή είναι αρκετά πιο απαιτητική από την προηγούμενη, τόσο από άποψη αριθμού συμμετεχόντων που τη χρησιμοποιούν, όσο και από άποψη πολυπλοκότητας κλήσεων και αλληλεπιδράσεων μεταξύ των gadgets που την απαρτίζουν.

Οι συμμετέχοντες σε αυτή την εφαρμογή είναι τρεις: ένας καθηγητής που ανακοινώνει και αναθέτει μία πτυχιακή εργασία, ένας φοιτητής που αιτείται την ανάθεση αυτής και ένας γραμματέας (δηλαδή μέλος προσωπικού) ο οποίος επιβλέπει και επικυρώνει την όλη διαδικασία. Για να είναι δυνατή η αλληλεπίδραση όλων των παραπάνω, πρέπει να υπάρχουν φυσικά και οι αντίστοιχες σχέσεις στα πλαίσια που ορίζει το Unity. Ο φοιτητής που προσπελαύνει την εφαρμογή πρέπει να έχει συνάψει σχέση *facilitator* με τον γραμματέα, όπως επίσης να έχει συνάψει και τη σχέση *tutor* με τον καθηγητή που τη χρησιμοποιεί. Εάν μία από τις δύο δεν υφίσταται, η διαδικασία ανάθεσης πτυχιακής εργασίας αποτυγχάνει.

5.2.2.2 Σενάριο Χρήσης

Έστω ότι υπάρχουν οι απαραίτητοι συμμετέχοντες και οι απαιτούμενες σχέσεις μεταξύ τους για την χρήση της εφαρμογής αυτής. Δεδομένου ότι τα αντίστοιχα gadgets έχουν εγκατασταθεί και προσπελαστεί, ένα σενάριο χρήσης της εφαρμογής είναι το εξής:

1. Ο καθηγητής μέσα από το gadget του ορίζει ένα νέο θέμα πτυχιακής εργασίας και συμπληρώνει την περιγραφή του ή επεξεργάζεται τα στοιχεία ενός ήδη υπάρχοντος.
2. Ο φοιτητής ενημερώνεται για τα νέα θέματα πτυχιακών εργασιών που έχουν ανακοινωθεί.
3. Ο φοιτητής επιλέγει ένα από τα θέματα πτυχιακών (έστω εκείνο του προαναφερθέντα καθηγητή) και αιτείται την ανάθεση του σε αυτόν.
4. Ο καθηγητής ανακτά και εξετάζει όλα τα αιτήματα των φοιτητών για το συγκεκριμένο θέμα.
5. Ο καθηγητής επιλέγει τον φοιτητή στον οποίο θα ανατεθεί το θέμα και στη συνέχεια ορίζει την τριμελή επιτροπή καθηγητών που θα εξετάσει την πτυχιακή, κατοχυρώνοντας τους με τις e-mail διευθύνσεις τους.
6. Ο φοιτητής ενημερώνεται για τις αλλαγές στα δεδομένα του gadget του καθηγητή και αιτείται την έγκριση της ανάθεσης από τον γραμματέα τον οποίο ακολουθεί.
7. Ο γραμματέας συγκεντρώνει όλες τις αιτήσεις για έγκριση ανάθεσης πτυχιακών εργασιών σε φοιτητές.
8. Ο γραμματέας εγκρίνει την ανάθεση της πτυχιακής εργασίας του φοιτητή του βήματος 6.
9. Ο φοιτητής ενημερώνεται και πλέον χάνει τη δυνατότητα χρήσης των επιλογών του gadget.

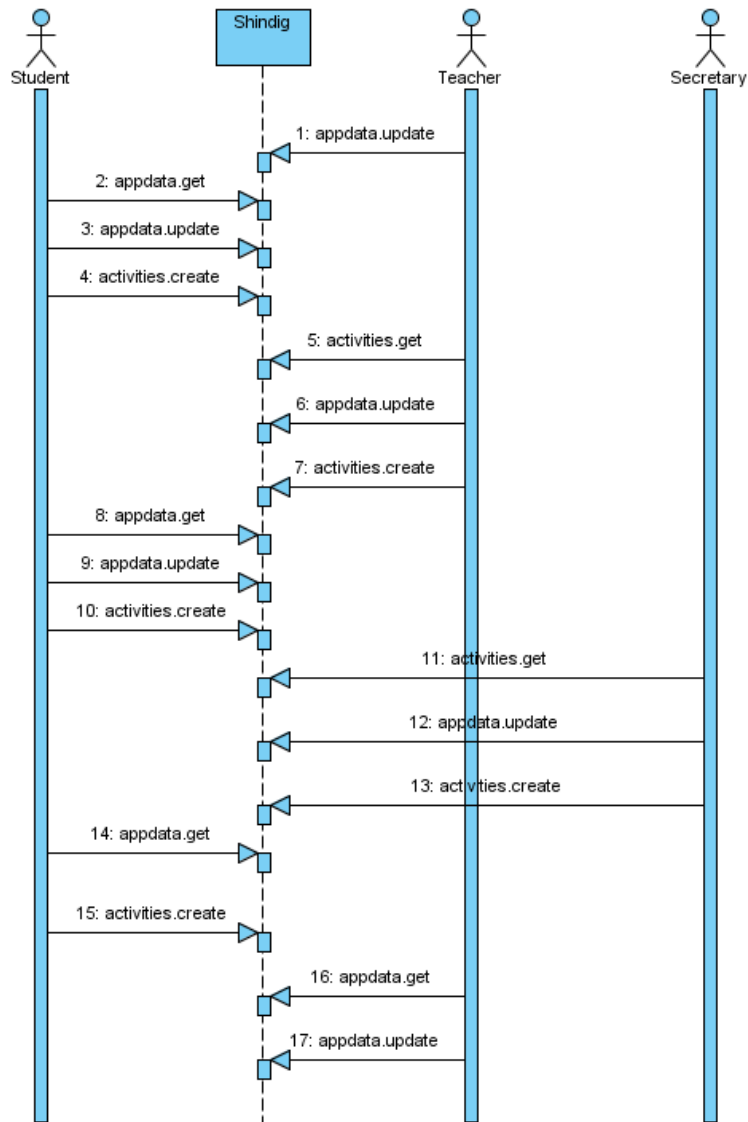
10. Ο καθηγητής επίσης ενημερώνεται για την ανάθεση και πλέον μπορεί να επαναφέρει το gadget του στην αρχική του κατάσταση, ώστε να το επαναχρησιμοποιήσει.

