

Ανάπτυξη εργαλείου αυτόματης ενημέρωσης λογαριασμού στο Facebook

Γκέγκα Ευρώπη
egkegka@it.teithe.gr
Κωστοπούλου Ειρήνη
eirkost@it.teithe.gr

Facebook

- ▶ 2^ο σε επισκεψιμότητα των χρηστών στο web καθημερινά
- ▶ Κοινωνικό δίκτυο με τους περισσότερους χρήστες
- ▶ 1^ο σε προτίμηση των φοιτητών
- ▶ 1^ο σε δημοσίευση εκπαιδευτικών θεμάτων

Σπουδαιότητα της εργασίας

- ▶ Αυτόματη ενημέρωση για τα δημόσια νέα μέσω του Facebook
- ▶ Απλοποιήθηκε η διαδικασία
- ▶ Συλλέγει μη μοντελοποιημένο περιεχόμενο, το οποίο μετατρέπει κατάλληλα και το δημοσιεύει μέσω του API στο Facebook

Θεωρητικό Υπόβαθρο (1 / 2)

- ▶ Για ανάκτηση και ανάρτηση περιεχομένου στον ιστό, χρησιμοποιούνται οι τεχνολογίες push και pull

- ▶ Pull



- ▶ Push



Pull

- ▶ Μοντέλο που ακολουθείται κατά κύριο λόγο
- ▶ Ο χρήστης κάνει αίτηση για λήψη περιεχομένου και ο διακομιστής αποστέλλει το περιεχόμενο
- ▶ Ο διακομιστής, όσο δεν του αποστέλλονται αιτήματα, παραμένει αδρανής
- ▶ Ο πελάτης είναι αυτός που ασκεί τον έλεγχο
- ▶ Αυξημένη κίνηση (traffic) στο δίκτυο

Push

- ▶ Αποστολή περιεχομένου σε ένα χρήστη, χωρίς ο ίδιος να το έχει ζητήσει
- ▶ Ο διακομιστής έχει τον έλεγχο
- ▶ Οι πελάτες δεν χρειάζεται να κάνουν κάτι.
- ▶ Πιθανότητα καθυστέρησης λόγω απασχολημένου διακομιστή

Θεωρητικό Υπόβαθρο (2/2)

- ▶ Οι υπηρεσίες επικοινωνίας σε πραγματικό χρόνο (Real-time Web Applications):
 - AJAX Polling
 - AJAX Long-Polling
 - HTML5 Server Sent Events
 - HTML5 Websockets.

AJAX Polling

- ▶ Ο πελάτης αιτείται μία ιστοσελίδα
- ▶ Ο διακομιστής απαντά
- ▶ Στη συνέχεια, μέσω περιοδικών AJAX αιτημάτων, ο πελάτης αιτείται δεδομένα από το διακομιστή
- ▶ Ο διακομιστής απαντά και μπορεί να διοχετεύσει νέα δεδομένα στο χρήστη

AJAX Long-Polling

- ▶ Απαιτεί από τον web server να κρατήσει τη σύνδεση πελάτη - διακομιστή ανοιχτή
- ▶ Πραγματοποιούνται μεγαλύτερα αιτήματα στον διακομιστή, έτσι ώστε όταν ο διακομιστής έχει μια ενημέρωση, να την στείλει στον πελάτη.
- ▶ Χρησιμοποιείται σε υπηρεσίες email

HTML5 Server Sent Events

- ▶ Αυτόματη ανανέωση ιστοσελίδας από τον διακομιστή
- ▶ Οι διακομιστές μπορούν να ξεκινήσουν την μετάδοση δεδομένων, εφόσον η αρχική σύνδεση μεταξύ διακομιστή και πελάτη έχει συσταθεί.
- ▶ Η σύνδεση είναι μόνιμη.
- ▶ Αποστολή μηνυμάτων ενημέρωσης ή συνεχών ροών δεδομένων στον browser ενός πελάτη
- ▶ Παραδείγματα εφαρμογών: ενημερώσεις σε Facebook και Twitter

