

Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού



ΑΛΕΞΑΝΔΡΕΙΟ Τ.Ε.Ι. ΘΕΣΣΑΛΟΝΙΚΗΣ  
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ  
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ



## ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

### <<Μελέτη Facebook API: Ανάπτυξη Facebook application>>



Του φοιτητή  
Θωμά Στυλιανού  
Αρ. Μητρώου: 991255

Επιβλέπων καθηγητής  
Ηλιούδης Χρήστος

Θεσσαλονίκη 2012

## ΠΡΟΛΟΓΟΣ

Το Facebook είναι πλέον εδώ και αρκετό καιρό μέσα στην καθημερινή ζωή του σύγχρονου ανθρώπου στις αναπτυγμένες χώρες του κόσμου. Όσοι ασχολούνται με το παγκόσμιο Διαδίκτυο είναι πολύ δύσκολο έως και απίθανο να μην έχουν ασχοληθεί με το Facebook ή να μην έχουν συναντήσει τα γνωστά “Like” ή κάποια παραπομπή σε κάποια σελίδα στο Facebook.

Ο προγραμματισμός στο Facebook είναι ένας ενδιαφέρον τομέας ανάπτυξης εφαρμογών ούτως ώστε αυτές να χρησιμοποιηθούν για εμπορικούς, για εκπαιδευτικούς ή άλλους σκοπούς.

Ο στόχος αυτού του πονήματος είναι να αναδείξει εκείνα τα χαρακτηριστικά τα οποία χρειάζεται ένας προγραμματιστής ούτως ώστε να δημιουργήσει μία εφαρμογή για το Facebook.

## ΠΕΡΙΛΗΨΗ

Για να δημιουργηθεί μία οποιαδήποτε εφαρμογή στο Facebook, θα πρέπει ο προγραμματιστής να έχει αποκτήσει γνώση των στοιχείων που αποτελούν το Facebook API.

Ξεκινώντας την εργασία, μπορεί να δει την μοντελοποίηση του Facebook. Το πώς δηλαδή είναι εσωτερικά δομημένο. Δηλαδή τι είναι ένας χρήστης πρακτικά για το Facebook, τι είναι ένας φίλος του, τι είναι οι συναπτόμενες σχέσεις φιλίας και τι σημαίνει για το Facebook το γνωστό “Like”.

Επίσης, η εργασία αυτή αναφέρεται στο αν μπορεί μία εφαρμογή να δει τα χαρακτηριστικά ενός χρήστη και ποια είναι αυτά, τι είναι Σελίδα στο Facebook και ποια χαρακτηριστικά της μπορούμε να διαχειριστούμε προγραμματιστικά, τι είναι ο Ανοιχτός Γράφος, πως κάνουμε μία αναζήτηση με την χρήση των εσωτερικών δομών του Facebook, πως διαγράφουμε ένα αντικείμενο, πως κάνουμε μία δημοσίευση προγραμματιστικά και το αν υπάρχει κάποιος τρόπος να παρακολουθούμε στατιστικά τη λειτουργία της εφαρμογής μας.

Ακόμη παρουσιάζεται το τι είναι η Πλατφόρμα του Facebook και τα περαιτέρω στοιχεία στα οποία αναλύεται, γίνεται ανάλυση της Σελίδας του Καμβά, των Κοινωνικών Καναλιών και των Αναλύσεων που προσφέρει η δομή του Facebook.

Στη συνέχεια παρουσιάζεται η χρήση των στοιχείων του Facebook κατά την ανάπτυξη των ιστοχώρων, τι είναι τα Κοινωνικά Πρόσθετα, πώς πετυχαίνεται η Αυθεντικοποίηση των χρηστών του Facebook μέσα από τον δικό μας ιστοχώρο, πώς γίνεται η Εξατομίκευση της παρουσίασης στοιχείων στον εκάστοτε χρήστη και πώς παρέχονται Αναλυτικά στοιχεία του ιστοχώρου μας με τη βοήθεια του Facebook.

Τέλος, με ένα παράδειγμα προβάλλονται όλα τα στάδια της διαδικασίας δημιουργίας μίας εφαρμογής για το Facebook.

## ABSTRACT

To create any application on Facebook, a programmer must have the knowledge of the components that the Facebook API is consisted of.

At the beginning of this thesis, the programmer can see the modeling of Facebook. How is it structured internally. In another way, what is a user practically for Facebook, what is a friend, what are the conjunctions of friendships and what does the well known “Like” mean for Facebook.

Also, this thesis refers to if an application can handle the characteristics of a user and which are they, what is a Facebook Page and which of its characteristics we can manage in a programming way, what is the Open Graph, how can we make a search with the use of the internal structures of Facebook, how do we delete an object, how do we make a post in a programming way and if there is a way to monitor statistically the function of our application.

Also, it is shown what is the Facebook Platform and its further elements that is is consisted of, the Page of Canvas is analyzed as much as the Social Channels and the Analytics that the structure of Facebook offers.

Next, the use of the components of the Facebook during the development of the websites is presented. Also, what are the Social Plugins, how the Authentication of the users of Facebook in our website can be achieved, how the Personalization of the presentation of the elements of any user can be done and how Analytics of our website can be provided with the help of Facebook.

At the end, all the stages of the process of the creation of a new application on Facebook are shown with an example.

Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

## **ΕΥΧΑΡΙΣΤΙΕΣ**

Ευχαριστώ τους καθηγητές που με βοήθησαν όλα τα έτη σπουδών μου στο Τμήμα Πληροφορικής του Α-ΤΕΙ Θεσσαλονίκης και τους συμφοιτητές με τους οποίους συνεργάστηκα και έμαθα πολλά από αυτές τις συνεργασίες που είχα μαζί τους. Προσωπικά ευχαριστώ τον καθηγητή μου κ. Ηλιούδη Χρήστο για την βοήθεια που προσέφερε και τις πολύτιμες συμβουλές του, οι οποίες ήταν καταλυτικές στην εκπόνηση της εργασίας μου.

## Κατάλογος περιεχομένων

Ευρετήριο πινάκων.....	6
Ευρετήριο σχημάτων.....	6
1.1 ΔΟΜΗ.....	12
1.2 ΣΥΝΔΕΣΕΙΣ.....	14
1.3 ID και USERNAME.....	15
1.4 ΠΡΟΣΒΑΣΗ ΣΕ ΑΛΛΑ ΣΤΟΙΧΕΙΑ ΤΟΥ ΧΡΗΣΤΗ Ή ΤΗΣ ΣΥΝΔΕΣΗΣ.....	17
1.5 ΤΕΚΜΗΡΙΑ ΠΡΟΣΒΑΣΗΣ (ΕΞΟΥΣΙΟΔΟΤΗΣΗΣ).....	18
1.6 ΕΠΙΛΟΓΗ.....	23
1.7 ΠΡΟΣΒΑΣΗ ΣΤΑ ΣΤΟΙΧΕΙΑ ΤΩΝ ΣΕΛΙΔΩΝ (PAGES).....	24
1.8 ID ΚΑΙ ΑΝΟΙΧΤΟΣ ΓΡΑΦΟΣ (OPEN GRAPH).....	26
1.9 ΑΝΑΖΗΤΗΣΗ.....	27
1.10 ΕΙΚΟΝΕΣ.....	31
1.11 ΗΜΕΡΟΜΗΝΙΕΣ.....	32
1.12 ΕΝΔΟΣΚΟΠΗΣΗ (INTROSPECTION).....	33
1.13 ΕΝΗΜΕΡΩΣΕΙΣ ΠΡΑΓΜΑΤΙΚΟΥ ΧΡΟΝΟΥ (REAL TIME UPDATES).....	35
1.14 ΔΗΜΟΣΙΕΥΣΕΙΣ.....	36
1.15 ΔΙΑΓΡΑΦΗ.....	38
1.16 ΑΝΑΛΥΣΕΙΣ.....	39
1.17 ΟΜΑΔΙΚΑ ΑΙΤΗΜΑΤΑ (BATCH REQUESTS).....	40
1.18 ΚΑΘΟΡΙΖΟΝΤΑΣ ΤΗΝ ΕΝΤΟΠΙΟΤΗΤΑ.....	41
1.19 ΣΕΛΙΔΟΠΟΙΗΣΕΙΣ (PAGING).....	42
ΕΠΙΛΟΓΟΣ.....	43
ΕΙΣΑΓΩΓΗ.....	44
2.1 Η ΠΛΑΤΦΟΡΜΑ ΤΟΥ FACEBOOK.....	45
2.2 ΤΑ ΣΤΟΙΧΕΙΑ ΤΗΣ ΠΛΑΤΦΟΡΜΑΣ ΚΑΙ ΕΦΑΡΜΟΓΕΣ ΣΤΟ FACEBOOK.....	46
2.3 FACEBOOK ΓΙΑ ΙΣΤΟΣΕΛΙΔΕΣ.....	62
ΕΠΙΛΟΓΟΣ.....	85
ΕΙΣΑΓΩΓΗ.....	86
3.1 ΔΗΜΙΟΥΡΓΙΑ ΣΕΛΙΔΑΣ ΣΤΟ FACEBOOK.....	87
ΕΠΙΛΟΓΟΣ.....	92

## Ευρετήριο πινάκων

Πίνακας 1: «Λειτουργίες εγγραφής σε αντικείμενα».....	37
---	----

## Ευρετήριο σχημάτων

Σχήμα 1: Αντικείμενα και συσχετίσεις.....	13
Σχήμα 2: Αντικείμενα και συσχετίσεις.....	14
Σχήμα 3: Δημιουργημένο προφίλ χρήστη στο Facebook.....	16
Σχήμα 4: Πρόσβαση στις ιδιότητες του χρήστη με όνομα antwnios.kyrilas.....	19
Σχήμα 5: Πρόσβαση στις ιδιότητες του χρήστη με ID 100003810151884.....	20
Σχήμα 6: Πρόσβαση στην εικόνα του προφίλ χρήστη.....	21
Σχήμα 7: Graph API Explorer.....	23

## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

Σχήμα 8: Επιλογή αδειών πρόσβασης.....	24
Σχήμα 9: Χρήση λέξης “me”.....	24
Σχήμα 10: Τα “likes” μας.....	25
Σχήμα 11: Βάζοντας το τεκμήριο πρόσβασης στην γραμμή διευθύνσεων.....	26
Σχήμα 12: Πρόσβαση στα στοιχεία του ονόματος και της φωτογραφίας του χρήστη. Το id και ο τύπος εμφανίζονται πάντα.....	27
Σχήμα 13: Το “feed” της Σελίδας ως στοιχείο της.....	28
Σχήμα 14: Τα στοιχεία της ανακοίνωσης.....	29
Σχήμα 15: Αναζήτηση με βάση το όνομα χρήστη.....	31
Σχήμα 16: Αναζήτηση τοποθεσίας.....	32
Σχήμα 17: Χρήση της παραμέτρου “metadata=1”. Εδώ φαίνονται όλες οι υποστηριζόμενες συσχετίσεις για το συγκεκριμένο αντικείμενο.....	38
Σχήμα 18: Η Σελίδα του Καμβά.....	54
Σχήμα 19: Αγαπημένες εφαρμογές και μετρητής Σελιδοδείκτη.....	56
Σχήμα 20: Ειδοποιήσεις.....	58
Σχήμα 21: Ιστορικά ανακαλύψεων.....	60
Σχήμα 22: Ιστορικό παιχνιδιών φίλων.....	61
Σχήμα 23: (Απ)ενεργοποιώντας την Κοινωνική Ανακάλυψη.....	61
Σχήμα 24: Δημοσιεύοντας Ιστορικά.....	62
Σχήμα 25: Έλεγχος Χρήστη.....	65
Σχήμα 26: Χρονοδιάγραμμα.....	66
Σχήμα 27: Πρόσφατη δραστηριότητα.....	67
Σχήμα 28: Αναζήτηση (Search).....	68
Σχήμα 29: Σελίδα (Page).....	68
Σχήμα 30: Ταμπλό Εφαρμογών και Παιχνιδιών.....	69
Σχήμα 31: Ενημερώσεις Εφαρμογών και Παιχνιδιών.....	70
Σχήμα 32: Αναλυτικά στοιχεία.....	71
Σχήμα 33: Κοινωνικά Πρόσθετα Δραστηριότητας και Συστάσεων.....	74
Σχήμα 34: Διαμορφωτής Κοινωνικού Πρόσθετου.....	76
Σχήμα 35: Πλήκτρο Σύνδεσης.....	79
Σχήμα 36: Αίτηση για άδεια πρόσβασης βασικών πληροφοριών.....	80
Σχήμα 37: Αίτηση άδειας πρόσβασης σε επιπλέον πληροφορίες.....	85
Σχήμα 38: Φόρμα Καταχώρησης.....	87
Σχήμα 39: Μήνυμα Καταχώρησης.....	88
Σχήμα 40: Αποστολή στον Τοίχο.....	94
Σχήμα 41: Κονσόλα JavaScript.....	95
Σχήμα 42: Βαθιά Γνώση (Insights).....	98
Σχήμα 43: Δημιουργία Νέας Εφαρμογής.....	102
Σχήμα 44: Έλεγχος για spammer.....	103
Σχήμα 45: Η εφαρμογή έχει καταχωρηθεί.....	103
Σχήμα 46: “App on Facebook”.....	104
Σχήμα 47: Οι αλλαγές θα πάρουν ίσως κάποια λεπτά για να μεταδωθούν.....	105
Σχήμα 48: Η εφαρμογή μας είναι έτοιμη.....	106

## ΕΙΣΑΓΩΓΗ

Η εισαγωγή στην εκμάθηση των βασικών χαρακτηριστικών της ανάπτυξης εφαρμογών στο Facebook και η εσωτερική δομή του παρουσιάζεται σε αυτή την εργασία.

Στην αρχή – Κεφάλαιο 1 – παρουσιάζεται η μοντελοποίηση της δομής του κοινωνικού αυτού δικτύου:

1.1) Τι είναι ο Γράφος του Facebook; Πώς τον χρησιμοποιούμε; Πώς αποθηκεύονται τα γνωστά προφίλ μας; Τι μορφή έχει ένα “Like” όσον αφορά τον τρόπο που το αντιλαμβάνεται εσωτερικά ο προγραμματιστής στο Facebook; Τι είναι ένα αντικείμενο στο Facebook;

1.2) Τι ρόλο παίζουν οι συσχετίσεις των αντικειμένων μεταξύ τους;

1.3) Υπάρχει κάποιος τρόπος ταυτοποίησής τους; Μπορούμε να δούμε τα χαρακτηριστικά ενός αντικειμένου και πώς;

1.4) Πώς μπορούμε να έχουμε πρόσβαση στις πληροφορίες καθενός χαρακτηριστικού κάποιου αντικειμένου ή στις συζεύξεις αυτού;

1.5) Μπορούν όλοι να έχουν πρόσβαση στα χαρακτηριστικά του προφίλ μας; Τι είναι τα τεκμήρια πρόσβασης και πώς μπορεί ο προγραμματιστής να κάνει χρήση αυτών; Εξαρτάται καθόλου από τον χρήστη; Υπάρχει κάποιο εργαλείο με την χρήση του οποίου μπορούμε αν κατανοήσουμε πιο εύκολα τον Γράφο του Facebook;

1.6) Μπορούμε να επιλέξουμε το ποιες ιδιότητες των αντικειμένων θα χρησιμοποιήσουμε; Ποιες είναι οι διάφορες χρήσεις των λειτουργιών του Facebook;

1.7) Πώς μπορούμε να έχουμε πρόσβαση στα στοιχεία των Σελίδων του Facebook; Πώς μορφοποιείται το “feed” μιας Σελίδας εν τω Facebook; Μπορούμε να έχουμε πρόσβαση σε Σελίδες που είναι διαχειρίσιμες από τους χρήστες μας;

1.8) Τι είναι ο Ανοιχτός Γράφος;

1.9) Μπορούμε να αναζητήσουμε μέσα από την εσωτερική δομή του Facebook άτομα (Χρήστες), Τοποθεσίες, δημόσιες Ανακοινώσεις, Σελίδες,



Εκδηλώσεις, Ομάδες, Επισκέψεις κάποιου χρήστη σε Σελίδες, αντικείμενα τα οποία έχουν πληροφορίες για τοποθεσία στην οποία βρίσκονται στον πραγματικό κόσμο, Εκδηλώσεις ή Ενημερώσεις και πώς; Μπορούμε να περιορίσουμε τα πεδία που μας επιστρέφονται από αυτές τις αναζητήσεις;

1.10) Μπορούμε να χρησιμοποιήσουμε την εικόνα του προφίλ μας ως εικόνα σε κάποια ιστοσελίδα μας;

1.11) Ποια είναι η εξ ορισμού μορφή της ημερομηνίας στο Facebook;

1.12) Μπορούμε να δούμε όλες τις συσχετίσεις ενός αντικειμένου; Τι είναι η ενδοσκόπηση;

1.13) Τα δεδομένα μας είναι ενημερωμένα σε πραγματικό χρόνο;

1.14) Με ποιον τρόπο γίνεται να κάνουμε τις δικές μας δημοσιεύσεις προγραμματιστικά (π.χ. δημοσίευση μιας νέας ανακοίνωσης στον Τοίχο ενός προφίλ) και σε ποιες περιπτώσεις;

1.15) Μας δίνεται η δυνατότητα να διαγράψουμε ένα αντικείμενο και σε ποια περίπτωση;

1.16) Πώς μπορούμε να δούμε αναλυτικά στατιστικές πληροφορίες για την εφαρμογή μας;

1.17) Μπορούμε να πραγματοποιήσουμε ομαδικά αιτήματα στο Facebook αντί πολλών ατομικών;

1.18) Έχει η εφαρμογή μας τη δυνατότητα ανάκλησης περιορισμένου τοπικά περιεχομένου, στη γλώσσα μιας συγκεκριμένης τοποθεσίας;

1.19) Μπορούμε να φιλτράρουμε και να κάνουμε σελιδοποίηση των δεδομένων συσχέτισης ή μιας αναζήτησης;

Πώς όμως αναπτύσσουμε εφαρμογές στο Facebook; Αυτό αναλύεται στη συνέχεια – Κεφάλαιο 2, όπου γίνεται λόγος στα βασικά και σε κάποια πιο εξειδικευμένα χαρακτηριστικά του Facebook API. Εκεί βλέπουμε τη χρησιμότητα και την χρησιμοποίηση των στοιχείων του Γράφου του Facebook που αναλύσαμε στο Κεφάλαιο 1.

Πιο αναλυτικά παρουσιάζονται:

## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

2.1) και εισαγωγή του 2.2) Γενικά περί της Πλατφόρμας του Facebook.

2.2.1) Τι είναι η Σελίδα του Καμβά και ποια η χρησιμότητά του. Τι είναι οι Σελιδοδείκτες;

2.2.2) Όταν φτιάξουμε την εφαρμογή μας, πώς θα την ανακαλύψει ο οποιοσδήποτε χρήστης; Ένας υπάρχων χρήστης πώς θα ξανασχοληθεί με την εφαρμογή μας; Πώς δημιουργούνται οι Σελιδοδείκτες και ποια η χρήση τους;

Πώς ειδοποιείται ο χρήστης για κάποια γεγονότα που σχετίζονται με την εφαρμογή μας; Τι είναι οι κλήσεις και ποιος τις δημιουργεί;

Πώς σχετίζονται οι γνωστές Ενημερώσεις με τα Ιστορικά Ανακαλύψεων; Τι σχέση έχουν τα Ιστορικά Ανακαλύψεων με τους φίλους μας στο Facebook; Μπορούν οι δημιουργοί εφαρμογών να απενεργοποιούν τα Ιστορικά Ανακαλύψεων για τις εφαρμογές τους και πώς; Μπορούν να δημοσιεύονται Ιστορικά από κάποιον χρήστη ή από την εφαρμογή μας; Μπορεί η εφαρμογή μας να κάνει μία δημοσίευση στο προφίλ ενός χρήστη; Μπορούν οι χρήστες να ορίσουν το ποιοι θα βλέπουν ότι οι συγκεκριμένοι θα φαίνεται ότι ασχολούνται με την εφαρμογή μας;

Τι είναι το Χρονοδιάγραμμα του χρήστη; Υπάρχουν ιδιαίτερα Ιστορικά που σχετίζονται με εφαρμογές που είναι Παιχνίδια; Πώς μπορούμε να δημοσιοποιήσουμε τις βαθμολογίες και τα επιτεύγματα ενός χρήστη στο Παιχνίδι που κατασκευάσαμε;

Πώς μπορεί να γίνει εφικτό να αναζητηθεί η εφαρμογή μας από την μπάρα αναζήτησης του Facebook και τι ρόλο παίζει ο δείκτης αναζήτησης του Facebook;

Πώς δημιουργούμε μία Σελίδα στο Facebook;

2.2.3) Μπορούμε να δούμε κάποια αναλυτικά στοιχεία της εφαρμογής μας; Τι είναι η επιλογή “Βαθιά Γνώση”; Μπορούμε να δούμε το πώς οι χρήστες αλληλεπιδρούν με την εφαρμογή μας; Υπάρχουν κάποια διαγνωστικά εργαλεία για την εφαρμογή μας;

2.3) Τι δυνατότητες υπάρχουν όσον αφορά την προσέγγιση του Facebook από την πλευρά της κατασκευής ιστοσελίδων;

2.3.1) Τι είναι τα Κοινωνικά Πρόσθετα και πώς μπορούμε να προσθέσουμε

στην ιστοσελίδα μας κάποιο από αυτά; Υπάρχουν περισσότεροι από ένας τρόποι; Υπάρχει κάποιο εργαλείο παραγωγής κώδικα για το Κοινωνικό Πρόσθετο που μας ενδιαφέρει να προσθέσουμε στον ιστοχώρο μας;

2.3.2) Μπορεί ένας χρήστης να συνδεθεί στην ιστοσελίδα μας χωρίς να χρειαστεί να δημιουργήσει πάλι έναν καινούργιο λογαριασμό, απλά χρησιμοποιώντας την τεχνολογία του Facebook;

Υπάρχει κάποιο πρότυπο το οποίο να εξασφαλίζει κάποιο ικανοποιητικό επίπεδο ασφάλειας στην συνδιαλλαγή του χρήστη με την εφαρμογή μας; Τι είναι η Ταυτότητα Εφαρμογής; Τι συμβαίνει ούτως ώστε να γίνεται εφικτή η σύνδεση του χρήστη με τον ιστοχώρο μας; Ποια στοιχεία του χρήστη του Facebook μπορεί να δει και να επεξεργαστεί ο ιστοχώρος μας;

Τι θα πρέπει να προσέχουμε κατά την αίτηση χορήγησης αδειών πρόσβασης από τους χρήστες στον ιστοχώρο μας; Μπορούμε να καταχωρήσουμε στοιχεία που δεν μας γίνονται γνωστά από το Facebook, με την χρήση όμως της τεχνολογίας του Facebook;

2.3.3) Πώς μπορούμε να εξατομικεύσουμε την εμπειρία του χρήστη στον ιστοχώρο μας και πως μπορούμε προγραμματιστικά να κάνουμε ανάκτηση στοιχείων του χρήστη με την χρήση του Γράφου API; Τι είναι η συνάρτηση FB.ui και πώς χρησιμοποιείται;

Υπάρχει κάποιο εργαλείο εκτέλεσης JavaScript SDK κώδικα για να πειραματιζόμαστε σε δυνητικές αλλαγές; Μπορούμε να δούμε τον κώδικα και από την πλευρά του πελάτη (client-side) και από την πλευρά του εξυπηρετητή (server-side);

2.3.4) Μπορούμε να έχουμε πρόσβαση σε αναλυτικά στοιχεία των χρηστών από τον ιστοτόπο μας, μέσω του Facebook;

Τέλος, στο Κεφάλαιο 3 αναπτύσσεται ο τρόπος δημιουργίας μίας εφαρμογής για το Facebook με σκοπό την διεπαφή του χρήστη-φοιτητή με την ηλεκτρονική σελίδα της Γραμματείας του Τμήματος Πληροφορικής του Α-ΤΕΙΘ μέσα από το Facebook.