5.2.2.3 Υλοποίηση

Για να γίνει η υλοποίηση της εφαρμογής που να καθιστά πραγματοποιήσιμο το παραπάνω σενάριο, απαιτείται αρχικά ο καθορισμός του αριθμού των gadgets που θα την απαρτίζουν. Έτσι, εφόσον οι όψεις της εφαρμογής είναι τρεις, όσοι και οι συμμετέχοντες, τόσα θα είναι και τα gadgets, στα οποία δόθηκαν τα εξής ονόματα: Thesis Assignment App-Student (gadget φοιτητή), Thesis Assignment App-Staff (gadget γραμματέα), Thesis Assignment App-Teacher (gadget καθηγητή).

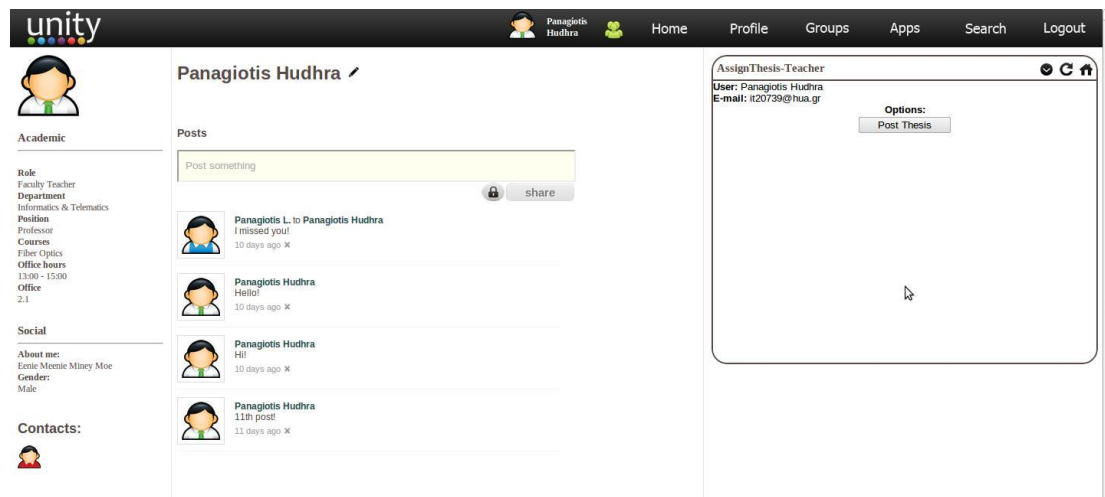
Από τα παραπάνω, τα gadgets που αποθηκεύουν κάποια πληροφορία είναι δύο: εκείνο του καθηγητή και εκείνο του φοιτητή. Το gadget του καθηγητή αποθηκεύει τον τίτλο της πτυχιακής εργασίας, την περιγραφή της, τον φοιτητή στον οποίον ανατίθεται και την τριμελή επιτροπή καθηγητών (εκ των οποίων ο ένας είναι ο κάτοχος της εφαρμογής) που την εξετάζει, ενώ το gadget του φοιτητή αποθηκεύει την κατάσταση του αιτήματος του και τον τίτλο της πτυχιακής για την οποία έχει εκδηλώσει ενδιαφέρον.

Περνώντας στο χαμηλό επίπεδο υλοποίησης τους, εννοείται δεδομένη η εγκατάσταση των παραπάνω gadgets και η προσπέλαση τους από τους συμμετέχοντες. Εφόσον έχει πραγματοποιηθεί αυτό το βήμα, οι υπόλοιπες «εσωτερικές» τους λειτουργίες ανατίθενται στο Shindig. Πιο συγκεκριμένα, η αλληλουχία των κλήσεων που πραγματοποιούν παρουσιάζεται στο διάγραμμα της Εικόνας 5.11.

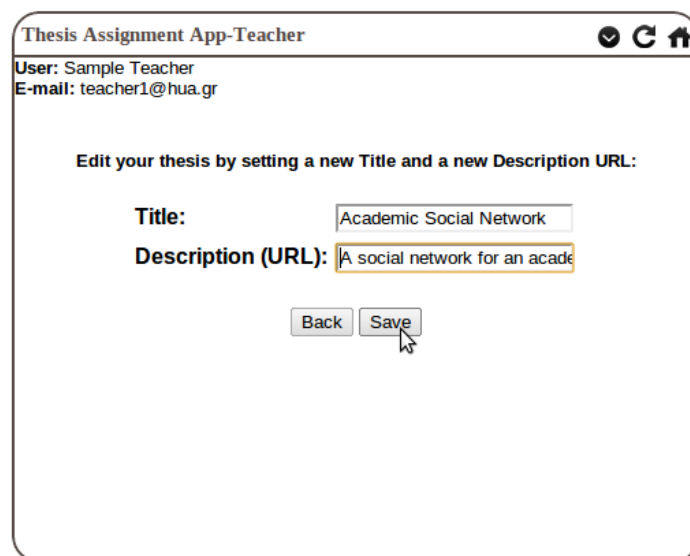


Εικόνα 5.11: Ακολουθιακό διάγραμμα κλήσεων εφαρμογής ανάθεσης πτυχιακής εργασίας

Καταρχήν, η αλληλουχία των διαδικασιών ξεκινά με την επίσκεψη του καθηγητή στο αντίστοιχο του gadget, όπου και θα του εμφανιστεί η αρχική οθόνη που φαίνεται στην Εικόνα 5.12. Σύμφωνα με το βήμα 1 του διαγράμματος, ο καθηγητής ορίζει ένα νέο θέμα και περιγραφή πτυχιακής εργασίας. Σε χαμηλότερο επίπεδο, με αυτή την ενέργεια πραγματοποιείται η κλήση `osapi.appdata.update`, η οποία τροποποιεί τα αντίστοιχα δεδομένα που διατηρεί το gadget. Εστιάζοντας τώρα στο παράθυρο του gadget, η συγκεκριμένη οθόνη εμφανίζεται στην Εικόνα 5.13.