HTML5 Websockets

- ▶ Αμφίδρομη σύνδεση με διακομιστή
- ▶ Ανεξάρτητο πρωτόκολλο που βασίζεται στο TCP
- ▶ Δημιουργείται μία TCP σύνδεση με τον διακομιστή, η οποία μένει ανοιχτή για όσο χρονικό διάστημα χρειάζεται
- ▶ Ο διακομιστής αποστέλλει περιεχόμενο στο browser χωρίς να έχει ζητηθεί από τον πελάτη και επιτρέπει στα μηνύματα να διακινούνται
- ▶ Αποτελεσματική για εφαρμογές που απαιτούν συχνά μηνύματα (live streaming και παιχνίδια)

Σύγκριση

	Long-polling	Server-Sent Events	WebSockets
Υποστήριξη από Browser	Υποστηρίζεται από τους περισσότερους browsers που χρησιμοποιούνται σήμερα	Υποστηρίζεται από τους Chrome 9+, Firefox 6+, Opera 11+, Safari 5+	Το πιο πρόσφατο hybi-10 πρωτόκολλο υποστηρίζεται από τον Chrome 14 και τον Firefox 7, ενώ το hybi-07 υποστηρίζεται από τον Firefox 6
Server-loading	Καταλαμβάνει λίγους πόρους της CPU, αλλά δημιουργεί αδρανείς διαδικασίες ανά χρήστη, δαπανώντας έτσι μνήμη στο διακομιστή	Δουλεύει κατά ένα ποσοστό σαν long polling, εκτός και αν το SSE απαιτεί να μην κλείνει η σύνδεση κάθε φορά που μια απάντηση αποστέλλεται	Η καλύτερη δυνατή λύση. Ο server έχει τη μοναδική μέθοδο που εξυπηρετεί οποιοδήποτε αίτημα. Χωρίς βρόχους επανάληψης, δαπάνη μνήμης / CPU ανά πελάτη, αλλά και ανά συναλλαγή του πελάτη
Client-loading	Εξαρτάται από την υλοποίηση, ωστόσο πάντα είναι ασύγχρονη διαδικασία.	Υλοποιείται εγγενώς στον browser, καταλαμβάνει ελάχιστους πόρους	Υλοποιείται εγγενώς στον browser, καταλαμβάνει ελάχιστους πόρους
Χρονικό πλαίσιο	Σχεδόν σε πραγματικό χρόνο. Υπάρχει ένα κενό διάστημα μεταξύ του αιτήματος και της απάντησης που μπορεί να προσθέσει μία μικρή καθυστέρηση στην εκτέλεση του επόμενου αιτήματος.	Υπάρχει μία προκαθορισμένη καθυστέρηση 3 δευτερόλεπτα, μπορεί όμως να μεταβληθεί	Σε πραγματικό χρόνο
Πολυπλοκότητα υλοποίησης	Υλοποιείται πολύ εύκολα	Υλοποιείται ακόμη ευκολότερα	Απαιτεί EventMachine διακομιστή και μία ειδική θύρα να είναι ανοιχτή.

Εφαρμογές αυτόματης ανάρτησης

▶ Mail

- Ο χρήστης κάνει subscribe σε μια σελίδα ώστε να λαμβάνει ένα mail κάθε φορά που δημοσιεύεται καινούριο περιεχόμενο

▶ RSS Feed

- Λαμβάνει ένα feed, δηλαδή μια σειρά από τίτλους ακολοθούμενους από σύντομες περιγραφές καινούριου περιεχομένου που έχει δημοσιευτεί