## ΚΕΦΑΛΑΙΟ 1

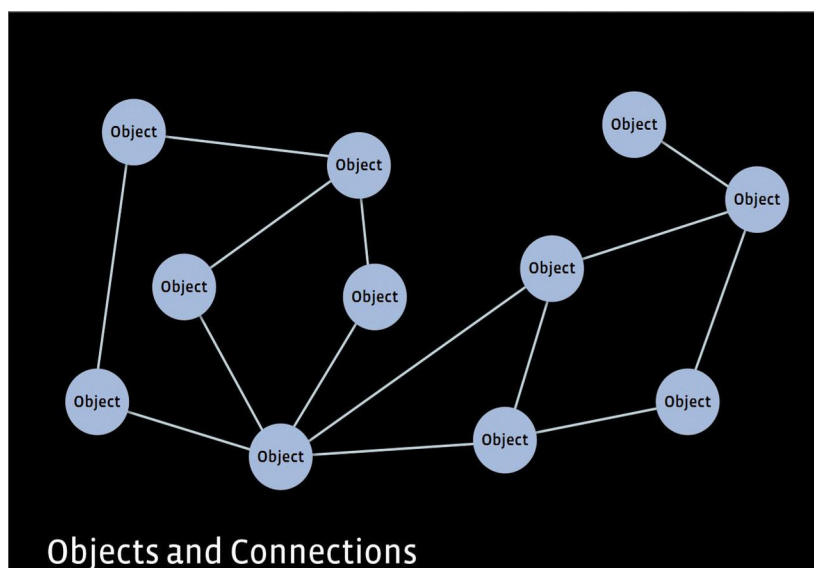
### Η ΜΟΝΤΕΛΟΠΟΙΗΣΗ ΤΟΥ FACEBOOK

#### **ΕΙΣΑΓΩΓΗ**

Σε αυτό το κεφάλαιο παρουσιάζονται τα χαρακτηριστικά δόμησης του [www.facebook.com](http://www.facebook.com). Βλέπουμε το τι χρειάζεται να υπάρχει για να λειτουργεί το facebook, τι είναι στην πραγματικότητα αυτά που μας παρουσιάζονται στον περιηγητή ιστοσελίδων όταν μπαίνουμε στο Facebook και πως ο αλληλεπιδρούν μεταξύ τους.

## 1.1 ΔΟΜΗ.

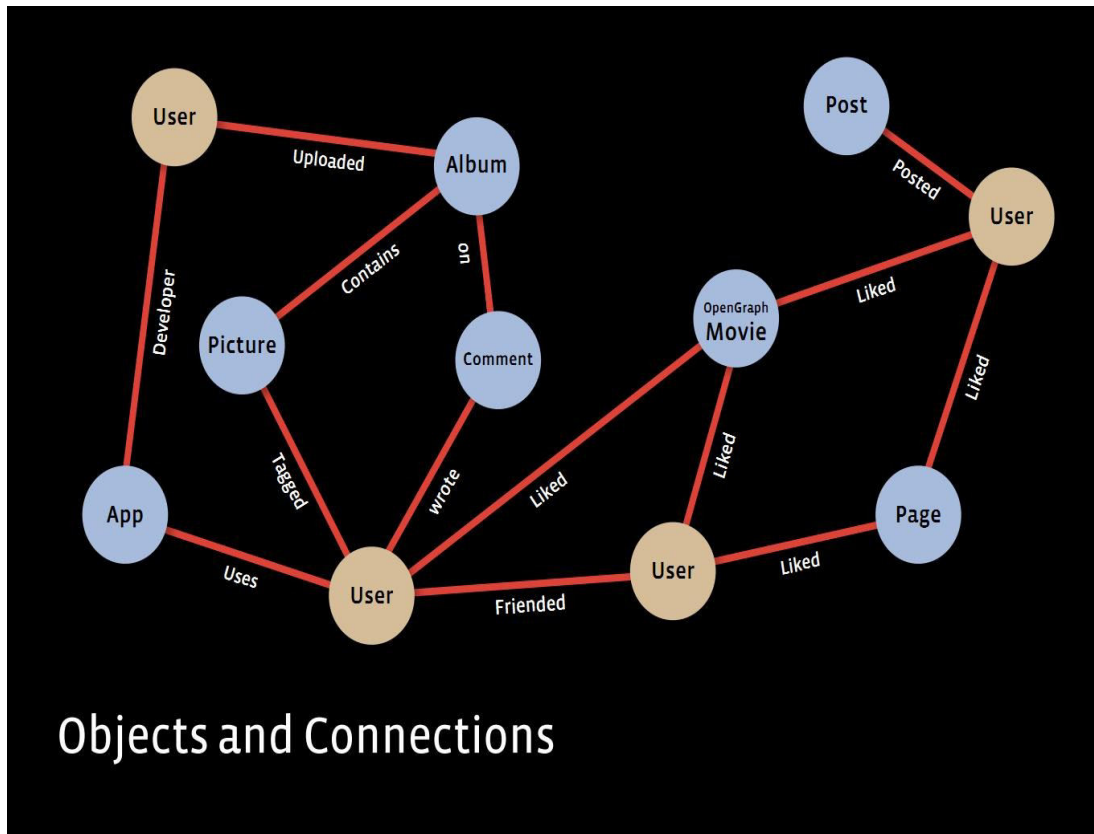
Η δομή του Facebook βασίζεται ουσιαστικά στη δομή ενός Γράφου, (Graph API). Όταν αναφερόμαστε στο Graph API τότε αναφερόμαστε στο Facebook API. Οι πληροφορίες δομούνται και αποθηκεύονται ως αντικείμενα με τις συσχετίσεις που τα συσχετίζουν (Σχήμα 1).



Σχήμα 1: Αντικείμενα και συσχετίσεις

Ο Γράφος API (εδώ θα το χρησιμοποιούμε και απλά ως Γράφο), μας επιτρέπει να διαβάσουμε ιδιότητες και συνδέσεις του κοινωνικού γράφου του Facebook. Μπορούμε να τον χρησιμοποιούμε για να διαβάσουμε συγκεκριμένα πεδία, να αποκτήσουμε τις φωτογραφίες ενός αντικειμένου, να εξετάσουμε τα μεταδεδομένα ενός αντικειμένου και να λάβουμε ενημερώσεις πραγματικού χρόνου για οποιοσδήποτε αλλαγές.

Τα αντικείμενα μπορούν να είναι χρήστες, φωτογραφικά άλμπουμ, σελίδες, σχόλια, εφαρμογές, ομάδες, ταινία Ανοιχτού Γράφου, κλπ. (Σχήμα 2).



Σχήμα 2: Αντικείμενα και συσχετίσεις

## **1.2 ΣΥΝΔΕΣΕΙΣ.**

Οι συνδέσεις (ή **σχέσεις** ή **συσχετίσεις**) μεταξύ τους εξαρτώνται από τον τύπο της σύνδεσης που μπορούν τα αντικείμενα που συνδέονται να υποστηρίξουν.

Παραδείγματα συσχετίσεων είναι τα εξής:

- ο χρήστης (User) μπορεί να συνδέεται μέσω μίας σχέσης φιλίας (Friended) με τους φίλους του (άλλους User),
- αν μας (User) αρέσει (Liked) κάτι (κάποιο αντικείμενο) τότε αυτή είναι μία συσχέτιση μεταξύ ημών και των αντικειμένων που μας αρέσουν,
- μπορεί να μας (User) αρέσει (Liked) μία ανακοίνωση (Post) που δημιουργήθηκε (Posted) από έναν χρήστη (User),
- μπορούμε (User) να γράψουμε (Wrote) ένα σχόλιο (Comment) σε ένα άλμπουμ (Album) που “ανέβηκε” (Uploaded) από έναν χρήστη (User),
- μπορούμε (User) να είμαστε οι σχεδιαστές (Developer) μίας εφαρμογής (App).

Με αυτόν ακριβώς τον τρόπο αποθηκεύονται τα δεδομένα που αφορούν το Facebook. Στην τεχνολογία του, δεν χρησιμοποιείται σχεδόν καθόλου η τεχνολογία RDBMS. Χρησιμοποιούνται κατά το πλείστο JOIN σε PHP και κάποια JOIN σε MySQL, αλλά όχι πολύ συχνά.

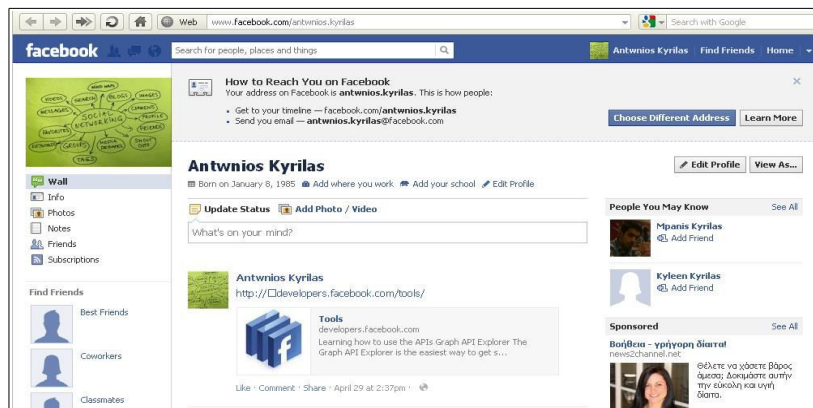
## **1.3 ID και USERNAME.**

## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

Το Graph API είναι αλλιώς γνωστό ως “κοινωνικός γράφος” (social graph). Το κάθε αντικείμενο στον κοινωνικό γράφο έχει και έναν ξεχωριστό αριθμό ταυτοποίησης (ID). Με αυτόν τον τρόπο μας δίνεται η δυνατότητα εύκολης πρόσβασης στις δημόσιες πληροφορίες ενός αντικειμένου. Δημόσιες πληροφορίες μπορεί αν είναι το όνομα, το επώνυμο ή η φωτογραφία του προφίλ ενός χρήστη.

Έτσι, μπορούμε να έχουμε πρόσβαση στις δημόσιες πληροφορίες ενός αντικειμένου ζητώντας πρόσβαση στη σελίδα <https://graph.facebook.com/ID>. Μπορούμε όμως να έχουμε πρόσβαση στις δημόσιες πληροφορίες του με το αντίστοιχο όνομα χρήστη στη θέση του ID – δηλαδή <https://graph.facebook.com/Username>. (Ως Username μπορεί να χρησιμοποιηθεί και το όνομα της σελίδας μιας εταιρίας ή ενός οργανισμού στο Facebook, και ως ID το αντίστοιχο ID της.)

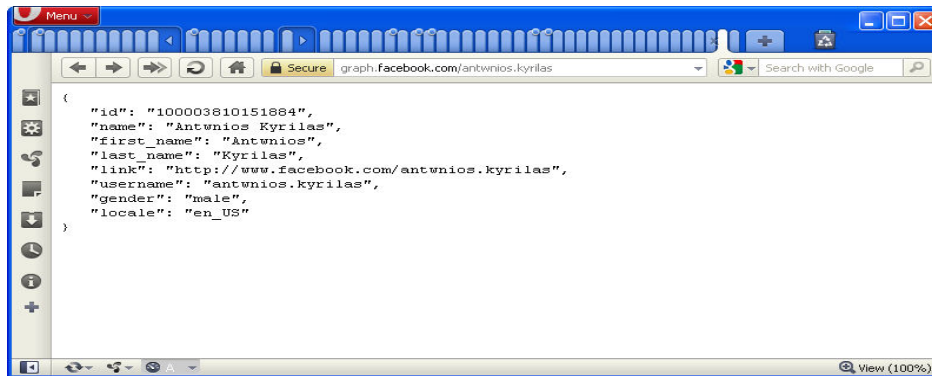
Για παράδειγμα, αν έχουμε δημιουργήσει τον χρήστη Antwnios Kyrilas στο Facebook (Σχήμα 3) και δεν ξέρουμε το ID του, τότε μπορούμε να έχουμε πρόσβαση στις ιδιότητές του μέσω της σελίδας <https://graph.facebook.com/antwnios.kyrilas> (Σχήμα 4).



**Σχήμα 3: Δημιουργημένο προφίλ χρήστη στο Facebook**

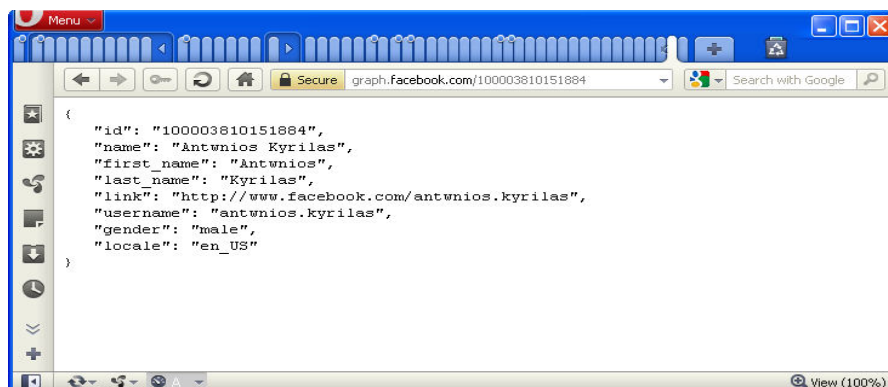


## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού



Σχήμα 4: Πρόσβαση στις ιδιότητες του χρήστη με όνομα antwnios.kyrilas

Αν όμως γνωρίζουμε το ID του (π.χ. αν ID=100003810151884), τότε μπορούμε να έχουμε πρόσβαση σε αυτές μέσω της χρήσης της διεύθυνσης <https://graph.facebook.com/100003810151884> (Σχήμα 5).



Σχήμα 5: Πρόσβαση στις ιδιότητες του χρήστη με ID 100003810151884

#### **1.4 ΠΡΟΣΒΑΣΗ ΣΕ ΑΛΛΑ ΣΤΟΙΧΕΙΑ ΤΟΥ ΧΡΗΣΤΗ Ή ΤΗΣ ΣΥΝΔΕΣΗΣ.**

Μπορούμε να έχουμε πρόσβαση σε στοιχεία του χρήστη με τον παρακάτω τρόπο: εάν π.χ. θέλουμε να έχουμε πρόσβαση στην εικόνα του προφίλ του χρήστη που δημιουργήσαμε, τότε θέτουμε “/picture” στο τέλος της μιας από τις ανωτέρω δύο διευθύνσεις (οπότε <https://graph.facebook.com/antwnios.kyrilas/picture> ή <https://graph.facebook.com/100003810151884/picture>) (Σχήμα 6).



*Σχήμα 6: Πρόσβαση στην εικόνα του προφίλ χρήστη*

Μπορούμε να εξετάσουμε τις συσχετίσεις ανάμεσα στα αντικείμενα χρησιμοποιώντας την URL δομή [https://graph.facebook.com/ID/CONNECTION\\_TYPE](https://graph.facebook.com/ID/CONNECTION_TYPE), όπου CONNECTION\_TYPE το είδος της σχέσης. Υποστηρίζονται διαφορετικοί τύποι συσχετίσεων για διαφορετικά αντικείμενα.

### **1.5 ΤΕΚΜΗΡΙΑ ΠΡΟΣΒΑΣΗΣ (ΕΞΟΥΣΙΟΔΟΤΗΣΗΣ).**

Εάν όμως θέλουμε περισσότερες πληροφορίες για έναν χρήστη, θα πρέπει να λάβουμε την άδειά του. Σε υψηλό επίπεδο, θα πρέπει να λάβουμε ένα τεκμήριο πρόσβασης (access token) για τον συγκεκριμένο χρήστη. Το Graph API χρησιμοποιεί το πρωτόκολλο εξουσιοδότησης OAuth 2.0 για τις εξουσιοδοτήσεις. Αφού λάβουμε το τεκμήριο, μπορούμε να εκτελούμε εξουσιοδοτημένες αιτήσεις εκ μέρους του εν λόγω χρήστη περιλαμβάνοντας το τεκμήριο πρόσβασης στις αιτήσεις του γράφου API (Graph API requests):

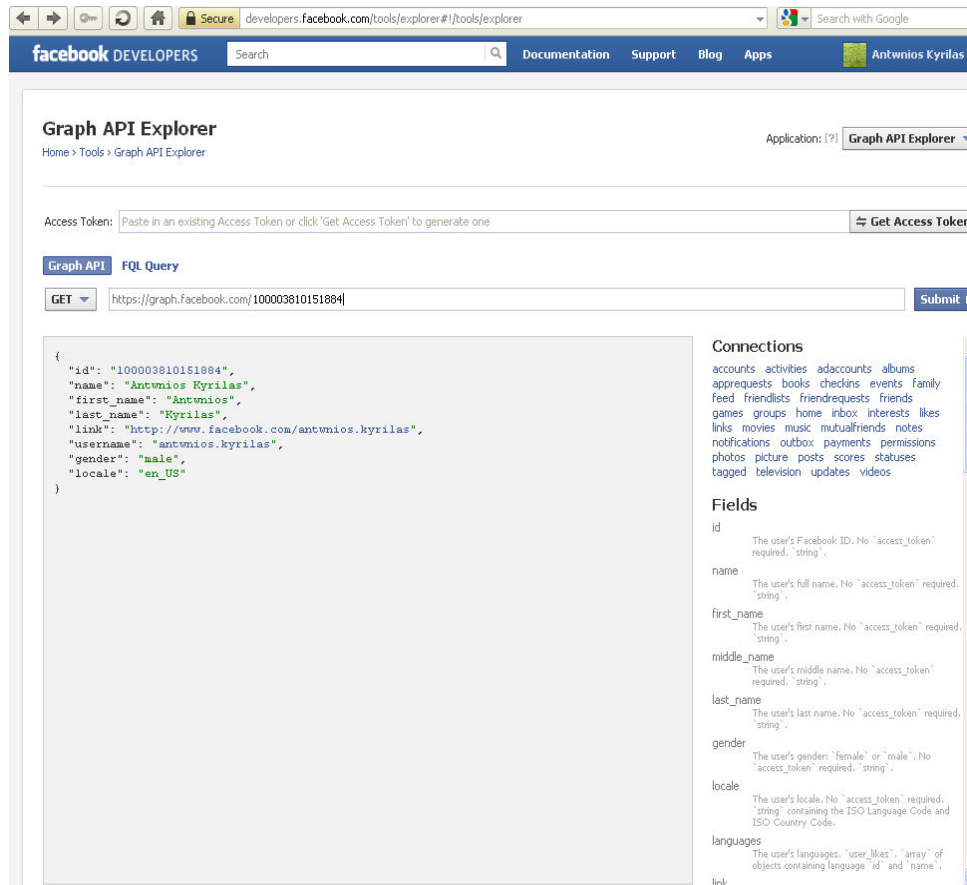
[https://graph.facebook.com/100003810151884?access\\_token=...](https://graph.facebook.com/100003810151884?access_token=...)

Ένα αρκετά ενδιαφέρον εργαλείο για την κατανόηση και τον πειραματισμό στον Γράφο του Facebook είναι το Graph API Explorer, στο οποίο έχουμε πρόσβαση από τον χρήστη με ID=100003810151884 μέσα από τη διεύθυνση:

<https://developers.facebook.com/tools/explorer#!/tools/explorer?method=GET&path=100003810151884>.

Μέσα από αυτή την εφαρμογή, μπορούμε να δούμε όλα τα στοιχεία τα οποία είμαστε εξουσιοδοτημένοι να δούμε. Π.χ. μπορούμε να δούμε τα κύρια στοιχεία του προφίλ μας (Σχήμα 7).

## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού



Σχήμα 7: Graph API Explorer

Για να κάνει κανείς κλήσεις διαχείρισης (administrative call) οι οποίες δεν απαιτούν ενεργό χρήστη, είναι απαραίτητη η απόκτηση ενός `access_token` για την εφαρμογή μας. Όπως βλέπουμε στην εικόνα, μέχρι στιγμής δεν έχουμε λάβει κάποιο τεκμήριο πρόσβασης από τον εν λόγω χρήστη.

Αυτό μπορούμε να το κάνουμε πατώντας στο “Get Access Token”. Έπειτα, βλέπουμε το μήνυμα του Σχήματος 8 και κάνουμε κλικ ανάλογα στην επιλογή αδειών πρόσβασης που μας ενδιαφέρει στο “Get Access Token”. Χωρίς εκτεταμένες άδειες (Extended Permissions), μπορούμε να έχουμε πρόσβαση στις πληροφορίες που ο χρήστης έχει διαθέσιμες προς όλους. Αν όμως θέλουμε πληροφορίες όπως το e-mail του ή το ιστορικό εργασίας του, τότε θα πρέπει να

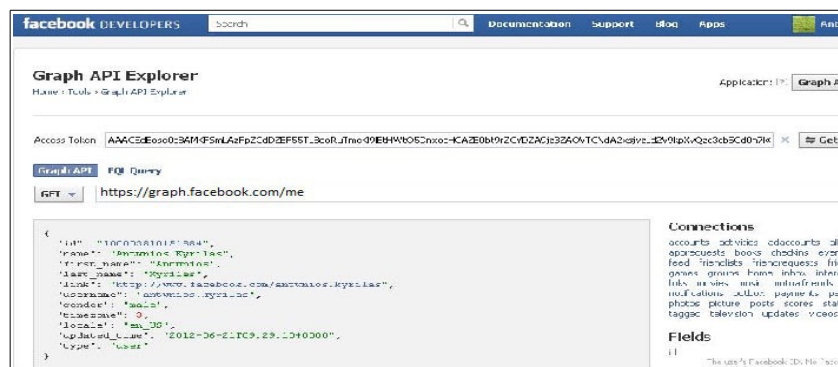
Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

ζητήσουμε εκτεταμένες άδειες. Η εικόνα μας τότε αλλάζει, βλέποντας λίγα περισσότερα στοιχεία του χρήστη.



Σχήμα 8: Επιλογή αδειών πρόσβασης

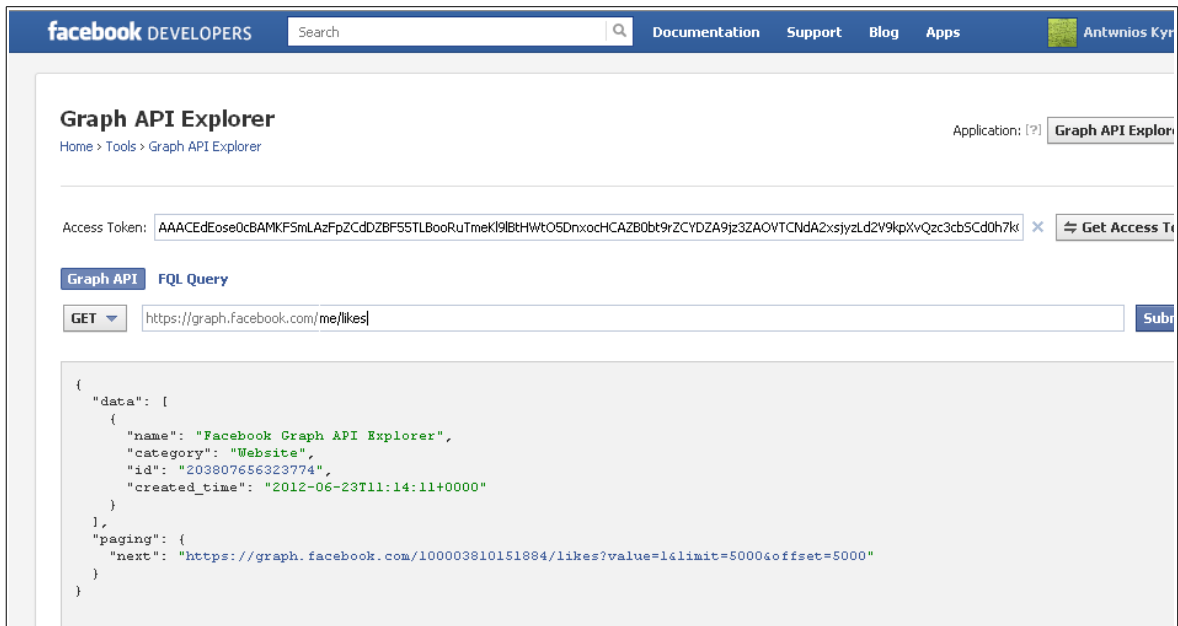
Μπορούμε επίσης να χρησιμοποιήσουμε την λέξη “me”, αντί του ονόματος χρήστη αν αναφερόμαστε στον δικό μας λογαριασμό (Σχήμα 9).



Σχήμα 9: Χρήση λέξης “me”

Έτσι, μπορούμε να δούμε τα likes μας, θέτοντας μετά το “me”, “/likes” (Σχήμα 10).

## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού



*Σχήμα 10: Τα "likes" μας*

Βεβαίως, ο παραδοσιακός τρόπος από την γραμμή διευθύνσεων του περιηγητή ιστοσελίδων που χρησιμοποιούμε είναι κι αυτός διαθέσιμος, αφού λάβουμε το τεκμήριο πρόσβασης που χρειαζόμαστε, με τον τρόπο που αναφέραμε παραπάνω (Σχήμα 11).

Ανάλογα, μπορούμε να βρούμε τους φίλους μας, πληκτρολογώντας [https://graph.facebook.com/100003810151884/friends?access\\_token=AAACEdEose0cBAE3EW7MgKM9CvBCST0iSrZBGSbuKmxTZAZBj05h8ZA2RJHaTtC8Lr9Via nZAcGGTCGyPfJkqgDSbx2yAOYDj0n9pYPxuGCAZBwUgkS0t5d](https://graph.facebook.com/100003810151884/friends?access_token=AAACEdEose0cBAE3EW7MgKM9CvBCST0iSrZBGSbuKmxTZAZBj05h8ZA2RJHaTtC8Lr9Via nZAcGGTCGyPfJkqgDSbx2yAOYDj0n9pYPxuGCAZBwUgkS0t5d).