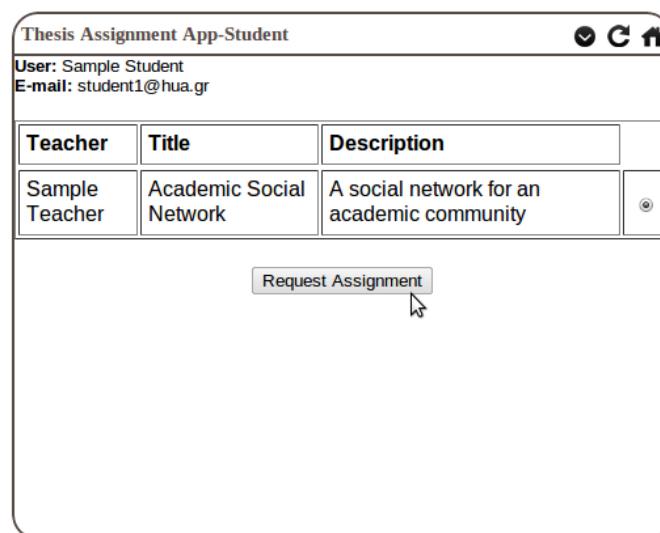
Εικόνα 5.12: Αρχική οθόνη gadget καθηγητή για την ανάθεση πτυχιακής εργασίας, όπως αυτό έχει ενσωματωθεί στην ιστοσελίδα του προφίλ του



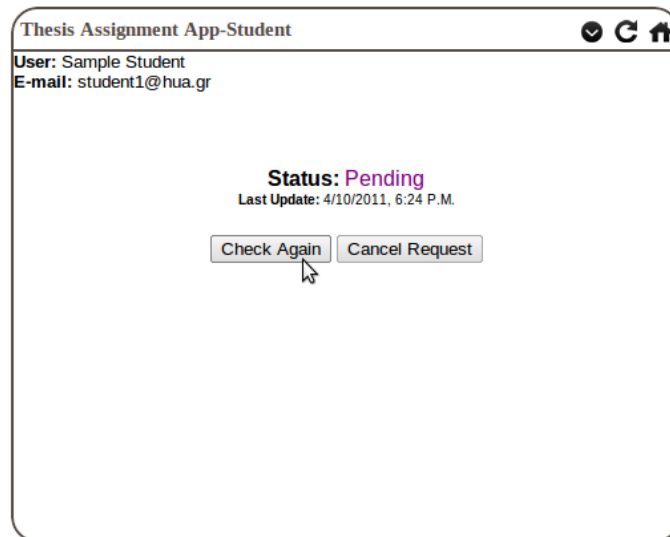
Εικόνα 5.13: Ορισμός νέου/Επεξεργασία υπάρχοντος θέματος πτυχιακής εργασίας (gadget καθηγητή)

Στη συνέχεια, το gadget του φοιτητή στο βήμα 2 πραγματοποιεί την κλήση `osapi.appdata.get` για τους tutors του φοιτητή, έτσι ώστε να συγκεντρώσει όλα τα δεδομένα από gadgets καθηγητών σχετικά με την ανακοίνωση θέματος πτυχιακής εργασίας. Στην οθόνη του φοιτητή εμφανίζεται μία φόρμα επιλογής με όλα τα θέματα που ανακοινώθηκαν και είναι ακόμα διαθέσιμα, όπως φαίνεται στην Εικόνα 5.14. Ο

φοιτητής επιλέγει ένα από αυτά, το οποίο επιθυμεί να του ανατεθεί και μεταφέρεται σε μία οθόνη αναμονής της επιβεβαίωσης της ανάθεσης (Εικόνα 5.15), όπου του δίνεται η ευκαιρία να ακυρώσει το αίτημα του, εάν το επιθυμεί. Τη στιγμή που πραγματοποιείται αυτή η αλλαγή, πραγματοποιείται και η κλήση `osapi.appdata.update` (βήμα 3), με την οποία τροποποιούνται τα δεδομένα του gadget του φοιτητή ώστε να λάβουν σαν τιμή το τίτλο της πτυχιακής εργασίας για την οποία εκδηλώνει ενδιαφέρον και την κατάσταση στην οποία βρίσκεται η αίτηση του (για την ώρα σε αναμονή). Ύστερα, στο βήμα 4 πραγματοποιείται η κλήση `osapi.activities.create`, με την οποία δημιουργείται η αντίστοιχη δραστηριότητα για την εκδήλωση ενδιαφέροντος, ώστε να ενημερωθεί έμμεσα ο καθηγητής.



Εικόνα 5.14: Οθόνη επιλογής θέματος πτυχιακής εργασίας (gadget φοιτητή)



Εικόνα 5.15: Οθόνη αναμονής για την επιβεβαίωση ανάθεσης πτυχιακής εργασίας (gadget φοιτητή)

Κατόπιν αυτών, μέσα από το gadget του, ο καθηγητής ενημερώνεται στο βήμα 5 για της αιτήσεις ανάθεσης με τη βοήθεια της κλήσης `osapi.activities.get` για τις δραστηριότητες των φοιτητών του (students). Με αυτό τον τρόπο, δημιουργείται μία λίστα των ενδιαφερόμενων φοιτητών, όπως εκείνη της Εικόνας 5.16 και ο καθηγητής επιλέγει τον φοιτητή στον οποίο αναθέτει την πτυχιακή εργασία που έχει ανακοινώσει. Στη συνέχεια, οδηγείται σε μία νέα οθόνη (Εικόνα 5.17) όπου του ζητείται να εισάγει τα αναγνωριστικά (e-mails) των άλλων δύο καθηγητών που θα συμμετάσχουν στην επιτροπή εξέτασης της πτυχιακής εργασίας. Εφόσον εισαχθούν και αυτά τα στοιχεία και ελεγχθεί η εγκυρότητα τους, πραγματοποιείται η κλήση `osapi.appdata.update` (βήμα 6), με την οποία τα δεδομένα που διατηρεί το gadget του καθηγητή αναλαμβάνουν τιμές που αφορούν τον φοιτητή στον οποίο ανατέθηκε η πτυχιακή εργασία και η τριμελής επιτροπή εξέτασης αυτής. Ταυτόχρονα, η ίδια κλήση πραγματοποιείται για να ενημερωθεί η κατάσταση της αίτησης στο gadget του φοιτητή. Τέλος, στο βήμα 7, δημιουργείται η αντίστοιχη δραστηριότητα που αναφέρεται στις παραπάνω αλλαγές με την κλήση `osapi.activities.create`. Ο καθηγητής τώρα δε μπορεί πλέον να εμφανίσει νέες αιτήσεις φοιτητών μέχρι να επικυρωθεί η συγκεκριμένη.