▶ Atom

- Μοιάζει πολύ με το RSS, αλλά είναι ανοιχτό πρότυπο

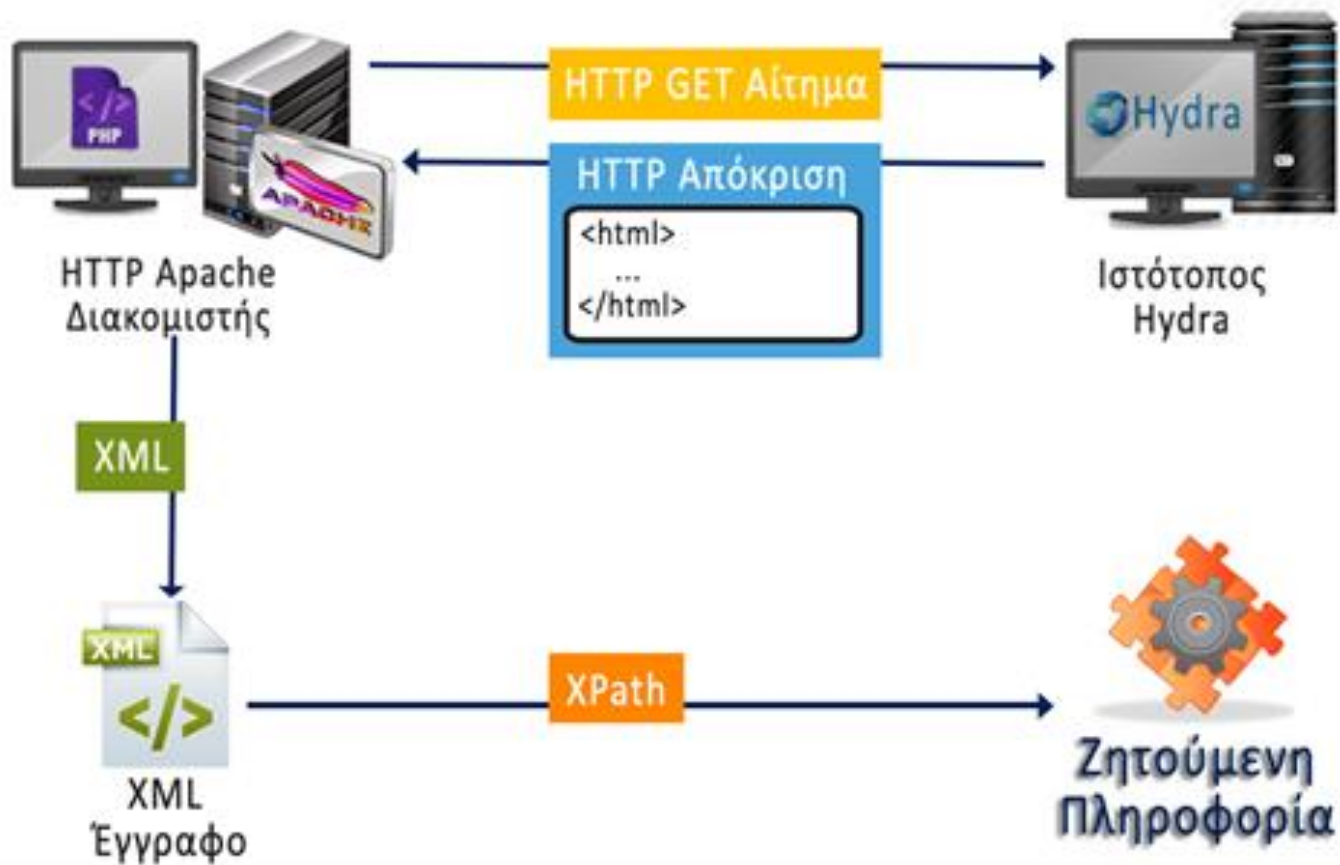
Υφιστάμενες λύσεις

- ▶ Argyle Social
- ▶ Buffer
- ▶ HootSuite
- ▶ Sprout Social
- ▶ Πηγή πληροφορίας: RSS feeds, κείμενο που εισάγει ο χρήστης μη αυτόματα
- ▶ Για δημοσίευση στο Facebook χρησιμοποιούν το Graph API του
- ▶ Επί πληρωμή
- ▶ Μη επεκτάσιμες

Ανάλυση Απαιτήσεων

- ▶ Για την ανάπτυξη του εργαλείου μας, είχαμε να ασχοληθούμε με δύο κύρια μέρη, τα οποία είναι:
 - πρώτον, η ανάκτηση και διαμόρφωση της ζητούμενης πληροφορίας και
 - δεύτερον, η δημοσίευση της στη σελίδα του Facebook.