Στο εξής θα χρησιμοποιούμε το εργαλείο Graph API Explorer, για ευκολία στη χρήση.

## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού



```
{
  "data": [
    {
      "name": "Facebook Graph API Explorer",
      "category": "Website",
      "id": "203807656323774",
      "created_time": "2012-06-23T11:14:11+0000"
    }
  ],
  "paging": {
    "next": "https://graph.facebook.com/100003810151884/likes?value=1&access_token=AAACEdEose0cBAMK"
  }
}
```

*Σχήμα 11: Βάζοντας το τεκμήριο πρόσβασης στην γραμμή διευθύνσεων*

## 1.6 ΕΠΙΛΟΓΗ.

Εξ αρχής, οι περισσότερες ιδιότητες των αντικειμένων επιστρέφονται όταν εκτελούμε ένα ερώτημα. Μπορούμε να διαλέξουμε τα πεδία (ή τις συνδέσεις) που θέλουμε να επιστραφούν, αναγράφοντάς τα ως παραμέτρους του ερωτήματος. Για παράδειγμα, για να επιστραφούν μόνο το όνομα και η εικόνα του προφίλ μας, αναγράφουμε μετά το όνομα “?fields=name,picture” (Σχήμα 12).



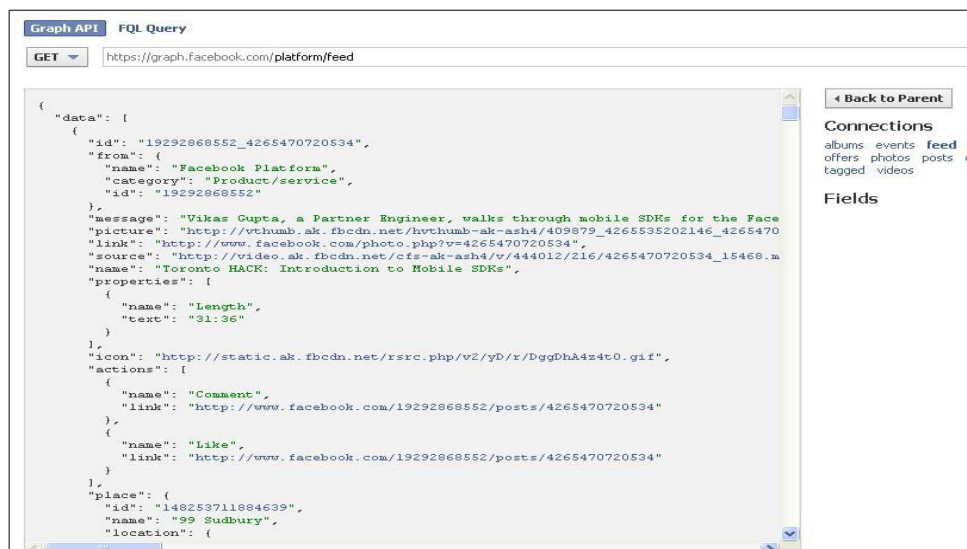
The screenshot shows the Graph API Explorer interface. At the top, it says "Graph API Explorer" and "Application: [?] Graph". Below that, there's a navigation bar with "Home > Tools > Graph API Explorer". The main area has an "Access Token" field containing a long alphanumeric string. Below the token, there are two tabs: "Graph API" and "FQL Query". The "FQL Query" tab is active, and the query field contains the URL: "https://graph.facebook.com/antwnios.kyrilas?fields=name,picture". To the left of the query field is a "GET" dropdown menu. Below the query field, there is a JSON response showing user profile information: {"name": "Antwnios Kyrilas", "id": "100003810151884", "type": "user", "picture": "http://profile.ak.fbcdn.net/hprofile-ak-snc4/565238\_100003810151884\_835680325...". To the right of the JSON response, there is a "Connections" section listing various connection types: accounts, activities, adaccounts, apprequests, books, checkins, ev, feed, friendlists, friendrequests, games, groups, home, inbox, int, links, movies, music, mutualfriend, notifications, outbox, payments.

Σχήμα 12: Πρόσβαση στα στοιχεία του ονόματος και της φωτογραφίας του χρήστη. Το id και ο τύπος εμφανίζονται πάντα.



## 1.7 ΠΡΟΣΒΑΣΗ ΣΤΑ ΣΤΟΙΧΕΙΑ ΤΩΝ ΣΕΛΙΔΩΝ (PAGES).

Για μία Σελίδα στο Facebook όπως αυτή του Facebook Platform, η οποία δεν είναι αντικείμενο τύπου χρήστη, μπορούμε ανάλογα να δούμε κάποια – όχι βεβαίως ίδια – χαρακτηριστικά. Η Σελίδα αυτή φερ' ειπείν δεν μπορεί να κάνει “like” σε κάποιο βίντεο. Μπορούμε όμως να δούμε το “feed” (ενημερώσεις) της Σελίδας (Σχήμα 13).

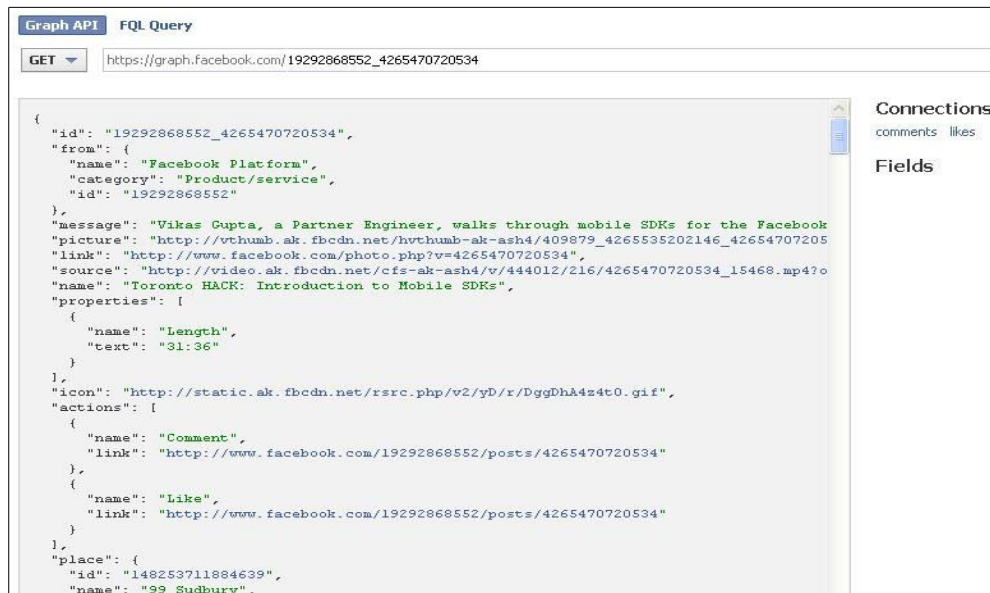


Σχήμα 13: Το “feed” της Σελίδας ως στοιχείο της

Εδώ έχουμε επιλέξει να δούμε τα στοιχεία της Σελίδας /platform και από τις συνδέσεις (connections) στα δεξιά, επιλέξαμε “feed”. (Αντίστοιχα, μπορούσαμε από την αρχή να πληκτρολογήσουμε /platform/feed). Το “feed” είναι ένας πίνακας (array) με id ανακοινώσεων (Post id). Οπότε, αντιγράφοντας το id κάποιας ανακοίνωσης στο πεδίο αναζήτησης (ή κάνοντας κλικ επάνω του), μας εμφανίζονται τα στοιχεία αυτής της ανακοίνωσης. Στο Σχήμα 14 φαίνονται τα στοιχεία της πρώτης στη σειρά ανακοίνωσης του Σχήματος 13. Έτσι, βλέπουμε

## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

τίνος είναι, την εικόνα της, κλπ. Επίσης εμφανίζονται πληροφορίες για τα σχόλια (comments) προς αυτήν.



The screenshot shows the Facebook Graph API FQL Query interface. The URL in the address bar is `https://graph.facebook.com/19292868552_4265470720534`. The response is a JSON object containing details about a post, including the sender (Facebook Platform), the message text, image and video links, and actions for commenting and liking. The post is located in the '99 Sudbury' place.

```
{
  "id": "19292868552_4265470720534",
  "from": {
    "name": "Facebook Platform",
    "category": "Product/service",
    "id": "19292868552"
  },
  "message": "Vikas Gupta, a Partner Engineer, walks through mobile SDKs for the Facebook",
  "picture": "http://vthumb.ak.fbcdn.net/hvthumb-ak-ash4/409879_4265535202146_4265470720534",
  "link": "http://www.facebook.com/photo.php?v=4265470720534",
  "source": "http://video.ak.fbcdn.net/cfs-ak-ash4/v/444012/216/4265470720534_15468.mp4?o",
  "name": "Toronto HACK: Introduction to Mobile SDKs",
  "properties": [
    {
      "name": "Length",
      "text": "31:36"
    }
  ],
  "icon": "http://static.ak.fbcdn.net/rsrc.php/v2/yD/r/DggDhA4z4t0.gif",
  "actions": [
    {
      "name": "Comment",
      "link": "http://www.facebook.com/19292868552/posts/4265470720534"
    },
    {
      "name": "Like",
      "link": "http://www.facebook.com/19292868552/posts/4265470720534"
    }
  ],
  "place": {
    "id": "148253711884639",
    "name": "99 Sudbury",
  }
}
```

*Σχήμα 14: Τα στοιχεία της ανακοίνωσης*

Όπως είναι αυτονόητο, η στήλη των συνδέσεων έχει αλλάξει. Οπότε, έχουμε τη δυνατότητα να δούμε τα σχόλια από εκεί. Τα τελευταία είναι ουσιαστικά οι ανακοινώσεις των σχολίων της προηγούμενης ανακοίνωσης. Πατώντας στο “likes”, βλέπουμε τους χρήστες που αρέσουν το συγκεκριμένο Post.

Μπορούμε να έχουμε πρόσβαση σε Σελίδες που είναι διαχειρίσιμες από τους χρήστες μας, ζητώντας την άδεια “manage\_pages”. Από τη στιγμή που ένας χρήστης έχει παραχωρήσει στην εφαρμογή μας την ανωτέρω άδεια, τότε η συσχέτιση “accounts” θα επιτρέψει την πρόσβαση στην λίστα των Σελίδων που διαχειρίζεται, όπως και την ανάκληση ενός access\_token για καθεμιά. Περισσότερες πληροφορίες μπορούμε να βρούμε στο: <http://developers.facebook.com/docs/authentication/pages/>.

### **1.8 ID ΚΑΙ ΑΝΟΙΧΤΟΣ ΓΡΑΦΟΣ (OPEN GRAPH).**

Έτσι, είναι φανερό ότι η διαδικασία λειτουργίας του Graph API είναι η περιήγηση συνδέσεων μεταξύ αντικειμένων.

Εάν θέλουμε να έχουμε πρόσβαση σε δύο ή παραπάνω χρήστες ή σελίδες, πληκτρολογούμε “?ids=” και το(-α) id ακολουθούμενο(-α) από κόμμα(-τα). Εάν π.χ. ενδιαφερόμαστε για τους χρήστες με id 8 και 10, τότε πληκτρολογούμε ?ids=8,10. Οπότε, εάν χρειαστούμε να χρησιμοποιήσουμε τους φίλους κάποιου χρήστη ή τα likes των φίλων κάποιου χρήστη, η σύνταξη αυτή μας βοηθάει.

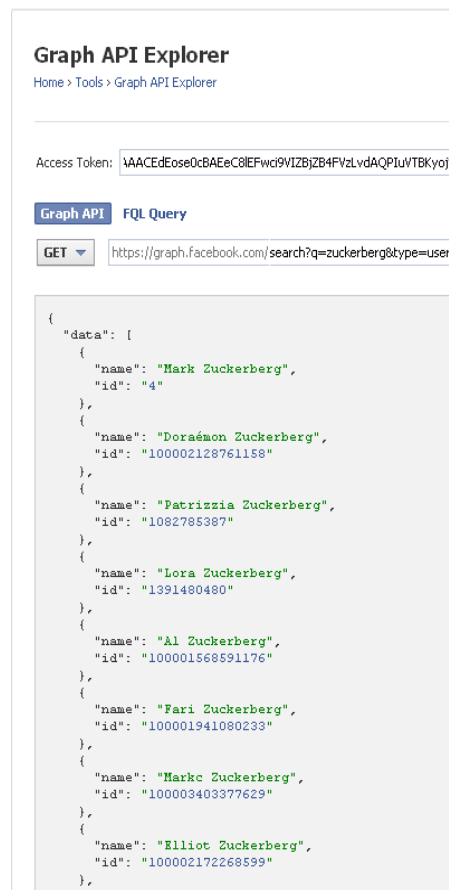
Η σύνταξη αυτή δέχεται και τη χρήση URL διευθύνσεων. Αυτό είναι χρήσιμο όταν θέλουμε να βρούμε το ID ενός URL στον Ανοιχτό Γράφο. Ο Ανοιχτός Γράφος ουσιαστικά δίνει τη δυνατότητα σε οποιαδήποτε ιστοσελίδα να γίνει ένα αντικείμενο στον Ανοιχτό Γράφο για χρήση στο Facebook. Για παράδειγμα, μπορούμε να χρησιμοποιήσουμε τη διεύθυνση <https://graph.facebook.com/?ids=http://www.imdb.com/title/tt0117500/>.

Όταν ψάχνουμε Post μέσω των συνδέσεων /home, /feed ή /posts, μπορούμε να περιορίσουμε τα αποτελέσματα σε αυτά που περιέχουν μία τοποθεσία, προσθέτοντας “with=location” στις παραμέτρους του URL. Π.χ.: <https://graph.facebook.com/me/home?with=location> (όπου location, θέτουμε κάποια τοποθεσία, π.χ. “thessaloniki”).

## 1.9 ΑΝΑΖΗΤΗΣΗ.

Μπορούμε να κάνουμε αναζήτηση μεταξύ όλων των δημόσιων αντικειμένων του Κοινωνικού Γράφου με την χρήση της διεύθυνσης <https://graph.facebook.com/search>. Η μορφή της είναι: [https://graph.facebook.com/search?q=QUERY&type=OBJECT\\_TYPE](https://graph.facebook.com/search?q=QUERY&type=OBJECT_TYPE). Υποστηρίζονται αναζητήσεις για τους ακόλουθους τύπους αντικειμένων:

- Άτομα: <https://graph.facebook.com/search?q=zuckerberg&type=user> (Σχήμα 15)



The screenshot shows the Graph API Explorer interface. At the top, it says "Graph API Explorer" with a breadcrumb "Home > Tools > Graph API Explorer". Below that, there is an "Access Token" field containing a long alphanumeric string. There are two tabs: "Graph API" (selected) and "FQL Query". Under the "Graph API" tab, there is a "GET" dropdown and a text input field containing the URL "https://graph.facebook.com/search?q=zuckerberg&type=user". Below the input fields, there is a large code block displaying the JSON response of the search query. The JSON response is a list of user profiles, each with a "name" and an "id" field.

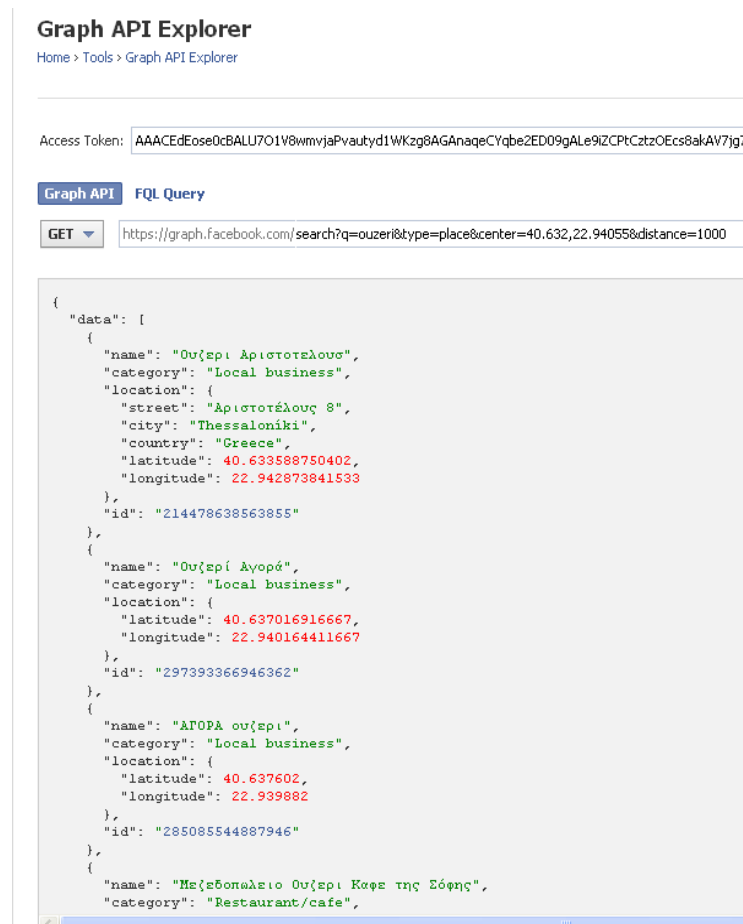
```
{
  "data": [
    {
      "name": "Mark Zuckerberg",
      "id": "4"
    },
    {
      "name": "Doraémon Zuckerberg",
      "id": "100002128761158"
    },
    {
      "name": "Patrizzia Zuckerberg",
      "id": "1082785387"
    },
    {
      "name": "Lora Zuckerberg",
      "id": "1391480480"
    },
    {
      "name": "Al Zuckerberg",
      "id": "100001568591176"
    },
    {
      "name": "Fari Zuckerberg",
      "id": "100001941080233"
    },
    {
      "name": "Markc Zuckerberg",
      "id": "100003403377629"
    },
    {
      "name": "Elliot Zuckerberg",
      "id": "100002172268599"
    },
  ],
}
```

Σχήμα 15: Αναζήτηση με βάση το όνομα χρήστη

## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

- Τοποθεσίες: <https://graph.facebook.com/search?q=coffee&type=place>  
Μπορούμε να μικρύνουμε το εύρος της αναζήτησης σε μία συγκεκριμένη τοποθεσία και απόσταση, προσθέτοντας την παράμετρο “center” (με γεωγραφικό πλάτος και μήκος) και μία προαιρετική παράμετρο “distance” (Σχήμα 16):

[https://graph.facebook.com/search?q=ouzeri&type=place  
&center=40.632,22.94055&distance=1000](https://graph.facebook.com/search?q=ouzeri&type=place&center=40.632,22.94055&distance=1000)



The screenshot shows the Graph API Explorer interface. The 'Access Token' field contains a long alphanumeric string. The 'Graph API' tab is selected, and the 'FQL Query' field contains the URL: `https://graph.facebook.com/search?q=ouzeri&type=place&center=40.632,22.94055&distance=1000`. The response is a JSON array of objects, each representing a business location with fields for name, category, location (street, city, country), and coordinates (latitude and longitude).

```
{
  "data": [
    {
      "name": "Ουζερι Αριστοτέλους",
      "category": "Local business",
      "location": {
        "street": "Αριστοτέλους 8",
        "city": "Thessaloniki",
        "country": "Greece",
        "latitude": 40.633588750402,
        "longitude": 22.942873841533
      },
      "id": "214478638563855"
    },
    {
      "name": "Ουζερι Αγριά",
      "category": "Local business",
      "location": {
        "latitude": 40.637016916667,
        "longitude": 22.940164411667
      },
      "id": "297393366946362"
    },
    {
      "name": "ΑΓΡΙΑ Ουζερι",
      "category": "Local business",
      "location": {
        "latitude": 40.637602,
        "longitude": 22.939882
      },
      "id": "285085544887946"
    },
    {
      "name": "Μεζεδοπωλείο Ουζερι Καφε της Σόφης",
      "category": "Restaurant/cafe",

```

*Σχήμα 16: Αναζήτηση τοποθεσίας*

- Όλες οι δημόσιες ανακοινώσεις: <https://graph.facebook.com/search?q=watermelon&type=post>
- Σελίδες: <https://graph.facebook.com/search?q=platform&type=page>
- Εκδηλώσεις: <https://graph.facebook.com/search?q=conference&type=event>
- Ομάδες: <https://graph.facebook.com/search?q=programming&type=group>
- Επίσκεψη (Checkin): <https://graph.facebook.com/search?type=checkin>  
Επιστρέφει τα τελευταία Checkin μας ή των φίλων μας ή σε αυτά στα οποία εμείς ή οι φίλοι μας έχουμε επικολληθεί. Τα Checkin αναπαριστούν μία απλή επίσκεψη ενός χρήστη σε μία τοποθεσία. Οι Χρήστες και οι Σελίδες έχουν συσχετίσεις Checkin.
- Αντικείμενα με τοποθεσία: Τα ακόλουθα επιστρέφουν πληροφορίες όσον αφορά αντικείμενα τα οποία έχουν πληροφορίες για την τοποθεσία τους. Επιπρόσθετα, τα επιστρεφόμενα αντικείμενα θα είναι εκείνα στα οποία εμείς ή οι φίλοι μας έχουμε επικολληθεί, ή εκείνα τα οποία δημιουργήσαμε εμείς ή οι φίλοι μας. Υπάρχουν σημαντικές διαφορές συμπεριφοράς στα αποτελέσματα, εξαρτώμενες από τις άδειες τις οποίες έχει αποκτήσει η εφαρμογή μας.
  - Για την αναζήτηση αντικειμένων κοντά σε μία γεωγραφική περιοχή, χρησιμοποιούμε “type=location” και προσθέτουμε τις παραμέτρους “center” και “distance”:  
<https://graph.facebook.com/search?type=location&center=37.76,-122.427&distance=1000>
  - Για την αναζήτηση αντικειμένων σε έναν συγκεκριμένο τόπο, χρησιμοποιούμε “type=location” και παραθέτουμε το ID της τοποθεσίας αυτής. Για παράδειγμα, για ανακοινώσεις στο Facebook HQ, χρησιμοποιούμε:  
<https://graph.facebook.com/search?type=location&place=166793820034304>

- Ονόματα Εκδηλώσεων (Event Names): Μπορούμε να περιορίσουμε τα πεδία που μας επιστρέφονται από αυτές τις αναζητήσεις χρησιμοποιώντας την URL παράμετρο “?fields=” με τον ίδιο τρόπο όπως όταν διαβάζουμε άλλα αντικείμενα. Για παράδειγμα:

[https://graph.facebook.com/search?fields=name&q=conference  
&type=event](https://graph.facebook.com/search?fields=name&q=conference&type=event)

Κάποια πεδία, όπως τα “id” και “start\_time” για τις Εκδηλώσεις, επιστρέφονται πάντα.

- Ενημερώσεις (News Feed): Μπορούμε επίσης να ψάξουμε για τις Ενημερώσεις ενός χρήστη (εδώ τις δικές μας), οι οποίες είναι περιορισμένες για τους φίλους του, προσθέτοντας την παράμετρο “q” στην “home” URL σύνδεση:

<https://graph.facebook.com/me/home?q=facebook>

Σημείωση: η χρήση του “/me/home” ανακτά μία μη ενημερωμένη καταχώρηση των Ενημερώσεων. Αυτό είναι ένα γνωστό θέμα, το οποίο οι διαχειριστές του Facebook δεν προτίθενται να λύσουν.

Όταν αναζητούμε δημοσιεύσεις στις Ενημερώσεις του χρήστη, μπορούμε να χρησιμοποιήσουμε τις παραμέτρους “since”, “until” και “limit”. Οι “since” και “until” δέχονται και οι δύο χρονοσφραγίδα τύπου Unix. Όταν κάνουμε αναφορές πίσω στο χρόνο πρέπει να χρησιμοποιούμε “until”, από κοινού με την “limit”, όπου η “until” είναι η Unix τιμή του χρόνου του πεδίου “created\_time” του πρώτου επιστρεφόμενου αντικειμένου του τελευταίου ερωτήματος, ενώ η “limit” χρησιμοποιείται για λόγους σελιδοποίησης. Θα πρέπει να αναφερθεί ότι μπορούμε να πάμε μόνο 1 με 2 εβδομάδες πίσω στις Ενημερώσεις.