The screenshot shows a web application window titled "Thesis Assignment App-Teacher". At the top, it displays the user information: "User: Sample Teacher" and "E-mail: teacher1@hua.gr". Below this is a table with three columns: "ID", "Student", and "Title". The table contains one row with the values "student1", "Sample Student", and "Academic Social Network". To the right of the table is a small circular icon. Below the table is a "Select" button with a mouse cursor pointing at it.

ID	Student	Title
student1	Sample Student	Academic Social Network

Select

Εικόνα 5.16: Οθόνη επιλογής ενδιαφερόμενου φοιτητή για ανάθεση εργασίας (gadget καθηγητή)

The screenshot shows the same web application window. Below the user information, it says "Please, set two thesis supervisors (filling in their *@hua.gr e-mail addresses):". There are two input fields labeled "Supervisor #1:" and "Supervisor #2:". Below these fields is a "Submit" button with a mouse cursor pointing at it.

Please, set two thesis supervisors (filling in their *@hua.gr e-mail addresses):

Supervisor #1:

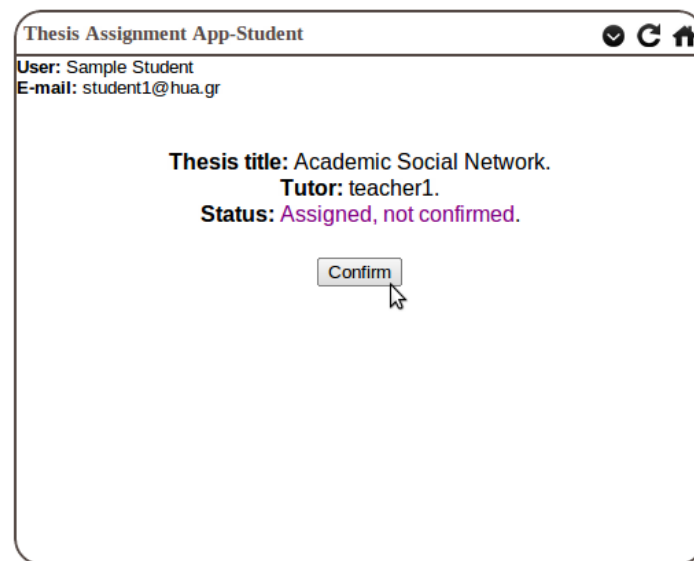
Supervisor #2:

Submit

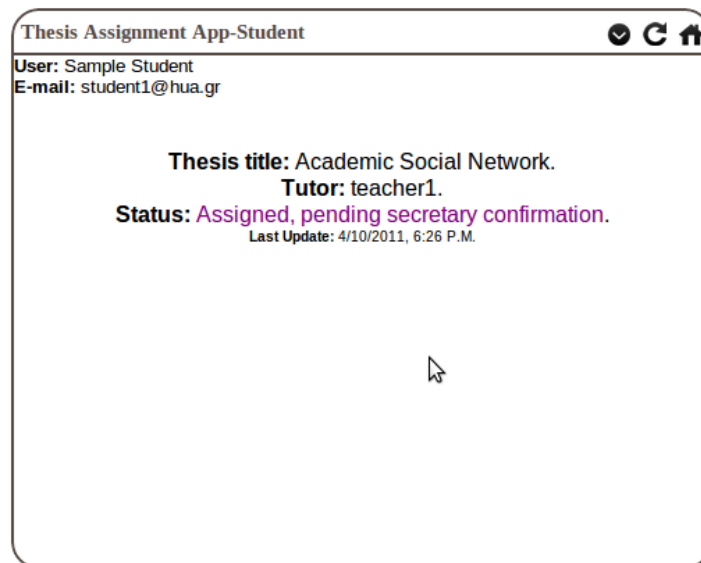
Εικόνα 5.17: Οθόνη επιλογής δύο καθηγητών για σύσταση τριμελούς επιτροπής (gadget καθηγητή)

Σ το βήμα 8, ο φοιτητής προσπελαίνει με τη σειρά του το δικό του gadget και παρατηρεί ότι του ανατέθηκε πτυχιακή εργασία, εφόσον η κλήση `osapi.appdata.get` για τα δεδομένα της εφαρμογής του επέστρεψε την αντίστοιχη απάντηση. Η νέα

κατάσταση του gadget αποτυπώνεται στην οθόνη της Εικόνας 5.18. Ο φοιτητής πλέον μπορεί να προχωρήσει στο αίτημα για επικύρωση της ανάθεσης που του έγινε, κάνοντας την αντίστοιχη επιλογή στο βήμα 9. Με αυτήν, πραγματοποιείται η κλήση `osapi.appdata.update` που ενημερώνει τα δεδομένα του. Ακόμα, το gadget εκπέμπει τη δραστηριότητα που προέκυψε από αυτή τη νέα κατάσταση με την κλήση `osapi.activities.create` στο βήμα 10. Κατόπιν των δύο τελευταίων βημάτων, η οθόνη του gadget πληροφορεί τον χρήστη ότι το αίτημα του μεταβιβάστηκε στους γραμματείς που ακολουθεί και αναμένει εξυπηρέτηση (Εικόνα 5.19).