Ανάκτηση και διαμόρφωση πληροφορίας



Δημοσίευση πληροφορίας στο Facebook

- ▶ Δημιουργία προφίλ διαχειριστή
- ▶ Δημιουργία σελίδας
- ▶ Δημιουργία εφαρμογής *hydranewsfeed*
- ▶ Αυθεντικοποίηση εφαρμογής
 - Απόκτηση `access_token` της σελίδας

Graph API

- ▶ Αποτελεί τον πυρήνα κάθε Facebook εφαρμογής
- ▶ Κάθε αντικείμενο έχει ένα μοναδικό id
- ▶ Επιτρέπει εισαγωγή και εξαγωγή δεδομένων από και προς το social graph του Facebook
- ▶ 3 βασικοί μέθοδοι: ανάκτηση, δημοσίευση και διαγραφή δεδομένων

Ανάκτηση δεδομένων

- ▶ Ανάκτηση πληροφορίας κάνοντας HTTP Get αιτήματα
- ▶ Κλήσεις στο API με ένα διακριτικό πρόσβασης
- ▶ Κόμβος /me δείχνει στο page_id

Δημοσίευση δεδομένων

- ▶ Γίνεται προς τον επιθυμητό κόμβο μέσω HTTP Post αιτημάτων στη σχετική σύνδεση
- ▶ `page_id/feed`
- ▶ Υποχρεωτική υπογραφή με ένα διακριτικό πρόσβασης (access token)

Ροές του προγράμματός μας (1 / 3)

1. Ξεκινάει το session και ανάγνωση μεταβλητής από προηγούμενη εκτέλεση.
2. Ανάκτηση περιεχομένων της ύδρας υπό μορφή HTML.
3. Αποθήκευση στο αρχείο data.xml τα περιεχόμενα της ύδρας σε μορφή XML.
4. Φόρτωμα των δεδομένων του αρχείου data.xml υπό μορφή SimpleXMLElement.

Ροές του προγράμματός μας

(2/3)

5. Χρήση της XPath έτσι ώστε να απομονώσουμε τα κομμάτια του αρχείου data.xml που επιθυμούμε. Όλα τα divs με γνώρισμα class='new'.
6. Αντιστροφή του αποτελέσματος και αποθήκευση σε ένα πίνακα news.
7. Ανάγνωση κάθε στοιχείου του πίνακα και αποθήκευση του αριθμού των παιδιών του. Αν είναι τρία, η ανακοίνωση δεν περιέχει συνημμένο και αποθηκεύεται σε μια μεταβλητή. Αν έχει τέσσερα, η ανακοίνωση έχει συνημμένο και αποθηκεύεται η επιθυμητή πληροφορία σε μια μεταβλητή.

Ροές του προγράμματός μας (3 / 3)

8. Έλεγχος για το αν η ανακοίνωση που είναι για δημοσίευση, είναι ήδη δημοσιευμένη.
9. Σύνδεση στο Facebook με τα απαραίτητα ζεύγη αναγνωριστικών.
10. Δημιουργία πίνακα με το μήνυμα που θα δημοσιευτεί (τελευταία/τελευταίες ανακοίνωση) και με το access token.
11. Κλήση στο API του Facebook και δημοσίευση του μηνύματος στο κοινωνικό δίκτυο.
12. Αποθήκευση μεταβλητής στο session για επόμενη χρήση.

Μελλοντικές επεκτάσεις

- ▶ Ως προς την πηγή που λαμβάνει τα δεδομένα
- ▶ Ως προς την επεξεργασία τους
- ▶ Ως προς το κοινωνικό μέσο στο οποίο δημοσιεύονται

Ερωτήσεις

Ευχαριστούμε για
την προσοχή σας!