## **1.10 ΕΙΚΟΝΕΣ.**

## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

Κατά την κατασκευή μιας ιστοσελίδας, μπορεί να αποδοθεί η φωτογραφία του προφίλ μας σε οποιοδήποτε αντικείμενο, προσθέτοντας την κατάληξη “/picture” στο URL του αντικειμένου. Για παράδειγμα, το παρακάτω θα αποδώσει την δημόσια φωτογραφία του χρήστη antwnios.kyrilas:

```

```

Μπορούμε να καθορίσουμε το μέγεθος μιας εικόνας με την παράμετρο “type”, η οποία μπορεί να έχει μία από τις τιμές:

“square” (50 pixel x 50 pixel), “small” (50 pixel μήκος, με μεταβλητό ύψος), “normal” (100 pixel μήκος, με μεταβλητό ύψος) και “large” (περίπου 200 pixel μήκος, και μεταβλητό ύψος). Π.χ.:

<http://graph.facebook.com/antwnios.kyrilas/picture?type=large>.

Εάν χρειάζεται να αποκτήσουμε μία εικόνα μέσα από μία ασφαλή σύνδεση, θέτουμε την παράμετρο “return\_ssl\_resources” σε 1. Δηλαδή: [http://graph.facebook.com/antwnios.kyrilas/picture?return\\_ssl\\_resources=1](http://graph.facebook.com/antwnios.kyrilas/picture?return_ssl_resources=1).

### **1.11 ΗΜΕΡΟΜΗΝΙΕΣ.**

Τα πεδία ημερομηνίας βρίσκονται ως ISO-8601 μορφοποιημένα string.



Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

Υπάρχει η δυνατότητα αλλαγής του τύπου της ημερομηνίας προσθέτοντας την παράμετρο ερωτήματος “date\_format”. Η αποδεκτή μορφή των string είναι ίδια με αυτή που είναι αποδεκτή από την συνάρτηση ημερομηνίας της php. Για παράδειγμα, εάν γράψουμε [http://graph.facebook.com/platform/feed?date\\_format=U](http://graph.facebook.com/platform/feed?date_format=U), τότε αυτό θα επιστρέψει το feed της σελίδας του Facebook Platform, με ημερομηνίες μορφοποιημένες κατά Unix.

### **1.12 ΕΝΔΟΣΚΟΠΗΣΗ (INTROSPECTION).**

Ο Γράφος API υποστηρίζει την αυτοεξέταση των αντικειμένων, η οποία μας επιτρέπει να δούμε όλες τις συνδέσεις τις οποίες έχει ένα αντικείμενο χωρίς

πρωτίτερα να γνωρίζουμε τον τύπο τους. Για να πάρουμε αυτήν την πληροφορία, προσθέτουμε “metadata=1” στο URL του αντικειμένου, και το αποτέλεσμα θα περιλαμβάνει μία ιδιότητα “metadata”, η οποία περιέχει όλες τις υποστηριζόμενες συνδέσεις για το συγκεκριμένο αντικείμενο. Για παράδειγμα, μπορούμε να δούμε όλες τις συνδέσεις του χρήστη antwnios.kyrilas χρησιμοποιώντας τη διεύθυνση: <https://graph.facebook.com/100003810151884?metadata=1> (Σχήμα 17).

Η ενδοσκόπηση είναι ένας χρήσιμος τρόπος για να βρεθούν όλα τα αντικείμενα με τα οποία είναι συνδεδεμένοι οι χρήστες.

## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού



```
{
  "id": "100003810151884",
  "name": "Antwnios Kyrilas",
  "first_name": "Antwnios",
  "last_name": "Kyrilas",
  "link": "http://www.facebook.com/antwnios.kyrilas",
  "username": "antwnios.kyrilas",
  "gender": "male",
  "locale": "en_US",
  "metadata": {
    "connections": {
      "home": "https://graph.facebook.com/100003810151884/home",
      "feed": "https://graph.facebook.com/100003810151884/feed",
      "friends": "https://graph.facebook.com/100003810151884/friends",
      "mutualfriends": "https://graph.facebook.com/100003810151884/mutualfriends",
      "family": "https://graph.facebook.com/100003810151884/family",
      "payments": "https://graph.facebook.com/100003810151884/payments",
      "activities": "https://graph.facebook.com/100003810151884/activities",
      "interests": "https://graph.facebook.com/100003810151884/interests",
      "music": "https://graph.facebook.com/100003810151884/music",
      "books": "https://graph.facebook.com/100003810151884/books",
      "movies": "https://graph.facebook.com/100003810151884/movies",
      "television": "https://graph.facebook.com/100003810151884/television",
      "games": "https://graph.facebook.com/100003810151884/games",
      "adaccounts": "https://graph.facebook.com/100003810151884/adaccounts",
      "likes": "https://graph.facebook.com/100003810151884/likes",
      "posts": "https://graph.facebook.com/100003810151884/posts",
      "tagged": "https://graph.facebook.com/100003810151884/tagged",
      "statuses": "https://graph.facebook.com/100003810151884/statuses",
      "links": "https://graph.facebook.com/100003810151884/links",
      "notes": "https://graph.facebook.com/100003810151884/notes",
      "photos": "https://graph.facebook.com/100003810151884/photos",
      "albums": "https://graph.facebook.com/100003810151884/albums",
      "events": "https://graph.facebook.com/100003810151884/events",
      "groups": "https://graph.facebook.com/100003810151884/groups",
      "videos": "https://graph.facebook.com/100003810151884/videos",
      "picture": "https://graph.facebook.com/100003810151884/picture",
      "inbox": "https://graph.facebook.com/100003810151884/inbox",
      "outbox": "https://graph.facebook.com/100003810151884/outbox",
      "updates": "https://graph.facebook.com/100003810151884/updates",
      "accounts": "https://graph.facebook.com/100003810151884/accounts",
      "checkins": "https://graph.facebook.com/100003810151884/checkins",
      "apprequests": "https://graph.facebook.com/100003810151884/apprequests",
      "friendlists": "https://graph.facebook.com/100003810151884/friendlists",
      "friendrequests": "https://graph.facebook.com/100003810151884/friendrequests",
      "permissions": "https://graph.facebook.com/100003810151884/permissions",
      "notifications": "https://graph.facebook.com/100003810151884/notifications",
      "scores": "https://graph.facebook.com/100003810151884/scores"
    }
  },
  "fields": [
    {
      "name": "id",
      "description": "The user's Facebook ID. No `access_token` required. `string`."
    }
  ]
}
```

***Σχήμα 17: Χρήση της παραμέτρου “metadata=1”. Εδώ φαίνονται όλες οι υποστηριζόμενες συσχετίσεις για το συγκεκριμένο αντικείμενο.***

### **1.13 ΕΝΗΜΕΡΩΣΕΙΣ ΠΡΑΓΜΑΤΙΚΟΥ ΧΡΟΝΟΥ (REAL TIME UPDATES).**

Ο Γράφος API υποστηρίζει τις ενημερώσεις πραγματικού χρόνου για να δώσει τη δυνατότητα στην εφαρμογή μας που χρησιμοποιεί το Facebook να συνδράμει στις αλλαγές των δεδομένων από το Facebook.

Έτσι, παρέχεται η δυνατότητα λήψης ενημερώσεων για όλους τους χρήστες της εφαρμογής μας, όσο τα δεδομένα τους αλλάζουν. Και είμαστε βέβαιοι ότι τα cached δεδομένα είναι ορθά, χωρίς να χρειάζεται να αναφερθούμε στους server του Facebook. Αυτό σημαίνει ότι βελτιώνεται η αξιοπιστία της εφαρμογής μας και η άμεση ανταπόκριση όσον αφορά την εμπειρία του χρήστη.

### 1.14 ΔΗΜΟΣΙΕΥΣΕΙΣ.

Μπορούμε να δημοσιεύσουμε στον Γράφο του Facebook, κάνοντας κλήσεις HTTP POST σε κατάλληλα URL συσχέτισεων, χρησιμοποιώντας κάποιο τεκμήριο πρόσβασης. Για παράδειγμα, μπορούμε να αποστείλουμε μία νέα ανακοίνωση στον Τοίχο του Antwnios Kyrilas, κάνοντας μια POST κλήση στο <https://graph.facebook.com/antwnios.kyrilas/feed>:

```
curl -F 'access_token=...' \  
-F 'message=Hello, Arjun. I like this new API.' \  
https://graph.facebook.com/antwnios.kyrilas/feed
```

Μπορούμε να σχολιάσουμε ή να κάνουμε “like” σε οποιοδήποτε αντικείμενο το οποίο έχει συσχέτιση “/comments” ή “/likes”, κάνοντας αποστολή στα [https://graph.facebook.com/OBJECT\\_ID/comments](https://graph.facebook.com/OBJECT_ID/comments) και [https://graph.facebook.com/OBJECT\\_ID/likes](https://graph.facebook.com/OBJECT_ID/likes) αντίστοιχα. Π.χ.:

```
curl -F 'access_token=...' \  
https://graph.facebook.com/313449204401/likes
```

Οι περισσότερες λειτουργίες εγγραφής απαιτούν εκτεταμένες άδειες για τον ενεργό χρήστη. Υποστηρίζεται η εγγραφή για τους ακόλουθους τύπους αντικειμένων που εμφανίζονται στον πίνακα 1.1.

<b>Μέθοδος</b>	<b>Περιγραφή</b>	<b>Παράμετροι</b>
/PROFILE_ID/feed	Δημοσίευση μιας νέας ανακοίνωσης (post) στον Τοίχο ή στις Ενημερώσεις ενός δοσμένου προφίλ	message, picture, link, name, caption, description, source, place, tags
/OBJECT_ID/comments	Σχόλιο στο δοσμένο αντικείμενο (εάν έχει μία συσχέτιση /comments)	message
/OBJECT_ID/likes	Κάνει “like” στο δοσμένο αντικείμενο (εάν έχει μία συσχέτιση /likes)	καμία

Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

<b>Μέθοδος</b>	<b>Περιγραφή</b>	<b>Παράμετροι</b>
/PROFILE_ID/notes	Δημοσίευση μίας σημείωσης στο δοσμένο προφίλ	message, subject
/PROFILE_ID/links	Δημοσίευση ενός συνδέσμου (link) στο δοσμένο προφίλ	link, message, picture, name, caption, description
/PROFILE_ID/events	Δημιουργία μίας εκδήλωσης (event) στο δοσμένο προφίλ	name, start_time, end_time
/EVENT_ID/attending	RSVP “attending” στο δοσμένο γεγονός	καμία
/EVENT_ID/maybe	RSVP “maybe” στο δοσμένο γεγονός	καμία
/EVENT_ID/declined	RSVP “declined” στο δοσμένο γεγονός	καμία
/PROFILE_ID/albums	Δημιουργία ενός άλμπουμ	name, message
/ALBUM_ID/photos	Ανέβασμα μίας φωτογραφίας σε ένα άλμπουμ	message, source, place (multipart/form-data)
/PROFILE_ID/checkins	Δημιουργία chekin σε τοποθεσία που αναπαρίσταται από μία Σελίδα (Page)	coordinates, place, message, tags

Πίνακας 1: «Λειτουργίες εγγραφής σε αντικείμενα»

### **1.15 ΔΙΑΓΡΑΦΗ.**

Μπορούμε να διαγράψουμε ένα αντικείμενο του Γράφου, κάνοντας HTTP DELETE αιτήματα στα URL των αντικειμένων, π.χ.:

```
DELETE https://graph.facebook.com/ID?access_token=... HTTP/1.1
```

Για την υποστήριξη των client που δεν υποστηρίζουν όλες τις HTTP μεθόδους (όπως Javascript client), μπορούμε εναλλακτικά να κάνουμε ένα POST αίτημα σε ένα URL αντικείμενο με την ανάλογη παράμετρο “method=delete” για να παρακάμψουμε την HTTP μέθοδο. Για παράδειγμα, μπορούμε να διαγράψουμε ένα σχόλιο, κάνοντας ένα αίτημα POST στο:

[https://graph.facebook.com/COMMENT\\_ID?method=delete](https://graph.facebook.com/COMMENT_ID?method=delete)

Μπορούμε να διαγράψουμε ένα “like”, κάνοντας ένα DELETE αίτημα στο /OBJECT\_ID/likes (αφού τα “like” δεν έχουν ID). Αυτό σημαίνει ότι θα εμφανίζεται πάλι το μήνυμα “Like” για το συγκεκριμένο αντικείμενο και ότι μπορούμε στο μέλλον να ξανακάνουμε “Like” σε αυτό.

## **1.16 ΑΝΑΛΥΣΕΙΣ.**

Όταν καταχωρούμε την εφαρμογή μας, με την επιλογή “Insights” (Βαθιά Γνώση) μπορούμε να λάβουμε λεπτομερείς αναλύσεις σχετικά με τα δημογραφικά στοιχεία των χρηστών μας και πως οι χρήστες κάνουν “share” από την εφαρμογή μας.

Ο Γράφος παρέχει πρόσβαση σε όλα αυτά τα δεδομένα, έτσι ώστε να μπορούμε να ενοποιούμε τα δεδομένα της Πλατφόρμας μέσα στα δικά μας αναλυτικά συστήματα.

Για να κατεβάσουμε τα δεδομένα του “Insights”, πρώτα θα πρέπει να αποκτήσουμε ένα τεκμήριο πρόσβασης εφαρμογής (“app access token”). Τότε μπορούμε να κατεβάσουμε δεδομένα αναλύσεων για την εφαρμογή μας στη διεύθυνση:

[https://graph.facebook.com/app\\_id/insights?access\\_token=...](https://graph.facebook.com/app_id/insights?access_token=...)

Αυτό το URL εμφανίζει όλα τα δεδομένα των αναλύσεων που είναι διαθέσιμα μέσα από το API, περιλαμβανομένου του πλήθους των χρηστών, του πλήθους των ενεργών χρηστών, και άλλες αναλυτικές μετρήσεις. Για παράδειγμα, μπορούμε να έχουμε το πλήθος των εμφανίσεων από την σελίδα του καμβά της εφαρμογής μας:

[https://graph.facebook.com/app\\_id/insights/application\\_canvas\\_views/day?access\\_token=...](https://graph.facebook.com/app_id/insights/application_canvas_views/day?access_token=...)

Κάνουμε χρήση του “since” και “until” για να συγκεκριμενοποιήσουμε το χρονικό εύρος για το οποίο θέλουμε τα δεδομένα μας. Και οι δύο παράμετροι δέχονται τον χρόνο σε όλες σχεδόν τις δυνατές μορφές:

[https://graph.facebook.com/app\\_id/insights?access\\_token=...&since=yesterday](https://graph.facebook.com/app_id/insights?access_token=...&since=yesterday).



### **1.17 ΟΜΑΔΙΚΑ ΑΙΤΗΜΑΤΑ (BATCH REQUESTS).**

Εάν η εφαρμογή μας χρειάζεται να έχει τη δυνατότητα να έχει πρόσβαση σε σημαντικό αριθμό δεδομένων μονομιάς ή είναι ανάγκη να κάνει αλλαγές σε πολλά αντικείμενα τη φορά, είναι συχνά πιο αποτελεσματικό το να ομαδοποιήσουμε τα αιτήματά μας συγκρίνοντας αυτές τις πράξεις, αντί να κάνουμε πολλαπλά ατομικά HTTP αιτήματα.

Για να γίνει αυτό, ο Γράφος υποστηρίζει την ομαδοποίηση αιτημάτων. Τα ομαδικά αιτήματα επιτρέπουν το πέρασμα οδηγιών για πολλές λειτουργίες με ένα μοναδικό αίτημα HTTP. Μπορούμε δηλαδή, αντί για πολλαπλά αιτήματα Γράφου, να φτιάξουμε ένα JSON (βλέπε Λεξιλόγιο) αντικείμενο το οποίο τα αναπαριστάνει, και το οποίο κάνουμε POST για να λάβουμε την ενοποιημένη απάντηση. Αυτό είναι καλό γιατί όταν κάνουμε πολλαπλές αιτήσεις, αυτό μας επιτρέπει να το κάνουμε με παράλληλο και αποδοτικό τρόπο.

### **1.18 ΚΑΘΟΡΙΖΟΝΤΑΣ ΤΗΝ ΕΝΤΟΠΙΟΤΗΤΑ.**

Εάν η εφαρμογή μας χρειάζεται να έχει τη δυνατότητα ανάκλησης περιορισμένου τοπικά περιεχομένου, στη γλώσσα μιας συγκεκριμένης τοποθεσίας, προσθέτουμε την παράμετρο “locale”:

[https://graph.facebook.com/...?access\\_token=...&locale=LOCALE](https://graph.facebook.com/...?access_token=...&locale=LOCALE)

Οι τοποθεσίες που υποστηρίζει το Facebook είναι διαθέσιμες στο XML αρχείο: <http://www.facebook.com/translations/FacebookLocales.xml>.

### **1.19 ΣΕΛΙΔΟΠΟΙΗΣΕΙΣ (PAGING).**

Όταν αποστέλλουμε αιτήματα στις συσχετίσεις, υπάρχουν μερικές χρήσιμες παράμετροι, οι οποίες μας επιτρέπουν να φιλτράρουμε και να κάνουμε σελιδοποίηση των δεδομένων της συσχέτισης:

- limit, offset: <https://graph.facebook.com/me/likes?limit=3>
- until, since (μία unix χρονοσφραγίδα ή οποιαδήποτε ημερομηνία αποδεκτή από το strtotime): <https://graph.facebook.com/search?until=yesterday&q=orange>

Η παράμετρος “limit” ορίζει το πόσα αποτελέσματα θέλουμε να εμφανιστούν στη σελίδα μας (στο πρώτο παράδειγμα 3 αποτελέσματα ανά σελίδα).

## **ΕΠΙΛΟΓΟΣ**

Σε αυτό το κεφάλαιο είδαμε την γενική ορολογία την οποία θα πρέπει να γνωρίζει ο

Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

επιστήμονας πληροφορικής ο οποίος θα χρειαστεί να αναπτύξει κάποια εφαρμογή που θα χρησιμοποιεί στοιχεία από το Facebook και τα διακριτά στοιχεία από τα οποία δομείται το Facebook. Στη συνέχεια θα δούμε το πώς αυτά χρησιμοποιούνται από το API του Facebook.

## ΚΕΦΑΛΑΙΟ 2

### ΒΑΣΙΚΑ ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΤΟΥ FACEBOOK API

#### ***ΕΙΣΑΓΩΓΗ***

Σε αυτό το κεφάλαιο παρουσιάζονται τα κύρια χρηστικά στοιχεία του Facebook

Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

API. Αναφέρονται αναλυτικά τόσο η Πλατφόρμα του Facebook για τον προγραμματισμό στο ίδιο το Facebook, όσο και ο τρόπος χρήσης κάποιων δομών του Facebook για τον προγραμματισμό δημιουργίας ιστοσελίδων.

## **2.1 Η ΠΛΑΤΦΟΡΜΑ ΤΟΥ FACEBOOK.**

Η Πλατφόρμα του Facebook γνώρισε τεράστια διόγκωση τα τελευταία χρόνια. Από μία απλή διαδικτυακή εφαρμογή έφτασε να γίνει ο πρώτος κοινωνικός ιστοχώρος στο παγκόσμιο Διαδίκτυο.

Το 2007 το Facebook δημιούργησε τη δική του Πλατφόρμα ανάπτυξης εφαρμογών. Αυτή περιλαμβάνει μία συμβολική γλώσσα βασισμένη σε HTML που λέγεται Facebook Markup Language (FBML), ένα API για τη δημιουργία κλήσεων

αναπαραστατικής μεταφοράς κατάστασης (representational state transfer calls – REST calls) στο Facebook, μία γλώσσα ερωτημάτων που ομοιάζει με SQL για την επικοινωνία με το Facebook και η οποία ονομάζεται Facebook Query Language (FQL), μία γλώσσα script που καλείται Facebook Javascript για τον εμπλουτισμό της εμπειρίας του χρήστη και ένα σύνολο βιβλιοθηκών προγραμματισμού client. Γενικά τα εργαλεία που φτιάχνουν την Πλατφόρμα του Facebook συνήθως αποκαλούνται απλά ως Facebook API.

Το Facebook, δημιουργώντας αυτή την Πλατφόρμα, έφτιαξε μία μηχανή η οποία επιτρέπει σε αυτούς που δημιουργούν εφαρμογές να δημιουργήσουν εξωτερικές εφαρμογές για να ενδυναμώσουν τους χρήστες του στο να αλληλεπιδρούν ο ένας με τον άλλο με νέους και συναρπαστικούς τρόπους – τρόπους που ο προγραμματιστής καλείται να ανακαλύψει. Υπάρχει η δυνατότητα – εκτός από να φτιάξουμε διαδικτυακές εφαρμογές – να δημιουργήσουμε Desktop εφαρμογές συνδεδεμένες με το Διαδίκτυο (Internet-connected desktop application) με τη βιβλιοθήκη Java client.

## **2.2 ΤΑ ΣΤΟΙΧΕΙΑ ΤΗΣ ΠΛΑΤΦΟΡΜΑΣ ΚΑΙ ΕΦΑΡΜΟΓΕΣ ΣΤΟ FACEBOOK.**

Φτιάχνοντας μία εφαρμογή στο Facebook μας δίνει την ευκαιρία να ολοκληρώσουμε σε βάθος την εμπειρία του πυρήνα του Facebook. Η εφαρμογή μας μπορεί να ενοποιηθεί με πολλές πλευρές του Facebook.com, περιλαμβάνοντας τις Ενημερώσεις (News Feed) και τις Ειδοποιήσεις (Notification). Όλος ο πυρήνας των τεχνολογιών της Πλατφόρμας του Facebook, όπως τα

κοινωνικά Plugin, ο Γράφος API και οι Διάλογοι Πλατφόρμας (Platform Dialogs) είναι διαθέσιμοι στις εφαρμογές του Facebook.

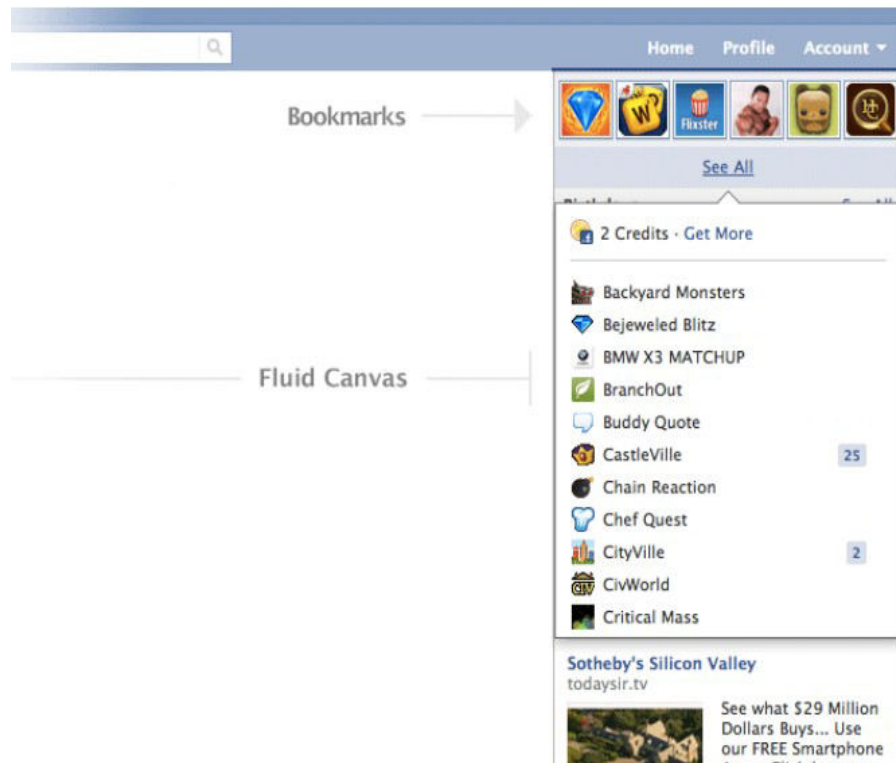
Σε αυτή την παράγραφο θα εξηγήσουμε τις εξής έννοιες:

- Σελίδα του Καμβά (Canvas Page): Η Σελίδα στην οποία η εφαρμογή μας φορτώνεται.
- Κοινωνικά Κανάλια (Social Channel): Τα κανάλια-κλειδιά που μας βοηθούν να αναπτύξουμε την βάση του χρήστη μας και να απασχολήσουν ξανά υπάρχοντες χρήστες.
- Αναλύσεις (Analytics): Πρόσβαση σε αναλυτικά στοιχεία της εφαρμογής μας.

### **Σελίδα του Καμβά (Canvas Page).**

Οι εφαρμογές στο Facebook είναι Διαδικτυακές εφαρμογές που φορτώνονται στο γενικό πλαίσιο του Facebook σε ό,τι αναφέρουμε ως Σελίδα του Καμβά (Σχήμα 18).

Υπάρχει η επιλογή του να μπορούμε να δημιουργήσουμε την δική μας εφαρμογή χρησιμοποιώντας οποιαδήποτε γλώσσα ή αλυσίδα εργαλείων που υποστηρίζει τον διαδικτυακό προγραμματισμό, όπως PHP, Python, Java ή C#.