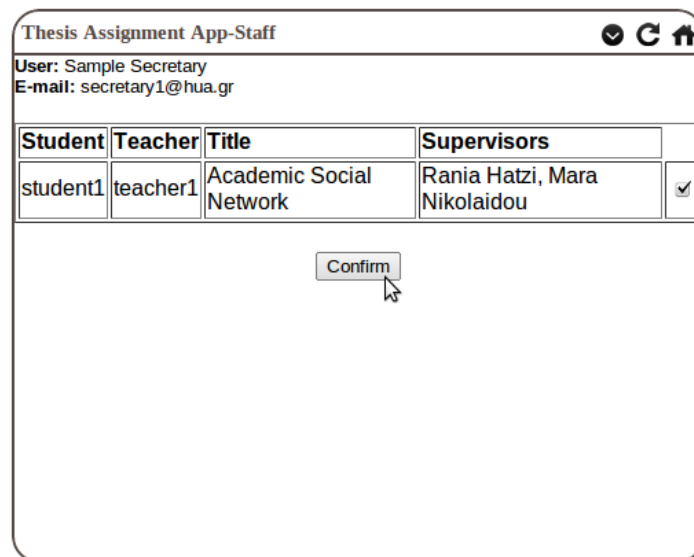


Εικόνα 5.18: Οθόνη για πραγματοποίηση αιτήματος για την επικύρωση ανάθεσης
πτυχιακής εργασίας (gadget φοιτητή)



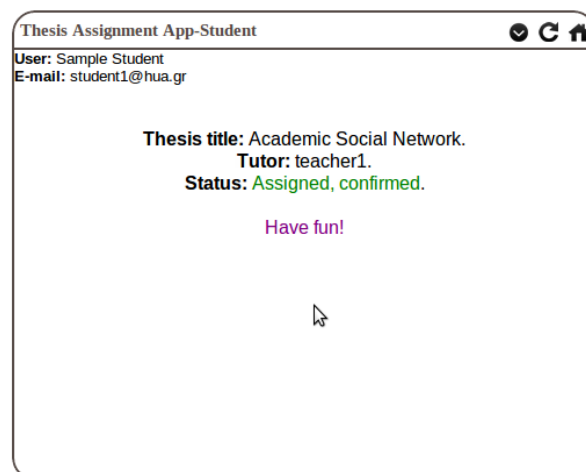
Εικόνα 5.19: Οθόνη αναμονής για την επικύρωση ανάθεσης πτυχιακής εργασίας (gadget φοιτητή)

Σε αυτό το σημείο εμφανίζεται στο προσκήνιο και το gadget του γραμματέα, του οποίου η λειτουργία είναι περισσότερο απλοϊκή. Όπως φαίνεται στο βήμα 11 του διαγράμματος, αρχικά το gadget αυτό συγκεντρώνει όλες τις δραστηριότητες που έχουν δημοσιευτεί από gadgets φοιτητών για την ανάθεση θεμάτων πτυχιακής εργασίας με τη βοήθεια της κλήσης `osapi.activities.get`. Έστερα, ο γραμματέας ελέγχει τις παραμέτρους κάθε ανάθεσης και την επικυρώνει (Εικόνα 5.20), εγγράφοντας την επιλογή του στα δεδομένα των gadgets των φοιτητών με την κλήση `osapi.appdata.update` (βήμα 12). Στο τέλος, παράγει δραστηριότητα που αναφέρεται σε αυτή την αλλαγή με την κλήση `osapi.activities.create` (βήμα 13) και με αυτή ολοκληρώνει τη λειτουργία του gadget.



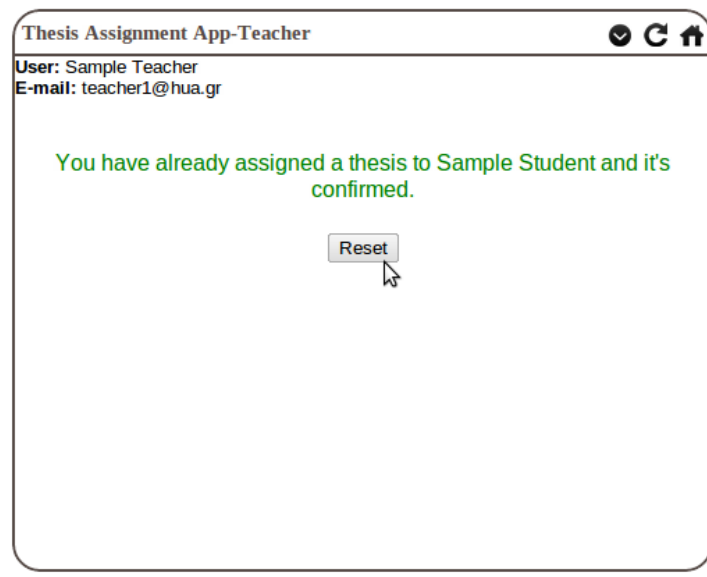
Εικόνα 5.20: Οθόνη επικύρωσης ανάθεσης πτυχιακών εργασιών (gadget γραμματέα)

Στο βήμα 14, ο φοιτητής ενημερώνεται στο gadget του για την ολοκλήρωση της ανάθεσης της πτυχιακής του εργασίας, με τη χρήση της κλήσης `osapi.appdata.get`. Ταυτόχρονα, στο βήμα 15 παράγεται η δραστηριότητα που περιλαμβάνει αυτή την πληροφορία, με τη χρήση της κλήσης `osapi.activities.create`. Το gadget του φοιτητή πλέον δε μπορεί να χρησιμοποιηθεί ξανά, καθώς έχει φτάσει στην τελική οθόνη που φαίνεται στην Εικόνα 5.21, η οποία δεν διαθέτει επιλογές.



Εικόνα 5.21: Τελική οθόνη gadget φοιτητή – Ολοκλήρωση ανάθεσης πτυχιακής εργασίας

Ο συμμετέχων καθηγητής ενημερώνεται στο βήμα 16 για την αλλαγή στα appdata του φοιτητή με την αντίστοιχη κλήση `osapi.appdata.get` (Εικόνα 5.22) και πλέον μπορεί να επαναχρησιμοποιήσει την εφαρμογή του, κάνοντας μία επαναφορά των δεδομένων στην αρχική τους κατάσταση με την βοήθεια της κλήσης `osapi.appdata.update` στο βήμα 17. Οι λειτουργίες της εφαρμογής έχουν πλέον ολοκληρωθεί.