Σχήμα 18: Η Σελίδα του Καμβά

### **Ο Καμβάς για την εφαρμογή μας.**

Οι εφαρμογές στο Facebook φορτώνονται μέσα στη Σελίδα του Καμβά. Μία Σελίδα του Καμβά είναι σχεδόν κυριολεκτικά ένας λευκός καμβάς (όπως αυτός ενός ζωγράφου) μέσα στο Facebook στον οποίο θα τρέξουμε την εφαρμογή μας. Δημιουργούμε τη Σελίδα του Καμβά ορίζοντας ένα URL του Καμβά το οποίο περιέχει τον κώδικα HTML, Javascript και CSS που φτιάχνουν την εφαρμογή μας. Όταν ένας χρήστης καλεί την Σελίδα του Καμβά, φορτώνεται το URL του Καμβά σε ένα iframe (βλέπε Λεξιλόγιο) σε εκείνη τη σελίδα. Αυτό έχει ως αποτέλεσμα να εμφανίζεται η εφαρμογή μας.

Εξ ορισμού ο καμβάς έχει 760 pixel μήκος, αλλά μπορούμε να το θέσουμε έτσι ώστε να είναι αριστερά ευθυγραμμισμένο και να λαμβάνει όλο το μήκος και



Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

ύψος του περιηγητή ιστοσελίδων του χρήστη.

### **Σελιδοδείκτες για γρήγορη επαναφορά.**

Όταν ο χρήστης ξεκινήσει να χρησιμοποιεί την εφαρμογή μας, δημιουργούμε έναν σελιδοδείκτη για να δώσουμε τη δυνατότητα στους χρήστες να περιηγηθούν εύκολα πίσω στην εφαρμογή μας μέσα από το Facebook. Αυτοί οι σελιδοδείκτες είναι διαθέσιμοι στην Αρχική Σελίδα του Facebook, όπως επίσης και στη Σελίδα του Καμβά – στην πάνω δεξιά γωνία.

Αυτοί οι σελιδοδείκτες είναι ταξινομημένοι βάσει του πόσο συχνά και πρόσφατα ο χρήστης χρησιμοποίησε την εφαρμογή μας. Οι μετρητές στους σελιδοδείκτες ενημερώνουν τους χρήστες για εκκρεμείς κλήσεις σχετιζόμενες με το πρότυπο “Requests 2.0” για τις εφαρμογές μας και τους ενθαρρύνει να ανταποκριθούν. (Περισσότερα στην επόμενη παράγραφο).

### **Κοινωνικά Κανάλια (Social Channel).**

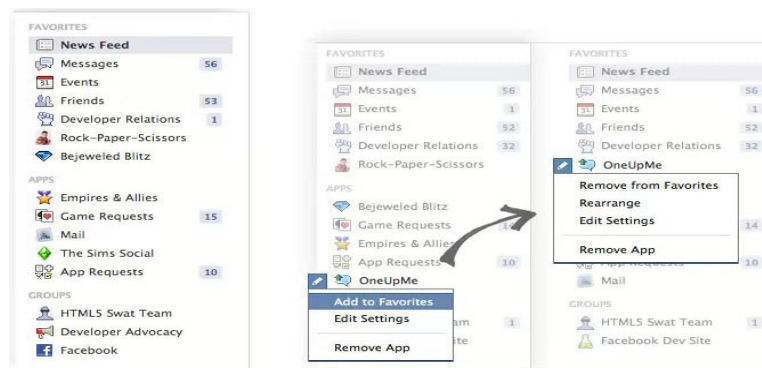
Για να δρομολογηθεί περισσότερη κίνηση στις εφαρμογές του Facebook, έχουν ενεργοποιηθεί ποικίλα κανάλια τα οποία δίνουν τη δυνατότητα στους νέους χρήστες να ανακαλύψουν την εφαρμογή μας όπως και στους υπάρχοντες χρήστες των εφαρμογών μας να ασχοληθούν πάλι με αυτές. Κάθε κανάλι είναι σχεδιασμένο για να βοηθά τους χρήστες και οι αλγόριθμοι του Facebook βοηθούν να έρθει στην επιφάνεια το καλύτερο περιεχόμενο για κάθε χρήστη. Τα σημερινά κανάλια περιλαμβάνουν τα εξής:

#### **Σελιδοδείκτες.**

Οι Σελιδοδείκτες ενεργοποιούν τους χρήστες να περιηγηθούν εύκολα πίσω στην εφαρμογή μας μέσα από το Facebook. Οι Σελιδοδείκτες προστίθενται

## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

αυτόματα για την εφαρμογή μας, όταν οι χρήστες ασχολούνται με αυτήν. Ο Σελιδοδείκτης θα εμφανίζεται στην αριστερή στήλη της Αρχικής Σελίδας, όπως επίσης και στο επάνω δεξί μέρος μίας Σελίδας Καμβά. Οι χρήστες μπορούν να προσθέσουν έναν Σελιδοδείκτη ως “Αγαπημένο” (favorite), πράγμα που τον “καρφιτσώνει” ψηλά. Οι Σελιδοδείκτες εμφανίζουν επίσης έναν μετρητή δίπλα τους, ο οποίος αναπαριστά τον αριθμό των εκκρεμών κλήσεων του χρήστη για την εφαρμογή μας. Η διασύνδεση “more” δίπλα στους Σελιδοδείκτες των εφαρμογών μεταφέρει τον χρήστη στο ταμπλό των εφαρμογών, το οποίο δείχνει τις εκκρεμείς κλήσεις των εφαρμογών, όπως και τον πλήρη κατάλογο των εφαρμογών τις οποίες ο χρήστης έχει αυθεντικοποιήσει. Στο Σχήμα 19 βλέπουμε το πώς προσθέτουμε ή αφαιρούμε μία εφαρμογή στα Αγαπημένα (Favorites).



Σχήμα 19: Αγαπημένες εφαρμογές και μετρητής Σελιδοδείκτη

### **Ειδοποιήσεις (Notification).**

Οι Ειδοποιήσεις στο Facebook παρέχουν πληροφορίες για ενδιαφέρουσες αλλαγές σε περιεχόμενο σχετικό με τους χρήστες. Κλήσεις (Request) από την εφαρμογή μας μπορεί να έχουν ως έναυσμα μία ειδοποίηση σε χρήστες που θα την βρουν σχετική (Σχήμα 20).



Σχήμα 20: Ειδοποιήσεις

### **Κλήσεις (Request).**

Οι Κλήσεις είναι ένας καλός τρόπος για να κάνουμε τους χρήστες να καλέσουν τους φίλους τους στην εφαρμογή μας. Μπορούμε επίσης να χρησιμοποιήσουμε τις Κλήσεις για να κάνουμε έναν χρήστη να ειδοποιήσει τους φίλους του για να κάνουν μία συγκεκριμένη πράξη στην εφαρμογή μας, όπως να δεχτούν ένα δώρο ή να βοηθήσουν τον χρήστη να συμπληρώσει μία αναζήτηση.

Υπάρχουν δύο τύποι Κλήσεων που μπορούν να αποσταλούν από μία εφαρμογή:

– Κλήσεις παραγόμενες από τον χρήστη (User-generated request): αυτές οι Κλήσεις έχουν επιβεβαιωθεί από μία σαφή ενέργεια ενός χρήστη σε έναν διάλογο απόκρισης. Αυτές οι Κλήσεις ενημερώνουν τον μετρητή Σελιδοδείκτη για τον παραλήπτη.

– Κλήσεις παραγόμενες από την εφαρμογή (App-generated request): Αυτές οι Κλήσεις μπορούν να αρχικοποιηθούν και να σταλούν μόνο σε χρήστες που έχουν αυθεντικοποιήσει την εφαρμογή μας. Πρέπει να χρησιμοποιούμε αυτές τις Κλήσεις για να ενημερώνουμε τον μετρητή του Σελιδοδείκτη έτσι ώστε να ενθαρρύνουμε τον χρήστη να ασχοληθεί ξανά με την εφαρμογή μας (π.χ. ο φίλος μας τελείωσε την κίνησή του σε ένα παιχνίδι και τώρα είναι η σειρά μας).

### **Ιστορικό Ενημερώσεων (Newsfeed story).**

Οι Ενημερώσεις είναι εμφανείς άμεσα στους χρήστες με την είσοδο στο Facebook, κάνοντάς τις αρκετά σημαντικές στην εμπειρία του Facebook.

- *Ιστορικά Ανακαλύψεων (Discovery stories).*

Το Facebook μοιράζεται το ιστορικό με φίλους όταν ένας χρήστης αρχίζει να χρησιμοποιεί μία νέα εφαρμογή ή εγκαθιστά ένα νέο παιχνίδι. Αυτά τα Ιστορικά Ανακάλυψης εγκατάστασης εμφανίζονται στις Ενημερώσεις των φίλων και κάνουν τους χρήστες να ανακαλύπτουν δημοφιλείς εφαρμογές που χρησιμοποιούν οι φίλοι τους (Σχήμα 21).



*Σχήμα 21: Ιστορικά ανακαλύψεων*

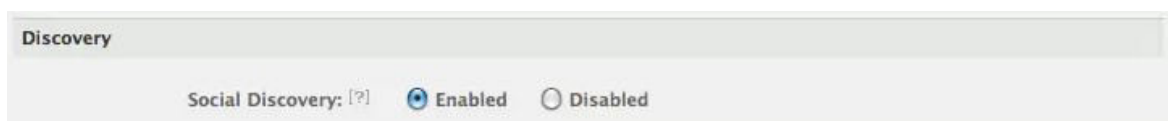
Επιπρόσθετα, το Facebook παράγει επίσης αθροιστικό ιστορικό των παιχνιδιών που έχουμε παίξει. Αυτά τα ιστορικά εμφανίζονται στις Ενημερώσεις των φίλων (Σχήμα 22). Οι παίκτες παιχνιδιών ή ακόμη και αυτοί που δεν παίζουν παιχνίδια θα δουν ιστορικά που καταγράφουν παιχνίδια που παίζουν περισσότερο οι φίλοι τους. Αυτά τα ιστορικά είναι σχεδιασμένα για να φέρνουν νέους χρήστες και να ασχολούνται ξανά οι υπάρχοντες με παιχνίδια, ενθαρρύνοντας τους χρήστες να ανακαλύψουν παιχνίδια που παίζουν οι φίλοι τους.

## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού



Σχήμα 22: Ιστορικό παιχνιδιών φίλων

Οι δημιουργοί εφαρμογών που σχετίζονται με πιο προσωπική συμπεριφορά των χρηστών τους (δηλαδή προσωπικά δεδομένα αυτών που μπορεί να μη θέλουν να δημοσιεύονται), όπως ραντεβού, διαχείριση βάρους ή εγκυμοσύνη, μπορούν να σβήσουν αυτά τα ιστορικά απενεργοποιώντας το πεδίο Κοινωνικής Ανακάλυψης (Social Discovery) στο Ταμπλό Εφαρμογών (App Dashboard) στο πλαίσιο “On Facebook” κάτω από το τμήμα του Καμβά (Σχήμα 23).



Σχήμα 23: (Απ)ενεργοποιώντας την Κοινωνική Ανακάλυψη

- Δημοσιεύσεις Ιστορικών (Publishing story).

Επιπρόσθετα, μπορούμε να δημοσιεύσουμε περιεχόμενο χρησιμοποιώντας

τον Διάλογο Ενημερώσεων (Feed Dialog). Μπορούμε να προτρέψουμε τους χρήστες να δημοσιεύσουν ιστορικά σχετικά με το τι κάνουν στην εφαρμογή μας. Εάν ένας χρήστης επιλέγει τη δημοσιοποίηση, το ιστορικό θα φαίνεται στο προφίλ του και μπορεί να φαίνεται και στις Ενημερώσεις των φίλων του χρήστη (Σχήμα 24). Ιστορικά που δημοσιεύονται από την εφαρμογή μας θα συμπεριλαμβάνουν έναν σύνδεσμο προς την εφαρμογή μας και προαιρετικά μπορεί να περιλαμβάνουν ένα πλούσιο συνημμένο.



Σχήμα 24: Δημοσιεύοντας Ιστορικά

Ο Διάλογος Ενημερώσεων προτρέπει τον χρήστη να δημοσιεύσει ένα ξεχωριστό ιστορικό στις Ενημερώσεις ενός προφίλ. Αυτό δεν απαιτεί καμία εκτεταμένη άδεια. Η εφαρμογή μας μπορεί επίσης να δημοσιεύσει απευθείας στις Ενημερώσεις ενός προφίλ χωρίς την αλληλεπίδραση του χρήστη, όπως φαίνεται στο παρακάτω παράδειγμα σε Javascript, στο οποίο χρησιμοποιείται η μέθοδος FB.ui σε JavaScript SDK:

```
<html xmlns="http://www.w3.org/1999/xhtml"
  xmlns:fb="https://www.facebook.com/2008/fbml">
  <head>
    <title>My Feed Dialog Page</title>
  </head>
  <body>
```

## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

```
<div id='fb-root'></div>
<script src='http://connect.facebook.net/en_US/all.js'></script>
<p><a onclick='postToFeed(); return false;'>Post to Feed</a></p>
<p id='msg'></p>

<script>
  FB.init({appId: "YOUR_APP_ID", status: true, cookie: true});

  function postToFeed() {

    // calling the API ...
    var obj = {
      method: 'feed',
      link: 'https://developers.facebook.com/docs/reference/dialogs/',
      picture: 'http://fbrell.com/f8.jpg',
      name: 'Facebook Dialogs',
      caption: 'Reference Documentation',
      description: 'Using Dialogs to interact with users.'
    };

    function callback(response) {
      document.getElementById('msg').innerHTML = "Post ID: " + response['post_id'];
    }

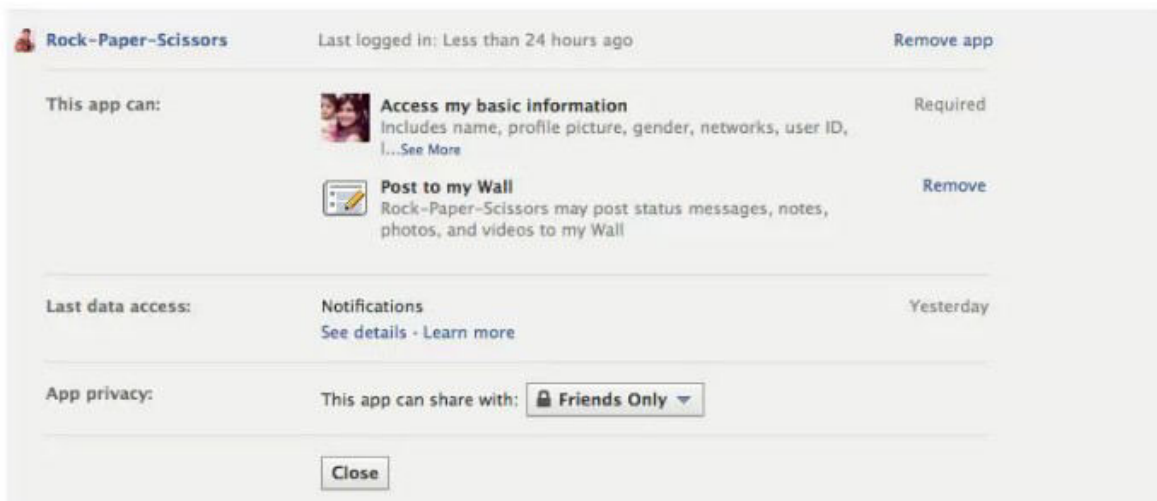
    FB.ui(obj, callback);
  }

</script>
</body>
</html>
```

Ας σημειώσουμε ότι το παραπάνω παράδειγμα υποθέτει ότι ο χρήστης είναι ήδη συνδεδεμένος και έχει εξουσιοδοτήσει την εφαρμογή μας.

- Έλεγχος Χρήστη (User Control).

Στη Σελίδα Ιδιοτήτων Εφαρμογής (Application Settings Page) οι χρήστες μπορούν να ελέγξουν το μέγιστο κοινό το οποίο η κάθε εφαρμογή μπορεί να μοιράζεται με αυτούς. Έτσι, εάν οι χρήστες θέλουν να βλέπουν οι υπόλοιποι ότι παίζουν ένα παιχνίδι, αλλά όχι κάποιο άλλο, μπορούν να το καθορίσουν (Σχήμα 25).



Σχήμα 25: Έλεγχος Χρήστη

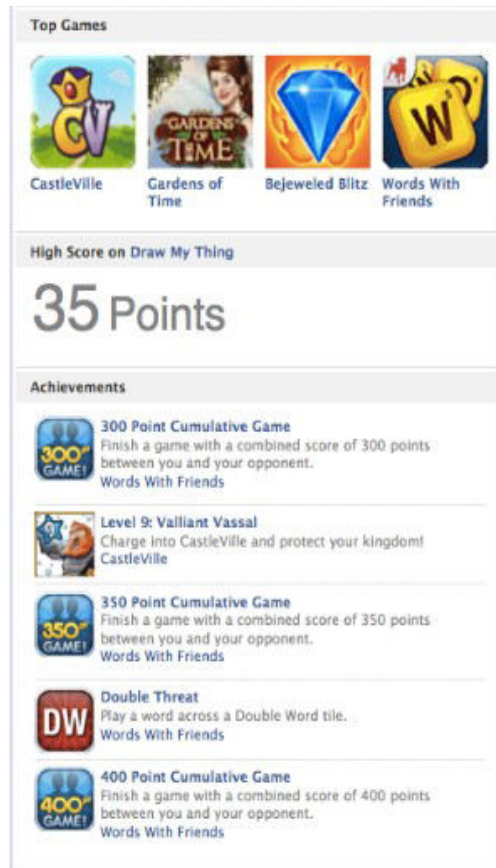
### Χρονοδιάγραμμα (Timeline)

Το χρονοδιάγραμμα του χρήστη (Σχήμα 26) ενοποιεί σε βάθος τη δραστηριότητα του παιχνιδιού που παίζει περισσότερο σε μία δοσμένη περίοδο χρόνου. Αυτή η μονάδα χρονοδιαγράμματος των παιχνιδιών τονίζει τα παιχνίδια που οι χρήστες παίζουν περισσότερο, όπως και τις πιο υψηλές βαθμολογίες και τα επιτεύγματα, επιτρέποντας γρήγορα το να ασχοληθούν ξανά με αυτά και την



Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

ανακάλυψή τους μέσα από τους φίλους.



Σχήμα 26: Χρονοδιάγραμμα

Οι προγραμματιστές των εφαρμογών που κατηγοριοποιούνται ως Παιχνίδια μπορούν να δημοσιοποιούν τα Ιστορικά χρησιμοποιώντας τις ενσωματώσεις του Ανοιχτού Γράφου για τα επιτεύγματα και τις βαθμολογίες (Σχήμα 27). Αυτά τα Ιστορικά φαίνονται στους χρήστες μόνο όταν χρησιμοποιούν μία εφαρμογή που κατηγοριοποιείται ως Παιχνίδι.

Τα επιτεύγματα στο API (Achievement API) (για περισσότερες πληροφορίες, βλέπε Λεξικό) μας επιτρέπουν να δημοσιοποιούμε Ιστορικά σχετικά με έναν χρήστη και τους φίλους του που πετυχαίνουν ένα σύνολο από επιτεύγματα που

ορίζουμε για την εφαρμογή μας. Οι βαθμολογίες API (score API) (για περισσότερες πληροφορίες, βλέπε Λεξικό) μας επιτρέπουν να δημοσιοποιούμε Ιστορικά σχετικά με έναν χρήστη και τους φίλους του διατρέχοντας τις βαθμολογίες όλων στην εφαρμογή μας όπως και τα Ιστορικά των υψηλών βαθμολογιών όταν ο χρήστης καταφέρνει μία νέα υψηλή βαθμολογία. Ο χρήστης χρειάζεται να παραχωρήσει στην εφαρμογή μας την publish\_actions άδεια πριν δημοσιεύσουμε τις βαθμολογίες και τα επιτεύγματά του.

Επιπρόσθετα, το Χρονοδιάγραμμα φέρνει στο προσκήνιο την πρόσφατη δραστηριότητα παιχνιδιών του χρήστη.



Σχήμα 27: Πρόσφατη δραστηριότητα

### **Αναζήτηση (Search)**

Όταν η εφαρμογή μας φτάσει τους δέκα μηνιαίους ενεργούς χρήστες, αυτόματα θα μπει στη σειρά για να συμπεριληφθεί στον δείκτη αναζήτησης για την επόμενη φορά που γίνει επανασύνταξη του δείκτη αναζήτησης του Facebook (αυτό γίνεται κάθε 2-4 εβδομάδες). Αφού ο δείκτης επανασυνταχθεί, η εφαρμογή μας θα είναι δυνατόν να ανακαλυφθεί μέσω της Αναζήτησης (Σχήμα 28).



Σχήμα 28: Αναζήτηση (Search)

### Σελίδα (Page)

Επιπροσθέτως, μπορούμε να δημιουργήσουμε μία Σελίδα στο Facebook για την εφαρμογή μας από το Ταμπλό Εφαρμογών (App Dashboard). Απλά θα πρέπει να επισκεφθούμε το τμήμα Πληροφοριών Επικοινωνίας (Contact Info) στο Advanced tab του Ταμπλό Εφαρμογών και να κάνουμε κλικ στο πλήκτρο “Create Facebook Page” για να δημιουργηθεί μία νέα Facebook Σελίδα (Σχήμα 29).



Σχήμα 29: Σελίδα (Page)

Πατώντας στο Confirm, στο επόμενο πλαίσιο διαλόγου, συσχετίζεται η νέα Σελίδα Facebook με την εφαρμογή μας.

Η Σελίδα διαφέρει ελαφρώς από τις συνηθισμένες Σελίδες. Θα έχει ένα πλήκτρο “Go To App” ή “Play Game”, όπως επίσης τους επόμενους συνδέσμους σχετιζόμενους με την εφαρμογή μας:

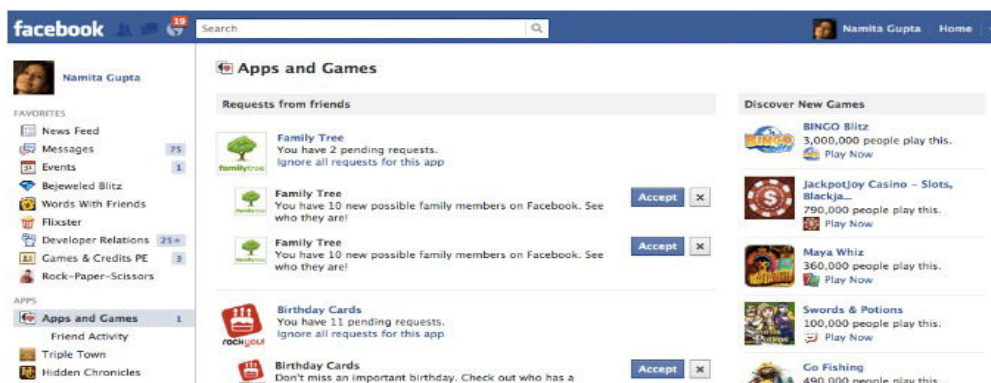
## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

- Φραγή Εφαρμογής (Block App)
- Αφαίρεση Εφαρμογής (Remove App)
- Επικοινωνία με τον δημιουργό (Contact Developer)
- Αναφορά Εφαρμογής (Report App)

Εξ' ορισμού η Σελίδα έχει τίτλο “[TO\_ONOMA\_ΤΗΣ] Community”. Μπορούμε να τροποποιήσουμε αυτό το όνομα κάθε στιγμή με την επιφύλαξη των ίδιων κανόνων που καθορίζουν όλες τις Σελίδες.

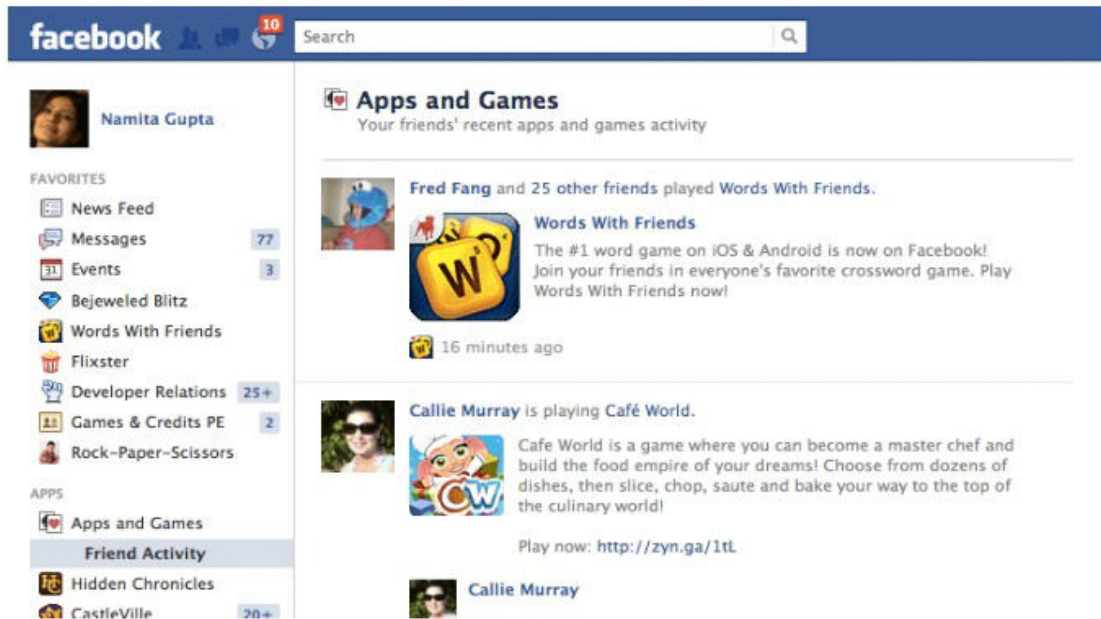
### Ταμπλό Παιχνιδιών και Εφαρμογών (Games and Apps Dashboard).

Το Facebook έχει ένα Ταμπλό Παιχνιδιών και Εφαρμογών το οποίο επιτρέπει στους χρήστες να δουν εκκρεμείς κλήσεις, όπως και προτινόμενες Εφαρμογές και Παιχνίδια για να τους δώσει τη δυνατότητα να ανακαλύψουν καινούργια παιχνίδια, όπως και να ασχοληθούν και πάλι με τα υπάρχοντα παιχνίδια (Σχήμα 30).



Σχήμα 30: Ταμπλό Εφαρμογών και Παιχνιδιών

Οι χρήστες εδώ, μπορούν επίσης να έχουν πρόσβαση σε όλες τις ενημερώσεις των εφαρμογών και των παιχνιδιών σε ένα μέρος (Σχήμα 31).



Σχήμα 31: Ενημερώσεις Εφαρμογών και Παιχνιδιών

## Αναλυτικά στοιχεία (Analytics)

Κάνουμε την επιλογή “Βαθιά Γνώση” (Insights) για την εφαρμογή μας για να ανιχνεύσουμε το πώς οι χρήστες αλληλεπιδρούν με την εφαρμογή μας, για να φτιάξουμε την καλύτερη πιθανή εμπειρία για τους χρήστες μας. Για τους διαχειριστές της εφαρμογής, η επιλογή Insights περιλαμβάνει ανατροφοδότηση για ιστορικά ροής (stream story), αναφορική κίνηση της εφαρμογής μας, μία ανάλυση με το ποιες ενέργειες χρήστη συνεισφέρουν στον μετρητή ενεργού χρήστη, δημογραφικά στοιχεία των εξουσιοδοτημένων και των ενεργών χρηστών, και το πλήθος των στιγμών που οι άδειες έχουν υποβληθεί και παραχωρηθεί.

Επιπρόσθετα παρέχει διαγνωστικά εργαλεία για την εφαρμογή μας, ώστε να μπορούμε να ανιχνεύουμε τα API λάθη, όπως και να έχουμε πρόσβαση σε πληροφορίες κατανομής και αυξομείωσης κίνησης.

## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

Μπορούμε να έχουμε πρόσβαση στην επιλογή “Βαθιά Γνώση” της εφαρμογής μας απευθείας από το Ταμπλό Βαθιάς Γνώσης (Insights Dashboard) ή επιλέγοντας την εφαρμογή μας από το Ταμπλό Εφαρμογών και κάνοντας κλικ στον σύνδεσμο “Insights” (Σχήμα 32).



Σχήμα 32: Αναλυτικά στοιχεία

### **2.3 FACEBOOK ΓΙΑ ΙΣΤΟΣΕΛΙΔΕΣ.**

Η Πλατφόρμα Facebook μας δίνει τη δυνατότητα να κάνουμε την ιστοσελίδα μας πιο κοινωνική. Μπορούμε να χρησιμοποιήσουμε τα κοινωνικά πρόσθετα (social plugin), όπως το πλήκτρο “Μου αρέσει” (Like Button) για να ωθήσουμε τον χρήστη στη επιλογή της ιστοσελίδας μας με μία απλή γραμμή HTML. Τα πλήκτρα Εισόδου (Login) και Καταχώρησης (Registration) μας αφήνουν να απλοποιήσουμε ή να εξαλείψουμε την διαδικασία Καταχώρησης και Εισόδου του δικού μας χρήστη. Τελευταία, ο Γράφος API αφήνει την πρόσβασή μας σε όλο τον Κοινωνικό Γράφο ενός οποιουδήποτε χρήστη, επιτρέποντάς μας να δημιουργήσουμε μία πραγματικά βαθιά προσωπική εμπειρία.

Παρακάτω θα οδηγηθούμε μέσα στις βασικές δομές της δημιουργίας μιας διαδικτυακής εφαρμογής που επιρρεάζουν τα ανωτέρω στοιχεία. Τα παραδείγματα που αναφέρονται χρησιμοποιούν PHP για προγραμματισμό προσανατολισμένο στον εξυπηρετητή (server-side programming) και HTML/Javascript για κώδικα προσανατολισμένο στον πελάτη (client-side code). Αυτά τα παραδείγματα είναι ξεκάθαρα και εύκολα μεταφράσιμα σε άλλες γλώσσες.

Σε αυτή την παράγραφο αναφέρουμε τα εξής:

- Κοινωνικά Πρόσθετα (Social Plugin)
- Αυθεντικοποίηση (Authentication)
- Εξατομίκευση (Personalization)
- Αναλυτικά στοιχεία (Analytics)

#### **Κοινωνικά Πρόσθετα (Social Plugin)**

Τα Κοινωνικά Πρόσθετα είναι ο ευκολότερος τρόπος για να αρχίσουμε με την Πλατφόρμα του Facebook. Σε αυτά τα Πρόσθετα είναι ενσωματωμένα κοινωνικά στοιχεία τα οποία μπορούν να ενοποιηθούν με την ιστοσελίδα μας με μία γραμμή

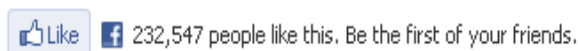
## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

κώδικα HTML. Επειδή φιλοξενούνται από το Facebook, αυτά τα πρόσθετα είναι εξατομικευμένα για όλους τους χρήστες που είναι συνδεδεμένοι στο Facebook, ακόμα κι αν επισκέπτονται την ιστοσελίδα μας για πρώτη φορά.

Το πιο σημαντικό Κοινωνικό Πρόσθετο είναι το πλήκτρο “Μου Αρέσει” (Like), το οποίο δίνει τη δυνατότητα στους χρήστες να μοιραστούν (share) τη σελίδα μας με άλλους φίλους με ένα κλικ. Μπορούμε να προσθέσουμε ένα πλήκτρο “Μου Αρέσει” σε οποιαδήποτε σελίδα με μία iframe ετικέτα (βλέπε Λεξιλόγιο):

```
<html>
  <head>
    <title>My Great Web page</title>
  </head>
  <body>
    <iframe src="http://www.facebook.com/plugins/like.php?href=YOUR_URL"
      scrolling="no" frameborder="0"
      style="border:none; width:450px; height:80px"></iframe>
  </body>
</html>
```

Υπάρχει ένα πλήθος από επιλογές για το πλήκτρο “Μου Αρέσει”, συμπεριλαμβάνοντας την επιλογή να περιλαμβάνει τα ονόματα και τις φωτογραφίες του προφίλ των φίλων των χρηστών που έχουν επίσης επιλέξει το “Μου Αρέσει” για την Σελίδα μας. Παρακάτω φαίνεται το γνωστό “Like” σε μια Σελίδα:

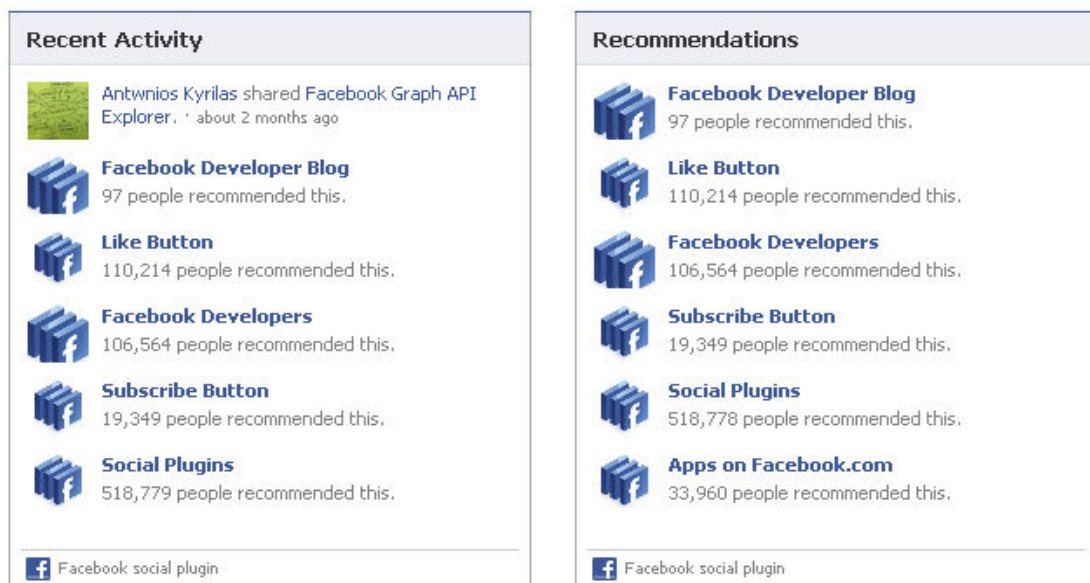


Από τη στιγμή που θα περιλάβουμε ένα πλήκτρο “Μου Αρέσει” στην ιστοσελίδα μας μπορούμε να χρησιμοποιήσουμε κι άλλα Κοινωνικά Πρόσθετα για να μετατρέψουμε την αλληλεπίδραση των χρηστών σε περισσότερο ελκυστική εμπειρία μέσα από τον ιστοχώρο μας.

Μπορούμε να χρησιμοποιήσουμε το “Πρόσθετο Ενημερώσεων



Δραστηριότητας” (Activity Feed Plugin) για να δείξουμε στους χρήστες μία δέσμη από πρόσφατα “Like” και σχόλια από τους φίλους τους στην ιστοσελίδα μας. Επίσης μπορούμε να χρησιμοποιήσουμε το “Πρόσθετο Συστάσεων” (Recommendations Plugin) για να δείξουμε εξατομικευμένες συστάσεις Σελίδας στους χρήστες μας, βασισμένες στα “Like” και στα σχόλια που αφορούν όλο τον ιστοχώρο μας. Στο Σχήμα 33 φαίνονται τα Πρόσθετα Δραστηριότητας και Συστάσεων για τον ιστοχώρο των Προγραμματιστών του Facebook (Facebook Developers' site):



Σχήμα 33: Κοινωνικά Πρόσθετα Δραστηριότητας και Συστάσεων

Τα περισσότερα Κοινωνικά Πρόσθετα μπορούν να ενοποιηθούν με τον ιστοχώρο μας απλά περιλαμβάνοντας μία ετικέτα “iframe” για το Πρόσθετο στη Σελίδα μας. Υπάρχουν διάφορα Κοινωνικά Πρόσθετα, όπως τα Σχόλια (Comments) και το Ζωντανό Κύμα (Live Stream), τα οποία απαιτούν τη χρήση XFBML (eXtended Facebook Markup Language). Η XFBML είναι ένα σύνολο από XML στοιχεία τα οποία μπορούν να περιληφθούν στις HTML σελίδες μας για να

## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

εμφανίσουν Κοινωνικά Πρόσθετα. Όταν φορτώνεται η σελίδα μας, οποιαδήποτε στοιχεία XFBML βρεθούν στο έγγραφο, θα τα επεξεργαστεί το JavaScript SDK, με αποτέλεσμα το κατάλληλο Πρόσθετο να εμφανιστεί κατάλληλα στη σελίδα μας.

Παρέχονται από το Facebook όλα τα XFBML στοιχεία για όλα τα Κοινωνικά Πρόσθετα. Για παράδειγμα, το πλήκτρο “Like” μπορεί επίσης να τεθεί στη σελίδα μας χρησιμοποιώντας XFBML ισοδύναμο:

```
<html>
  <head>
    <title>My Great Web page</title>
  </head>
  <body>
    <div id="fb-root"></div>
    <script>
      // Load the SDK Asynchronously
      (function(d){
        var js, id = 'facebook-jssdk', ref = d.getElementsByTagName('script')[0];
        if (d.getElementById(id)) {return;}
        js = d.createElement('script'); js.id = id; js.async = true;
        js.src = "//connect.facebook.net/en_US/all.js#xfbml=1";
        ref.parentNode.insertBefore(js, ref);
      })(document);
    </script>
    <div class="fb-like"></div>
  </body>
</html>
```

Οι εκδόσεις iframe των Πρόσθετων του Facebook είναι οι πιο ευρεία χρησιμοποιούμενες, αφού απαιτούν την ελάχιστη κατανόηση από την Πλατφόρμα Facebook. Οι εκδόσεις XFBML τυπικά χρησιμοποιούνται από τους πιο εξεζητημένους προγραμματιστές που ψάχνουν για περισσότερο έλεγχο και

Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

σταθερότητα στη βάση του κώδικά τους.

Απλά μπορούμε να αρχίσουμε ως εξής:

επιλέγουμε το Πρόσθετο από τη σελίδα των Κοινωνικών Πρόσθετων (Social Plugin page – <http://developers.facebook.com/docs/plugins>) και ακολουθούμε τα βήματα από τον παρεχόμενο διαμορφωτή. Αυτοί οι διαμορφωτές, όπως αυτός του Σχήματος 34, μας βοηθούν να εγκαταστήσουμε το πρόσθετό μας και παράγουν όλο τον κώδικα που χρειαζόμαστε για να προσθέσουμε στην ιστοσελίδα μας.



Σχήμα 34: Διαμορφωτής Κοινωνικού Πρόσθετου

## **Αυθεντικοποίηση (Authentication)**

Το Facebook μας βοηθάει να απλοποιήσουμε και να ενδυναμώσουμε την Εγγραφή του χρήστη και τη Σύνδεση (Login) χρησιμοποιώντας το Facebook ως το σύστημα εισόδου. Οι χρήστες δε χρειάζονται πλέον να συμπληρώνουν ακόμη μία φόρμα εγγραφής ή να θυμούνται άλλο ένα όνομα χρήστη και έναν κωδικό για να χρησιμοποιήσουν τον ιστοχώρο μας. Όσο ο χρήστης είναι συνδεδεμένος με το Facebook, τότε αυτόματα είναι επίσης συνδεδεμένος με την ιστοσελίδα μας. Χρησιμοποιώντας το Facebook για τη Σύνδεση (Login), μας παρέχεται όλη η πληροφορία που χρειαζόμαστε για να δημιουργήσουμε μία κοινωνική,

## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

εξατομικευμένη εμπειρία από τη στιγμή που ο χρήστης επισκεφθεί την ιστοσελίδα μας στον περιηγητή ιστοσελίδων του.

Η Πλατφόρμα Facebook χρησιμοποιεί το πρότυπο OAuth 2.0 για την αυθεντικοποίηση και την εξουσιοδότηση. Ενώ μπορούμε να προσθέσουμε την Σύνδεση (Login) στον ιστοχώρο μας χρησιμοποιώντας απευθείας το OAuth 2.0, το Ανοιχτού Κώδικα JavaScript SDK είναι ο πιο απλός τρόπος για να χρησιμοποιήσουμε το Facebook για σύνδεση.

Το JavaScript SDK απαιτεί να καταχωρήσουμε τον ιστοχώρο μας στο Facebook για να πάρουμε μία Ταυτότητα Εφαρμογής (App ID ή appId). Η appId είναι ένα μοναδικό αναγνωριστικό για τον ιστοχώρο μας, το οποίο κάνει σίγουρο ότι υπάρχει το σωστό επίπεδο ασφάλειας μεταξύ του χρήστη και του ιστοχώρου. Με σκοπό να χρησιμοποιήσουμε τις μεθόδους αυθεντικοποίησης, η εφαρμογή πρέπει να διαμορφωθεί με τον ορισμό ενός Χώρου Αρμοδιότητας (ή Τμήμα) Εφαρμογής (App Domain). Οι ρυθμίσεις της εφαρμογής μπορούν να αλλαχθούν στη σελίδα ρυθμίσεων (settings page). Το επόμενο παράδειγμα δείχνει το πώς φορτώνουμε τον κώδικα σε JavaScript SDK αφού πρώτα έχουμε λάβει μία Ταυτότητα Εφαρμογής:

```
<html>
  <head>
    <title>My Facebook Login Page</title>
  </head>
  <body>
    <div id="fb-root"></div>
    <script>
      window.fbAsyncInit = function() {
        FB.init({
          appId      : 'YOUR_APP_ID', // App ID
          channelUrl : '://WWW.YOUR_DOMAIN.COM/channel.html', // Channel File
          status     : true, // check login status
```

## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

```
        cookie    : true, // enable cookies to allow the server to access the session
        xfbml     : true // parse XFBML
    });
    // Additional initialization code here
};
// Load the SDK Asynchronously
(function(d){
    var js, id = 'facebook-jssdk', ref = d.getElementsByTagName('script')[0];
    if (d.getElementById(id)) {return;}
    js = d.createElement('script'); js.id = id; js.async = true;
    js.src = "//connect.facebook.net/en_US/all.js";
    ref.parentNode.insertBefore(js, ref);
})(document);
</script>
</body>
</html>
```

Αφού έχουμε φορτώσει τον κώδικα JavaScript SDK στη σελίδα μας και τον έχουμε αρχικοποιήσει με την Ταυτότητα Εφαρμογής, απλά προσθέτουμε το Πλήκτρο Σύνδεσης (Login Button) στη σελίδα μας χρησιμοποιώντας το στοιχείο `<div class="fb-login-button">` όπως φαίνεται παρακάτω:

```
<html>
  <head>
    <title>My Facebook Login Page</title>
  </head>
  <body>
    <div id="fb-root"></div>
    <script>
      window.fbAsyncInit = function() {
        FB.init({
          appId    : 'YOUR_APP_ID', // App ID
```

## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

```
channelUrl : '/WWW.YOUR_DOMAIN.COM/channel.html', // Channel File
status    : true, // check login status
cookie    : true, // enable cookies to allow the server to access the session
xfbml     : true // parse XFBML
});
};
// Load the SDK Asynchronously
(function(d){
    var js, id = 'facebook-jssdk', ref = d.getElementsByTagName('script')[0];
    if (d.getElementById(id)) {return;}
    js = d.createElement('script'); js.id = id; js.async = true;
    js.src = "//connect.facebook.net/en_US/all.js";
    ref.parentNode.insertBefore(js, ref);
})(document);
</script>
<div class="fb-login-button">Login with Facebook</div>
</body>
</html>
```

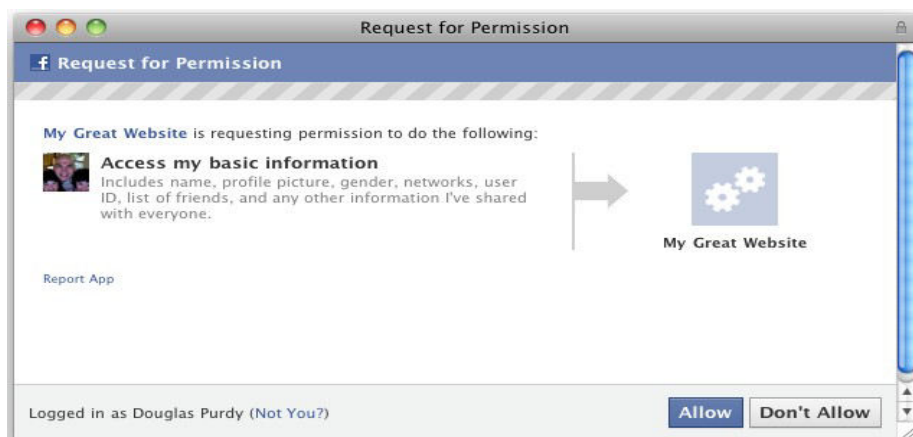
Όταν ο χρήστης φορτώσει τη σελίδα στον περιηγητή ιστοσελίδων του, το JavaScript SDK τοποθετεί το Πλήκτρο Σύνδεσης στη σελίδα μας. (Σχήμα 35)



Σχήμα 35: Πλήκτρο Σύνδεσης

Με σκοπό να συνδέσουμε (log) τον χρήστη στον ιστοχώρο μας, πρέπει να συμβούν τρία πράγματα. Πρώτα, το Facebook πρέπει να αυθεντικοποιήσει τον χρήστη. Αυτό εγγυάται ότι ο χρήστης είναι αυτός που ισχυρίζεται ότι είναι. Δεύτερον, το Facebook χρειάζεται να αυθεντικοποιήσει τον ιστοχώρο μας. Αυτό εγγυάται ότι ο χρήστης δίνει τις πληροφορίες του στον δικό μας ιστοχώρο και όχι σε κάποιον άλλο. Τέλος, ο χρήστης πρέπει σαφώς να εξουσιοδοτήσει τον ιστοχώρο μας για την πρόσβαση στις πληροφορίες του. Αυτό εξασφαλίζει ότι ο χρήστης γνωρίζει ακριβώς τι δεδομένα φανερώνει στον ιστοχώρο μας.

Αυτά τα βήματα μπορεί να φαίνονται σύνθετα, αλλά ευτυχώς το μόνο που χρειάζεται να κάνουμε είναι να χρησιμοποιήσουμε το Πλήκτρο Σύνδεσης (Login Button), και το μόνο που έχει να κάνει ο χρήστης είναι να κάνει κλικ επάνω του. Όταν ο χρήστης το κάνει, το JavaScript SDK θα εξασφαλίσει ότι ο χρήστης έχει συνδεθεί (signed into) στο Facebook και ότι η αίτηση έρχεται από τον ιστοχώρο μας. Στη συνέχεια παρουσιάζεται ένα πλαίσιο διαλόγου εξουσιοδότησης στον περιηγητή ιστοσελίδων ενός χρήστη ο οποίος παρέχει πληροφορίες στον ιστοχώρο μας και τα δεδομένα του χρήστη τα οποία αιτούμαστε. Εξαρχής, αυτό το πλαίσιο φαίνεται όπως στο παρακάτω σχήμα:



Σχήμα 36: Αίτηση για άδεια πρόσβασης βασικών πληροφοριών

## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

Αν ο χρήστης πατήσει το “Allow”, θα δοθεί στον ιστοτόπο μας από το Facebook πρόσβαση στις αιτούμενες πληροφορίες. Εάν ο χρήστης πατήσει το “Don't Allow”, τότε το πλαίσιο διαλόγου θα κλείσει και καμία πληροφορία δε θα είναι διαθέσιμη στον ιστοτόπο μας. Εξ' ορισμού, δίνεται πρόσβαση στο όνομα του χρήστη, τη φωτογραφία και σε κάθε άλλο δεδομένο που μοιράζεται με όλους στο Facebook.