Εικόνα 5.22: Οθόνη επαναφοράς αρχικών ρυθμίσεων του gadget του καθηγητή.

6. Επισκόπηση

6.1 Συμπεράσματα

Η κοινωνική δικτύωση είναι μία τεράστια τάση στο διαδίκτυο σήμερα. Στα λίγα χρόνια ύπαρξής της κατάφερε να προσελκύσει ένα πολύ μεγάλο ποσοστό του διαδικτυακού κοινού. Η δυναμική αυτή της κοινωνικής δικτύωσης μπορεί να χρησιμοποιηθεί και από την ακαδημαϊκή κοινότητα. Ο καλύτερος τρόπος για να πραγματοποιηθεί κάτι τέτοιο είναι η ανάπτυξη ενός νέου ιστότοπου κοινωνικής δικτύωσης βασισμένο σε είδη υπάρχοντα πρότυπα, ώστε να εκμεταλλευτεί κανείς με βέλτιστο τρόπο τις βασικές του λειτουργίες. Παράλληλα, ο ιστότοπος αυτός πρέπει να είναι διαμορφωμένος και επεκταμένος με τέτοιο τρόπο που να εναρμονίζεται με τις ανάγκες και τις απαιτήσεις για κοινωνική δικτύωση μιας συγκεκριμένης κοινότητας και, ιδιαίτερα, της ακαδημαϊκής.

Δεδομένου ενός τέτοιου ιστότοπου, ορίζονται διαφορετικοί τύποι ατόμων και σχέσεων μεταξύ τους που εκμαιεύονται κάθε φορά από τους ακαδημαϊκούς ρόλους των μελών ενός Πανεπιστημιακού Ιδρύματος. Για πρώτη φορά, στα πλαίσια μιας ακαδημαϊκής κοινότητας το σύνολο των σχέσεων προσεγγίζεται μέσω ενός ευρύτερου πρίσματος που τις αποκρυσταλλώνει και τις κατατάσσει σε δύο γενικές κατηγορίες: κοινωνικές και ακαδημαϊκές, χωρίς να θεωρεί τη μία υποδεέστερη της άλλης. Μέσω αυτών, προσπαθεί να ενισχύσει τους κοινωνικούς δεσμούς των μελών της ακαδημαϊκής κοινότητας, όπως επίσης και να τους δώσει την ευκαιρία να διαμορφώσουν το ακαδημαϊκό τους γίνεσθαι μέσα από τις ακαδημαϊκές τους συσχετίσεις με τα υπόλοιπα μέλη. Φυσικό ακόλουθο των παραπάνω είναι η υποστήριξη του αισθήματος της ενότητας μεταξύ των μελών.

Για να εκφραστεί και να ενδυναμωθεί η ενότητα που προαναφέρθηκε, ο ιστότοπος ακαδημαϊκής κοινωνικής δικτύωσης υποστηρίζει τη φιλοξενία εφαρμογών, οι οποίες προσδίδουν επιπρόσθετη λειτουργικότητα στην υπηρεσία. Η κοινωνική δικτύωση με αυτές αποκτά καινούριο νόημα, καθώς δεν αλλοιώνεται, αλλά υποστηρίζεται και εξυψώνεται, μιας και η αλληλεπίδραση των χρηστών μπορεί να πραγματοποιηθεί με περισσότερους, πιο δυναμικούς και ενδιαφέροντες τρόπους.

Σημαντικότερη κατάκτηση του δοθέντος ιστότοπου ακαδημαϊκής κοινωνικής δικτύωσης, όμως, είναι η υποστήριξη εφαρμογών, οι οποίες έχουν τη δυνατότητα να προσφέρουν ολοκληρωμένες υπηρεσίες που επιταχύνουν και αυτοματοποιούν διαδικασίες παραδοσιακών ακαδημαϊκών συναλλαγών. Με αυτό τον τρόπο, οι υπηρεσίες σταματούν να απαιτούν τη συνεχή φυσική παρουσία των συμμετεχόντων για τη διεκπεραίωση γραφειοκρατικών υποθέσεων, εφόσον αναλαμβάνουν ένα μεγάλο κομμάτι τους *ηλεκτρονικά*. Ταυτόχρονα, οι συναλλασσόμενοι σε τέτοιες υπηρεσίες ιεραρχούν καλύτερα τις υποχρεώσεις τους προς την ακαδημαϊκή κοινότητα και αποκτούν ευκολότερα πρόσβαση στις εκκρεμούσες υποθέσεις που τους αφορούν άμεσα. Τελικό αποτέλεσμα από το εν λόγω εγχείρημα είναι η χρησιμοποίηση της συλλογικής δύναμης των χρηστών για την ευκολότερη πρόσβαση και υποστήριξη υπηρεσιών.

Ένα ακόμα σημαντικό χαρακτηριστικό των παραπάνω εφαρμογών είναι η υποστήριξη ανοικτών προτύπων. Με την ενσωμάτωση του προτύπου του OpenSocial (έκδοση 0.9) μέσω της υλοποίησης του Shindig (έκδοση 2.0), οι εφαρμογές λειτουργούν με την εγγύηση ενός παγκοσμίως αναγνωρισμένου και υιοθετημένου προτύπου, υποδηλώνοντας σημαντικό βαθμό *ποιότητας υπηρεσίας*. Με άλλα λόγια, ο παρών ιστότοπος ακαδημαϊκής κοινωνικής δικτύωσης και ο τρόπος με τον οποίον ενσωματώνονται εφαρμογές σε αυτόν αποτελεί έναν ώριμο καρπό δοκιμασμένων πρακτικών και προτύπων που εδώ και χρόνια έχουν επηρεάσει και συνεχίζουν να επηρεάζουν παγκοσμίως την κοινωνική δικτύωση αποφασιστικά.