Εάν χρειαζόμαστε πρόσβαση σε περισσότερες πληροφορίες, όπως η διεύθυνση του ηλεκτρονικού ταχυδρομείου του χρήστη, κ.α., θα πρέπει να αιτηθούμε άδειες πρόσβασης γι' αυτές τις πληροφορίες. Μπορούμε να το κάνουμε αυτό προσθέτοντας τις άδειες που χρειαζόμαστε στην ιδιότητα “scope” του Πλήκτρου Σύνδεσης, όπως παρακάτω:

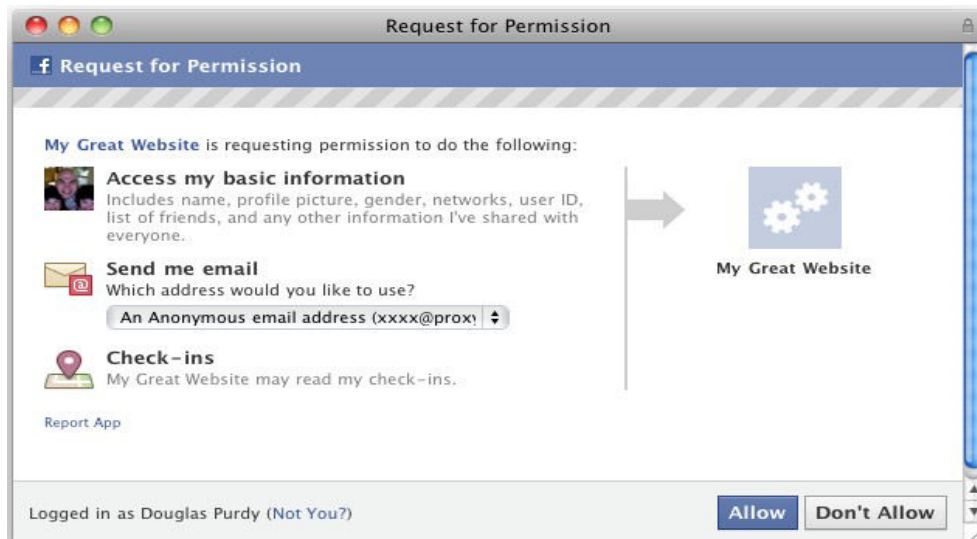
```
<html>
  <head>
    <title>My Facebook Login Page</title>
  </head>
  <body>
    <div id="fb-root"></div>
    <script>
      window.fbAsyncInit = function() {
        FB.init({
          appId      : 'YOUR_APP_ID', // App ID
          channelUrl : '://WWW.YOUR_DOMAIN.COM/channel.html', // Channel File
          status     : true, // check login status
          cookie     : true, // enable cookies to allow the server to access the session
          xfbml      : true // parse XFBML
        });
      };
      // Load the SDK Asynchronously
      (function(d){
        var js, id = 'facebook-jssdk', ref = d.getElementsByTagName('script')[0];
```



## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

```
if (d.getElementById(id)) {return;}  
js = d.createElement('script'); js.id = id; js.async = true;  
js.src = "//connect.facebook.net/en_US/all.js";  
ref.parentNode.insertBefore(js, ref);  
}(document));  
</script>  
<div class="fb-login-button" scope="email,user_checkins">  
  Login with Facebook  
</div>  
</body>  
</html>
```

Όταν ο χρήστης κάνει κλικ στο Πλήκτρο Σύνδεσης από αυτή την σελίδα, θα δει το παρακάτω πλαίσιο διαλόγου εξουσιοδότησης. Σημειώνουμε ότι υπάρχουν τώρα δύο νέα τμήματα τα οποία σκιαγραφούν τις επιπλέον πληροφορίες για τις οποίες ο ιστοτόπος μας αιτείται άδεια πρόσβασης (email και user\_checkins) (Σχήμα.37).



Σχήμα 37: Αίτηση άδειας πρόσβασης σε επιπλέον πληροφορίες

Πλήρης κατάλογος των αδειών πρόσβασης είναι διαθέσιμος στην αναφορά αδειών (<http://developers.facebook.com/docs/authentication/permissions>). Υπάρχει ένας δυνατός αντίστροφος συσχετισμός ανάμεσα στον αριθμό των αδειών που αιτείται ο ιστοχώρος μας και στον αριθμό των χρηστών που θα επικυρώσουν αυτές τις άδειες. Όσο μεγαλύτερος είναι ο αριθμός των αδειών που ζητούμε, τόσο μικρότερος είναι ο αριθμός των χρηστών οι οποίοι θα μας τις επικυρώσουν. Έτσι, συνίσταται να αιτούνται οι άδειες τις οποίες χρειαζόμαστε απαραίτητα για τον ιστοτόπο μας.

Εκτός από το Πλήκτρο Σύνδεσης, υπάρχει και το Πρόσθετο Καταχώρησης (Registration Plugin). Αυτό προσφέρει στοιχεία δύο περιπτώσεων, που δεν είναι διαθέσιμα με το Πλήκτρο Σύνδεσης: υποστηρίζεται για τους χρήστες χωρίς λογαριασμό στο Facebook και παρέχει την ικανότητα αίτησης επιπλέον πληροφοριών που το Facebook μπορεί να μην έχει ή να μην κάνει διαθέσιμες.

Η χρήση του Πρόσθετου Καταχώρησης είναι τόσο απλή όσο και του Πλήκτρου Σύνδεσης. Απλώς χρειάζεται να χρησιμοποιήσουμε το στοιχείο `<div class="fb-registration">` στη σελίδα μας, αντί του Πλήκτρου Σύνδεσης:

```
<html>
  <head>
    <title>My Facebook Login Page</title>
  </head>
  <body>
    <div id="fb-root"></div>
    <script>
      window.fbAsyncInit = function() {
        FB.init({
          appId      : 'YOUR_APP_ID', // App ID
          channelUrl : '://WWW.YOUR_DOMAIN.COM/channel.html', // Channel File
          status     : true, // check login status
          cookie     : true, // enable cookies to allow the server to access the session
          xfbml     : true // parse XFBML
        });
      };
    </script>
  </body>
</html>
```

## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

```
});  
};  
// Load the SDK Asynchronously  
(function(d){  
    var js, id = 'facebook-jssdk', ref = d.getElementsByTagName('script')[0];  
    if (d.getElementById(id)) {return;}  
    js = d.createElement('script'); js.id = id; js.async = true;  
    js.src = "//connect.facebook.net/en_US/all.js";  
    ref.parentNode.insertBefore(js, ref);  
})(document);  
</script>  
<div  
    class="fb-registration"  
    data-fields="{['name':'name'], ['name':'email'],  
        ['name':'favorite_car','description':'What is your favorite car?',  
            'type':'text']}"  
    data-redirect-uri="URL_TO_LOAD_AFTER_REGISTRATION"  
</div>  
</body>  
</html>
```

Αν ο χρήστης δεν είναι ακόμη συνδεδεμένος στο Facebook, θα δει την παρακάτω φόρμα (Σχήμα 38). Ας σημειώσουμε ότι η φόρμα δεν είναι προσυμπληρωμένη με καμία πληροφορία του χρήστη, αλλά υπάρχει ένας σύνδεσμος για να συνδεθούμε με το Facebook ώστε να προσυμπληρωθεί.



My Facebook Registration Page

Have a Facebook account? Log in to prefill the form below with your profile information.

Name:

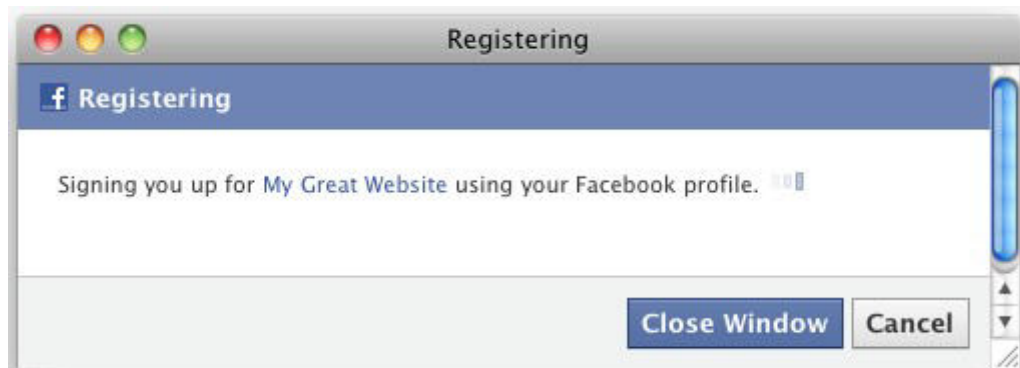
Email Address:

What is your favorite car?:

**Σχήμα 38: Φόρμα Καταχώρησης**

Εάν ο χρήστης είναι ήδη συνδεδεμένος στο Facebook ή επιλέξει να συνδεθεί από την Φόρμα Καταχώρησης, θα δει την φόρμα συμπληρωμένη με τα στοιχεία από το Facebook.

Αφού ο χρήστης κάνει μία ανασκόπηση των πληροφοριών και/ή μπει σε κάποιο στοιχείο όπως αυτό που αφορά το αγαπημένο του αυτοκίνητο της προηγούμενης εικόνας και το συμπληρώσει από μόνος του, ο χρήστης κάνει κλικ στο Καταχώρηση (Register). Αυτό θα ξεκινήσει τη διαδικασία αυθεντικοποίησης και εξουσιοδότησης η οποία φαίνεται από το επόμενο παράθυρο διαλόγου:



*Σχήμα 39: Μήνυμα Καταχώρησης*

Με την συμπλήρωση της αυθεντικοποίησης και εξουσιοδότησης, επανακατευθύνουμε τον χρήστη σε όποιο URL θέσαμε στην ιδιότητα *data-redirect-uri* του Πρόσθετου, το οποίο ολοκληρώνει τη διαδικασία καταχώρησης.

Το Πλήκτρο Σύνδεσης και το Πρόσθετο Καταχώρησης μας επιτρέπουν να φέρουμε εύκολα τους πολυπληθείς χρήστες του Facebook στον ιστοτόπο μας.

### **Εξατομίκευση (Personalization)**

Ενώ τα Κοινωνικά Πρόσθετα προσφέρουν έναν εύκολο τρόπο για να εξατομικεύσουμε τον ιστοχώρο μας, από τη στιγμή που έχουμε προσθέσει τη

## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

Σύνδεση σε αυτόν, μπορούμε να έχουμε πρόσβαση στην πλήρη ισχύ του Γράφου API για να δημιουργήσουμε την ακόμα πιο βαθιά εξατομικευμένη εμπειρία για τους χρήστες. Μπορούμε να χρησιμοποιήσουμε τον Γράφο API για να έχουμε πρόσβαση στο προφίλ του χρήστη του Facebook. Μπορούμε επίσης να χρησιμοποιήσουμε τον Γράφο API για να κάνουμε μία δημοσίευση στον Τοίχο ή στις Ενημερώσεις ενός χρήστη του Facebook ή ακόμη για να έχουμε πρόσβαση στον Κοινωνικό Γράφο του χρήστη και να φέρουμε τους φίλους του απευθείας στον ιστοχώρο μας, κι όλα αυτά καθαρά εξατομικευμένα.

Το JavaScript SDK παρέχει έναν απλό τρόπο για την πρόσβαση στον Γράφο API: το FB.api. Αυτή η συνάρτηση δέχεται μία συμβολοσειρά ως όρισμα, η οποία καθορίζει το τμήμα του Γράφου το οποίο στοχεύεται και περιέχει μία συνάρτηση που καλείται όταν η κλήση της ολοκληρώνεται. Ο ακόλουθος κώδικας δεικνύει τη χρήση του FB.api() για την ανάκτηση της εικόνας και του ονόματος του χρήστη από τον Γράφο API και την εμφάνισή τους στον ιστοχώρο μας:

```
<html>
  <head>
    <title>My Facebook Login Page</title>
  </head>
  <body>

    <div id="fb-root"></div>
    <script>
      window.fbAsyncInit = function() {
        FB.init({
          appId      : 'YOUR_APP_ID', // App ID
          channelUrl : '://WWW.YOUR_DOMAIN.COM/channel.html', // Channel File
          status     : true, // check login status
          cookie     : true, // enable cookies to allow the server to access the session
          xfbml      : true // parse XFBML
        });
      };
    </script>
  </body>
</html>
```

## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

```
FB.api('/me', function(user) {
  if (user) {
    var image = document.getElementById('image');
    image.src = 'http://graph.facebook.com/' + user.id + '/picture';
    var name = document.getElementById('name');
    name.innerHTML = user.name
  }
});
};
// Load the SDK Asynchronously
(function(d){
  var js, id = 'facebook-jssdk', ref = d.getElementsByTagName('script')[0];
  if (d.getElementById(id)) {return;}
  js = d.createElement('script'); js.id = id; js.async = true;
  js.src = "//connect.facebook.net/en_US/all.js";
  ref.parentNode.insertBefore(js, ref);
})(document);
</script>

<div align="center">
  <img id="image"/>
  <div id="name"></div>
</div>

</body>
</html>
```

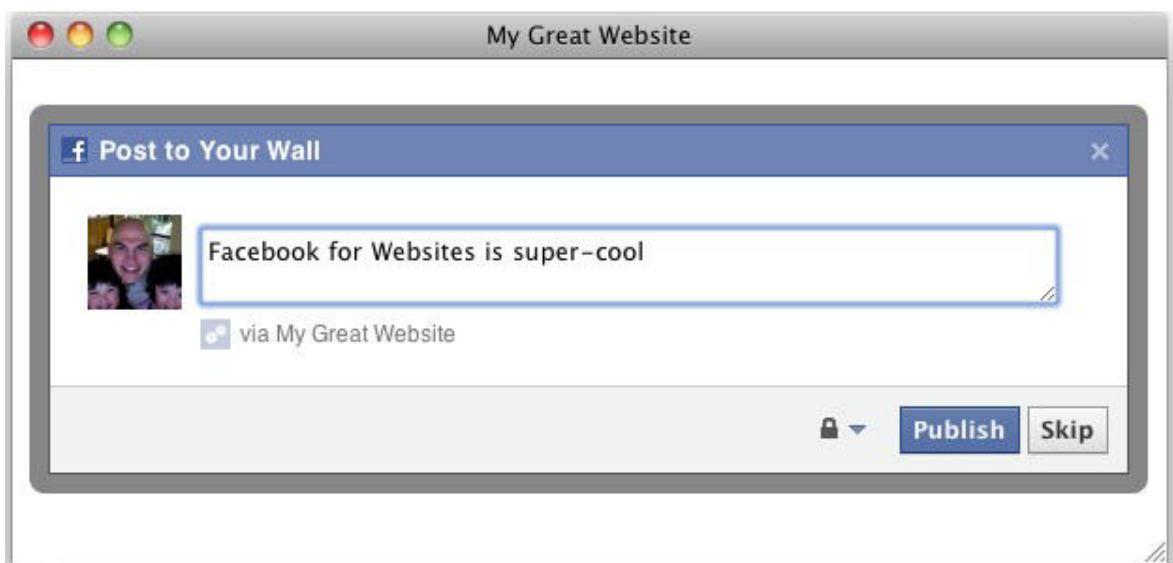
Άλλος ένας τρόπος για να εξατομικεύσουμε τον ιστοχώρο μας με το JavaScript SDK είναι η συνάρτηση FB.ui. Η συνάρτηση επικαλείται τους Διαλόγους Πλατφόρμας (Platform Dialogs) μέσα στο πλαίσιο του ιστοχώρου μας. Μπορούμε να κάνουμε χρήση της συνάρτησης FB.ui για να γίνει κάποια αποστολή περιεχομένου στις Ενημερώσεις των χρηστών ή να επιτρέψουμε σε χρήστες να

## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

καλέσουν νέους φίλους. Ο ακόλουθος κώδικας επιδεικνύει τη χρήση των Διαλόγων Πλατφόρμας από τον ιστοχώρο μας.

```
<html>
  <head>
    <title>My Facebook Login Page</title>
  </head>
  <body>
    <div id="fb-root"></div>
    <script>
      window.fbAsyncInit = function() {
        FB.init({
          appId      : 'YOUR_APP_ID', // App ID
          channelUrl : '://WWW.YOUR_DOMAIN.COM/channel.html', // Channel File
          status     : true, // check login status
          cookie     : true, // enable cookies to allow the server to access the session
          xfbml      : true // parse XFBML
        });
        FB.ui({ method: 'feed',
              message: 'Facebook for Websites is super-cool'});
      };
      // Load the SDK Asynchronously
      (function(d){
        var js, id = 'facebook-jssdk', ref = d.getElementsByTagName('script')[0];
        if (d.getElementById(id)) {return;}
        js = d.createElement('script'); js.id = id; js.async = true;
        js.src = "://connect.facebook.net/en_US/all.js";
        ref.parentNode.insertBefore(js, ref);
      })(document);
    </script>
  </body>
</html>
```

Όταν αυτή η σελίδα φορτώνεται στον περιηγητή ιστοσελίδων του χρήστη, το JavaScript SDK θα εμφανίσει το πλαίσιο διαλόγου του Σχήματος 40, το οποίο μπορεί να χρησιμοποιήσει ο χρήστης για να αποστείλει στον Τοίχο του. Μπορούμε να θέσουμε ένα πλήθος από προκαθορισμούς για το πλαίσιο διαλόγου, τους οποίους ο χρήστης μπορεί να τροποποιήσει ή να υπερπηδήσει πριν από την αποστολή.

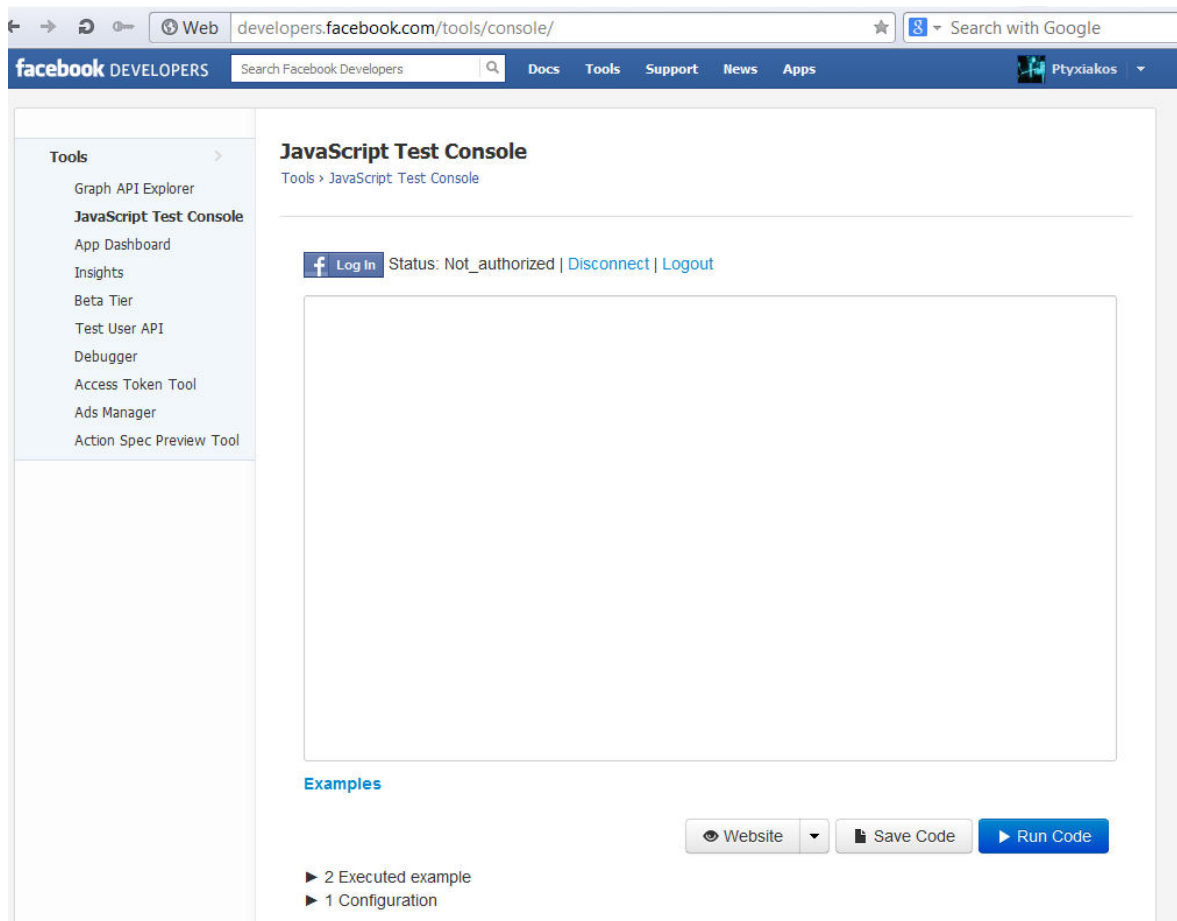


*Σχήμα 40: Αποστολή στον Τοίχο*

Ένας από τους καλύτερους τρόπους για να μάθει κανείς τι είναι δυνατόν με το JavaScript SDK είναι η κονσόλα JavaScript (JavaScript Console - <http://developers.facebook.com/tools/console/> - Σχήμα 41). Το εργαλείο αυτό προσφέρει ένα περιβάλλον στο οποίο μπορούμε να εκτελέσουμε απευθείας ό,τι θέλουμε πριν κάνουμε οποιαδήποτε αλλαγή στον ιστοχώρο μας.



## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού



*Σχήμα 41: Κονσόλα JavaScript*

Επίσης, παραδείγματα μπορούμε να βρούμε στη διεύθυνση: <http://www.fbrell.com/examples/>.

Το JavaScript SDK μας δίνει τη δυνατότητα να έχουμε πρόσβαση στον Γράφο API και στους Διαλόγους Πλατφόρμας από τον κώδικα της πλευράς του Πελάτη (client-side code) αλλά κάποια από τα πιο ενδιαφέροντα στοιχεία είναι η πρόσβαση στον Γράφο API από τον κώδικα από την πλευρά του Εξυπηρετητή (server-side code) που τρέχει στον δικό μας εξυπηρετητή. Το JavaScript SDK αποθηκεύει πληροφορίες για τον συνδεδεμένο χρήστη με τέτοιο τρόπο ώστε να υπάρχει πρόσβαση από το PHP SDK. Αυτό μας επιτρέπει να κάνουμε κλήσεις από

## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

την πλευρά του εξυπηρετητή στην Πλατφόρμα του Facebook χωρίς να χρειάζεται επιπλέον εργασία. Το ακόλουθο παράδειγμα σε PHP δείχνει το πώς να χρησιμοποιούμε το JavaScript SDK για σύνδεση και το PHP SDK για εξατομίκευση:

```
<?php

define('YOUR_APP_ID', 'YOUR APP ID');

//uses the PHP SDK. Download from https://github.com/facebook/php-sdk
require 'facebook.php';

$facebook = new Facebook(array(
    'appId' => YOUR_APP_ID,
    'secret' => 'YOUR APP SECRET',
));

$userId = $facebook->getUser();

?>

<html>
<body>
<div id="fb-root"></div>
<?php if ($userId) {
    $userInfo = $facebook->api('/' + $userId); ?>
    Welcome <?= $userInfo['name'] ?>
<?php } else { ?>
<fb:login-button></fb:login-button>
<?php } ?>
<div id="fb-root"></div>
<script>
```

## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

```
window.fbAsyncInit = function() {
  FB.init({
    appId      : 'YOUR_APP_ID', // App ID
    channelUrl : '//WWW.YOUR_DOMAIN.COM/channel.html', // Channel File
    status     : true, // check login status
    cookie     : true, // enable cookies to allow the server to access the session
    xfbml     : true // parse XFBML
  });
  FB.Event.subscribe('auth.login', function(response) {
    window.location.reload();
  });
};
// Load the SDK Asynchronously
(function(d){
  var js, id = 'facebook-jssdk', ref = d.getElementsByTagName('script')[0];
  if (d.getElementById(id)) {return;}
  js = d.createElement('script'); js.id = id; js.async = true;
  js.src = "//connect.facebook.net/en_US/all.js";
  ref.parentNode.insertBefore(js, ref);
})(document);
</script>

</body>
</html>
```

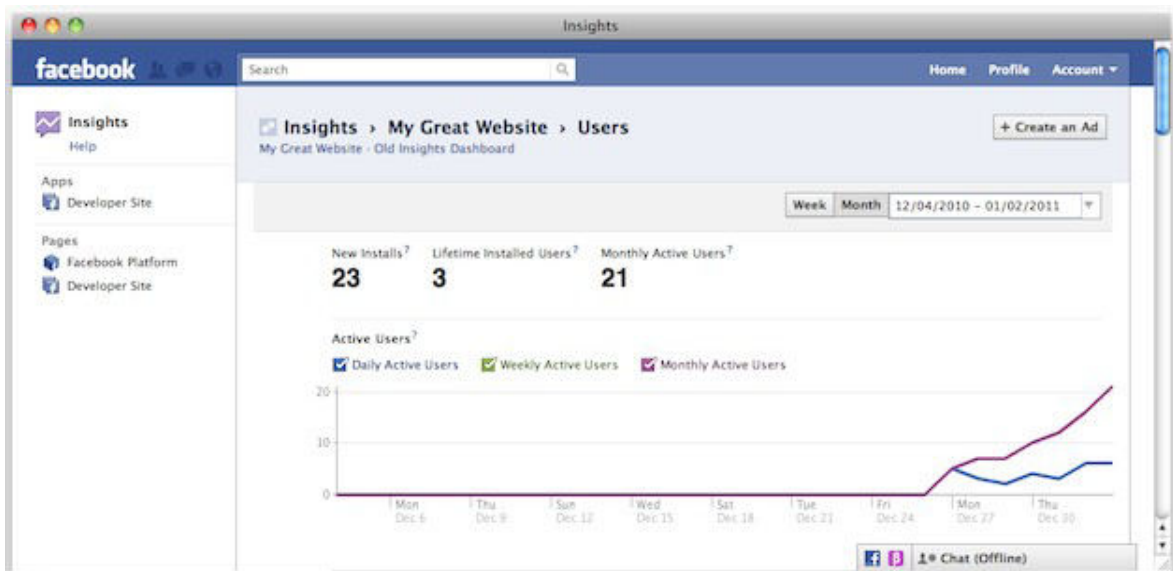
Η χρήση του JavaScript SDK και του PHP SDK ταυτόχρονα, είναι μόνο ένας τρόπος από τους πολλούς για να έχουμε πρόσβαση στα διαπιστευτήρια και τις πληροφορίες του χρήστη μέσω κώδικα από την πλευρά του εξυπηρετητή. Ο Οδηγός Αυθεντικοποίησης (Authentication Guide - <http://developers.facebook.com/docs/authentication/>) τονίζει το πως να πετύχουμε την αυθεντικοποίηση και την εξουσιοδότηση απευθείας από τον εξυπηρετητή μας, επιτρέποντάς μας να έχουμε πρόσβαση στον Γράφο API, χωρίς να χρησιμοποιούμε κώδικα από την πλευρά

Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

του πελάτη (client-side code).

## Αναλυτικά στοιχεία (Analytics)

Μπορούμε να λάβουμε αναλύσεις σχετικά με τα δημογραφικά στοιχεία των χρηστών από τον ιστοτόπο μας με την χρήση της Βαθιάς Γνώσης (“Insights”), όπως φαίνεται στο Σχήμα 42.



*Σχήμα 42: Βαθιά Γνώση (Insights)*

Η επιλογή “Insights” παρέχει αναφορές διαιρεμένες ανά τμήμα (domain) και ανά εφαρμογή. Αυτές οι αναφορές περιλαμβάνουν πλούσια δεδομένα σχετικά με τους χρήστες διαμοιράζοντας περιεχόμενο από τον ιστοχώρο μας μέσα στο Facebook και άλλες εφαρμογές ενεργοποιούμενες από το Facebook, άσχετα με το από που αυτή η δραστηριότητα προήλθε. Για παράδειγμα, αν ένας χρήστης βάλει μία διεύθυνση URL από τον ιστοχώρο μας σε ένα μήνυμα κατάστασης Facebook, τα δεδομένα θα περιληφθούν στα αναλυτικά στοιχεία για το domain μας.

## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

Τα δεδομένα αυτά από το Insights είναι επίσης διαθέσιμα στον Γράφο API έτσι ώστε να μπορούμε να ενοποιήσουμε τα αναλυτικά στοιχεία του Facebook με τα δικά μας - εντός του σπιτιού ή εργασίας - αναλυτικά συστήματα.

Εάν χρησιμοποιούμε ένα εξωτερικό αναλυτικό σύστημα, υπάρχουν μερικά βήματα που μπορούμε να κάνουμε για να επιβεβαιώσουμε τις κατάλληλες μετρήσεις για τις επισκέψεις (hit) στο τμήμα (domain) μας, που όμως απαιτούν πιο εξειδικευμένα βήματα.

## **ΕΠΙΛΟΓΟΣ**

Στο κεφάλαιο αυτό είδαμε όλη τη θεωρία σχετικά με το πώς τελικά θα χρησιμοποιήσουμε τις εσωτερικές δομές του Facebook ούτως ώστε να φτάσουμε στο σημείο να γνωρίζουμε την γλώσσα επικοινωνίας με το Facebook και να αποκτήσουμε τις δυνάμεις να δημιουργήσουμε μία εφαρμογή είτε για το Facebook είτε για τον ιστοτόπο μας.

Ας δούμε παρακάτω κάποιο πρακτικό παράδειγμα που θα μας βοηθήσει να κατανοήσουμε περισσότερο τα ανωτέρω.

## ΚΕΦΑΛΑΙΟ 3

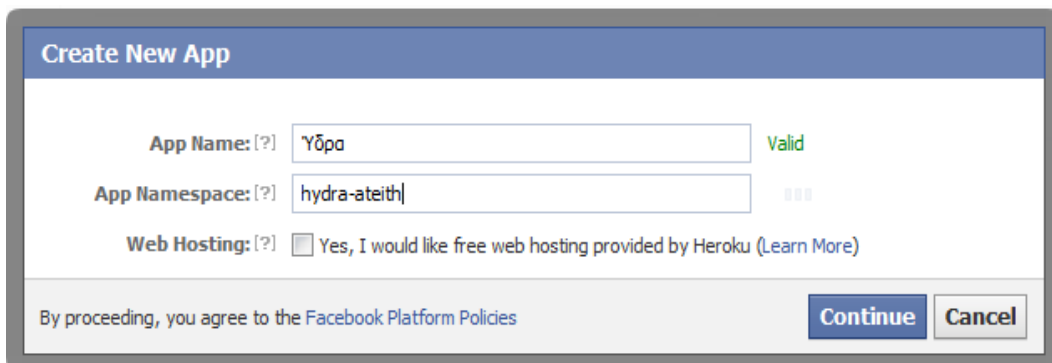
### ΠΡΑΚΤΙΚΗ ΕΦΑΡΜΟΓΗ ΤΗΣ ΘΕΩΡΙΑΣ-ΔΗΜΙΟΥΡΓΙΑ ΣΕΛΙΔΑΣ ΣΤΟ FACEBOOK

#### ***ΕΙΣΑΓΩΓΗ***

Εδώ θα παρουσιαστεί η δημιουργία μίας Εφαρμογής για το Facebook η οποία εμφανίζεται ως Σελίδα και το πώς μπορούμε να εμφανίσουμε μέσα σε αυτήν, την ιστοσελίδα του <http://hydra.it.teithe.gr/s/> - της ιστοσελίδας δηλαδή της Γραμματείας του Τμήματος Πληροφορικής του Α-ΤΕΙ Θεσσαλονίκης. Αποκτούμε έτσι τη γνώση του να μπορούμε να εμφανίζουμε με όλη τη λειτουργικότητα που περιέχει οποιοδήποτε περιεχομένο που αναφέρεται σε κάποιο σύνδεσμο στο διαδίκτυο.

### 3.1 ΔΗΜΙΟΥΡΓΙΑ ΣΕΛΙΔΑΣ ΣΤΟ FACEBOOK

Ξεκινούμε γράφοντας τη διεύθυνση <https://developers.facebook.com/apps>, αφού κάνουμε είσοδο στον προσωπικό λογαριασμό μας στο Facebook. Θα μας ζητηθεί να πληκτρολογήσουμε άλλη μία φορά τον κωδικό μας. Επιλέγουμε “Create New App” και μετά – εμφανίζεται το πλαίσιο διαλόγου του Σχήματος 43, όπου και πληκτρολογούμε τα στοιχεία της εφαρμογής μας. Το “App Name” δεν πρέπει να είναι μεγαλύτερο από 32 χαρακτήρες, ούτε μικρότερο από 3 και δε θα πρέπει να παραβαίνει τους κανόνες εμπορικού σήματος (trademark). Αυτό θα είναι εμφανιζόμενο το όνομα της εφαρμογής μας. Το “App Namespace” θα είναι το όνομα το οποίο θα συγχωνευτεί μετά με το <http://apps.facebook.com> και θα είναι η διεύθυνση στην οποία μπορούμε να δούμε την εφαρμογή μας.

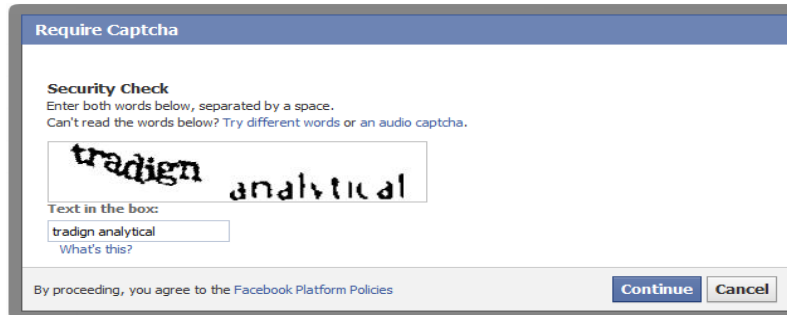


*Σχήμα 43: Δημιουργία Νέας Εφαρμογής*

Εάν δεν έχουμε καταχωρήσει τα προσωπικά μας στοιχεία (κινητό τηλέφωνο ή πιστωτική κάρτα), αυτά θα μας ζητηθούν για να συνεχίσουμε. Στη συνέχεια γίνεται ο τυπικός έλεγχος ασφάλειας μας για απαγορευτεί σε spammer να δημιουργήσουν ψεύτικους λογαριασμούς για την διεξαγωγή spam κίνησης. (Σχήματος44)

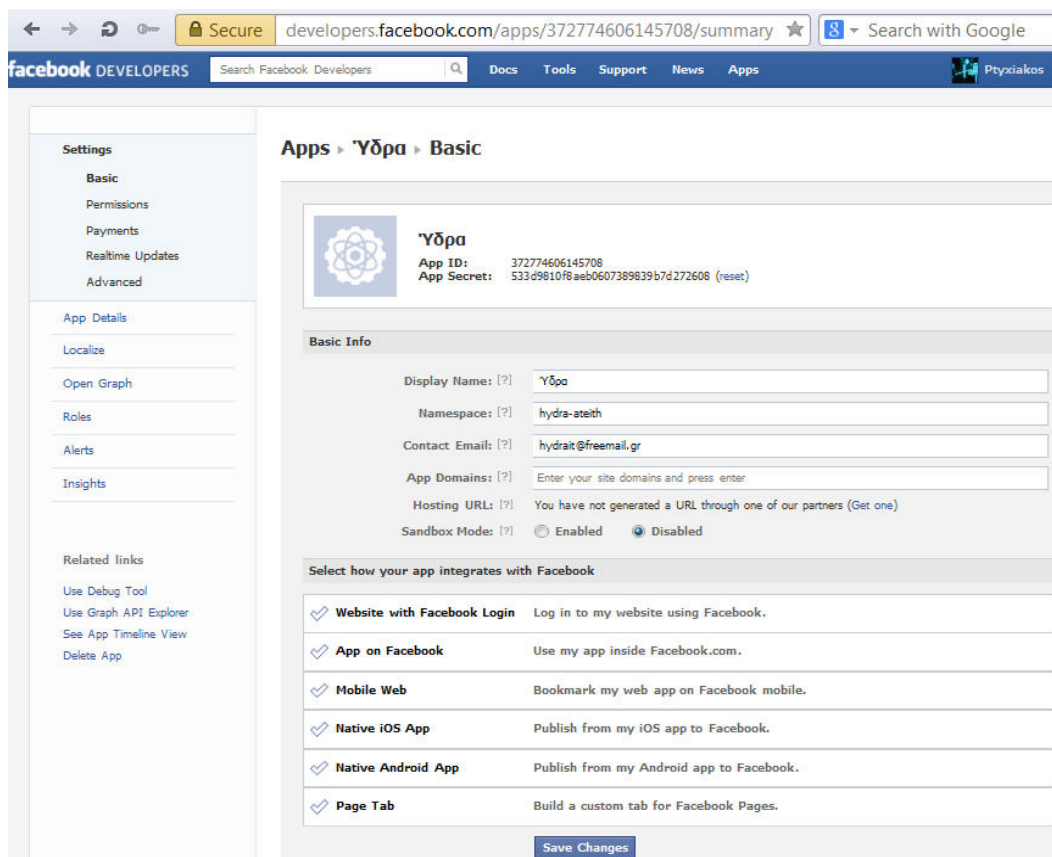


## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού



Σχήμα 44: Έλεγχος για spammer

Η εφαρμογή μας έχει καταχωρηθεί και είναι έτοιμη για τις περαιτέρω ρυθμίσεις. (Σχήμα 45)



Σχήμα 45: Η εφαρμογή έχει καταχωρηθεί

Αυτό που έχουμε να κάνουμε είναι να κάνουμε κλικ στο “App on Facebook” (Σχήμα 46).

The screenshot shows the Facebook App Settings interface. On the left is a navigation menu with sections: Settings (Basic, Permissions, Payments, Realtime Updates, Advanced), App Details, Localize, Open Graph, Roles, Alerts, Insights, and Related links (Use Debug Tool, Use Graph API Explorer, See App Timeline View, Delete App). The main content area is titled 'Apps > Υδρα > Basic'. It displays the app's logo, name 'Υδρα', App ID (372774606145708), and App Secret (533d9810f8aeb0607389839b7d272608). Below this is the 'Basic Info' section with fields for Display Name (Υδρα), Namespace (hydra-ateith), Contact Email (hydrat@freemail.gr), App Domains, Hosting URL, and Sandbox Mode (Disabled). The 'Select how your app integrates with Facebook' section is expanded for 'App on Facebook', showing fields for Canvas Page, Canvas URL, Secure Canvas URL, Canvas Width (Fixed at 760px), and Canvas Height (Fluid). Other integration options like 'Website with Facebook Login', 'Mobile Web', 'Native iOS App', 'Native Android App', and 'Page Tab' are also visible. A 'Save Changes' button is at the bottom.

Σχήμα 46: “App on Facebook”

Το πρώτο πεδίο μας δείχνει τη διεύθυνση στην οποία θα εμφανίζεται η


## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

εφαρμογή μας και δεν επιτρέπεται κάποια αλλαγή σε αυτό (το ορίσαμε παραπάνω). Έπειτα θα πρέπει να συμπληρώσουμε το πεδίο “Canvas URL”, το οποίο θέτει σε ένα iframe (βλέπε Λεξικό) το περιεχόμενο που αντιστοιχεί στη διεύθυνση που θέσαμε και εμφανίζει αυτό στην εφαρμογή μας. Ένα σημαντικό στοιχείο είναι επίσης το “Secure Canvas URL” το οποίο απαιτείται από την 01/10/2012 για λόγους που άπτονται της ασφάλειας της σύνδεσης. Για την δική μας εφαρμογή, συμπληρώνουμε και στα δύο πεδία τη διεύθυνση της σελίδας των ανακοινώσεων της Ύδρας. Τα πεδία “Canvas Width” και “Canvas Height” τα αφήνουμε ως έχουν και επιλέγουμε “Save Changes”.

Όπως μας δείχνει το μήνυμα της Σχήμα 47, οι αλλαγές μπορεί να χρειαστεί αρκετά λεπτά για να μεταδωθούν σε όλους τους εξυπηρετητές.

### Apps > Ύδρα > Basic

Changes saved. Note that your changes may take several minutes to propagate to all servers.



**Ύδρα**  
App ID: 372774606145708  
App Secret: 533d9810f8aeb0607389839b7d272608 (reset)

---

**Basic Info**

Display Name: [?]	Ύδρα
Namespace: [?]	hydra-ateith
Contact Email: [?]	hydrat@freemail.gr
App Domains: [?]	Enter your site domains and press enter
Hosting URL: [?]	You have not generated a URL through one of our partners (Get one)

Σχήμα 47: Οι αλλαγές θα πάρουν ίσως κάποια λεπτά για να μεταδωθούν

Η πρόσβαση στη σελίδα της εφαρμογής, δηλαδή στη διεύθυνση <https://apps.facebook.com/hydra-ateith/> είναι πλέον εφικτή. (Σχήμα 48)

## Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

The image shows a screenshot of a Facebook application page. The browser's address bar displays the URL <https://apps.facebook.com/hydra-ateith/>. The Facebook logo and a search bar are visible at the top. The page title is "Τμήμα Πληροφορικής - Φοιτητές". On the left, there is a navigation menu with the following items: Είσοδος, Διδασκαλίες, Ανακοινώσεις, and Αξιολόγηση. The main content area features a horizontal menu with the following categories: Νέα Τμήματος, Τεχνικά Θέματα, Erasmus, Διακηρύξεις - Προκηρύξεις, Μεταπτυχιακά, Ερευνητικά Προγράμματα, IEEE Student Branch, Άλλα δημόσια νέα, and Ανακοινώσεις Γραμματείας. Below this menu is a link: [Δημόσιος σύνδεσμος προς αυτόν τον πίνακα](#). A section titled "Ανακοινώσεις" contains a table with the following columns: Υποβλήθηκε από, Προστέθηκε, Τίτλος, Επισύναψη, and Προβολή. Below the table, the text "Δηλώσεις μαθημάτων" is visible.

*Σχήμα 48: Η εφαρμογή μας είναι έτοιμη*

## **ΕΠΙΛΟΓΟΣ**

Σε αυτό το κεφάλαιο είδαμε τον τρόπο με τον οποίο μπορεί να κατασκευαστεί μία εφαρμογή στο Facebook. Είδαμε τον τρόπο χρήσης του iFrame και το πώς μπορούμε να δούμε στη συνέχεια την εφαρμογή μας, ως μία απλή Σελίδα στο Facebook, η οποία όμως έχει όλη τη λειτουργικότητα της ιστοσελίδας την οποία θέτουμε στο iFrame.

## ΣΥΜΠΕΡΑΣΜΑΤΑ

Η εργασία αυτή αναφέρεται στην μελέτη του Facebook API. Αυτή η μελέτη μπορεί να αποβεί αρκετά χρήσιμη σε κάθε δυνητικό προγραμματιστή εφαρμογών που λειτουργούν στο Facebook ή και προγραμματιστή κατασκευής ιστοσελίδων οι οποίες χρησιμοποιούν με κάποιον τρόπο το Facebook. Μελετήθηκαν σε ένα ευρύ φάσμα οι γενικές δυνατότητες που υπάρχουν για να κάνει την αρχή ο οποιοσδήποτε φοιτητής ή επιστήμονας Πληροφορικής σε αυτόν τον τομέα της τεχνολογίας.

Αναδεικνύονται τόσο η εσωτερική δομή και λειτουργία του Facebook, όσο και οι προγραμματιστικές δυνατότητες που υπάρχουν σε αυτό.

Αρχικά ο σκοπός ήταν να δημιουργηθεί μία εφαρμογή για ένα προφίλ στο Facebook, το οποίο θα εμφάνιζε την σελίδα των ανακοινώσεων της Ύδρας. Αυτό όμως δεν κατέστη δυνατόν λόγω της απαλοιφής αυτής της δυνατότητας από τη Διαχείριση του Facebook, σύμφωνα με το <https://developers.facebook.com/blog/post/611/>.

Πλέον είναι δυνατόν να κατασκευάζονται εφαρμογές για την λειτουργία τους εντός του Facebook, μόνο για Σελίδες του Facebook. Έτσι, αυτή η εργασία μπορεί να χρησιμοποιηθεί στο μέλλον ως βάση για μία μελλοντική εργασία για την δημιουργία μιας εφαρμογής για Σελίδα του Facebook. Αυτό όμως μπορεί να γίνει προς το παρόν με δύο τρόπους:

είτε δίνοντας την “φιλοξενία” της εφαρμογής στους εξυπηρετητές της εταιρίας Heroku, η οποία είναι συμβεβλημένη με το Facebook (βλέπε Εικόνα 3.1), είτε η δημιουργία της θα πρέπει να γίνει σε εξυπηρετητή που θα βρίσκεται στην διάθεση του προγραμματιστή.

Εάν επιλεγεί η δεύτερη περίπτωση, τότε θα χρειαστεί να οριστεί μία Πολιτική Ασφάλειας για τον χώρο (domain) φιλοξενίας η οποία και αποτελεί και αναγκαία προϋπόθεση για τη δημιουργία της εφαρμογής.

Θα πρέπει να σημειώσουμε ότι η εφαρμογή μας λειτουργεί χωρίς να είναι

Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

υποχρεωμένος ο χρήστης να κάνει είσοδο στο Facebook. Οπότε σε κάποια μελλοντική εργασία θα μπορούσε να δημιουργηθεί εφαρμογή η οποία να απαιτεί είσοδο στο Facebook για την ελάχιστη ταυτοποίηση του χρήστη.

## ΑΝΑΦΟΡΕΣ

<https://developers.facebook.com>

<http://en.wikipedia.org>

<http://www.json.org>

<http://www.fbrell.com/examples/>



Πτυχιακή εργασία του φοιτητή Θωμά Στυλιανού

## ΒΙΒΛΙΟΓΡΑΦΙΑ

“Τα στέκια του δικτύου” Ειδικό τεύχος του περιοδικού RAM του τεύχους του Ιανουαρίου 2012 – Εκδόσεις “ΤΕΣΣΕΡΑ ΠΙ” (“4π”)

## ΠΑΡΑΡΤΗΜΑΤΑ

### ΛΕΞΙΛΟΓΙΟ

#### **Βαθμολογίες στο API (score API)**

Ο Γράφος του API για τις βαθμολογίες επιτρέπει στους δημιουργούς παιχνιδιών να δημοσιοποιούν τις βαθμολογίες των χρηστών αφού συγκριθούν με αυτές των φίλων τους για να παρέχουν πιο σχετικά Ιστορικά και με περισσότερο νόημα. Αυτό το API είναι διαθέσιμο μόνο στις εφαρμογές που κατηγοριοποιούνται ως Παιχνίδια. Μπορούμε να κατηγοριοποιήσουμε την εφαρμογή μας ως Παιχνίδι στο “Developer app” παίρνοντας τη διαδρομή About > Basic Info και επιλέγοντας “Games” στο αναδυόμενο μενού στο πεδίο της κατηγορίας.

#### **Επιτεύγματα στο API (Achievement API)**

Ο Γράφος API για τα επιτεύγματα επιτρέπει στους δημιουργούς παιχνιδιών να δημοσιοποιούν τα επιτεύγματα των χρηστών στο παιχνίδι τους για να παρέχουν πιο σχετικά Ιστορικά και με περισσότερο νόημα. Αυτό το API είναι διαθέσιμο μόνο σε εφαρμογές που κατηγοριοποιούνται ως Παιχνίδια. Μπορούμε να κατηγοριοποιήσουμε την εφαρμογή μας ως Παιχνίδι στο “Developer app” παίρνοντας τη διαδρομή About > Basic Info και επιλέγοντας “Games” στο αναδυόμενο μενού στο πεδίο της κατηγορίας.

Μία εφαρμογή μπορεί να ορίσει ένα σύνολο επιτευγμάτων για την εφαρμογή τους χρησιμοποιώντας τον τύπο αντικειμένου `game.achievement`.

- Κάθε επίτευγμα έχει πόντους συσχετιζόμενους με αυτό.
- Κάθε παιχνίδι λαβαίνει ένα σύνολο από 1000 πόντους για να διανείμει μεταξύ όλων των επιτευγμάτων του.
- Κάθε παιχνίδι λαβαίνει ένα μέγιστο από 1000 επιτεύγματα.
- Επιτεύγματα που είναι σπανιότερα και έχουν υψηλότερες τιμές πόντων θα λαβαίνουν περισσότερη κατανομή. Για παράδειγμα, επιτεύγματα που έχουν τιμές πόντων λιγότερο από 10 δε θα κατανέμονται σχεδόν

καθόλου. Οι εφαρμογές πρέπει να στοχεύουν μεταξύ των 50-100 επιτευγμάτων περιλαμβάνοντας ένα μείγμα των επιτευγμάτων με τιμές πόντων 50 (δύσκολο), 25 (μέτριο) και 10 (εύκολο).

- Ένας χρήστης μπορεί να πετύχει ένα συγκεκριμένο επίτευγμα για ένα παιχνίδι μόνο μία φορά.
- Είναι εναντίον της πολιτικής του Facebook να σβήνει ένα επίτευγμα, εκτός από την περίπτωση των δοκιμών στα επιτεύγματα.

### **JSON αντικείμενα**

Είναι συντομογραφία των λέξεων JavaScript Object Notation. Είναι ένα ελαφρύ και βασισμένο σε κείμενο Ανοιχτό Πρότυπο, σχεδιασμένο για την αναγνώσιμη από τον άνθρωπο ανταλλαγή δεδομένων. Προέρχεται από την Javascript γιατί αναπαριστά απλές δομές δεδομένων και συσχετιστικούς πίνακες, τα οποία καλούνται αντικείμενα. Παρά τη σχέση της με την Javascript, είναι ανεξάρτητη από Γλώσσες, με αναλυτές για πολλές Γλώσσες προγραμματισμού.

### **iFrame**

Προέρχεται από τις λέξεις "Inline Frame" (ευθύγραμμο πλαίσιο) και βασικά είναι ένα έγγραφο HTML εμφωλευμένο μέσα σε ένα άλλο έγγραφο HTML ενός ιστοχώρου. Ένα iFrame μεταφέρει το περιεχόμενο ενός ιστοχώρου μέσα σε έναν άλλο. Έτσι, στο πλαίσιο του Facebook, ένα iFrame μεταφέρει (ή καλύτερα έλκει) το περιεχόμενο ενός ιστοχώρου σε μία περιοχή μέσα σε μία Σελίδα του Facebook.

Τα iFrame είναι αρκετά ισχυρά διότι οτιδήποτε μπορούμε να δημιουργήσουμε σε έναν ιστοχώρο, μπορούμε να το μεταφέρουμε στη Σελίδα μας, στο Facebook, παρέχοντας μία μοναδική και πλούσια εμπειρία στην κοινότητά μας.