Τέλος, η υιοθέτηση ανοικτών προτύπων δίνει τη δυνατότητα εξέλιξης της προσφερόμενης υπηρεσίας. Εφόσον το πρότυπο του OpenSocial συνεχώς εξελίσσεται και το Shindig ακολουθεί κατά πόδας, δίνονται συνεχώς νέες ευκαιρίες για προσαρμογή της υπηρεσίας και των εφαρμογών στις νέες επιταγές των τεχνολογιών αυτών. Εκτός αυτού, η διασημότητα των προδιαγραφών αυτών των τεχνολογιών είναι ένας παράγοντας που μπορεί να συνάδει στον εμπλουτισμό του δοθέντος ιστότοπου ακαδημαϊκής κοινωνικής δικτύωσης με περισσότερες εφαρμογές, προερχόμενες από τους χρήστες. Για την οργάνωση και την καλλιέργεια του αισθήματος της ενότητας στην ακαδημαϊκή κοινότητα, τέτοιου είδους προσφορά φαντάζει ιδανική, καθώς καλλιεργεί το αίσθημα της συμμετοχικότητας, που τόσο έχει ανάγκη η εν λόγω κοινότητα.

6.2 Προτεινόμενη μελλοντική μελέτη και πιθανές επεκτάσεις

Ο αριθμός των πιθανών επεκτάσεων που μπορούν να πραγματοποιηθούν στον ιστότοπο που έχει οριστεί με το πέρας αυτής της εργασίας δε μπορεί να εκτιμηθεί. Οι προσθήκες που χρειάζονται περισσότερο αφορούν εφαρμογές που προσφέρουν υπηρεσίες που μπορεί να συναντήσει κανείς σε μία ακαδημαϊκή κοινότητα, οι οποίες είναι πολλές και ποικίλες. Εν τούτοις, σε καμία περίπτωση δε μπορεί να αρνηθεί κάποιος ότι η πλειοψηφία των διαδικασιών στις εν λόγω υπηρεσίες μπορεί να αποδοθεί σε μορφή gadgets.

Επιπλέον, όσον αφορά την πρόσθετη λειτουργικότητα που μπορεί να προσφερθεί μέσω των εφαρμογών, ο μηχανισμός προσθήκης χαρακτηριστικών (features) στο Shindig παρέχει αρκετές ενδιαφέρουσες προοπτικές. Στα πλαίσια της ακαδημαϊκής κοινωνικής δικτύωσης, θα μπορούσαν να προστεθούν στο Shindig χαρακτηριστικά τα οποία να επιτρέπουν την προσπέλαση λογισμικών υποδομών του Πανεπιστημιακού Ιδρύματος ή ακόμα και εξωτερικών παραγόντων μέσα από gadgets. Λόγου χάρη, ένα gadget που χρησιμοποιεί ένας Γραμματέας ενός Πανεπιστημιακού Ιδρύματος θα μπορούσε να επικοινωνεί με τις Δικτυακές Υπηρεσίες (Web Services) της Γραμματείας και να πραγματοποιεί ταυτοποίηση χρηστών, καθολική δημοσίευση ανακοινώσεων, τροποποίηση στοιχείων μητρώων φοιτητών κλπ. Σημειώνεται πως μία τέτοια διαδικασία προσθήκης χαρακτηριστικών έχει προβλεφθεί από τους δημιουργούς του Shindig και γι' αυτό το λόγο παρέχεται ένας τυποποιημένος τρόπος, έτσι ώστε αυτή να πραγματοποιείται σχετικά εύκολα και απλοϊκά.

Ξεφεύγοντας λίγο από τη θεματική περιοχή των gadgets, το ακαδημαϊκό κοινωνικό δίκτυο μπορεί να βελτιωθεί με την προσθήκη δύο νέων συνιστωσών: της συνιστώσας των *μαθημάτων* και της συνιστώσας του *χρόνου*:

- Ναι μεν ο ιστότοπος υποστηρίζει την παροχή υποστήριξης σχετικά με μαθήματα μέσω των αντίστοιχων ομάδων, αλλά για να γίνει πλήρως ορατή η οντότητα *μάθημα* σαν ανεξάρτητη ακαδημαϊκή πληροφορία και να προσπελαύνεται μέσα από εφαρμογές, αξίζει να δοκιμαστεί η υλοποίηση της. Κατόπιν αυτής, η οντότητα *μάθημα* θα μπορεί να χρησιμοποιείται σαν μέρος της ακαδημαϊκής πληροφορίας στα προφίλ των χρηστών (πχ. «χρωστούμενα μαθήματα» στα

προφίλ των φοιτητών). Ταυτόχρονα, διάφορες εφαρμογές θα μπορούσαν να χρησιμοποιούν την οντότητα μάθημα για την διευκόλυνση υλοποίησης διαφόρων υπηρεσιών, όπως για παράδειγμα το πέρασμα της βαθμολογίας των φοιτητών για ένα συγκεκριμένο μάθημα. Αυτό, βέβαια, προϋποθέτει για άλλη μία φορά επέκταση του Shindig.

- Η συνιστώσα του χρόνου αφορά κατά κύριο λόγο την εφαρμογή του ακαδημαϊκού ημερολογίου στις ακαδημαϊκές συναλλαγές που λαμβάνουν χώρα μέσα στον ιστότοπο ακαδημαϊκής κοινωνικής δικτύωσης. Για παράδειγμα, ο υπολογισμός του έτους φοίτησης ενός φοιτητή ή η τήρηση προθεσμιών όσων αφορά την διεκπεραίωση εργασιών στα πλαίσια υπηρεσιών είναι δύο βασικές απαιτήσεις μίας σύγχρονης ακαδημαϊκής ατζέντας. Όμως, ένα τέτοιο εγχείρημα για την ενσωμάτωση της συνιστώσας του χρόνου στον ιστότοπο είναι αρκετά σύνθετο τόσο από άποψη σύλληψης, όσο και από άποψη υλοποίησης.

Τέλος, εκτός από τις παραπάνω σημαντικές αλλαγές, υπάρχουν και πιθανές επεκτάσεις που αφορούν την βελτιστοποίηση ή ανάπτυξη νέων λειτουργιών που θα βελτιώσουν την ποιότητα της κοινωνικής δικτύωσης που παρέχεται σε πιλοτικό στάδιο. Τέτοιες λειτουργίες μπορεί να είναι η προσθήκη μηχανισμού σχολίων σε δημοσιεύσεις, η δυνατότητα ανταλλαγής προσωπικών μηνυμάτων μεταξύ των χρηστών ή ακόμα και δομικές αλλαγές που θα επιτρέψουν τη δημιουργία λιστών φίλων κλπ. Οι επιλογές είναι αμέτρητες, αλλά όλες μπορούν να επιφέρουν ένα καλύτερο αποτέλεσμα όσον αφορά την επικοινωνία και την αλληλεπίδραση μεταξύ των χρηστών.

Βιβλιογραφία

1. ["Big Goals, Big Game, Big Records". blog.twitter.com.](#) 18 June 2010. Retrieved 4 August 2010.
2. Acquisti, A., & Gross, R. (2006). Imagined communities: Awareness, information sharing, and privacy on the Facebook. In P. Golle & G. Danezis (Eds.), *Proceedings of 6th Workshop on Privacy Enhancing Technologies* (pp. 36-58). Cambridge, UK: Robinson College.
3. Adam Ostrow (2010): "You can now login to Myspace with Facebook".
4. Barnes, S. (2006). A privacy paradox: Social networking in the United States. *First Monday*, 11 (9). http://www.firstmonday.org/issues/issue11_9/barnes/index.html
5. Betsy Schiffman (2008): In Praise of Friendster
6. Boyd, d. (2006a). Friends, Friendsters, and MySpace Top 8: Writing community into being on social network sites. *First Monday*, 11 (12). Retrieved July 21, 2007 from http://www.firstmonday.org/issues/issue11_12/boyd/
7. Boyd, d. (2008). Why youth (heart) social network sites: The role of networked publics in teenage social life. In D. Buckingham (Ed.), *Youth, Identity, and Digital Media* (pp. 119-142). Cambridge, MA: MIT Press.
8. Boyd, d. m., & Ellison, N. B. (2007). Social network sites: Definition, history, and scholarship. *Journal of Computer-Mediated Communication*, 13(1), article 11. <http://jcmc.indiana.edu/vol13/issue1/boyd.ellison.html>
9. Cliff Lampe, Nicole B. Ellison, Charles Steinfield (2007), «The Benefits of Facebook "Friends:" SocialCapital and College Students' Use ofOnline Social Network Sites», *Journal of Computer-Mediated Communication* 12 (2007) International Communication Association
10. D. Dasgupta & T. Dasgupta, (2009):Social Networks using Web 2.0. IBM Corporation

11. Donath, J., & boyd, d. (2004). Public displays of connection. *BT Technology Journal*, 22 (4), 71-82.
12. E. Athanasopoulos, A. Makridakis, S. Antonatos, D. Antoniadis, S. Ioannidis, K. G. Anagnostakis, E. P. Markatos. Antisocial Networks: Turning a Social Network into a Botnet
13. Edward Mischaud 2007 MEDIA@LSE Electronic Dissertation Series Compiled by Professor Robin Mansell and Dr. Bart Cammaerts Twitter: Expressions of the Whole Self An investigation into user appropriation of a web based communications platform.
14. Facebook marketing : designing your next marketing campaign. Page 187
15. Festa, P. (2003). Investors snub Friendster in patent grab. *CNet News*. Retrieved August 26, 2007 from http://news.com.com/2100-1032_3-5106136.html
16. GoogleAdPlanner
https://www.google.com/adplanner/planning/site_profile?hl=en#siteDetails?identifier=facebook.com&geo=GR&trait_type=1&lp=true
17. Gross, R., & Acquisti, A. (2005). Information revelation and privacy in online social networks. *Proceedings of WPES'05* (pp. 71-80). Alexandria, VA: ACM.
18. <http://chronicle.com/article/New-College-Networks-Unlike/124871/>
19. <http://mashable.com/2010/11/18/you-can-now-login-to-myspace-with-facebook/>
20. <http://techradar1.wordpress.com/2008/01/11/facebookmyspace-statistics/>
21. <http://valleywag.gawker.com/tech/myspace/myspace-the-business-of-spam-20-exhaustive-edition-199924.php>
22. <http://webtrends.about.com/od/webmashups/tp/7-great-twitter-mashups.htm>
23. <http://www.wired.com/epicenter/2008/05/friendster-inpr/>
24. ISC '08 Proceedings of the 11th international conference on Information Security
25. Kumar, R., Novak, J., & Tomkins, A. (2006). Structure and evolution of online social networks. *Proceedings of 12th International Conference on Knowledge Discovery in Data Mining* (pp. 611-617). New York: ACM Press.

26. Lampe, C., Ellison, N., & Steinfield, C. (2007). A familiar Face(book): Profile elements as signals in an online social network. *Proceedings of Conference on Human Factors in Computing Systems* (pp. 435-444). New York: ACM Press.
27. Lampe, C., Ellison, N., & Steinfield, C., (2006). A Face(book) in the crowd: Social searching vs. social browsing. *Proceedings of CSCW-2006* (pp. 167-170). New York: ACM Press.
28. Lange, P. G. (2007). Publicly private and privately public: Social networking on YouTube. *Journal of Computer-Mediated Communication*, 13(1), article 18 <http://jcmc.indiana.edu/vol13/issue1/lange.html>
29. Levy, Justin R. (Justin Robert) book (2010)
30. Liu, H., Maes, P., & Davenport, G. (2006). Unraveling the taste fabric of social networks. *International Journal on Semantic Web and Information Systems*, 2 (1), 42-71.
31. O'Shea, W. (2003, July 4-10). Six Degrees of sexual frustration: Connecting the dates with Friendster.com. <http://www.villagevoice.com/news/0323,oshea,44576,1.html>
32. Paul Anderson, (2007) What is Web 2.0? Ideas, technologies and implications for
33. Siwal (2008): "Facebook/Myspace statistics".
34. Todd Kelsey (2010) Social Networking Spaces: From Facebook to Twitter and Everything In Between. Page 323.
35. Trent Lapinski (2006): "Myspace: the Business of Spam (Exhaustive Edition)".
36. Google (2009): OpenSocial 0.9 Specification.